

METHODS/STATA MANUAL FOR SCHOOL OF PUBLIC  
POLICY

OREGON STATE UNIVERSITY

SOC 516

Alison Johnston

Version 2.1

© *Johnston, A. 2013*

This manual provides an overview of statistical concepts learned in SOC 516. It also provides tutorials for the regression software program STATA, which you will use in the course. The approach used within this manual is an applied, rather than a theoretical one: exploration into STATA with the provided datasets is encouraged! The only request we make is that you record your work, so you are able to re-create your output on alternative datasets.

I owe a huge debt of gratitude to Dwaine Plaza, Michael Nash, and *especially* Brent Steel for providing datasets which are featured within this manual. Brett Burkhardt co-wrote the chapter on count models with me, and I am very appreciative on his help with the lesson and for providing the data. Carol Tremblay, Elizabeth Schroeder, Dan Stone, and Todd Pugatch offered invaluable comments and clarifications for the concepts discussed within this manual. Roger Hammer and my SOC 516 students also provided valuable feedback on how to improve the flow of the lessons, while Marie Anselm, Daniel Hauser, and Joanna Carroll provided valuable editing assistance. Any errors within this manual are my sole responsibility and should not be implicated with anyone above.

## Table of Contents

Pre-lab 1: How to log into STATA via Umbrella .....	5
Pre-lab 2: Loading Datasets into STATA and Saving Records of Work .....	9
Practice Problems.....	21
Lesson 1: Samples and Populations .....	22
1.1 STATA Lab Lesson 1 .....	25
1.2 Practice Problems .....	35
Lesson 2: Descriptive Statistics .....	36
2.1 STATA Lab Lesson 2 .....	43
2.2 Practice Problems .....	50
Lesson 3: Cross-tabulations .....	51
3.1 STATA Lab Lesson 3 .....	54
3.2 Practice Problems .....	61
Lesson 4: Significance Testing .....	62
4.1 STATA Lab Lesson 4 .....	68
4.2 Practice Problems .....	80
Lesson 5: Difference-in-Means Testing for Independent Groups .....	81
5.1 STATA Lab Lesson 5 .....	85
5.2 Practice Problems .....	91
Lesson 6: Univariate (OLS) Regression Analysis .....	92
6.1 STATA Lab Lesson 6 .....	92
6.2 Practice Problems .....	102
Lesson 7: Multivariate (OLS) Regression Analysis .....	103
7.1 STATA Lab Lesson 7 .....	103
7.2 Practice Problems .....	112
Lesson 8: Constants, Dummy Variables, Interaction Terms, and Non-Linear Variables in Multivariate OLS Regressions .....	113
8.1 STATA Lab Lesson 8 .....	113
8.2 Practice Problems .....	127

Lesson 9: Omitted Variable Biases, Irrelevant Variables, Outliers and Influential Cases in OLS.....	128
9.1 STATA Lab Lesson 9 .....	128
9.2 Practice Problems .....	141
Lesson 10: Multicollinearity and Heteroskedasticity .....	142
10.1 STATA Lab Lesson 10 .....	142
10.2 Practice Problems .....	153
Lesson 11: Logistic Regression Analysis .....	154
11.1 STATA Lab Lesson 11 .....	154
11.2 Practice Problems .....	164
Lesson 12: Model Specification for Logistic Regression Analysis.....	165
12.1 STATA Lab Lesson 12 .....	165
12.2 Practice Problems .....	179
Lesson 13: Ordinal Logistic Regression Analysis.....	180
13.1 STATA Lab Lesson 13 .....	180
13.2 Practice Problems .....	198
Lesson 14: Multinomial Logistic Regression Analysis.....	199
14.1 STATA Lab Lesson 14 .....	199
14.2 Practice Problems .....	222
Lesson 15: Counts Modeling (Poisson and Negative Binomial Regression).....	223
15.1 STATA Lab Lesson 15 .....	223
15.2 Practice Problems .....	239
Appendix I: Helpful Commands for Data Cleaning/Management .....	240
A.I Practice Problems .....	259
Appendix II: Useful Links .....	261

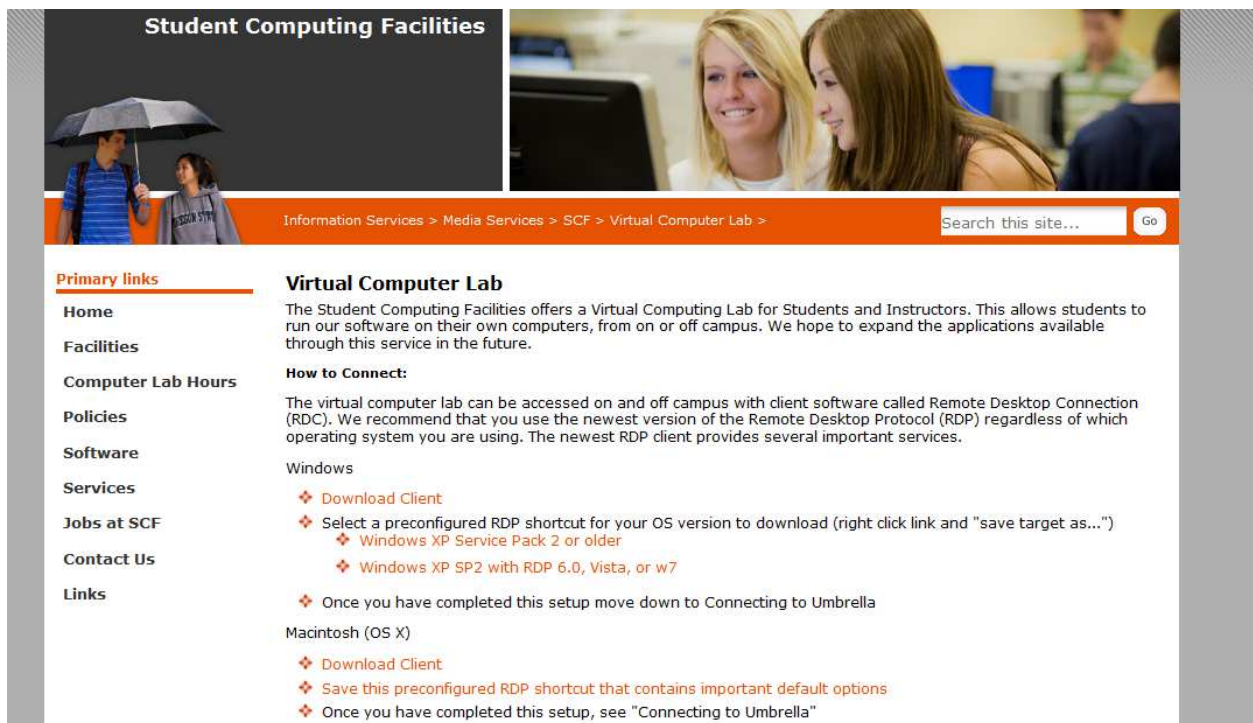
## Pre-Lab 1: How to log into STATA via Umbrella

---

While statistical programs are not available on some computer labs on campus, all programs which OSU has licenses to can be accessed via Umbrella (i.e. “Client” which enables Remote Desktop Connection). What is convenient about Umbrella is that it not only enables you to access statistical programs from computers on campus, but also from any computer off campus. In order to log onto Umbrella you need to go to the following site:

<http://oregonstate.edu/is/mediaservices/scf/virtual-lab>

This will bring up the Oregon State University Virtual Computer lab. You should see the following page below:



If you are on a campus computer, you should already have Remote Desktop Connection.

- For PCs:
  - Go to “Start”
  - Then go to “All Programs”
  - Next go to “Accessories”, and click on Remote Desktop Connection will be in the “Accessories” folder. If you have Windows XP, you may be prompted to “Download Client” but will not need to as the program should already exist within XP. However, if you cannot find it, you can always download it again.

- For Macs: If you are on a MAC Remote Desktop Connection is not in the “Accessories” folder, it should be in the “Communications” folder, which lies in the “Accessories” folder.

In order to “Download Client”, click on one of the “Download Client” that applies to your operating system (i.e. Windows or Mac OS). If you click on the Windows version, the following window should appear:

Remote Desktop in Windows XP Professional provides remote access to the desktop of your computer running Windows XP Professional, from a computer at another location. Using Remote Desktop you can, for example, connect to your office computer from home and access all your applications, files, and network resources as though you were in front of your computer at the office.

### Quick details

<b>Version:</b>	5.1.2600.2180	<b>Date Published:</b>	7/26/2007
<b>Change Language:</b>	English ▼		

File Name	Size	
MSRDPCLL.EXE	3.0 MB	<b>DOWNLOAD</b>

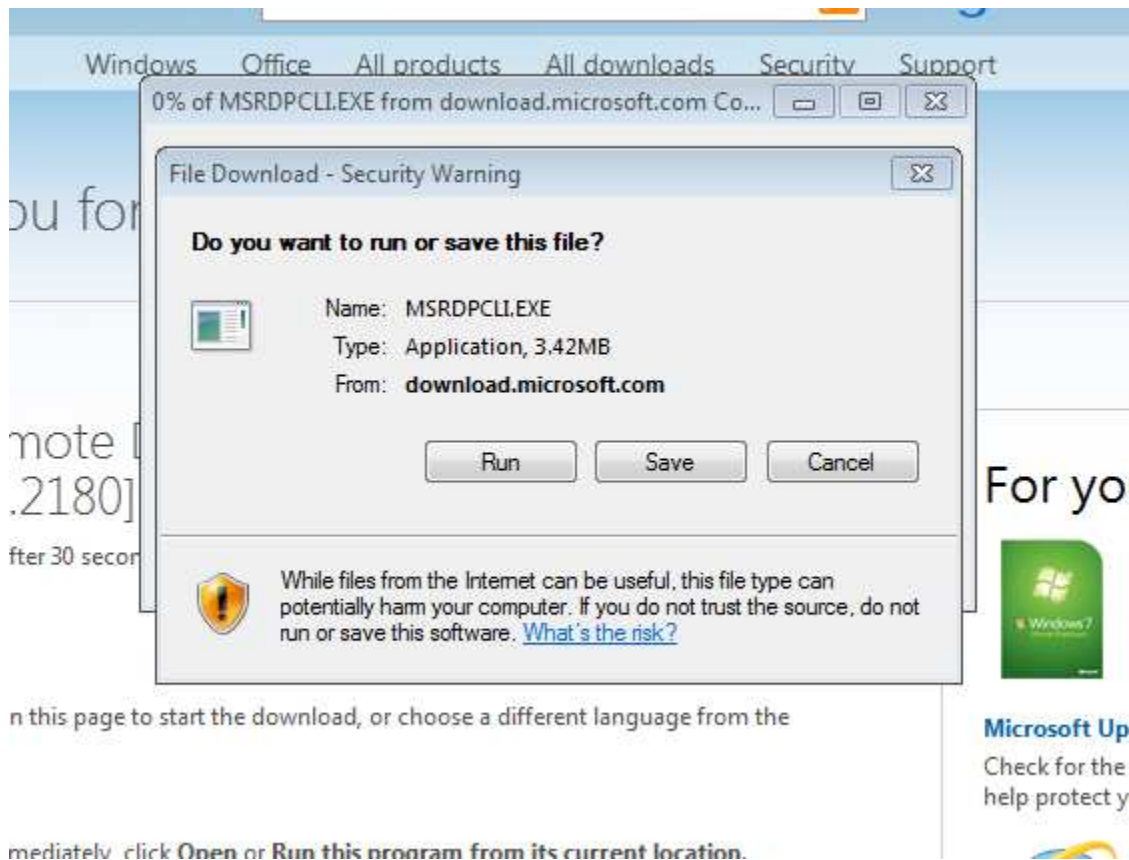
### Overview

This software package will install the client portion of Remote Desktop on any of the following operating systems: Windows 95, Windows 98 and 98 Second Edition, Windows Me, Windows NT 4.0, Windows 2000, and Windows 2003. (This is the same version of the client software as in Windows XP Service Pack 2.) When run, this software allows older Windows platforms to remotely connect to a computer running Windows XP Professional with Remote Desktop enabled.

This package provides flexible deployment options of the full Terminal Services Client, including auto-repair through Windows Installer technology and application publishing via IntelliMirror™ management technologies or Microsoft Systems Management Server (SMS).

**Note** The Remote Desktop Connection software is pre-installed with Windows XP. To run it, click **Start**, click **All Programs**, click **Accessories**, click **Communications**, and then click **Remote Desktop Connection**. This software package can also be found on the Windows XP Professional and Windows XP Home Edition product CDs and can be installed on any supported Windows platform. To install from the CD, insert the disc into the target machine's CD-ROM drive, select **Perform Additional Tasks**, and then click **Install Remote Desktop Connection**.

Click “Download”. This will open the following window, where you will need to click “Start Download”. The bottom information is a useful reminder (if you are on a campus computer) about where to find the Remote Desktop Connection. Remember though, it may simply be in the “Accessories” folder and not the “Communications” folder. After you click download, the following window will appear:



Click “Save” and save the program into a folder on your computer that you will remember. Once you save it to a folder, click “Run” and it will install the program on your computer. Once it’s installed, in the folder in which you stored the program, there should be the following icon:



Click on it and the following screen will appear:



In the “Computer” section, you need to type in `umbrella.scf.oregonstate.edu`. In the “Username” section, you will need to type in your ONID ID (i.e “ONID\idname”). If you are logged onto your ONID account on a campus computer, the program may enter your ONID user name for you.

Once you have entered the following information, save the connection settings, and click “Connect”. This will bring you to a page that will ask you for your ONID password; after giving this information you will be connected to the host computer through umbrella. You will enter into a blue screen with the server name on the top in the center.

Once in umbrella, you may wonder how to obtain to your documents on your ONID account. The easy way to do this is to click on the “Folder” icon, which will be either in the upper left of the desktop or in the toolbar. Within this window you will see a folder with your ONID username on it; this is your ONID or z drive where you are instructed to save your documents.

To open STATA on the host computer, click on the “Start” Menu. Then, when you look through “All Programs”, open the “Statistics” folder you should see a folder that says “STATA”. Click on the folder and it will open up three STATA programs (STATA 10, STATA 11, and STATA 12). These are all the same thing, if you click on one it will open up the software program STATA for you!

## Pre-Lab 2: Loading Datasets into STATA and Saving Records of Work

---

### **Learning Objective 1: Uploading a Database into STATA**

### **Learning Objective 2: Creating and saving a (log) record of your work in STATA**

There are three types of files in STATA. The first two we are going to create in this lesson. These are:

1. Data files (.dta): These files contain your data that you have uploaded into STATA. It is important to save this file, as you want to be able to re-use and re-access your dataset.
2. Log (output) files (.smcl): These files store all work that you do in STATA. Not only do they record the commands that you program into the software, but they also record the output that results from these commands. Log files can be very convenient if you failed to write down your output and you do not want to re-run your commands from scratch!
3. Do (input) files (.do): These files store all the commands you type into STATA. Unlike log files, they do not present your output. Do files are convenient if you want to re-run your commands on your data in different sittings. However, this lab will emphasize the log file, as it records both inputs and outputs.

The easiest way to load datasets into STATA is to first input/download them into excel. Below I have a simple spreadsheet pulled from a dataset of mine on United Kingdom (UK) graduate earnings. It presents estimated salaries in pounds sterling of 20 random UK graduates and was pulled from a greater sample of 20,000. With any dataset you construct you want to make sure that the label of your variables is in the first row.

Sample Excel Worksheet

File

Home

Insert

Page Layout

Formulas

Data

Review

View

Paste

Clipboard

Font

Alignment

General

\$ % ,

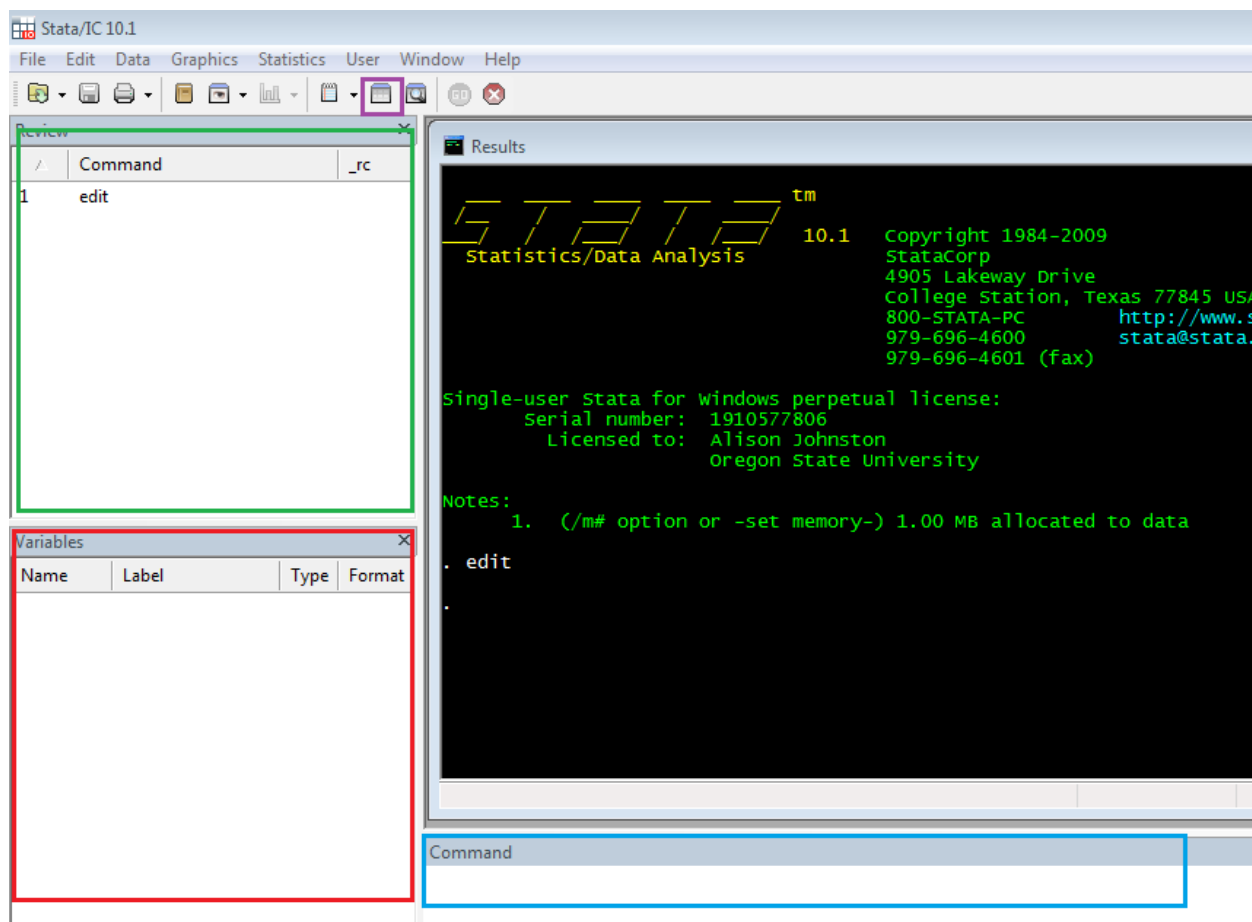
Number

J12

fx

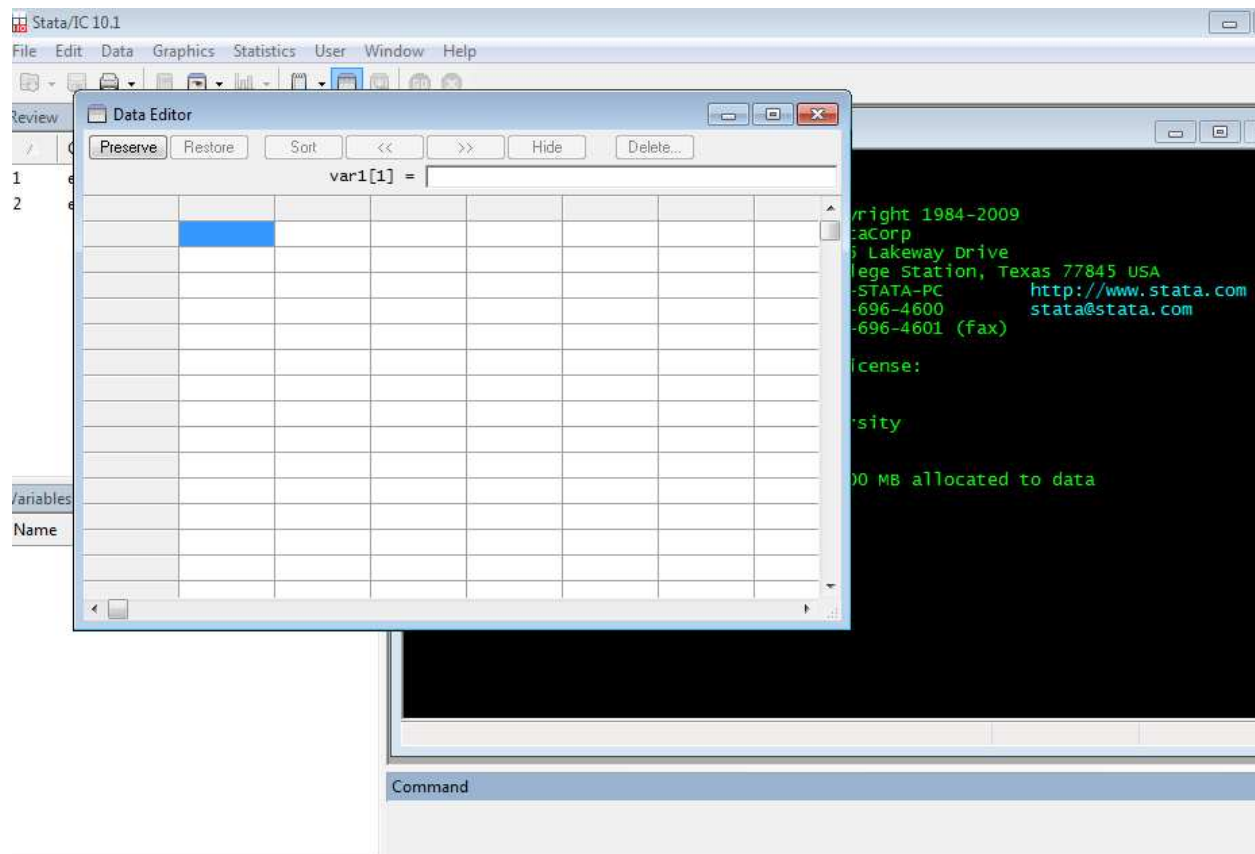
	A	B	C	D	E	F	G	H
1	Graduate ID	Sex	earns22	earns23	earns24	earns25		
2	1951	Male	11469.14	15534.49	16819.94	17801.23		
3	948	Female	0	13549.67	20357.82	18525.86		
4	4966	Female	0	0	22978.99	27709.72		
5	8573	Male	15276.23	19486.08	14392.44	18197.88	Variable Names	
6	9327	Female	17788.38	21896.97	23477.71	27628.67		
7	5317	Male	18429.38	27783.68	32813.86	37556.77		
8	4340	Male	28797.46	39036.21	45362.8	58944.42		
9	8974	Male	28502.5	40120.79	60672.89	65457.4		
10	4632	Female	18509.62	23380.29	28809.23	27726.78		
11	9217	Male	0	21224.86	23778.35	23114.92		
12	4434	Female	14448.28	17620.76	19599.03	20667.93		
13	1714	Female	15592.26	21770.97	25355.1	28419.71		
14	2663	Male	21558.6	30140.74	34078.95	0		
15	9801	Female	14750.46	19919.07	23640.95	23124.26		
16	2520	Male	7756.391	19198.26	21533.77	21622.29		
17	7512	Male	0	19776.17	17883.52	26469.22		
18	5148	Female	21091.45	27931.92	34390.24	31701.86		
19	9172	Female	11895.62	15564.2	13539.02	19460.33		
20	8176	Male	13035.77	17465.03	0	18533.97		
21	8181	Female	18372.86	21808	23959.87	28944.57		
22								

The easiest way to load a spreadsheet into STATA from Excel is simply via copy/paste. Open up STATA. You should see the following screen below (I present the screen for STATA 10):

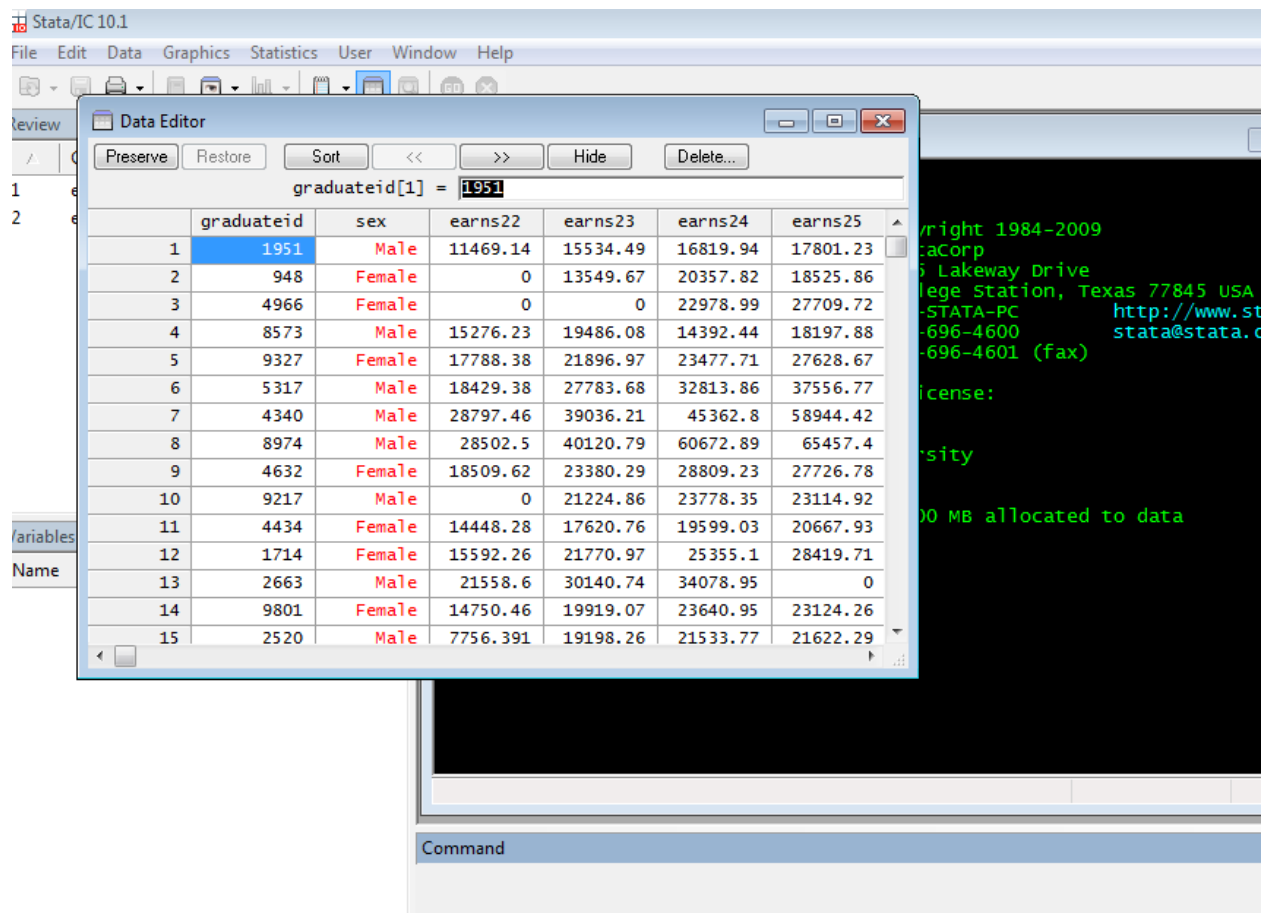


The black screen will display all your output – this is where all your statistical results will come out of. I’ve highlighted four important boxes on the screen. The first box, highlighted in red, is the box that contains all our variables. Note there is nothing in this box yet, as we have not yet inserted any data into STATA. The second box is the command box, highlighted in light blue. In this box, we will be entering all our coding, which will tell STATA what to do with our data. The other command box, highlighted in green, provides a record for every command we’ve inserted into the program. This command box is very useful if you’ve been running a lot of commands and need to reinsert them again, or slightly modify commands you’ve already run. Do not worry too much about both command boxes for this lesson.

The tiny fourth box, highlighted in purple, is the “Data Editor” box. When you click on this box, you should obtain the following window:



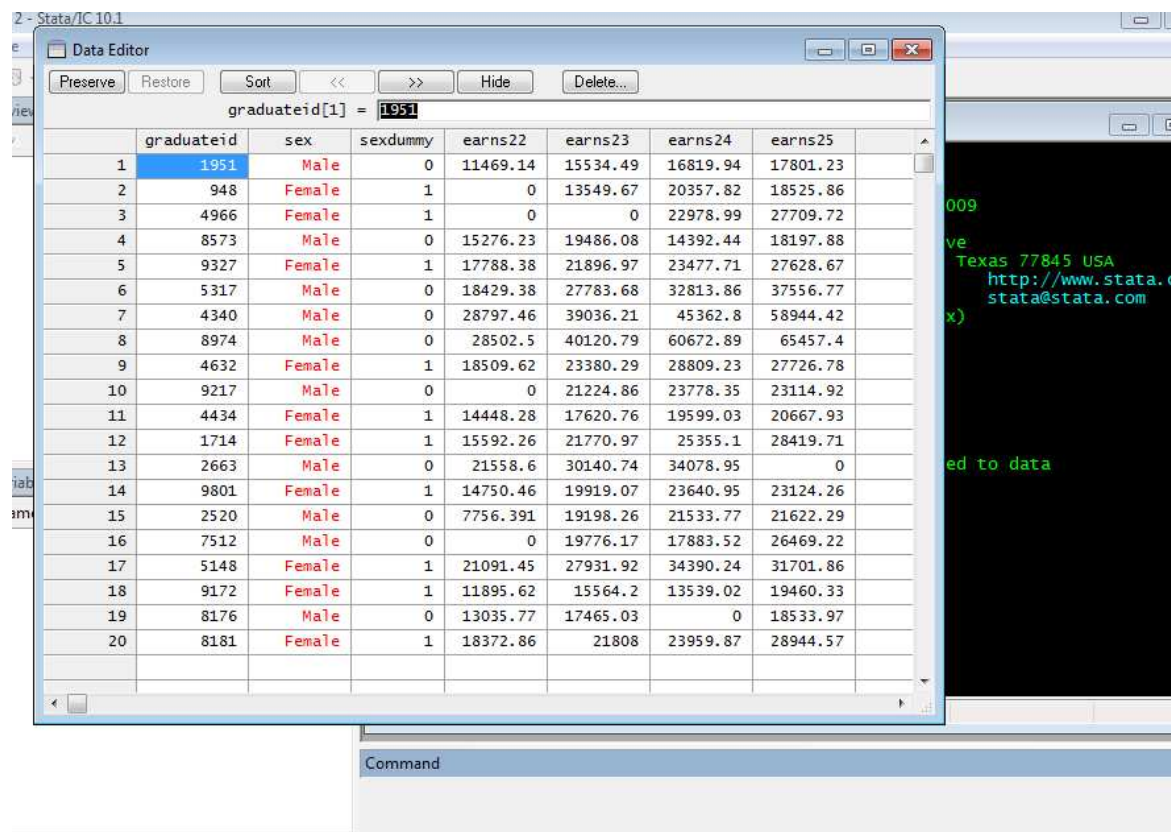
It is in the data editor that you are going to paste your dataset in from Excel. Copy all the available data from Excel and paste it into the first cell of the Data Editor (highlighted in blue in the above picture). You should obtain the following page, where your data is automatically transferred into the data editor:



Note that all data in the in the editor is black except for “sex” which is red. STATA will not recognize the “sex” variable for statistical commands because it is a word, rather than a number – all variables you record into STATA must be codified as numbers!

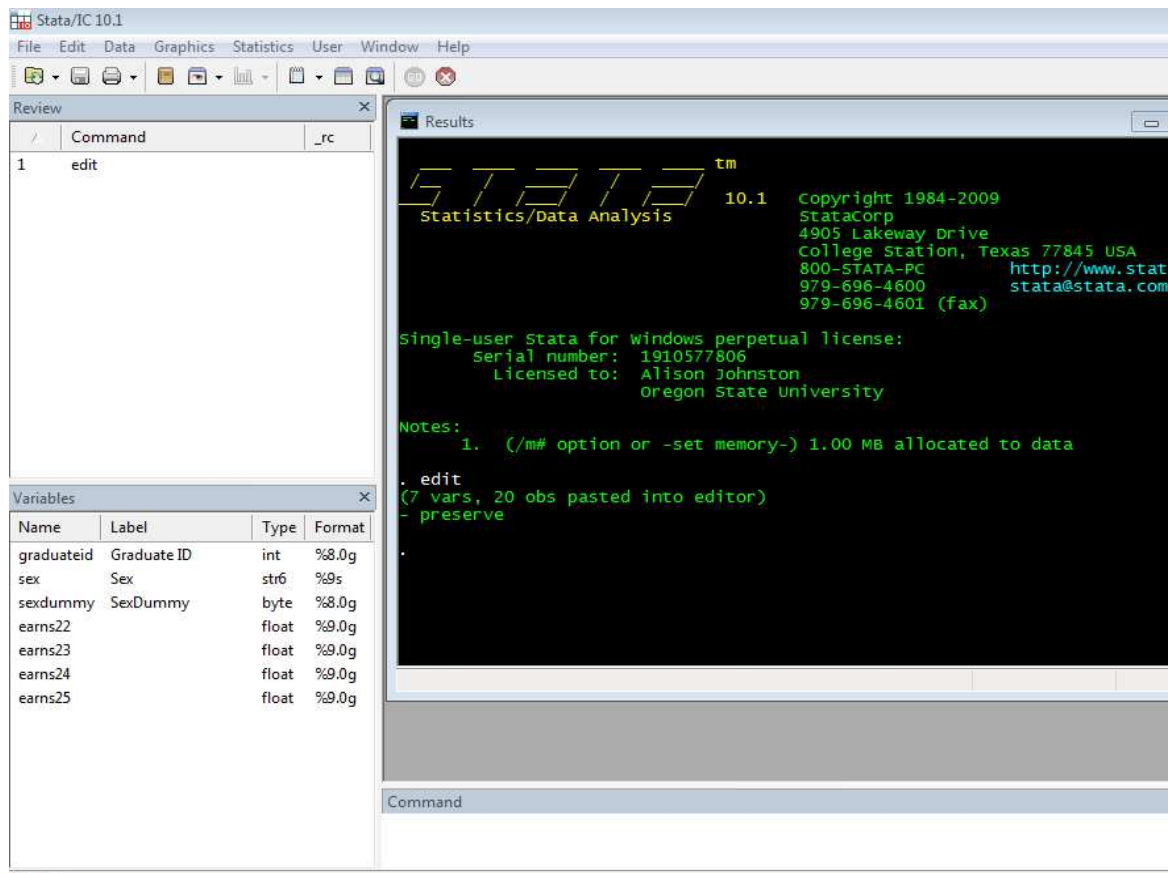
In order to convert sex into a number code that STATA will recognize, we need to convert it into a “dummy variable”: a dummy variable is one that takes the code 0 or 1, reflective of a binary characteristic. Since we only have two categories of “sex”, let’s codify men as “0” and women as “1”.

The easiest way to code these variables is in Excel. In a new column next to the “sex” column, recode all Males as 0 and Females as 1; call this new variable “sex dummy”. Then, re-copy the dataset back into STATA. If you do this, you should see the following output in the data editor:



Notice how the “sex dummy” variable is in black. This means that STATA will recognize it as a variable.

Click on the “Preserve” editor. You should see the following screen below (notice how our variable box, highlighted in red, now has seven variables in it: graduateid, sex, sexdummy, earns22, earns23, earns24, and earns25):



## CONGRATULATIONS! You've just uploaded a dataset into STATA!

You can also upload data into STATA using the “insheet” command. This command may be more helpful for data uploading if your data file is large or if your data is in a .txt or .raw rather than .xls format. Excel files usually need to be converted into .csv files<sup>1</sup> in order to be uploaded via the “insheet” command. To upload a dataset using the “insheet” command, you must know the exact name of your datafile, including the main folders where it is saved (i.e. C:/documents/sppfolder/dataset.csv). Simply type the following into the STATA command box: “insheet using filename”. You should see the data from the file uploaded in your data editor.

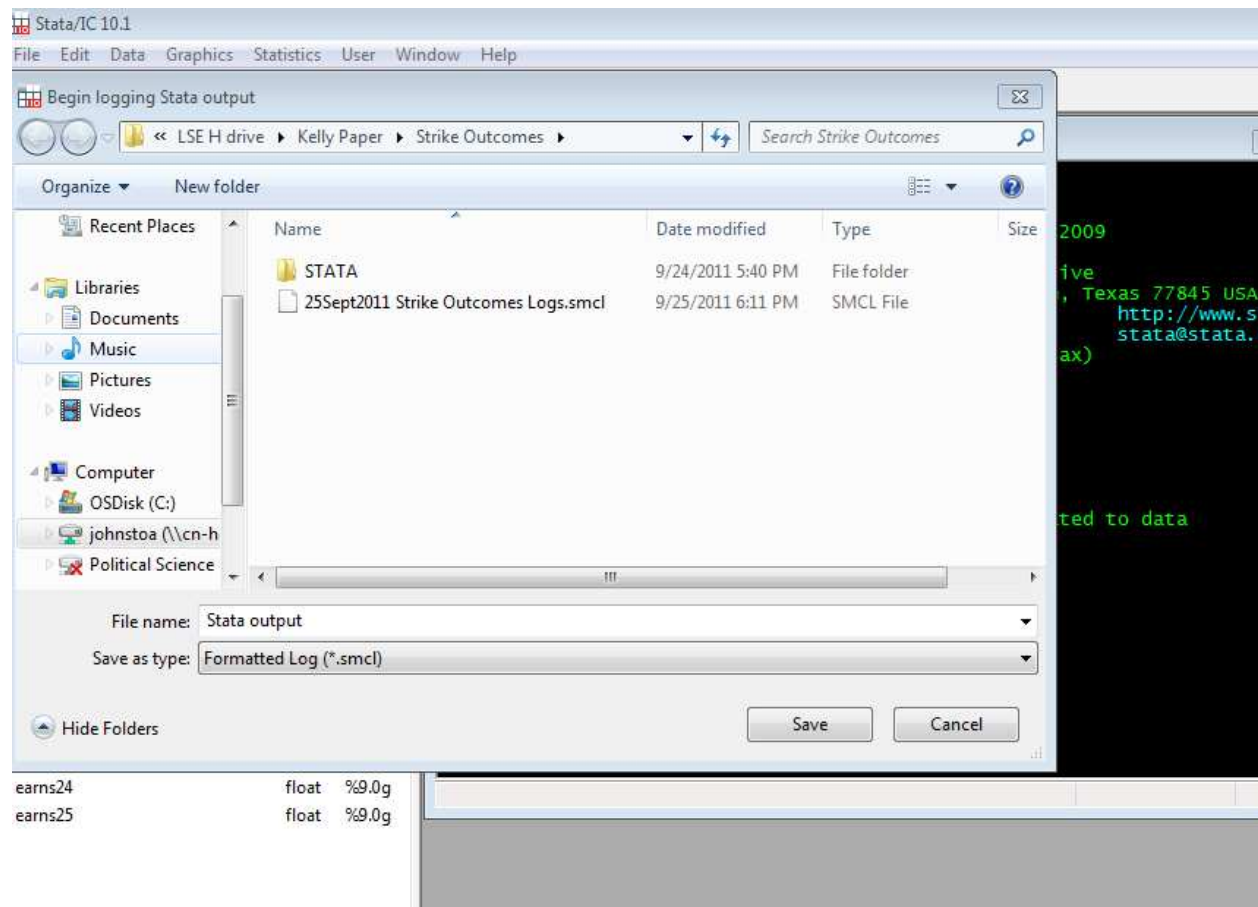
### STATA COMMAND PL2.1:

*Code: “insheet using filename”* where *filename* is the dataset you wish to upload.

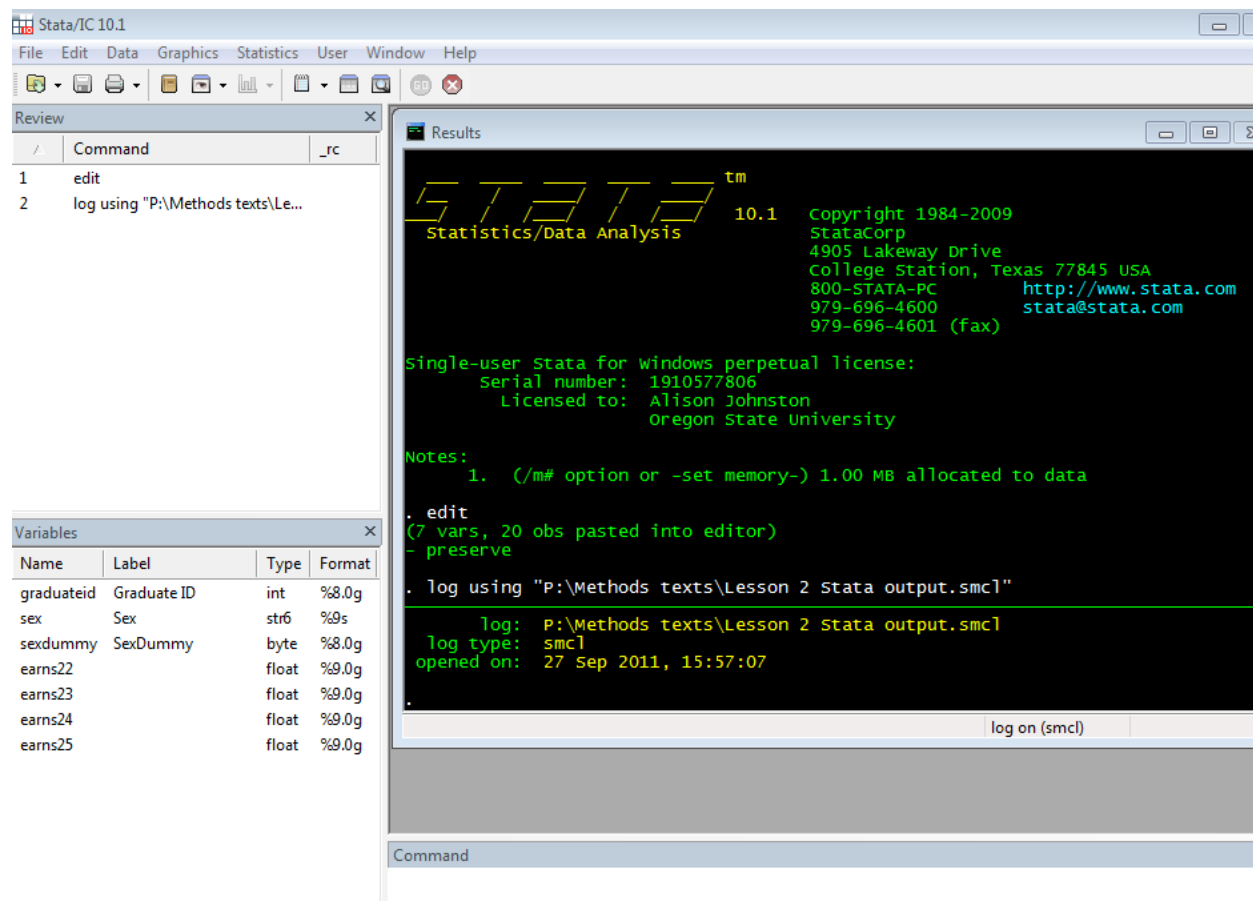
*Output produced:* Uploads the specified file into STATA.

<sup>1</sup> This can be done via the “save as” function in excel – on the “save as type” button - indicate you want to save the file as CSV (Comma delimited). If your excel file possesses multiple tabs, you must select only one tab to save as a CSV file.

Shifting now to creating a record of your work, click on “File” and then click on “Log” followed by “Begin”. You should be directed to the following window:



Once you save this log file (which is a .smcl file) to a folder, it will record everything that you do in STATA as well as all your results. After you save the log, you should see the following window below:

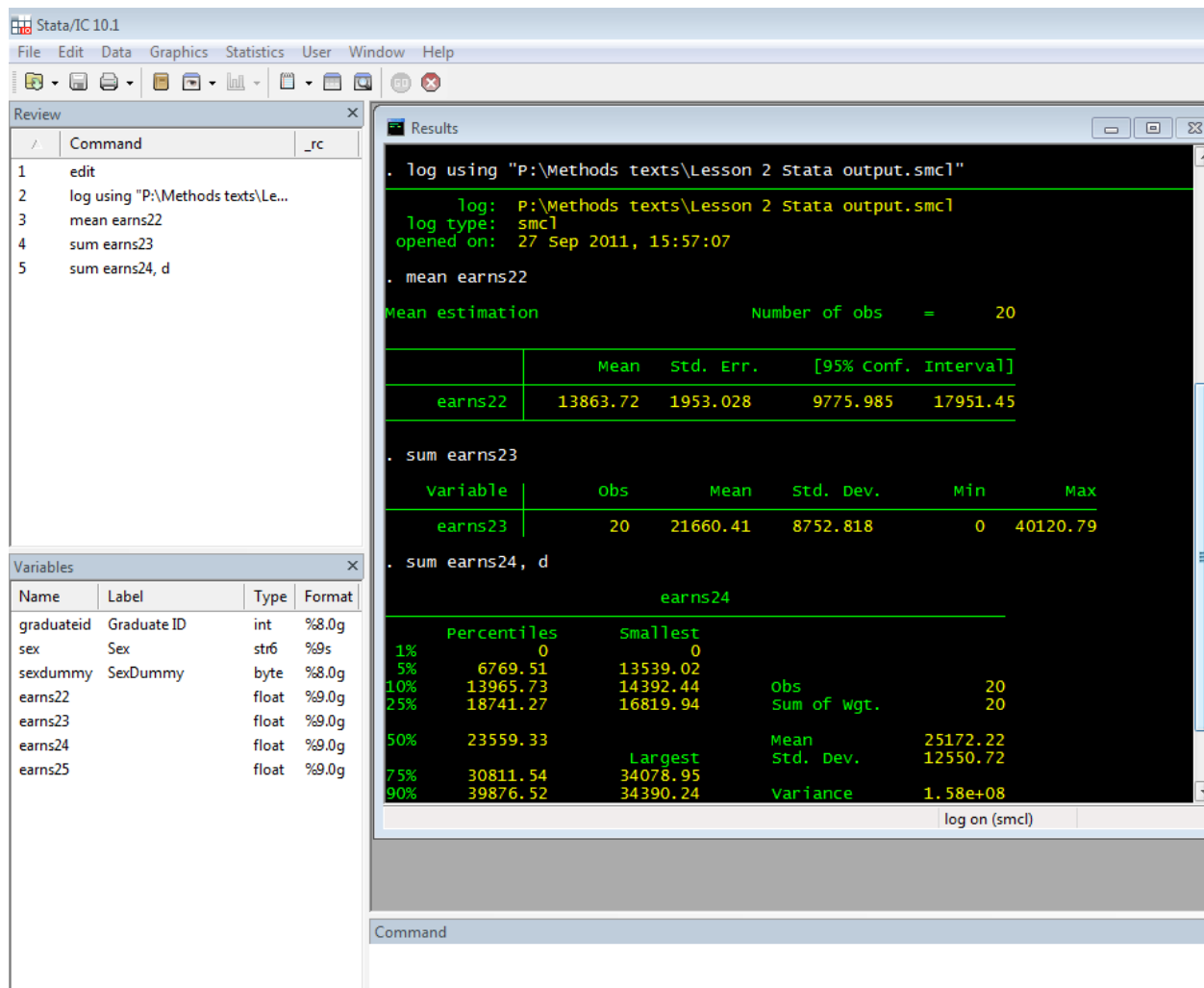


Notice how in the black box, STATA acknowledges that you are creating a log of your work. Every subsequent command you run, as well as the results, will be presented in this log.

This can be very helpful for your research, especially if you are running a lot of commands on STATA and you realize the day after that you forgot what your output was, and/or even worse, that you forgot what your commands were. To briefly demonstrate how STATA saves this log, I am going to run three simple commands (don't worry about the coding of these commands right now, we will come back to this):

1. I am going to calculate the mean of my earns22 variable. This can be done by typing in "mean earns22" into the command box, and then pressing "Enter".
2. I am going to ask STATA to present the summary statistics of my earns23 variable. This can be done by typing in "sum earns23" into the command box and then pressing "Enter".
3. I am going to ask STATA to produce a histogram table of my earn24 variable. This will be done by typing in "sum earns24, d" and then pressing "Enter".

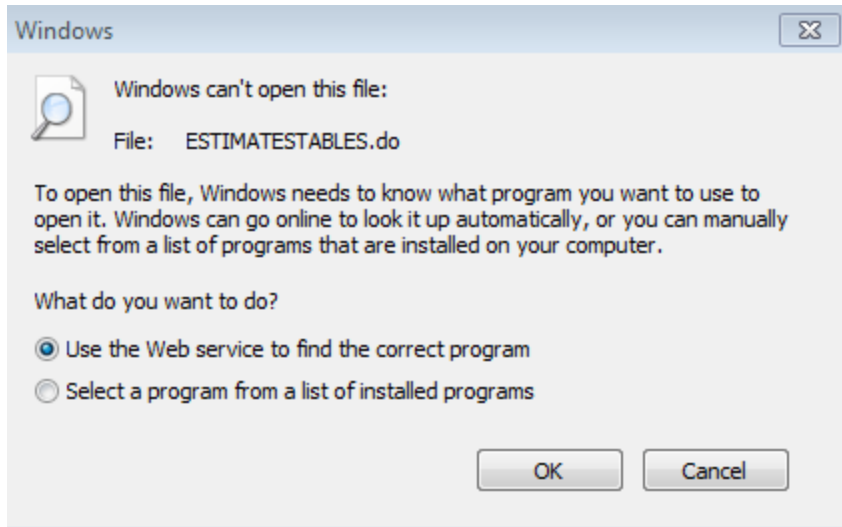
After running these three commands, you should see the following output:



Note how STATA has recorded all three of these codes in the right hand command log box. All the typed commands will show up in white in the black box. All the output will show up in green and yellow.

Now that I am done, I am going to close the log by clicking on “File”, then “Log”, then “Close”. STATA will acknowledge the closure of the log in the black box. Save the data file by clicking “File” then “Save”. It will ask you to save a “.dta” file, which will hold your dataset that you just uploaded into excel.

In order to review your saved log, go to the documents folder in which you saved your log in (it should be a .smcl file). Click the file. You may see the following image:



Note: Your computer may not automatically open your STATA log, so you may have to tell it which program to open it in. If this is the case, click on the “Select a program from a list of installed programs”. If STATA is not listed within the box, click “Browse” and select it from the “STATA” folder, which is found by opening “All Programs” and clicking “Statistics”. Once you select STATA as the program that you are opening up your log with, you should see the following screen:



---

### **Practice Problems:**

Lab Practice Problem 1: Upload a dataset that you have collected and codified in Excel into STATA. Save the dataset in a folder you can remember (you will be using this for future lessons!).

## Lesson 1: Sampling and Populations

---

**Learning Objective 1: To understand the notion of sampling and problems that arise from sampling when making inferences about a population**

**Learning Objective 2: To understand the notion of normal distributions and what they indicate about our data**

**Learning Objective 3: To create a numerical variable in STATA that is a function of existing variables**

**Learning Objective 4: To codify a non-numerical variable into a numerical one in STATA**

**Learning Objective 5: To create a histogram in STATA in order to view the distribution of your data**

Generally when we want to empirically test a research question, we want to see how something impacts a population, that is an entire group of items that are of interest to us. Some researchers can examine entire populations if the number of observations within a population is small. For example, if your unit of analysis is US states, countries, etc., it is possible to capture the entire population of observations as the total state/country population is 50/204. Other researchers, on the other hand, may examine units of analysis whose populations are much larger; this is the case if your unit of analysis is individuals or households; in this case the total population may be impossible to examine due to its size. Because researchers in this latter category cannot look at the entire population, they must select a sample that is representative.

Statistical inference (the testing of the impact of one variable on another – say policy on human behavior) involves using a sample to draw conclusions about the characteristics of a population from which it came. If ever you use a sample, you must ask yourself two important things:

1. Is this sample representative of the population?
2. Is there a chance our sample is biased (i.e. over-representative of a certain group)?

In order for the sample to be representative of the population, we need to randomly select it. If we do not randomly select it, our sample could be biased. We want to avoid bias in research, because if our sample contains bias, we might produce conclusions about populations that are over/under-stated.

To give a brief example, pretend that two students (the over-achiever and the over-sleeper) are given an assignment to survey Oregonians about their use of alternative energy. The over-achiever knows that if he wants to properly assess Oregonians' use of alternative energies, he needs to find a random sample. To do so, he takes the entire Oregon census and randomly calls every 100<sup>th</sup> phone number, in order to ask them questions about their use of alternative fuels, collecting a total of 4,000 individual responses. The over-sleeper, however, wants to limit his work, and surveys only 4,000 Corvallis residents.

Before seeing the survey results from both students, we are likely to witness much higher alternative energy use from the over-sleeper's sample than the over-achiever's. Why would this be the case? According to the Environmental Protection Agency in 2009, Corvallis ranked as the number one city in the US for the use of green energy. Corvallis' exceptionalism, in other words, does not make it representation of Oregon. Had the over-sleeper adopted the same approach as the over-achiever, he would have avoided the fact that his sample is heavily biased towards individuals who use more green energy, on average, than the rest of the state.

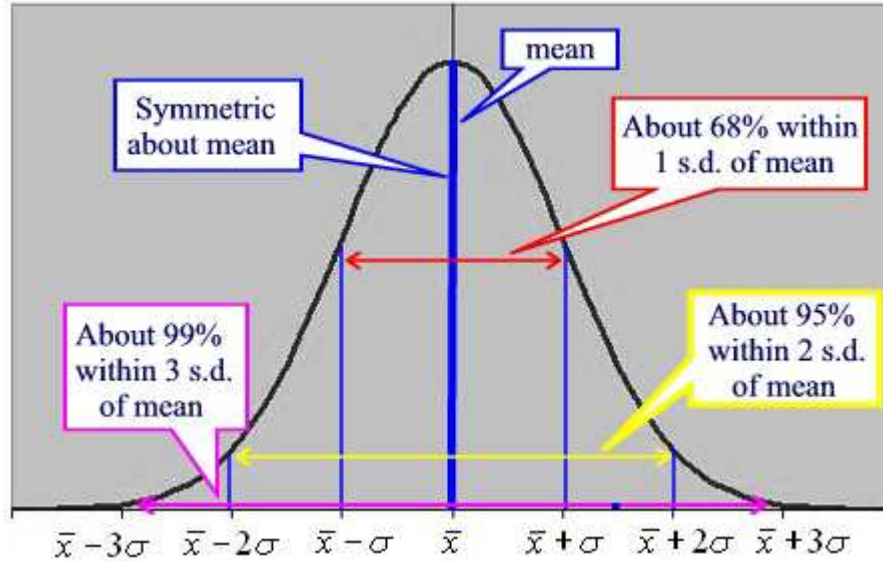
There are three types of selection biases that you want to avoid in your research. These biases include:

1. **Selection Bias:** Discussed above, this occurs when you select a sample that is not representative of the population. Another example of selection bias is conducting an opinion survey on 2012 presidential election outcomes, but asking the Fox News network to conduct it. The selection bias here is that a disproportionate amount of right-wing individuals tend to watch Fox news; hence their conclusions of a Republican candidate winning may be over-stated
2. **Survivor Bias:** This is an issue for retrospective studies, where you bias your sample because you only consider cases that have "survived". Say you are interested in analyzing a long-term trend, such as what features make a NYSE listed company successful throughout economic booms and busts. If you randomly select a group of companies listed on the NYSE exchange, your sample may have a survivor bias even though it was randomly selected. This is because firms that failed (i.e. went bankrupt) will not be listed on the NYSE, and hence your sample is over-representative of firms that have succeeded (i.e. remained in business).
3. **Nonresponse Bias:** If you are conducting a survey, you may notice that some people will chose to not participate. If non-participation, however, is systematic over certain groups, your sample will suffer from a non-responsive bias. For example, if you are conducting a phone survey on Oregonians' perception to the use of green energy, individuals who are apathetic about green energy use may choose to not to spend time responding to the survey. However, if apathetic green energy users refrain from the survey en masse, your survey is no longer representative of the population of Oregon. This is not due to problems with selection, but due to problems of non-responsiveness. Providing incentives for all participants, including apathetic green energy users, to respond (such as providing free gift vouchers) is a possible way to overcome systematic non-responsiveness.

The most basic rule of thumb for selecting a sample is to select it randomly. This may mean drawing individual telephone-numbers/addresses out of a hat. However, you may discover as you conduct your research that random selection is not as simple as it seems – particularly if you conduct surveys because you are forced to work with the responses that are given. If this is the case, do not despair; just make sure to make a note of it in your analysis and how it may impact your results!

In statistics, one of the most important concepts is sample/population distributions. Think of a distribution as a road-map to where your data lie. It provides a reference for all possible values of a variable that you're looking at in relation to the mean (average). The most common type of distribution used in statistics is the normal distribution. A normal distribution looks like a bell curve (see Figure 1.1 below).

Figure 1.1: Normal Distribution



Source: Roberts, 2012 (<http://www.regentsprep.org/Regents/math/algtrig/ATS2/NormalLesson.htm>)

Why is a normal distribution so helpful in statistics? Normal distributions can tell us the probability of where the majority of our data/observations lie, relative to the mean. Under a normal distribution, roughly 68% of our data should lie within  $\pm 1$  standard errors/distributions away from the mean, roughly 95% of our data should lie within  $\pm 2$  standard errors/distributions from the mean, and roughly 99% of our data should lie within  $\pm 3$  standard errors/distributions of the mean. This is a very powerful rule and it is not conditional on sample size (i.e. whether we are looking at a large or a small sample). As long as the distribution is normal, these three probabilities will hold true. When we cover standard errors in the next lesson, however, you will notice that our standard errors become smaller, and hence we improve accuracy as our sample size grows.

Normal (bell-curve) distributions are not only useful for determining where our data lie relative to a mean. They are also a central requirement for t-tests, one of the classical tests of hypotheses and a central component to data analysis. These features will be discussed in greater depth in future lessons, as well as the important Central Limit Theorem which is discussed in the next lesson. Before we advance to these topics, however, you will learn some basic coding commands in STATA, as well as how to plot a histogram of your data. Histograms are helpful graphics, as they reveal the distribution of a variable across all recorded observations. For this lesson, we will construct histograms from two datasets. One of these datasets you've used before; it contains earnings information on 20 randomly selected UK graduates. The second dataset contains earnings information on 200 randomly selected UK graduates.

## STATA LAB (LESSON 1):

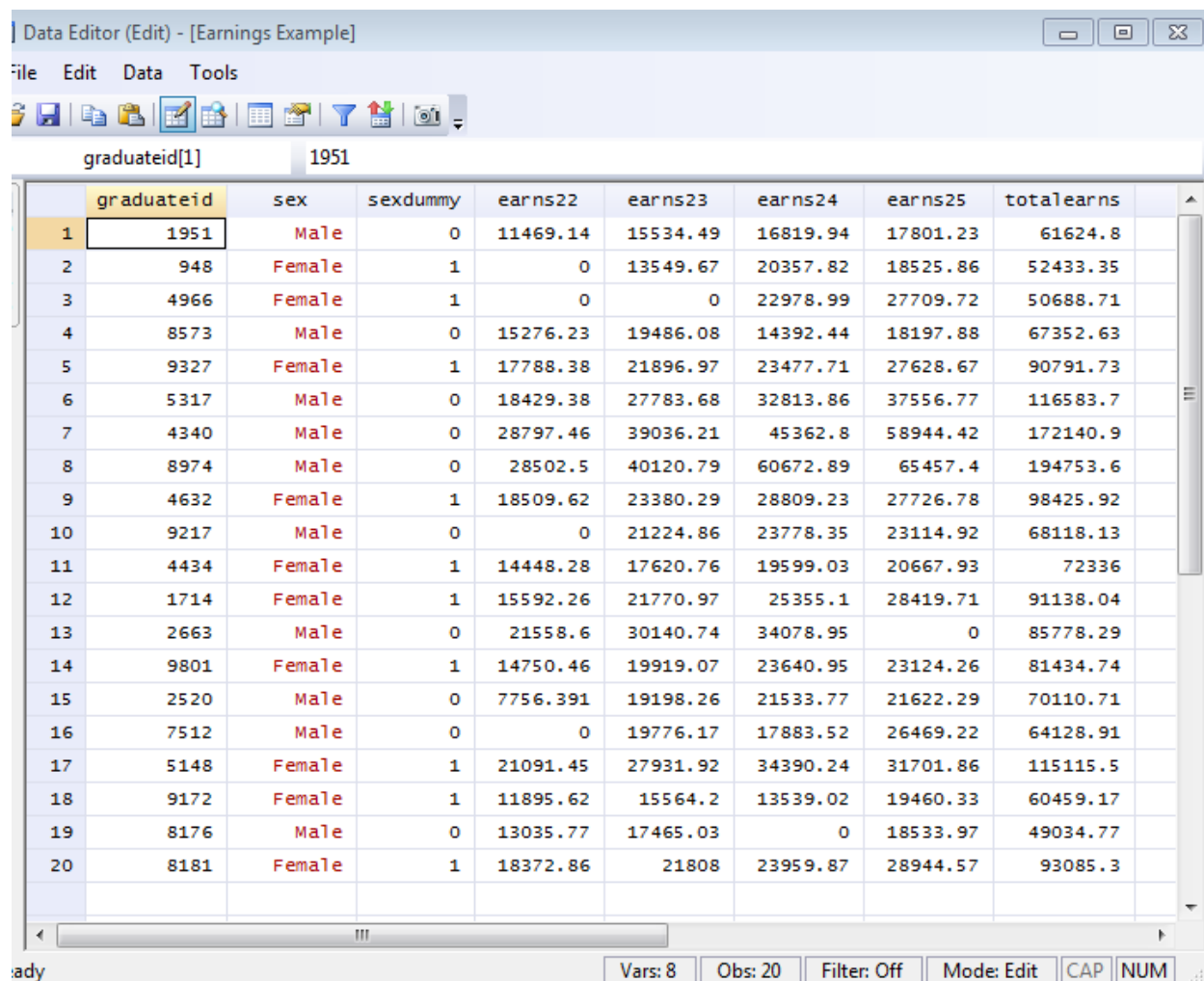
Open up the two excel datasets on graduate earnings, and upload them into two separate STATA windows via the copy/paste method we used before (see Pre-Lab 1). Before we begin, you will learn the “generate”, or “gen” for short, command which creates new variables from those which we have uploaded.

Let’s create a new variable that is the sum of earnings that a UK graduate makes between age 22 and 25 – we will call this “totalearns”. Starting with our smaller dataset, go to the command box (outlined in red below) and type the following command: “gen totalearns = earns22 + earns23 + earns24 + earns25” as demonstrated below:

The screenshot shows the STATA 11.0 interface. The top menu bar includes File, Edit, Data, Graphics, Statistics, User, Window, and Help. The main window displays the STATA logo and version 11.0, along with copyright information for StataCorp. The command box at the bottom is highlighted with a red rectangle and contains the command: `gen totalearns = earns22 + earns23 + earns24 + earns25`. The variable list on the left shows the following variables: graduateid (int), sex (str6), sexdummy (byte), earns22 (float), earns23 (float), earns24 (float), and earns25 (float).

Name	Label	Type
graduateid	Graduate ID	int
sex	Sex	str6
sexdummy	SexDummy	byte
earns22		float
earns23		float
earns24		float
earns25		float

After typing “gen totalearns = ....” into the command box, press “enter”. The command should appear in white in the black output box. To confirm that STATA has created the variable, open up the data editor (highlighted in blue above). You should see the following screen:



	graduateid	sex	sexdummy	earns22	earns23	earns24	earns25	total earns
1	1951	Male	0	11469.14	15534.49	16819.94	17801.23	61624.8
2	948	Female	1	0	13549.67	20357.82	18525.86	52433.35
3	4966	Female	1	0	0	22978.99	27709.72	50688.71
4	8573	Male	0	15276.23	19486.08	14392.44	18197.88	67352.63
5	9327	Female	1	17788.38	21896.97	23477.71	27628.67	90791.73
6	5317	Male	0	18429.38	27783.68	32813.86	37556.77	116583.7
7	4340	Male	0	28797.46	39036.21	45362.8	58944.42	172140.9
8	8974	Male	0	28502.5	40120.79	60672.89	65457.4	194753.6
9	4632	Female	1	18509.62	23380.29	28809.23	27726.78	98425.92
10	9217	Male	0	0	21224.86	23778.35	23114.92	68118.13
11	4434	Female	1	14448.28	17620.76	19599.03	20667.93	72336
12	1714	Female	1	15592.26	21770.97	25355.1	28419.71	91138.04
13	2663	Male	0	21558.6	30140.74	34078.95	0	85778.29
14	9801	Female	1	14750.46	19919.07	23640.95	23124.26	81434.74
15	2520	Male	0	7756.391	19198.26	21533.77	21622.29	70110.71
16	7512	Male	0	0	19776.17	17883.52	26469.22	64128.91
17	5148	Female	1	21091.45	27931.92	34390.24	31701.86	115115.5
18	9172	Female	1	11895.62	15564.2	13539.02	19460.33	60459.17
19	8176	Male	0	13035.77	17465.03	0	18533.97	49034.77
20	8181	Female	1	18372.86	21808	23959.87	28944.57	93085.3

Notice a new column has emerged titled “total earns” – this is your new generated variable.

**CONGRATULATIONS! You have just generated a new variable in STATA!**

The “generate” command is very versatile and enables you to create any type of variable from your data<sup>2</sup>; you can multiply, divide, add, subtract, take percentages, etc. for any variable in your data editor and make a new variable. It also saves you a lot of time from having to do it in Excel!

<sup>2</sup> This command is covered more in depth in the data management appendix.

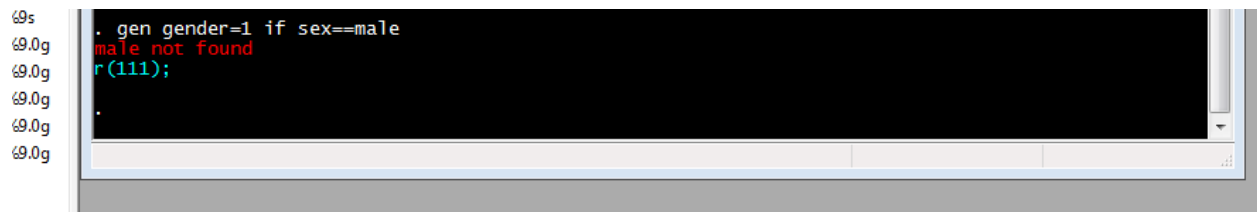
### STATA COMMAND 1.1:

*Code:* “**gen newvariable = f(var1, var2, ..., var<sub>n</sub>)**”, where newvariable is the variable you wish to create and f(var1, var2, ..., var<sub>n</sub>) is the function of current variables which already exist

*Output produced:* Creates a new variable from existing ones

*Caveats:* Works ONLY on numerical values.

One major advantage of the “gen” command is that it saves us a lot of work in Excel in terms of creating new variables. One major disadvantage, however, is that “gen” does not work on non-numerical values. For example, say rather than assigning numerical values to “sex” in our database with 200 observations (a long and tedious process), we would rather convert this into a numerical variable in STATA. Start first with the following command, which specifies that we want to create a new variable, “gender”, where the “male” variable is equal to 1: “gen gender=1 if sex==male”. You should see the following image:

A screenshot of the STATA command window. The command prompt shows the command ". gen gender=1 if sex==male" being entered. Below the command, the output "male not found" is displayed in red text, followed by "r(111);". The command window has a black background with white text for the command and red text for the error message. The STATA logo is visible in the top left corner of the window.

Notice that we see a red command that states STATA is unable to find the “male” variable. This is because, as mentioned in Pre-lab 2, STATA does not recognize non-numerical values.

The “encode” command<sup>3</sup>, however, converts non-numerical values into numerical values for you. Within the command box, let’s again try to create a coded value for the “sex” variable, calling the new variable “gender”. Type in the following command: “encode sex, generate(gender)”. You should see the following screen:

<sup>3</sup> This command is also covered more in depth in the data management appendix.

Review

	Command	_rc
1	edit	
2	generate str gender = =0 if sex...	198
3	generate str gender = 0 if sex=...	111
4	edit	
5	gen gender=1 if sex= male	109
6	gen gender=1 if sex== male	111
7	encode sex, generate(gender)	

Variables

Name	Label	Type	Format
idorig		int	%8.0g
sex	Sex	str6	%9s
earns22		float	%9.0g
earns23		float	%9.0g
earns24		float	%9.0g
earns25		float	%9.0g
earns26		float	%9.0g
gender	Sex	long	%8.0g

Results

```

979-696-
single-user Stata for windows perpetual licens
Serial number: 1910577806
Licensed to: Alison Johnston
Oregon State University

Notes:
1. (/m# option or -set memory-) 1.00 MB

. edit
(7 vars, 200 obs pasted into editor)
- preserve

. generate str gender = =0 if sex==female
=0 invalid name
r(198);

. generate str gender = 0 if sex==female
female not found
r(111);

. edit
- preserve

. gen gender=1 if sex= male
type mismatch
r(109);

. gen gender=1 if sex== male
male not found
r(111);

. encode sex, generate(gender)

.

```

Command

Notice how there is a new variable within our “Variables” box (highlighted in red).

**CONGRATULATIONS! You have just codified a non-numerical variable into a numerical one in STATA!**

To check that we have created a numerical variable, open the “Data Editor” at the top of the page. You should see the following:

Data Editor

Preserve Restore Sort << >> Hide Delete...

gender[4] = 2

	idorig	sex	earns22	earns23	earns24	earns25	earns26	gender
1	1951	Male	11469.14	15534.49	16819.94	17801.23	21091.5	Male
2	948	Female	0	13549.67	20357.82	18525.86	17933.5	Female
3	4966	Female	0	0	22978.99	27709.72	26179.74	Female
4	8573	Male	15276.23	19486.08	14392.44	18197.88	25118.59	Male
5	9327	Female	17788.38	21896.97	23477.71	27628.67	35983.36	Female
6	5317	Male	18429.38	27783.68	32813.86	37556.77	41664.79	Male
7	4340	Male	28797.46	39036.21	45362.8	58944.42	56249.53	Male
8	8974	Male	28502.5	40120.79	60672.89	65457.4	74173.16	Male
9	4632	Female	18509.62	23380.29	28809.23	27726.78	36796.45	Female
10	9217	Male	0	21224.86	23778.35	23114.92	28616.44	Male
11	4434	Female	14448.28	17620.76	19599.03	20667.93	19776.33	Female
12	1714	Female	15592.26	21770.97	25355.1	28419.71	35025.8	Female
13	2663	Male	21558.6	30140.74	34078.95	0	27574.33	Male
14	9801	Female	14750.46	19919.07	23640.95	23124.26	0	Female
15	2520	Male	7756.391	19198.26	21533.77	21622.29	20334.68	Male
16	7512	Male	0	19776.17	17883.52	26469.22	28182.05	Male
17	5148	Female	21091.45	27931.92	34390.24	31701.86	33621.91	Female
18	9172	Female	11895.62	15564.2	13539.02	19460.33	21369.85	Female
19	8176	Male	13035.77	17465.03	0	18533.97	19662.4	Male
20	8181	Female	18372.86	21808	23959.87	28944.57	29113.02	Female
21	470	Male	0	0	33765.81	38307.72	54486.56	Male
22	2859	Female	30080.42	38747.93	45974.2	47006.33	54772.35	Female
23	9372	Female	0	0	10013.38	12246.32	18761.89	Female

Notice that our new “gender” variable is the same as our old “sex” variable, with two major differences: 1.) it is highlighted in blue indicating that STATA recognizes the variable, rather than in red indicating that STATA does not recognize it, and 2.) when you click on the “female” or “male” cells in the value toolbar at the top of the data editor (highlighted in red) you see a numerical value (1 for female, 2 for male). The “encode” command always starts from 1 rather than 0; notice that our dummy values range now from 1 to 2, rather than from 0 to 1 when we manually coded it in excel. The basic rule of thumb in data analysis is that dummy variables (i.e. variables which have a binary value, such as “yes” and “no” or “male” and “female”) should be codified in terms of 0 and 1. To re-codify our gender variable into a new 0/1 dummy variable, which we will call “gender2”, type the following two commands into the command box:

1. “generate gender2=1 if gender==1”, and then “Enter”; this assigns all “female” variables a value of 1
2. “replace gender2=0 if missing(gender2)”, and then “Enter”; this replaces all our “missing” male gender values (i.e. those which gender had a value of 2”) with 0

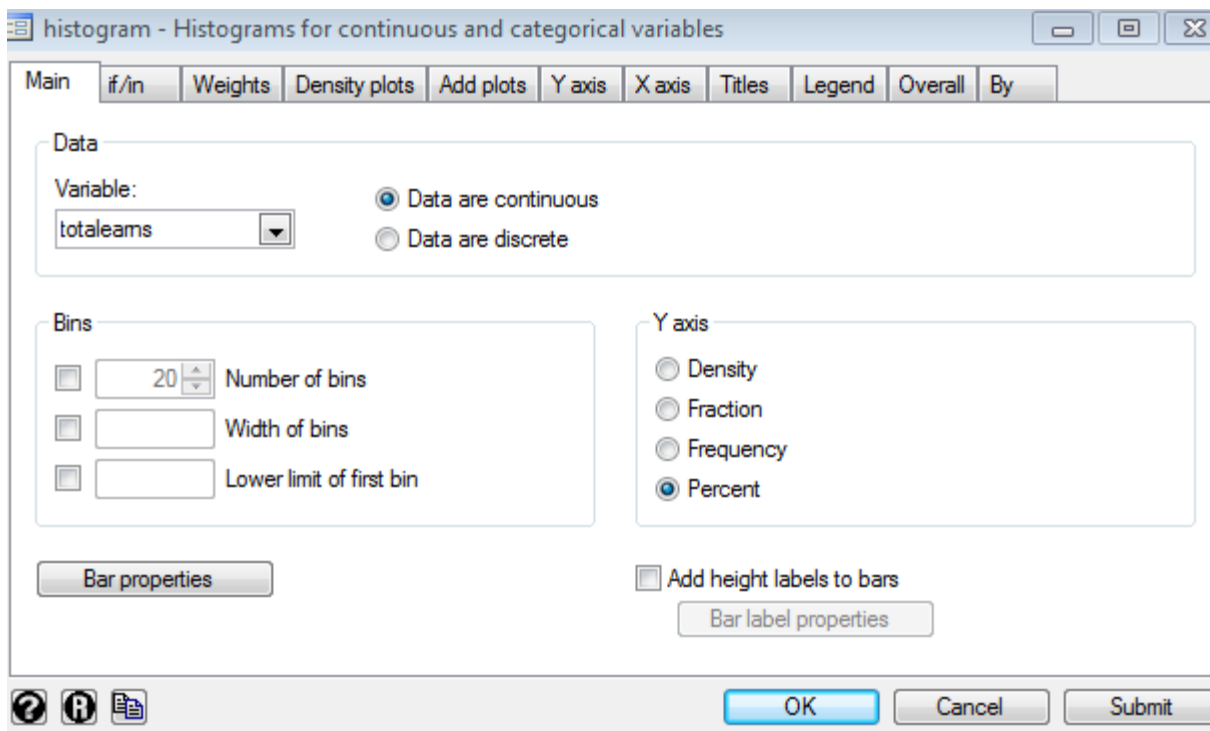
The “encode” command is also particularly helpful if you wish to codify categorical variables which have more than one non-numerical value (i.e. hair/eye-color, employment status, occupation, etc.).

## STATA COMMAND 1.2:

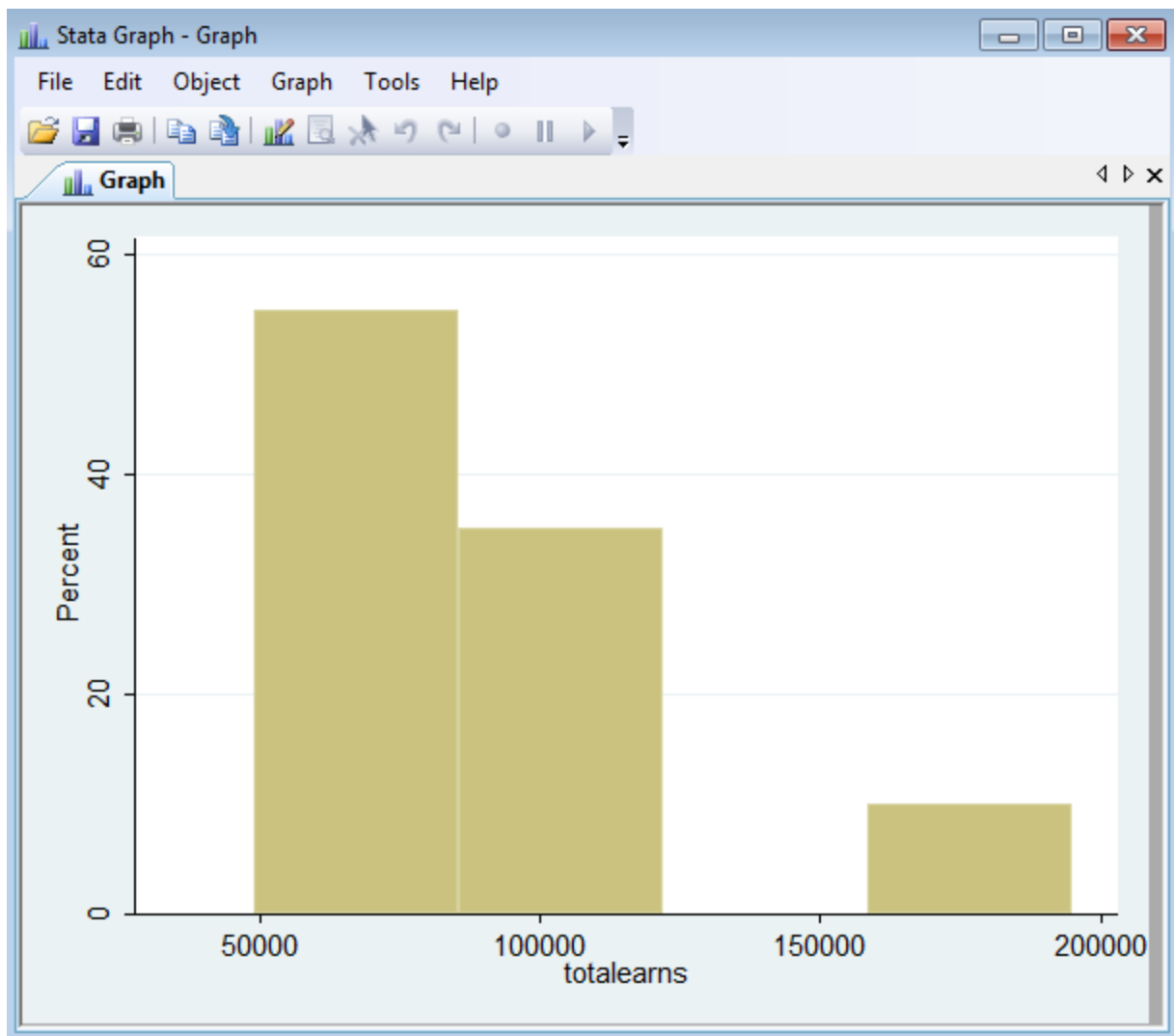
*Code:* “**encode var1, generate(newvariable)**”, where newvariable is the (numerically-coded) variable you wish to create and var1 is the non-numerical variable which already exists, and which you want to codify.

*Output produced:* Creates numerically-coded variables from existing non-numerically coded ones

Going back to our totalearns variable, let’s see what its distribution looks like by creating a histogram. In order to create a histogram, click on the “Graphics” tab in the toolbar, then click “histogram”. You should see the following window:



The “histogram” command in STATA presents you a visual for how your data are distributed once you place them in numerical order. In the “Variable” box, enter “totalearns”. Under the “y-axis box” click the “percent” bubble – this will tell you what percentage of your data lie within each bar range. After you’ve done this, click “Ok”. You should be presented with the following graphic (wait for it, sometimes it takes a while to appear...):



**CONGRATULATIONS! You have just created a histogram in STATA!**

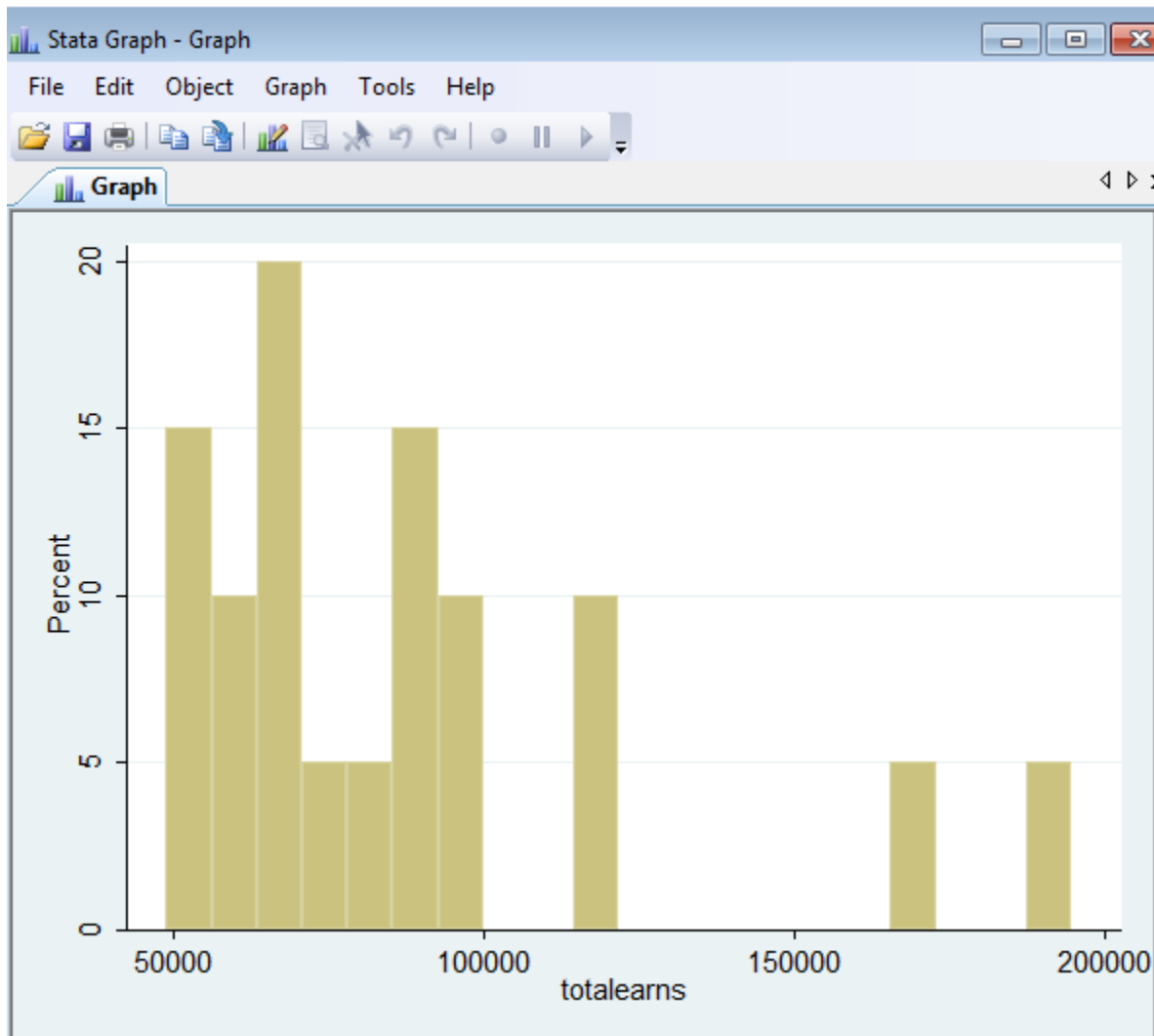
The “histogram” command can also be generated in the command box using the following coding:

**STATA COMMAND 1.3:**

*Code:* “**histogram var1, percent**” where var1 is the variable of interest.

*Output produced:* Creates a histogram graphic of specified variable.

You may notice the above graphic does not closely resemble the normal distribution that we saw in Figure 1.1. One reason for this is because our bar length is quite wide. One way to reduce the bar length and thus increase the number of bars, is increasing the number of “Bins”. Go back to “Graphics” tab and select “Histogram”. This time, in the “Bins” box, click the first box, “number of bins”, and enter the number 20. Click “Ok” and you should see the following histogram:



Notice that this histogram is slightly more spread out than the first, but still displays a skew to the left (i.e. towards 0). Let’s determine if a similar distribution of “totalearns” emerges in the larger sample size. Upload the larger dataset (of 200 randomly selected graduates) into a new STATA window.

We are going to do exactly the same commands to our large STATA dataset as we did with our small dataset. First, generate the variable “totalearns” which is the sum of earnings that a UK graduate makes between age 22 and 25 by typing “gen totalearns = earns22 + earns23 + earns24 + earns25” in the

command window and pressing “Enter”. Check the data editor to make sure there is a new column with your newly created variable.

Now let’s create a histogram of our “totalearns” variable in our large dataset. Click on the “Graphics” tab, and then click “histogram”. You should see the following box (again):

histogram - Histograms for continuous and categorical variables

Main | if/in | Weights | Density plots | Add plots | Y axis | X axis | Titles | Legend | Overall | By

Data

Variable:

☒ Data are continuous  
☐ Data are discrete

Bins

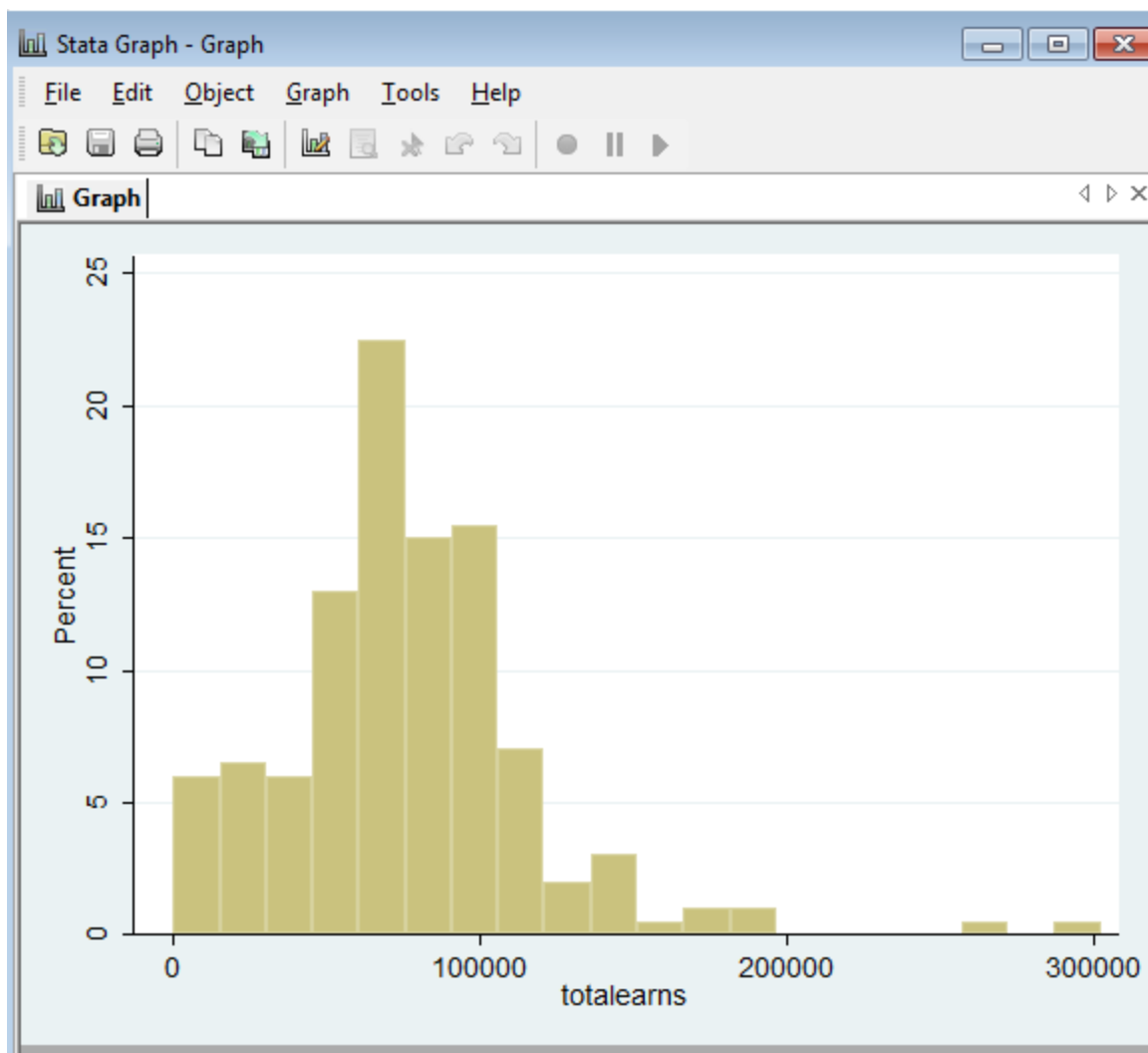
☒  Number of bins  
☐  Width of bins  
☐  Lower limit of first bin

Y axis

☐ Density  
☐ Fraction  
☐ Frequency  
☒ Percent

☐ Add height labels to bars

In order to standardize output, in the “Bins” box (highlighted in red) enter 20 bins like we did for the previous histogram from our smaller sample. Also click the “Percent” option under the “y-axis box” and click “Ok”. You should see the following histogram:



**CONGRATULATIONS! You have just created another histogram in STATA!**

You may notice something different about this histogram: its distribution resembles a normal distribution much better than that from our sample size of 20. It is not perfectly bell-shaped, but the distribution converges more closely to normal than that for “totalearns” in the smaller dataset.

---

### **Practice Problems:**

Practice Problem 1: Suppose a researcher in the military wants to examine how often people contract the common cold. She is unable to leave her military base as she is on active duty, and so she examines the health of all 1,000 individuals living on the base. List some reasons why her conclusions on cold frequency may be over/under-stated due to her sample selection.

Practice Problem 2: You want to examine the aptitude of high-school seniors in mathematics. As your sample, you select all high-school seniors that take the Scholastic Aptitude Test (SAT). Explain why this sample may bias your conclusions.

Practice Problem 3: Open the two data files containing UK graduate earnings that you used in this lesson plan. Create a new variable which is the average earnings of a graduate's first four years out of university. (Check the data editor to make sure STATA generated the variable)

Practice Problem 4: Create a histogram of your new "average earnings" variable for the small and large dataset. How do the two distributions compare?

Practice Problem 5: Create a histogram of `earn22` for the small and large dataset. How do the two distributions compare? What do you notice that is different about the large sample's distribution, compared to the other histograms we calculated in the large dataset?

## Lesson 2: Descriptive Statistics

---

**Learning Objective 1: To understand the notion of means, medians, variance, standard deviations, and standard errors (i.e. descriptive statistics) and how they compare between populations and samples**

**Learning Objective 2: To understand what happens to the distribution of a sample's mean as the number of randomly drawn samples increases (i.e. the Central Limit Theorem)**

**Learning Objective 3: To understand what happens to descriptive statistics as one's sample size increases**

**Learning Objective 4: To understand how to calculate descriptive statistics from your data in EXCEL (means, medians, variation, and standard deviations, etc.)**

**Learning Objective 5: To understand how to calculate descriptive statistics from your data in STATA**

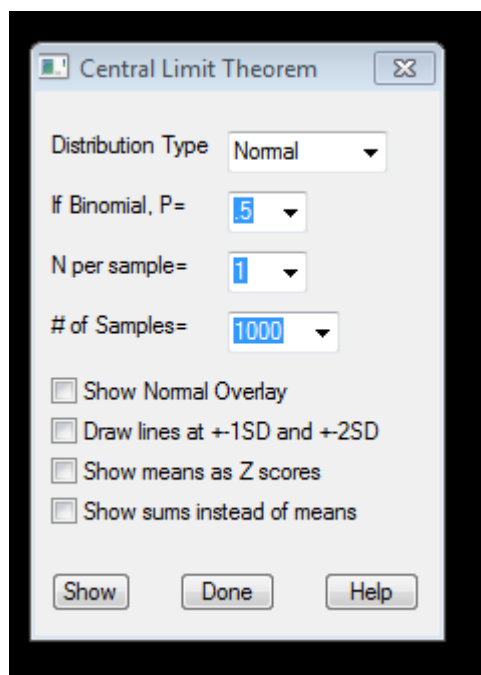
In the last lesson we talked about a very commonly used (and very helpful) type of distribution: the normal distribution. If you recall back to Figure 1.1, one convenient property of normally distributed data is that we can determine where a certain percentage of this data lie, in relation to the mean. To recall, roughly 68% of our data should lie within  $\pm 1$  standard deviations/errors away from the mean, roughly 95% of our data should lie within  $\pm 2$  standard deviations/errors from the mean, and roughly 99% of our data should lie within  $\pm 3$  standard deviations/errors of the mean. In this lesson, we are going to calculate what the mean and standard deviation/error are. These descriptive statistics can summarize effectively the important characteristics of our probability distribution.

A mean is one of the most common descriptive statistics used in econometrics. This is the average value of your variable over your sample or population: the sum of total observations divided by the number of observations. It provides a nice reference point, because if our data is normally distributed, we can determine where x% of our variables lie in relation to the mean. Yet even if our data are not normally distributed, the mean of our sample should approach normal as we increase the number of sample draws. According to the Central Limit Theorem, a pivotal theorem in probability theory, if you draw a random sample/sub-sample from a population/sample, the distribution of the mean of that (sub-)sample should approach normal as the number of random draws (i.e. the number of samples) increases. To illustrate this better, we will turn to the “clt” command in STATA which more effectively demonstrates the Central Limit Theorem in action.<sup>4</sup>

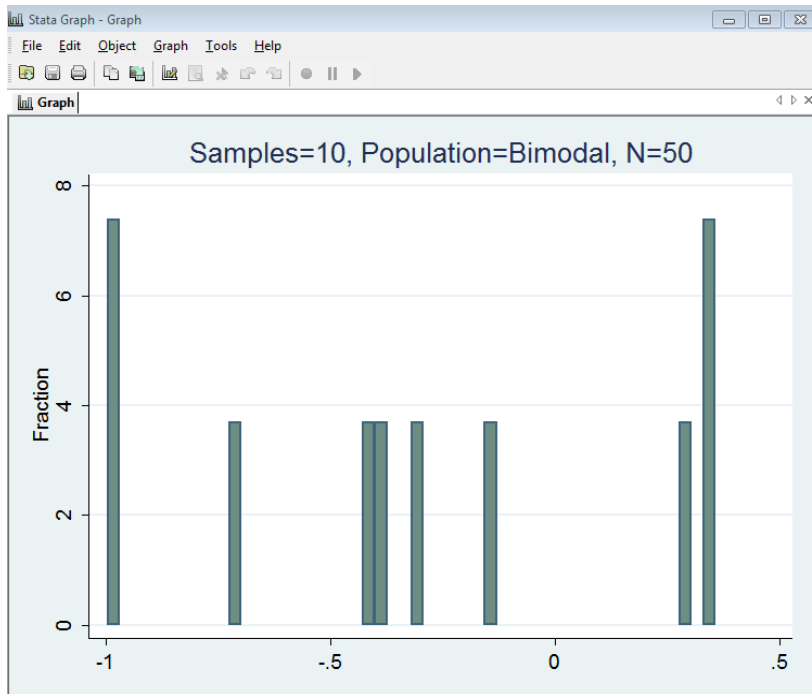
Type in “clt” into the STATA command editor, and you should be presented with the following box:

---

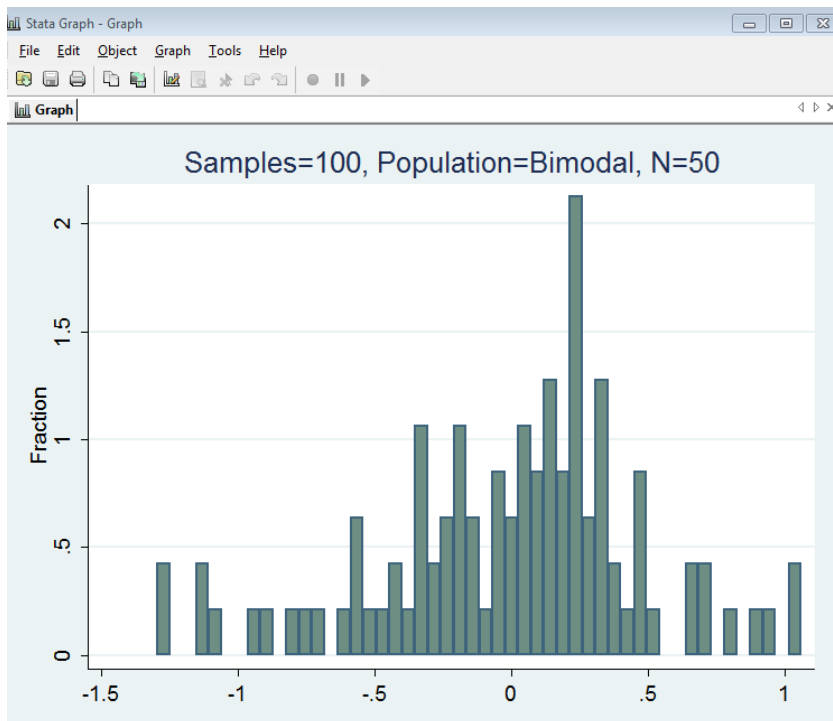
<sup>4</sup> Note, if you do not have the “clt” command in STATA, you will have to download it from UCLA. To do so, type in “findit clt” into the STATA command box. Click on the “clt from <http://www.ats.ucla.edu/stat/stata/ado/teach>” link, and then click on “Click here to install”.



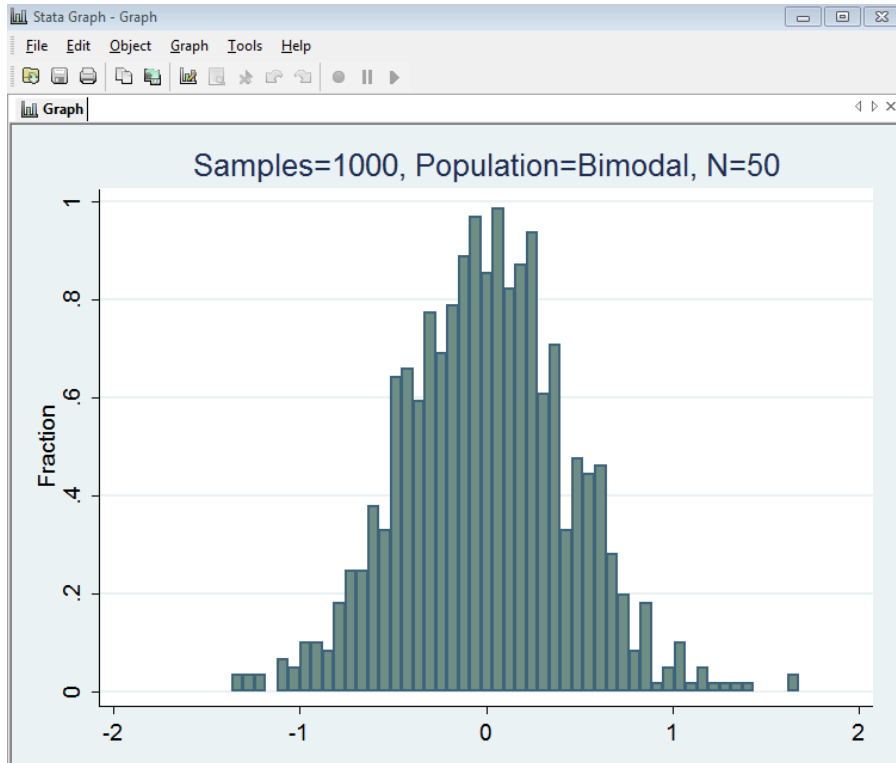
We are first going to simulate a random draw exercise with a small number of sub-samples (let's choose 10, and select 50 for the number of observations in our total sample). Let's specify to STATA that we want the distribution of our total sample to be bimodal (i.e. NOT normal) and we want to examine the distribution of the means for the 10 randomly drawn sub-samples. Enter 50 into "N per sample" and manually enter 10 into "# of samples" and click "Show". You should see some variant of the following histogram produced (note: it may not be identical as these are random draws):



Notice the histogram shows the 10 averages of our sub-samples (longer bars indicate that average value emerged in two of the ten sub-samples draws). The figure does not resemble a normal distribution in the slightest. Now, close the histogram box and go back to the CLT simulation box. This time, specify that you want 100 “# of samples” – in other words 100 random sub-sample draws. Click “Show” and you should see the following (again, it may not be identical given that these draws are randomly simulated):



Notice that the means of our sample draws, though not perfectly normally distributed, appear much more normally distributed than our first histogram. Let's repeat this command one more time with 1000 sample draws (enter 1000 into "# of samples"). Click "Show" and you should see a histogram that looks somewhat like the following diagram:



This histogram which plots the means of our 1000 sub-samples falls much more closely in line with what a normal distribution looks like. Comparing the distributions of sub-sample means between the three histograms, you should notice distribution convergence towards normal – this is the Central Limit Theorem in action! Also notice that it did not matter if our original sample had a bimodal (i.e. not normal) distribution – we still obtained a normal distribution for the mean of our sub-samples as that number of samples increases. This is one of the (many!) attractions of the Central Limit Theorem. The type of distribution of our original sample or population is irrelevant; as we increase the number of sub-sample draws, the means of these sub-samples will converge more closely to a normal distribution.

Moving on to the second descriptive statistic covered in this lesson, the variance is a measure of the spread of our data relative to the mean. While the variance is an important statistic, measures that we will use more commonly in regression analysis are the standard deviation and, especially, the standard error. The technical definition of the standard error is the standard deviation divided by the square root of our sample size. There is often confusion about whether the standard error and standard deviation can be used interchangeably. From a strict definition of terms, the standard deviation is a descriptive statistic, whereas the standard error describes bounds on a random sampling process (this is why standard errors are sometimes referred to as the standard deviation of the sampling distribution of the sample mean). Despite their similar nature, the

small difference in how the two are measured changes the meaning of what is being reported from a description of the variation in measurements (standard deviation) to a probabilistic statement about how the number of samples will provide a better bound on estimates of the population mean (standard error). In other words, the standard error is an estimate of how close to the population mean your sample mean is likely to be, whereas standard deviation is the degree to which individuals within the sample differ from the sample mean. Given that standard errors incorporate a (further) square root of the sample size in the denominator of its formulaic expression, standard errors will decrease with larger sample sizes. Standard deviations, on the other hand, will be unaffected by sample size.

Because samples are a representative subset of a population, they have slightly different descriptive statistic notations than populations. The table below provides a summary – in words, and formulaically – of these three sample statistics, as well as the median, which is the middle value of our data, if it is sorted in numerical order.

Table 2.1: Descriptive Statistics for Populations and Samples

DESCRIPTIVE STATISTIC	DEFINITION	SAMPLE NOTATION	POPULATION NOTATION
<i>Mean (aka average)</i>	Sum of all observations divided by the total number of observations	$\bar{X} = \frac{\sum(X)}{n}$	$\mu = \frac{\sum(X)}{n}$
<i>Median (aka middle value)</i>	If observations within a sample/population are ordered from lowest to highest, the median is the observation which divides the sample in half (i.e. the top 50% and the bottom 50%)	50 <sup>th</sup> percentile	50 <sup>th</sup> percentile
<i>Variance</i>	The average of the squared deviations about the mean (tell us how spread out our observations are relative to the mean)	$s^2 = \frac{\sum(X - \bar{X})^2}{n-1}$	$\sigma^2 = \frac{\sum(X - \mu)^2}{N}$
<i>Standard Deviation</i>	The square root of the variance	$s = \sqrt{\frac{\sum(X - \bar{X})^2}{n-1}}$	$\sigma = \sqrt{\frac{\sum(X - \mu)^2}{N}}$
<i>Standard Error</i>	Sample estimate of the population's standard deviation divided by the square root of the sample size	$se = \frac{s}{\sqrt{n}}$	NA

You may notice that populations and samples have slightly different variances and standard deviations: populations have a variance/standard deviation denominator of N and samples have a dominator of n-1. The reason for this is a technical one which we will not require you to go into in greater depth. To be

brief, it can be shown mathematically that if the variance in a random sample is used to estimate the variance of the population from which these data came, this estimate will, on average, be correct if we divide by  $n-1$ .

There are two things worth noting about standard deviations/standard errors. One is that standard deviations/standard errors will always be greater than or equal to zero. Secondly, the only occasion they will be equal to zero is when all our observations are identical and equal to the mean. The numerator of the variance and standard deviation is called the sum of squares. What this equates to is the sum of all the squared differences between each observation from the mean. If there is a greater spread, the distance between an observation and the mean will be larger, and hence the sum of squares, and consequently the variance, standard deviation and standard error, will be larger. If this spread is smaller, this distance between an observation and the mean will be smaller also, making the variance, standard deviation and standard error smaller. Only when each of our observations is equal to the mean will we obtain a zero sum-of-squares difference.

If we know the mean and standard error (i.e. the estimated standard deviation divided by the square root of our sample size) of our sample and our data is normally distributed, we will be able to predict where  $x\%$  of our data lie, in relation to the mean. While this is a very powerful prediction tool, it comes with strings attached: it's contingent on the size of the standard error. If we have a large standard error say 100, we can predict that 68% of our data will lie within a 200 unit range! This is not so helpful for prediction; a much smaller standard error (i.e. 10), can give us a much more precise range (i.e. 20 rather than 200) for where 68% of our data lie. You will find as we discuss t-testing, confidence intervals, and regression analysis, that the primary goal is to obtain a small standard error so you can gauge your results with greater accuracy. The easiest way to obtain a smaller standard error is to increase your sample size (notice from our standard error equation in Table 2.1, if  $n$  increases, the standard error will automatically decrease due to a larger denominator).

Large sample sizes are important for another reason in probability theory. According to a very central theorem in statistics, the law of large numbers, the mean from a large number of samples should approach the population mean. Large sample sizes are important, therefore, for two reasons. One, a mean of a large sample should be closer to that of the mean population of interest (i.e. greater resemblance). Two, the standard error of a large sample will be smaller, and hence more accurate (i.e. greater reliability). To prove the latter to you, the following lab will instruct you how to calculate these descriptive statistics in STATA, and how standard errors compare for our large sample of UK graduate earners ( $n=200$ ) versus our smaller sample ( $n=20$ ).

## STATA LAB (LESSON 2):

Open the two Excel spreadsheets with the 20/200 observations of UK graduate annual salaries for the first four years after graduation. Copy and paste both of these into two STATA windows (or if you saved these as .dta files, open the .dta files in STATA). If you did not save the .dta file, re-create the “totalearns” variable that we created in the last lab via the “gen” command (if you are still unfamiliar with this, refer back to the STATA commands from Lesson 1).

There are two ways you can calculate sample statistics for your data: manually, via typing the explicit command into STATA, or using the “Statistics” tab at the top of the page (this lesson will show you how to use both). Selecting the spreadsheet with only 20 UK graduates, you should start with the screen below (see the variable names in the variable box to the right):

2 - Stata/IC 10.1

File Edit Data Graphics **Statistics** User Window Help

Review

Command	_rc
1 edit	
2 gen totalearns = earns22 + ear...	
3 edit	

Variables

Name	Label	Type	Format
graduateid	Graduate ID	int	%8.0g
sex	Sex	str6	%9s
sexdummy	SexDummy	byte	%8.0g
earns22		float	%9.0g
earns23		float	%9.0g
earns24		float	%9.0g
earns25		float	%9.0g
totalearns		float	%9.0g

Results

STATA<sup>tm</sup> 10.1 Copyright 1984-2009  
StataCorp  
4905 Lakeway Drive  
College Station, Texas 77845 U  
800-STATA-PC http://www  
979-696-4600 stata@stat  
979-696-4601 (fax)

Single-user Stata for windows perpetual license:  
Serial number: 1910577806  
Licensed to: Alison Johnston  
Oregon State University

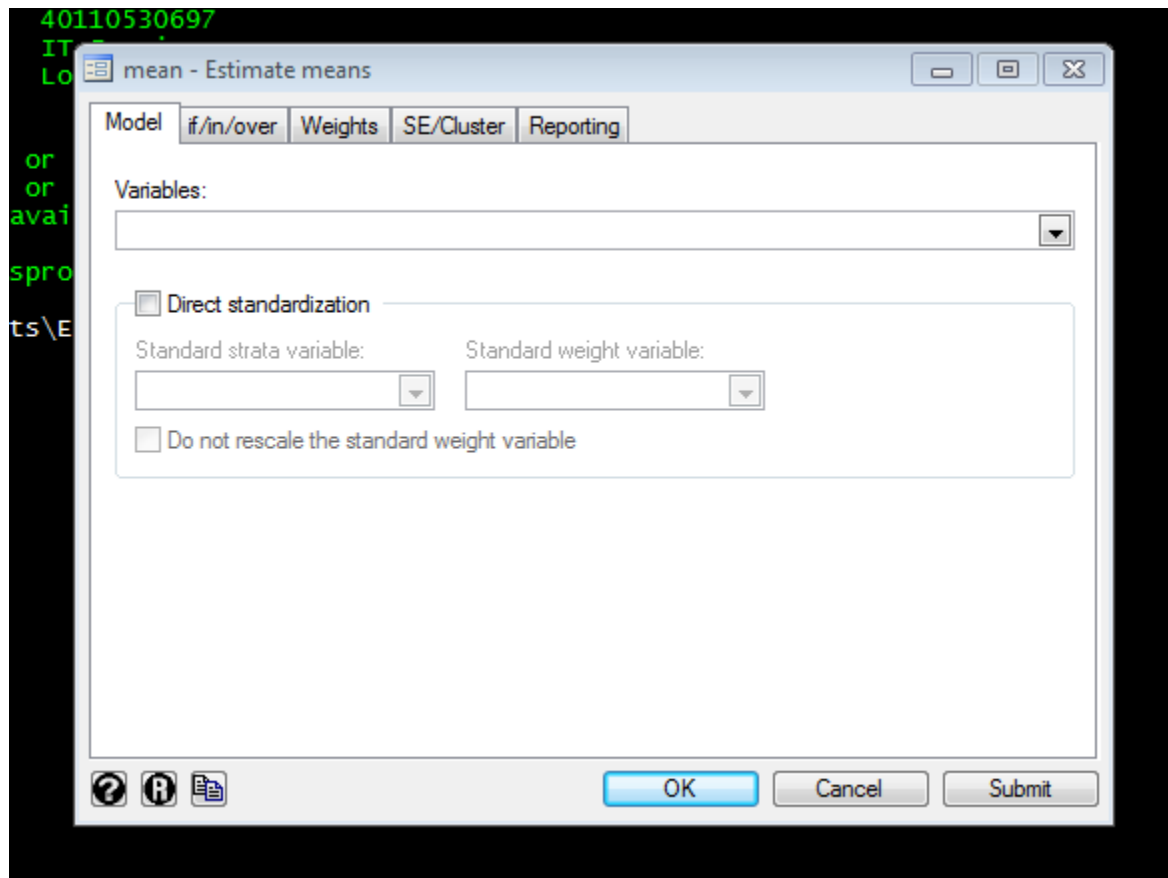
Notes:  
1. (/m# option or -set memory-) 1.00 MB allocated to data

. edit  
(7 vars, 20 obs pasted into editor)  
- preserve  
. gen totalearns = earns22 + earns23 + earns24 + earns25  
. edit  
- preserve  
.

Command

C:\data

Click on the “Statistics” table highlighted in red, then select “Summaries, Tables and Tests” then select “Summary and Descriptive Statistics” and then select “Means”. You should see the following box:



In the “Variables” box, click on the down arrow and you should find all your variables within the “Variables” box on the left hand side of the viewer. One convenient property of STATA, as opposed to Excel, is that you can compute descriptive statistics for multiple variables. Add earns22, earns23 and earns24 to the “Variables” box and click “Ok”. You should see the following output window.

. use "P:\Methods texts\Earnings Example.dta"				
. mean earns22 earns23 earns24				
Mean estimation		Number of obs = 20		
	Mean	Std. Err.	[95% Conf. Interval]	
earn22	13863.72	1953.028	9775.985	17951.45
earn23	21660.41	1957.19	17563.96	25756.85
earn24	25172.22	2806.427	19298.3	31046.14

**CONGRATULATIONS! You have just calculated the mean for multiple variables in STATA!**

Notice, that STATA also presents information on standard errors for our earnings data, as well as confidence intervals of the means (don't worry about this too much right now, we will approach this issue in a future lesson). In order to calculate standard deviations from the standard errors, you must multiply the standard errors by the square root of your sample size. In our case, the standard deviation of earns22, earns23 and earns24 would be as follows:

$$\text{Earns22} = 1953.028 * \sqrt{(20)} = 8,734.207$$

$$\text{Earns23} = 1957.19 * \sqrt{(20)} = 8,752.818$$

$$\text{Earns24} = 2806.427 * \sqrt{(20)} = 12,550.723$$

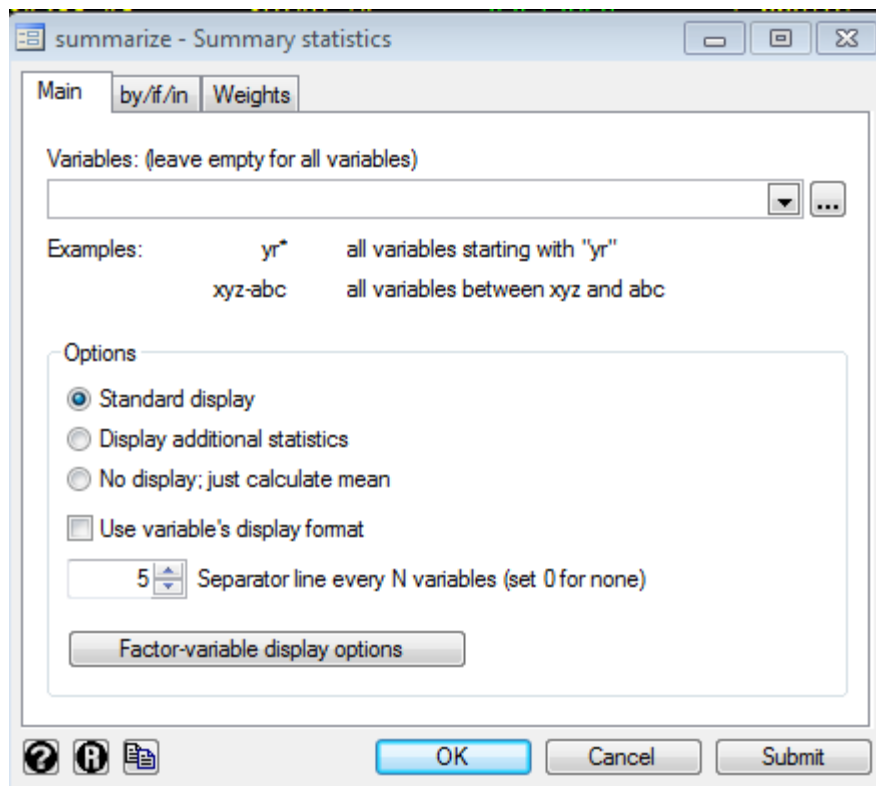
Whenever you run a command from toolbar tabs, STATA will automatically enter the code in your output. In the window above, notice the white text “mean earns22 earns23 earns24”; this is the typed command which generates means. Try calculating the mean again, except this time type the command “mean earns22 earns23 earns24” into the command box below the output screen. You should obtain the output that you see above.

### STATA COMMAND 2.1:

*Code:* “**mean var1 var2 var3...**”, where var1, var2, var3, ... are your variables of interest

*Output produced:* Calculates the mean of specified variables

Another command within STATA that will calculate means is the “Summarize” command. To find this command, click on the “Statistics” tab at the top of the viewer, then select “Summaries, Tables and Tests” then select “Summary and Descriptive Statistics” and then select “Summary Statistics”. You should view the following box:



Notice there are three options for your variables: “Standard display”, “Display additional statistics” and “No display; just calculate mean”. The “Standard display” button will present the following information for each variable: the total number of observations, the mean, the standard deviation (note, not the standard error!), the minimum value and the maximum value. The “Display additional statistics” option provides much more information on your variables. Enter `earn22`, `earn23` and `earn24` into the “Variables” box, click the “Display additional statistics” circle and then click “OK”. You should see the following output box:

. summarize earns22 earns23 earns24, detail				
earn22				
Percentiles		Smallest		
1%	0	0		
5%	0	0		
10%	0	0	Obs	20
25%	9612.765	0	Sum of wgt.	20
50%	15013.35			
75%	18469.5	Largest	Mean	13863.72
90%	25030.55	21091.45	Std. Dev.	8734.207
95%	28649.98	21558.6	Variance	7.63e+07
99%	28797.46	28502.5	Skewness	-.2634833
		28797.46	Kurtosis	2.405327
earn23				
Percentiles		Smallest		
1%	0	0		
5%	6774.835	13549.67		
10%	14542.08	15534.49	Obs	20
25%	17542.89	15564.2	Sum of wgt.	20
50%	20571.96			
75%	25581.98	Largest	Mean	21660.41
90%	34588.48	27931.92	Std. Dev.	8752.818
95%	39578.5	30140.74	Variance	7.66e+07
99%	40120.79	39036.21	Skewness	.1045912
		40120.79	Kurtosis	4.184555
earn24				
Percentiles		Smallest		
1%	0	0		
5%	6769.51	13539.02		
10%	13965.73	14392.44	Obs	20
25%	18741.27	16819.94	Sum of wgt.	20
50%	23559.33			
75%	30811.54	Largest	Mean	25172.22
90%	39876.52	34078.95	Std. Dev.	12550.72
95%	53017.85	34390.24	Variance	1.58e+08
99%	60672.89	45362.8	Skewness	.9519619
		60672.89	Kurtosis	4.944203

**CONGRATULATIONS!** You just calculated descriptive statistics for multiple variables in STATA!

Notice that the “summarize ..., detail” command presents not only the number of observations, mean, standard deviation (which are equal to those we calculated above from the standard errors), the min (first

listed observation in the “Smallest” column) and the max (last listed observation in the “Largest” column), but also information on the variance of each variable, the skewness, and the 1<sup>st</sup>, 5<sup>th</sup>, 10<sup>th</sup>, 25<sup>th</sup>, 50<sup>th</sup>, 75<sup>th</sup>, 90<sup>th</sup>, 95<sup>th</sup> and 99<sup>th</sup> percentile (the 50<sup>th</sup> percentile is the median). Also notice that like the “mean” application, STATA automatically presents the code of the summary command in white above the output.

## STATA COMMAND 2.2:

*Code:* “`summarize var1 var2 var3..., detail`”, where var1, var2, var3, ... are your variables of interest

*Output produced:* Calculates descriptive statistics of the specified variables.

Turning to our “totalearns” variable, we now rely upon both datasets to demonstrate that larger sample sizes produce smaller, and hence more accurate, standard errors. Calculate the mean for “totalearns” within your small dataset (n=20) and your large dataset (n=200). You should see the following output:

Small Dataset:

```
. mean totalearns
```

Mean estimation		Number of obs		=	20
	Mean	Std. Err.	[95% Conf. Interval]		
totalearns	87776.74	8555.378	69870.13	105683.4	

Large Dataset:

```
. mean totalearns
```

Mean estimation		Number of obs		=	200
	Mean	Std. Err.	[95% Conf. Interval]		
totalearns	75994.1	2944.111	70188.44	81799.76	

Unfortunately, we are unable to obtain information about the “true” population mean for earnings in the first four years after graduation because it is impossible to survey all UK university graduates. Hence, we

have to accept the law of large numbers prediction that the mean from our 200 graduate sample will be closer to the true population mean than the 20 graduate sample. We can assess, however, is how larger sample sizes impact our standard error. Notice which sample has the smaller standard error – that which contains 200 graduates. This is not a coincidence; the lab exercises that you do below will demonstrate that for every comparable variable between these two datasets, our 200 graduate sample will always have a smaller, and hence more accurate for description purposes, standard error. This is because as our sample size increases the denominator of our standard error equation becomes larger, and our total standard error becomes smaller.

---

### Practice Problems:

Practice Problem 1: Calculate variance, the median, and the 25<sup>th</sup> and 75<sup>th</sup> percentile of `earn25` within the small dataset (20 graduates) and large dataset (200 graduates).

Practice Problem 2: If you did not save the data-file and variables created in the previous, re-create the `aveearn` variable, which is the average earnings of a UK graduate in the 4 years after leaving university, using the “`gen`” command. Calculate the mean, standard error, and standard deviation of this variable for both the large and small sample datasets. What do you notice about the standard error between the two datasets?

Practice Problem 3: Compare the standard errors of `earn22`, `earn23`, `earn24` and `earn25` between our small and large sample of UK graduates. Is the trend consistent with what you witnessed above? What do you notice about the range of the confidence intervals of our means between the two samples?

Practice Problem 4: Create any variable from the earnings variables in both dataset. Test again, whether the standard error is smaller in the large dataset, regardless of the variable.

## Lesson 3: Cross-Tabulations

---

**Learning Objective 1: Understanding the use for and construction of cross-tabs**

**Learning Objective 2: Understanding notions of contingency and independence between two categorical variables**

**Learning Objective 3: Creating cross-tabulation tables in STATA**

**Learning Objective 4: Determining contingency/independence of two categorical variables in STATA**

Some researchers may be interested in data that is categorical (i.e. whose value describes a grouping, and is meaningless when described on a numerical scale) rather than a numerical variable. Say for example, you are interested in categorical response outcomes (i.e. “yes”, “no”, “maybe”, or whether someone is a “Democrat”, “Republican” or “Independent”) rather than a count variable (i.e. income or age) which can be assessed for its numerical value. If this is the case, descriptive statistics may be less meaningful for your data, especially if your categories cannot be ordered. This does not mean that people who use categorical data cannot use statistical techniques. There are regression methods one can use to assess the impact of a categorical (independent) variable on a numerical dependent variable, or the impact of numerical independent variables on a categorical (dependent) variable. We will cover some of these techniques, specifically binary and ordinal logistic regression, in later lessons. In this lesson, however, we begin with cross-tabulations, or cross-tabs.

Cross-tabs are to categorical variables what descriptive statistics are to numerical variables; they outline summary data by category, and what percentage (i.e. frequency) of your data lies within a particular category. One nice thing about cross-tabs, compared to sample statistics, is that it enables us to understand the distribution of categorical data over two variables rather than just one. Cross tabulations are contingency tables, and enable one to explore the relationship between (normally just) two categorical variables. A cross-tab table will show you the joint frequency distribution of the two variables, and whether such frequency is different within different categories.

To give you an example, consider the following observations on political party affiliation by gender listed below:

Table 3.1: Party-affiliation by gender: A hypothetical sample

<i>Gender</i>	<i>Party-Affiliation</i>
Male	Democrat
Female	Republican
Female	Democrat
Male	Democrat
Female	Republican
Male	Democrat
Male	Republican
Male	Republican
Female	Republican
Female	Republican

If we want to determine the political spread of individuals by gender, we do so by creating a cross-tab. When constructing a cross-tab, it is helpful to designate your dependent variable (i.e. the variable you wish to explain) as the row variable, and your independent variable (i.e. the variable you believe may influence your dependent variable) as the column variable. Since a person's gender is pre-determined, let's select party affiliation as the dependent (row) variable and gender as the independent (column) variable.

In order to create a cross-tab, we need to allocate our observations in Table 3.1 above in the four following categories: "female republican", "female democrat", "male republican" and "male democrat". Based upon these four groupings, and information in Table 3.1, we can create the following cross-tab:

Table 3.2: Cross-tab of party-affiliation by gender

	<i>Males</i>	<i>Females</i>	<i>Total</i>
<i>Democrat</i>	3	1	4
<i>Republican</i>	2	4	6
<i>Total</i>	5	5	10

Notice that the interior four cells tell us the frequency (i.e. number) of individuals within our sample which fit under the specified category. The row total (numbers in red) outlines a frequency summary of our column variables (i.e. our sample has 5 men and 5 women), and the column total (numbers in blue) outlines a frequency summary of our row variables (i.e. our sample has 4 Democrats and 6 Republicans). Cross-tabs are convenient for summary purposes, because rather than mulling through each individual data-observation, you can examine the number of observations that lie within each category; they are especially handy for summarizing categorical data if your sample is large.

Another convenient property of cross-tabs is that we can test the degree of association between two categorical variables. Within a cross-tab, categorical variables display either contingency (i.e. the distributional frequency of an outcome is different across groups) or independence (i.e. the distributional

frequency of an outcome is similar across groups). For the above example, our cross-tab would indicate that gender and party-affiliation are contingent if men are significantly more likely to fall into one political party compared to women. However, gender and party-affiliation would be independent, if the distribution of men and women was roughly similar for both political parties. To assess contingency/independence between two categorical variables, we rely upon a Pearson's Chi-squared statistic. The use and calculation of this statistic will be explained in greater detail in the STATA lab below.

## STATA LAB (LESSON 3):

---

For this exercise, we will use a different dataset than the one from the previous lessons. Open up “facebooksurvey” Excel dataset. This online survey asked a number of questions to Trinidadian users of Facebook regarding their use of the social networking site to maintain contact with family, friends and fictive kin in the international Trinidadian diaspora. The dataset you see is a condensed version, with the following variables:

1. The ID of the survey respondent
2. The gender of the respondent codified as a dummy variable: 0 for female, 1 for male
3. The age category of the respondent codified as a categorical variable: 1 for 18-24, 2 for 25-34, 3 for 35-44, 4 for 45-54, 5 for 55-64, and 6 for 65+
4. Whether the respondent has experienced discrimination because of his/her color or race in their current country of residency codified as a categorical variable: 1 for maybe, 2 for no, and 3 for yes).
5. The number of hours the respondent spends online in a regular day
6. The number of years the respondent has been using the internet
7. The number of Facebook friends the respondent has.

Upload this dataset from Excel into STATA (via the copy/paste method you used in Lesson 2). You should see the following screen:

individualid[1] = 1

	individualid	gender	genderdummy	age	agecat	discrim	discrimcat	hoursonline	yearsusing-t	numberfreis
1	1	M	1	25-34	2	M	1	10	18	297
2	2	M	1	35-44	3	M	1	1.5	18	70
3	3	M	1	35-44	3	M	1	1	15	205
4	4	M	1	35-44	3	M	1	1.5	18	70
5	5	F	0	45-54	4	M	1	2	7	35
6	6	F	0	18-24	1	M	1	5	13	182
7	7	F	0	18-24	1	M	1	2	10	495
8	8	F	0	35-44	3	M	1	2	12	423
9	9	F	0	45-54	4	M	1	2	16	305
10	10	M	1	25-34	2	M	1	4	20	350
11	11	F	0	35-44	3	M	1	2	17	330
12	12	F	0	25-34	2	N	2	5	12	2125
13	13	F	0	35-44	3	N	2	5	20	117
14	14	F	0	45-54	4	N	2	1	17	150
15	15	F	0	45-54	4	N	2	1	10	100
16	16	F	0	45-54	4	N	2	1	17	150
17	17	M	1	25-34	2	N	2	3	12	500
18	18	M	1	25-34	2	N	2	10	8	300
19	19	F	0	25-34	2	N	2	1	13	80
20	20	F	0	25-34	2	N	2	2	10	280
21	21	M	1	25-34	2	N	2	3	12	153
22	22	M	1	35-44	3	N	2	3	15	1300

Notice that the un-coded gender, age, and discrimination variable show up in red. In Pre-Lab 2, you were informed that STATA does not recognize non-numerical values including symbols such as -, +, \*, \, /, |, etc., for statistical purposes. Due to this, you will be unable to calculate sample statistics or advanced forms of statistical analysis on these variables in STATA. You can, however, make cross-tabulations with categorical names in STATA; this is because cross-tabs summarize the frequency of your data, rather than computing their statistical properties.

Click the “Preserve/Save” button and then click the red box in the top right hand corner to upload the dataset into STATA. Turning to cross-tabulations of two categorical variables, let’s examine whether experiences with discrimination significantly differ by age group. To create a cross-tab of these two variables, click the “Statistics” tab in the STATA toolbar, then “Summaries, tables and tests”, then “Tables”, then “Two way tables with measures of association”. You should see the following box:

tabulate2 - Two-way tables

Main by/if/in Weights Advanced

Row variable: discrim

Column variable: age

Test statistics

- ☐ Pearson's chi-squared
- ☐ Fisher's exact test
- ☐ Goodman and Kruskal's gamma
- ☐ Likelihood-ratio chi-squared
- ☐ Kendall's tau-b
- ☐ Cramer's V

Cell contents

- ☐ Pearson's chi-squared
- ☐ Within-column relative frequencies
- ☐ Within-row relative frequencies
- ☐ Likelihood-ratio chi-squared
- ☐ Relative frequencies
- ☐ Expected frequencies
- ☐ Suppress frequencies

☐ Treat missing values like other values

☐ Do not wrap wide tables

☐ Show cell contents key

☐ Suppress value labels

☐ Suppress enumeration log

OK Cancel Submit

Since we want to assess whether age, our independent variable, produces different experiences of discrimination (our dependent variable), place the “discrim” variable in the “Row” box and the “age” variable in the “Column” box. Then click “Ok” and you should see the following output.

```
. tabulate discrim age
```

Discrim	18-24	25-34	35-44	45-54	55-64	65+	Total
M	2	2	5	2	0	0	11
N	8	18	15	12	1	0	54
Y	2	13	7	7	3	3	35
Total	12	33	27	21	4	3	100

**CONGRATULATIONS! You have just created a cross-tab of two categorical variables in STATA!**

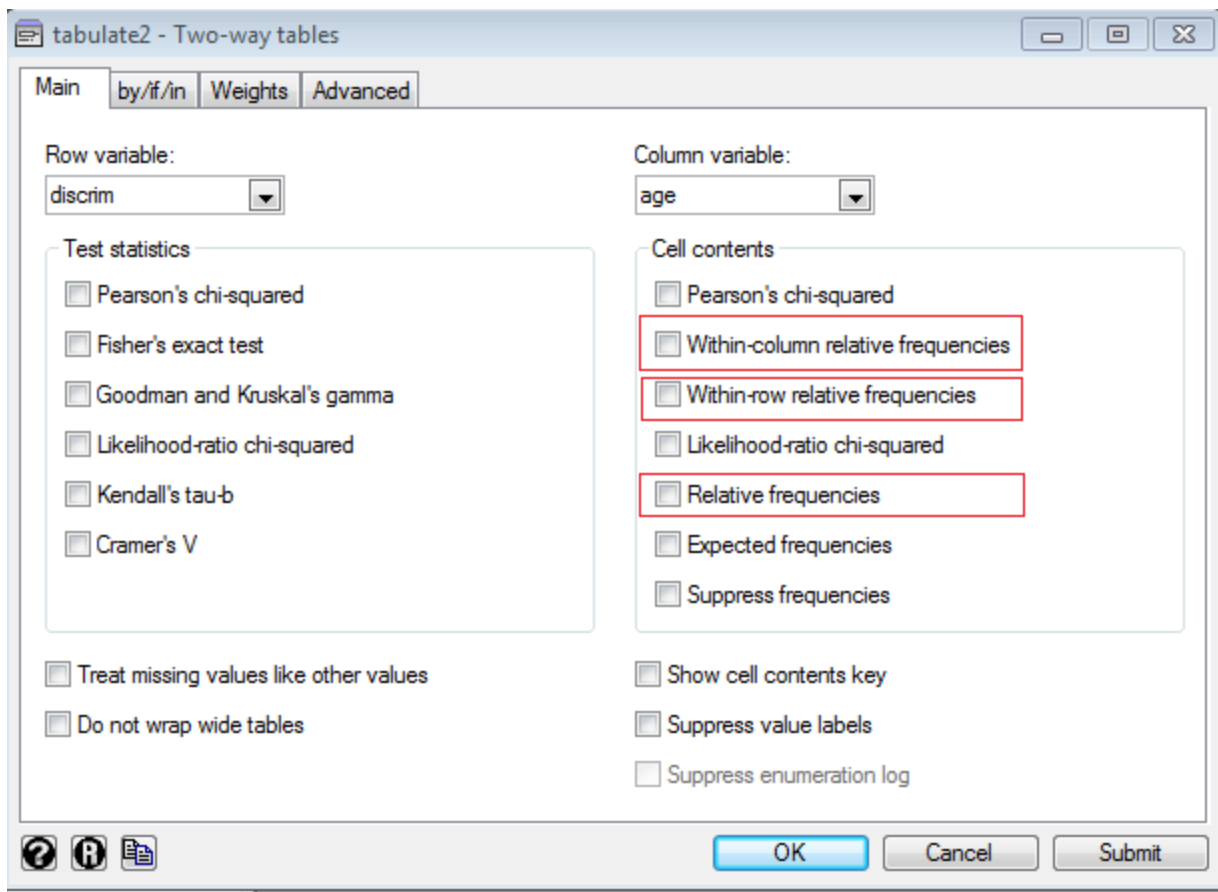
This cross-tab indicates the number of experiences of discrimination by age group. The table above can also be produced by typing the command below into the command text box.

### STATA COMMAND 3.1:

*Code:* “**tabulate var1 var2**”, where var1 is your dependent variable and var2 is your independent variable

*Output produced:* Calculates a cross-tab of the specified variables.

STATA can also generate frequencies (i.e. percentages) of the data in each cell. Going back to the “Tabulate two-way table” box, notice there are the following frequency options (boxed in red below): within-column relative frequencies, within-row relative frequencies, and relative frequencies.



Go back to the “Two way tables with measures of association” (accessible via “Statistics” then “Summaries, tables and tests”, then “Tables”), and click the “Within-column relative frequency” option. You should see the following output:

```
. tabulate discrim age, column
```

Key							
		frequency					
		column percentage					
		Age					
Discrim		18-24	25-34	35-44	45-54	55-64	65+
M		2 16.67	2 6.06	5 18.52	2 9.52	0 0.00	0 0.00
N		8 66.67	18 54.55	15 55.56	12 57.14	1 25.00	0 0.00
Y		2 16.67	13 39.39	7 25.93	7 33.33	3 75.00	3 100.00
Total		12 100.00	33 100.00	27 100.00	21 100.00	4 100.00	3 100.00

**CONGRATULATIONS! You have just created a column-frequency cross-tab of two categorical variables in STATA!**

This cross-tab demonstrates the frequency (in percent) of discrimination experiences for each age group. Among other things, this cross-tab informs us that 16.67% of survey respondents in the 18-24 age group experienced discrimination, while 75% of survey respondents in the 55-64 age group experienced discrimination. We can also produce similar cross-tab frequencies by rows, which would show us how each discrimination experience broke-down by age category. Or, we can generate cross-tab frequencies for the entire sample (this is the relative frequency command). These frequency tables can also be produced by typing the commands below into the command text box:

### STATA COMMAND 3.2:

*Code:* “**tabulate var1 var2, column**”, where var1 is your dependent variable and var2 is your independent variable

*Output produced:* Calculates a cross-tab of the specified variables with frequency percentages by column.

### **STATA COMMAND 3.3:**

*Code:* “**tabulate var1 var2, row**”, where var1 is your dependent variable and var2 is your independent variable

*Output produced:* Calculates a cross-tab of the specified variables with frequency percentages by row.

### **STATA COMMAND 3.4:**

*Code:* “**tabulate var1 var2, cell**”, where var1 is your dependent variable and var2 is your independent variable

*Output produced:* Calculates a cross-tab of the specified variables with frequency percentages for the total sample.

As mentioned above, when we generate cross-tabs, we are generally interested in determining whether the two variables display contingency (i.e. whether the proportion of discrimination experiences vary by age group) or independence (i.e. whether the proportion of discrimination experiences is relatively similar by age group and hence there is no association between the two). To assess this, we can rely upon the Pearson’s Chi-squared statistic, one of the most common measures of statistical significance. Re-tabulate the cross tab using the following command below:

### **STATA COMMAND 3.5:**

*Code:* “**tabulate var1 var2, chi2**”, where var1 is your dependent variable and var2 is your independent variable

*Output produced:* Calculates a cross-tab of the specified variables along with the Pearson Chi squared statistic.

After typing “tabulate discrim age, chi2” into the command box, you should be presented with the following output:

```
. tabulate discrim age, chi2
```

Discrim	18-24	25-34	35-44	45-54	55-64	65+	Total
M	2	2	5	2	0	0	11
N	8	18	15	12	1	0	54
Y	2	13	7	7	3	3	35
Total	12	33	27	21	4	3	100

Pearson chi2(10) = 13.3694 Pr = 0.204

**CONGRATULATIONS! You just conducted a Pearson Chi<sup>2</sup> test of independence in STATA!**

Notice the Pearson Chi<sup>2</sup> statistic result at the bottom of the table. This value tells us whether the two variables display contingency or independence. If the statistic produces a p-value less than 0.100, we can reject the null hypothesis that the two variables display independence. In our case, we cannot reject the null hypothesis. This indicates that there is a chance that these two variables are independent, meaning that they have no relationship with each other. If our p-value were smaller than 0.100, we could reject (with 90% confidence or higher) that there is no association between the row and column variable.

---

### **Practice Problems:**

Practice Problem 1: Generate a generic cross-tab (i.e. without frequencies) of the association of gender and experiences of discrimination; use gender as the independent variable and discrimination as the independent variable.

Practice Problem 2: Generate a cross-tab of the association of gender and experiences with information about discrimination frequency by gender (i.e. if gender is the independent variable, display relative-frequency by columns)

Practice Problem 3: Assess the likelihood of whether there is a significant associational relationship between gender and discrimination in the sample (i.e. whether a greater proportion of Trinidadian women have/may have experienced discrimination than men).

## Lesson 4: Significance Testing

---

**Learning Objective 1: To understand the research design of significance testing**

**Learning Objective 2: To understand how to calculate a test statistic, and use its corresponding p-value to determine whether to reject/fail-to-reject a null hypothesis**

**Learning Objective 3: To understand the construction of confidence intervals**

**Learning Objective 4: To understand how to manually calculate a t-statistic and p-value from descriptive statistics calculated in STATA to test the validity of a null hypothesis**

**Learning Objective 5: To understand how to conduct significance testing of a null hypothesis against a one-tailed and two-tailed alternative hypothesis in STATA**

**Learning Objective 6: To be able to calculate confidence intervals of means within STATA**

In statistics and econometrics, one thing that you will realize is that the significance of your results (i.e. the ability for you to reject a hypothesis with at least 90% certainty) is everything. Unless you are testing a well-established theory on empirical data, policy-makers/researchers/funding-councils are less interested in the identification of factors that lack a significant influence of a dependent variable. Rather they are more interested in factors that significantly impact a dependent variable. A significance test is a technique of statistical inference that is used to examine specific claims about the values of population parameters from a sample. Such a test provides an assessment of whether it is plausible, given the evidence (i.e. mean, standard error, sample size) of the observed data, that a population parameter has a particular value specified by the researcher.

In this section we are going to focus specifically on significance testing for the values of our sample's basic parameters, specifically means. In Lesson 5, we will extend significance testing to differences in means, one of the preliminary forms of data analysis you will do before you learn regressions. In later lessons, we will continuously use significance testing for all methods we employ, including linear regression which estimates the effects of independent (explanatory) variables on dependent variables (i.e. the variable to be explained). Significance testing relies upon 5 basic building blocks, each of which is discussed below:

1. A null hypothesis ( $H_0$  of the test), and an alternative hypothesis ( $H_A$ ) against which  $H_0$  is to be tested.
2. A t-statistic to assess the validity of the null hypothesis
3. A normal sampling distribution
4. A corresponding p-value to the t-statistic, which tells us the plausibility of our null hypothesis
5. A conclusion based upon our p-value on whether we should reject or not reject the null hypothesis at a stated significance level

### Null and Alternative Hypotheses:

The null hypothesis is a specific claim about the population, which we will test from our sample data that is representative of the population. Let's depart from hypothesis testing of the mean of our sample. A null-hypothesis could be any statement about our mean to be tested. For example, say in our sample of 20 British graduate earners we want to test whether the mean earnings in the first 4 years out of college is 70,000. In this case, we would write our null hypothesis as follows:

$$H_0: \bar{X} = 70,000$$

For hypothesis testing of basic parameters (i.e. means) the arbitrary value we pick for the null hypothesis generally does not matter. However, you will notice that as we cover difference of means testing (Lesson 5) and regression analysis, we generally select 0 as our null hypothesis (i.e. that there is no difference in means of two populations/sub-populations or, in the case of regression analysis, the independent variable has no effect on the dependent variable). We will revisit this again in Lesson 5 and later when we cover regression analysis, so don't worry too much now if this is unclear.

All null hypotheses are tested against an alternative hypothesis. Once you've identified your null hypothesis, it's relatively simple to state your alternative hypothesis, which basically describes the scenario that the null hypothesis is not true. Both your null and alternative hypotheses should exhaust all numerical possibilities of what your mean (or in regression analysis, beta coefficient) can equal. There are two types of alternative hypotheses: two-sided and one-sided. A two-sided alternative hypothesis states explicitly that your parameter is not equal to your null value. In other words, relating to our example above, both hypotheses can be written as follows:

$$\begin{aligned} H_0: \bar{X} &= 70,000 \\ H_A: \bar{X} &\neq 70,000 \end{aligned}$$

For a one-sided alternative hypothesis, however, we would seek to test only if our mean value is above/below the value stated in our null-hypothesis. Hence, let's say we want to set up an alternative hypothesis, where we think that British graduates make more than 70,000 in their first four years out of university, against the null hypothesis that they make the equivalent of 70,000 or less (remember, null and alternative hypotheses should be exhaustive of all numerical possibilities). Our two hypotheses would be written as follows:

$$\begin{aligned} H_0: \bar{X} &\leq 70,000 \\ H_A: \bar{X} &> 70,000 \end{aligned}$$

Alternatively, if we wanted to test whether British graduates, after four years of work, make less than 70,000 in total, our two hypotheses would be written as follows:

$$\begin{aligned} H_0: \bar{X} &\geq 70,000 \\ H_A: \bar{X} &< 70,000 \end{aligned}$$

When we cover differences-in-means and regression analysis, we will predominantly focus on two-sided alternative hypotheses (i.e. whether the true value is not equal to zero). We introduce both to you for reference and will discuss both in the corresponding STATA lab.

### Test-Statistics/Normal Sampling Distribution:

The estimator to test the validity of our null hypothesis is our test statistic (also known as our t-statistic). Significant testing relies on the crucial assumption that the t-statistic is normally distributed. This is a very important assumption, as the t-statistic, like any distribution, provides a map for when we should reject an arbitrarily selected mean value and when we should not.

A test-statistic can be calculated as follows:

$$T-stat = \frac{\mu - \bar{X}}{se}$$

Where  $\mu$  is the calculated mean of your sample,  $\bar{X}$  is the arbitrary value specified by your null hypothesis (note, in our example above,  $\bar{X}$  would be 70,000), and “se” is the calculated standard error of your sample. The t-value measures how many standard errors  $\bar{X}$ , your predicted mean value, is away from your sample mean. The farther away the t-value is from zero (either positive or negative) the greater the likelihood that  $\bar{X}$  is from the “true” sample mean,  $\mu$ , and hence the greater the evidence there is against the null hypothesis.

Larger t-values indicate a greater likelihood of rejecting the null hypothesis. Recall from Lesson 2, that

our standard error can be written as  $se = \frac{s}{\sqrt{n}}$ . The influence of the law of large numbers indicates that

as your sample size increases, the denominator of your t-stat will decrease and hence the entire t-stat will increase. Put otherwise, we are likely to obtain larger, more precise, t-statistics with larger rather than smaller sample sizes. Another nice thing about larger versus smaller sample sizes, is that the critical t-value which we test our hypothesis (discussed in our p-value section below) becomes smaller as our number of observations increase. Hence, reiterating the importance of large sample sizes, as our number of observations increases, the test-statistic we calculate will increase, and the critical value which we compare this against will decrease, increasing the likelihood of rejecting our null hypothesis. This is a good thing because when we cover regression analysis and difference in means, where our null hypothesis will be equal to zero, this means we will be able to provide evidence that sample means are different, or, in the case of regression analysis, that there is evidence for some influence of the independent variable on our dependent variable.

### P-values and critical t-values:

You now know how to calculate a t-statistic, but are probably wondering, “how do I know whether to reject my null hypothesis or not?”. The calculation of a t-statistic is useless unless you have a critical value with which to compare it against; this is where t-tables (pictured in Figure 4.1 below – notice the normal distribution...) and p-values come in. The t-table below is a standard reference for hypothesis

testing (keep one with you for reference!), and helps you determine the critical t-value which to compare your t-statistic against. For significance testing, if our t-statistic is greater than our critical value, we reject our null hypothesis. If our t-statistic is less than our critical value, we fail to reject our null hypothesis.

In order to determine your critical value in the t-table below, you need two pieces of information: 1) your degrees of freedom (listed in the first column), which is your sample size minus the number of parameters we wish to estimate (since we are only doing a means test above, our number of parameters is only 1; hence for our 20 student example sample, our degrees of freedom would be  $20-1=19$ ), and; 2) the level of confidence you want to test your hypothesis against, which is listed in the final row of the table below (these confidence intervals are for two-tailed alternative hypothesis tests). Higher levels of confidence (i.e. those nearer to 100%) reduce your probability of falsely rejecting a null hypothesis when it may be true. As a rule of thumb in econometrics, you should **never** choose critical values for confidence intervals below 90%.

Using our example of the mean test of British graduates' total earnings four years out of school, our 19 degrees of freedom indicate that we would need a t-statistic of 1.729 or higher to reject our null hypothesis that average total earnings are 70,000 with 90% confidence. Similarly, we need a t-statistic of 2.093 to reject our null hypothesis with 95% confidence and a t-statistic of 2.861 to reject our null hypothesis with a 99% confidence level. Notice that in order to increase confidence in rejecting our null hypothesis, we require a larger critical-value to compare our t-statistic against. This is why larger t-statistics (which are more likely with larger samples) are so beneficial – they enhance the confidence in which we can reject our null hypothesis! Another important discovery you may have noticed with the t-table below is what happens to our critical values as our sample size (and hence degrees of freedom) increases: the critical-values decrease. As mentioned before, larger sample sizes are beneficial for two reasons in hypothesis testing; they increase our t-statistic, and they decrease our critical value, hence enhancing our ability to reject our null hypothesis.

P-values, the probability of rejecting a null hypothesis in a sample when in fact it is true within the population, are automatically tied to your confidence interval. One way to think about a p-value is that it is the difference between 100% and your confidence interval. If we produce a t-statistic that perfectly aligns with a 90% critical value, our p-value would be 0.100. If we produced a t-statistic that perfectly aligns with a 97% critical value, our p-value would be 0.03. Since p-values are essentially the inverse of confidence levels, we want to produce a test-statistic whose comparison against a critical value will produce a low p-value (i.e. a p-value less than 0.1). Lower p-values imply that there is strong evidence against the null hypothesis. P-values have their own tables, but generally require tedious conversions in order to be absolutely precise. Consequently, we will rely heavily on STATA to compute p-values for us.

Figure 4.1 T-table critical values

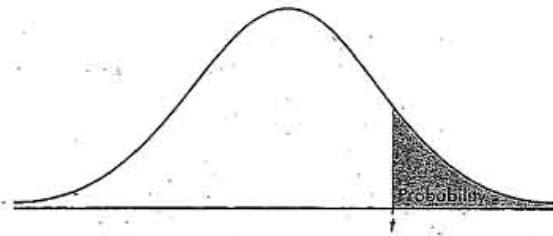


TABLE B: t-DISTRIBUTION CRITICAL VALUES

df	Tail probability $p$											
	.25	.20	.15	.10	.05	.025	.02	.01	.005	.0025	.001	.0005
1	1.000	1.376	1.963	3.078	6.314	12.71	15.89	31.82	63.66	127.3	318.3	636.6
2	.816	1.061	1.386	1.886	2.920	4.303	4.849	6.965	9.925	14.09	22.33	31.60
3	.765	.978	1.250	1.638	2.353	3.182	3.482	4.541	5.841	7.453	10.21	12.92
4	.741	.941	1.190	1.533	2.132	2.776	2.999	3.747	4.604	5.598	7.173	8.610
5	.727	.920	1.156	1.476	2.015	2.571	2.757	3.365	4.032	4.773	5.893	6.869
6	.718	.906	1.134	1.440	1.943	2.447	2.612	3.143	3.707	4.317	5.208	5.959
7	.711	.896	1.119	1.415	1.895	2.365	2.517	2.998	3.499	4.029	4.785	5.408
8	.706	.889	1.108	1.397	1.860	2.306	2.449	2.896	3.355	3.833	4.501	5.041
9	.703	.883	1.100	1.383	1.833	2.262	2.398	2.821	3.250	3.690	4.297	4.781
10	.700	.879	1.093	1.372	1.812	2.228	2.359	2.764	3.169	3.581	4.144	4.587
11	.697	.876	1.088	1.363	1.796	2.201	2.328	2.718	3.106	3.497	4.025	4.437
12	.695	.873	1.083	1.356	1.782	2.179	2.303	2.681	3.055	3.428	3.930	4.318
13	.694	.870	1.079	1.350	1.771	2.160	2.282	2.650	3.012	3.372	3.852	4.221
14	.692	.868	1.076	1.345	1.761	2.145	2.264	2.624	2.977	3.326	3.787	4.140
15	.691	.866	1.074	1.341	1.753	2.131	2.249	2.602	2.947	3.286	3.733	4.073
16	.690	.865	1.071	1.337	1.746	2.120	2.235	2.583	2.921	3.252	3.686	4.015
17	.689	.863	1.069	1.333	1.740	2.110	2.224	2.567	2.898	3.222	3.646	3.965
18	.688	.862	1.067	1.330	1.734	2.101	2.214	2.552	2.878	3.197	3.611	3.922
19	.688	.861	1.066	1.328	1.729	2.093	2.205	2.539	2.861	3.174	3.579	3.883
20	.687	.860	1.064	1.325	1.725	2.086	2.197	2.528	2.845	3.153	3.552	3.850
21	.686	.859	1.063	1.323	1.721	2.080	2.189	2.518	2.831	3.135	3.527	3.819
22	.686	.858	1.061	1.321	1.717	2.074	2.183	2.508	2.819	3.119	3.505	3.792
23	.685	.858	1.060	1.319	1.714	2.069	2.177	2.500	2.807	3.104	3.485	3.768
24	.685	.857	1.059	1.318	1.711	2.064	2.172	2.492	2.797	3.091	3.467	3.745
25	.684	.856	1.058	1.316	1.708	2.060	2.167	2.485	2.787	3.078	3.450	3.725
26	.684	.856	1.058	1.315	1.706	2.056	2.162	2.479	2.779	3.067	3.435	3.707
27	.684	.855	1.057	1.314	1.703	2.052	2.158	2.473	2.771	3.057	3.421	3.690
28	.683	.855	1.056	1.313	1.701	2.048	2.154	2.467	2.763	3.047	3.408	3.674
29	.683	.854	1.055	1.311	1.699	2.045	2.150	2.462	2.756	3.038	3.396	3.659
30	.683	.854	1.055	1.310	1.697	2.042	2.147	2.457	2.750	3.030	3.385	3.646
40	.681	.851	1.050	1.303	1.684	2.021	2.123	2.423	2.704	2.971	3.307	3.551
50	.679	.849	1.047	1.299	1.676	2.009	2.109	2.403	2.678	2.937	3.261	3.496
60	.679	.848	1.045	1.296	1.671	2.000	2.099	2.390	2.660	2.915	3.232	3.460
80	.678	.846	1.043	1.292	1.664	1.990	2.088	2.374	2.639	2.887	3.195	3.416
100	.677	.845	1.042	1.290	1.660	1.984	2.081	2.364	2.626	2.871	3.174	3.390
1000	.675	.842	1.037	1.282	1.646	1.962	2.056	2.330	2.581	2.813	3.098	3.300
$\infty$	.674	.841	1.036	1.282	1.645	1.960	2.054	2.326	2.576	2.807	3.091	3.291
	50%	60%	70%	80%	90%	95%	96%	98%	99%	99.5%	99.8%	99.9%
	Confidence level $C$											

### Conclusion:

After you have calculated your t-statistic, and compared it against its critical value within the standard t-table, you are ready to make a conclusion about your null hypothesis. You should always mention your confidence level/p-value with your conclusions, in order to give readers an idea of the significance of your results. Mentioned above, if you produce a high t-statistic that exceeds your critical value on a 90%/95%/99% confidence level, you can conclude that “the null hypothesis of British graduates earning, on average, 70,000 in their first four years out of college can be rejected with 90%/95%/99% confidence”. Likewise, if your t-statistic is less than the 90% critical value, you should state “the null hypothesis of British graduates earning, on average, 70,000 in their first four years out of college cannot be rejected with over 90% confidence”. Notice, that in the case of the latter, we do not automatically accept the null hypothesis. Statisticians and econometricians are very careful in their language with hypothesis testing, especially when it is conducted on samples which may hold bias (refer to Lesson 1). Such bias - or other factors such as sample size - may influence the reliability of your conclusions, and therefore in terms of reporting you should report on the side of caution, and avoid the outright acceptance of your null hypothesis.

An alternative means for reporting the results of a two-sided statistical test is the use of confidence intervals. Recall from our second STATA lab when we calculated the mean of our variables via the “mean” command, we were already presented with our (95%) confidence intervals. If you conduct statistical testing on your mean/parameters with confidence intervals, a null hypothesis would be rejected if and only if it lies outside a corresponding confidence interval. In order to calculate a confidence interval, you need 3 pieces of information: 1) your sample mean (not the arbitrary value which our null hypothesis is based); 2) your sample standard error (i.e. your standard deviation divided by the square root of your sample size), and; 3) the critical t-value of your desired confidence level (either 90%, 95% or 99%). The formula to calculate a confidence interval is as follows:

$$CI = \mu \pm t_{critical} * SE$$

In the case of our 20-observation sample of British graduates, our sample mean is 87,776.74 and our standard error is 8555.37. Constructing a 95% confidence interval (from Figure 4.1, with 19 degrees of freedom, our critical value would be 2.093), we would construct the following interval:

$$\begin{aligned} &87,776.74 \pm (2.093)*(8555.379) \\ &87,776.74 \pm 17,906.41 \\ \text{Confidence Interval: } &(69,870.33, 105,683.10) \end{aligned}$$

Because our 70,000 value lies within this confidence interval, we would fail to reject the null hypothesis that our mean is equal to 70,000 with 95% confidence. The reason why we fail to reject is because the range of confidence interval assigns the possibility of a possible population mean of 70,000. In the STATA lab below, you will further test this hypothesis using a large and small sample from respondents of the US World Values Survey.

## STATA LAB (LESSON 4):

---

For this exercise, we will rely upon two datasets from the US World Values Survey (WVS): one with 49 observations, and one with 999 observations. Open up “WVS” Excel dataset. This particular survey was conducted between 2005 and 2006, where Americans were randomly sampled and asked about their values and beliefs. The datasets you see are a condensed version, with the following variables:

1. The gender of the respondent; 1 for male, 2 for female
2. The age of the respondent
3. The recorded “happiness” level of the respondent; each respondent was asked to rank their level of happiness on the following ordinal scale: 1 for “very happy”; 2 for “rather happy”; 3 for “Not very happy”; and 4 for “Not at all happy”
4. The recorded “health” level of the respondent; each respondent was asked to self-assess their state of health on an ordinal scale: 1 for “very good”, 2 for “good”, 3 for “fair” and 4 for “poor”
5. Whether the individual is an active member of a sport or recreational organization: 2 for “active member”, 1 for “inactive member” and 0 for “don’t belong”
6. Whether the individual is an active member of an art, music or educational organization: 2 for “active member”, 1 for “inactive member” and 0 for “don’t belong”
7. How often the respondent uses the computer, assessed on an ordinal scale: 1 for “never”, 2 for “occasionally”, 3 for “frequently”, and 4 for “don’t know what a computer is”

Upload both datasets from Excel into STATA (via the copy/paste method you used in Lesson 2). Before we move to significance testing, we are going to re-visit our lessons on cross-tabs/contingency tables and the influence of sample sizes on significance testing. I want you to construct a contingency table on the influence of participation in a sports club on reported health levels for both datasets (n=49 and n=999), presenting column frequency levels. Remember to distinguish your independent variable (in this case, participation in a sports club – which will be your column variable) and your dependent variable (in this case, reported health-status – which will be your row variable). Use the “tabulate depvar indepvar, column” learned in Lesson 2 on the n=49 dataset. You should be presented with the following contingency table:

. tabulate health sportsorgmember, column

Key	
frequency	column percentage
Health	SportsOrgMember
	0 1 2 Total
1	2 3 1 6 5.56 50.00 14.29 12.24
2	25 3 5 33 69.44 50.00 71.43 67.35
3	8 0 1 9 22.22 0.00 14.29 18.37
4	1 0 0 1 2.78 0.00 0.00 2.04
Total	36 6 7 49 100.00 100.00 100.00 100.00

Type | Format

byte %8.0g

byte %8.0g

byte %8.0g

byte %8.0g

byte %8.0g

byte %8.0g

byte %8.0g

byte %8.0g

Notice that for individuals that are inactive and active members, there appears to be slightly better reported health status, either 1 for “very good” health or 2 for “good” health, than for individuals who do not belong to sports organizations; of these individuals only 25% report either a health status of 3, “fair”, or 4, “poor”. We learned in Lesson 3 that in order to test whether there was contingency in categories between our dependent and independent variable (i.e. whether the reported health status varies by participation in a sports organization) or independence (i.e. whether the reported health status is relative constant across all types of participation in a sports organization), we relied upon a Pearson’s Chi-squared statistic. A Pearson Chi-squared statistic is another method of significance testing to the t-statistic method, which tests the null hypothesis of whether there is independence between cross-tab categories. Let’s employ this method again for our 49 observation dataset to determine whether there is contingency between our dependent and independent variable. Type in the command “tabulate health sportsorgmember, column chi2” that you learned from Lesson 3. You should see the following screen:

`. tabulate health sportsorgmember, column chi2`

Key	
	frequency column percentage
Health	SportsOrgMember
	0 1 2 Total
1	2 3 1 6 5.56 50.00 14.29 12.24
2	25 3 5 33 69.44 50.00 71.43 67.35
3	8 0 1 9 22.22 0.00 14.29 18.37
4	1 0 0 1 2.78 0.00 0.00 2.04
Total	36 6 7 49 100.00 100.00 100.00 100.00
Pearson chi2(6) = 10.4509 Pr = 0.107	

Format

%8.0g  
%8.0g  
%8.0g  
%8.0g  
%8.0g  
%8.0g  
%8.0g

While the Chi-squared statistic appears quite large, its associated critical values are much larger than associated critical values from t-statistics that you witnessed in Figure 4.1. Note that the corresponding p-value, just to the right of the reported Chi-squared statistic, is just above 0.100. This means that under the Pearson Chi-squared test, you cannot reject the null hypothesis with at least 90% confidence that there is independence (i.e. no association) between the two variables.

Let's now shift analysis to our larger dataset (n=999). Calculate the same cross-tab/contingency table, producing column frequencies and specify that you would like to test for independence between participation in a sports organization and reported health status via the Pearson Chi-squared statistic. Use the following command: "tabulate health sportsorgmember, column chi2". You should see the following output screen:

. tabulate health sportsorgmember, column chi2

Key	
frequency	column percentage
Health	SportsOrgMember
	0 1 2 Total
1	190 22.54 29 40.85 29 34.52 248 24.85
2	448 53.14 34 47.89 44 52.38 526 52.71
3	183 21.71 8 11.27 9 10.71 200 20.04
4	22 2.61 0 0.00 2 2.38 24 2.40
Total	843 100.00 71 100.00 84 100.00 998 100.00
Pearson chi2(6) = 22.0281 Pr = 0.001	

Notice, again, that before looking at the Pearson Chi-squared statistic, it appears that individuals who are either inactive or active in sports organizations appear to report lower levels of low health (i.e. values of 3 “fair” or 4 “poor”). Shifting focus to the Pearson Chi-Squared test statistic, what do you notice about its significance? From its corresponding p-value (0.001) we can reject the null hypothesis of independence between categories with over 99% confidence! Notice in particular, what a larger sample size did to our significance testing; as mentioned above, it produced a larger test statistic, and hence increased our certainty in rejecting the null hypothesis. Larger sample sizes, in other words, produce larger test-statistics which can be compared against smaller critical values, increasing our confidence precision.

Let’s now apply our analysis for means testing above to both datasets. Let’s say, for both of our datasets, we want to test whether individuals have a mean happiness of “not very happy” (i.e. a happiness value of 3). Therefore our null hypothesis and alternative hypothesis would be the following:

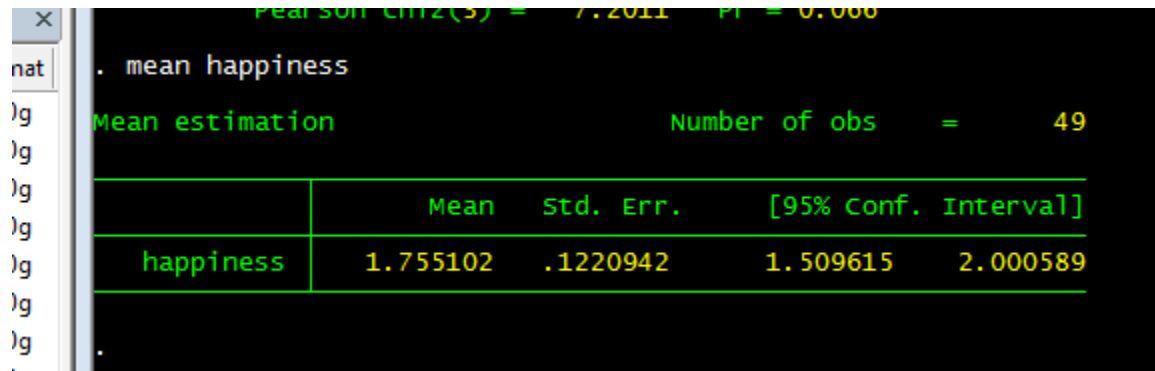
$$H_0: \bar{X} = 3$$

$$H_A: \bar{X} \neq 3$$

Before we discover the command for conducting tests of means in STATA, I want you to calculate our t-statistic manually within both datasets, and report whether we can reject or fail to reject the null hypothesis. Recall from above the three things we need for a t-test are our arbitrary hypothesis testing value (in our case 3), the mean of our sample, and the standard error of our sample. To calculate the

mean/standard error, we can use the “mean” command that we learned in Lesson 2. Calculate both for the large and small dataset. You should see the following windows:

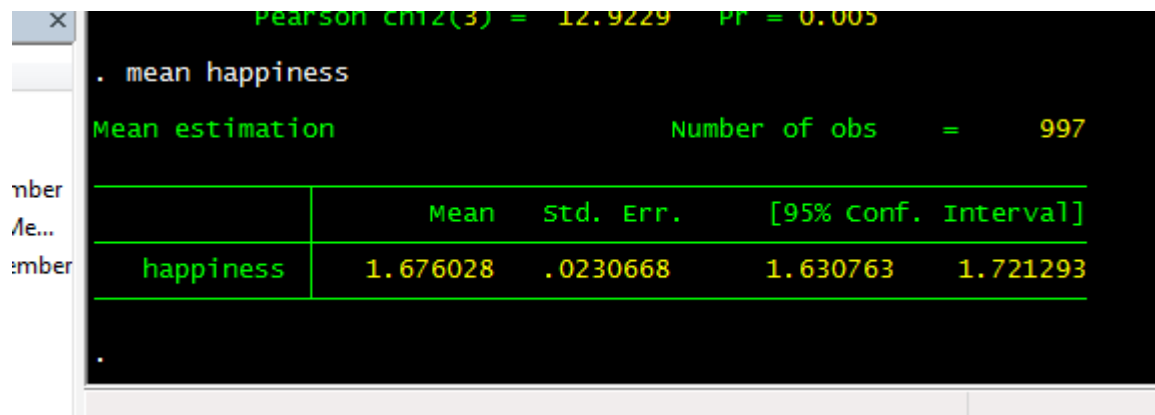
For the small dataset:



Stata output window showing the command `. mean happiness` and the results for the small dataset (49 observations). The output includes the mean estimation, standard error, and 95% confidence interval for the variable happiness.

Mean estimation		Number of obs = 49		
	Mean	Std. Err.	[95% Conf. Interval]	
happiness	1.755102	.1220942	1.509615	2.000589

For the large dataset:



Stata output window showing the command `. mean happiness` and the results for the large dataset (997 observations). The output includes the mean estimation, standard error, and 95% confidence interval for the variable happiness.

Mean estimation		Number of obs = 997		
	Mean	Std. Err.	[95% Conf. Interval]	
happiness	1.676028	.0230668	1.630763	1.721293

Notice, from Lesson 2, that the standard error of our larger dataset is smaller than that of our small dataset (review lesson 2 and the definition of standard errors if this is unclear to you). Calculating the t-statistic for the small and large dataset manually, you should obtain:

Small dataset:

$$T\text{-statistic: } (1.755102 - 3)/0.1220942 = -10.196$$

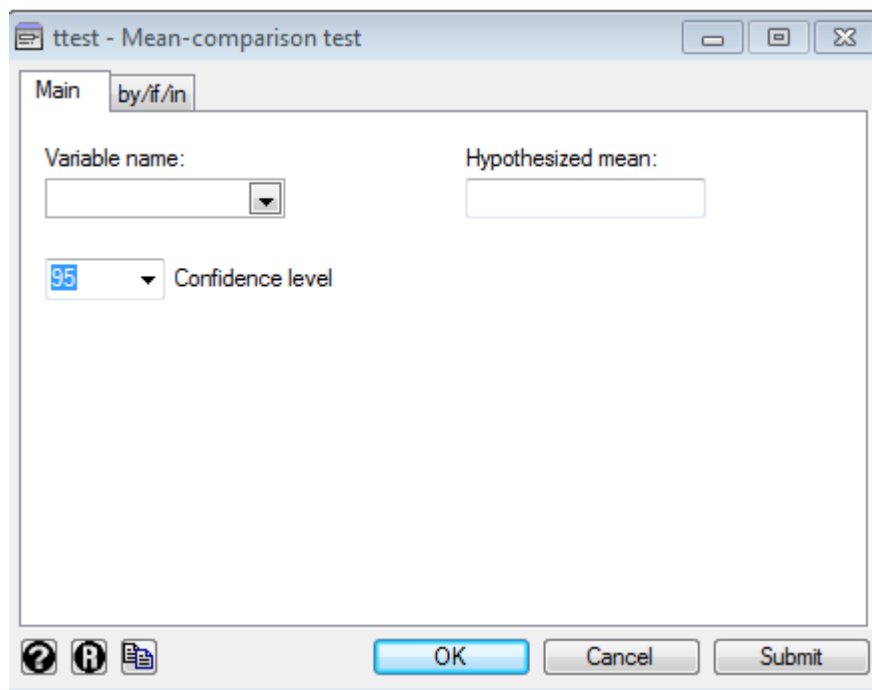
Large dataset:

$$T\text{-statistic: } (1.676028 - 3)/0.0230668 = -57.397$$

Comparing these t-statistics to their relative critical values in Figure 4.1 (the first against a critical value with 48 degrees of freedom [ $df = n - p = 49 - 1$ ], the second against 998 degrees of freedom [ $df = n - p = 999 - 1$ ]), both are so significantly high in absolute terms that we can reject the null hypothesis that average happiness is “not very happy” with over 99.9% confidence. Notice too, that the larger dataset has a

significantly larger t-statistic in absolute terms than the smaller one. Again, this is no coincidence; larger sample sizes have smaller standard errors (see Lesson 2) and hence should have larger t-statistics.

Let's now ask STATA to conduct a test of means, testing that our average level of happiness is equivalent to 3. Starting with the small dataset first, click on "Statistics", then "Summaries, Tables and Tests", and then "Classic Tests of Hypotheses" and then "One Sample Mean Comparison Test". You should see the following box:



In the "Variable name" tab, specify which variable whose mean you wish to test (in our case, you would select "happiness"). In the "Hypothesized mean" tab, specify what value you wish to test the mean against (in our case, this would be 3). Notice that you can manipulate the level of confidence in the "Confidence interval" tab, which we will cover later; we will keep it at 95%, but you can lower this to 90% or raise it to 99/99.9% for the presentation of confidence intervals. Click "ok" and you should see the following output:

```
. ttest happiness == 3, level(95)

One-sample t test
```

variable	obs	Mean	Std. Err.	Std. Dev.	[95% Conf. Interval]	
happin~s	49	1.755102	.1220942	.8546591	1.509615	2.000589

```

      mean = mean(happiness)
Ho: mean = 3                      t = -10.1962
                                degrees of freedom = 48

      Ha: mean < 3                Ha: mean != 3                Ha: mean > 3
Pr(T < t) = 0.0000                Pr(|T| > |t|) = 0.0000        Pr(T > t) = 1.0000

```

Notice that in the highlighted yellow box, the same t-statistic as you calculated above emerges.

### CONGRATULATIONS! You have just conducted a means test in STATA!

You may notice, however, that rather than testing our null hypothesis against one alternative hypothesis (mean does not equal 3 – in STATA the “!” symbol means “does not”), the program has presented us with three alternative hypotheses. These hypotheses are: a two-tailed  $\neq$  alternative hypothesis, which is presented in the middle, and two one-tailed alternative hypotheses,  $>$  and  $<$  which are presented on either side of the two-tailed hypothesis. The p-values that STATA presents under these hypotheses correspond to the original null hypothesis! Hence, regarding the first alternative hypothesis, the means test is framed as follows:

$$\begin{aligned} H_0: \bar{X} &= 3 \\ H_A: \bar{X} &< 3 \end{aligned}$$

And the corresponding p-value against the null hypothesis is  $\Pr(T < t) = 0.000$ . Hence, for this t-test, we can reject the null hypothesis that the mean equals three, against the alternative hypothesis that the mean is less than three. Now, compare the other one-sided alternative hypothesis at the right of the output:

$$\begin{aligned} H_0: \bar{X} &= 3 \\ H_A: \bar{X} &> 3 \end{aligned}$$

In this case, STATA has given us a corresponding p-value of 1.000, which is significantly higher than our baseline value of 0.1, against the null hypothesis. This is not to say that the null hypothesis equals three is validated! This is to say that we fail to reject the null hypothesis against the alternative hypothesis which asserts that the sample mean is greater than 3. Particular care must be assigned to the interpretation of your null hypothesis against your alternative hypothesis. Generally, for two-tailed alternative hypotheses, this is much easier to do; the null is rejected if STATA produces a probability which is 0.100 or less, and we fail to reject the null if STATA produces a probability that is above 0.100).

#### STATA COMMAND 4.1:

*Code:* “`ttest var1 == #, level(95)`”, where var1 is the variable of interest, # is the value which you wish to test the mean of var1 against, and level(95) is the significance level you wish to test.

*Output produced:* Calculates a mean test for the specified variable.

Let’s now run the “ttest” command for the larger dataset. This time, type the following command into the command box: “`ttest happiness == 3, level(95)`”. You should see the following output:

`. ttest happiness ==3, level(95)`

One-sample t test

Variable	Obs	Mean	Std. Err.	Std. Dev.	[95% Conf. Interval]	
happin~s	997	1.676028	.0230668	.7283399	1.630763	1.721293

mean = mean(happiness)  
Ho: mean = 3  
Ha: mean < 3  
Pr(T < t) = 0.0000

t = -57.3974  
degrees of freedom = 996  
Ha: mean != 3  
Pr(|T| > |t|) = 0.0000

Ha: mean > 3  
Pr(T > t) = 1.0000

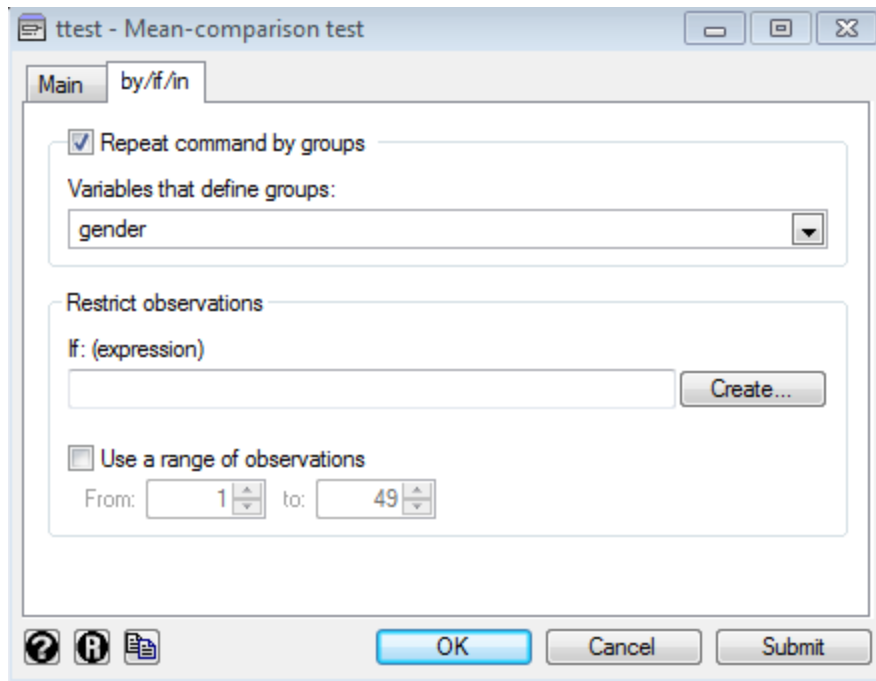
Notice again, highlighted in the yellow box, that you are presented with the same t-statistic that we calculated above! Also, notice, similarly to the small dataset, STATA presents you with three possible alternative hypotheses for which to test the null against:  $H_A: < 3$ ,  $H_A: \neq 3$ , and  $H_A: > 3$ . As was the case above, the corresponding p-value for the first two alternative hypotheses ( $H_A: < 3$ ,  $H_A: \neq 3$ ) are significantly low (i.e. less than 0.100) so that we can reject our null hypothesis against these alternatives. However, our corresponding p-value for the third alternative hypothesis ( $H_A: > 3$ ) is so significantly high, that we must fail to reject our null hypothesis against the alternative.

One very nice feature about STATA is that it also calculates means tests for subgroups within your sample; for example, say we want to conduct a means test that reported health is on average “very good” (value of 1) by gender. Specifically, we want to test the following null/hypothesis for men and women separately within our sample:

$$H_0: \bar{X} = 1$$

$$H_A: \bar{X} \neq 1$$

Starting with the small dataset, go back to the “Statistics” tab and click “Summaries, Tables and Tests”, and then “Classic Tests of Hypotheses” and then “One Sample Mean Comparison Test”. You should see the box you did above. Enter “health” in for the variable, 1 for the mean value, but then click on the “by/if/in” tab. You should see the following box:



Click on the “Repeat commands by groups” and specify in the “Variables that define groups” box that you want to repeat the test of means by gender. Click “Ok”. You should see the following output:

```
. by gender, sort : ttest health == 1

-> gender = 1
One-sample t test

```

Variable	obs	Mean	Std. Err.	Std. Dev.	[95% Conf. Interval]
health	33	1.939394	.0967208	.5556187	1.74238 2.136408

```

      mean = mean(health)                                t =    9.7124
Ho: mean = 1                                           degrees of freedom =    32

      Ha: mean < 1          Ha: mean != 1          Ha: mean > 1
Pr(T < t) = 1.0000      Pr(|T| > |t|) = 0.0000      Pr(T > t) = 0.0000

-> gender = 2
One-sample t test

```

Variable	obs	Mean	Std. Err.	Std. Dev.	[95% Conf. Interval]
health	16	2.4375	.1572882	.6291529	2.102248 2.772752

```

      mean = mean(health)                                t =    9.1393
Ho: mean = 1                                           degrees of freedom =    15

      Ha: mean < 1          Ha: mean != 1          Ha: mean > 1
Pr(T < t) = 1.0000      Pr(|T| > |t|) = 0.0000      Pr(T > t) = 0.0000

```

**CONGRATULATIONS! You have just conducted a means test for sub-groups in STATA!**

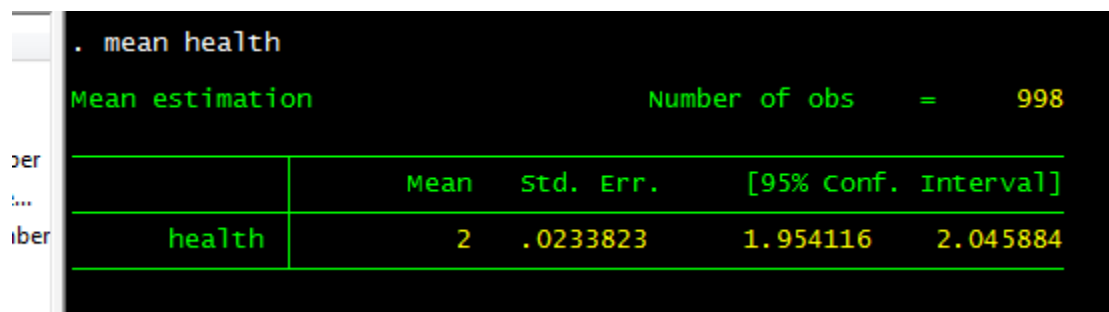
Notice that for both men (gender = 1) and women (gender =2) we obtain a high t-statistic so we can reject our null hypothesis against our two-tailed alternative hypothesis with high confidence (i.e. above 99%); the corresponding p-value associated with the null is significantly less than 0.100. Also notice, that if we specify that our one-tailed alternative hypothesis is that the mean is less than one ( $H_A: < 1$ ), we obtain a resoundingly high p-value (1.000) for both men and women, which means we are unable to reject our null hypothesis that our mean is equal to one against an alternative hypothesis that it is less than 1.

#### STATA COMMAND 4.2:

*Code:* “**by var2, sort : ttest var1 == #**” where var1 is the variable of interest, # is the value which you wish to test the mean of var1 against, and var2 is the sub-group category you wish to conduct means tests for.

*Output produced:* Calculates a mean test for the specified variable, by specified sub-group.

Throughout this lab, you may have noticed that the confidence intervals of all our means tests are automatically produced within our t-test tables. Recall from above, that the manual computation of a confidence interval relies upon 3 pieces of information: 1) sample mean, 2) sample standard error, and, 3) the critical t-value of your desired confidence level (either 90%, 95% or 99%). Focusing on average reported health status, let’s manually calculate a 90% confidence interval for our large dataset. Using the “mean” command from Lesson 2, calculate the mean and the standard error of reported health. You should see the following output:



Mean estimation		Number of obs		= 998	
	Mean	Std. Err.	[95% Conf. Interval]		
health	2	.0233823	1.954116	2.045884	

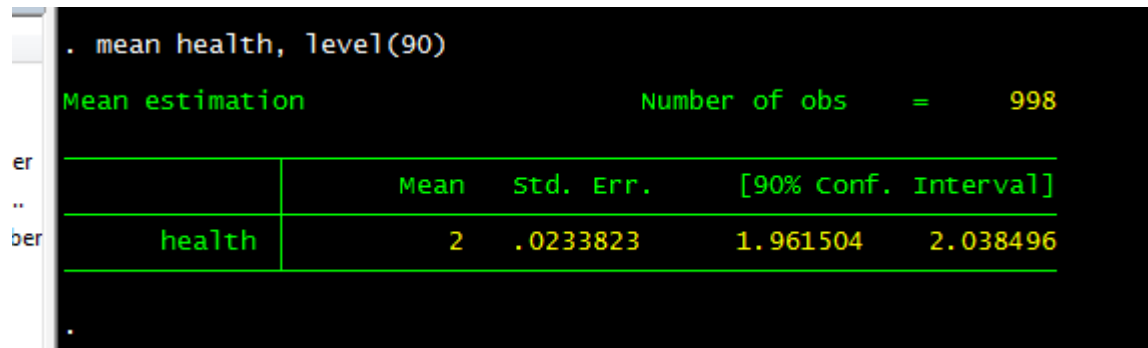
Notice that STATA automatically produces the 95% confidence interval; I have specified however, a 90% CI! This means you will have to compute it manually! Also notice that STATA only records 998 observations of 999 – this is because one of our observations is lacking a health response and thus the entire observation is dropped from the analysis. One thing you will learn quickly in STATA is that whenever you run a statistical test, your observations must have values for all variables which are included in the test; if it does not, the observation will automatically be dropped.

Taking our critical value first, because our degrees of freedom ( $df = n - p = 998 - 1 = 997$ ) is close enough to 1000, we will select 1.646 for our critical t-stat value from Figure 4.1 above. This means our confidence interval can be assembled as follows:

$$\text{Confidence Interval} = 2 \pm 1.646 * (0.0233823)$$

Confidence Interval =  $2 \pm 0.03849$   
Confidence Interval: (1.96151, 2.03849)

Now, let's have STATA compute the confidence interval of our mean. Type in the command for calculating the mean, but at the end, specify that you want a 90% confidence level: "mean health, level(90)". You should see the following output:



```
. mean health, level(90)
```

Mean estimation		Number of obs		=	998
	Mean	Std. Err.	[90% Conf. Interval]		
health	2	.0233823	1.961504	2.038496	

**CONGRATULATIONS! You have just calculated a confidence interval of a mean in STATA!**

Notice that our 90% confidence interval is identical to the one we calculated above, with the slight rounding differences. Confidence intervals provide an alternative form of significance testing for means, because they provide us with a range which our true population mean should lie. In other words, if the projected mean value lies outside this range, we can reject with 90%/95%/99% confidence that our mean is equal to it.

#### STATA COMMAND 4.3:

*Code:* "**mean var1, level(X)**" where var1 is the variable of interest, and X is the desired confidence interval which you want to calculate (i.e. 90, 95 or 99, **never select a value less than 90**).

*Output produced:* Calculates the confidence interval of specified mean.

Like t-statistics, confidence intervals will be smaller (i.e. more accurate) for larger datasets, because larger datasets have smaller standard errors. As a check, calculate the 90% confidence interval for reported health in our 49 observation sample. You should see the following output:

at	. mean health, level(90)			
	Mean estimation	Number of obs	=	49
		Mean	Std. Err.	[90% Conf. Interval]
	health	2.102041	.088664	1.953331 2.25075
	.			

Notice that though our mean is slightly higher for our smaller sample, the 90% confidence level is larger than that for our 999 observational sample. Again, this is due to the fact that our larger sample has a smaller standard error.

---

### Practice Problems:

Practice Problem 1: Open the two data WVS files that you used in the STATA lab. Create a cross-tab which presents the frequency reported happiness levels by participation in an arts/music/educational organization (make sure to differentiate between the dependent/row and independent/column variable). Conduct a Pearson test of independence between happiness and arts/music/educational program participation for both datasets. What conclusions can you offer for the small versus large dataset?

Practice Problem 2: Create a cross-tab which presents the frequency of reported health levels by gender (make sure to differentiate between the dependent/row and independent/column variable). Conduct a hypothesis test of independence between health and gender for both datasets. Can you confidently reject the null hypothesis of independence for the small dataset? For the large dataset? Is the Chi-squared statistic larger for one dataset over the other?

Practice Problem 3: Establish a null and alternative hypothesis to test whether the average happiness level for both the small and the large dataset is equal to 2 (i.e. “rather happy”). Do this via two ways; first calculate the t-statistic for both datasets manually, by obtaining the mean and standard errors of both populations. Second, run a t-test in STATA for both groups, specifying a comparison mean value of 2. Do your t-statistics match-up between the manual test and the result in STATA. What can you conclude for the small and large sample, regarding the null hypothesis?

Practice Problem 4: Repeat Lab Practice Problem 3 on both datasets, but specify that you wish to run a test of (happiness) means (equal to 2, or “rather happy”) by gender. Compare your test of means results for women only between the large and small datasets. Can you reject the null hypothesis of women having an average happiness level equal to “rather happy” against an alternative hypothesis of women having an average happiness level that is higher than “rather happy” (i.e. closer to 1, “very happy”, or  $H_A: < 2$ ) for both samples?

Practice Problem 5: Calculate the 95% and 99% confidence interval for reported happiness levels within the large and small datasets. Construct them manually first, as was done above, and then compute them via STATA. What do you notice about the confidence level interval for the smaller versus large sample?

Practice Problem 6: Calculate the 99% confidence interval for the mean of reported use of computers by gender for both samples (note, you can do this via the “ttest” command or the “mean” command, but you must specify that you are calculating the mean for the separate subgroups). How does the CI of women compared to men in the small sample? How does the CI of women compare between the large and small sample?

## Lesson 5: Difference-in-Means Testing (for Independent Groups)

---

**Learning Objective 1: To understand the research design of testing whether the variable mean of two independent groups are equal**

**Learning Objective 2: To understand how to calculate a test statistic, and, using its corresponding p-value, to determine whether to reject/fail-to-reject a null hypothesis that the variable mean of two groups are equivalent.**

**Learning Objective 3: Manually calculating a t-statistic and p-value for a difference-in-means test from descriptive statistics calculated in STATA.**

**Learning Objective 4: Conducting a difference-in-means test in STATA**

From the last lesson, you learned how to test whether the value of a variable's mean was equivalent to some arbitrary value. While this can be a helpful exercise, you will discover when you conduct your research that it is much more useful to compare the variable mean of two groups, rather than to test their specific values. The difference-in-means test, the first important test of statistical inference you will be exposed to in this manual, enables you to determine whether the mean of the same variable for two independent groups (groups that are mutually exclusive; say, men and women, old and young, high and low income, etc.), are statistically different. For many of you, difference-in-means testing will be a central test for comparing outcomes for groups in your MPP essay.

Conducting a difference-in-means test relies not only on the means of two groups. It also requires the testing of whether the differences between these two values are significant. Hence, difference-in-means test require all four steps for hypothesis testing that we discussed in the previous lesson; 1.) the establishment of a null and alternative hypothesis; 2.) the calculation of a test statistic; 3.) the comparison of this statistic to some critical value, and its corresponding p-value, and; 4.) a conclusion based upon the value of the t-statistic relative to the critical statistic. To give you an empirical example of how comparisons of means alone can be misleading, consider the evidence presented in below. In Table 5.1, hypothetical examination results from two independent groups of students – 17 individual that took a course from a campus module, and 19 individuals that took the course via a distance e-learning module – are presented. Considering their means alone, it appears that the campus-based students have performed better than their distance-learning counter-parts; campus-based students scored an average 68.71% on their exams, while distance-learning students scored an average of 64.53%. However, you being the bright statistics students you are, know that in order to test whether means are comparable, a significance test, along the lines of that learned in lesson 4, must be established. We will go through each step with the data below.

Table 5.1: Examination Results of Campus-Based and Distance-Learning Students

<i>Student Number</i>	<i>Mean Test Score</i>	<i>Student Number</i>	<i>Mean Test Score</i>
Campus Based 1	65	Distance Based 1	53
Campus Based 2	87	Distance Based 2	67
Campus Based 3	54	Distance Based 3	63
Campus Based 4	82	Distance Based 4	57
Campus Based 5	81	Distance Based 5	69
Campus Based 6	70	Distance Based 6	67
Campus Based 7	76	Distance Based 7	50
Campus Based 8	65	Distance Based 8	74
Campus Based 9	76	Distance Based 9	79
Campus Based 10	68	Distance Based 10	66
Campus Based 11	63	Distance Based 11	68
Campus Based 12	69	Distance Based 12	53
Campus Based 13	58	Distance Based 13	70
Campus Based 14	67	Distance Based 14	82
Campus Based 15	68	Distance Based 15	61
Campus Based 16	51	Distance Based 16	73
Campus Based 17	68	Distance Based 17	48
<i>MEAN</i>	<i>68.7059</i>	Distance Based 18	71
<i>VARIANCE</i>	<i>92.4706</i>	Distance Based 19	55
		<i>MEAN</i>	<i>64.5263</i>
		<i>VARIANCE</i>	<i>94.8187</i>

Step 1: Establishment of the null (and alternative) hypothesis

Whenever we conduct a difference-in-means test, the null hypothesis should always be that the difference between the two values is zero. Establishing a two-tailed alternative hypothesis (that one mean is larger/smaller than the other), this produces the following hypotheses:

$$H_0: \bar{X}_1 - \bar{X}_2 = 0$$

$$H_A: \bar{X}_1 - \bar{X}_2 \neq 0$$

where  $\bar{X}_1$  is the mean of the first group and  $\bar{X}_2$  is the mean of the second group. In our example above, we interpret our null hypothesis as the scenario where average test scores between campus-based and distance-based students are relatively similar. Our alternative hypothesis is that these two scores are not similar.

### Step 2: Calculation of a test statistic

After establishing the null hypothesis above, you need to calculate the t-statistic. From Lesson 4, you learned that the equation for a t-statistic was the following:

$$T-stat = \frac{\mu - \bar{X}}{se}$$

T-statistics for difference-in-means tests are slightly different because you are examining two different samples, which may vary by size (and hence so will the denominators of the t-statistic, their standard errors). The formula for a difference-in-means test statistic can be written as follows:

$$T-stat = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\left(\frac{\sigma_1^2}{n_1}\right) + \left(\frac{\sigma_2^2}{n_2}\right)}}$$

$\bar{X}_1 - \bar{X}_2$  is our difference in means for group 1 (call this our campus-based group) and group 2 (call this our distance based group).  $\sigma_1^2$  is the variance (or the squared value of the standard deviation) of group 1 (campus-based group – which from Table 5.1 we see is 92.4706) and  $n_1$  is the number of observations/individuals within the group (in our case, 17). Likewise,  $\sigma_2^2$  is the variance (or the squared value of the standard deviation) of group 2 (distance-learning group – which from Table 5.1 we see is 94.8187) and  $n_2$  is the number of observations/individuals within the group (in our case, 19). Given these values, if we wanted to calculate the difference-in-means test statistic from our sample above, we would obtain the following:

$$T-stat = \frac{68.7059 - 64.5263}{\sqrt{\left(\frac{92.4706}{17}\right) + \left(\frac{94.8187}{19}\right)}}$$

$$T-stat = \frac{4.1796}{\sqrt{(5.4394) + (4.9905)}}$$

$$T-stat = \frac{4.1796}{3.2295}$$

$$T-stat = 1.2942$$

Given our descriptive statistics, the t-statistic from our campus-based/distance-learning average test result comparison is 1.2942. As we did with the previous lesson, we compare this against a critical t-value in our next step.

### Step 3: Comparison of the t-statistic against a critical value

In the previous lesson, if we wanted to find a critical t-value, we needed two pieces of information: the confidence level for which we wish to report (which should never be below 90%) and degrees of freedom, our number of observations minus our parameters. With difference of means tests, the degrees of freedom is the sum of the degrees of freedom from both groups. In the case above, if we are examining only the mean for both groups, we can calculate degrees of freedom as:

$$df_1 + df_2 = (17-1) + (19-1) = 34$$

Applying this to our critical value, go back to the t-statistic table in Figure 4.1 (in lesson 4). Under 34 degrees of freedom (between 30 and 40), our 90% confidence critical value is just under 1.697, our 95% confidence critical value is just under 2.042, and our 99% confidence critical value is just under 2.750. We will use these three critical values to compare our t-statistic above in our conclusions.

### Step 4: Conclusions

From our two samples of campus-based and distance learners above, we have calculated our t-statistic ( $t=1.2942$ ), and established three reference values for which to compare it against. Even in our most generous case (90% confidence), we find that our t-statistic is smaller than our critical value (1.697). Given such results, we fail to reject the null hypothesis that there are significant differences in average test scores between the two groups. We cannot conclude with over 90% confidence (and hence having less than a 0.100 chance of falsely rejecting a null hypothesis that is true) that the campus based group performed better on the exam than the distance-learning group.

Notice that the conclusions reached from conducting the four-step significance test for difference-in-means (that both means were not significantly different) was contrary to a mere means comparison (that both means were different). Whenever you wish to compare means between two groups, you must do so via the above four-step process; you have to factor in the variance (spread) of your data for both independent groups, as different means may merely be reflective of large spread between the data within groups (i.e. one may have an extreme outlier, which biases its mean), rather than a “true” mean difference.

## STATA LAB (LESSON 5):

---

For this exercise, we will rely upon datasets from the US World Values Survey (WVS), except we will only utilize the small dataset with 49 observations (you will be asked to run the same commands in the larger dataset for the practice problems). Recall that the datasets consists of the following variables:

1. The gender of the respondent; 1 for male, 2 for female
2. The age of the respondent
3. The recorded “happiness” level of the respondent; each respondent was asked to rank their level of happiness on the following ordinal scale: 1 for “very happy”; 2 for “rather happy”; 3 for “Not very happy”; and 4 for “Not at all happy”
4. The recorded “health” level of the respondent; each respondent was asked to self-assess their state of health on an ordinal scale: 1 for “very good”, 2 for “good”, 3 for “fair” and 4 for “poor”
5. Whether the individual is an active member of a sport or recreational organization: 2 for “active member”, 1 for “inactive member” and 0 for “don’t belong”
6. Whether the individual is an active member of an art, music or educational organization: 2 for “active member”, 1 for “inactive member” and 0 for “don’t belong”
7. How often the respondent uses the computer, assessed on an ordinal scale: 1 for “never”, 2 for “occasionally”, 3 for “frequently”, and 4 for “don’t know what a computer is”

Upload the small dataset from Excel into STATA (via the copy/paste method you used in Lesson 2). We are going to run a number of difference-in-means tests by group, but before we learn the command in STATA, you will learn how to manually compute difference-in-means test statistics. Let’s first conduct a difference in means test on reported health levels by gender. In other words, we want to test whether men report level of health that is roughly equivalent to what women report. Follow the four steps above in order to test this hypothesis:

### Step 1: Establishment of the null (and alternative) hypothesis

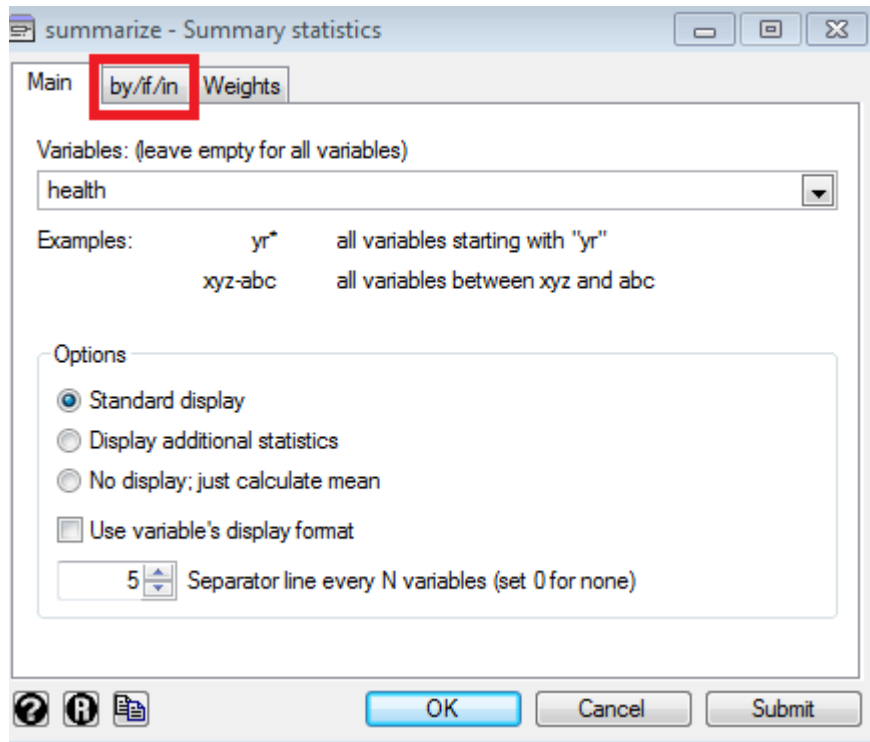
We want to test whether the mean health value of men is equivalent to that of women. Hence, our null and two-tailed alternative hypothesis can be written as follows:

$$\begin{aligned}H_0: \bar{X}_M - \bar{X}_F &= 0 \\H_A: \bar{X}_M - \bar{X}_F &\neq 0\end{aligned}$$

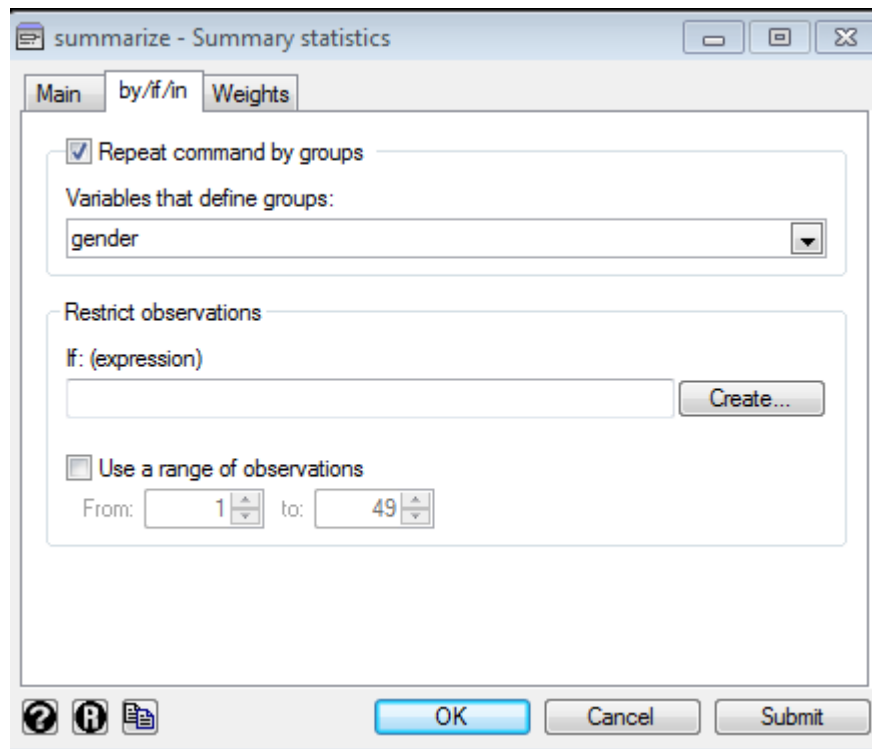
### Step 2: Calculate the t-statistic

From our difference-in-means t-statistic above, we need three pieces of information from both of our samples; the sample mean, the sample variance (or standard deviation), and the number of observations within each sample. Recall from Lesson 2, that the “Summarize” command provides us with the standard deviation, mean, number of observations for a specified variable. One thing that was not mentioned is that you can also calculate these variable descriptive statistics by groups. Click on “Statistics” in the top

tool bar, then “Summaries, Tables and Tests”, then “Summary and Descriptive Statistics”, and then “Summary Statistics”. You should see the following box:



This box may look familiar from Lesson 2, when we used it to calculate sample statistics. Enter “health” into the “Variables” box, but before you click “Ok”, click on the “by/if/in” tab highlighted in red. You should see the following:



Click on the “Repeat command by groups” box, and specify that you want to calculate summary statistics for the health variable by “gender”. Click “OK”. You should see the following output:

```
. by gender, sort : summarize health
```

Variable	Obs	Mean	Std. Dev.	Min	Max
<b>-&gt; gender = 1</b>					
health	33	1.939394	.5556187	1	3
<b>-&gt; gender = 2</b>					
Variable	Obs	Mean	Std. Dev.	Min	Max
health	16	2.4375	.6291529	2	4

**CONGRATULATIONS! You’ve just computed sample statistics by groups in STATA.**

This command is very helpful for listing summary statistics, as you do not have to sort through your data in order to divide your sample by groups. Also, this command provides you with all the data you need for the calculation of the difference-in-means test statistic.

**STATA COMMAND 5.1:**

*Code:* “**by groupvar, sort : summarize var1**”, where var1 is the variable of interest and groupvar is the grouping category which you wish generate your summary statistics by.

*Output produced:* Calculates the summary statistics for the specified variable by the specified group variable.

Taking men (gender variable 1) as  $\bar{X}_1$  and women (gender variable 2) as  $\bar{X}_2$  we can compose our t-statistic as follows

$$T - stat = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\left(\frac{\sigma_1^2}{n_1}\right) + \left(\frac{\sigma_2^2}{n_2}\right)}}$$

$$T - stat = \frac{1.9394 - 2.4375}{\sqrt{\left(\frac{0.5556^2}{33}\right) + \left(\frac{0.6292^2}{16}\right)}}$$

$$T - stat = \frac{-0.4981}{\sqrt{(0.0094) + (0.0247)}}$$

$$T - stat = \frac{-0.4981}{0.1847}$$

$$T - stat = -2.6968$$

*Step 3: Comparison of the t-statistic against a critical value*

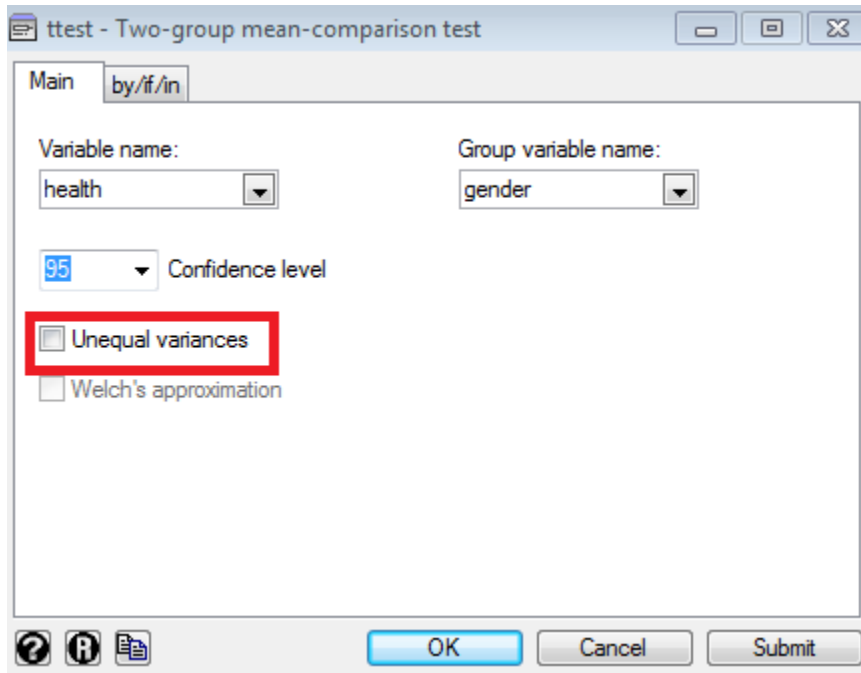
Now that we have our t-statistic (-2.6968), we can compare its absolute value it to a critical t-value. In order to do so, we need our degrees of freedom (which in this case is  $df = (33 - 1) + (16 - 1) = 47$ ), and we need to select a confidence interval for which to base our significance test upon. From Figure 4.1, 47 degrees of freedom yields a 90% significant critical t-value of just under 1.684, a 95% significant critical t-value of just under 2.021, and a 99% significant critical value of 2.704. You will notice that (the absolute value of) our test statistic exceeds the 90% and 95% critical value, but not the 99% critical value. This will impact our conclusions.

*Step 4: Make a conclusion about the null hypothesis*

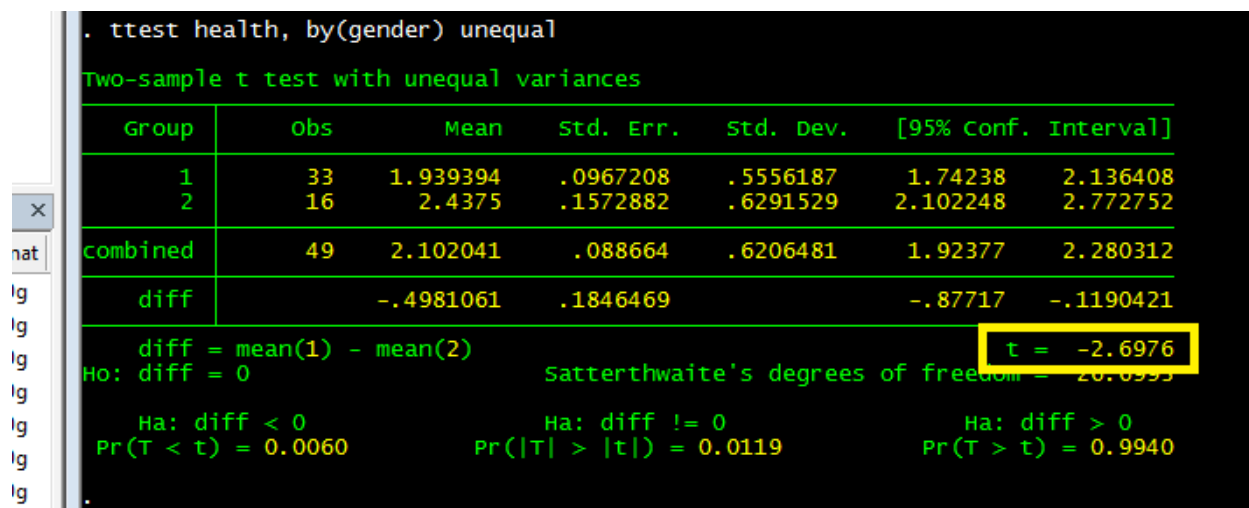
Given our t-statistic and critical values, what can you claim about your original hypothesis? If we select the more generous 90% and 95% critical values, we can reject our null hypothesis (with 90%/95% confidence) that the average reported health level is equal between men and women. This indicates that there is evidence that men reported a lower value (and hence had higher reported health) than women. We cannot make this assertion with 99% confidence; however, a 95% confidence level yields a low enough likelihood (i.e. p-value of 0.050 or less) of falsely rejecting a true null hypothesis. Therefore, in

your reporting, it is sufficient to reject the null hypothesis, but you must specify your confidence-level/p-value!

Now that we've manually calculated a difference-in-means test statistic, you will learn below how to ask STATA to generate this statistic for you. Click on the "Statistics" tab, followed by "Summaries, Tables and Tests", then "Classic Tests of Hypotheses", and click "Two-group mean-comparison test". You will see the following box:



Because we want to determine whether reported health significantly differs, on average, for women and men, enter "health" into the "Variable name" box, and "gender" into the "Group variable name" box. Throughout this lesson, we have operated under the assumption that both groups we examine do not have equal variances. Hence, click the "unequal variances" box highlighted in red above. Click "ok" and you should see the following output:



```
. ttest health, by(gender) unequal
```

Two-sample t test with unequal variances

Group	Obs	Mean	Std. Err.	Std. Dev.	[95% Conf. Interval]	
1	33	1.939394	.0967208	.5556187	1.74238	2.136408
2	16	2.4375	.1572882	.6291529	2.102248	2.772752
combined	49	2.102041	.088664	.6206481	1.92377	2.280312
diff		-.4981061	.1846469		-.87717	-.1190421

diff = mean(1) - mean(2)  
 Ho: diff = 0      Satterthwaite's degrees of freedom = 20.6995  
 Ha: diff < 0      Ha: diff != 0      Ha: diff > 0  
 Pr(T < t) = 0.0060      Pr(|T| > |t|) = 0.0119      Pr(T > t) = 0.9940

t = -2.6976

**CONGRATULATIONS! You've just computed a difference-in-means test in STATA!**

Notice that the t-statistic (-2.6976), highlighted in yellow, is roughly equivalent to that we computed above (it will not be exact because we rounded in our computations, whereas STATA does not).

## STATA COMMAND 5.2:

*Code:* “**ttest var1, by(groupvar) unequal**”, where var1 is the variable of interest and groupvar is the grouping category which you wish to calculate a difference-in-means for.

*Output produced:* Calculates a difference-in-means test-statistic against three alternative hypotheses.

Also notice above, like with significance testing in Lesson 4, STATA offers you three alternative hypotheses: 1) a one-tailed hypothesis where the difference in means is less than zero (hence men would report lower values of health – i.e. be healthier - than women); 2) a two-tailed hypothesis where the difference in means is not equal to zero (hence men would merely report a different value of health than women), and 3) a one-tailed hypothesis where the difference-in-means is greater than zero (hence men would report higher values of health – i.e. be less healthy- than women). Like basic significance testing, STATA presents you with the p-value against the null hypothesis. In the case the first one-tailed hypothesis, STATA presents strong evidence in a low p-value to reject the null hypothesis against an alternative hypothesis where men report a lower value of health (i.e. are more healthy). For the second one-tailed hypothesis, however, STATA presents weak evidence in a high p-value to reject the null hypothesis against an alternative hypothesis where men report a higher value of health (i.e. are less healthy).

---

### Practice Problems:

Practice Problem 1: Open the small WVS data file that you used in the STATA lab. Manually calculate a difference in means test for men and women's reported level of happiness (make sure to identify the null and alternative hypotheses, and all descriptive statistics required to calculate the t-statistic). Compare the statistic you generate against the statistic STATA generates. Can you reject the null hypothesis that women and men's reported happiness levels are equivalent?

Practice Problem 2: Manually calculate a difference in means test for men and women's reported participation in an arts, music or entertainment group (make sure to identify the null and alternative hypotheses, and all descriptive statistics required to calculate the t-statistic). Compare the statistic you generate against the statistic STATA generates. Can you reject the null hypothesis that women and men's participation levels are equivalent?

Practice Problem 3: Now open the large WVS data file used in Lesson 4, and recalculate your difference-in-means test statistic for the lab exercise, as well as lab questions 1 and 2 above. Use the "ttest var1, by(groupvar) unequal" command to verify whether you have calculated the correct statistic. What do you notice on your conclusions compared to the smaller dataset? Do they differ for average reported health, happiness, or arts/music/entertainment group participation?

Practice Problem 4: Calculate a binary variable in both the large and small dataset for whether an individual participates in a sports organization (with a coding of 1 for either being active or inactive, and a coding of 0 for not being involved in a sports organization) via the following commands:

```
"generate sports = 1 if sportsorgmember>0"  
"replace sports = 0 if missing(sports)"
```

Conduct a difference-in-means test for whether individuals in sports organizations are similarly healthy and similarly happy in both datasets (do not do so manually, just do so in STATA). Can you make similar conclusions regarding your null hypotheses in the small versus large datasets for both tests?

## Lesson 6: Univariate (OLS) Regression Analysis

---

**Learning Objective 1: Creating a two-way scatter plot in STATA between a dependent and independent variable**

**Learning Objective 2: Calculating a pair-wise correlation coefficient**

**Learning Objective 3: Conducting a univariate and multivariate regression in STATA and interpreting its beta coefficients, F-statistic, and reported R-squared**

Regression analysis centers around the construction of a model, where we attempt to explain variation in a dependent variable through the variation in one (or multiple) independent variables. In other words, we are attempting to quantify how a marginal increase in X (our independent variable) influences Y, our dependent variable.

Ordinary least squares regression models involve the estimation of a line that best fits the variation of your data - the best fit line. The simplest form of this best fit line for a univariate analysis (i.e. with only one independent variable)<sup>5</sup> is:

$$Y = \hat{\beta}_0 + \hat{\beta}_1 X + \varepsilon$$

Where Y is the dependent variable,  $\hat{\beta}_0$  the (predicted) y-intercept of the best-fit line, X is our independent variable,  $\hat{\beta}_1$  the (predicted) slope of the best fit regression line, and  $\varepsilon$  is our error term which captures everything that cannot be explained by the included independent variable(s). Ultimately, what we want to estimate in regression analysis is  $\hat{\beta}_1$ ; this tells us the impact of a marginal change of X on Y. There are two features specifically that we want to gauge with  $\hat{\beta}_1$ ; 1) whether it is positive/negative, hence explaining whether a change in X leads to an increase/decrease in Y, and; 2) whether it is significant (i.e. is greater/less than zero, which requires the production of a large t-statistic, with a corresponding p-value of 0.100 or less). The second feature is especially important, because if  $\hat{\beta}_1$  is not significant, we cannot claim that X has an impact on Y.

In this lesson you will complete the three learning objectives above on a dataset which will enable you to visually conceptualize a best-fit (univariate) regression line and empirically predict its y-intercept and slope. For the lesson you are given a dataset produced by Fisman and Miguel (2007)<sup>6</sup>, who examine the social norms regarding corruption by analyzing the parking behavior of UN diplomats in New York. Prior to 2003, diplomatic immunity protected UN diplomats from parking enforcements. Diplomats, therefore, were unconstrained by legal requirements on parking actions. The authors question, however, whether they were constrained by cultural constraints, specifically corruption; the hypothesis tested was

---

<sup>5</sup> We will discuss multivariate models which involve more than one independent variable in the next lesson.

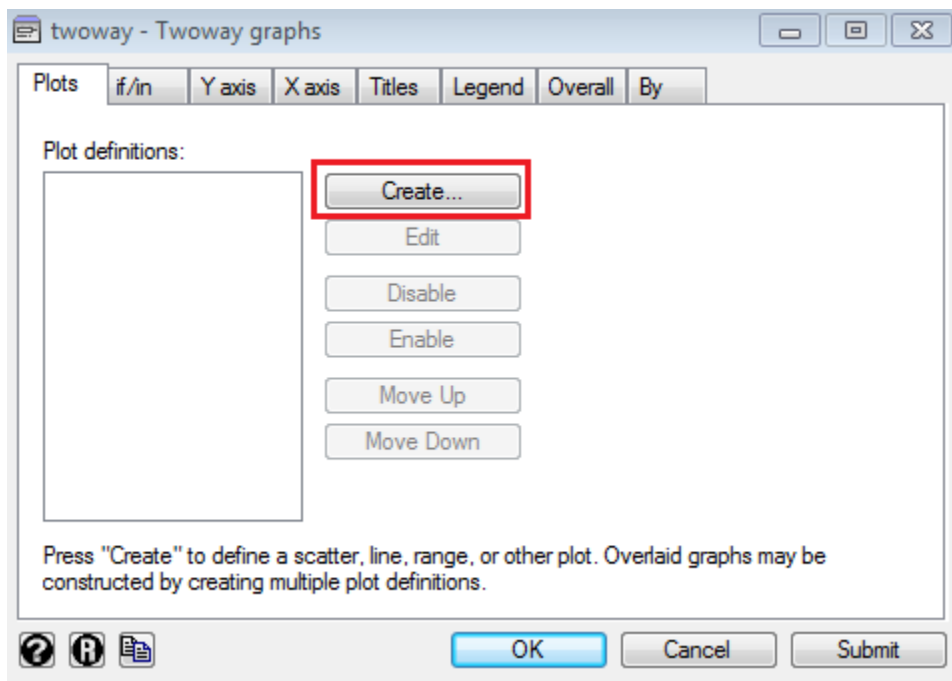
<sup>6</sup> Fisman, R. and Miguel, E. (2007) "Corruptions, Norms and Legal Enforcement: Evidence from Diplomatic Parking Tickets". *Journal of Political Economy*, Vol 115, No. 6: pg. 1020-1048. I thank the LSE's Methodological Institute for the availability of this data.

to see whether diplomats from highly-corrupt countries accumulated more unpaid parking violations, on average, than those from less corrupt countries. The variables in the Excel document include:

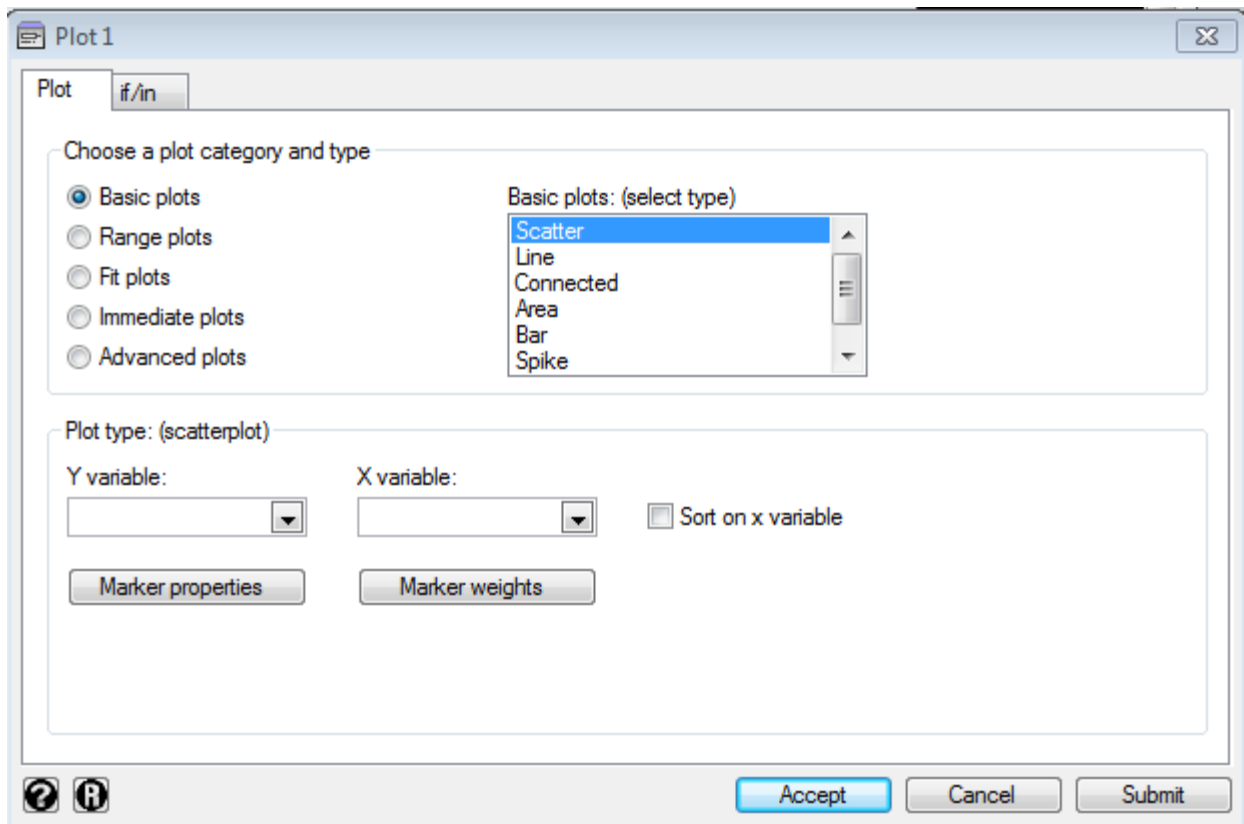
- Violrate: The number of unpaid parking ticket violations per UN diplomat by country (1997-2002 average)
- Corruption: The country's 1998 corruption index, ranging from -2.5158 (least corrupt) to 1.502979 (most corrupt)
- LogGDP: The logarithm of the country's GDP per capita (in 2000 US dollars) in 1998 (if you are unsure about what a log is, don't worry, we'll cover this in Lesson 8!)
- Regional Dummies: Based on whether the country lies in Europe and North America, Latin America, Africa, the Middle East, Asia or Oceania (if you are unsure about what a dummy variable is, don't worry, we'll also cover this in Lesson 8!)

Both authors hypothesize that diplomats from countries with higher corruption indices should be more prone to unpaid parking violations than those from less corrupt countries, given cultural norms associated with the abuse of power for private gain. We will assess the authors' hypothesis via a scatter-plot, a Pearson correlation coefficient, and a basic univariate regression analysis.

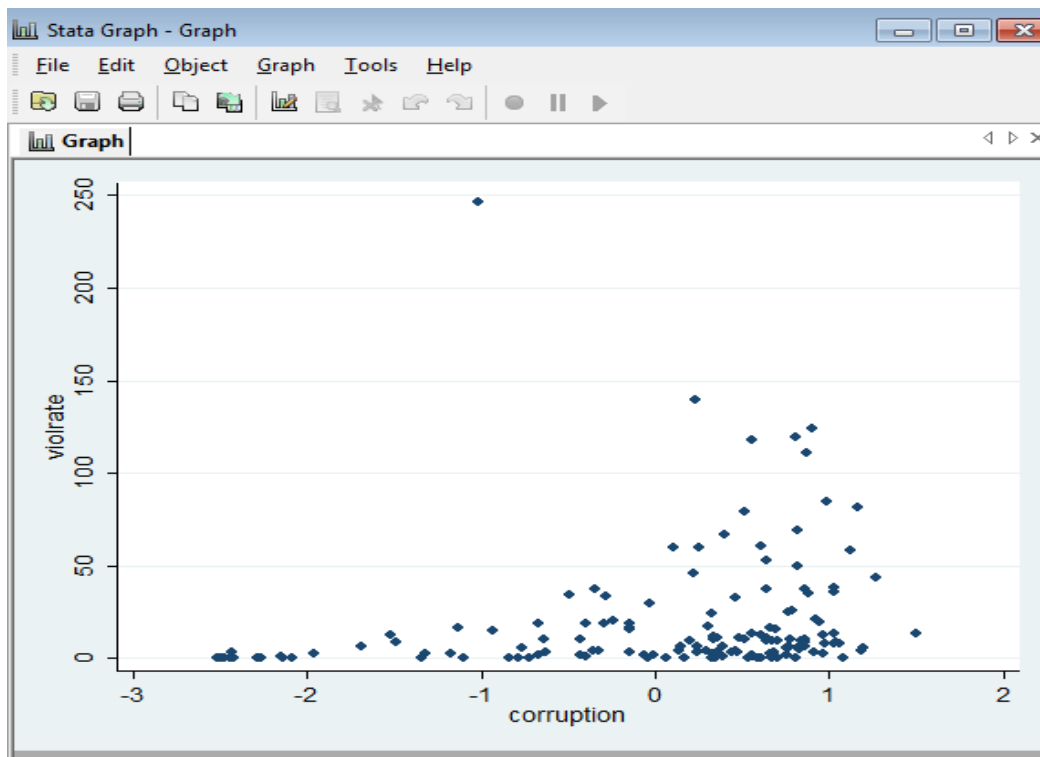
Copy and paste the data into the STATA data editor. We begin first by making a simple scatter plot, with our dependent variable (parking violation rates) on the y-axis and our independent variable (corruption index) on the x-axis. Click on "Graphics" and then "Twoway graph (scatter, line, etc.)". You should be presented with the following box:



Click on the "Create" box. You should be presented with the following box:



A number of graphs are available to you. Highlight the “Scatter” option in the “Basic plots” box, and specify that your Y (dependent) variable is parking violations per diplomat (violrate), and your X (independent) variable is corruption. Just a note for scatter plots: your dependent variable should always be on the y-axis! Click “Accept”. This will send you back to the initial twoway graph box, but “Plot 1” should be in your “Plot definitions” box. Click “ok”. You should be presented with the following graphic:



**CONGRATULATIONS! You have just made a scatter plot in STATA!**

The above scatter plot can be reproduced by entering the code below in the STATA command box.

### STATA COMMAND 6.1:

*Code:* “**twoway (scatter var1 var2)**”, where var1 is your dependent variable and var2 is the independent variable of interest.

*Output produced:* Produces a scatter plot graphic which demonstrates relationships between your dependent and independent variable

*Caveats:* Does not produce a best-fit line

You may notice that though a slight positive trend can be detected within the data, the command above fails to produce a predicted best fit line. We can produce this, along the lines of the estimated  $Y = \hat{\beta}_0 + \hat{\beta}_1 X + \varepsilon$  univariate equation by altering the code in the command box as specified in the STATA Command 6.2 box below.

## STATA COMMAND 6.2:

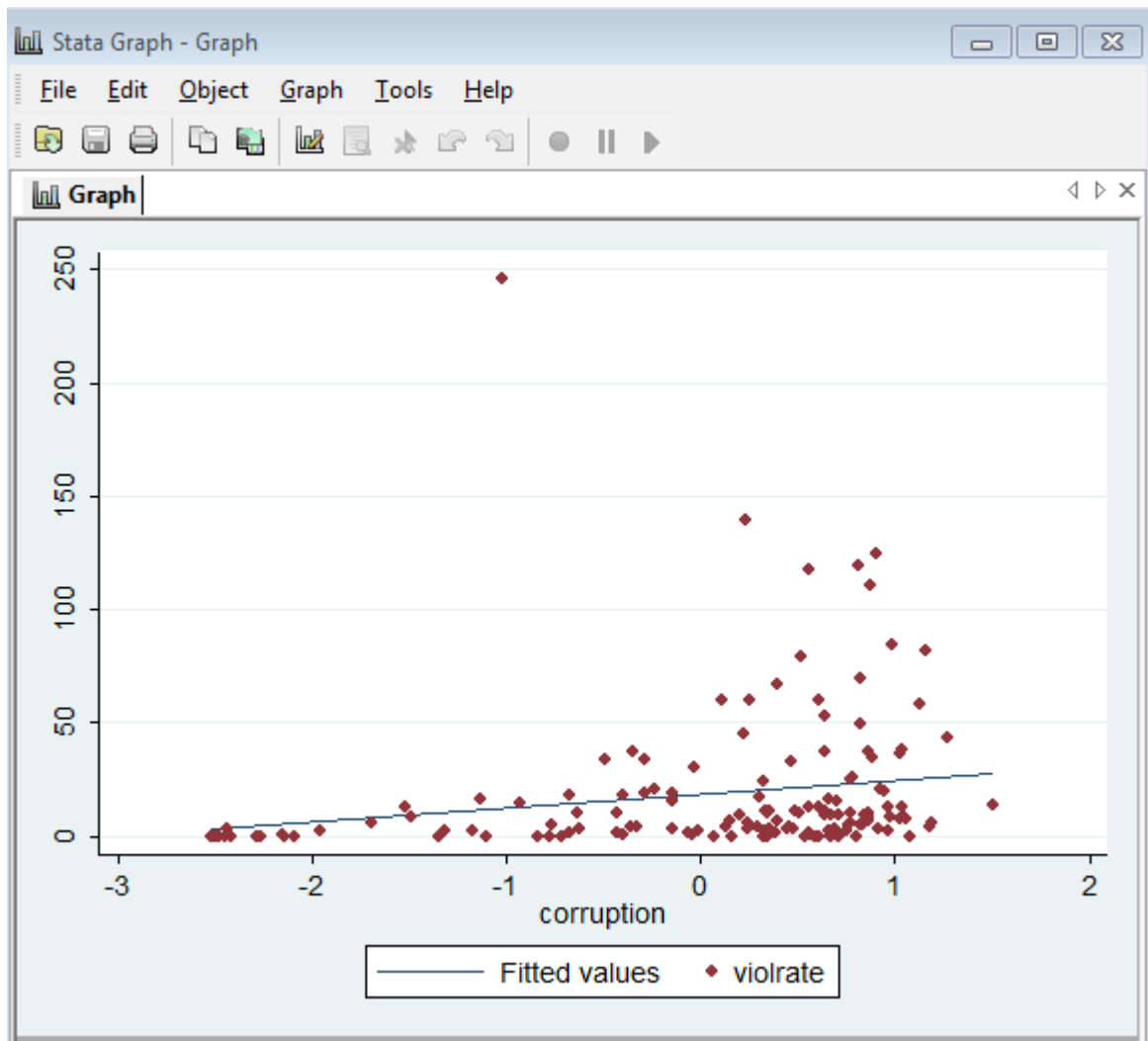
*Code:* “**graph twoway (lfit var1 var2) (scatter var1 var2)**”, where var1 is your dependent variable and var2 is the independent variable of interest.

*Output produced:* Produces a scatter plot graphic with a best fit line

*Modifications:* “**graph twoway (lfitci var1 var2) (scatter var1 var2)**”, where var1 is your dependent variable and var2 is the independent variable of interest.

*Output produced:* Produces a scatter plot graphic and a best fit line with  $\hat{\beta}_1$ 's estimated 95% confidence intervals

Type the following code into your command box, followed by enter: “graph twoway (lfit violate corruption) (scatter violate corruption)”. You should be presented the following image:

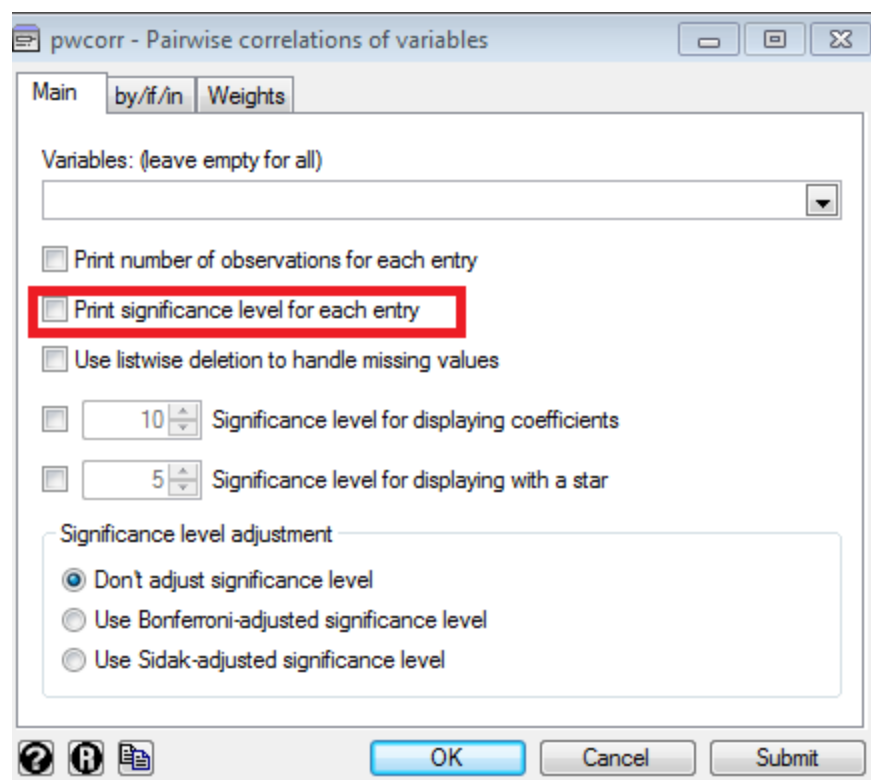


**CONGRATULATIONS! You have just made a scatter plot with a corresponding best-fit line in STATA – you can also regenerate this with 95% confidence intervals as specified above!**

The blue line you see above is the fitted linear regression equation which STATA has predicted best fits the data. It not only includes the y-intercept (not shown above), but also the slope ( $\hat{\beta}_1$ ). The value of the slope, however, is not presented; in order do so, we must estimate a univariate regression model. Before we move on to univariate modeling however, we will briefly cover correlation coefficients.

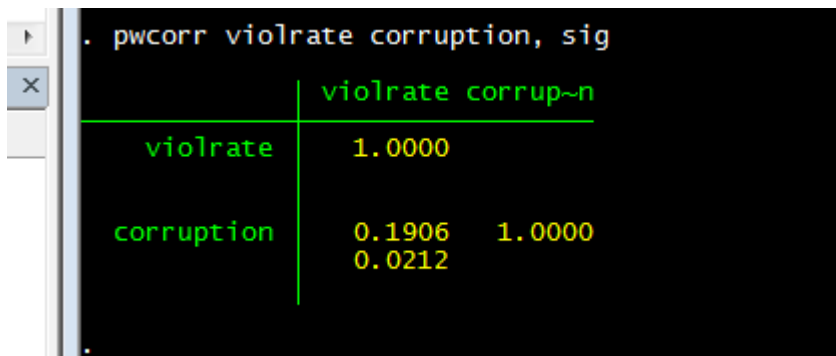
Correlation coefficients measure the strength of the linear relationship between two variables, and are defined in terms of covariance. Correlation values range between -1 and 1. A value of -1 indicates perfect negative collinearity – i.e. a one unit increase in standardized X corresponds exactly with a 1 unit decrease in standardized y – while a value of 1 indicates (perfect positive collinearity – i.e. a one unit increase in X corresponds exactly with a 1 unit increase in Y). It is helpful to conduct correlation coefficient tests to assess whether there is some degree of association between your dependent and independent variable. This command will become particularly useful when we assess the strength of the no perfect multicollinearity assumption in OLS in Lesson 10.

In order to calculate pair-wise correlation coefficients in STATA, click on the “Statistics” tab, then “Summaries, tables and tests”, “Summary and descriptive statistics”, and “Pairwise correlations”. You should be presented with the following box:



For pair-wise correlations, you can add as many variables as you like, which is especially useful when you want to determine whether multicollinearity exists in multiple independent variables but for now we

will stick with two variables. Within the variables box, enter “violrate” and “corruption”, and click the “Print significance level for each entry” box, then click “ok”. You should see the following output:



	violrate	corruption
violrate	1.0000	
corruption	0.1906 0.0212	1.0000

**CONGRATULATIONS! You have just calculated a correlation coefficient for two variables, and calculated its corresponding significance level in STATA!**

The significance value STATA gives you below the correlation coefficient is a p-value; hence, harking back to significance testing, lower p-values indicate the likelihood of a significant relationship between the two variables.

### STATA COMMAND 6.3:

*Code:* “***pwcorr var1 var2 var3 ...***”, where var1, var2, var3... are your variables of interest

*Output produced:* Calculates pair-wise correlation coefficients for the specified variables, with corresponding significance levels (p-values)

We can interpret the above pair-wise correlation coefficient as follows: a 1 unit increase in a standardized measure of corruption is significantly associated (at over a 95% confidence level) with a 0.191 unit increase in a standardized measure of parking violation rates. Pair-wise correlation coefficients are helpful for grasping basic linear relationships between variables. They are not equivalent, however, to the predicted slope of our best fit line in the above scatter plot, because the above scatter plot is expressed not in terms of standardized data, but the data’s original measurement. In order to estimate  $\hat{\beta}_1$ , we need to conduct a formal regression model.

To calculate the formula for the best-fit line in the scatter plot above, click on the “Statistics” tab, and then on the “Linear modes and related” option. Click on the “Linear regression” tab, you should see the following box:

regress - Linear regression

Model by/if/in Weights SE/Robust Reporting

Dependent variable: violrate

Independent variables: corruption

Treatment of constant

☐ Suppress constant term

☐ Has user-supplied constant

☐ Total SS with constant (advanced)

OK Cancel Submit

Remember how to access this box as we will be using this, as well as its corresponding coded command, quite frequently in future lessons. In the dependent variable box, specify your dependent variable (violrate), and in the independent variable box, specify your independent variables (since we are doing univariate analysis for now, only specify corruption as your independent variable – we will cover multivariate analysis later). There are numerous options with the linear regression command which we will explore throughout the following weeks. For now, we concentrate simply on estimating a basic model, assuming all OLS assumptions are fulfilled. Click “Ok” and you should see the following output:

```
. regress violrate corruption
```

Source	SS	df	MS
Model	5740.17109	1	5740.17109
Residual	152204.406	144	1056.97504
Total	157944.577	145	1089.27295

	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
violrate					
corruption	6.201699	2.661219	2.33	0.021	.9415998 11.4618
_cons	18.79664	2.693977	6.98	0.000	13.47179 24.12148

Number of obs = 146  
F( 1, 144) = 5.43  
Prob > F = 0.0212  
R-squared = 0.0363  
Adj R-squared = 0.0297  
Root MSE = 32.511

**CONGRATULATIONS!** You have just conducted a univariate regression model within STATA, and predicted the formula for a best-fit line!

## STATA COMMAND 6.4:

*Code:* “**regress var1 var2**”, where var1 is your dependent variable and var2 is your independent variable. “reg” is also used as shorthand coding for “regress” in STATA.

*Output produced:* Conducts a univariate OLS estimated model (best-fit line) for your data

There are a number of important outcomes to interpret from this output. First is the estimated coefficient for corruption; this is the estimated slope of the best-fit line in the scatter plot above. We can interpret it as a 1 unit increase in a country’s corruption index will lead to 6.2 more parking violations on average. Notice that this is not synonymous with a 1% increase, which we can assess with a log transformation of our dependent and independent variables (we will discuss this in Lesson 8). Whenever you are interpreting your beta coefficients, careful consideration must be paid to the range of your independent variable; in our case, corruption varies from -2.5 to 1.5, so a 1 unit increase is quite large in relation to the overall range. If we wanted a standardized beta (i.e. a beta coefficient that is expressed in terms of -1 to 1, like a Pearson correlation coefficient), type the following command into the STATA command window: “reg violrate corruption, beta”. You should see the following output:

```
. reg violrate corruption, beta
```

Source	SS	df	MS
Model	5740.17109	1	5740.17109
Residual	152204.406	144	1056.97504
Total	157944.577	145	1089.27295

Number of obs =	146
F( 1, 144) =	5.43
Prob > F =	0.0212
R-squared =	0.0363
Adj R-squared =	0.0297
Root MSE =	32.511

violrate	Coef.	Std. Err.	t	P> t	Beta
corruption	6.201699	2.661219	2.33	0.021	.1906383
_cons	18.79664	2.693977	6.98	0.000	.

**CONGRATULATIONS! You have just conducted a univariate regression model within STATA with standardized beta coefficients, and predicted the formula for a best-fit line!**

Notice that your output is identical to that above, with one exception: you are provided with a column which expresses the standardized beta coefficient of corruption. Also notice that this coefficient is identical to your pair-wise correlation coefficient. Once we pursue multiple regression analysis in Week 3, standardized beta coefficients will provide the means to compare the magnitude of effects of two independent variables on y.

### STATA COMMAND 6.5:

*Code:* “**regress var1 var2, beta**”, where var1 is your dependent variable and var2 is your independent variable

*Output produced:* Conducts a univariate OLS estimated model (best-fit line) for your data with standardized beta coefficients

Turn to the t-statistic/p-value of our beta coefficient to determine significance; if we have a high p-value (i.e. greater than 0.1) we cannot reject the null hypothesis that corruption has no influence parking violations! In our case, we are presented with a relatively low p-value (0.021), so we therefore can reject the null hypothesis, with over 95% confidence, that corruption has no influence on parking violations (the 95% confidence interval of our estimated beta coefficient also confirms this).

The other two statistics of interest is the F-statistic, and the R-squared. The F-statistic and its corresponding p-value tell us the significance of our model. If the F-statistic produces a corresponding p-value that is lower than 0.1, our estimated model - which includes all our independent variables - is significant on at least a 90% confidence level. If our F-statistic’s p-value is higher than 0.1, we cannot reject the null hypothesis that our estimated model is insignificant. Lower F-statistics (with corresponding p-values greater than 0.1) indicate that we’ve produced a model that fails to explain our dependent variable – hence we have to throw it out! Since we have produced only a univariate regression model, the entire model consists of one independent variable (corruption); hence the F-statistic’s p-value perfectly corresponds with the p-value of corruption’s estimated beta coefficient.

Finally, the R-squared, or the coefficient of determination, describes the degree of variation explained by the model. R-squared values range from 0 to 1 with 0 indicating no variation is explained by the model and 1 indicating that 100% of variation is explained by the model. In our case, we have a very low R-squared (roughly 3.63%/2.97% of the variation in traffic violations can be determined by corruption alone). Low R-squares indicate that much of our model is explained by our error term,  $\epsilon$ . Generally, we prefer higher R-squares because this indicates our model explains a greater proportion of variation in our data. However, a low R-squared value does not necessarily indicate that you have a model that should be abandoned. Whether a high/low R-squared is “good” depends on the nature of variables under consideration. Most associations between variables in the social sciences, for example, involve much more unexplained variation than more concrete models in the sciences. Moreover, models for aggregates (such as countries or states) generally produce higher R-squared values than those which model the characteristics of individuals, which exhibit much greater variation between each other. Hence, achieving a high R-squared should not be the ultimate criterion of your model; models with low R-squares can be equally useful if it explains how variables relate to each other. What really matters is significance (i.e. your F-statistic).

---

### Practice Problems:

Practice Problem 1: Without running any commands in STATA, establish the null and alternative hypothesis for the impact of log per capita GDP on parking violations. Do you anticipate the influence of per capita income on parking violations to be zero, significantly positive, or significantly negative? Why?

Practice Problem 2: Create a scatter plot (with 95% confidence intervals attached) demonstrating the relationship between per capita income (logGDP) and parking violations. Specify your dependent and independent variable. What type of relationship do you perceive on your scatter plot?

Practice Problem 3: Calculate pair-wise correlation coefficients between parking violations, corruption and per capita GDP. What can you say about the association between parking violations and a country's per capita income? The association between corruption and a country's per capita income?

Practice Problem 4: Conduct a univariate regression analysis, examining the impact of national (per-capita) income on parking violations with and without standardized beta coefficients. How can you interpret your results relative to the hypotheses that you created in question 2? Did your suspicion on the influence of per capita income on UN parking violations reveal itself in the data? Is your model significant? How much variation does it explain?

Practice Problem 5: Conduct a multivariate regression analysis, examining the impact of both corruption and (per capita) income on parking violations (you can do this by adding the additional independent variable onto the "regress" code). What do you notice about your beta coefficients' significance for both variables (this outcome is a problem of multi-collinearity between both independent variables. We will address this outcome in greater depth in Lesson 10).

## Lesson 7: Multivariate (OLS) Regression Analysis

---

**Learning Objective 1: Formulate a hypothesis and test it via multivariate regression analysis while factoring in control variables**

**Learning Objective 2: Conduct a multivariate regression in STATA and interpret its beta coefficients, F-statistic, and reported R-squared**

**Learning Objective 3: Analyze how beta coefficients change in multivariate analysis as controls are added**

While univariate regression models are helpful in examining the influence of one independent variable on a dependent variable, generally we want to incorporate more than one independent variable into a model. The reasons for doing so are twofold. One, we may want to test multiple hypotheses, (i.e. how several independent variables influence a dependent variable – the world is a bit more complicated than a one variable model!). Two, even if we are interested in examining the impact of one independent variable, we still need to control for alternative variables that could influence the dependent variable, as omitting them may skew our results (we will discuss problems with omitted variable biases in Lesson 9). The baseline model for a multivariate regression is similar to a univariate model, except it includes more than one independent variable. Multivariate regression can be expressed as:

$$Y = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\beta}_3 X_3 + \hat{\beta}_4 X_4 + \dots + \hat{\beta}_n X_n + \varepsilon$$

The number of independent variables in a multivariate linear regression is arbitrary and will depend on the research question that you are interested in examining; you may wish to test multiple hypotheses and include multiple controls, or you may be interested in two or three. If you have a small sample size however, some caution must be paid to how many controls you use for degrees-of-freedom purposes. Recall from Lesson 4 that significance testing requires the degrees of freedom of your test, which is equal to the number of observations minus the number of estimated parameters (i.e. the number of independent variables plus your constant term) in your model. If you have low degrees of freedom, your estimates will become less reliable. If you have more parameters than observations, you will have negative degrees of freedom, which means you will be unable to make any claims about variable relationships with your constructed model.

Hypothesis testing for multivariate analysis is identical to that of univariate analysis; the null hypothesis assumes our beta coefficient equals zero (hence the independent variable in question has no impact on the dependent variable), while the alternative hypothesis assumes the contrary. Both univariate and multivariate OLS regression models are subject to the seven following crucial assumptions (you should memorize these):

1. The regression model is linear, correctly specified and has an additive error term,  $\varepsilon$
2. The error term,  $\varepsilon$ , has a zero population mean

3. All explanatory variables are uncorrelated with the error term,  $\epsilon$
4. Observations of the error term are uncorrelated with each other (no serial correlation); this assumption applies mostly to time series linear regression, which we will not be discussing in this class, but can also apply to cross-sectional data if there is spatial correlation between observations. We will ignore autocorrelation for now.
5. The error term has a constant variance (no heteroskedasticity)
6. No explanatory variable is a perfect linear function of any other explanatory variable(s) (no perfectly multicollinearity)
7. The error term is normally distributed<sup>7</sup>

If assumptions 1-6 are fulfilled, then OLS is said to be BLUE: the Best Linear Unbiased Estimator. Rarely do the models that we want to test fulfill all these assumptions. For now, we will assume in the regressions we conduct below that the assumptions are fulfilled. In succeeding lessons, we will test the validity of these assumptions, particularly Assumption 1 (Lessons 8 and 9), Assumptions 5 and 6 (Lesson 10) and Assumption 7 focusing particularly on cases where the error term is binomially distributed (Lessons 11 and 12), and what alternative techniques you can introduce to correct for these failed assumptions.

In this lesson, you will conduct hypothesis testing using multivariate analysis. The dataset you have been provided contains data on American, European, and Asian vehicles manufactured between 1970 and 1982. You will test how characteristics of these vehicles impact the miles per gallon they attain on the highway. The variables in the Excel/STATA file include:

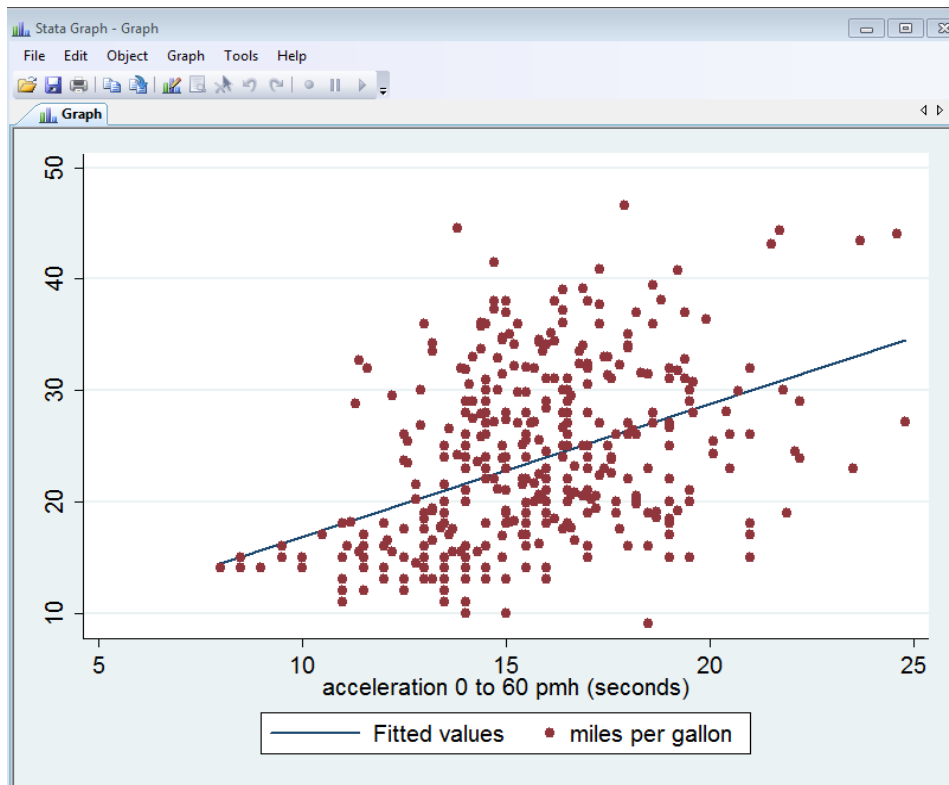
- Name: The model and make of the vehicle
- mpg: Miles per gallon the vehicle obtains on the highway
- Cylinder: The number of cylinders the vehicle has
- Horsepow: The horsepower of the vehicle's engine (in kW)
- Weight: The weight of the vehicle (in 1000 lbs)
- Acceleration: The time period of acceleration of the vehicle from 0 to 60 mph (in seconds)
- Year: The year the vehicle was manufactured
- Origin: The region the vehicle was manufactured in (1 for North America, 2 for Europe, 3 for Asia)
- European: A dummy variable for whether the vehicle was manufactured in Europe (1 for yes, 0 for no)
- Asian: A dummy variable for whether the vehicle was manufactured in Asia (1 for yes, 0 for no)

We will determine how a vehicle's characteristics (number of cylinders, acceleration time, horsepower, weight, etc.) influences its miles per gallon, while controlling for other attributes, specifically when the vehicle was made, and where. Copy and paste the data into the STATA data editor. Let's begin by determining the relationship between acceleration time (the independent variable) and mpg (dependent variable).

---

<sup>7</sup> This assumption is a requirement for hypothesis testing. It is also important regarding the influence of outliers, which may potentially skew your results.

Start off by creating a simple scatter plot (with a fitted best fit line) to roughly eyeball the relationship between acceleration time and mpg via the “graph twoway (lfit mpg accel) (scatter mpg accel)” command in STATA. You should see the following screen:



The above univariate scatterplot diagram illustrates a positive linear relationship between acceleration time and mpg: it appears that cars with faster acceleration times achieve low mpg on the highway, while vehicles with slower acceleration times obtain higher mpg. To formally test the hypothesis that acceleration time has a significant, positive effect, run a brief univariate regression analysis via the “regress mpg accel” command (alternatively, you can click on “Statistics”, then “Linear Models and Related” and then “Linear Regression” tabs in the toolbar). You should see the following output:

```
. regress mpg accel
```

Source	SS	df	MS
Model	4284.04181	1	4284.04181
Residual	19968.5334	396	50.4255893
Total	24252.5752	397	61.08961

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
accel	1.191204	.1292364	9.22	0.000	.9371292 1.44528
_cons	4.969794	2.043208	2.43	0.015	.9529032 8.986685

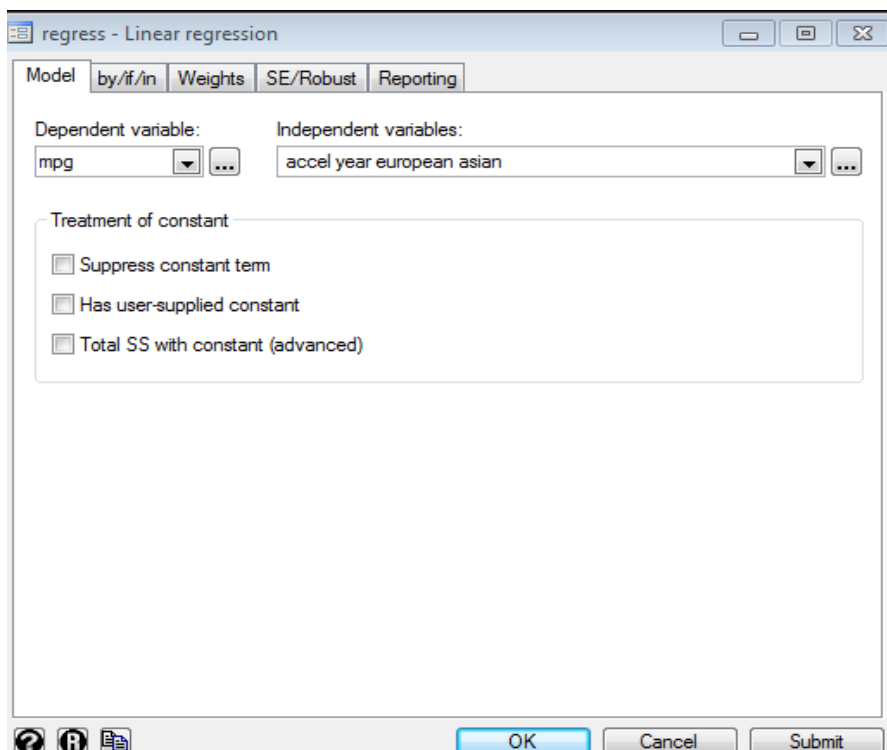
  

Number of obs =	398
F( 1, 396) =	84.96
Prob > F =	0.0000
R-squared =	0.1766
Adj R-squared =	0.1746
Root MSE =	7.1011

The above output indicates a significant positive relationship between acceleration and mpg (notice the high t-statistic and low corresponding p-value; given these values, we can reject the null hypothesis that acceleration has no impact on mpg with high confidence). Regarding beta coefficient interpretation, an increase in acceleration time by one second, on average, should enhance mpg by 1.19. The model's R-squared value indicates that roughly 17.66% of variation in the data can be explained by acceleration alone. Moreover, the F-statistic (also boxed in yellow) is highly significant, producing a low p-value. Therefore, we can reject the null hypothesis that the overall model is not significant.

Let's now move to multivariate regression analysis, and see how adding controls impacts the beta coefficient of acceleration. One problem with univariate analysis that we will discuss in greater depth in Lesson 9 is that our beta coefficients may be influenced by an omitted variable bias. Omitted variable biases can force the expected value of an estimated coefficient away from the true value of the population coefficient (i.e. overstate/understate it), because we are not controlling for other factors that influence the dependent variable. In order to avoid this problem, we must bring in other independent variables - including those we are not interested in measuring the effects of - that may influence y.

For the next regression model, specify that you also want to control for the year the car was manufactured, whether it was made in Europe (the European dummy), and whether it was made in Asia (the Asian dummy)<sup>8</sup>. Click on the "Statistics" tab, followed by "Linear models and related", and then "Linear Regression". You should see the following box:



<sup>8</sup> This implies vehicles manufactured in North America are the baseline category. Do not worry about the interpretation of these dummy variables for now – we will come to this in Lesson 8.

In the “Dependent variable” box, specify that mpg is the dependent variable. In the “Independent variable” box, specify that you want to control for acceleration, year, whether the vehicle was made in Europe, and whether it was made in Asia. Click “Ok”; you should see the following output:

. regress mpg accel year european asian						
Source	SS	df	MS	Number of obs = 398		
Model	14684.6138	4	3671.15345	F( 4, 393) = 150.79		
Residual	9567.96137	393	24.3459577	Prob > F = 0.0000		
Total	24252.5752	397	61.08961	R-squared = 0.6055		
				Adj R-squared = 0.6015		
				Root MSE = 4.9342		
mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
accel	.5006474	.0967777	5.17	0.000	.3103807	.6909142
year	.9664675	.0710486	13.60	0.000	.8267845	1.10615
european	6.733047	.6879087	9.79	0.000	5.380606	8.085488
asian	8.026009	.6535291	12.28	0.000	6.741158	9.310859
_cons	-60.5181	5.171622	-11.70	0.000	-70.6856	-50.35059

**CONGRATULATIONS! You have just conducted a multivariate regression in STATA!**

The above output tells you the estimated effect of acceleration time on mpg, while controlling for the year the car was manufactured, whether it was made in Europe, and whether it was made in Asia. We will not focus on interpretation of the dummy variables for this lesson. The year the car was made produces a significant beta coefficient (t-statistic of 13.60); we can interpret this as a vehicle obtains 0.97 mpg, on average, more than a model made the year before. Inclusion of three control variables increases the R-squared of the model to roughly 0.6 (60% of the variation in our data can now be explained by our model), and the F-stat to 150.79.

For the acceleration coefficient, notice that with the inclusion of our three control variables, the beta coefficient of acceleration, while still significantly positive, drops from 1.19 in the univariate model, to 0.5. Comparing the confidence intervals of the two beta coefficients, you may also notice that there is no overlap between the two; hence acceleration’s beta coefficient in the second model is significantly less than in the first. This is the omitted variable bias in action! As we include more controls in our model that also explain our dependent variable, the beta coefficient of our independent variable of interest should become more reflective of the population’s true beta value. You may find that including more controls reduces the value of the beta coefficient of interest. This is desired, however, because you want to produce a regression model that is BLUE (Best Linear Unbiased Estimator).

### STATA COMMAND 7.1:

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable and var2, var3... are your independent variables

*Output produced:* Conducts a multivariate OLS estimated model (best-fit line) for your data

Multivariate regression analysis, compared to its univariate counter-part, overcomes omitted variable biases and enables one to test multiple hypotheses regarding the impact of an independent variable on a dependent variable. You may be tempted to include all variables that could influence the dependent variable in the equation. This approach, while helpful in addressing omitted variable biases, can create other complications, however, namely they can reduce the statistical accuracy of some of your independent variables and cause them to lose significance (this is the imperfect multicollinearity problem which is slightly different from perfect multicollinearity, an assumption of OLS. We will distinguish between these two types of multicollinearity in Lesson 9). To demonstrate this, add the weight of the car, “weight”, as a further control to the regression above. You should see the following output:

```
. regress mpg accel year european asian weight
```

Source	SS	df	MS	Number of obs = 398		
Model	19881.0948	5	3976.21897	F( 5, 392) = 356.56		
Residual	4371.48034	392	11.1517356	Prob > F = 0.0000		
Total	24252.5752	397	61.08961	R-squared = 0.8198		
				Adj R-squared = 0.8175		
				Root MSE = 3.3394		

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
accel	.0546713	.0686798	0.80	0.426	-.0803556	.1896982
year	.7612705	.049016	15.53	0.000	.6649033	.8576376
european	2.08001	.5130515	4.05	0.000	1.071333	3.088687
asian	2.25384	.5168514	4.36	0.000	1.237693	3.269987
weight	-5.834922	.2703035	-21.59	0.000	-6.366348	-5.303497
_cons	-18.68176	4.000883	-4.67	0.000	-26.54764	-10.81589

Notice that while even more variation in the data is explained by the model (roughly 82%, as indicated by R-squared), acceleration’s beta coefficient is not only reduced with the inclusion of weight as a control, but it also becomes insignificant. In other words, if we include weight in the regression model, we can no longer reject the null hypothesis that acceleration has no influence on mpg. This drop in significance can be attributed to the fact that weight is highly correlated with acceleration time. To check this, type “pwwcorr accel weight, sig” into the command box. You should see the following output:

```
. pwwcorr accel weight, sig
```

	accel	weight
accel	1.0000	
weight	-0.4301 0.0000	1.0000

The pair-wise correlation coefficient between acceleration and weight is significantly negative. As pair-wise correlations between independent variables become larger in absolute terms, i.e. approach 1 or -1, we encounter more significant multi-collinearity problems with our independent variables. Unlike omitted variable biases, which misconstrue the true value of the beta coefficient of our independent variable of interest, imperfect multicollinearity misconstrues the standard error of our variable, increasing the likelihood of obtaining an insignificant beta coefficient. Coupling this issue with the omitted variable bias, you will discover in later lessons that oftentimes researchers will encounter trade-offs between omitted variable bias and multicollinearity problems; as more variables are added to a regression, correcting for the former, we run greater risk of encountering the latter. Final model selection can be very sensitive to these specifications, and oftentimes inclusion (or exclusion) of variables on grounds of violated assumptions depends on judgment calls from the researcher.

As you add further controls to your models in STATA, it is helpful to compare your multivariate models side-by-side in order to gauge how beta coefficients and standard errors change. Re-run the three above models again in STATA, but in between each regression store the estimates of the model. The coding of this can be summarized as follows:

```

“regress mpg accel”
“estimates store Model1” (make sure there is no space between “Model” and “1”!)
“regress mpg accel year european asian”
“estimates store Model2”
“regress mpg accel year european Asian weight”
“estimates store Model3”

```

After you have re-run these regression and stored their output, type in the following command: “estimates table Model1 Model2 Model3, b(%7.3f) se(%7.3f) stats(N r2)” (Note, you must repeat this command verbatim! STATA is case and coding sensitive!). You should see the following output:

Variable	Model1	Model2	Model3
accel	1.191 0.129	0.501 0.097	0.055 0.069
year		0.966 0.071	0.761 0.049
european		6.733 0.688	2.080 0.513
asian		8.026 0.654	2.254 0.517
weight			-5.835 0.270
_cons	4.970 2.043	-60.518 5.172	-18.682 4.001
N	398	398	398
r2	0.177	0.605	0.820

Legend: b/se

## CONGRATULATIONS! You have just created a table of the regressions you have conducted in STATA!

The “estimates table” command is not only a convenient way to compare regression models for the same dependent variable, but it also provides a concise summary of your work (rather than copying and pasting three separate output tables, the “estimates table” command condenses it into one). There are also different modifications of this command, where instead of presenting the standard error below the beta coefficient, as in the above table, you can present the t-statistic, the p-value, or you can provide significance asterisks (i.e. beta coefficients that are significant on a 90%, 95% and 99% confidence level will be indicated as such with \*, \*\* and \*\*\*, respectively) which is standard presentation style in empirical papers. The box below has exact coding details on these features. One important note however; the “estimates” command should be run for regression models that share the same dependent variable!

### STATA COMMAND 7.2:

*Code:* “**estimates table ModelX ModelY ModelZ..., b(%7.3f) se(%7.3f) stats(N r2)**”, where ModelX, ModelY, ModelZ... are the regression models conducted (whose estimates have been stored!), “b” indicates your beta coefficient, “se” indicates your standard error, “N” indicates the number of observations in your sample, and “r2” is the r-squared of your model.

*Output produced:* Produces a regression output table of multiple equations, with the standard error listed below the beta coefficient.

### STATA COMMAND 7.3:

*Code:* “**estimates table ModelX ModelY ModelZ..., b(%7.3f) p(%7.3f) stats(N r2)**”, where ModelX, ModelY, ModelZ... are the regression models conducted (whose estimates have been stored!), “b” indicates your beta coefficient, “p” indicates your p-value, “N” indicates the number of observations in your sample, and “r2” is the r-squared of your model.

*Output produced:* Produces a regression output table of multiple equations, with the p-value listed below the beta coefficient.

### STATA COMMAND 7.4:

*Code:* “**estimates table ModelX ModelY ModelZ..., b(%7.3f) t(%7.3f) stats(N r2)**”, where ModelX, ModelY, ModelZ... are the regression models conducted (whose estimates have been stored!), “b” indicates your beta coefficient, “t” indicates your p-value, “N” indicates the number of observations in your sample, and “r2” is the r-squared of your model.

*Output produced:* Produces a regression output table of multiple equations, with the t-statistic listed below the beta coefficient.

### **STATA COMMAND 7.5:**

*Code:* “`estimates table ModelX ModelY ModelZ..., b(%7.3f) star(.1 .05 .01) stats(N r2)`”, where ModelX, ModelY, ModelZ... are the regression models conducted (whose estimates have been stored!), “b” indicates your beta coefficient, “star” indicates whether the beta coefficient is significant on a 90%, 95% and 99% confidence level, “N” indicates the number of observations in your sample, and “r2” is the r-squared of your model.

*Output produced:* Produces a regression output table of multiple equations, with the beta coefficients starred for significant levels.

---

### **Practice Problems:**

Practice Problem 1: Conduct a univariate regression model which examines the linear impact of a vehicle's horsepower on its mpg. What can you claim about the impact of horsepower on mpg (in terms of the impact of a marginal change of horsepower on mpg, as well as its significance)? How well does horsepower alone explain the variation in the data? Is the model significant?

Practice Problem 2: Conduct a multivariate regression analysis which examines the impact of a vehicle's horsepower on its mpg, while controlling for year the vehicle was made, whether it was made in Europe, and whether it was made in Asia. What happens to the influence of horsepower on mpg with the inclusion of these three controls? Is the difference between the new beta coefficient, and that in the linear model significant (hint, you will need to look at the betas' confidence intervals for this). How does the new model explain the variation in the data?

Practice Problem 3: Conduct a multivariate regression analysis which examines the impact of a vehicle's horsepower on its mpg, while controlling for year the vehicle was made, whether it was made in Europe, whether it was made in Asia, and weight. What happens to the influence of horsepower on mpg with the inclusion of the last control? What do you think is driving this (hint, the "pwcrr" command will be helpful).

## Lesson 8: Constants, Dummy Variables, Interaction Terms, and Non-Linear Variables in Multivariate OLS Regressions

---

**Learning Objective 1: Examining the influence of the exclusion of a constant term on estimated beta coefficients**

**Learning Objective 2: Conducting a multivariate regression in STATA with dummy variables and interpreting their output**

**Learning Objective 3: Comparing the magnitude of dummy variables via an F-test of variable equivalence**

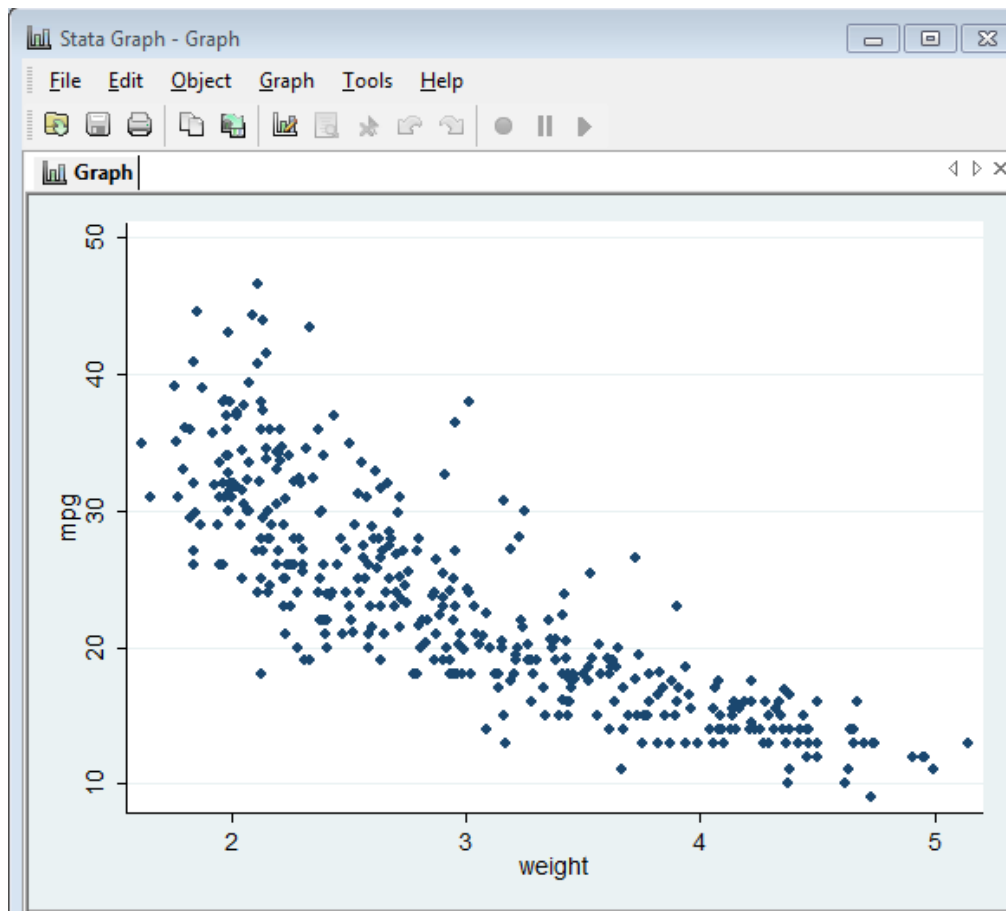
**Learning Objective 4: Creating interaction terms between dummy variables and other variables, interpreting their output in linear regression models, and graphing their influence on a dependent variable**

**Learning Objective 5: Creating quadratic and inverse independent variables in STATA, understanding when to use them, and interpreting their output**

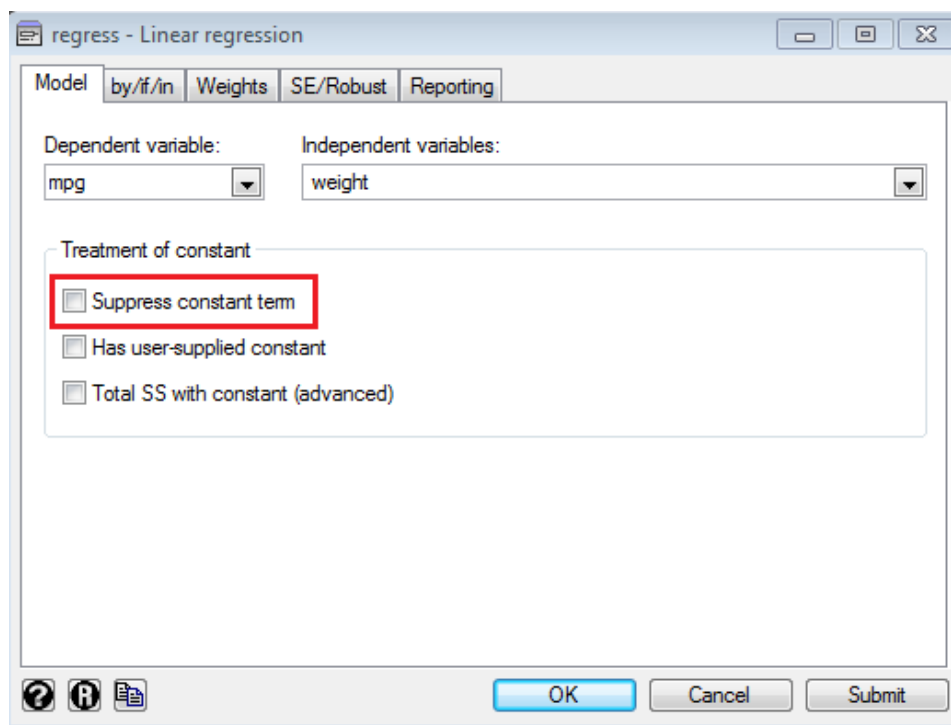
The functional form of a regression model is vital for its proper specification. Thus far, we have focused predominately on the interpretation of the influence of interval independent variables (i.e. real numbers) on a dependent variable. We have ignored other crucial aspects of the basic OLS model, specifically the presence of the constant term ( $\beta_0$ ), independent variables that are not interval measured (i.e. nominal variables, such as binary categorical variables), and independent variables that are interval measured but may not have a linear relationship with  $y$ . In this lab, we will examine how these specifications influence OLS estimates.

In all linear regressions we have conducted thus far, we have included a constant term in the model specification (STATA includes a constant term by default). It is possible to exclude a model without a constant term, which would produce a best-fit line with a zero y-intercept. While there may be some circumstances where one might expect a zero value for  $\beta_0$ , excluding  $\beta_0$  introduces a severe bias, and violates the second assumption of OLS, that the error term has an expected value of zero. This classical assumption of OLS can only be met if a constant term absorbs any non-zero mean of the stochastic error. The exclusion of a constant term will also severely impact the estimated beta coefficients on your independent variables, and may in some cases change their sign. While we cannot rely upon the constant term for statistical inference (because it also includes variation from omitted variables that we are unable to account for) it is important to include it in order to avoid biasing the beta coefficients of your independent variables. To demonstrate these effects empirically, we will rely upon the dataset from the previous lesson, which provides data on American, European, and Asian vehicles manufactured between 1970 and 1982.

In this lab, we will focus on the influence of a vehicle's weight on the miles per gallon it obtains on the highway. Begin by creating a scatter plot (without a best fit line), by typing the following command into the STATA command box: "scatter mpg weight". You should see the following image:



Notice that there appears to be a clear negative relationship between weight and the amount of mpg a vehicle obtains on the highway (although not necessarily linear, we will address this below). This negative relationship, however, is contingent upon a non-zero constant. In other words, if we had to plot our best fit line through the origin, we could not demonstrate the same estimate for  $\beta_1$ . To prove this empirically, let's run two basic univariate regressions, assessing the impact of weight on mpg. Click on "Statistics", then "Linear models and related" and then "Linear Regression". You should see the following (familiar) box:



Specify your dependent variable (mpg) and independent variable (weight). Notice that STATA has the option to suppress the constant (boxed in red). For your first regression, do not suppress it. You should be presented with the following output:

```
. regress mpg weight
```

Source	SS	df	MS			
Model	16777.7611	1	16777.7611	Number of obs = 398		
Residual	7474.81412	396	18.8757932	F( 1, 396) = 888.85		
Total	24252.5752	397	61.08961	Prob > F = 0.0000		
				R-squared = 0.6918		
				Adj R-squared = 0.6910		
				Root MSE = 4.3446		

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
weight	-7.67661	.2574869	-29.81	0.000	-8.182822	-7.170398
_cons	46.31736	.7952452	58.24	0.000	44.75393	47.88079

Notice, as expected, the relationship between the weight of the car and the mpg it obtains on the highway, as indicated by  $\hat{\beta}$ , is significantly negative. Now run the regression again, expect this time click the “Suppress constant term” box. You should be presented with the following output:

. regress mpg weight, noconstant						
Source	SS	df	MS	Number of obs = 398		
Model	172814.88	1	172814.88	F( 1, 397) = 959.47		
Residual	71505.8799	397	180.115566	Prob > F = 0.0000		
Total	244320.76	398	613.871255	R-squared = 0.7073		
				Adj R-squared = 0.7066		
				Root MSE = 13.421		
mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
weight	6.746881	.2178152	30.98	0.000	6.318665	7.175096

**CONGRATULATIONS! You have just conducted a regression with a suppressed constant term in STATA!**

Unlike our previous regression, the sign of the beta coefficient on weight is positively significant. With our new model, the increase in weight of a vehicle is associated with higher, rather than lower, mpg, contrary to what one would expect. This is a classic example of biasing problems attributed to excluding a constant term – a zero-constant implies the best fit line must pass through zero, which implies the OLS will always produce a positive beta coefficient. Since our model no longer includes a non-zero constant, STATA has estimated a best-fit line with a reverse (positive) slope, in order to account for variation in our data. The best way to avoid this bias is to estimate all regression models with a constant, even if you suspect it to be zero.

#### STATA COMMAND 8.1:

*Code:* “**regress var1 var2 var3 ..., noconstant**”, where var1 is your dependent variable and var2, var3... are your independent variables

*Output produced:* Conducts an OLS estimated multivariate model (best-fit line) for your data while suppressing the constant term

In the previous lesson, we estimated regression models with dummy variables, a nominal, categorical variable which encompasses two values, 0 or 1. These categories must be mutually exclusive – an observation cannot lie in more than one category. The inclusion of a dummy term is convenient when attempting to gauge the effects of categorical variables on an interval variable in OLS. Their inclusion, however, requires careful interpretation, as they alter not only the intercept (constant term) of our regression model, but they can also impact the slope of our model if we include an interaction term. Even though dummy variables can only assume the values 0 and 1, they need not represent only dichotomous categories. We can also use dummy variables to assess the influence of nominal variables with more than 2 categories, as long as we convert each category into its own dummy variable. For example, say we want to assess the influence of the origin of the vehicle manufacturer on mpg. If we have three regions of

origin as we do in our dataset – Asia, Europe, and North America – we simply codify each region as its own dummy variable (i.e. a separate dummy for Asian cars, a separate dummy for European cars, and a separate dummy for North American cars).

When including dummy variables with  $n$  categories in regression models, only  $n-1$  variables must be included. Hence, if we wanted to assess how an individual's gender influenced their level of income, rather than including two dummy variables, one for "male" and one for "female", you would only need to include one (i.e. one for "male"). This is because dummy variables are relative – we state their effects relative to a baseline category. Furthermore, including both dummy variables in the same equation would violate the perfect multi-collinearity assumption of OLS, as one is perfectly defined by the other.<sup>9</sup> The category which is not included (i.e. "female") is known as the reference/baseline category, the category which the dummy is compared against. Hence, in our model, the beta coefficient on the "male" dummy variable would represent the difference in income relative to that of a "female". Likewise, if we had three categories (as we do with vehicle origin) we would include only two dummy variables; the inclusion of a European and North American dummy variable would be assessed against a baseline category of Asian vehicles.

Reverting back to our dataset, include the European dummy and American dummy in the linear regression model which assesses the influence of weight on mpg (note this means that the reference category is Asian vehicles). Type "reg mpg weight european american" into the STATA command box. You should be presented with the following output:

. reg mpg weight european american						
Source	SS	df	MS		Number of obs = 398	
Model	17013.2642	3	5671.08805		F( 3, 394) = 308.65	
Residual	7239.31103	394	18.3738859		Prob > F = 0.0000	
Total	24252.5752	397	61.08961		R-squared = 0.7015	
					Adj R-squared = 0.6992	
					Root MSE = 4.2865	
mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
weight	-7.023439	.3183983	-22.06	0.000	-7.649411	-6.397467
european	-1.139963	.706544	-1.61	0.107	-2.529031	.249105
american	-2.355435	.6620306	-3.56	0.000	-3.656989	-1.053881
_cons	46.05129	.8560155	53.80	0.000	44.36836	47.73422

The beta coefficients on European and American cars express the difference in mpg that these vehicles obtain relative to those made in Asia. In the case of European vehicles, notice that the beta coefficient lacks significance at the 90% confidence level; hence we cannot reject the null hypothesis that European cars obtain similar mpg *relative to Asian vehicles*. The dummy on North American vehicles, however, is significantly negative. We can interpret this to mean that American cars, on average, obtain (2.36) less mpg than Asian vehicles. This result would also reveal itself if you had selected American cars as the baseline category (i.e. if you replaced the American dummy with the Asian dummy). To check, type "reg

<sup>9</sup> If you attempt to include all dummy categories within a regression equation, STATA will automatically drop one

mpg weight european asian” into the STATA command box. You should be presented with the following output:

. reg mpg weight european asian						
Source	SS	df	MS			
Model	17013.2642	3	5671.08805	Number of obs = 398		
Residual	7239.31103	394	18.3738859	F( 3, 394) = 308.65		
				Prob > F = 0.0000		
				R-squared = 0.7015		
				Adj R-squared = 0.6992		
Total	24252.5752	397	61.08961	Root MSE = 4.2865		
mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
weight	-7.023439	.3183983	-22.06	0.000	-7.649411	-6.397467
european	1.215472	.6523736	1.86	0.063	-.0670966	2.498041
asian	2.355435	.6620306	3.56	0.000	1.053881	3.656989
_cons	43.69586	1.104363	39.57	0.000	41.52467	45.86704

Notice by replacing the Asian vehicle baseline category with American vehicles, STATA has altered nothing regarding the estimated beta coefficient for weight, the F-statistic for the overall model, or the model's R-squared. Also notice, that by including the Asian rather than American dummy, the sign of the beta coefficient is reversed, but the standard error and p-value remain the same; Asian cars still obtain roughly 2.36 mpg, plus or minus the respective standard error, more than American vehicles. This is not coincidental. Regardless of the reference category, dummy variables will perform identically relative to each other and changing reference categories will not alter other (non-dummy) beta coefficients.

The above regressions depict the dummy-intercept effect of European and Asian vehicles relative to American vehicles. We cannot, however, make inferences about the performance of European vehicles relative to Asian ones. To do so, we must do one of two things; 1) re-run the regression with either the Asian or European dummy as the baseline category (as we did before), or, 2) conduct a (restricted) F-test of equivalence after we run the above regression. Recall from Week 1 that one weakness of significance testing was its inability to compare the magnitude of the beta coefficients for two (unstandardized) independent variables. We are unable to compare the magnitude of unstandardized independent variables, because the two variables in question may have different units, variation, and different meanings. Categorical dummies however, do not fall victim to this trap, because such categories are by construction standardized against each other. Hence, we can use an F-test to compare whether their beta magnitudes are equivalent. In our case, the respective null and alternative hypotheses for an F-test of equivalence would be:

$$H_0: \beta_{\text{European}} = \beta_{\text{Asian}}$$

$$H_A: \beta_{\text{European}} \neq \beta_{\text{Asian}}$$

Like all significance testing, if the F-statistic was higher than a 90% critical value, we would reject the null hypothesis that European cars have similar mpg to Asian cars. If the opposite is the case, however, we must fail to reject the null. In order to conduct an F-test of beta equivalence, type “test european=asian” into the STATA command window immediately after the above regression command. You should see the following output:

```
. reg mpg weight european asian
```

Source	SS	df	MS	Number of obs = 398		
Model	17013.2642	3	5671.08805	F( 3, 394) = 308.65		
Residual	7239.31103	394	18.3738859	Prob > F = 0.0000		
Total	24252.5752	397	61.08961	R-squared = 0.7015		
				Adj R-squared = 0.6992		
				Root MSE = 4.2865		

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
weight	-7.023439	.3183983	-22.06	0.000	-7.649411	-6.397467
european	1.215472	.6523736	1.86	0.063	-.0670966	2.498041
asian	2.355435	.6620306	3.56	0.000	1.053881	3.656989
_cons	43.69586	1.104363	39.57	0.000	41.52467	45.86704

```
. test european=asian
```

```
( 1)  european - asian = 0
```

```
      F( 1, 394) = 2.60
```

```
      Prob > F = 0.1075
```

**CONGRATULATIONS! You have just conducted F test of beta coefficient equality in STATA!**

Notice that the corresponding p-value is just above 0.107. Because our p-value (just) exceeds 0.1, we must fail to reject the null hypothesis that there is a difference in mpg performance between European and Asian vehicles. This outcome is identical to what you would have obtained had you compared both dummy variables by re-running the above regression with the Asian dummy as the baseline category (notice, the F-stat p-value is identical to that produced in the regression model where the Asian dummy served as the baseline category). Again, this is not coincidental; dummy variables will always perform identically relative to each other.

## STATA COMMAND 8.2:

*Code:* “**regress var1 var2 var3 var4...**”

“**test var2=var3**”

where var1 is your dependent variable and var2 and var3 are the independent variables whose relative magnitudes you seek to assess

*Output produced:* Conducts an F test of equivalence for the magnitude of the beta coefficients for the specified independent variables

Thus far, we have only examined the impact of intercept dummy variables (i.e. dummy variables that change the intercept of our best fit line, but do not impact our slope). We may want to examine, however, whether the presence of a dummy variable influences the slope of another independent variable. For example, the above regression results indicate that 1) American vehicles have lower mpg than Asian vehicles, and 2) heavier vehicles have lower mpg than lighter ones. One reason for the former result, however, could be attributed to the fact that American car manufacturers, on average, make heavier cars than Asian manufacturers (you can check this via a difference-in-means test of vehicle weight for North American vs. Asian cars). The inclusion of an interaction term (i.e. an independent variable that is the multiple of two or more independent variables) can tell us whether the change in our dependent variable (mpg) with respect to one independent variable (say weight), depends on the level (or presence) of another independent variable (say whether the vehicle was manufacturing in North American compared to Asia). While an interaction term can be between any two types of independent variables (interval or nominally measured), it is called a slope dummy when the term is a multiple of an interval variable and a dummy variable.

The general form of a regression model with an interaction term is as follows:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \epsilon$$

One crucial property to remember with interaction terms is that hierarchy matters! A model with an interaction term should also include the two variables which compose the interaction term separately. When you run a regression with an interaction term, you cannot interpret the  $\beta_3$  coefficient in isolation! The total effect of  $X_1$  on  $y$  depends on  $\beta_1$  and  $\beta_3$ :  $\beta_1$  represents the baseline effect, while  $\beta_3$  represents the partial/conditional effect.

To demonstrate how to interpret interaction terms within a regression model, we turn back to our vehicle data and our regressions assessing the impact of weight and origin of the car on mpg. Recall from above that when we conducted a regression model which examined these variables' influence on mpg, we found that vehicle weight had a significantly negative relationship with mpg, North American vehicles obtained lower mpg than Asian vehicles, and European vehicles failed to obtain significantly different mpg than Asians ones. American vehicles' lower mpg, however, could be attributed to the fact that American manufacturers, on average, make heavier cars than Asian ones. Introducing an interaction term between the American dummy and weight variables would enable us to assess how the beta coefficient on weight changes based upon origin of manufacture – in other words, we could assess whether heavier Asian vehicles obtain better mpg than their American counter-parts.

Create an interaction term between the American dummy and weight variables via the “generate” command (type the following code into the STATA command box: “generate americanweight = american \* weight”). Since our interaction term involves a dummy variable, our interpretation of this interaction term will be relative to baseline category of Asian vehicles. If your interaction terms are slope dummies, and your dummy variable spans over more than two categories, you must also create interactions between other (n-1) dummy coefficients, even if these interactive effects are not your primary focus. Doing so maintains the reference category for your interactive term. Hence, you also should create an interaction between the European dummy and weight in order to maintain a baseline category of Asian vehicles for the interaction term. Run a regression with the hierarchal terms only (i.e. “reg mpg weight european

america”) and then run a second regression which includes the interaction terms (“reg mpg weight european america americaweight europeanweight”). You should see the following output (I present it in estimates table form, with asterisks indicating significance):

```
. estimates table model1 model2, b(%7.3f) star(.1 .05 .01) stats(N r2)
```

Variable	model1	model2
weight	-7.023***	-10.719***
european	-1.140	-8.884**
american	-2.355***	-11.268***
weightamer~n		3.905**
weighteuro~n		3.504*
_cons	46.051***	54.260***
N	398	398
r2	0.702	0.706

legend: \* p<.1; \*\* p<.05; \*\*\* p<.01

## CONGRATULATIONS! You have just conducted an interactive model in STATA!

Notice that the beta coefficients on weight and the American dummy for the baseline and interactive model hold the same negative sign, yet the beta coefficients for the interactive term on the American (and European) dummy and weight hold a (significant) positive sign. We can interpret this to mean that while heavier cars are associated with lower mpg (roughly 10.719 less for each additional ton of weight, as indicated by weight’s beta coefficient), if heavy cars are produced in North America, they are estimated to gain an additional 3.504 mpg (as indicated by the interaction term’s beta coefficient) for each ton of additional weight added, compared to if they are manufactured in Asia. What this means in layman’s terms is that North American vehicles lose less mpg as more weight is added, relative to Asian vehicles. We can also graphically demonstrate this with the “predxcon” command, which estimates predicted values for interaction terms, as well as graphs them.<sup>10</sup> For this command, specify that you wish to examine how weight’s predicted effect on mpg varies by origin by typing the following into the STATA command box: “predxcon mpg, xvar(weight) from(0) to(6) inc(1) graph class(origin)”. You should be presented with the following output and graphic:

<sup>10</sup> You may need to install this via the “findit predxcon” command, if it is not present in your version of STATA.

```
. predxcon mpg, xvar(weight) from(0) to(6) inc(1) graph class(origin)
Predicted values and 95% Confidence Intervals

Model Type:      Linear Regression
Outcome:         -- mpg
X Variable:      -- weight
Class:           -- origin
Interaction:      weight by origin
Covariates:      (none)
Observations:    398
```

---

```
-> origin = 1
```

weight	pred_y	lower	upper
0	42.99228	40.68664	45.29792
1	36.17812	34.51505	37.84119
2	29.36396	28.31195	30.41596
3	22.5498	21.96784	23.13175
4	15.73563	15.05615	16.41512
5	8.921473	7.706652	10.13629
6	2.107312	.2685781	3.946045

---

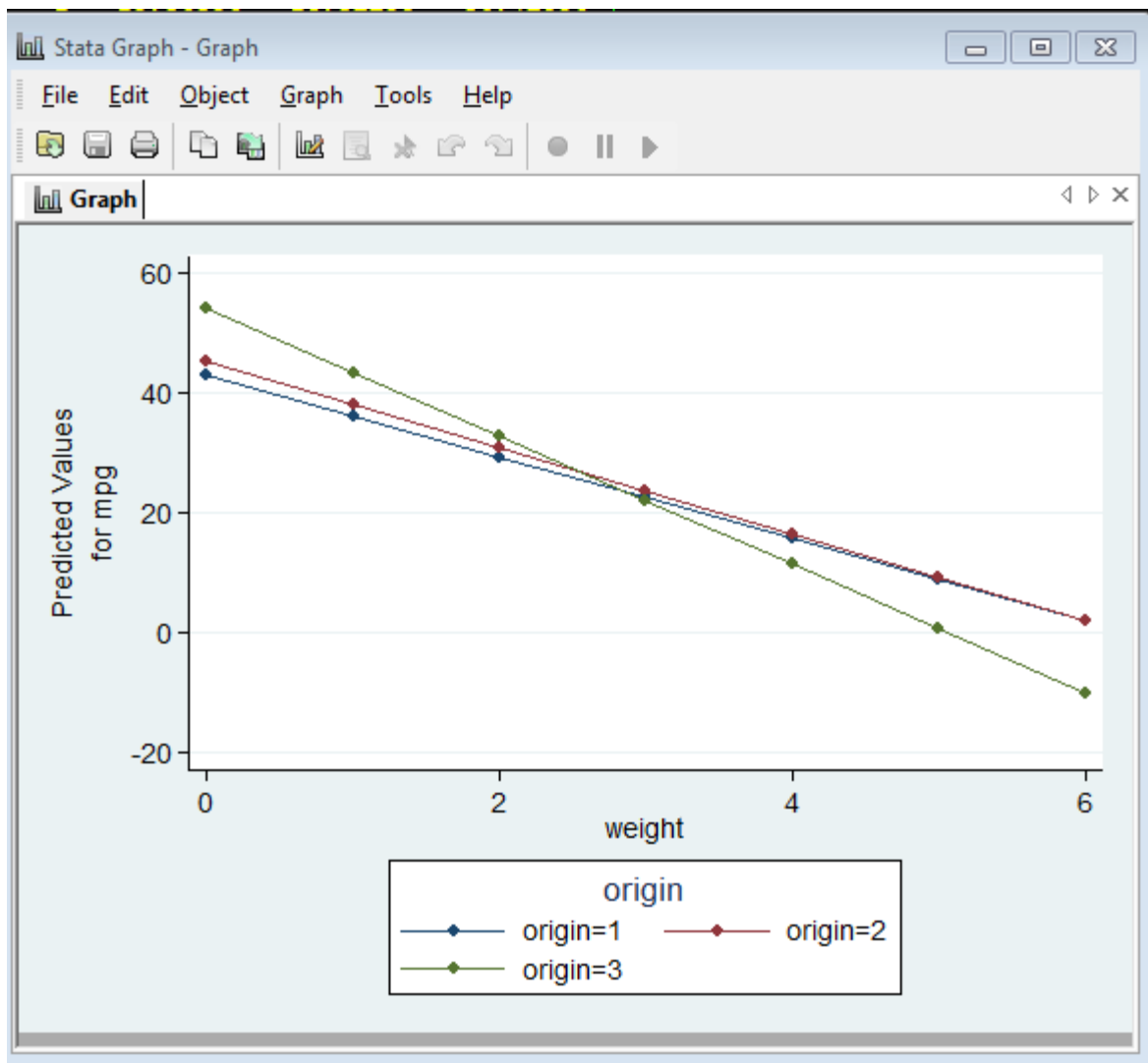
```
-> origin = 2
```

weight	pred_y	lower	upper
0	45.37579	40.30299	50.4486
1	38.16069	35.07357	41.24781
2	30.94558	29.622	32.26917
3	23.73048	22.18192	25.27904
4	16.51537	13.12881	19.90194
5	9.300265	3.918427	14.6821
6	2.085159	-5.323226	9.493545

---

```
-> origin = 3
```

weight	pred_y	lower	upper
--------	--------	-------	-------



**CONGRATULATIONS! You have just graphed the predicted values of a slope dummy interaction term in STATA!**

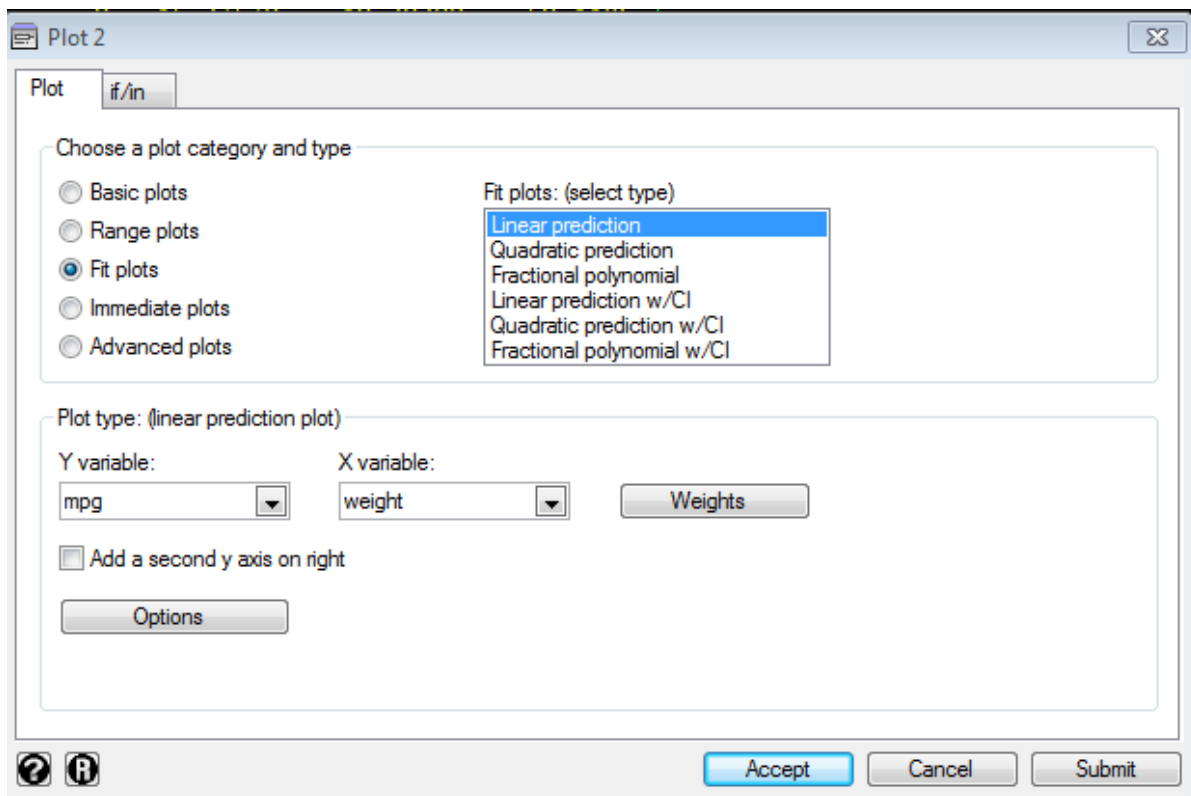
If we compare the predicted values for Asian cars (origin value of 3, the green line in the graph above), to those of American cars (origin value of 1, the blue line), the above graphic re-affirms our conclusions from the interactive model above. While Asian cars obtain higher mpg on average, this is conditional on vehicle weight. Asian vehicles lose more mpg as more weight is added to the vehicle compared to North American cars.

### STATA COMMAND 8.3:

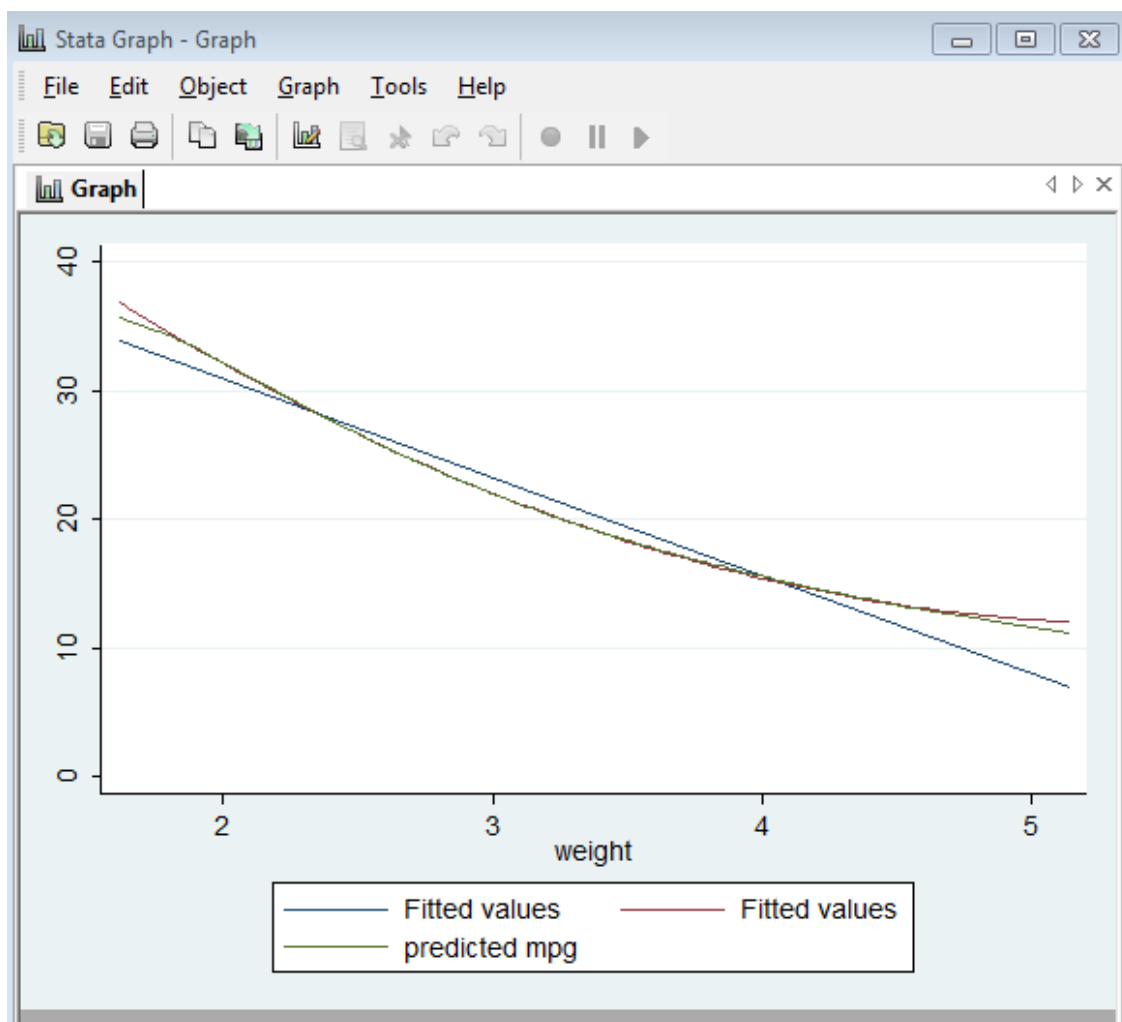
*Code:* “**predxcon var1, xvar(var2) from(min) to(max) inc(interval) graph class(dummy/category)**” where var1 is your dependent variable and var2 is the (continuous) independent variable in your interaction term, “min” is the minimum value (roughly) of var2, “max” is the maximum value (roughly) of var2, “interval” is the increment desired between bottom and top values, and “dummy/category” are the n dummy categories which compose your interaction term.

*Output produced:* Computes predicted values for interaction terms between specified continuous and dummy independent variables, graphs them, and tests the significance of these interactions

Dummy variables and interaction terms are not the only non-linear independent variables that one can use for model specification in OLS; a researcher may also want to test whether the relationship between an independent variable and dependent variable is quadratic or inverse. You may notice from the scatter plot created in the beginning of the lesson that weight and mpg may share a curved rather than a linear relationship. We end the lesson by re-estimating our above models with a quadratic term and an inverse term. Before we estimate beta coefficients for a quadratic and inverse term, however, briefly graph a fitted line for a linear, quadratic, and inverse relationship between vehicle weight and mpg in STATA. Click the “Graphics” tab, followed by “twoway graph (scatter, line, etc.)”, and then click the “Create” button. You should see the following box:



Click on the “Fit Plots” button, and specify that you want to create a linear prediction between your dependent variable (mpg) and independent variable (weight) – we will create them without confidence intervals for now. Click “Accept”, and then repeat the process, creating another fitted line with a quadratic prediction between mpg and weight, and a third with a fractional prediction between mpg and weight. After you have created the three plots, click “Ok”. You should be presented with the following image:



**CONGRATULATIONS! You have just created two graphics that estimate a non-linear relationship between your independent and dependent variable in STATA!**

The graphics which predict a quadratic (in red) and inverse (in green) relationship between mpg and weight almost perfectly overlap, but both diverge slightly from the linear prediction (in blue). We can estimate whether either of these functional forms are significant, by creating non-linear terms of the independent variable and estimating their effect in a linear model.

In the STATA command box, create both a quadratic term for weight (“generate weight2 = weight\*weight”) and an inverse term for weight (“generate inverseweight = 1/weight”). Then run three regression models with mpg as the dependent variable and the following independent variable

combinations: 1) weight, the European and American dummies as the independent variables; 2) weight, weight squared (note, like an interaction term, we must include the linear term of a variable when we include its quadratic form in a regression!), the European and American dummies as the independent variables, and; 3) the inverse of weight, the European and American dummies as the independent variables. You should be presented with the following output (I present it in estimates table form, with asterisks indicating significance):

```
. estimates table modellinear modelquadratic modelinverse, b(%7.3f) star(.1 .05 .01) stats(N r2)
```

Variable	modellin~r	modelqua~c	modelinv~e
weight	-7.023***	-17.018***	
european	-1.140	-0.758	-0.605
american	-2.355***	-1.474**	-1.397**
weight2		1.525***	
inversewei~t			62.503***
_cons	46.051***	60.572***	1.746
N	398	398	398
r2	0.702	0.718	0.710

Legend: \* p<.1; \*\* p<.05; \*\*\* p<.01

**CONGRATULATIONS! You have just conducted regressions with quadratic and inverse independent variables in STATA!**

The linear term (first column), as in our previous regression is significantly negative. In our quadratic model (second column), the linear term is also significantly negative, but the quadratic term is significantly positive, indicating a U-shaped, rather than hump-shaped, relationship between weight and mpg. Recall from lecture that we can calculate the vertex (i.e. the minimum of our parabola), from the following formula:  $-(\beta_{\text{linear term}}) / (2 * \beta_{\text{quadratic term}})$ . In this case, the inflection point for the quadratic model would be  $(-17.018) / (-2 * 1.525) = 5.58$  tons, which exceeds the heaviest vehicle in the sample (the Pontiac Safari, weighing 5.14 tons). If we predicted the relationship between mpg and weight to be convex without an inflection point, we would need to examine the relationship between an inverse, rather than quadratic functional form. The sign on the inverse weight independent variable (column 3), is significantly positive, which also indicates a declining (curvature) relationship between weight and mpg.

You may notice that the beta coefficients on weight for the linear, quadratic and inverse independent variable forms are all significant – the important question then becomes, which functional form is right? This is a question which STATA cannot answer, but rather relies upon theoretical insight. While STATA may produce significant models which explain a large degree of variation in the data, empirical testing alone will not lead you to the correct one. This is why a review of the theoretical literature, or the establishment of your own well-specified theoretical model, is critical to regression analysis. Theory explains the relationships between two variables, and having a thorough knowledge of theoretical relationships between your dependent and independent variable will assist you in selecting which functional form is most appropriate.

---

### Practice Problems:

Practice Problem 1: Conduct a multivariate regression analysis which examines the impact of a vehicle's horsepower on its mpg, while controlling for year the vehicle was made, whether it was made in Europe, and whether it was made in Asia. How can the dummy variables be interpreted, in terms of where the vehicle was made? How do European and Asian vehicles compare to those of North American origin in terms of mpg efficiency? How do European and Asian vehicles compare relative to each other (i.e. is there a significant difference between the two in terms of mpg?)

Practice Problem 2: Create a scatter plot between mpg (on the y-axis) and horsepower (on the x-axis). Does the relationship look perfectly linear?

Practice Problem 3: Create a quadratic term for horsepower and rerun the above three models from Question 4 with the new quadratic term. From the simple model (where the independent variables are only horsepower and its quadratic term), what is the inflection point and what is the shape of the quadratic relationship? Is more data variation explained by a quadratic model rather than a linear model? How does the beta coefficient on the linear and quadratic term compare as year, the European, and Asian manufacturing dummy are added? As year, the European, the Asian manufacturing dummy, and weight are added?

Practice Problem 4: Create an inverse term for horsepower and rerun the three models from Question 4 with the new inverse term. What is the shape of the relationship between the inverse term of horsepower and mpg. Is more data variation explained by a simple inverse model rather than a linear model? How does the beta coefficient on the inverse term compare as year, the European, and Asian manufacturing dummy are added? As year, the European, the Asian manufacturing dummy, and weight are added?

Practice Problem 5: Create two interaction terms, one between the American dummy and horsepower and one between the European dummy and horsepower. Conduct an interactive model between the American and European dummies and horsepower, with mpg as the dependent variable (make sure you pay attention to hierarchy and the baseline category!). Can you determine a significant interaction between vehicle origin and horsepower? If so, interpret this interaction effect on fuel efficiency in words and graphically using the “predxcon” command.

## Lesson 9: Omitted variable biases, irrelevant variables, outliers and influential cases in OLS

---

**Learning Objective 1: Interpreting the (potentially biasing) influence of an omitted variable on the beta coefficient of an independent variable in STATA**

**Learning Objective 2: Conducting a Ramsey RESET test for linear model specification in STATA**

**Learning Objective 3: Interpreting the influence of an irrelevant variable on the standard error of an independent variable's beta coefficient in STATA**

**Learning Objective 4: Identifying outliers and influential cases, and determining their impact on regression output in STATA.**

Proper regression analysis, like any research design, is contingent upon proper model/variable specification. In Week 3, you learned about several assumptions which OLS is based upon that makes it the best unbiased linear estimator (or BLUE). Failure to properly specify your model, however, may lead to the violation of several of these assumptions, causing OLS to produce biased estimates. Studenmund (2011), outlines that proper model specification focuses around three components: 1) choosing the correct independent variables; 2) choosing their correct functional form, and; 3) choosing the correct form of a stochastic error term. We covered functional form in the previous lesson. In the next lesson, we will focus on types of error terms one can use in their model. In this lesson, however, we focus on the first component, emphasizing how excluding relevant independent variables (the omitted variable bias), including irrelevant ones, and including outliers in our model influence OLS's estimates.

When we introduced you to regression analysis, we first did so via univariate models (i.e. models with one independent variable). While this was done with the intention to present a simplified version of regression analysis to you, it violated one crucial assumption of OLS: that the explanatory variables are independent of the error term. This violation exists due to the omitted variable bias: in excluding other independent variables which may also have an influence on  $y$ , we have shifted these omissions into the error term of our model. If there is the faintest level of correlation between the included independent variable ( $X_1$ ) and the excluded independent variable ( $X_2$ ), this ultimately means that a change in  $X_2$  will change both  $X_1$  and  $\varepsilon$ , prompting violation of Classical Assumption III. There are two conditions when an omitted variable will not introduce bias to the beta coefficient of  $X_1$ : 1) the true coefficient of  $X_2$  equals zero, and; 2) the included and omitted variables are uncorrelated. If, however, neither of these conditions are true, the estimated value  $\hat{\beta}_1$  of will be skewed away from its true value.

Beta coefficient bias can be difficult to interpret with the omitted variable bias for two reasons; 1) we are never really sure what the true value of  $\beta_1$  is, because we generally have sample, rather than population data, and; 2) we may not know we are committing an omitted variable bias, because theory may not have discussed that a certain (omitted) variable has a relationship with  $y$  (the black box problem – it's out

there, but we don't know what's in it!). We can, however, demonstrate the effect of an omitted variable bias via univariate vs. multivariate modeling, i.e. assessing how beta coefficients change as more independent variables are added, which will be the primary focus of this lab.

For this lesson, you are presented with a random sub-sample with information on individual attitudes towards immigration from the 2006 European Social Survey. The dataset you are provided with contains the following information (notice, the last three variables are unrelated to the ESS dataset, they will be introduced when we discuss the inclusion of irrelevant variables):

- Immigration\_good: An index composite of a respondent's attitude towards immigration, with large values indicating that immigration is good for society
- Age: The respondent's age in years
- Eduyrs: The number of years of full-time education completed
- Socialtrust: An index composite of trust towards other people in society, with large values corresponding to high levels of trust
- Random1: A uniform, randomly generated variable between 0 and 20
- Random2: A uniform, randomly generated variable with a zero mean, and standard deviation of 1
- Random3: A uniform, randomly generated variable with zero mean, and a standard deviation of 1.654.

We begin our analysis by focusing on the relationship between age and attitudes towards immigration (i.e. do older/younger individuals believe immigration is good for society). Conduct a linear regression, with immigration attitudes as the dependent variable and an individual's age as the independent variable. You should be presented with the following output:

```
. reg immigration_good age
```

Source	SS	df	MS	Number of obs = 1050		
Model	20.6766387	1	20.6766387	F( 1, 1048) = 25.84		
Residual	838.645526	1048	.800234281	Prob > F = 0.0000		
Total	859.322165	1049	.819182235	R-squared = 0.0241		
				Adj R-squared = 0.0231		
				Root MSE = .89456		

immigratio~d	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
age	-.0079476	.0015635	-5.08	0.000	-.0110156	-.0048796
_cons	.434194	.0770396	5.64	0.000	.2830246	.5853634

Note from the above output, that a 1 year increase in age is significantly associated with a 0.008 drop in an individual's immigration-attitude index, suggesting that as people become older their favoritism for immigration declines. The above model, however, is missing other crucial independent variables which could also explain attitudes towards immigration – the R-squared value of 2% would certainly suggest so – meaning that the beta coefficient for age may be biased! For the next regression, add social trust as an independent variable to the above model. You should be presented with the following output:

. reg immigration_good age socialtrust						
Source	SS	df	MS	Number of obs = 1044		
Model	149.309535	2	74.6547677	F( 2, 1041) = 110.72		
Residual	701.896187	1041	.67425186	Prob > F = 0.0000		
Total	851.205722	1043	.816112869	R-squared = 0.1754		
				Adj R-squared = 0.1738		
				Root MSE = .82113		
immigratio~d	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
age	-.0077684	.0014387	-5.40	0.000	-.0105915	-.0049453
socialtrust	.4083129	.0296748	13.76	0.000	.3500836	.4665422
_cons	.349894	.0712912	4.91	0.000	.2100033	.4897848

Notice for the new regression output, that the beta coefficient for age has not significantly changed, while the model's explanation of the variation in the data (our adjusted R-squared) has increased substantially! Does this mean that the omitted variable bias fails to hold? No! Recall from above that there are two cases where the exclusion of one variable will fail introduce bias to another (included) variable's coefficient: 1) if the (now) included variable's true coefficient zero (which there is not much evidence to support in the above regression output), or, 2) if the included and previously omitted independent variable are uncorrelated with each other. To test whether the second case is correct for age and socialtrust, calculate the pair-wise correlation between the two. You should be presented with the following output:

. pwcorr age socialtrust, sig		
	age social~t	
age	1.0000	
socialtrust	-0.0238	1.0000
	0.4245	

The above pairwise correlation coefficients suggest a lack of correlation between age and socialtrust (the correlation coefficient is not only low, but its associated p-value of significance is quite high). This simple exercise demonstrates one important lesson of omitted variables; omitted variables do not always introduce bias. If our omitted variable has no relationship with y, or if it is uncorrelated with our included independent variable, Classical Assumption III will not be violated.

Let's now turn to an example of where omitted variables introduce bias. Check briefly whether there is a relationship between age and the number of completed years of education with the "pwcorr" command (hint: there should be some degree of correlation between the two variables). In your next regression model, include age, social trust and years of completed education as your right-hand side variables. You should be presented with the following output:

. reg immigration_good age socialtrust eduyrs						
Source	SS	df	MS			
Model	178.885639	3	59.6285462	Number of obs = 1034		
Residual	661.372093	1030	.642108829	F( 3, 1030) = 92.86		
				Prob > F = 0.0000		
				R-squared = 0.2129		
				Adj R-squared = 0.2106		
				Root MSE = .80132		
Total	840.257732	1033	.813415036			
immigratio~d	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
age	-.0042823	.0015064	-2.84	0.005	-.0072383	-.0013263
socialtrust	.3723253	.029857	12.47	0.000	.3137378	.4309127
eduyrs	.0438225	.0067362	6.51	0.000	.0306043	.0570407
_cons	-.3220342	.1253665	-2.57	0.010	-.5680372	-.0760313

Notice that unlike our previous regression model, the beta coefficient of age has dropped from -0.0079 to -0.0043, suggesting that with the inclusion of education, the impact of age on attitudes towards immigration becomes mitigated. This is the omitted variable bias in action! Though age continues to remain highly significant, its beta coefficient has nearly halved compared to the univariate equation where education was excluded. Unlike social trust, education violates both omitted variable bias exceptions; its beta coefficient (influence on immigration attitudes) is not equal to zero, and it is correlated with age. We can see that in correcting for this exclusion, we have shifted age's beta coefficient away from its previous estimate towards (in theory) its "truer" value.

A more formal means to test whether a model suffers from an omitted variable bias, which may result from the exclusion of an independent variable or the exclusion of an independent variable's proper functional form, is the Ramsey Regression Equation Specification Test (Ramsey RESET). The Ramsey RESET test uses non-linear combination of the fitted values of a regression (polynomial forms of  $\hat{y}$ ) to determine whether a model is properly specified. It is a post-estimation test (i.e. one which is unique to a defined regression model), which estimates the following regression:

$$y = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 \bar{y}^2 + \beta_5 \bar{y}^3 + \beta_6 \bar{y}^4 + \dots + \beta_{k-2} \bar{y}^k + \varepsilon$$

where  $X_1$ ,  $X_2$ , and  $X_3$  are the independent variables of our model, and  $\bar{y}^2, \dots, \bar{y}^k$  are the k polynomial forms of our fitted values. In order to determine whether a model is properly specified, we would conduct an F-test for whether the entire model was significant. The null hypothesis of a Ramsey RESET F-test is that the model is properly specified (i.e. suffers no omitted variable bias). Hence, if you were provided with a significant F-stat, you would reject the null that the model is properly specified and would be presented with sufficient evidence that it suffers from an omitted variable bias.

We can conduct a Ramsey RESET test in STATA via the "ovtest" post-estimation command. After you conduct a regression model (let's use the one above, where immigration attitudes is a function of an individual's age, years of education and level of social trust), simply type "ovtest" into the STATA command box. You should be presented with the following output:

```
. reg immigration_good eduyrs age socialtrust
```

Source	SS	df	MS
Model	178.885639	3	59.6285462
Residual	661.372093	1030	.642108829
Total	840.257732	1033	.813415036

Number of obs = 1034  
F( 3, 1030) = 92.86  
Prob > F = 0.0000  
R-squared = 0.2129  
Adj R-squared = 0.2106  
Root MSE = .80132

immigratio~d	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
eduyrs	.0438225	.0067362	6.51	0.000	.0306043 .0570407
age	-.0042823	.0015064	-2.84	0.005	-.0072383 -.0013263
socialtrust	.3723253	.029857	12.47	0.000	.3137378 .4309127
_cons	-.3220342	.1253665	-2.57	0.010	-.5680372 -.0760313

```
. ovtest
```

Ramsey RESET test using powers of the fitted values of immigration\_good  
Ho: model has no omitted variables  
F(3, 1027) = 0.39  
Prob > F = 0.7579

**CONGRATULATIONS! You have just conducted a Ramsey RESET test for model specification in STATA!**

In this particular case, we are presented with an insignificant F-statistic, which indicates that we can fail to reject that the model has no omitted variables (and hence does not suffer from an omitted variable bias). If you discover that you have a significant F-statistic, however, one model specification remedy you may want to try first is the inclusion of polynomial (quadratic or inverse) forms of your interval-measured dependent variables. The general intuition behind a Ramsey RESET test is that non-linear combinations of your independent variables may exhibit significance in explaining your dependent variable.

#### STATA COMMAND 9.1:

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable, var2, var3... are your independent variables  
“**ovtest**”

*Output produced:* Conducts a Ramsey RESET test for model specification. If the produced F-statistic is high, we must reject the null hypothesis that there are no omitted variables.

Like the omission of relevant variables, the inclusion of irrelevant variables within your model may distort your regression results. Unlike omitted variables, however, irrelevant variables will not introduce bias to the coefficients of your other independent variables; recall that one of the conditions where an omitted variable will not introduce bias is if its beta coefficient is actually zero, or in other words, if it is

irrelevant. It may, on the other hand, increase their variance (and hence standard error) of some, though not all, of your betas. This means that while the inclusion of such variables will not promote OLS bias (a good thing, as this implies all OLS assumptions hold), it may decrease the absolute magnitude of our t-statistics for selective beta coefficient, threatening their significance, and the adjusted R-squared of the model (a bad thing). To demonstrate the irrelevant variable impact on our variables' significance levels, let's turn back to our most recently specified model, where our independent variables include age, education, and social trust. Let's now add three randomly generated variables to our above regression model (Random1, Random2, and Random3). Note that since these variables were randomly generated, they should have no relationship with attitudes towards immigration, unless by chance. In other words, they should be completely irrelevant in our model specification. After including these three variables within your model, you should be presented with the following output:

```
. reg immigration_good age socialtrust eduyrs Random1 Random2 Random3
```

Source	SS	df	MS	Number of obs = 1034		
Model	179.31929	6	29.8865483	F( 6, 1027) = 46.44		
Residual	660.938443	1027	.643562261	Prob > F = 0.0000		
Total	840.257732	1033	.813415036	R-squared = 0.2134		
				Adj R-squared = 0.2088		
				Root MSE = .80222		

immigratio~d	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
age	-.0042319	.0015097	-2.80	0.005	-.0071944	-.0012694
socialtrust	.372323	.0299153	12.45	0.000	.3136208	.4310251
eduyrs	.0436506	.0067515	6.47	0.000	.0304022	.0568989
Random1	-.002564	.0042753	-0.60	0.549	-.0109533	.0058252
Random2	-.0106931	.0253836	-0.42	0.674	-.0605028	.0391166
Random3	-.005561	.0157343	-0.35	0.724	-.036436	.0253139
_cons	-.2961089	.1321058	-2.24	0.025	-.555337	-.0368808

The beta coefficients on Random1, Random2 and Random3 are insignificant, as expected. The beta coefficients of age, socialtrust and education have not changed. However, by including these three variables, the t-statistic associated with age, education and socialtrust have slightly decreased, although not significantly. You'll also notice that the adjusted R-squared has declined slightly, though not significantly. This is an (mitigated) example of the irrelevant variable effect. You may notice that, given the highly significant predicted effect of age, education and socialtrust, the t-values of these variables still remain high in absolute terms even with the inclusion of Random1, Random2, and Random3. Hence, the inclusion of irrelevant variables has not introduced such severe variance problems for this particular dataset, given the large number of observation (n=1034) and the predicted strength in significance of our variables. For smaller datasets, however, the inclusion of irrelevant controls may mean the difference between a statistically significant relationship and an insignificant one.

For the above exercises, we have focused predominantly on the inclusion of proper independent variables within our regression, rather than the distribution of the data within the 2006 ESS itself. One central assumption of significance testing (Assumption VII) is that the error term is normally distributed. A condition which violates this assumption, however, is the presence of an outlier. An outlier is an observation with a large residual, one with an unusual dependent variable value. It can exist because the

observation itself is an anomaly or because it was measured/recorded erroneously. Outliers violate Assumption VII because they introduce skew to the distribution of our error term (given the outlier's substantial residual value, the location of the residual mean is pulled away from the residual median). While the violation of Assumption VII does not necessarily bias OLS per se, it can introduce uncertainty to our results given that they may be distorted by anomalous observations. For some datasets, notably smaller datasets, the presence of an outlier(s) may appear obvious when the dependent variable is paired with the independent variable in a scatter plot. For large datasets, however, it may be difficult to identify outliers with the naked eye. We can gauge whether an observation is an outlier (i.e. has "too large" a residual) via the construction of studentized residuals.

To calculate the studentized residual for your regression model (note these residuals are unique to the specified model, and the independent variables included), immediately after you run your regression (use the regression above whether your independent variables were age, education and socialtrust), type "predict r, rstudent". This command will present you with another variable – the residuals of each observation for your specified model. We can plot how these residuals are distributed on a histogram, if they were listed in numerical order by typing in "stem r" into the STATA command box. You should be presented with the following output box:

```

. stem r

Stem-and-leaf plot for r (Studentized residuals)

r rounded to nearest multiple of .01
plot in units of .01

-28* | 551
-27* | 41
-26* | 8
-25* | 833
-24* | 775440
-23* | 9887300
-22* | 6420
-21* | 8666111
-20* | 66421
-19* | 98632
-18* | 8532222110
-17* | 997765520
-16* | 865554443200
-15* | 9877666554321110
-14* | 9764322222211
-13* | 76443221000
-12* | 9876665442110000
-11* | 99997765555444443333332210000
-10* | 987766543332111000
-9* | 988554333221000000000
-8* | 9876664333222210
-7* | 8876655443321100
-6* | 999887776666544443222211110000
-5* | 98776665444333222111000
-4* | 99988888777776665554443322211111000
-3* | 9999988877777666655544444333222111110000
-2* | 9999999987766665544444333322211111000000
-1* | 9998888877766666666655554444444333222221110
-0* | 99999877777666665555444444443321
0* | 001111111222222334444445555556666666677777888
1* | 000001111222222334555556666777888889999999
2* | 00000111111222222333334455566667888899999
3* | 011111222222333334444555555666666666667788888999
4* | 0000012222233344455556666666777778889999
5* | 0001112222333445555556666667777788889999
6* | 00111223334445567888888899999
7* | 0111111222222333334455555666666667778899999
8* | 000111122233455566667788888999999
9* | 000111122233333444556777888
10* | 00122233445788
—more—

```

**CONGRATULATIONS! You have just identified outlier cases using STATA!**

The general rule of thumb is that if an observation has a studentized residual whose absolute value is greater than 2, it is considered unusual and may be considered an outlier. Observations with studentized residuals whose absolute value is greater than 2.5 should be subject to more suspicion, those with absolute values greater than 3 indicate something really out of the ordinary, and observations with absolute values greater than 4 are freaks of nature (i.e. these shouldn't happen). We can see from the stem diagram that there are some cases which lie within these categories.

## STATA COMMAND 9.2:

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable, var2, var3 ... are your independent variables  
“**predict r, rstudent**”  
“**stem r**”

*Output produced:* Calculates studentized residuals of your observations for the specified model, and plots them on a stem (histogram) diagram.

*Additional notes:* Outliers can be identified if the absolute value of their studentized residuals exceed 2. To determine which and how many observations these are, typing “**sort r**” in the STATA command box will numerically order the studentized residuals of your observations from lowest to highest

To see how these cases influence our beta coefficients, we will employ a technique called “jack-knifing”, which is based upon analyzing how regression output changes if we exclude (peculiar) cases. There are four jackknife analyses we will perform for the above model: 1) one excluding cases where the absolute value of the residual exceeds 4; 2) one excluding cases where the absolute value of the residual exceeds 3; 3) one excluding cases where the absolute value of the residual exceeds 2.5; and 4) one excluding cases where the absolute value of the residual exceeds 2. In order to re-run the baseline model without these (outlier) cases, type the following codes into your STATA command box:

- “reg immigration\_good age eduyrs socialtrust if abs(r)<4” (this indicates to STATA that you wish to run the regression for all observations whose absolute r value is less than 4)
- “reg immigration\_good age eduyrs socialtrust if abs(r)<3” (this indicates to STATA that you wish to run the regression for all observations whose absolute r value is less than 3)
- “reg immigration\_good age eduyrs socialtrust if abs(r)<2.5” (this indicates to STATA that you wish to run the regression for all observations whose absolute r value is less than 2.5)
- “reg immigration\_good age eduyrs socialtrust if abs(r)<2” (this indicates to STATA that you wish to run the regression for all observations whose absolute r value is less than 2)

For these four commands you should see the following output (I present a condensed estimates table with the original model, plus the four jackknife models above, with p-values below the listed beta coefficient):

× . estimates table Original ExCRLess2 ExCRLess25 ExCRLess3 ExCRLess4, b(%7.4f) p(%7.3f) stats(r2 F N)

variable	Original	ExCRL~2	ExCRL~25	ExCRL~3	ExCRL~4
age	-0.0043 0.005	-0.0043 0.001	-0.0041 0.004	-0.0045 0.003	-0.0045 0.003
socialtrust	0.3723 0.000	0.4128 0.000	0.4003 0.000	0.3886 0.000	0.3826 0.000
eduyrs	0.0438 0.000	0.0460 0.000	0.0456 0.000	0.0433 0.000	0.0439 0.000
_cons	-0.3220 0.010	-0.3108 0.005	-0.3449 0.004	-0.3147 0.011	-0.3192 0.010
r2	0.2129	0.3016	0.2482	0.2255	0.2215
F	92.8636	140.6646	112.0049	99.7455	97.6100
N	1034	981	1022	1032	1033

Legend: b/p

You'll notice a couple things about how our original baseline model changes, as we exclude outlier cases. Starting with general statistics, both the R-squared value and the F-statistic of our model increase as we exclude outliers, meaning the overall model has become more significant and it explains a greater percentage of our data variation. This is expected because outliers are strong deviations from the trend. The second detail you may realize is that for some beta coefficients (socialtrust for example), the inclusion of outliers has a dampening effect; in other words, their inclusion moves their beta coefficients closer to zero. This is one crucial problem about the inclusion of outliers within your sample; they skew the beta coefficients of your independent variables away from their true value.

Outliers with high leverage (i.e. where an observation with a large residual also has an extreme independent variable value) are those whose inclusion actually drives one's results. Outliers with low leverage may not impact the estimated beta coefficient, but will influence the degree of "best-fit" for our least squares line (i.e. our R-squared). Outliers with high leverage, also known as influential cases, on the other hand, may produce beta coefficients that are over/understated or significant when they should not be. If this is the case, we need to identify these observations and determine whether their inclusion drives our estimates. To do so, we rely upon difference-in-fits (dfits). Dfits combine both information on residual size (i.e. whether it's considered an outlier) and leverage (whether an observation is driving the results). The general rule of thumb is if an observation's calculated dfit value is greater than  $2 \cdot \sqrt{\frac{k}{n}}$  where  $k$  is the number of parameters (including your constant term) in your regression model, and  $n$  is your sample size, then it is an overly-influential case.

Like studentized residuals and other post-estimation commands, difference-in-fits are unique to a regression model. To calculate differences-in-fit for your data, first run your properly specified regression model – in our case, with age, eduyrs, and socialtrust as independent variables. Immediately afterwards, type the following into the STATA command box: "predict dfit, dfits". This will create a new variable, the difference-in-fit for each observation. In order to gauge the spread of your difference-in-fits, type in "sort dfit", followed by "stem dfit". You should see the following output:

```

. stem dfit

Stem-and-leaf plot for dfit (Dfits)

dfit rounded to nearest multiple of .001
plot in units of .001

-24* | 5
-23* | 60
-22* | 4
-21* | 2
-20* | 1
-19* |
-18* | 41
-17* | 9720
-16* | 9754
-15* | 98766310
-14* | 9875432
-13* | 8754410
-12* | 987766655531100
-11* | 9876643211
-10* | 998877765544220
-9* | 9887766555221000
-8* | 888777665432222
-7* | 999888776554444333210
-6* | 99988887776666655444433322111110
-5* | 99888777766554443332222100000
-4* | 988888777766665555444333333322111000
-3* | 999888877766666665555444443333322211111000000
-2* | 999888877776665555544444443333322222111111111110000000
-1* | 9999888877777766666665555554444444443333332222221111 ... (77)
0* | 999999988888877777766666666555544444444445555556666666777 ... (79)
1* | 0000000011111111112222222333333444444444455555556666666777 ... (89)
2* | 0000001111122222333344444555555666667777788888888999999
3* | 00000011111222222222233333344444445555556666666777778888 ... (73)
4* | 000000111112222223333344455555666666777777888888999999
5* | 00000011111222222333344455555666666777888888899999
6* | 00000113334455555666777788
7* | 000111222233345566778899
8* | 00111223334556777779999
9* | 001112334456677779
10* | 022346688999
11* | 01135577
12* | 12558
13* | 1245
14* | 1466
—more—

```

**CONGRATULATIONS! You have just identified influential cases using STATA!**

After viewing the entire stem diagram (you can do this by clicking “more”) you may notice that there are a number of cases whose absolute value exceeds our critical dfits value of 0.12439 ( $2 \cdot \sqrt{\frac{4}{1034}}$ ). These observations are influential cases and may possibly skew your results.

### STATA COMMAND 9.3:

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable, var2, var3 ... are your independent variables  
“**predict dfit, dfits**”  
“**stem dfit**”

*Output produced:* Calculates difference-in-fits of your observations for the specified model, and plots them on a stem (histogram) diagram.

*Additional notes:* Influential cases can be identified if the absolute value of their dfit exceeds  $2\sqrt{\frac{k}{n}}$ . To determine which and how many observations these are, typing “sort dfit” in the STATA command box will numerically order the difference-in-fit of your observations from lowest to highest

In order to determine how influential cases impact your results, re-run your regression with a jackknife analysis (i.e. by dropping observations whose dfit value exceeds 0.12439). You can do this by entering the following code in the STATA command box: “reg immigration\_good age eduyrs socialtrust if abs(dfit)<2\*sqrt(4/1034)”. You should see the following output (I present you a condensed estimates table with the original model, plus the dfit jackknife model, with p-values below the listed beta coefficients):

```
. estimates table Original ExcDFIT, b(%7.4f) p(%7.3f) stats(r2 F N)
```

Variable	Original	ExcDFIT
age	-0.0043 0.005	-0.0029 0.037
socialtrust	0.3723 0.000	0.3718 0.000
eduyrs	0.0438 0.000	0.0490 0.000
_cons	-0.3220 0.010	-0.4039 0.000
r2	0.2129	0.2460
F	92.8636	104.3944
N	1034	964

Legend: b/p

You’ll notice that with the exclusion of our influential cases (70 observations, or roughly 7% of our original data sample), one notable result has emerged. Age’s beta coefficient has reduced by almost half, and its significance has decreased. From this result, we can conclude that a small number of influential cases may be overstating the linear relationship for age’s influence on immigration attitudes. If our primary hypothesis revolved around testing the specific relationship between age and attitudes towards

immigration, we should acknowledge these differences in results which stem from the inclusion/exclusion of influential cases.

In summary, even after you have selected your dependent variable, independent variable of interest, and control variables, careful consideration must be given to whether your model is properly specified and whether exceptional observations may skew your results. Omitted variables, be they excluded independent variables or excluded polynomial forms of included independent variables, will automatically introduce bias to your results if they have a significant influence on  $y$  and they hold some level of correlation with your independent variables of interest. Ramsey RESET tests can help you identify whether your model suffers from omitted variable bias, although they will not tell you what the proper model specification remedy is. If you find yourself unable to control for omitted variables, due to the lack of data availability, you should indicate in your results that your coefficients may suffer from some bias. If you discover that including variables which consistently have no significant sign may be irrelevant, the decision to drop them depends on a judgment call. If these variables have been hypothesized to influence  $y$  in the theoretical literature, you should include them, as their lack of significance could arise from measurement error rather than an irrelevant variable bias. If there has been no general consensus of how these variables influence  $y$ , and their inclusion increases the variation of your other independent variables, causing their significance to wane, it may be beneficial to exclude them. Finally, the presence of outliers and/or influential cases and their impact on your results must be considered. The exclusion of outliers, based solely on the size of their residuals and not leverage, in our dataset above failed to significantly alter our regression results. Their presence in smaller datasets (i.e. those with  $N$  less than 100), however, may skew beta coefficients more significantly and/or cause a reduction/increase in beta significance levels. The exclusion of influential cases (i.e. outliers with leverage), on the other hand, did alter the significance of our results above – their impact on smaller datasets may also be equally dramatic. Because outliers and influential cases have the potential to introduce skew to your beta coefficients, proper quantitative research design should discuss whether regression results are sensitive to the inclusion/exclusion of outliers.

---

### Practice Problems:

Practice problems for Lesson 9 require the dataset on UN delegate parking violations and corruption used in Lesson 6.

Practice Problem 1: Re-run the same univariate regression in lab 2, which assesses the influence of corruption on parking violations per UN delegate. Then add (n-1) regional dummies to your model (use the middle east as a baseline category). How can you interpret the beta coefficients of these regional dummies? What happens to the beta coefficient on corruption? What happens to its beta coefficient's significance? What problems could account for these changes between the univariate and multivariate models? Be specific.

Practice Problem 2: Conduct a Ramsey RESET test for the model above. What can you conclude about your model specification?

Practice Problem 3: In the linear regression model where corruption and the regional dummies serve as your independent variables, identify which countries are 1) outlier cases (via studentized residuals), and; 2) influential cases (via difference-in-fits). What happens to the values and significance of your beta coefficient on corruption and the regional dummies and your adjusted R-squared if you exclude countries with studentized residuals with absolute values larger than 2? What happens to the values and significance of your beta coefficients on corruption and regional dummies and your adjusted R-squared if you exclude countries with dfits values less than the critical dfits value? Why might the regional dummy's beta coefficients change so dramatically?

Practice Problem 4: Conduct a univariate regression which assesses the influence of loggdp on parking violations per UN delegate. With your understanding of semilog form, how would you interpret the beta coefficient on loggdp? What happens to the beta coefficient and standard error on loggdp if you add the regional dummies into your regression? What problems could account for these changes between the univariate and multivariate models? Be specific.

Practice Problem 5: In the linear regression model where loggdp and the regional dummies serve as your independent variables, identify which countries are 1.) outlier cases (via studentized residuals), and; 2.) influential cases (via difference-in-fits). (Hint: When you predict  $r$  and  $d_{fit}$ , you will have to specify that you are predicting residuals/dfits for a different model! Do this via typing "predict r2, rstudent" and "predict dfit2, dfits" after your original model). What happens to the values and significance of your beta coefficient on loggdp and the regional dummies and your adjusted R-squared if you exclude the country with the largest (absolute value) studentized residual and difference-in-fit (hint, this will be the same country)?

## Lesson 10: Multicollinearity and Heteroskedasticity

---

**Learning Objective 1: Identifying multi-collinearity within your regression model and how it influences your results and how to correct for it in STATA**

**Learning Objective 2: Constructing residual/fitted values graphics to visually detect heteroskedasticity in STATA**

**Learning Objective 3: Conducting a White test for heteroskedasticity in STATA**

**Learning Objective 4: Correcting for heteroskedasticity in your regression model via the use of robust standard errors**

Our final lesson on ordinary least squares regarded the reconciliation of two OLS assumptions: 1) Assumption V, that the error term has a constant variance (no heteroskedasticity), and 2) Assumption VI, that no explanatory variable is a perfect linear function of any other explanatory variable(s), in other words no perfect multi-collinearity. We will deal with multi-collinearity assumption first, as we have partially alluded to problems of collinearity of independent variables in the previous lessons when we examined the influence of vehicle characteristics on miles per gallon achieved on the highway (data which we also utilize for our lab today).

Perfect multicollinearity occurs when an independent variable is a perfect linear function of another. That is, the variation in one dependent variable can be perfectly explained by changes in another. It is very rare that independent variables are perfectly correlated with one another. We have, however, mentioned one case in previous lessons when perfect multicollinearity happens between two variables: if we include  $n$ , rather than  $n-1$ , categories of dummy variables within our regression model. Say we have a dummy variable which represents two categories, male and female. If we include a male and a female dummy, both of these variables will be perfectly correlated with each other, because if the observation embodies a “male” value for 1, it will automatically embody a “female” value for 0.

The good news is that if you incorrectly include two or more independent variables that are perfectly collinear, STATA will automatically drop one. In other words, it is impossible for you to violate Assumption VI in practice, because STATA will automatically modify your independent variables for you so perfect multicollinearity does not arise. Let’s start by examining the influence of the number of a vehicle’s engine size on its mpg, while controlling for the region in which it was manufactured (Asia, Europe or North America). You include regional dummies to account for the latter, but let’s suppose you forget to include 2 ( $n-1$ ) categories, and rather include all three. Such a specified model will produce the following output in STATA:

. reg mpg engsize european asian america						
Source	SS	df	MS	Number of obs = 398		
Model	15946.9617	3	5315.65389	F( 3, 394) = 252.16		
Residual	8305.61351	394	21.0802373	Prob > F = 0.0000		
Total	24252.5752	397	61.08961	R-squared = 0.6575		
				Adj R-squared = 0.6549		
				Root MSE = 4.5913		
mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
engsize	-.0563113	.0029136	-19.33	0.000	-.0620395	-.0505831
european	(dropped)					
asian	2.196897	.7538807	2.91	0.004	.7147656	3.679029
america	-.1068292	.7379552	-0.14	0.885	-1.557651	1.343993
_cons	34.03741	.6342478	53.67	0.000	32.79047	35.28434

Notice that STATA automatically drops the European dummy due to perfect collinearity (also indicated directly under the regression command in green). In this case, STATA has selected the European dummy as your default category. Models with perfect multi-collinearity cannot be estimated empirically. Most statistical software, STATA included, will automatically correct this problem for you by dropping one variable.

While perfect multicollinearity, outside of dummy variable categories, is rarely an issue for independent variables, imperfect multicollinearity (i.e. when there is a strong linear relationship between two independent variables, but one independent variable does not perfectly determine the other) can be a problem for OLS. Unlike perfect multicollinearity, imperfect multicollinearity will not bias the beta coefficients – hence its presence does not violate any OLS assumptions, preserving its title as the best linear unbiased estimator. However, imperfect multicollinearity can substantially increase the error term of the independent variables that share a strong linear relationship. This could prompt a researcher to incorrectly conclude that they have no significant influence on y, or less significance than if multicollinearity were absent; you may notice this effect slightly in the above output, given American cars' tendency to have larger engines than their European and Asian counter-parts. This is problematic because it increases our chances of committing a Type II error (failing to reject a null hypothesis that is actually false). Due to artificially low t-statistics which imperfect multicollinearity produces, we may fail to reject that  $\beta$  is significantly different from zero when, in the absence of multi-collinearity, we would reject the null. Add the weight of a car as an independent variable to the above regression (remove the European dummy). You should be presented with the following output:

```
. reg mpg engsize asian america weight
```

Source	SS	df	MS	Number of obs = 398		
Model	17098.1263	4	4274.53158	F( 4, 393) = 234.80		
Residual	7154.44885	393	18.2047045	Prob > F = 0.0000		
Total	24252.5752	397	61.08961	R-squared = 0.7050		
				Adj R-squared = 0.7020		
				Root MSE = 4.2667		

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
engsize	-.0131065	.0060705	-2.16	0.031	-.0250412	-.0011719
asian	1.333097	.7089497	1.88	0.061	-.0607115	2.726905
america	-.7118557	.6899865	-1.03	0.303	-2.068382	.6446707
weight	-5.650357	.7105571	-7.95	0.000	-7.047326	-4.253389
_cons	43.01443	1.273503	33.78	0.000	40.51069	45.51816

Notice that with the inclusion of weight, the beta coefficient on engine size drops dramatically; it is still significant, but now at a 95% confidence level rather than over a 99% confidence level. If a 99% confidence level was your decision rule, you may be tempted to conclude that engine size produces no significant impact on its mpg at a 99% level of confidence. If you made this conclusion in the presence of multi-collinearity, however, there is a chance that you would have committed a Type II error. If we had omitted weight, we would be presented with evidence that engine size does have an effect on mpg with over 99% confidence.

There are two ways to detect whether the drop in engine size's significance is the cause of multicollinearity. One is via the construction of simple pair-wise correlation coefficients between independent variables, which we did in Lesson 6. To compare the pairwise correlation coefficients between independent variables, type "pwwcorr engsize asia america weight, sig" into the STATA command window. You should see the following output:

```
. pwwcorr engsize asian america weight, sig
```

	engsize	asian	america	weight
engsize	1.0000			
asian	-0.4318 0.0000	1.0000		
america	0.6557 0.0000	-0.6354 0.0000	1.0000	
weight	0.9325 0.0000	-0.4405 0.0000	0.6010 0.0000	1.0000

You'll notice that there is significant correlation between most of our independent variables, but the correlation coefficient between weight and engine size is especially high. Correlation coefficients with an absolute value higher than 0.75-0.8 are taken as an indicator of severe multicollinearity (we will discuss what to do with these cases below). However, others attribute lower but significant correlation coefficients as cause for concern if they prompt a substantial rise in the standard error of their key independent variables. Such indications of multicollinearity are more subjective, but the same theoretical principle applies – the greater the level of collinearity between two independent variables, the more likely we are to witness a drop in the significance of their beta coefficients.

A second method for detecting whether your model suffers from imperfect multicollinearity is through variance inflation factors (VIF). Variance inflation factors measure the extent to which an independent variable is explained by all other independent variables in the model. We measure an independent variable's ( $X_1$ ) VIF as  $\frac{1}{1-R^2}$  of the model for which  $X_1$  serves as the dependent variable. Please note that the R-squared value is the unadjusted, not the adjusted, R-squared. A “high” VIF (generally greater than 5 although others use lower benchmarks) suggests the presence of severe multi-collinearity. Rather than computing the VIF manually for each independent variable, STATA does this for you via the “vif” post-estimation command. Immediately after you run a regression in STATA, type “vif” into the command box if you suspect the presence of multicollinearity. You should be presented with the following output:

```
. reg mpg engsize asian america weight
```

Source	SS	df	MS	Number of obs = 398		
Model	17098.1263	4	4274.53158	F( 4, 393) = 234.80		
Residual	7154.44885	393	18.2047045	Prob > F = 0.0000		
Total	24252.5752	397	61.08961	R-squared = 0.7050		
				Adj R-squared = 0.7020		
				Root MSE = 4.2667		

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
engsize	-.0131065	.0060705	-2.16	0.031	-.0250412	-.0011719
asian	1.333097	.7089497	1.88	0.061	-.0607115	2.726905
america	-.7118557	.6899865	-1.03	0.303	-2.068382	.6446707
weight	-5.650357	.7105571	-7.95	0.000	-7.047326	-4.253389
_cons	43.01443	1.273503	33.78	0.000	40.51069	45.51816

```
. vif
```

variable	VIF	1/VIF
engsize	8.74	0.114454
weight	7.90	0.126646
america	2.44	0.410204
asian	1.75	0.572029
Mean VIF	5.20	

**CONGRATULATIONS!** You have just calculated variance inflation factors for your independent variables in STATA!

Notice that both engine size and weight possess VIF scores greater than 5 (our critical value) reinforcing the information we received from the pair-wise correlation coefficient that high collinearity exists when both are included in the equation.

#### **STATA COMMAND 10.1:**

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
“**vif**”

*Output produced:* Computes variance inflation factors for your independent variables. If the VIF for an independent variable is greater than 5, there is indication of severe multicollinearity.

If you discover that your model suffers from severe multi-collinearity problems, there are three things you can do. These usually revolve around how multi-collinearity influences the significance of the beta coefficient you are interested in. One option is to do nothing and include both independent variables in your model, even though they are strongly correlated. Researchers are more likely to do nothing if the presence of multi-collinearity does not prompt the decline in significance of their beta coefficient below the 90%/95% confidence level (some researchers may even brag that the significance of their results hold in the presence of multicollinearity!). Given that the inclusion of weight fails to influence the significance of engine size beyond the 95% confidence level, we may be persuaded to include both in our models, even though they are highly correlated with each other. Doing nothing also introduces a second advantage of discouraging bias in our beta coefficients, thus limiting variable exclusions which may produce omitted variable bias. The second option, generally used if multi-collinearity reduces the significance of your independent variable of interest below a 90% level of confidence, is to drop the independent variable that it is highly correlated with it. In this case, dropping the second independent variable preserves the variance/standard error of the first. If you opt for the second option, make sure to clearly specify that the presence of multi-collinearity has severely influenced the significance of your independent variable and that doing so may preserve this significance, but at the expense of an omitted variable bias.

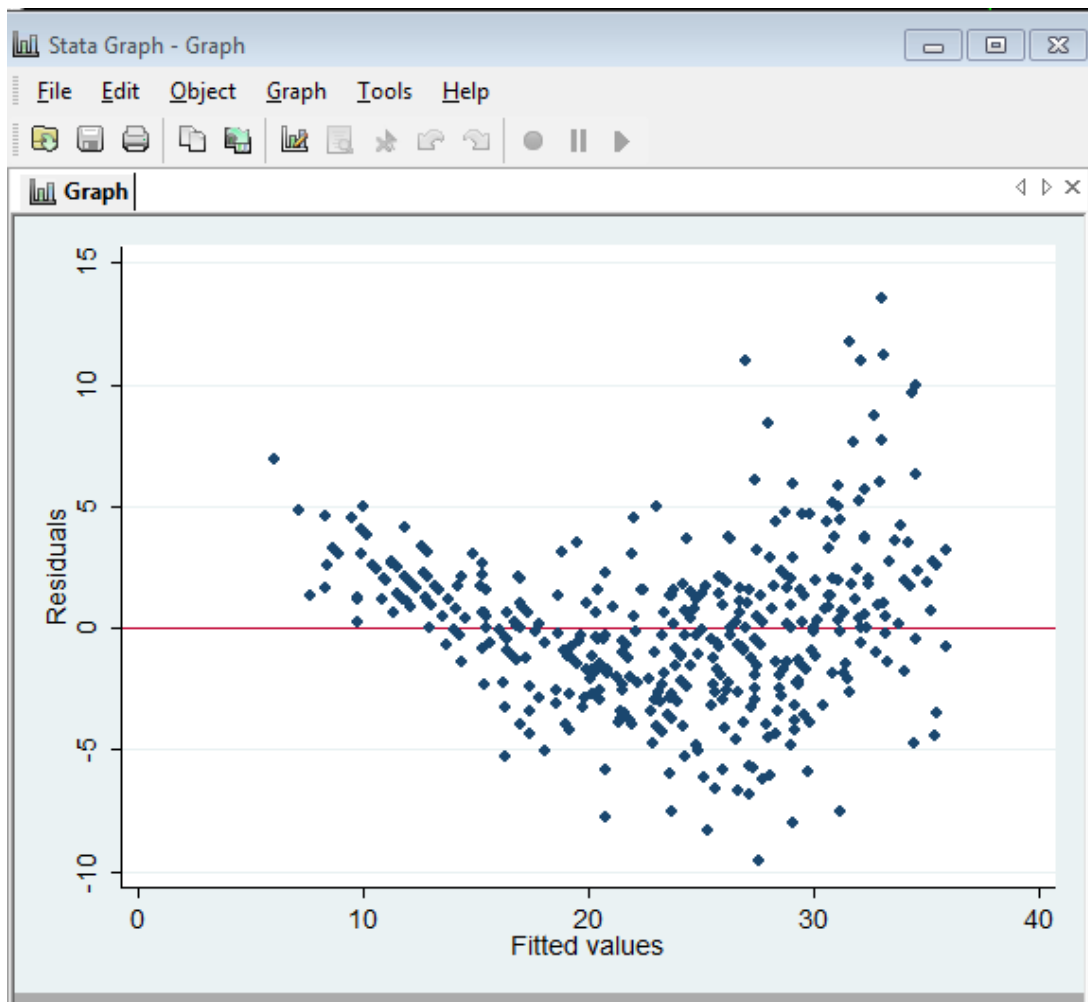
A third alternative, which may not be readily available depending on how you collected your data, is to increase sample size. Recall back to Lesson 2, that as sample size increases, the standard error automatically increases (standard errors =  $\frac{\sigma}{\sqrt{n}}$ ). Hence, this effect should partially cancel out the increase in standard error(s) which results from a multicollinearity problem. Increasing sample size, in some circumstances, can be much easier said than done. If you conducted a survey under contingent conditions (i.e. a survey which required approval from a board of ethics, etc.) you may not be able to conduct a second round of surveys. Also, if you have collected a dataset for a population (i.e. all US states or all countries), it may be impossible to collect additional cross-sectional observations, although you could add a time dimension (in which case you would have to apply a panel regression method, rather than simple OLS). If your sample size is quite constraining, your options for multi-collinearity will revolve around the first two options, rather than the last.

A second violation of OLS, which we have not referred to until this lesson, is the presence of non-constant error variance, or heteroskedasticity, which violates Assumption V. Heteroskedasticity emerges when the residuals associated with some observations are larger/smaller than others. One helpful example to conceptualize this, is calculating national economic indicators for small vs. large countries. If we computed national economic indicators for smaller countries, say Iceland, Liechtenstein and Luxembourg whose populations are less than 600,000, the distribution of the residual associated with our measurements may be much smaller than the distribution of the residual associated with economic indicators for countries with larger populations, like India, China and the United States. In other words, we would expect that the error term distribution for large observations to have a larger variance, while those for smaller observations would have a smaller variance.

In this lesson, you will learn three things about heteroskedasticity; 1) what heteroskedasticity does to your estimates, 2) how to detect it, both graphically and via statistical tests, and 3) how to correct for it. Before we discuss these, it is important to distinguish between pure heteroskedasticity (i.e. where Assumption V is violated in a properly specified model), and impure heteroskedasticity (i.e. where Assumption V is violated in the presence of a specification error, say an omitted variable bias). In the first case scenario, our properly specified model has heteroskedasticity. In the second case scenario, our hypothetical properly specified model may demonstrate homoscedasticity, but due to an omitted variable, the error absorbs a heteroskedasticity component (omitted variables are automatically included in the error term). We will not distinguish between these two types of heteroskedasticity in this lesson, but their differences emphasize the importance on testing for heteroskedasticity on fully specified models! In other words, you should not test for heteroskedasticity on models that do not yet have all of your relevant independent variables!

Heteroskedasticity, like imperfect multi-collinearity, does not introduce bias to our estimated beta coefficients. Like imperfect multicollinearity, heteroskedasticity influences the size of our beta coefficients' standard error. Rather than increasing our standard errors like imperfect multicollinearity does, however, heteroskedasticity, if not properly controlled for, tends to understate standard errors. Hence, uncontrolled heteroskedasticity will produce artificially high t-statistics, increasing the likelihood of committing a Type I error (rejecting a true null hypothesis).

There are two methods that we can use to identify heteroskedasticity within a well specified regression models. Let's select the model from above which excludes engine size but includes year of manufacture; hence we want to test whether heteroskedasticity exists within our model where weight, year of manufacture, and the North America and Asia regional dummies are the independent variables. Type `"reg mpg weight year asian america"` into the STATA command box, and immediately after your regression type `"rvfplot, yline(0)"`. You should be presented with the following graphic:



**CONGRATULATIONS! You have just plotted the residuals versus fitted (predicted) values of your model in STATA!**

We have drawn a horizontal line at 0, because, according to Assumption II of OLS, the mean of our error term should be zero. You may be puzzled as to how one detects heteroskedasticity from the above plot. Recall from above that heteroskedasticity emerges if the distribution of our errors changes as our dependent variable (proxied by our fitted values) becomes larger/smaller. In the graphic above, the distribution of our error increases as the fitted values increase (a funnel shape looks apparent). Hence, heteroskedasticity within our error terms appears to be present. However, scatter plots can often be deceiving. You therefore should not conclude heteroskedasticity exists without a more objective econometric test.

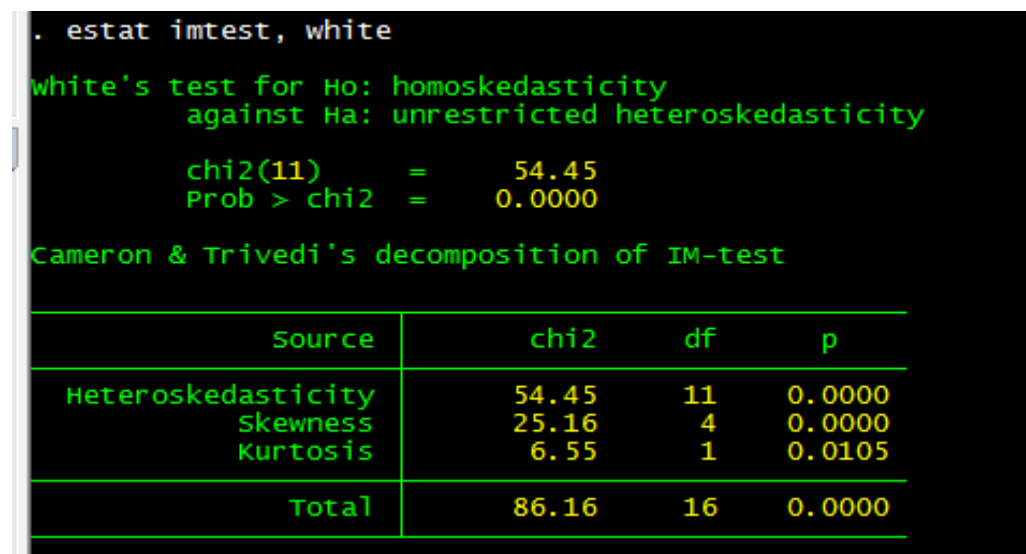
## STATA COMMAND 10.2:

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
“**rvfplot, yline(0)**”

*Output produced:* Calculates a scatter plot of your model’s residuals by its fitted values, along an (assumed) zero mean. If the distribution of residuals increases/decreases as fitted values increase, heteroskedasticity is likely present.

*Caveats:* Does not provide an objective measure/statistic of whether heteroskedasticity is present.

The White test is one of the most widely used tests to detect heteroskedasticity. The test relies upon the post-estimation model (where the squared residuals of an estimated model serve as the dependent variable) and the independent variables (where their squared terms and cross products serve as the independent variables).<sup>11</sup> The White test for heteroskedasticity is a post estimation command; it is unique to the specification of the regression model. After you have run your regression, type the following code into your STATA command box “estat imtest, white”. You should be presented with the following output:



```
. estat imtest, white

white's test for Ho: homoskedasticity
      against Ha: unrestricted heteroskedasticity

      chi2(11)      =      54.45
      Prob > chi2    =      0.0000

Cameron & Trivedi's decomposition of IM-test
```

Source	chi2	df	p
Heteroskedasticity	54.45	11	0.0000
Skewness	25.16	4	0.0000
Kurtosis	6.55	1	0.0105
Total	86.16	16	0.0000

**CONGRATULATIONS! You have just conducted a White test for heteroskedasticity in STATA!**

Notice that the White test automatically presents your null hypothesis that homoscedasticity is present and your alternative hypothesis that heteroskedasticity is present. If you find your test yields a high Chi-squared statistic, you can reject with high confidence that homoscedasticity exists within your model. In other words, there is strong indication that heteroskedasticity is present.

<sup>11</sup> For further clarification, consult Chapter 10 in Studenmund (2011).

### STATA COMMAND 10.3:

*Code:* “**regress var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
“**estat imtest, white**”

*Output produced:* Conducts a White test for the presence of heteroskedasticity. High Chi-squared statistics indicate high likelihood that the assumption of homoscedasticity is violated.

We have found evidence of severe heteroskedasticity within our model, yet the question now becomes how does one correct for non-constant residual variances. The good news is that we can continue to utilize OLS, but we must do so using a different specification. The most popular remedy, and the only one covered in this manual, is the inclusion of heteroskedasticity-corrected, or robust, standard errors. Robust standard errors are calculated to avoid the consequences of heteroskedasticity, and generally (although not always) produce a larger standard errors in comparison to models where homoscedasticity is assumed. Larger standard errors which correct for heteroskedasticity, consequently, reduce the Type II error bias. There are different types of heteroskedasticity-corrected standard errors. We will introduce only three simple robust standard errors today: HC1, HC2, and HC3, which are asymptotically equivalent to HC3 but produce slightly larger (and hence less forgiving) standard errors.<sup>12</sup>

In order to run regressions with the above robust standard errors, simply insert your regression equation into the STATA command box, with a comma after it and the word “robust” (for HC1) or “hc2” (for HC2) or “hc3” (for HC3) after it. Let’s start with the standard HC1 robust standard error. Type the following code into the command box: “reg mpg weight year asian america, robust”. You should see the following output:

. reg mpg weight year asian america, robust						
Linear regression			Number of obs = 398			
			F( 4, 393) = 416.69			
			Prob > F = 0.0000			
			R-squared = 0.8195			
			Root MSE = 3.3379			
mpg	Coef.	Robust Std. Err.	t	P> t	[95% Conf. Interval]	
weight	-5.899649	.2381351	-24.77	0.000	-6.367827	-5.431471
year	.7691001	.0486781	15.80	0.000	.6733982	.8648021
asian	.1143759	.7060538	0.16	0.871	-1.273739	1.502491
america	-2.113521	.6151662	-3.44	0.001	-3.322949	-.9040925
_cons	-16.12073	3.674967	-4.39	0.000	-23.34578	-8.895676

<sup>12</sup> If you are using panel data, you should use more rigorous types of robust standard errors, such as panel corrected, or clustered standard errors.

**CONGRATULATIONS! You have just conducted a regression with robust standard errors in STATA!**

To compare how HC1 standard errors compare to HC2 and HC3 standard errors, run the same regression for two different model specifications. For the first, run the following model: “reg mpg weight year asia america, hc2”. For the second, run the following model: “reg mpg weight year asia america, hc3”. You should be presented with the following results (I provide a summarized estimates table, with t-statistics provided below the beta coefficients):

```
. estimates table original HC1 HC2 HC3, b(%7.4f) t(%7.3f) stats(N F r2)
```

variable	original	HC1	HC2	HC3
weight	-5.8996	-5.8996	-5.8996	-5.8996
year	-22.897	-24.774	-24.724	-24.517
asian	0.7691	0.7691	0.7691	0.7691
america	16.024	15.800	15.773	15.646
_cons	0.1144	0.1144	0.1144	0.1144
	0.206	0.162	0.162	0.160
	-2.1135	-2.1135	-2.1135	-2.1135
	-4.135	-3.436	-3.427	-3.396
	-16.1207	-16.1207	-16.1207	-16.1207
	-4.159	-4.387	-4.379	-4.344
N	398	398	398	398
F	445.9524	416.6927	415.4203	408.9430
r2	0.8195	0.8195	0.8195	0.8195

Legend: b/t

Notice that the R-squared value for each model does not change if you use different types of robust standard errors. In using robust standard errors, we have not added/subtracted variables from our model. Hence, the variation of the dependent variable explained by all independent variables should not change. Also notice that the estimated beta coefficients on our variables have not changed with the inclusion of robust standard errors. This is because, as mentioned previously, heteroskedasticity does not introduce bias to beta coefficients, it only understates their standard errors. Finally, notice that, despite for the weight independent variable, the use of robust standard errors has decreased our t-statistics for the year and the origin dummy variables. Moreover, as we move from HC1 to HC3 standard errors, our t-statistics become smaller, indicating that HC3 standard errors produce larger, more unforgiving standard errors than its predecessors. While MacKinnon and White (1985) demonstrate that HC1, HC2 and HC3 are asymptotically equivalent, and hence one should not be preferred over the other, Long and Ervin (2000) demonstrate via simulations that HC3 should be preferred for datasets with small sample sizes (less than 250 observations).

**STATA COMMAND 10.4:**

*Code:* “***regress var1 var2 var3 ..., robust/hc2/hc3***”, where var1 is your dependent variable, and var2, var3, ... are your independent variables and robust/hc2/hc3 are your HC1/HC2/HC3 standard errors.

*Output Produced:* Conducts an OLS regression with robust standard errors.

---

### **Practice Problems:**

Practice problems for Lesson 10 require the dataset on UN delegate parking violations and corruption used in Lesson 6.

Practice Problem 1: Construct a model examining the influence of corruption and regional dummies on violation rates. What happens to corruption's predicted beta coefficient if you add loggdp as a further independent variable? What problem could account for this change (hint: you should provide two types of evidence which identifies this problem)? What could you do to correct for this problem?

Practice Problem 2: Re-run the regression model which examines the influence of corruption and regional dummies on violation rates. Create a scatter plot with the residuals of this model on the y-axis and the fitted values on the x-axis. Do you suspect the presence of heteroskedasticity? Does a White test support or refute your suspicions?

Practice Problem 3: If heteroskedasticity is present in the above model, what type of standard errors should you include in its specification. Be specific. Re-run the above model with the properly specified standard error. What happens to the significance of corruption? What happens to the significance of the Europe, Asia and Oceania dummy? What happens to the values of their beta coefficients?

Practice Problem 4: Run a regression model which examines the influence of log GDP per capita and regional dummies on LOG violation rates; you will need to create a natural log of violation rates via the "gen" command. Create a scatter plot with the residuals of this model on the y-axis and the fitted values on the x-axis. Do you suspect the presence of heteroskedasticity? Does a White test support or refute your suspicions? Why do you think you obtained the result that you did?

## Lesson 11: Logistic Regression Analysis

---

**Learning Objective 1: Estimating and interpreting the output of logistic regression in STATA**

**Learning Objective 2: Calculating fitted probabilities for the likelihood of y=1 outcomes in STATA**

**Learning Objective 3: Graphing fitted probabilities which demonstrate the relationship between the value of X and the likelihood of a y=1 outcome in STATA**

Logistic (logit) regression is used when the dependent variable is dichotomous (binary). It is used to predict the likelihood of whether a dependent variable is present, for example:

$$y = \begin{cases} 1 & \text{if an individual believes global warming is man-made} \\ 0 & \text{if otherwise} \end{cases}$$

The appropriate probability distribution for a binary variable is the binomial distribution. Unlike OLS or linear probability models, in a logit regression model, we are unable to make claims about how a marginal change in X influences the likelihood of a y=1 outcome. Rather, when we explain a dependent variable within a logistic model, we attempt to explain the probability of a certain outcome occurring.

The use of a linear probability model [ $\text{Pr}(y=1) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k$ ] to estimate a dichotomous dependent variable is problematic for several reasons. One, the homoscedasticity assumption is automatically violated. The error terms [ $\varepsilon = y - \text{Pr}(y = 1)$ ] are no longer normally distributed, and their variance becomes  $\sigma^2 = \text{Pr}(y = 1) * [1 - \text{Pr}(y = 1)]$ , which implies that the variance of the errors depends on  $\text{Pr}(y=1)$ , and hence the values of one's independent variables. The primary problem associated with linear probability models, however, is that expected probabilities produced are unbounded, which means predicted probabilities may be smaller than 0 or greater than 1. Logistic (logit) regression models avoid this problem by forcing the output (predicted values) to be bounded by 0 and 1. Logistic regression estimates the transformation of the predicted probability of y via an iterative maximum likelihood estimator. In logit, we express a predicted y=1 outcome via the following logistic transformation:

$$\text{Pr}(y = 1 | X_1, X_2, \dots, X_k) = e^{(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k)} / (1 + e^{(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k)}) \quad (\text{Eq 11.1})$$

Unlike OLS, logit models possess a cumulative standard logistic distribution (an s-shaped probability curve which indicates the likelihood of a y=1 outcome for all values of  $X_1$ ).<sup>13</sup> The “logit” command in STATA specifies the use of this non-linear technique, but careful attention must be paid to the interpretation of its output. Logit coefficients are log odds units – expressed in equation 11.1 – and cannot be read in the same manner as beta coefficients for OLS. This lesson will instruct you how to conduct a logistic regression analysis, how to interpret your output, and how to construct and graph fitted

---

<sup>13</sup> The primary difference between logit and probit models is that the latter possesses a cumulative standard normal distribution. Other than differences in distributions, both models tend to produce similar outputs.

probabilities. The data used in this lab stems from the 2008 Oregon Energy Policy Survey. You will assess whether there is a relationship between the type of news programs a respondent listens to (we will focus only on general, local news programs and Oregon Public Broadcasting) and whether an individual believes that global warming is a man-made phenomenon. The variables in the dataset include:

- **manmadeglobalwarm:** A binary variable assessing whether the respondent believes that global warming is attributed to man-made causes with 1 for yes, and 0 for no
- **renewinformed:** The respondent's self-reported information level on renewable energy policy in Oregon on a 4 point ordinal scale: 1 for "Not informed"; 2 for "Somewhat informed"; 3 for "Informed"; and 4 for "Very well informed"
- **income:** The respondent's reported annual pre-tax income on a 10 point scale: 1 for "less than \$10,000", 2 for "\$10,000-\$14,999", 3 for "\$15,000-\$24,999", 4 for "\$25,000-\$34,999", 5 for "\$35,000-\$49,999", 6 for "\$50,000-\$74,999", 7 for "\$75,000-\$99,999", 8 for "\$100,000-\$149,999", 9 for "\$150,000-\$199,999" and, 10 for "\$200,000+"
- **education:** The highest level of education completed by the respondent: 1 for Elementary School, 2 for Middle or Junior High School, 3 for High School, 4 for Vocational School, 5 for Some College, 6 for College Graduate and, 7 for Graduate School
- **collegedummy:** A binary variable assuming the value of 1 if the respondent has had some level of college, and 0 if otherwise
- **female:** A binary variable assuming the value of 1 if the respondent is female, and 0 if otherwise
- **oregonpb:** The frequency which the respondent watches Oregon Public Broadcasting (OPB) on a 4 point ordinal scale: 1 for "never", 2 for "infrequently", 3 for "frequently", and 4 for "very frequently"
- **news:** The frequency which the respondent watches local television news and special programs on a 4 point ordinal scale: 1 for "never", 2 for "infrequently", 3 for "frequently", and 4 for "very frequently"
- **ideology:** The respondent's self-reported political view-point on domestic policy issues on a 5 point ordinal scale: 1 for "Very liberal", 2 for "Liberal", 3 for "Moderate", 4 for "Conservative" and, 5 for "Very conservative"
- **age:** The age of the respondent in years
- **liveinORE:** The number of years the respondent has lived in Oregon

Let's begin with a simple logit regression analysis that analyzes the relationship between frequency of engagement with local news and OPB, and whether an individual believes global warming is man-made. To conduct a logit regression, click on the "Statistics" tab at the top of the page. Then click "Binary outcomes", followed by "Logistic regression". You should see the following box:

logit - Logistic regression, reporting coefficients

Model by/if/in Weights SE/Robust Reporting Max options

Dependent variable: manmadeglobalwarm

Independent variables: oregonpb news

☐ Suppress constant term

Options

Offset variable:

☐ Retain perfect predictor variables

OK Cancel Submit

Select manmadeglobalwarming as your dependent variable and oregonpb and news as your primary independent variables. In this case we do not run the risk of imperfect multicollinearity with both variables, as they share a weak, and insignificant, correlation coefficient. Click “Ok” and you should see the following output:

```
. logit manmadeglobalwarm oregonpb news
Iteration 0:  log likelihood = -245.59229
Iteration 1:  log likelihood = -213.0986
Iteration 2:  log likelihood = -212.12837
Iteration 3:  log likelihood = -212.12414

Logistic regression               Number of obs   =       406
                                LR chi2(2)         =       66.94
                                Prob > chi2         =       0.0000
                                Pseudo R2          =       0.1363
Log likelihood = -212.12414

+-----+-----+-----+-----+-----+-----+
| manmadeglo~m |      Coef.   | Std. Err.   |      z    | P>|z|    | [95% Conf. Interval] |
+-----+-----+-----+-----+-----+-----+
| oregonpb     |   .9439486   |   .1269529   |    7.44   |  0.000   |   .6951256   1.192772   |
| news         |  -.2174229   |   .1406289   |   -1.55   |  0.122   |  -.4930505   .0582047   |
| _cons        |  -.9127442   |   .4922264   |   -1.85   |  0.064   |  -1.87749   .0520018   |
+-----+-----+-----+-----+-----+-----+

```

**CONGRATULATIONS! You have just conducted a logistic regression in STATA!**

Notice the presence of iterations before the output table – this is the result of the maximum likelihood estimation which is used to estimate (non-linear) logit coefficients. Similar to OLS, there are three pieces

of information from the above output that are important for statistical inference. The first is the value and corresponding p-value of our log likelihood LR Chi-squared statistic. Similar to our F-statistic in OLS, if we are presented with an insignificant Chi-squared statistic, we cannot reject the null hypothesis that our entire model is insignificant (that all beta coefficients of our dependent variables are equal to zero). In other words, if you are presented with an insignificant Chi-squared statistic, you must either abandon your model or re-specify it until it becomes significant. For the output above, we are presented with a sufficiently large Chi-squared statistic (66.94, p-value=0.0000); we can therefore comfortably reject the null hypothesis that the overall model is insignificant.

#### **STATA COMMAND 11.1:**

*Code:* “**logit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

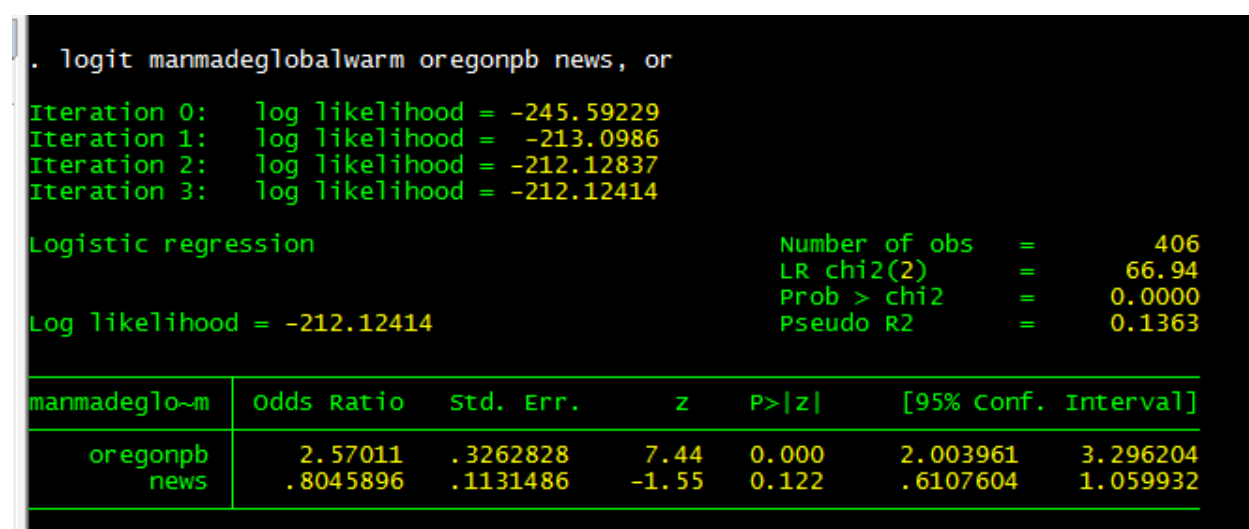
*Output produced:* Generates logistical regression output, where beta coefficients of independent variables are expressed in terms of log-odds.

The second and third pieces of vital information correspond to the sign of the beta coefficient and its significance. Unlike OLS, logit models use z-statistics to indicate significance, rather than t-statistics. The computation and mechanics of both, however, operate very similarly, though their critical values for 90%, 95% and 99% confidence are not identical. Luckily, p-values produced in STATA will help you overcome these problems. Similar to OLS, if the beta coefficient of an independent variable exhibits a significant z-statistic (i.e. one higher than a 90%/95% critical value), then we can claim it has a significant influence on the predicted probability of a y=1 outcome. From our output above, the beta coefficient on the frequency of use of Oregon Public Broadcasting is highly significant, yet the beta coefficient associated with local news programs is insignificant at the 90% level. Hence, we are unable to conclude that the frequency of local news program viewing influences the likelihood that an individual believes global warming is a man-made phenomenon.

Unlike the log likelihood Chi-Squared statistic and z-statistics associated with beta coefficients, which share relatively similar interpretation across logit and OLS, the value of the beta coefficients for logit cannot be read in a similar manner as OLS. OLS beta coefficients indicate the influence of an independent variable’s marginal change (i.e. one unit increase) on a dependent variable. Logit beta coefficients, on the other hand, are expressed in terms of log-odds units, specified by equation 11.1. If the coefficient is significantly negative, higher values of X will produce a lower likelihood of a y=1 outcome. If the coefficient is significantly positive, higher values of X will produce a higher likelihood of a y=1 outcome. Aside from this general assessment, we cannot assess the magnitude of the probability of a y=1 outcome from the above output. There are, however, two methods to better interpret how different values of  $X_1$  influence the likelihood of a y=1 outcome; 1) odds ratios, and 2) fitted probabilities.

Odds ratios indicate how the odds of a y=1 outcome varies between two values of  $X_1$ . If an odds ratio is larger than 1, this indicates that the odds of a y=1 outcome increases as  $X_1$  assumes a higher numerical value. If an odds ratio is smaller than 1, this indicates that the odds of a y=1 outcome decreases as  $X_1$

increases. Interpretations of odds ratios are always relative to a baseline category/value of  $X_1$ , because they are, by construction, the ratio of the odds for a  $y=1$  outcome for one value of  $X_1$  versus another (i.e.  $\frac{Odds(OPB=3)}{Odds(OPB=2)}$  = odds ratio).<sup>14</sup> To present your output in terms of odds ratios, retype the following modification of the above logit model into the STATA command box: “logit manmadeglobalwarm oregonpb news, or”. You should be presented with the following output:



```
. logit manmadeglobalwarm oregonpb news, or
```

```
Iteration 0:  log likelihood = -245.59229
Iteration 1:  log likelihood = -213.0986
Iteration 2:  log likelihood = -212.12837
Iteration 3:  log likelihood = -212.12414
```

```
Logistic regression               Number of obs   =       406
                                LR chi2(2)         =       66.94
                                Prob > chi2        =       0.0000
                                Pseudo R2         =       0.1363
```

```
Log likelihood = -212.12414
```

manmadeglo~m	Odds Ratio	Std. Err.	z	P> z	[95% Conf. Interval]
oregonpb	2.57011	.3262828	7.44	0.000	2.003961 3.296204
news	.8045896	.1131486	-1.55	0.122	.6107604 1.059932

## CONGRATULATIONS! You have just conducted a logistic regression with odds ratios in STATA!

Odds ratios will always be larger than zero. We can interpret the beta coefficient for oregonpb as follows: for an individual who listens to OPB “frequently” (i.e. an oregonpb coding of 3), the odds that he/she believes that global warming is man-made is 2.57 times that of an individual who listens to OPB “infrequently” (i.e. an oregonpb coding of 2). We could alternatively state, “as the frequency of listening to OPB increases by 1 unit, the odds that an individual will believe global warming is man-made increases by 2.57 times”. Supposing that the beta coefficient for news were significant, we would interpret its odds ratio as follows: for an individual who listens to local news “frequently” (i.e. a news coding of 3), the odds that he/she believes that global warming is man-made is 0.805 times that of an individual who listens to local news “infrequently” (i.e. a news coding of 2). We could alternatively state, “as the frequency of listening to local news increases by 1 unit, the odds that an individual will believe global warming is man-made increases by 0.80 times (a 20% reduction)”.

<sup>14</sup> Odds ratios are especially convenient for interpreting dummy independent variables in logit, as the reference category automatically serves as the baseline value.

### STATA COMMAND 11.2:

*Code:* “**logit var1 var2 var3 ..., or**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

*Output produced:* Generates logistical regression output, where beta coefficients of independent variables are expressed in terms of odds ratios.

You may notice that the interpretation of odds ratio requires a benchmark of comparison (i.e. the likelihood of witnessing a  $y=1$  outcome for one value of  $X_1$  over the likelihood of witnessing a  $y=1$  outcome for another value of  $X_1$ ). If you want to gauge the probability of witnessing a  $y=1$  outcome for an absolute value of  $X_1$  without a benchmark value, odds ratios will not be of much help. Rather you must compute predicted probabilities of a  $y=1$  outcome manually, by substituting in the predicted logit model (expressed in log odds, not odds ratios), back into equation 11.1. Say you wanted to determine the probability of a  $y=1$  outcome for an individual who watches Oregon Public Broadcasting “frequently” (value of 3) from the original logit output above. To do so, first solve the log odds output for an OPB value of 3; make sure to include the constant, and other variables even if their beta coefficients are insignificant. For simplicity, we will insert the mean value for the other variables – 2.848 for the local news variable – into the model. You should obtain the following computation:

$$\text{Logit} = -0.9127 + 0.9439(3) - 0.2174(2.848) = 1.2998 \quad (\text{Eq. 11.2})$$

Once you have calculated your logit log odds value for your model, substitute this number for your log odds equations in Equation 11.1:

$$\text{Pr}(y=1 | \text{OPB}=3) = \frac{e^{(1.2998)}}{(1 + e^{(1.2998)})} = 0.785 \quad (\text{Eq. 11.3})$$

This output tells us that an individual who frequently listens to OPB has a 78.5% probability of believing that global warming is man-made, holding all other independent variables at their mean. While this computation was tedious to obtain, the good news is that STATA will compute these probabilities for you via the “prvalue” command. The “prvalue” command is a post-estimation command and is unique to the regression output that precedes it. To calculate the predicted probability of a  $y=1$  outcome for an individual who watches OPB “frequently” in STATA, type in the following code into your STATA command box immediately after your regression output: “prvalue, x(oregonpb=3) rest(mean)”. You should be presented with the following output:

```

. logit manmadeglobalwarm oregonpb news

Iteration 0:  log likelihood = -245.59229
Iteration 1:  log likelihood = -213.0986
Iteration 2:  log likelihood = -212.12837
Iteration 3:  log likelihood = -212.12414

Logistic regression               Number of obs   =       406
                                LR chi2(2)           =       66.94
                                Prob > chi2            =       0.0000
                                Pseudo R2              =       0.1363

Log likelihood = -212.12414

```

manmadeglo~m	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
oregonpb	.9439486	.1269529	7.44	0.000	.6951256 1.192772
news	-.2174229	.1406289	-1.55	0.122	-.4930505 .0582047
_cons	-.9127442	.4922264	-1.85	0.064	-1.87749 .0520018

```

. prvalue, x(oregonpb=3) rest(mean)

logit: Predictions for manmadeglobalwarm

Confidence intervals by delta method

Pr(y=1|x):      0.7859   [ 0.7395, 0.8324]
Pr(y=0|x):      0.2141   [ 0.1676, 0.2605]

    oregonpb      news
x=          3  2.8448276

```

**CONGRATULATIONS!** You have just calculated the predicted probability of a y=1 outcome for specified independent variable value in STATA!

As you can see, STATA provides the identical predicted probability for a “frequent” OPB listener, holding all other variables at their mean, which we computed above. STATA also provides the 95% confidence level of our predicted probability. The “prvalue” command computes fitted probabilities for a variety of specified independent variables. For example, if age, education, and gender were added to our above model, and we wanted to compute the probability that a 42-year-old “frequent” user of OPB with a college degree believed that global warming was man-made, we would simply modify the “prvalue” code after the extended logit model (“logit manmadeglobalwarm oregonpb news age education female”) to the following: “prvalue, x(oregonpb=3 age=42 education=6) rest(mean)”.

### STATA COMMAND 11.3:

*Code:* “**logit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

“**prvalue, x(var2=# var3=# ...) rest(mean)**”

*Output produced:* Calculates the fitted probability of a y=1 outcome, for the specified values for var2 and var3, while holding other independent variables at their mean.

The final logit presentation command you will learn in lab today relates to the graphical presentation of predicted probabilities in STATA. The strengths of the “prvalue” command is that it can compute predicted probabilities for any specified value of your independent variable(s). The weakness of the “prvalue” command, however, is that it can only compute one predicted probability at a time. While the “prvalue” command is convenient when assessing the significant likelihood of a y=1 outcome for a dummy variable, which has limited responses, it can be quite tedious when assessing the likelihood of a y=1 outcome across an entire range of a continuous independent variable. The “prgen” command overcomes this weakness, in plotting the fitted probabilities of a y=1 outcome across all possible values of  $X_1$ . The code, which is user-written, is quite technical, so make sure you have inserted the proper syntax into your STATA command box.

Like the “prvalue” command, “prgen” is a post-estimation command that is unique to regression output. Let’s revert back to our simple logit model, where the probability of a y=1 outcome for the belief that global warming was man-made was solely a function of OPB and local news watching frequency. We will use the “prgen” command to depict fitted probabilities of a y=1 outcome for all possible values of OPB listening frequency. First, re-run the logit model in STATA by typing “logit manmadeglobalwarm oregonpb news” into the STATA command box. Immediately after the output is presented, type the following code into the STATA command box “prgen oregonpb, from(1) to(4) generate(prOPB) rest(mean) gap(1) ci”. The “from() to()” syntax should be the complete range of your independent variable (i.e. the maximum and minimum values). The “gap()” syntax is the desired interval spacing between your maximum and minimum independent variable value, and the “ci” syntax indicates that you want STATA to produce 95% confidence intervals associated with your fitted probabilities.<sup>15</sup> You should see the following output:

Variables			Logistic regression					
Name	Label	Ty	Number of obs = 406					
manmadeglobalwarm	ManMadeGlobal...	by	LR chi2(2) = 66.94					
renewinformed	RenewInformed	by	Prob > chi2 = 0.0000					
income	Income	by	Pseudo R2 = 0.1363					
education	Education	by	Log likelihood = -212.12414					
female	Female	by						
news	News	by						
oregonpb	OregonPB	by						
ideology	Ideology	by						
age	Age	by						
livedinoregon	LivedinOregon	by						
prOPBx	Changing value o...	flk						
prOPBp0	pr(0)	flk						
prOPBp1	pr(1)	flk						
prOPBp0lb	LB pr(0)	flk						
prOPBp1lb	LB pr(1)	flk						
prOPBp0ub	UB pr(0)	flk						
prOPBp1ub	UB pr(1)	flk						

manmadeglo~m	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
oregonpb	.9439486	.1269529	7.44	0.000	.6951256	1.192772
news	-.2174229	.1406289	-1.55	0.122	-.4930505	.0582047
_cons	-.9127442	.4922264	-1.85	0.064	-1.87749	.0520018

```

prgen oregonpb, from(1) to(4) generate(propb) rest(mean) gap(1) ci
logit: Predicted values as oregonpb varies from 1 to 4.
oregonpb      news
x=  2.7339901  2.8448276

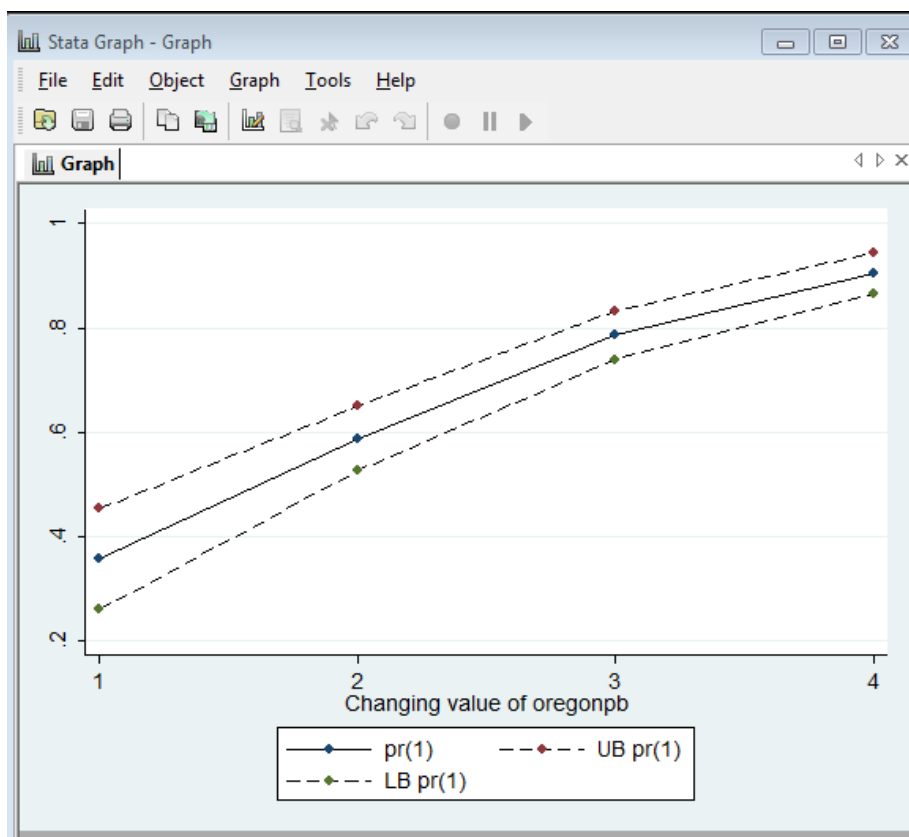
```

Before we proceed to graphing, there are two features to emphasize about the generated output of the “prgen” command. The first feature is the creation of new variables – your fitted probabilities of a y=0 and y=1 outcome. Notice that after you run “prgen” in STATA, 7 new variables are added to your variables box (highlighted in red). The variables include the predicted probability of a y=0 outcome

<sup>15</sup> If you are unsure what the range of your independent variable is, you can determine it via the “summarize” command.

across all values of oregonpb and its associated upper and lower (95% CI) bound (“prOPBp0”, “prOPBp0ub” and “prOPBp0lb”, respectively), as well as the predicted probability of a y=1 outcome across all values of oregonpb and its associated upper and lower (95% CI) bound (“prOPBp1”, “prOPBp1ub” and “prOPBp1lb”, respectively). Because there are only two outcomes for the dependent variable in a logistical model, we only should graph one of these predicted probabilities (either prOPBp0 or prOPBp1). When we cover ordinal logistic regression analysis, however, attention must be paid to fitted probabilities of (n-1) other dependent variable outcomes. The second feature to emphasize about the “prgen” command is that it can only be conducted for the entire range of one independent variable (in our case oregonpb). While this feature seems obvious (as we can only have one x-axis on a two-dimensional fitted probability graphic), it limits our ability to graphically depict fitted probabilities for independent variables that may share a quadratic relationship with the log likelihood of y. In other words, we cannot run prgen on oregonpb and oregonpb<sup>2</sup> simultaneously. We will discuss ways to overcome this problem in applied logistic analysis in the next lesson.

After you have calculated your fitted probability of a y=1 outcome for all possible values of oregonpb, you can graph these values, along with their 95% confidence interval bound via the “graph twoway” command. Type the following syntax (verbatim) into your STATA command box: “graph twoway (connected prOPBp1 prOPBx, clcolor(black) clpat(solid)) (connected prOPBp1ub prOPBx, clcolor(black) clpat(dash)) (connected prOPBp1lb prOPBx, clcolor(black) clpat(dash))”. You should be presented with the following graphic:



**CONGRATULATIONS! You have just graphed the predicted probabilities of a y=1 outcome for all possible values of an independent variable in STATA!**

You'll notice that the above graphic produces the same predicted probability for an individual who frequently (oregonpb value of 3) listens to Oregon Public Broadcasting that we estimated earlier. It also simultaneously produces fitted probabilities for all other values of oregonpb, a feature which is not shared by "prvalue".

#### **STATA COMMAND 11.4:**

*Code:* **"*logit var1 var2 var3 ...*"**, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

**"*prgen var2, from(min) to (max) generate(prFIT) rest(mean) gap(interval) ci*"**, where var2 is the

Independent variable whose range for which you want to construct fitted probabilities of y=1, from(min) to(max) is the range of your independent variable, generate(prFIT) indicates the constructed fitted probabilities, gap(interval) is your interval spacing on your x-axis, and ci indicates you want 95% confidence intervals calculated

**"*graph twoway (connected prFITp1 prFITx, clcolor(black) clpat(solid)) (connected prFITp1ub prFITx, clcolor(black) clpat(dash)) (connected prFITp1lb prFITx, clcolor(black) clpat(dash))*"**

*Output produced:* Graphs the fitted probability of a y=1 outcome, across all possible values of var2, while holding other independent variables at their mean.

*Caveats:* Cannot graphically depict fitted probabilities of independent variables in polynomial forms

Logistic regression analysis enables a researcher to conduct statistical inference with a binary dependent variable. Logit models share several features with OLS, including: the log likelihood Chi-squared statistic can be interpreted in a similar fashion as OLS's F-statistic, z-statistics associated with beta coefficients can be interpreted in similar manners as t-statistics produced in OLS, and logistic models encounter similar problems regarding multi-collinearity, omitted variable bias and (quadratic) functional form. The interpretation of beta coefficients in a logit model, however, cannot be treated in the same manner as OLS. Logistic regression models are logistically transformed. Their beta coefficients, when interpreted properly, indicate how absolute values of independent variables affect the predicted probability of a y=1 outcome, as well as the odds of witnessing a y=1 outcome for two relative values of an independent variable (i.e. odds ratios). They cannot be interpreted as the influence of a marginal change in  $X_1$  on y.

---

### Practice Problems:

Practice Problem 1: Run a logit model which predicts the effect of the four following independent variables on the likelihood that an individual believes global warming is man-made: OPB watching frequency, local news watching frequency, an individual's gender and whether the individual attended some level of college (collegedummy). Before examining the odds ratios, how would a respondent's gender and college attendance influence the probability that he/she believed climate change is man-made? What can you conclude about the model's significance as a whole?

Practice Problem 2: Re-run the above logit model with odds ratios, rather than log-odds beta coefficients. How can you interpret the odds ratio for gender? For whether an individual attended college?

Practice Problem 3: Using the above model, calculate the probability that a woman who has attended some level of college will believe that climate change is man-made. How does this probability compare for a woman who has not attended college? Is the difference significant?

Practice Problem 4: Add income as an independent variable to the logit model you estimated in problem 1. Does the probability that an individual believes that climate change is man-made increase or decrease with income? Is this effect significant? Using the "prgen" command, graphically depict how an individual's probability of believing that climate change is man-made as an individual's income bracket increases (make sure to include 95% confidence intervals!).

## Lesson 12: Model Specification for Logistic Regression Analysis

---

**Learning Objective 1: Understanding how omitted variable bias influences odds ratios and predicted probabilities of logit models in STATA**

**Learning Objective 2: Understanding how to detect for multicollinearity within logit models in STATA**

**Learning Objective 3: Assessing the interaction of independent variables on fitted probabilities within logit models**

**Learning Objective 4: Creating quadratic independent variables in for logit models, and interpreting their (fitted probability) output in STATA**

In the last lesson, you were introduced to the foundations of logistic analysis. Like OLS, logit models are subject to several assumptions, although they are not as severe as OLS's. These assumptions include:

1. Observations are randomly sampled from a population where  $Y_i$  has a binomial distribution with the probability parameter  $\Pr(Y_i=1) = E(Y_i)$
2. The outcome,  $Y_i$ , is discrete and can only embody the values 0 or 1
3. The logit of  $\Pr(Y_i = 1)^{16}$  depends on the logistically transformed values of all explanatory variables through the properly specified linear function  $\bar{\beta}_0 + \bar{\beta}_1 X_1 + \dots + \bar{\beta}_n X_n$
4.  $\text{Logit}[\Pr(D_i = 1)]$  possesses a cumulative standard logit distribution
5. No (perfect) multi-collinearity

Logit shares several assumptions of OLS. Assumption 3 assumes that the logistically transformed model is properly specified and linear, while Assumption 5 assumes no (perfect) multi-collinearity. Hence, omitted variables, quadratic functional form, and collinearity issues between independent variables influence the results of logit models in a similar fashion that they do for those of OLS models. Given Assumptions 1 and 2, however, logit models do not suffer outlier problems like OLS – extreme residuals in  $Y_i$  cannot occur because the outcomes of  $Y_i$  are limited to 0 and 1. Moreover, because  $Y_i$  is dichotomous and is assumed to have a binomial distribution, no further assumptions about error terms ( $\varepsilon = Y_i - \Pr(Y_i = 1)$ ) or their variances are required; this explains why error terms are usually not explicitly included in formulas for logistic regression.<sup>17</sup> In this lab, we will first focus on how omitted variables and imperfect multicollinearity yield similar effects on estimators in logit as they do in OLS,

---

<sup>16</sup> Written formally as  $\Pr(y = 1 | X_1, \dots, X_n) = e^{(\beta_0 + \beta_1 X_1 + \dots + \beta_n X_n)} / (1 + e^{(\beta_0 + \beta_1 X_1 + \dots + \beta_n X_n)})$

<sup>17</sup> This is not to say that logit models never encounter problems with heteroskedasticity. However, given that the errors of  $\Pr(Y_i = 1)$  exhibit a standard logit distribution, unlike probit models where errors exhibit a standard normal distribution, heteroskedasticity is less of a problem for logit models. Nevertheless, it is possible to conduct logit models with robust standard errors in STATA, and researchers often employ robust standard errors to control for heteroskedasticity, even without formally testing for it.

using the 2008 Oregon Energy Survey data from the previous lab. We then focus on the use of quadratic terms and how to account for interactions between independent variables, whose interpretation is dramatically different from that of OLS.

Let's begin with the discussion of omitted variable biases in logit. Despite the fact that residuals can only embody two values in a logit model (either  $1 - \Pr(Y_i)$  or  $0 - \Pr(Y_i)$ ), they continue to absorb omitted variables that are not accounted for within our specified model. If these variables are correlated with other independent variables, they will introduce bias to their beta coefficients, skewing results away from their "true" value. Recall from Lesson 9 that omitted variables do not introduce bias to OLS if they hold an insignificant relationship with the outcome variable. Likewise, they do not introduce bias if they are uncorrelated with included independent variables. Re-run the baseline logit regression we used for the last lab, where the predicted probability of an individual believing that climate change was man-made was solely a function of the frequency they listened to/watched local news and Oregon public broadcasting. You should be presented with the output below:

```
. logit manmadeglobalwarm oregonpb news
```

```
Iteration 0:  log likelihood = -245.59229
Iteration 1:  log likelihood = -213.0986
Iteration 2:  log likelihood = -212.12751
Iteration 3:  log likelihood = -212.12414
Iteration 4:  log likelihood = -212.12414
```

```
Logistic regression
```

```
Log likelihood = -212.12414
```

Number of obs	=	406
LR chi2(2)	=	66.94
Prob > chi2	=	0.0000
Pseudo R2	=	0.1363

manmadeglo~m	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
oregonpb	.9439486	.126956	7.44	0.000	.6951194 1.192778
news	-.2174229	.1406314	-1.55	0.122	-.4930553 .0582095
_cons	-.9127442	.4922337	-1.85	0.064	-1.877505 .0520162

Estimate the pair-wise correlation coefficient of age with oregonpb; you should obtain a low and insignificant correlation coefficient. Now add age as a control to the model. You should obtain the following output:

```
. logit manmadeglobalwarm oregonpb news age
```

```
Iteration 0:  log likelihood = -245.59229
Iteration 1:  log likelihood = -210.43778
Iteration 2:  log likelihood = -209.20589
Iteration 3:  log likelihood = -209.2034
Iteration 4:  log likelihood = -209.2034
```

```
Logistic regression               Number of obs   =       406
                                LR chi2(3)          =       72.78
                                Prob > chi2          =       0.0000
                                Pseudo R2           =       0.1482
```

```
Log likelihood = -209.2034
```

manmadeglo~m	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
oregonpb	.9335802	.1286223	7.26	0.000	.6814851 1.185675
news	-.1269949	.1469041	-0.86	0.387	-.4149217 .1609319
age	-.0177515	.0074249	-2.39	0.017	-.0323041 -.003199
_cons	-.1759076	.5809788	-0.30	0.762	-1.314605 .9627899

Notice that the beta coefficient associated with the log odds of oregonpb has not drastically changed with the inclusion of the previously omitted age variable; you will notice the same result if you estimate odds ratios. This is because age and oregonpb lack a significant correlation between each other. Now include the female dummy as an independent variable, which shares a weak albeit significant relationship with oregonpb (check this via the “pwcorr” command). You should be presented with the following output:

```
. logit manmadeglobalwarm oregonpb news age female
```

```
Iteration 0:  log likelihood = -245.59229
Iteration 1:  log likelihood = -199.13161
Iteration 2:  log likelihood = -197.16548
Iteration 3:  log likelihood = -197.15452
Iteration 4:  log likelihood = -197.15452
```

```
Logistic regression               Number of obs   =       406
                                LR chi2(4)          =       96.88
                                Prob > chi2          =       0.0000
                                Pseudo R2           =       0.1972
```

```
Log likelihood = -197.15452
```

manmadeglo~m	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
oregonpb	.8684411	.1316318	6.60	0.000	.6104475 1.126435
news	-.1148255	.1559031	-0.74	0.461	-.42039 .1907391
age	-.0185655	.0077848	-2.38	0.017	-.0338234 -.0033076
female	1.213596	.2531026	4.79	0.000	.7175237 1.709667
_cons	-.5718466	.6127565	-0.93	0.351	-1.772827 .6291341

While the reduction in oregonpb’s estimated log-odds coefficient has not become significantly different from the baseline model, its value has been reduced given the inclusion of the female dummy. This is the omitted variable bias applied to logit – notice that the empirical result is similar to that witnessed in OLS. If correlations between omitted and included variables are more severe, the biasing effect should be more substantial. However, as we add independent variables that share high(er) correlation with each other, not

only will the omitted variable bias become more apparent, like OLS, but also we encounter a greater likelihood of imperfect multicollinearity.

Logit is subject to a similar (no) perfect multicollinearity assumption as OLS, and hence encounters similar problems associated with imperfect multicollinearity. Like OLS, the inclusion of two independent variables that are highly correlated fails to bias logit's log-odds coefficients, but does increase the standard errors associated with them. To demonstrate this effect, we will add an independent variable assessing how many years the respondent has lived in Oregon (liveinORE). Before you run your logit model, calculate a pair-wise correlation with age and liveinORE. You should be presented with the following output:

```
. pwcorr liveinORE age, sig
```

	livein~E	age
liveinORE	1.0000	
age	0.7919 0.0000	1.0000

The number of years lived in Oregon and a respondent's age are highly correlated with each other, and the correlation coefficient is well within range of the "severe" multicollinearity benchmark of 0.75-0.8 discussed in Lesson 10. Let's see what happens to our logit output above if we add liveinORE as an independent variable. Type "logit manmadeglobalwarm oregonpb news age female liveinORE" into the STATA command box. You should be presented with the following output (I present the output in a condensed estimates table, with p-values below the associated beta coefficients):

```
. estimates table Model1 Model2 Model3, b(%7.4f) p(%7.3f) stats(chi2 N)
```

variable	Model1	Model2	Model3
oregonpb	0.9439 0.000	0.8684 0.000	0.8694 0.000
news	-0.2174 0.122	-0.1148 0.461	-0.1089 0.486
age		-0.0186 0.017	-0.0144 0.235
female		1.2136 0.000	1.1838 0.000
liveinORE			-0.0049 0.651
_cons	-0.9127 0.064	-0.5718 0.351	-0.5759 0.348
chi2	66.9363	96.8755	97.0832
N	406	406	406

legend: b/p

Notice that the p-value associated with age's log-odds beta coefficient becomes insignificant (i.e.  $p > 0.1$ ) when we add the number of years lived in Oregon as a dependent variable. This is the imperfect multicollinearity problem applied to logit! Adding variables that are highly collinear with each other produces similar output to OLS, namely that while the (log odds) beta coefficient remains relatively stable, the standard error associated with this beta coefficient increases substantially.

In OLS, we detected the presence of (severe) multi-collinearity via two means. The first was a pair-wise correlation assessment, identical to the one conducted above. If the absolute value of the correlation coefficient was high (i.e. between 0.75 and 0.8), there was strong evidence for the presence of severe multicollinearity. The second method we used to detect multicollinearity was variance inflation factors (VIFs). STATA does not have a post-estimation command for variance inflation factors after logit like it does for OLS, because these auxiliary regressions rely upon OLS, rather than logit.<sup>18</sup> You can, however, calculate variance inflation factors manually. Recall from Lesson 10 that variance inflation factors are defined as  $\frac{1}{(1-R^2)}$ , where  $R^2$  is the unadjusted  $R^2$  from the auxiliary regression of  $X_1$ .<sup>19</sup> To (manually) calculate the VIF for liveinORE, run a linear regression with liveinORE as the dependent variable, and oregonpb, news, age, and female as the independent variables. You should be presented with the following output:

. reg liveinORE oregonpb news female age						
Source	SS	df	MS	Number of obs = 406		
Model	94220.4637	4	23555.1159	F( 4, 401) = 181.68		
Residual	51989.519	401	129.649673	Prob > F = 0.0000		
Total	146209.983	405	361.012303	R-squared = 0.6444		
				Adj R-squared = 0.6409		
				Root MSE = 11.386		
liveinORE	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
oregonpb	.4582946	.5709109	0.80	0.423	-.6640577	1.580647
news	.7072093	.6825311	1.04	0.301	-.6345769	2.048995
female	-5.720226	1.153063	-4.96	0.000	-7.98703	-3.453422
age	.8661333	.0356089	24.32	0.000	.79613	.9361367
_cons	-1.320759	2.799307	-0.47	0.637	-6.82391	4.182391

Given that the unadjusted R-squared for this auxiliary regression is 0.6444, our VIF for liveinORE would be  $\frac{1}{(1-0.6444)} = 2.812$ .

**CONGRATULATIONS! You have just calculated a manual VIF score for an independent variable!**

<sup>18</sup> Auxiliary regressions which serve as the foundation for VIF scores for dummy (i.e. binary) independent variables can be estimated via the “vif” command in (OLS) linear probability models.

<sup>19</sup> For further information on the computation of variance inflation factors, consult Studenmund, 2011, Chapter 8, Section 3.

You'll notice that the VIF does not exceed Studenmund's (2011) threshold of 5, although it does exceed Allison's (2009) threshold of 2.6. While liveinORE's VIF score is not sufficiently high, according to some econometricians' standards, its significant pair-wise correlation coefficient raises concern for severe multi-collinearity. Your treatment of multi-collinearity in logit should be identical to how you correct for multi-collinearity in OLS; you can either ignore the problem, drop the (repetitive) independent variable, or expand your sample-size.

The above exercises demonstrated empirical similarities between logit and OLS in regards to the fulfillment of their assumptions. Issues relating to bias and accuracy, which result from omitted variables and imperfect multi-collinearity, effect logit and OLS in similar manners. There are specification features, however, where the two regression techniques radically differ. One, mentioned above, is the issue of outliers and influential cases. Given that only two outcome values are possible for logit, 0 and 1, and hence only two possible values for its residuals, logit will never encounter problems with outliers. Consequently, because logit does not encounter problems with outliers, it also will not suffer from skew regarding influential cases outliers with high leverage.

A second major difference between logit and OLS, which is often disregarded in the literature, is the use and interpretation of interaction terms. Unlike OLS, logit by definition is an interactive model. The probability of a  $Y_i=1$  outcome is not only dependent upon a change in  $X_1$ , but is also contingent upon the values of all other independent variables. Moreover, outlined by Norton, Wang, and Ai (2004), product terms in logit cannot be read similarly to OLS because: 1) The sign on the interaction term may switch for different values of  $X_1$ , and 2) the statistical significance of an interaction term cannot be determined from the z-statistic reported in a regression output, as it too may change for different values of  $X_1$ .<sup>20</sup> While this means that product terms are problematic, as their coefficients cannot be read at face value, the good news regarding logit's non-linear design is that we can study interactions between independent variables without having to formally construct an interaction term. We will utilize the "prgen" command to depict possible interactions between an individual's gender and the frequency with which they watch Oregon Public Broadcasting.

Re-run the logit model above, where the probability of an individual believing that global warming is man-made is a (log-odds) function of whether they watch Oregon Public Broadcasting, local news programs, the individual's age, and the individual's gender; do so by typing "logit manmadeglobalwarm oregonpb news age female" into the STATA command box. Using the "prgen" command, let's produce two fitted probability graphics. The first will be the likelihood of a woman (i.e. female=1) believing that climate change is man-made is a function of her watching frequency of OPB; the second will be whether the likelihood of a man (i.e. female=0) believing that climate change is man-made is a function of his watching frequency of OPB. This will require the creation of two fitted probability graphics with "prgen". Let's start with the fitted probability graphic of a woman. Modify the "prgen" command used in the last lesson to the following code: "prgen oregonpb, from(1) to(4) generate(prOPBF) x(female=1) rest(mean) gap(1) ci". Notice, that the only thing that has changed between the prgen syntax here and the prgen used in last lab, is that we have identified a specific value for a second independent variable, in this case a female respondent. Also, we have changed the name of our generated probability (prOPBF, you

---

<sup>20</sup> Berry, DeMeritt, and Esarey (2010) argue that a statistically significant interaction term in logit and probit models is neither necessary nor sufficient for variables to interact meaningfully in influencing  $Pr(Y)$ .

should allocate different names to your fitted probabilities in order to distinguish between them). You should be presented with the following output, and new variables:

```

5/   logit manmadeglobalwarm oregonpb news age female
88   prgen oregonpb, from(1) to(4) ...
89   drop prOPBx- prOPBP1ub
90   logit manmadeglobalwarm oregonpb news age female
91   prgen oregonpb, from(1) to(4) ...

```

Name	Label
news	News
oregonpb	OregonPB
ideology	Ideology
age	Age
livedinoregon	LivedinOregon
liveinORE	
_est_Model1	esample() from e...
_est_Model2	esample() from e...
_est_Model3	esample() from e...
collegedummy	
prOPBFx	Changing value o...
prOPBFp0	pr(0)
prOPBFp1	pr(1)
prOPBFp0lb	LB pr(0)
prOPBFp1lb	LB pr(1)
prOPBFp0ub	UB pr(0)
prOPBFp1ub	UB pr(1)

```

Logistic regression
Iteration 0:   log likelihood = -245.59229
Iteration 1:   log likelihood = -199.37265
Iteration 2:   log likelihood = -197.18589
Iteration 3:   log likelihood = -197.15453
Iteration 4:   log likelihood = -197.15452

Log likelihood = -197.15452
Number of obs   =      406
LR chi2(4)      =     96.88
Prob > chi2     =     0.0000
Pseudo R2       =     0.1972

manmadeglo~m      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
+-----+-----+
oregonpb          .8684411   .1316318     6.60   0.000   .6104476   1.126435
news             -.1148255   .1559031    -0.74   0.461   -.42039   .1907391
age              -.0185655   .0077848    -2.38   0.017   -.0338234 -.0033076
female           1.213596   .2531025     4.79   0.000   .7175237   1.709667
_cons            -.5718466   .6127565    -0.93   0.351   -1.772827 .6291341

prgen oregonpb, from(1) to(4) generate(prOPBF) x(female=1) rest(mean) gap(1) ci
logit: Predicted values as oregonpb varies from 1 to 4.
+-----+-----+
oregonpb      news      age      female
x=  2.7339901  2.8448276  53.339901      1

```

STATA has created 7 new variables, your fitted probabilities of a y=0 and y=1 outcome, as well as their upper and lower bounds (highlighted in red). These probabilities have been constructed for a female outcome of 1 only; hence they only represent the predicted probabilities of a y=1 outcome for women. To calculate fitted probabilities for men, type in the following code into the STATA command box: “prgen oregonpb, from(1) to(4) generate(prOPBM) x(female=0) rest(mean) gap(1) ci”. In this case we have specified fitted probabilities for a male respondent (i.e. female=0). You should be presented with the following output, and new variables:

```

88   prgen oregonpb, from(1) to(4) ...
89   drop prOPBx- prOPBP1ub
90   logit manmadeglobalwarm oregonpb news age female
91   prgen oregonpb, from(1) to(4) ...
92   prgen oregonpb, from(1) to(4) ...

```

Name	Label
_est_Model2	esample() from e...
_est_Model3	esample() from e...
collegedummy	
prOPBFx	Changing value o...
prOPBFp0	pr(0)
prOPBFp1	pr(1)
prOPBFp0lb	LB pr(0)
prOPBFp1lb	LB pr(1)
prOPBFp0ub	UB pr(0)
prOPBFp1ub	UB pr(1)
prOPBMx	Changing value o...
prOPBMp0	pr(0)
prOPBMp1	pr(1)
prOPBMp0lb	LB pr(0)
prOPBMp1lb	LB pr(1)
prOPBMp0ub	UB pr(0)
prOPBMp1ub	UB pr(1)

```

Logistic regression
Iteration 0:   log likelihood = -245.59229
Iteration 1:   log likelihood = -199.37265
Iteration 2:   log likelihood = -197.18589
Iteration 3:   log likelihood = -197.15453
Iteration 4:   log likelihood = -197.15452

Log likelihood = -197.15452
Number of obs   =      406
LR chi2(4)      =     96.88
Prob > chi2     =     0.0000
Pseudo R2       =     0.1972

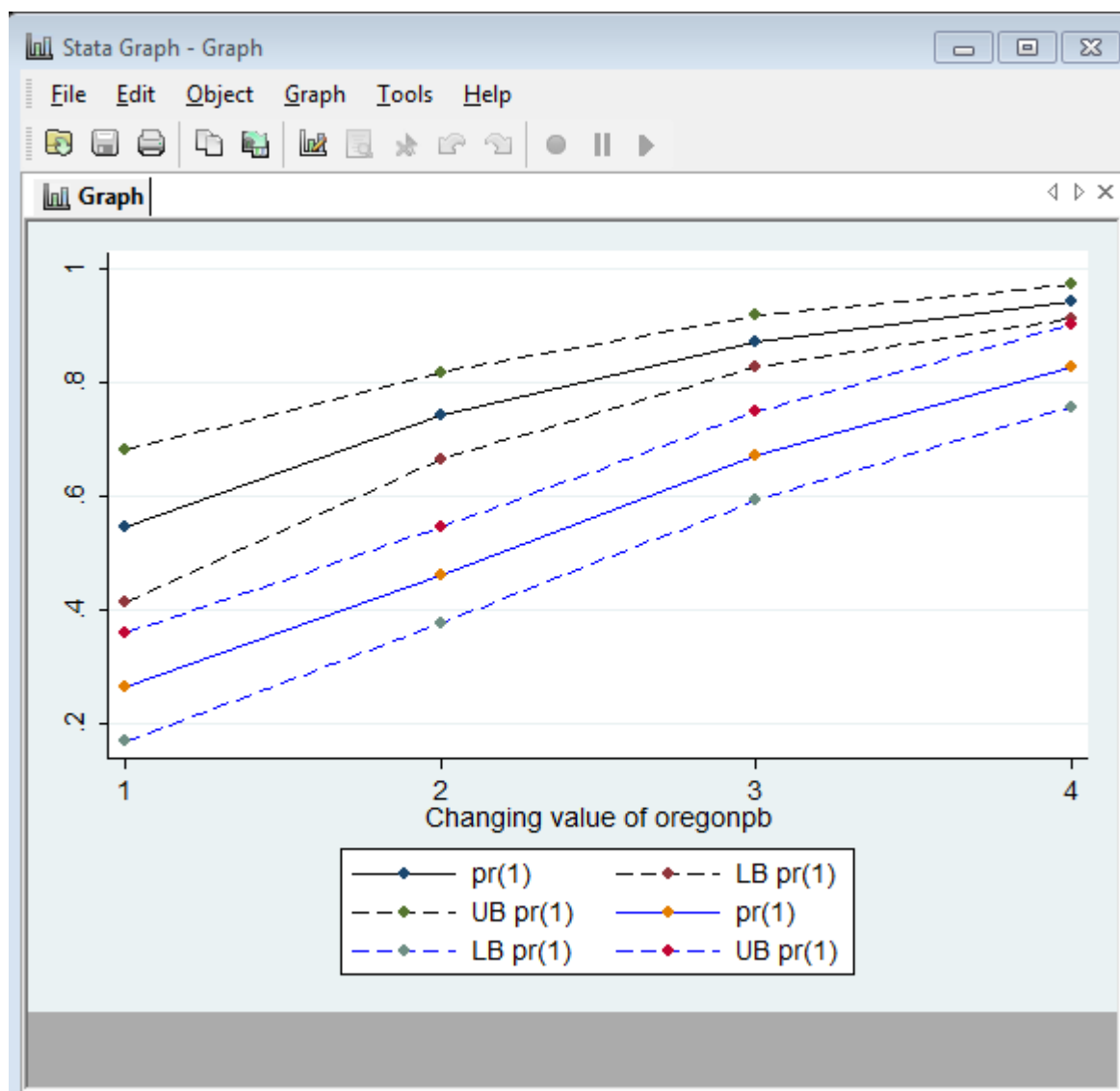
manmadeglo~m      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
+-----+-----+
oregonpb          .8684411   .1316318     6.60   0.000   .6104476   1.126435
news             -.1148255   .1559031    -0.74   0.461   -.42039   .1907391
age              -.0185655   .0077848    -2.38   0.017   -.0338234 -.0033076
female           1.213596   .2531025     4.79   0.000   .7175237   1.709667
_cons            -.5718466   .6127565    -0.93   0.351   -1.772827 .6291341

prgen oregonpb, from(1) to(4) generate(prOPBF) x(female=1) rest(mean) gap(1) ci
logit: Predicted values as oregonpb varies from 1 to 4.
+-----+-----+
oregonpb      news      age      female
x=  2.7339901  2.8448276  53.339901      1

prgen oregonpb, from(1) to(4) generate(prOPBM) x(female=0) rest(mean) gap(1) ci
logit: Predicted values as oregonpb varies from 1 to 4.
+-----+-----+
oregonpb      news      age      female
x=  2.7339901  2.8448276  53.339901      0

```

Similar to above, STATA has created 7 new variables, your fitted probabilities of a  $y=0$  and  $y=1$  outcome, as well as their upper and lower bounds (highlighted in red) for a male respondent. Now that we have constructed our fitted probabilities for the female dummy, we can enlarge our “twoway graph” syntax to depict fitted probabilities by OPB use for both men and women on the same graph. Type the following syntax into the STATA command box - verbatim - into your STATA command box: “graph twoway (connected prOPBFp1 prOPBFx, clcolor(black) clpat(solid)) (connected prOPBFp1lb prOPBFx, clcolor(black) clpat(dash)) (connected prOPBFp1ub prOPBFx, clcolor(black) clpat(dash)) (connected prOPBMp1 prOPBMx, clcolor(blue) clpat(solid)) (connected prOPBMp1lb prOPBMx, clcolor(blue) clpat(dash)) (connected prOPBMp1ub prOPBMx, clcolor(blue) clpat(dash))” Note: you’ll need to distinguish between the two fitted probability graphics by color, so indicate that you want to depict fitted probabilities for women in black and men in blue. You should be presented with the following output:



**CONGRATULATIONS! You have just graphed how the interaction between two independent variables influence the predicted probabilities of a  $y=1$  outcome in STATA!**

There are some important conclusions to take away from the above graphic. Both men and women witness increasing probabilities of a  $y=1$  outcome, as their OPB use increases. However, women exhibit a more pronounced impact of OPB use on the probability of a  $y=1$  outcome than men for lower values of OPB watching frequency (i.e. 1-3) than for higher levels (i.e. 4). In other words, the graphic indicates that there is slight convergence between the two genders, not only in the predicted probability of a  $y=1$  outcome but also its associated confidence intervals, as OPB watching frequency increases. We can interpret this to mean that being a female further magnifies the likelihood that an individual with low OPB listening frequency will believe that global warming is man-made. This interaction is contingent upon two features, however. One is the presence of convergence or divergence between the two graphics. If the probability curves were parallel across all values of OPB, there would be no interactive effect between OPB levels and the female dummy. The second feature which is required to determine a significant interaction effect, is the lack of overlap between (95%) confidence levels across all values of OPB. If these intervals did overlap across the entire range of OPB, we could not claim with confidence that the influence of OPB on the probability of a  $y=1$  outcome significantly differed by varying degrees between men and women across the range of OPB.

#### STATA COMMAND 12.1:

*Code:* “**logit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

“**prgen var2, from(min) to (max) generate(prFITv1) x(var3=value1) rest(mean) gap(interval) ci**”

“**prgen var2, from(min) to (max) generate(prFITv2) x(var3=value2) rest(mean) gap(interval) ci**”

where var3 is the independent variable whose interaction you wish to test with var2, and value1 and value2 are selected values to distinguish between a high/present vs. low/absent levels of var3

“**graph twoway (connected prFITv1p1 prFITv1x, clcolor(black) clpat(solid)) (connected prFITv1p1ub prFITv1x, clcolor(black) clpat(dash)) (connected prFITv1p1lb prFITv1x, clcolor(black) clpat(dash)) (connected prFITv2p1 prFITv2x, clcolor(blue) clpat(solid)) (connected prFITv2p1ub prFITv2x, clcolor(blue) clpat(dash)) (connected prFITv2p1lb prFITv2x, clcolor(blue) clpat(dash))**”

*Output produced:* Graphs how the interaction between two independent variables influences the predicted probabilities of a  $y=1$  outcome

Though logit models are interactive by nature, and hence enable us to examine the interaction between two independent variables without introducing a product term, their construction fails to account for non-linear forms of independent variables. Assumption 3 of logit supposes that the log-transformed model is of linear functional form. If the distribution of probabilities is quadratic across values of  $\mathbf{X}_1$ , a traditional logit model will be unable to account for this. We can correct for the log-transformed linear functional form assumption by introducing a quadratic term into our logit model, as we did in OLS. The influence of a quadratic term, particularly on predicted  $y=1$  outcomes, should be interpreted cautiously given

problems which arise from introducing product terms into logit. The shape of the curve indicated by the quadratic term's sign and its inflection point can be interpreted in a similar manner as OLS. However, you should double check that the quadratic form assumes its significant U/hump-shape across all values of  $X_1$  via the “prvalue” command.

Let's end this lesson with the examination of a possible quadratic relationship between age and the probability that an individual believes climate change is man-made. Create a quadratic term of age via the “gen” command.<sup>21</sup> Then run the following logit model above which also includes the quadratic term of age. You should be presented with the following output:

```
. logit manmadeglobalwarm oregonpb news age age2 female
```

Iteration 0: log likelihood = -245.59229  
Iteration 1: log likelihood = -198.17481  
Iteration 2: log likelihood = -195.68005  
Iteration 3: log likelihood = -195.63142  
Iteration 4: log likelihood = -195.63139

Logistic regression

Log likelihood = -195.63139

				Number of obs	=	406
				LR chi2(5)	=	99.92
				Prob > chi2	=	0.0000
				Pseudo R2	=	0.2034

manmadeglo~m	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
oregonpb	.8925557	.1327496	6.72	0.000	.6323712 1.15274
news	-.1155373	.1563602	-0.74	0.460	-.4219976 .190923
age	-.0955995	.0459749	-2.08	0.038	-.1857088 -.0054903
age2	.0007144	.0004178	1.71	0.087	-.0001044 .0015332
female	1.166716	.2548722	4.58	0.000	.6671758 1.666256
_cons	1.277614	1.258676	1.02	0.310	-1.189346 3.744575

Both age's linear and quadratic terms are significant, and the sign on the quadratic term is positive, indicating a U-shaped relationship between age and the predicted probability of an individual believing that global warming is man-made. Both the linear and quadratic term are significant; we can therefore calculate the inflection point [recall from Lesson 8 that this is  $-\beta_{\text{linear}}/(2 * \beta_{\text{quadratic}})$ ]. Given the output above, the minimum of our U should lie at 66.91  $[ -(-0.0955995)/(2*0.0007144) ]$ . We can now use our “prvalue” command to approximate whether this significant U-shape persists for all possible values of age.<sup>22</sup>

When assessing whether a quadratic relationship persists between  $X_1$  and the probability of a  $y=1$  outcome across all values of  $X_1$ , it is helpful to use the “prvalue” command to assess how fitted probabilities change relative to the inflection point. You should calculate at least five fitted probabilities of  $X_1$ 's linear and quadratic term: 1) the probability at the minimum of  $X_1$ ; 2) the probability for a value

<sup>21</sup> To create a quadratic term of a variable, var, type “gen varsq = var\*var” into the STATA command box.

<sup>22</sup> Though the use of prgen would be more ideal, given that it plots the fitted probability across all possible values of an independent variable, we are limited in its application to quadratic relationships, because it cannot plot fitted probabilities for more than one term.

of  $X_1$  between its minimum value and the inflection point; 3) the probability at the inflection point; 4) the probability for a value of  $X_1$  between its the inflection point and its maximum value; and 5) the probability at the maximum of  $X_1$ . If the calculated probabilities conform to a U/hump-shape, and more importantly the confidence intervals associated with these probabilities do not expand for specific segments of  $X_1$ , then you can be confident that a quadratic relationship is significantly consistent across all values of  $X_1$ . Given the minimum and maximum values of age (18 and 94), let's calculate the fitted probability of a  $y=1$  outcome, for age's linear and quadratic term for the following values; 18, 30, 66.91, 80 and 94.

Similar to the previous lab, we will utilize the "prvalue" command to calculate the fitted probability of a  $y=1$  outcome for a specified value of  $X_1$ . This time, however, you must not only include the linear value, of age, but also the quadratic value of your independent variable. Type the following five post-estimation commands, immediately after your (quadratic) logit model, into the STATA command box:

```
"prvalue, x(age=18 age2=324) rest(mean)"
"prvalue, x(age=30 age2=900) rest(mean)"
"prvalue, x(age=66.91 age2=4476.95) rest(mean)"
"prvalue, x(age=80 age2=6400) rest(mean)"
"prvalue, x(age=94 age2=8836) rest(mean)"
```

You should be presented with the following predicted probabilities:

```
. prvalue, x(age=18 age2=324) rest(mean)
logit: Predictions for manmadeglobalwarm
Confidence intervals by delta method

      Pr(y=1|x):      0.9258   95% Conf. Interval
      Pr(y=0|x):      0.0742   [-0.0007, 1.0007]
      Pr(y=0|x):      0.0742   [-0.0007, 0.1492]

      oregonpb      news      age      age2      female
x= 2.7339901 2.8448276      18      324 .53448276

. prvalue, x(age=30 age2=900) rest(mean)
logit: Predictions for manmadeglobalwarm
Confidence intervals by delta method

      Pr(y=1|x):      0.8566   95% Conf. Interval
      Pr(y=0|x):      0.1434   [ 0.7875, 0.9258]
      Pr(y=0|x):      0.1434   [ 0.0742, 0.2125]

      oregonpb      news      age      age2      female
x= 2.7339901 2.8448276      30      900 .53448276

. prvalue, x(age=66.91 age2=4476.95) rest(mean)
logit: Predictions for manmadeglobalwarm
Confidence intervals by delta method

      Pr(y=1|x):      0.6931   95% Conf. Interval
      Pr(y=0|x):      0.3069   [ 0.6246, 0.7616]
      Pr(y=0|x):      0.3069   [ 0.2384, 0.3754]

      oregonpb      news      age      age2      female
x= 2.7339901 2.8448276    66.91    4476.95 .53448276

. prvalue, x(age=80 age2=6400) rest(mean)
logit: Predictions for manmadeglobalwarm
Confidence intervals by delta method

      Pr(y=1|x):      0.7185   95% Conf. Interval
      Pr(y=0|x):      0.2815   [ 0.6017, 0.8353]
      Pr(y=0|x):      0.2815   [ 0.1647, 0.3983]

      oregonpb      news      age      age2      female
x= 2.7339901 2.8448276      80      6400 .53448276

. prvalue, x(age=94 age2=8836) rest(mean)
logit: Predictions for manmadeglobalwarm
Confidence intervals by delta method

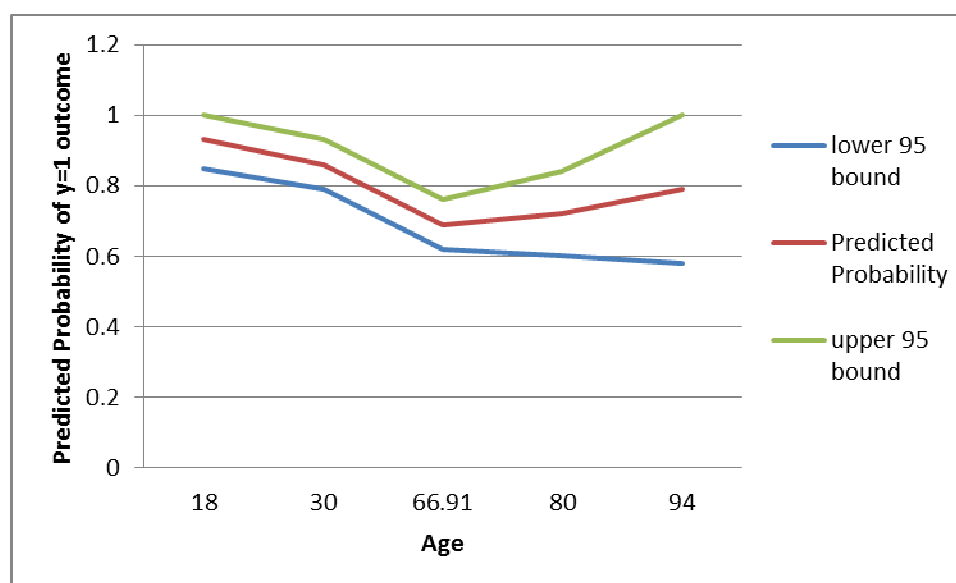
      Pr(y=1|x):      0.7923   95% Conf. Interval
      Pr(y=0|x):      0.2077   [ 0.5816, 1.0030]
      Pr(y=0|x):      0.2077   [-0.0030, 0.4184]

      oregonpb      news      age      age2      female
x= 2.7339901 2.8448276      94      8836 .53448276
```

**CONGRATULATIONS! You have just assessed a quadratic relationship between an independent variable and the probability of a y=1 outcome in STATA!**

Notice that the predicted probabilities of a y=1 outcome conform to the quadratic relationship. However, the 95% confidence intervals of the relationship between the quadratic form of age and the predicted probability of a y=1 outcome expand for higher values of age; these bounds are presented in Figure 12.1 below. Given that the confidence intervals expand at larger values of age, and given that the lower (blue) bound of our predicted probability continues to conform to a linear relationship, we must conclude that the quadratic relationship between age and a y=1 outcome is not significant across all values of age. Consequently, we should drop the quadratic term from our logit model. This exercise further emphasizes the false sense of security a researcher may obtain from logit results, particularly those relating to quadratic and interaction terms; significant coefficients on both the linear and quadratic term do not guarantee that a quadratic relationship will hold across all values of X. Confidence intervals for quadratic terms are more likely to expand at extreme values of  $X_1$ . This enhances the possibility of an insignificant quadratic effect across  $X_1$ 's entire range, in the following circumstances: 1) your quadratic term displays “low” significance (i.e. it is significant at a 90 or 95% confidence level rather than a 99% confidence level); 2) the inflection point of the parabola is near the min/max of  $X_1$ ; or 3) there is little variation in your data near the min or max region due to the sparse number observations present at these extreme  $X_1$  values (this would be a classic heteroskedasticity problem, which could be corrected with the use of robust standard errors).

Figure 12.1: Fitted probabilities and upper/lower bounds for the likelihood of global warming



### STATA COMMAND 12.2:

*Code:* “**logit var1 var2 var2<sup>2</sup> ...**”, where var1 is your dependent variable, and var2, var2<sup>2</sup>, ... are your independent variables.

“**prvalue, x(var2=# var2<sup>2</sup>=#<sup>2</sup>, ...) rest(mean)**”

*Output produced:* Calculates the fitted probability of a y=1 outcome, for the specified values for var2’s linear and quadratic term, while holding other independent variables at their mean. Multiple probability calculation, and assessment of their upper and lower bounds enable the determination of a (significant) quadratic relationship between var2 and Pr(Y=1).

To conclude, logit shares several assumptions with OLS – correct model specification, no perfect multicollinearity – which yields similar outcomes in how the violation of these assumption influence model results. However, given that logit is a logistically transformed, and hence interactive, model by design, several properties associated OLS (i.e. the presence outliers and the marginal interpretation of interaction terms, and the assessment of quadratic terms) do not apply similarly to logit analysis. Careful consideration must be given to logit’s unique construction and how the log odds transformation influences result interpretation and model design. Failure to do so can yield incorrect conclusions regarding hypothesis testing.

---

### Practice Problems:

Practice Problem 1: Run a logit model which predicts the effect of the five following independent variables on the likelihood that an individual believes global warming is man-made: OPB watching frequency, local news watching frequency, an individual's gender, whether the individual attended some level of college (collegedummy), and income bracket. Can you detect a significant interaction between income bracket and college status on the probability that an individual believes global warming is man-made? (Hint: You will need to display this graphically via the "prgen" command).

Practice Problem 2: Add education as an independent variable to the logistic regression above. What happens to the beta coefficient on the college dummy variable? What phenomenon do you suppose is causing this change? Provide two types of evidence which would lead you to this prognosis.

Practice Problem 3: Run a logit model estimating the probability that an individual will believe that global warming is man-made is a (log odds) function of the following independent variables: watching frequency of OPB, watching frequency of local news, whether the individual is a female, whether the individual has some level of college education (collegedummy), age, and a linear and quadratic term of an individual's (domestic) political ideology (assume this is a discrete scalar variable where lower values indicate the individual is more liberal, and higher values indicate the individual is more conservative). What happens to the beta coefficient for OPB watching frequency, relative to a logit model where the probability of man-made global warming beliefs is solely a function of OPB and local news watching frequency? What statistical phenomenon can account for this difference in beta coefficients? Is the difference in OPB's beta coefficient significant between the two models?

Practice Problem 4: Using the output from practice problem 3, what is the predicted relationship between an individual's political ideology and the probability that he/she believes that global warming is man-made? What is the inflection point? Is the (quadratic) relationship significant across all values of political ideology (Hint: you should compare the prvalue output for all values of ideology as well as its inflection point value)? If the relationship is not significant, would you describe it as linear? Present an (Excel) graphic of the predicted probabilities and their upper and lower bounds to support your conclusions.

## Lesson 13: Ordinal Logistic Regression Analysis

---

**Learning Objective 1: Estimating and interpreting the output of ordinal logistic regression in STATA**

**Learning Objective 2: Calculating and graphing fitted probabilities for the likelihood of ordinal outcomes in STATA**

**Learning Objective 3: Assessing the transferability of significance in continuous independent variables in ordinal analysis**

**Learning Objective 4: Testing the proportional odds/parallel regression assumption for ordinal analysis via the Brant test and what estimation alternatives one can use if the assumption is violated**

For the analysis of dichotomous dependent variables, we used logistic regression, a maximum likelihood estimator technique. Logit could be used for the analysis of dichotomous (dummy) variables that are nominally measured (i.e. have no numerical meaning and cannot be ranked) or ordinally measured (i.e. have no numerical meaning but can be ranked, such as a “high” or “low” outcome). While logit regression is helpful in the analysis of qualitative data it suffers one major caveat; only two outcome categories are allowed for the dependent variable. Multinomial logit and ordinal logit regression analysis overcome these caveats. The former is a technique used to perform statistical analysis on a nominally-measured dependent variable with more than two categories and is discussed in the next lesson. The coding of these categories have no numerical meaning and they cannot be ranked. The latter, which is the subject of this lesson, is a technique used to perform statistical analysis on an ordinal-measured variable with more than two categories. The coding of these categories have no numerical meaning but they can be sequentially ranked (i.e. Likert scales, response outcomes of “low”, “medium”, and “high”, etc.). Given that Likert scales are frequently used to assess attitudes, beliefs and preferences, ordinal logit is a popular method among social scientists who examine these social phenomena via surveys. Assuming we had an ordinal variable with three categories, we would code the y outcome as follows:

$$y = \begin{cases} 1 & \text{if low} \\ 2 & \text{if medium} \\ 3 & \text{if high} \end{cases}$$

Ordinal logit stems from similar (empirical) underpinnings as logit. Like logit, ordinal models are logistically transformed from a baseline linear model. However, the functional form of ordinal logit is not identical to logit because the probability of an observed outcome for a set of independent variables ( $X_1, \dots, X_k$ ) depends on the logistically transformed model between a pair of cut-off points for y (the values when we transition from one ordinal category to another). For logit, we did not have to worry about cutoff points. Because there were only two outcomes, 0 and 1, the (implicit) assumption was that the value that divided these outcomes was 0. In other words, if we did not realize an outcome of 0, we could automatically assume that the outcome would be 1. If we have 3 or more outcomes however (i.e. 1,

2, and 3), we cannot implicitly assume that the value that divides a 1 and 2 outcome is the same as the value that divides a 2 and 3 outcome. That is to say that we may realize different changes in the probability of realizing a y=3 outcome, if we compare it to a probability of realizing a y=1, versus y=2, outcome. For this reason, we must incorporate the (n-1) cut-off points into ordinal logit's functional form, which can be written as follows:

$$\begin{aligned}\Pr(y = 1 | X_1, \dots, X_k) &= \Pr(v \leq \text{cut1}) \\ \Pr(y = 2 | X_1, \dots, X_k) &= \Pr(\text{cut1} \leq v \leq \text{cut2}) \\ \Pr(y = 3 | X_1, \dots, X_k) &= \Pr(\text{cut2} \leq v)\end{aligned}\tag{Eq.13.1}$$

where  $v = \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k + \varepsilon$ . Notice, that there are no y-intercepts in ordinal logit. These are replaced with cut-off points which can be used to calculate predicted probabilities of being in a particular category for an observation which has given values of an independent variable(s). On the right hand side, unlike logit, we have introduced multiple y outcomes. Consequently, in ordinal logit, we shift the analysis away from the probability of realizing a y=1 outcome in the dependent variable and towards the probability of realizing a y outcome for higher versus lower categories. Hence, with ordinal logit, we do not gauge the probability of one outcome, but multiple outcomes. This makes the interpretation of our results a bit different, and caution must be given to the interpretation of beta coefficients.

Because ordinal logit models are log transformed, they share several of logit's general assumptions (we will not discuss assumptions of multicollinearity and the omitted variable bias in this class, although these statistical problems apply similarly to ordinal logit as they did to logit). Five central assumptions of ordinal logit are:

1. The logit of  $\Pr(Y_i = 1, 2, 3\dots)$  depends on the logistically transformed values of all explanatory variables through the properly specified linear function  $\beta_1 X_1 + \dots + \beta_n X_n$  (similarly to logit, this assumes that the model does not suffer from an omitted variable bias)
2.  $\text{Logit}[\Pr(Y_i = 1, 2, 3\dots)]$  possesses a cumulative standard logit distribution (hence correcting the “unbounded” problem associated with linear probability models)
3. The relationship between each pair of outcome groups (i.e. “low”, “medium” and “high”) is the same; in other words, ordered logistic regression assumes that the coefficients that describe the relationship between the lowest versus the medium and high categories of the response variable are the same as those that describe the relationship between the medium and highest category, and so on. This is referred to as the proportional odds/parallel regression assumption.
4. No severe under- or over-representation of a dependent variable outcome (more likely to be fulfilled if there are fewer ordinal categories)
5. No (perfect) multicollinearity

Ordinal logit's first assumption implies that the model is non-linear. Hence, the interpretation of beta coefficients should not be treated in a similar manner as OLS; the influence of  $X_1$  on the likelihood of a y=1, 2, 3... outcome depends on the levels of all other independent variables.<sup>23</sup> We begin the lab with a

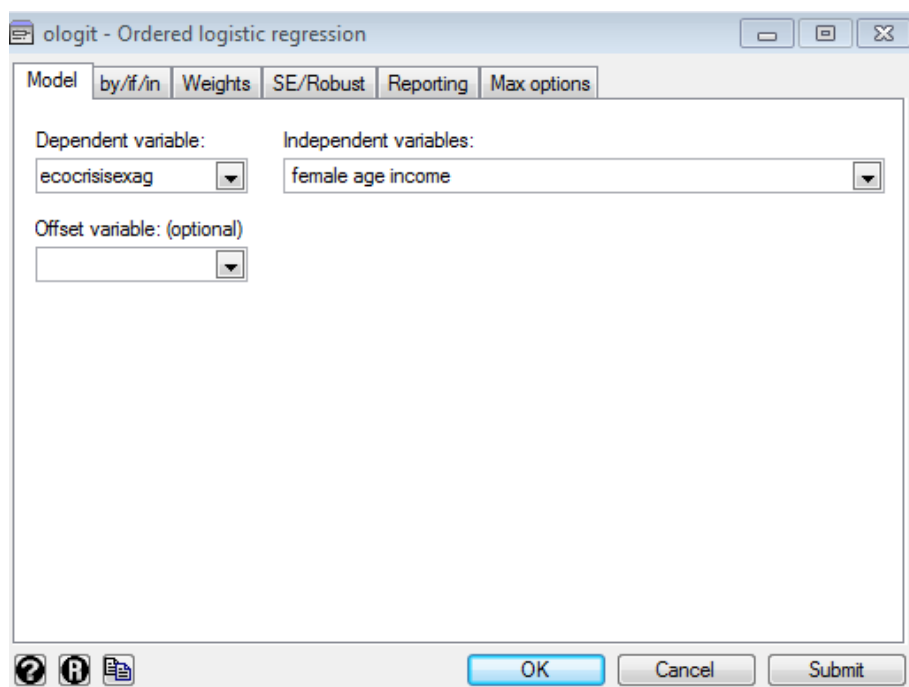
---

<sup>23</sup> Similar problems with interaction terms and quadratic terms in logit also apply to ordinal logit.

general presentation of ordinal logit output and how to interpret it. We will utilize data from the Oregon Energy survey, yet this data set has two additional variables which will serve as our (ordinal) dependent variables:

- ecocrisisexag: The respondent's self-reported opinion on whether the so called "ecological crisis" facing human kind has been exaggerated on a 5 point ordinal scale: 1 for "Strongly Disagree", 2 for "Mildly Disagree", 3 for "Neutral", 4 for "Mildly Agree" and, 5 for "Strongly Agree"
- humanmod: The respondent's self-reported opinion on whether humans have the right to modify the natural environment to suit their needs on a 5 point ordinal scale: 1 for "Strongly Disagree"; 2 for "Mildly Disagree"; 3 for "Neutral"; 4 for "Mildly Agree" and 5 for "Strongly Agree" (this will be the dependent variable for the practice problems)

Let's begin with a simple ordinal logit regression that analyzes the relationship between gender, income, and age (the independent variables), and an individual's (ordinal) opinion on whether the ecological crisis has been exaggerated (the dependent variable). To conduct an ordinal logit regression, click on the "Statistics" tab at the top of the page. Then click "Ordinal outcomes", followed by "Ordinal logistic regression". You should see the following box:



The screenshot shows a software dialog box titled "ologit - Ordered logistic regression". It has a tabbed interface with the "Model" tab selected. Other tabs include "by/if/in", "Weights", "SE/Robust", "Reporting", and "Max options". The "Dependent variable:" field contains "ecocrisisexag". The "Independent variables:" field contains "female age income". There is an "Offset variable: (optional)" field which is currently empty. At the bottom of the dialog are three buttons: "OK", "Cancel", and "Submit".

Select ecocrisisexag as your dependent variable and female, age and income as your primary independent variables. Click "Ok", you should see the following output:

```
. ologit ecocrisisexag female age income
```

```
Iteration 0:  log likelihood = -1021.2285
Iteration 1:  log likelihood = -981.64091
Iteration 2:  log likelihood = -981.46501
Iteration 3:  log likelihood = -981.46491
```

```
ordered logistic regression
```

```
Log likelihood = -981.46491
```

Number of obs	=	649
LR chi2(3)	=	79.53
Prob > chi2	=	0.0000
Pseudo R2	=	0.0389

ecocrisisexag	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
female	-1.185286	.1498928	-7.91	0.000	-1.47907 - .8915015
age	.0183817	.004094	4.49	0.000	.0103576 .0264059
income	-.064567	.0371436	-1.74	0.082	-.1373671 .0082331
/cut1	-.82931	.3302766			-1.47664 -.1819797
/cut2	-.0097153	.3266163			-.6498715 .630441
/cut3	.7926811	.3263546			.1530378 1.432324
/cut4	1.772311	.3353733			1.114991 2.42963

**CONGRATULATIONS! You have just conducted an ordered logistic regression in STATA!**

Ordinal logit output is similar to logit output in four respects. Firstly, the LR chi-squared statistic can be treated identically; if the statistic is highly significant, we can reject the null hypothesis that the overall model is insignificant (i.e. the log-odds values of our beta coefficients are all equal to zero). The second similarity between logit and ordinal logit is that both use maximum likelihood iteration techniques. This means that like with logit, STATA will continue to estimate a model via multiple iterations until the lowest (absolute value) log likelihood is produced. The third and fourth similarities relate to the beta coefficients of ordinal logit. Like logit, ordinal logit's beta coefficients are expressed in terms of log odds and hence you should not express your results in terms of marginal changes as you did in OLS. Fourthly, significance of the log odds beta coefficients can be determined by the z-statistic and its corresponding p-value. If a variable exhibits high significance with either a significantly large z-statistic, or a significantly low p-value, we can claim with high confidence that higher levels of that variable should significantly increase (or decrease) the likelihood of falling into higher (versus lower) ordinal categories.

### STATA COMMAND 13.1:

*Code:* “**ologit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

*Output produced:* Generates ordinal logistical regression output, where beta coefficients of independent variables are expressed in terms of log-odds.

Ordinal logit and logit differ from each other in three respects. The first, mentioned above, is that ordinal logit requires cut off points (these are estimated as the “cuts”, highlighted in yellow, above) to distinguish between the transition from one ordinal outcome to another. You will always be presented with n-1 cut-

off points, as indicated in the functional form of Equation 13.1 above. These tell STATA, when you estimate your fitted probabilities, for what values  $v(\beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k + \varepsilon)$  should be bounded by when estimating the likelihood of falling into (higher) categories. The second and third differences between ordinal logit and logit relate to the interpretation of beta coefficients and significance (differences in significance claims are assessed below). In logit, if we were presented with a significantly positive beta coefficient, we could claim that an increase in  $X_1$  led to a higher probability of a  $y=1$  outcome. In ordinal logit, however, there is more than one possible outcome. Hence you have to interpret your results more generally. In the case of the beta coefficient on the female dummy, we can claim that “women are less likely to fall into higher agreement categories than the belief that the ecological crisis is exaggerated, compared to men”.<sup>24</sup> However, we cannot determine the ordinal category (i.e. “strongly” disagree, “mildly disagree”, “neutral”, “mildly agree”, and “strongly agree”) that women would lose this enhanced probability. Hence, you could not claim that women are less likely to fall into the “strongly agree” “mildly agree” and “neutral” categories, relative to men. The “prvalue” command can be quite helpful in helping you distinguish between which two ordinal categories enhanced probabilities for  $X_1$  shift to reduced probabilities.

Let’s determine the likelihood that women, relative to men, will fall into higher agreement categories with the statement that the ecological crisis is exaggerated. Type in “prvalue, x(female=1) save” into the STATA command box. This is a slight difference in syntax from our previous lessons, where we only examined the probability value for a  $y=1$  outcome, relative to a  $y=0$  outcome. You should be presented with the following outcome:

```
. prvalue, x(female=1) save
ologit: Predictions for ecocrisisexag
Confidence intervals by delta method
```

		95% Conf. Interval
Pr(y=1 x):	0.4212	[ 0.3711, 0.4712]
Pr(y=2 x):	0.2017	[ 0.1683, 0.2351]
Pr(y=3 x):	0.1637	[ 0.1349, 0.1924]
Pr(y=4 x):	0.1210	[ 0.0958, 0.1462]
Pr(y=5 x):	0.0925	[ 0.0698, 0.1152]

```

      female      age      income
x=         1    54.14792  4.9768875

```

We can deduce from this particular model’s output that women are highly represented within the “strongly disagree” category concerning the exaggeration of the ecological crisis. However, you may also notice, given the overlap in categorical confidence intervals, that women are not significantly less likely of falling in the “mildly disagree” compared to the “neutral category”. Moreover, they are not significantly less likely of falling in the “neutral”, compared to the “mildly agree” category, at least at a 95% confidence level. Similar to logit, given multiple categories for the dependent variable, you should never interpret probability output for one value of  $X_1$  in isolation. Rather, ordinal output needs to be

<sup>24</sup> Alternatively, one could claim that women are more likely to fall into higher disagreement categories than men.

compared and interpreted across multiple values of  $X_1$ . We can do this via the “save” and “diff” syntax modifications of the “prvalue” command. In the STATA command box, immediately after you typed in the command “prvalue, x(female=1) save”, type “prvalue, x(female=0) diff”. You should be presented with the following output:

```
. prvalue, x(female=1) save
ologit: Predictions for ecocrisisexag
Confidence intervals by delta method

Pr(y=1|x):      0.4212      [ 0.3711,    0.4712]
Pr(y=2|x):      0.2017      [ 0.1683,    0.2351]
Pr(y=3|x):      0.1637      [ 0.1349,    0.1924]
Pr(y=4|x):      0.1210      [ 0.0958,    0.1462]
Pr(y=5|x):      0.0925      [ 0.0698,    0.1152]

      female      age      income
x=          1    54.14792    4.9768875

. prvalue, x(female=0) diff
ologit: Change in Predictions for ecocrisisexag
Confidence intervals by delta method

Pr(y=1|x):      Current      Saved      Change      95% CI for Change
Pr(y=2|x):      0.1819      0.4212     -0.2392     [-0.2960,   -0.1825]
Pr(y=3|x):      0.1535      0.2017     -0.0482     [-0.0679,   -0.0284]
Pr(y=4|x):      0.1942      0.1637      0.0305     [ 0.0143,    0.0468]
Pr(y=5|x):      0.2203      0.1210      0.0993     [ 0.0703,    0.1283]
Pr(y=5|x):      0.2501      0.0925      0.1576     [ 0.1140,    0.2012]

      female      age      income
Current=         0    54.14792    4.9768875
Saved=           1    54.14792    4.9768875
Diff=            -1         0         0
```

**CONGRATULATIONS! You have just conducted an assessment of fitted probabilities for ordinal logit in STATA!**

The change in predictions table above provides a better assessment for how  $X_1$  influences the probability of falling into a particular ordinal outcome. Notice that for “current” output (i.e. output for “men”, a female coding of 0), there is a significantly lower likelihood that men will fall into either the “strongly disagree” (coding of 1), or “mildly disagree” (coding of 2) category than women. We can assess this significance based upon the 95% confidence interval for change (far-right column). If this column were straddling 0 (i.e. the upper and lower bound held alternative signs), we could not claim with significance that men were less likely to lie within lower attitudinal categories. Moreover, you may notice that women’s significance probability of lying within in lower ordinal categories ends at the “neutral” category (coding of 3). In other words, while women more likely to fall into the 1<sup>st</sup> and 2<sup>nd</sup> ordinal category (strongly or mildly disagree with the statement that the ecological crisis is exaggerated), men are more likely to fall into the 3<sup>rd</sup>, 4<sup>th</sup> and 5<sup>th</sup> ordinal category relative to women.

### STATA COMMAND 13.2:

*Code:* “**ologit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

“**prvalue, x(var2=#) save**”

“**prvalue, x(var2=#+1) diff**”

*Output produced:* Calculates the change in fitted probability of all ordinal outcomes for a (1 unit) increase in var2, as well as whether the differences in probabilities within each individual category is significant for the two values of var2.

A third difference between logit and ordinal logit pertains to the issue of significance of continuous variables. For logit, only two outcomes were possible for the dependent variable: 0 and 1. If the influence of  $X_1$  was significant on a  $y=1$  outcome in logit, we could assume that it would also be significant for an outcome of  $y=0$ ; significance of continuous variables, in other words, was symmetrical across dependent variable outcome categories. In ordinal logit, however, outcome categories are not inversely identical to each other on their own. Hence, significance of continuous and dummy independent variables may be asymmetrical across categories, even when the log odds beta coefficient is highly significant.  $X_1$  may have a significant relationship across all values of one ordinal category, but may lack significance in another. Under this pretense, it is imperative that a researcher check the significance of continuous variables across the entire range of  $X_1$  for all  $y$  outcomes with the “prvalue” or “prgen” command. Failure to do so may result in faulty conclusions and an improper diagnosis of a negative/positive relationship between  $X_1$  and a certain  $y$  outcome across all values of  $X_1$ . To demonstrate the danger of loosely interpreting the significance associated with a beta coefficient, we turn to the influence of age on the level of agreement with the claim that the ecological crisis is greatly exaggerated.

From the output above, age’s beta coefficient is negative and shares a highly significant (well above a 99% confidence level) relationship with agreement over ecological crisis exaggeration. From this output, we may be tempted to conclude that younger individuals are more likely to fall into higher disagreement categories of the belief that the ecological crisis is exaggerated, compared to older individuals. If you made such a statement across all values of age, however, you would be slightly incorrect in your treatment of the “mildly disagree” category. To demonstrate, let’s start by graphing the fitted probabilities of lying within the 5 ordinal categories across all values of age. Using the “prgen” command, type the following syntax into the STATA command box immediately after the ordinal logit regression output above: “prgen age, from(18) to(94) generate(prAGE) rest(mean) gap(4) ci”. You should see the following added independent variables:

Variables										
Name	Label	Type	F							
unemployed	Unemployed	byte	%							
collegedummy		float	%							
prAge <sub>x</sub>	Changing value of age	float	%							
prAge <sub>p1</sub>	pr(1)	float	%							
prAge <sub>p2</sub>	pr(2)	float	%							
prAge <sub>p3</sub>	pr(3)	float	%							
prAge <sub>p4</sub>	pr(4)	float	%							
prAge <sub>p5</sub>	pr(5)	float	%							
prAge <sub>p1lb</sub>	LB pr(1)	float	%							
prAge <sub>p2lb</sub>	LB pr(2)	float	%							
prAge <sub>p3lb</sub>	LB pr(3)	float	%							
prAge <sub>p4lb</sub>	LB pr(4)	float	%							
prAge <sub>p5lb</sub>	LB pr(5)	float	%							
prAge <sub>p1ub</sub>	UB pr(1)	float	%							
prAge <sub>p2ub</sub>	UB pr(2)	float	%							
prAge <sub>p3ub</sub>	UB pr(3)	float	%							
prAge <sub>p4ub</sub>	UB pr(4)	float	%							

female	-1.185286	.1498928	-7.91	0.000	-1.47907	-.8915015
age	.0183817	.004094	4.49	0.000	.0103576	.0264059
income	-.064567	.0371436	-1.74	0.082	-.1373671	.0082331

/cut1	-.82931	.3302766			-1.47664	-.1819797
/cut2	-.0097153	.3266163			-.6498715	.630441
/cut3	.7926811	.3263546			.1530378	1.432324
/cut4	1.772311	.3353733			1.114991	2.42963

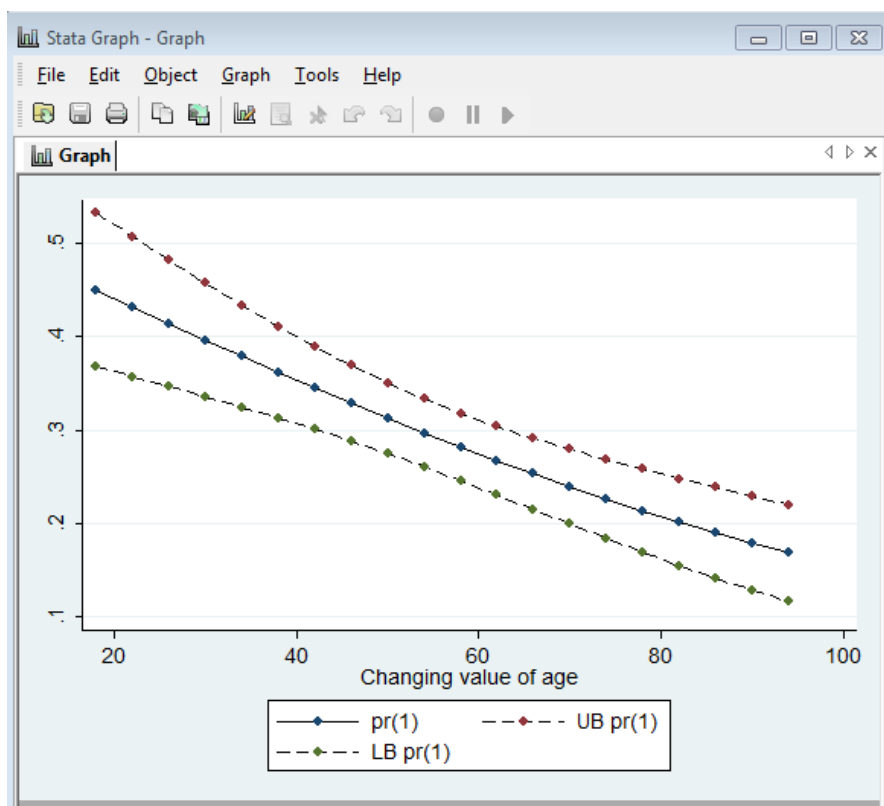
  

```

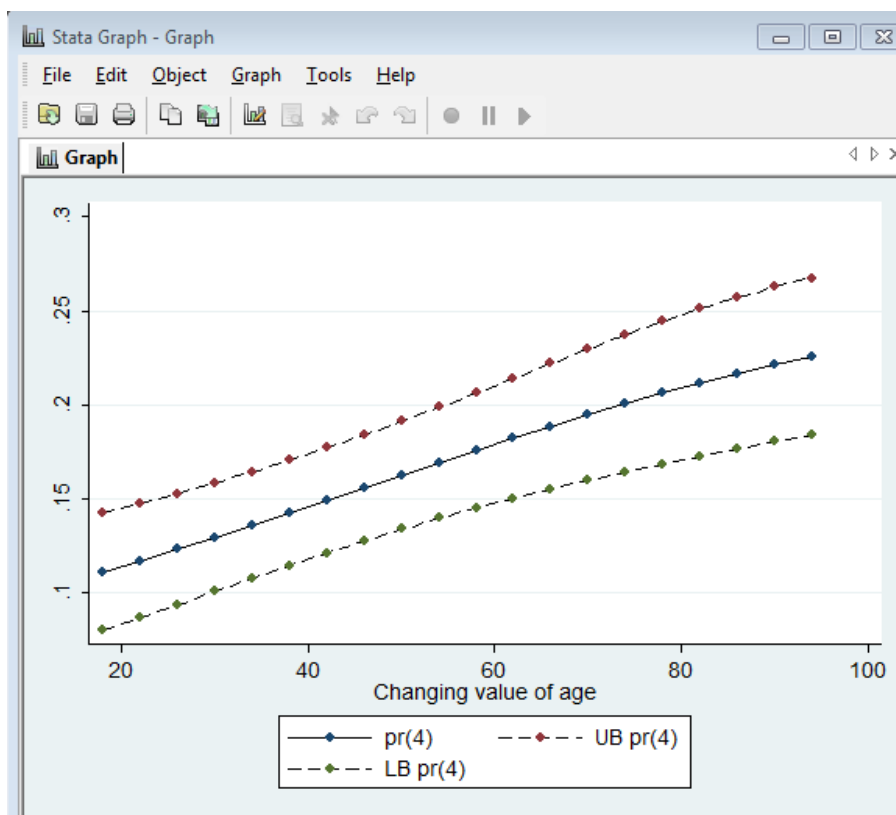
. prgen age, from(18) to(94) generate(prAge) rest(mean) gap(4) ci
logit: Predicted values as age varies from 18 to 94.
      female      age      income
x=   .53929122   54.14792   4.9768875

```

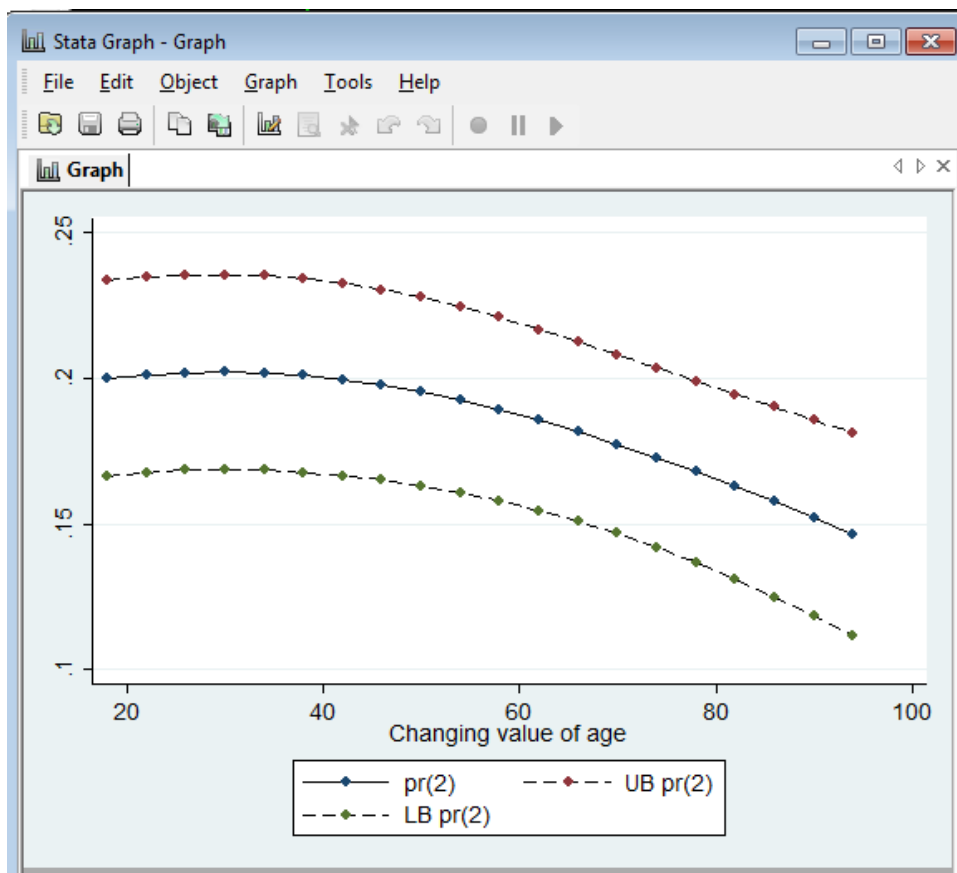
Like the logit lessons, the “prgen” command has added new variables to our variable box. Contrary to logit, however, 5 ordinal categories and their upper and lower bounds are represented rather than 2 (0 and 1), in accordance to the 5 point scale of how the dependent variable is measured. We can use the same syntax as we did for logit to graph the fitted probabilities across all values of age. Let’s start with calculating the fitted probability of a “strongly disagree” outcome (coding of 1). Type the following code into the STATA command box: “graph twoway (connected prAge<sub>p1</sub> prAge<sub>x</sub>, clcolor(black) clpat(solid)) (connected prAge<sub>p1ub</sub> prAge<sub>x</sub>, clcolor(black) clpat(dash)) (connected prAge<sub>p1lb</sub> prAge<sub>x</sub>, clcolor(black) clpat(dash))”. You should be presented with the following output:



Notice that as age increases, the likelihood of strongly disagreeing with the statement that the ecological crisis is greatly exaggerated wanes. Moreover, this decline is persistent across all values of age. Now change the above syntax to graph the fitted probability of “mildly agreeing” with the statement that the ecological crisis is greatly exaggerated (coding of 4) by typing the following code into the STATA command box: “graph twoway (connected prAgep4 prAgex, clcolor(black) clpat(solid)) (connected prAgep4ub prAgex, clcolor(black) clpat(dash)) (connected prAgep4lb prAgex, clcolor(black) clpat(dash))”. You should be presented with the following output:



Again, a similar result emerges when we examine the impact of age on categories with greater agreement. As age increases, the likelihood of mildly agreeing with the statement that the ecological crisis is greatly exaggerated increases, as we would deduce from the sign on our beta coefficient in the ordinal logit output. Let’s produce one more graphic, however, examining the influence of age on mildly disagreeing with the statement that the ecological crisis has been greatly exaggerated. Type in the following syntax into the STATA command editor: “graph twoway (connected prAgep2 prAgex, clcolor(black) clpat(solid)) (connected prAgep2ub prAgex, clcolor(black) clpat(dash)) (connected prAgep2lb prAgex, clcolor(black) clpat(dash))”. You should see the following output:



Notice that though mild disagreement with the statement that the ecological crisis is greatly exaggerated decreases as age increases, this decline only happens after the age of 50; between our minimum age and 50, there is no significant difference in likelihood of mildly disagreeing with the statement that the ecological crisis is greatly exaggerated. To confirm this, type “prvalue, x(age=20) save” and then “prvalue, x(age=50) diff” into the STATA command box. You should see the following output:

```

. prvalue, x(age=20) save
ologit: Predictions for ecocrisisexag
Confidence intervals by delta method

      Pr(y=1|x):      0.4412      [ 0.3628,      0.5196]
      Pr(y=2|x):      0.2006      [ 0.1670,      0.2342]
      Pr(y=3|x):      0.1581      [ 0.1248,      0.1913]
      Pr(y=4|x):      0.1142      [ 0.0835,      0.1450]
      Pr(y=5|x):      0.0859      [ 0.0576,      0.1142]

      female      age      income
x= .53929122      20  4.9768875

. prvalue, x(age=50) diff
ologit: Change in Predictions for ecocrisisexag
Confidence intervals by delta method

      Pr(y=1|x):      Current      Saved      Change      95% CI for Change
      Pr(y=2|x):      0.3126      0.4412      -0.1285      [-0.1890, -0.0681]
      Pr(y=3|x):      0.1953      0.2006      -0.0053      [-0.0150,  0.0044]
      Pr(y=4|x):      0.1893      0.1581      0.0312      [ 0.0118,  0.0506]
      Pr(y=5|x):      0.1626      0.1142      0.0483      [ 0.0271,  0.0695]
      Pr(y=5|x):      0.1402      0.0859      0.0543      [ 0.0353,  0.0734]

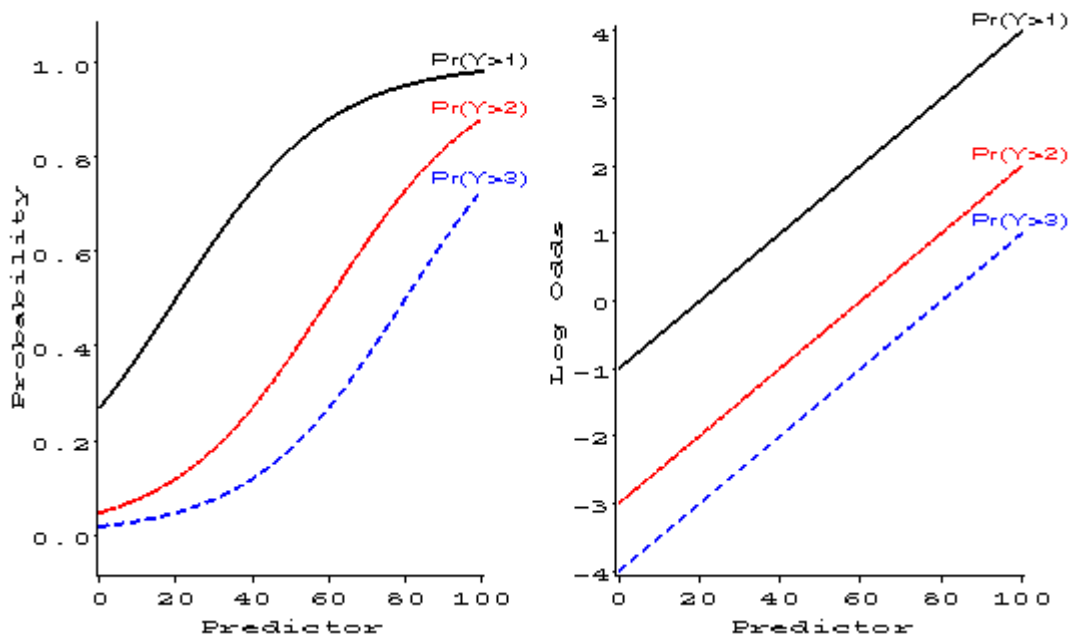
      female      age      income
Current= .53929122      50  4.9768875
Saved=   .53929122      20  4.9768875
Diff=    0           30      0

```

The “prvalue” output documenting the difference in fitted probabilities for a 20 year old and a 50 year old confirm the output yielded by the fitted probability graphics above. 50 year olds, on average, are more likely to either be “neutral” or mildly or strongly agree with ecological crisis exaggeration than 20 year olds, while the latter are more likely to strongly disagree with this statement. However, outlined in yellow above, the two age groups do not significantly differ in mild disagreement over the statement. If a more drastic age comparison were made, say between a 20 year old and a 60 year old, a significant difference would emerge between younger and older age groups in the mild disagreement category. Such an example emphasizes that, when dealing with ordinal logit, one must be careful in how the significance of results, particularly for continuous independent variables, are interpreted. Highly significant beta coefficients do not imply significance across all values of  $X_1$  for all ordinal outcomes. Researchers should be scrupulous and check significance across all possible values of  $X_1$ .

One assumption which distinguishes ordinal logit from logit analysis is Assumption 3, the proportional odds assumption. The proportional odds/parallel regression assumption is unique to ordinal logit analysis. In layman’s terms, this assumption means that the relationship between any two outcome groups is statistically identical. If the proportional odds assumption is satisfied, the beta coefficients associated with the independent variables for each ordinal outcome (i.e. 1, 2, 3, 4, and 5) would be identical, and probability distributions for each ordinal outcome would look like the following:

## Proportional Odds Model



Source: Friendly, 1995 (<http://www.datavis.ca/courses/grcat/grc6.html>)

In this particular diagram, we are presented with an ordinal logit analysis for four ordinal outcome variables. To test the proportional odds assumptions, we run (J-1) binary regressions; in this case J is the number of categorical outcomes. This allows us to determine whether the beta coefficients on the independent variables for the regression whose binary dependent variable compares an outcome of 1 ( $y=1$ ) to an outcome of 2, 3, and 4 ( $y=0$ ) are identical to those produced when the dependent variable compares an outcome of 1 and 2 ( $y=1$ ), to 3 and 4 ( $y=0$ ), and (finally) whether these are similar to the regression betas where the dependent variable compares an outcome of 1, 2 and 3 ( $y=1$ ), to 4 ( $y=0$ ). If all three regressions have similar beta coefficients, their (logistic) distribution will be parallel across all values of X (see figure above), indicating that the proportional odds assumption holds. Though the proportional odds assumption is a central corner-stone for ordinal logistic analysis, it is frequently violated.

Like the White test of heteroskedasticity, the Ramsey RESET test, and variance inflation factors (VIF), the Brant test<sup>25</sup>, which determines whether the proportional odds assumption is fulfilled, is a post-estimation command which is unique to ordinal model specification. The null hypothesis for the Brant test is that the parallel regression/proportional odds assumption is fulfilled. Hence, a significant Chi-squared statistic indicates one would have to reject that the proportional odds assumption is valid. Let's test whether the proportional odds assumption is fulfilled on the above regression model, where the level of agreement with the statement that the ecological crisis is greatly exaggerated is a function of age, income and gender. Immediately after you type the ordinal logit regression "ologit ecocrisisexag female age income", type "brant" in the STATA command box. You should see the following output:

<sup>25</sup> If your version of STATA does not have the Brant test, type "findit brant" into the STATA command editor. Click on the "spostado from <http://www.indiana.edu/~jslsoc/stata>" hyperlink in the "Web resources from STATA and other users" section, and then click "click here to install". This installation package will also download the "prgen" and "prvalue" command for your STATA program.

```
. ologit ecocrisisexag female age income
```

```
Iteration 0:  log likelihood = -1021.2285
Iteration 1:  log likelihood = -981.64091
Iteration 2:  log likelihood = -981.46501
Iteration 3:  log likelihood = -981.46491
```

```
Ordered logistic regression
```

```
Log likelihood = -981.46491
```

```
Number of obs   =      649
LR chi2(3)      =      79.53
Prob > chi2     =      0.0000
Pseudo R2      =      0.0389
```

ecocrisis~g	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
female	-1.185286	.1498928	-7.91	0.000	-1.47907 - .8915015
age	.0183817	.004094	4.49	0.000	.0103576 .0264059
income	-.064567	.0371436	-1.74	0.082	-.1373671 .0082331
/cut1	-.82931	.3302766			-1.47664 -.1819797
/cut2	-.0097153	.3266163			-.6498715 .630441
/cut3	.7926811	.3263546			.1530378 1.432324
/cut4	1.772311	.3353733			1.114991 2.42963

```
. brant
```

```
Brant Test of Parallel Regression Assumption
```

variable	chi2	p>chi2	df
All	57.07	0.000	9
female	8.88	0.031	3
age	30.04	0.000	3
income	21.57	0.000	3

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

**CONGRATULATIONS!** You have just conducted a Brant test of the parallel regression assumption in STATA!

You'll notice that the associated Chi-squared statistic for all independent variables (57.07) is highly significant, and hence we must reject that the parallel regression/proportional odds assumption holds. A more detailed version of the Brant test can be obtained by typing the following regression command into the STATA command editor: "brant, detail". You should be presented with the following:

```

. brant, detail

Estimated coefficients from j-1 binary regressions

female   -1.1182708   -1.3259815   -1.0163885   -1.3484918
age       .02506526    .00625174    .01578112    .02421135
income    -.12427094     -.15086037    .00785441    -.04736719
_cons     .74092342      1.1742072    -1.0998324    -2.1586112

Brant Test of Parallel Regression Assumption

+-----+-----+-----+-----+
| variable | chi2 | p>chi2 | df |
+-----+-----+-----+-----+
| All      | 57.07 | 0.000  | 9  |
+-----+-----+-----+-----+
| female   | 8.88  | 0.031  | 3  |
| age      | 30.04 | 0.000  | 3  |
| income    | 21.57 | 0.000  | 3  |
+-----+-----+-----+-----+

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

```

Using the detailed Brant test, we receive the same test-statistic overall, but are presented with stacked (J-1) (binary) logistic regressions. The results present the beta coefficients of the four re-coded binary logistic regression: 1) a 1 versus a 2, 3, 4, and 5 outcome, 2) a 1 and 2 outcome versus a 3, 4, and 5 outcome, 3) a 1,2, and 3 outcome versus a 4 and 5 outcome, and 4) a 1, 2, 3, and 4 outcome versus a 5 outcome. If the parallel regression/proportional odds assumption were satisfied, the beta coefficients for each independent variable (except the constant) would be the same across all auxiliary regressions. In accordance with our general Brant test result, we can see this is not the case.<sup>26</sup>

### STATA COMMAND 13.3:

*Code:* “**ologit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**“brant”**

*Output produced:* Conducts a Brant test for the proportional odds/parallel regression assumption in ordinal logit. If you are presented with a significant chi-squared statistic (i.e. with a p-value below 0.10), you must reject the null hypothesis that the assumption is satisfied.

<sup>26</sup> In some circumstances (i.e. when one or two ordinal categories are under-represented), you may discover that you are unable to run a Brant test with your ordinal model. If this is the case, simply type “**omodel logit DV IV1 IV2 IV3...**” into the STATA command editor. This estimation command will provide you with the Chi-squared statistic for the proportional odds assumption below the regression output. You may have to download the “omodel” command before you do this.

#### STATA COMMAND 13.4:

*Code:* “**ologit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
“**brant, detail**”

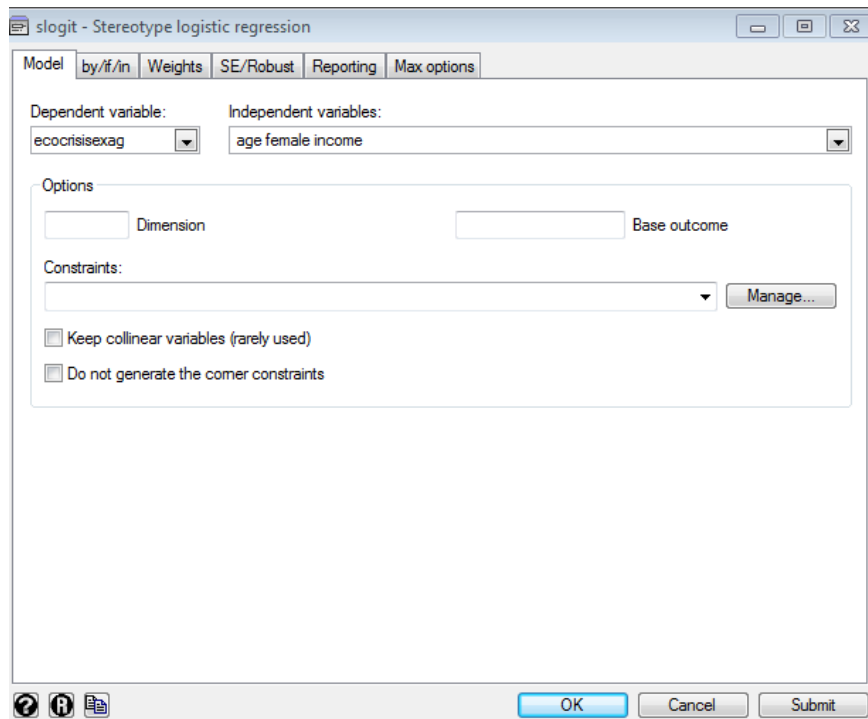
*Output produced:* Provides detailed beta coefficients of the (n-1) binary logistic auxiliary regressions used in the Brant test. The beta coefficients of the independent variable should be identical if the parallel assumption is fulfilled .

The bad news about the Brant test, and the parallel regression/proportional odds assumption, is that the assumption is frequently violated. Moreover, unlike the White test (for heteroskedasticity) or the Ramsey RESET test (for model specification), better specifying your model by adding more independent variables or increasing your sample size may result in a larger Brant test statistic. If you discover that the Brant test is violated, there are two general approaches you can take. The first is you can simply ignore the violation and continue using ordinal logit, which is the most common route taken in social science research. If you decide to do this, however, you must be cautious of the following: 1) you cannot use or interpret odds ratios, as they are not identical between outcome categories, and 2) you have to manually check significance and influence of your independent variables, similar to the process we employed above.

A second option you can pursue if you find the Brant test violated is to utilize an alternative ordinal (or multinomial) logistics model which relaxes the proportional odds assumption. There are several alternatives one can utilize, but due to its simplicity we will focus on the stereotype logistic model, developed by Anderson (1984).<sup>27</sup> A stereotype logistic model can be used for ordinal and nominal dependent variables. Unlike ordinal logit, the stereotype logistic model allows the beta coefficient for each independent variable to vary by outcome category (we will discuss how this is done below). In order to conduct a stereotype regression, click on the “Statistics” tab, followed by “Categorical Outcomes”. Click on “Stereotype logistic regression”. You should be presented with the following box:

---

<sup>27</sup> The generalized ordered logit model is another method which relaxes the proportional odds assumption, but its output is more complex as it treats ordinal categories like multinomial categories (hence J-1 sets of beta coefficients are produced for J ordinal outcomes).



Insert your dependent variable, *ecocrisisexag*, into the dependent variable box, and *age*, *income* and *female* into the independent variable box. Click “Ok”. You should be presented with the following output:

```
. slogit ecocrisisexag age female income
```

```
Iteration 0: log likelihood = -1366.8721 (not concave)
Iteration 1: log likelihood = -1092.3213 (not concave)
Iteration 2: log likelihood = -1018.5782 (not concave)
Iteration 3: log likelihood = -986.44333 (not concave)
Iteration 4: log likelihood = -981.36795
Iteration 5: log likelihood = -980.65838
Iteration 6: log likelihood = -979.56463
Iteration 7: log likelihood = -979.42842 (backed up)
Iteration 8: log likelihood = -978.61991
Iteration 9: log likelihood = -977.95776
Iteration 10: log likelihood = -977.9066
Iteration 11: log likelihood = -977.90567
Iteration 12: log likelihood = -977.90567
```

```
Stereotype logistic regression      Number of obs   =      649
                                Wald chi2(3)         =      65.30
Log likelihood = -977.90567         Prob > chi2      =      0.0000
```

```
( 1) [phi1_1]_cons = 1
```

ecocrisisexag	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
age	.0308837	.0093101	3.32	0.001	.0126362 .0491311
female	-2.047441	.2763719	-7.41	0.000	-2.58912 -1.505762
income	-.1977495	.0703146	-2.81	0.005	-.3355636 -.0599354
/phi1_1	1	.	.	.	.
/phi1_2	.6260673	.1157668	5.41	0.000	.3991685 .8529661
/phi1_3	.3630785	.122764	2.96	0.003	.1224654 .6036916
/phi1_4	.4731726	.105903	4.47	0.000	.2656065 .6807387
/phi1_5	0	(base outcome)			
/theta1	.3033771	.7185189	0.42	0.673	-1.104894 1.711648
/theta2	.000124	.4913953	0.00	1.000	-.9629931 .9632411
/theta3	.0950786	.2868141	0.33	0.740	-.4670667 .657224
/theta4	-.0023455	.3681824	-0.01	0.995	-.7239697 .7192787
/theta5	0	(base outcome)			

```
(ecocrisisexag=5 is the base outcome)
```

## CONGRATULATIONS! You have just conducted a stereotype logistic regression!

Notice that the beta coefficients hold the same sign, but the significance on income has improved. Without having to worry about the proportional odds/parallel regression assumption, we can interpret the above output to indicate that individuals with higher incomes will be less likely to fall in higher agreement categories compared to individuals with low incomes. Likewise, women are less likely to fall into higher agreement categories that ecological crisis is exaggerated than men. Finally, older individuals should be more likely to fall within strong agreement categories, relative to younger individuals.

A word of caution must be applied to the actual value of the beta coefficients for stereotype logistic regression. You may notice that some of these coefficients differ markedly from those produced in ordinal logit. This is because the beta coefficients are not equivalent for each ordinal outcome category. Recall from above that in stereotype logit models, the beta coefficient for each independent variable can vary by outcome category – this is because they must be multiplied by the ratio of respective  $\phi$ s (phi), which will always be positive and will be declining as the numerical category increases (like the thetas, which are similar to ordinal logit's “cuts”, a base category is always selected for  $\phi$ , and hence dropped). The proportional odds ratio assumption is relaxed in stereotype logit, because of these parameters  $\phi$ ,

which measure the difference between categories, and rescale probabilities associated with beta coefficients accordingly – hence beta coefficients must be interpreted simultaneously with, not in isolation of, the scalars  $\phi$ . If you find that two  $\phi$ s are equal, then the model does not distinguish between these two categories, and they can be collapsed (the “replace” command in the Appendix on data cleaning will teach you how to do this).

#### **STATA COMMAND 13.5:**

*Code:* “**slogit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

*Output produced:* Conducts a stereotype logistic regression, which relaxes the proportional odds assumption, in STATA.

The “prvalue” command can be used for the “slogit” command, but only in more recent versions of STATA (these commands work for “slogit” in STATA 12 and 11.2, but not STATA 10). You may need to check whether this command is compatible with slogit in your version of STATA by typing “help prvalue” into the STATA command box and checking to see whether “slogit” is on the list of models which “prvalue” serves as a post-estimation command. If it does, you can use it in a manner identical to that used for ordinal logit; however, unlike ordinal logit, it will not produce confidence intervals associated with your fitted probabilities! “Prgen” is not compatible as a post-estimation command of slogit – you will have to revert back to ordinal logit for its use. Ordinal logit is frequently used as the ordinal analysis tool of choice, even though one of its corner-stone assumptions is frequently violated. As long as caution is exerted in result interpretation, and conclusions are checked via the “prgen” and “prvalue” commands, ordinal logit is sufficient in the analysis of the impact of continuous and dummy independent variables on ordinal outcomes.

---

### **Practice Problems:**

Practice Problem 1: Run an ordinal logit model which predicts the effect of the four following independent variables on the level of agreement that an individual believes humans have the right to modify the natural environment to suit their needs (humanmod): gender, age, income and political ideology. Which variables exhibit a significant influence on agreement levels with the statement that humans have the right to modify the environment? What general conclusions can you make regarding these variables?

Practice Problem 2: Using the “prvalue” command on the regression output above, in which agreement categories are women significantly more likely to fall, relative to men, regarding beliefs in environmental modification?

Practice Problem 3: Using the “prgen” command on the regression output above, is there a significantly consistent decline across the political ideological scale for the “strongly disagree” category? For the “mildly disagree” category? How would you modify your general conclusions about political ideology’s influence on agreement with environmental modification in question 1 for the “mildly disagree” category? How would you interpret the influence of political ideology on an individual’s likelihood of falling into “neutral” category on environmental modification?

Practice Problem 4: Conduct a Brant test to determine if the above model fulfills the proportional odds assumption. Can you conclude that it does? If it does not hold, conduct a stereotype logistic regression. What happens to your beta coefficients and significance of your independent variables? Are there any significant changes relative to the ordinal logit model?

## Lesson 14: Multinomial Logistic Regression Analysis

---

**Learning Objective 1: Estimating and interpreting the output of multinomial logistic regression in STATA**

**Learning Objective 2: Determining the significance of an independent variable in multinomial logistic regression via an LR test and a Wald test**

**Learning Objective 3: Interpreting beta coefficients in multinomial logit as factor changes in relative odds**

**Learning Objective 4: Calculating and graphing fitted probabilities for the likelihood of categorical outcomes in STATA**

**Learning Objective 5: Testing the independence of irrelevant alternatives assumption for multinomial logistic analysis via the Hausman test**

In the previous lessons, we used logistic regression (or a variation of it) to examine two types of dependent variables: nominal variables with only two categories (for which logistic regression was used) and nominal variables with multiple categories which can be ranked (for which ordinal logistic regression was used). In the last lesson, you discovered some limitations of ordinal logistic regression, namely that its results (in terms of significance and sign) may not transcend all categories if the proportional odds/parallel regression assumption was violated. In this lesson, we focus on a type of logistic regression which can overcome this problem: multinomial logistic regression.

Multinomial logistic regression is primarily used for nominal dependent variables with multiple categories whose categories cannot be ranked. Like logistic and ordinal logistic regression, these categories **MUST** be mutually exclusive (an observation cannot belong to two categorical outcomes of the dependent variable). Examples of such variables include the type of college major a student chooses for his/her primary major, an individual's political party affiliation (assuming more than two political parties – i.e. Republican, Democrat, Green, Independent), the candidate an individual votes for in an election (assuming more than two candidates are running), and occupation (i.e. whether one is a teacher, a manual laborer, an administrator, a businessman/women, etc.). Multinomial logistic regression can also be used for ordinal dependent variables, in the case that ordinal logistic regression is not a suitable modeling technique. In the last lesson, we discussed that if the proportional odds/parallel regression assumption was violated, we could not rely on ordinal logistic regression to produce transitive results for all ordinal categories. Multinomial logistic regression, however, does not rely upon this restrictive assumption, and therefore its flexibility permits the examination of ordinal categories vis-à-vis one another (although you will soon note that this can involve a very tedious analysis of pair-wise comparison between categorical outcomes!).

Multinomial logit, like ordinal logit and standard logistic regression, involves a logistically transformation of a baseline linear regression model. Unlike ordinal logit, however, it does not involve

cut-off points which mark the shift from a lower ordinal category to a higher ordinal one. Multinomial logistic regression treats every outcome category as unique; though these categories are assigned a numerical code to tell them apart, their code has no numerical value and cannot be ranked as “higher” or “lower” relative to others. Consequently, multinomial logistic output is strictly comparative; the (log odds) of an outcome is assessed relative to other outcomes. Assume we have three categorical outcomes for a dependent variable (suppose we are interested in whether a student’s primary major is within the Arts and Humanities – AH – the sciences – SCI – or the social sciences –SOCIAL). Choosing the Arts and Humanities as the baseline outcome category of comparison, the functional form of multinomial logistic regression can be expressed as follows:

$$\Pr(y = \text{AH, SCI or SOCIAL} | x) = \begin{cases} \frac{\Pr(\text{SCI} | x)}{\Pr(\text{AH} | x)} = \Lambda(\beta_{0, \text{SCI} | \text{AH}} + \beta_{1, \text{SCI} | \text{AH}} X_1 + \dots + \beta_{k, \text{SCI} | \text{AH}} X_k) \\ \frac{\Pr(\text{SOCIAL} | x)}{\Pr(\text{AH} | x)} = \Lambda(\beta_{0, \text{SOCIAL} | \text{AH}} + \beta_{1, \text{SOCIAL} | \text{AH}} X_1 + \dots + \beta_{k, \text{SOCIAL} | \text{AH}} X_k) \end{cases}$$

where  $\Lambda$  is the logistic transformation  $\frac{e^{(\beta_0 + \beta_1 X_1 + \dots + \beta_k X_k)}}{1 + e^{(\beta_0 + \beta_1 X_1 + \dots + \beta_k X_k)}}$  or  $\frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \dots + \beta_k X_k)}}$ . Notice from the functional form that results are expressed relative to a reference outcome category (in the above example, Arts and Humanities). In this respect, multinomial logit behaves very similarly to multiple categorical dummies; we can always compare one categorical outcome to a baseline/reference category, but if we want to compare between two non-baseline categories, in our case compare SCI and SOCIAL, we must change the reference category.

Like logit and ordinal logit, multinomial logit produces beta coefficients in terms of (relative) log odds. Yet compared to the beta coefficients for OLS, logistic regression, and even ordinal logistic regression, the interpretation of beta coefficients for multinomial logistic regression is most restrictive. Because the log odds are relative (comparing an outcome category to a reference category) all we can conclude from a beta coefficient in multinomial logit is how an increase in an independent variable will change the likelihood of falling in one outcome category relative to a baseline category. In other words, if we wanted to assess the impact of  $X_1$  on primary college major decisions by college students, we would have to do so on a pair-wise comparison basis, as  $X_1$  has two beta coefficients associated with it in the above model ( $\beta_{1, \text{SCI} | \text{AH}}$  and  $\beta_{1, \text{SOCIAL} | \text{AH}}$ ). If  $\beta_{1, \text{SCI} | \text{AH}}$  were positive, the only claim we could make is that an increase in  $X_1$  increases the likelihood of a respondent being a science major compared to an arts and humanities major. Likewise, if  $\beta_{1, \text{SCI} | \text{AH}}$  were negative, the only claim we could make is that an increase in  $X_1$  decreases the likelihood of a respondent being a science major compared to an arts and humanities major. Other than these general comparative states, we could determine nothing more about the magnitude of  $X_1$  on being a science relative to an arts and humanities major. We could also make no statement about how  $X_1$  influences the likelihood of being a social sciences major by only looking at  $\beta_{1, \text{SCI} | \text{AH}}$ . Our interpretation of multinomial logistic beta coefficients is general in terms of its influence (either increase or decreasing a likelihood) but specific in its outcome comparisons (examining how general likelihoods change on a pair-wise basis only).

Because multinomial logit models are log transformed, they share several of logit’s (and ordinal logit’s) general assumptions (problem with multicollinearity and the omitted variable bias apply similarly to

multinomial logit as they did in logit and ordinal logit). Five central assumptions of multinomial logit are:

6. The logit of  $\Pr(Y_i = 1, 2, 3 \dots)$  depends on the logistically transformed values of ALL explanatory variables through the properly specified linear function  $\beta_1 X_1 + \dots + \beta_n X_n$  (similarly to logit, this assumes that the model does not suffer from an omitted variable bias).
7.  $\text{Logit}[\Pr(Y_i = 1, 2, 3 \dots)]$  possesses a cumulative standard logit distribution (hence correcting the “unbounded” problem associated with linear probability models). Categorical outcomes for the dependent variable must be mutually exclusive.
8. The odds of witnessing one categorical outcome versus another are not influenced by the introduction/removal of an alternative category. This is known as the independence of irrelevant alternatives assumption (or IIA)
9. No severe under- or over-representation of a dependent variable outcome (more likely to be fulfilled if there are fewer categories)
10. No (perfect) multicollinearity

Multinomial logit’s first assumption implies that the model is non-linear. Hence, the interpretation of beta coefficients should NOT be treated in a similar manner as OLS; the influence of  $X_1$  on the likelihood of a  $y=1, 2, 3 \dots$  outcome depends on the levels of all other independent variables.<sup>28</sup> We begin the lab with a general presentation of multinomial logit output and how to interpret it. We will utilize data from the 2006 General Social Survey (GSS). For this lab, we are going to concentrate on explaining the influence of individual characteristics (age, sex, race, income and education level) as well as personal opinions on wealth redistribution, on an individual’s political party (our dependent variable, which you’ll note is a non-ranked categorical variable). The dataset you are provided with includes the following variables:

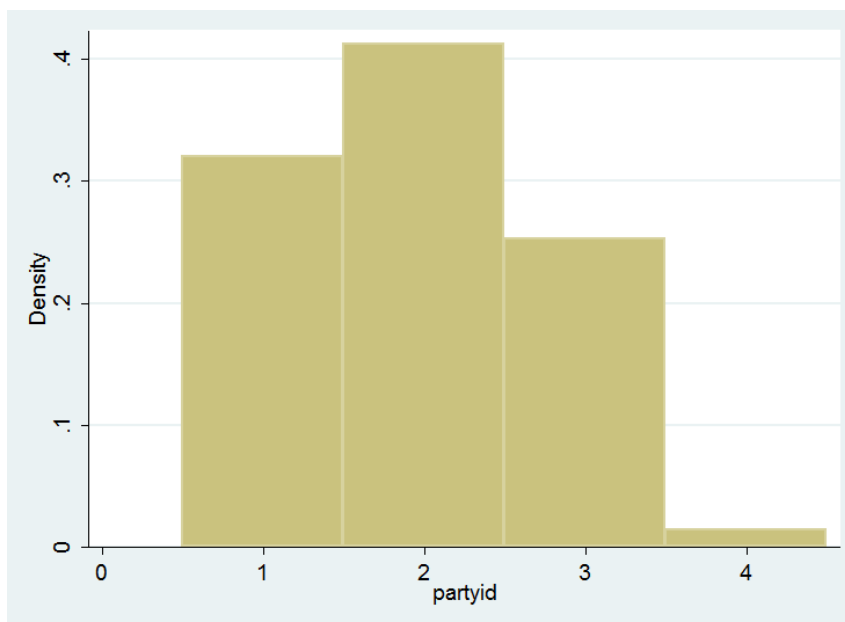
- **partyid**: The respondent’s self-reported political party: 1 for Democrat; 2 for Independent; 3 for Republican; and 4 for Other
- **female**: A dummy variable embodying a value of 1 if the respondent is a female, 0 if otherwise
- **white**: A dummy variable embodying a value of 1 if the respondent is white, 0 if otherwise
- **age**: The respondent’s age in years
- **realincome**: A respondent’s real income at the time of the survey
- **eqwlth**: A scalar variable measuring whether an individual believes that the government should attempt to reduce the income differences between rich and poor (**lower values** indicate a respondent strongly believes the government should engage in income distribution, while **higher values** indicate a respondent strongly believe the government should NOT engage in income distribution)
- **A series of educational dummies**: **HSdiploma**, which embodies a coding of 1 if the respondent’s highest level of education is a high school diploma, 0 if otherwise; **AssocDeg**, which embodies a coding of 1 if the respondent’s highest level of education is an Associate’s Degree, 0 if otherwise; **BachDeg**, which embodies a coding of 1 if a respondent’s highest level of education is a

---

<sup>28</sup> Similar problems with interaction terms and quadratic terms in logit and ordinal logit also apply to multinomial logit.

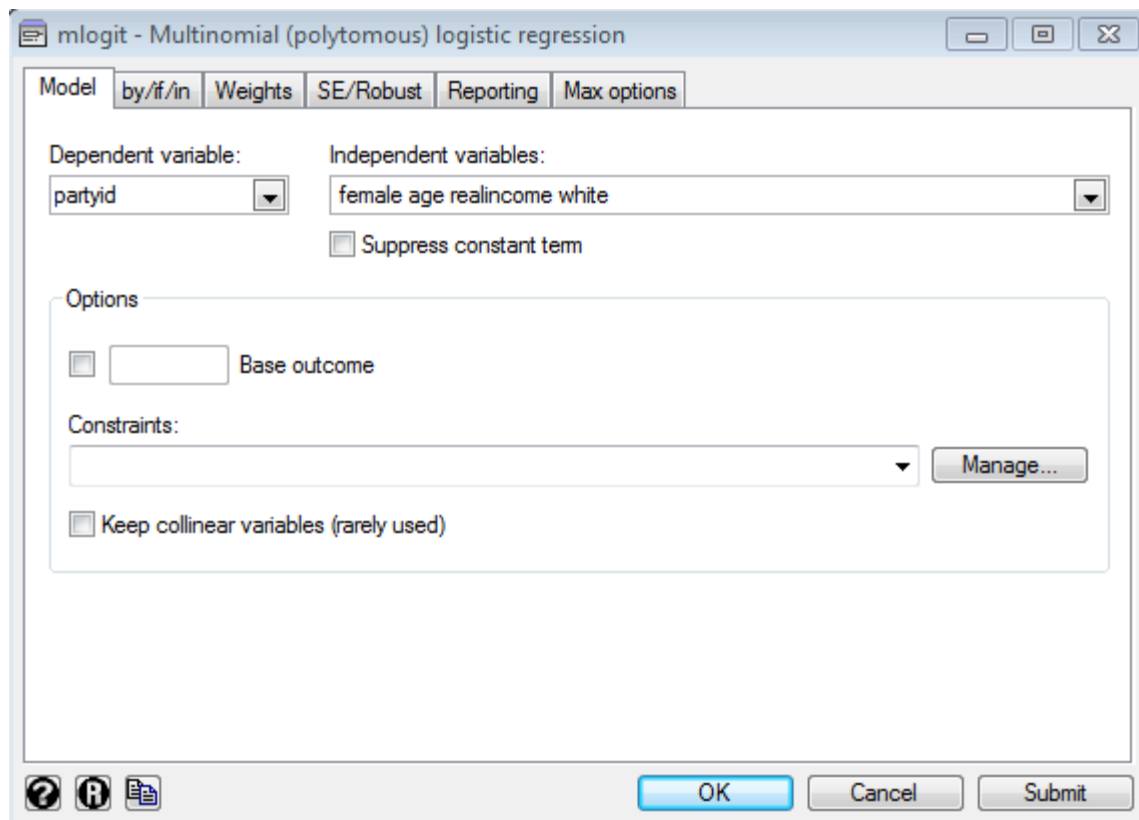
Bachelor's degree, 0 if otherwise, and; **GradDeg**, which embodies a coding of 1 if the respondent's highest level of education is either a Masters or Doctoral degree, 0 if otherwise (note, the "**Less than High School**" education category is not included within the dummy variables, and hence serves as the reference category).

Before we begin with a multinomial logistic regression, let's examine how our data is distributed across our dependent variable outcomes. Assumption IV of multinomial logit requires that the distribution of our data between the outcome categories should be relatively even. If one category is extremely under-represented, that may skew its pair-wise results to non-significance, as we are not able to pick up sufficient variation within our model. In order to examine the distribution of our four political party outcomes, create a histogram for discrete data. You can do this by typing the following command into the STATA command box: `"histogram partyid, discrete"`. You should be presented with the following histogram:



There are two things you will notice with this histogram. One, the "Other" category is extremely under-represented compared to the other political parties. This may lead to non-significant results for this category given the lack of data availability. Secondly, notice that most respondents identify as independents (category 2). For multinomial logistic regression, STATA automatically chooses the most represented category as the baseline. By default, STATA will choose the "Independent" category as our baseline for our regressions below.

To conduct a multinomial logistic regression model, click on the "Statistics" tab at the top of the STATA window. In the drop-down box, click "Categorical Outcomes". You should obtain another drop-down tab; click on "Multinomial Logistic Regression". You should see the following box:



Select *partyid* as your dependent variable and *female*, *age*, *realincome* and *white* as your primary independent variables. Click “Ok”, you should see the following output:

```
. mlogit partyid female age realincome white
```

```
Iteration 0:  log likelihood = -5106.2864
Iteration 1:  log likelihood = -4824.1855
Iteration 2:  log likelihood = -4811.9675
Iteration 3:  log likelihood = -4811.7321
Iteration 4:  log likelihood = -4811.7319
```

```
Multinomial logistic regression
```

```
Log likelihood = -4811.7319
```

```
Number of obs   =      4484
LR chi2(12)     =      589.11
Prob > chi2     =      0.0000
Pseudo R2      =      0.0577
```

	partyid	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
1	female	.3398251	.0732273	4.64	0.000	.1963022 .4833481
	age	.02007	.0021721	9.24	0.000	.0158127 .0243272
	realincome	5.59e-06	1.29e-06	4.35	0.000	3.07e-06 8.11e-06
	white	-.697431	.0788137	-8.85	0.000	-.8519029 -.542959
	_cons	-1.076264	.1214412	-8.86	0.000	-1.314285 -.838244
3	female	.018347	.0784321	0.23	0.815	-.1353771 .1720711
	age	.0126519	.0023212	5.45	0.000	.0081024 .0172014
	realincome	.0000106	1.24e-06	8.51	0.000	8.12e-06 .000013
	white	1.300713	.1211247	10.74	0.000	1.063313 1.538113
	_cons	-2.489897	.1574104	-15.82	0.000	-2.798416 -2.181379
4	female	-.7547588	.2679439	-2.82	0.005	-1.279919 -.2295984
	age	-.0014001	.0078025	-0.18	0.858	-.0166927 .0138925
	realincome	8.16e-06	3.60e-06	2.27	0.023	1.10e-06 .0000152
	white	1.516307	.4730855	3.21	0.001	.589076 2.443537
	_cons	-4.461373	.5598139	-7.97	0.000	-5.558588 -3.364158

(partyid==2 is the base outcome)

**CONGRATULATIONS! You have just conducted a multinomial logistic regression in STATA!**

Notice that multinomial logit output is very different from ordinal logit and logit output – the most obvious difference is that rather than obtaining one model, you are presented with (J-1), J being the number of outcome categories you have. There are some similarities, however, between multinomial logit and ordinal and standard logistic regression. The LR chi-squared statistic of joint significance operates in the same manner; if the statistic is highly significant, we can reject the null hypothesis that the overall model is insignificant (i.e. the log-odds values of our beta coefficients are all equal to zero). The second similarity between multinomial logit and other logistic models is that they all use maximum likelihood iteration techniques. Unlike OLS where the sum of squared residuals are minimized in order to obtain the final model, STATA estimates all types of logistic models via multiple iterations until the lowest (absolute value) log likelihood is produced. The final similarity relates to the beta coefficients. Like logit and ordinal logit, multinomial logit's beta coefficients are expressed in terms of log odds (hence you should NOT express your results in terms of marginal changes as you did in OLS). However, unlike logit and ordinal logit, these log odds are *relative*. In other words, the beta coefficients for outcome category 1 (Democrat) can ONLY be interpreted relative to the baseline category (Independent). We can NOT interpret the beta coefficients for Democrats relative to Republicans (category 3) unless we change the baseline category. To do that, go back to the multinomial regression box (see output below),

and tick the “options” box. Specify that you want to select the Republican category (coding of 3) as the baseline.

mlogit - Multinomial (polytomous) logistic regression

Model by/if/in Weights SE/Robust Reporting Max options

Dependent variable: partyid

Independent variables: female age realincome white

☐ Suppress constant term

Options

☒ 3 Base outcome

Constraints:

☐ Keep collinear variables (rarely used)

OK Cancel Submit

Click “Ok” and you should receive the following output:

```
. mlogit partyid female age realincome white, baseoutcome(3)

Iteration 0:  log likelihood = -5106.2864
Iteration 1:  log likelihood = -4824.1855
Iteration 2:  log likelihood = -4811.9675
Iteration 3:  log likelihood = -4811.7321
Iteration 4:  log likelihood = -4811.7319

Multinomial logistic regression               Number of obs   =       4484
                                                LR chi2(12)      =       589.11
                                                Prob > chi2      =       0.0000
Log likelihood = -4811.7319                   Pseudo R2       =       0.0577
```

partyid		Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
1	female	.3214781	.0845494	3.80	0.000	.1557643 .4871919
	age	.0074181	.0024556	3.02	0.003	.0026051 .0122311
	realincome	-4.97e-06	1.24e-06	-4.00	0.000	-7.40e-06 -2.53e-06
	white	-1.998144	.1228279	-16.27	0.000	-2.238882 -1.757406
	_cons	1.413633	.1646431	8.59	0.000	1.090938 1.736327
2	female	-.018347	.0784321	-0.23	0.815	-.1720711 .1353771
	age	-.0126519	.0023212	-5.45	0.000	-.0172014 -.0081024
	realincome	-.0000106	1.24e-06	-8.51	0.000	-.000013 -8.12e-06
	white	-1.300713	.1211247	-10.74	0.000	-1.538113 -1.063313
	_cons	2.489897	.1574104	15.82	0.000	2.181379 2.798416
4	female	-.7731057	.2701584	-2.86	0.004	-1.302606 -.243605
	age	-.014052	.0078614	-1.79	0.074	-.02946 .001356
	realincome	-2.40e-06	3.56e-06	-0.67	0.502	-9.38e-06 4.59e-06
	white	.2155936	.4824973	0.45	0.655	-.7300837 1.161271
	_cons	-1.971476	.5701685	-3.46	0.001	-3.088986 -.8539661

(partyid==3 is the base outcome)

**CONGRATULATIONS! You have just conducted a multinomial logistic regression with a different baseline category specification in STATA!** Notice from the output above that because we have changed the outcome category from “Independent” to “Republican” the regression output for the Democratic category (coding of 1) and “Other” category (coding of 4) exhibit different beta coefficients and are associated with different levels of significance. This is to be expected; because beta coefficients for multinomial logistic regression are relative to a baseline category, if the baseline category changes, the entire output changes. The only output which bears some semblance to the former regression output is that for the “Independent” category (coding of 2). Notice that the beta coefficients are of the same magnitude and significance but hold the opposite sign of the “Republican” category in the output from the first regression. This is also to be expected, because we have merely inverted our baseline categories between the two regressions. Rather than examining the Republican category relative to the Independent baseline, we have changed our base outcome to examine the Independent category relative to a Republican baseline.

#### **STATA COMMAND 14.1:**

*Code:* “**mlogit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

*Output produced:* Generates multinomial logistical regression output, where beta coefficients of independent variables are expressed in terms of relative log-odds. Note: the baseline outcome category will be the category with the most observations.

#### **STATA COMMAND 14.2:**

*Code:* “**mlogit var1 var2 var3 ..., baseoutcome(#)**”, where var1 is your dependent variable, var2, var3, ... are your independent variables, and # is the baseline outcome category designated by the researcher.

*Output produced:* Generates multinomial logistical regression output with a specified baseline outcome category.

Multinomial logit differs in two important respects from ordinal logit and logit. First, its beta coefficients can NOT be interpreted across all outcome categories, but rather can ONLY be interpreted between two categories at a time. In other words, the reporting of your results (like dummy variables) is CONTINGENT upon a baseline outcome category. Taking age as the independent variable of interest from the regression output above where “Republican” is the baseline outcome category, notice that it takes on three separate values; 0.0074 for the “Democrat vs Republican” category, -0.0127 for the “Independent vs Republican” category, and -0.0141 for the “Other vs Republican” category (all are significant on a confidence level of 90% or higher). Because we are given three different beta coefficients for age, we have three different results interpretation for the same variable! This is very different from ordinal logit or logit, where we could make specific (for logit) or more general (for ordinal logit) interpretations for how an independent variable impacted the dependent variable across the dependent variable’s entire range. For multinomial logit, we are restricted to making pairwise interpretations only. Given that the beta coefficients are in terms of relative log odds, we would have to (generally) interpret age’s influence on party affiliation as follows: *as age increases, the odds of an individual being a*

*Democrat increases relative to a Republican; as age increases, the likelihood of an individual being an Independent decreases relative to a Republican, and; as age increases, the odds of an individual being affiliated to an “Other” political party decreases relative to the Republican party.* Like ordinal logit and logit, we interpreted the log odds beta coefficients in a general matter (i.e. increases or decreases the odds/likelihood...), however, we had to do so in a relative manner (only comparing the influence of an independent variable on the likelihood of observing one categorical outcome relative to a baseline).

The second major difference between multinomial logit and other forms of logit is the transfer of beta significance across outcome categories. We witnessed some problems with this in the ordinal logit lesson, but for multinomial logit, problems associated with the transferability of independent variable significance across categories are more obvious. Examining the output above (where “Republican” is selected as the baseline category), notice that the female dummy is significant when comparing the Democratic category to the Republican category and the “Other” category to the Republican category, but it is not significant when comparing the Independent category to the Republican category. How do we determine whether a person’s gender has a significant influence on political party affiliation as a whole? The bad news is we cannot determine this from multinomial logit output on its own. We need to run a post-estimation test of significance. We will learn two in this lesson, the Wald test and the Likelihood Ratio test. Both are post-estimation commands that are contingent upon previous regression results.

A Wald test is the more simplistic test of independent variable significance. It examines the significance of an independent variable across all pair-wise comparisons (so from the output above, it would test the significance of the female dummy across the Democrat vs Republican output, the Independent vs Republican output, and the “Other” vs Republican output). One disadvantage of the Wald test is that it can only test significance for one variable at a time (also, its results are contingent on the baseline outcome category). However, if data is missing and the multinomial logistic model is complex in its independent variable specification, the Wald test is more likely than the LR test to produce a test statistic. Let’s use the multinomial logistic model from above (where Republican was the baseline category) and test the joint significance of the female dummy. To do, run the regression first, and then immediately after type the following syntax into the STATA command box: “test female”. You should obtain the following output:

1	female	.3214781	.0845494	3.80	0.000	.1
	age	.0074181	.0024556	3.02	0.003	.0
	realincome	-4.97e-06	1.24e-06	-4.00	0.000	-7.
	white	-1.998144	.1228279	-16.27	0.000	-2.
	_cons	1.413633	.1646431	8.59	0.000	1.
2	female	-.018347	.0784321	-0.23	0.815	-.1
	age	-.0126519	.0023212	-5.45	0.000	-.0
	realincome	-.0000106	1.24e-06	-8.51	0.000	-.1
	white	-1.300713	.1211247	-10.74	0.000	-1.
	_cons	2.489897	.1574104	15.82	0.000	2.
4	female	-.7731057	.2701584	-2.86	0.004	-1.
	age	-.014052	.0078614	-1.79	0.074	-.1
	realincome	-2.40e-06	3.56e-06	-0.67	0.502	-9.
	white	.2155936	.4824973	0.45	0.655	-.7
	_cons	-1.971476	.5701685	-3.46	0.001	-3.
(partyid==3 is the base outcome)						
. test female						
( 1)	[1]female = 0					
( 2)	[2]female = 0					
( 3)	[4]female = 0					
	chi2( 3) =	34.54				
	Prob > chi2 =	0.0000				

**CONGRATULATIONS!** You have just tested whether an independent variable is significant in a multinomial logistic regression using a Wald test! Because the Wald test produces a significant Chi-squared statistics (whose p-value is less than 0.10), we can reject the null hypothesis that gender has no effect on party affiliation.

#### STATA COMMAND 14.3:

*Code:* “**mlogit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**“test var1”**

*Output produced:* Conducts a Wald test examining whether an independent variable is significant across most/all dependent variable categorical outcomes in multinomial logistic regression.

A more sophisticated test for examining the significance of independent variables is the likelihood-ratio test (the LR test). The LR test of significance is more advantageous than a Wald test for two reasons: 1.) it tests the significance of all independent variables across all possible pair-wise combinations of the dependent variables (hence, you do not need to conduct multiple tests for individual independent

variables like you would do with the Wald test), and; 2.) it can conduct nested F-tests for multiple independent variables (i.e. for independent variables that are multi-categorical dummy variables). To demonstrate both of these strengths, let's produce a new multinomial logistic regression which inserts our (n-1) degree dummies. Type the following syntax into the STATA command box: *"mlogit partyid female age realincome white HSdiploma AssocDeg BachDeg GradDeg"* (note, "No High School Diploma" is the baseline degree category, and since we have not identified a baseline outcome category, STATA will resort to the category with the most outcomes – "Independent"). You should be presented with the following output:

```
. mlogit partyid female age realincome white HSdiploma AssocDeg BachDeg GradDeg
```

```
Iteration 0:  log likelihood = -5106.2864
Iteration 1:  log likelihood = -4777.2133
Iteration 2:  log likelihood = -4763.7766
Iteration 3:  log likelihood = -4763.5009
Iteration 4:  log likelihood = -4763.5005
```

Multinomial logistic regression

Log likelihood = -4763.5005

Number of obs = 4484  
 LR chi2(24) = 685.57  
 Prob > chi2 = 0.0000  
 Pseudo R2 = 0.0671

partyid		Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
1	female	.3290186	.0738127	4.46	0.000	.1843483 .4736889
	age	.0212618	.0022116	9.61	0.000	.0169271 .0255965
	realincome	2.86e-06	1.37e-06	2.09	0.036	1.84e-07 5.53e-06
	white	-.7974786	.0812706	-9.81	0.000	-.9567661 -.638191
	HSdiploma	.5331157	.1054224	5.06	0.000	.3264916 .7397399
	AssocDeg	.3443013	.1582085	2.18	0.030	.0342183 .6543843
	BachDeg	.7457054	.1371947	5.44	0.000	.4768087 1.014602
	GradDeg	.8614675	.1568554	5.49	0.000	.5540366 1.168898
	_cons	-1.478524	.1466215	-10.08	0.000	-1.765897 -1.191151
3	female	-.0037504	.0791682	-0.05	0.962	-.1589172 .1514164
	age	.0148647	.0023688	6.28	0.000	.0102219 .0195074
	realincome	8.61e-06	1.31e-06	6.58	0.000	6.05e-06 .0000112
	white	1.196983	.1228402	9.74	0.000	.9562206 1.437745
	HSdiploma	.5982192	.1322963	4.52	0.000	.3389232 .8575151
	AssocDeg	.6318526	.1784373	3.54	0.000	.282122 .9815833
	BachDeg	1.069322	.1551854	6.89	0.000	.7651639 1.37348
	GradDeg	.3473834	.187187	1.86	0.063	-.0194963 .7142632
	_cons	-3.027081	.1915013	-15.81	0.000	-3.402417 -2.651745
4	female	-.7589224	.268556	-2.83	0.005	-1.285282 -.2325623
	age	-.0004159	.0077999	-0.05	0.957	-.0157035 .0148717
	realincome	7.27e-06	3.82e-06	1.90	0.057	-2.25e-07 .0000148
	white	1.505982	.4777024	3.15	0.002	.5697026 2.442262
	HSdiploma	.0243456	.3921783	0.06	0.951	-.7443098 .793001
	AssocDeg	-.5714197	.6831221	-0.84	0.403	-1.910314 .767475
	BachDeg	.6132433	.4436282	1.38	0.167	-.256252 1.482738
	GradDeg	-.8051818	.6976657	-1.15	0.248	-2.172581 .5622177
	_cons	-4.513441	.6263959	-7.21	0.000	-5.741154 -3.285727

(partyid==2 is the base outcome)

In order to conduct an LR test of variable significance, type in “mlogtest, lr” in the STATA command box. You should be presented with the following output:

```

      _cons      -4.513441      .6263959      -7.21      0.000      -5.741154
(partyid==2 is the base outcome)
. mlogtest, lr
Problem determining number of categories.
**** Likelihood-ratio tests for independent variables
Ho: All coefficients associated with given variable(s) are 0.

```

partyid	chi2	df	P>chi2
female	34.362	3	0.000
age	104.490	3	0.000
realincome	46.226	3	0.000
white	348.916	3	0.000
HSdiploma	36.762	3	0.000
AssocDeg	15.497	3	0.001
BachDeg	59.719	3	0.000
GradDeg	33.520	3	0.000

**CONGRATULATIONS!** You have just tested whether multiple independent variables are significant in a multinomial logistic regression using a likelihood ratio test! Notice that the LR test has produced test statistics for all independent variables. Rather than testing the significance of one at a time, we can examine each independent variable’s significance in all tests. For our output above, the LR test suggests that all independent variables significantly influence political party affiliation.

We can also conduct a nested F-test, testing whether multiple independent variables are jointly significantly different from 0, in multinomial logit using the mlogtest command. Recall in OLS that we used the “test” post-estimation command to test whether two (or more) independent variables within a regression were significantly different than zero. Nested F-tests are especially convenient to test the significance of an independent variable that is manifested in multiple dummies (i.e. dummies representing multiple race categories, regional categories, or in our case, multiple educational categories). Say we were interested in testing whether education on its own significantly predicted a person’s political affiliation; in other words, we want to test whether *HSdiploma*, *AssocDeg*, *BachDeg*, and *GradDeg* are jointly significantly different from zero. To do this using the mlogtest, modify the syntax we previously used by typing the following in the STATA command box: “mlogtest, lr set(*HSdiploma AssocDeg BachDeg GradDeg*)”. You should obtain the following output:

```

. mlogtest, lr set(HSdiploma AssocDeg BachDeg GradDeg)
Problem determining number of categories.
*** Likelihood-ratio tests for independent variables
Ho: All coefficients associated with given variable(s) are 0.

```

partyid	chi2	df	P>chi2
female	34.362	3	0.000
age	104.490	3	0.000
realincome	46.226	3	0.000
white	348.916	3	0.000
HSdiploma	36.762	3	0.000
AssocDeg	15.497	3	0.001
BachDeg	59.719	3	0.000
GradDeg	33.520	3	0.000
set_1: HSdiploma AssocDeg BachDeg GradDeg	96.463	12	0.000

**CONGRATULATIONS! You have just conducted a joint F-test of significance in STATA!** Notice that STATA produces the same significance output above, identifying the associated significance test-statistic for each independent variable in isolation. However, contrary to the output previously, it produces a new “set” output; this is the result from the joint significance test of our educational dummies.

#### STATA COMMAND 14.4:

*Code:* “**mlogit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**“mlogtest, lr”**

*Output produced:* Conducts a likelihood ratio test examining whether independent variables are significant across most/all dependent variable categorical outcomes in multinomial logistic regression.

#### STATA COMMAND 14.5:

*Code:* “**mlogit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**“mlogtest, lr set(var2 var3 var4)”**

*Output produced:* Conducts nested F-test, examining the joint significance of the specified variables (in this case var2 var3, and var4).

Like logit, beta coefficients in log odds cannot tell a researcher anything meaningful in terms of magnitude; it can only tell the researcher if the odds or probability of observing an outcome relative to another increases or decreases. In order to obtain a more meaningful we can rely upon factor changes. Factor changes operate like odds ratios, but in multinomial logit, they tell us how a change in an independent variable would impact the relative odds of witnessing one outcome category compared to another; they do NOT tell us the change in odds of remaining in one, and only one, category, as odds ratios in logistic regression do. Factor changes operate very similar to odds ratios, in that they revolve around a value of one, and are always greater than zero. Positive log odds beta coefficients are associated with factor changes that are greater than one. If the factor change is greater than 1 (say, 3.2), we would express the influence of  $X_1$  in the following manner: for a one unit change in  $X_1$ , the odds of  $y_{j1}$  relative to  $y_{j2}$ , will increase by a factor of 3.2. Negative log odds beta coefficients are associated with factor changes that are less than one. If the factor change is less than 1 (say, 0.2), we would express the influence of  $X_1$  in the following manner: for a one unit change in  $X_1$ , the odds of  $y_{j1}$  relative to  $y_{j2}$ , will increase by a factor of 0.2 (or a relative odds reduction of 80%)

Unlike for logit and ordinal logit, where we STATA computes odds ratios for all variables via the “, or” addition, STATA can only compute factor changes in multinomial logit for one independent variable at a time. The “listcoef” command is a post-estimation command which will provide factor changes for all pair-wise comparisons of the outcome variable. Let’s calculate the factor changes for the white racial dummy by first running the following multinomial logistic regression: “mlogit partyid female age realincome white”. Immediately after this regression, type the following syntax into the STATA editor: “listcoef white”. You should obtain the following output:

```
. listcoef white
```

mlogit (N=4484): Factor Change in the Odds of partyid

Variable: white (sd=.44462168)

Odds comparing Group 1 vs Group 2		b	z	P> z	e^b	e^bstdx
1	-2	-1.99814	-16.268	0.000	0.1356	0.4113
1	-3	-2.21374	-4.671	0.000	0.1093	0.3737
1	-4	-0.69743	-8.849	0.000	0.4979	0.7334
2	-1	1.99814	16.268	0.000	7.3754	2.4313
2	-3	-0.21559	-0.447	0.655	0.8061	0.9086
2	-4	1.30071	10.739	0.000	3.6719	1.7830
3	-1	2.21374	4.671	0.000	9.1498	2.6759
3	-2	0.21559	0.447	0.655	1.2406	1.1006
3	-4	1.51631	3.205	0.001	4.5554	1.9624
4	-1	0.69743	8.849	0.000	2.0086	1.3636
4	-2	-1.30071	-10.739	0.000	0.2723	0.5608
4	-3	-1.51631	-3.205	0.001	0.2195	0.5096

**CONGRATULATIONS! You have just calculated factor changes in relative odds for multinomial logistic regression in STATA!** The “listcoef” command provides six different columns. The first tells the researcher which outcome categories are being compared. The second presents the raw beta

coefficients (expressed as the relative log odds). The third column provides the z-statistic associated with this beta coefficient for the pair-wise comparison, and the fourth presents the corresponding p-value. The fifth column (highlighted in yellow) presents the factor change in relative odds (the sixth column presents the factor change expressed in terms of standard deviations, we will focus exclusively on column five for this lesson). We interpret the factor changes presented in column 5 in the following manner (note, I only present three interpretations below, but the others would be similar):

- If an individual is white, his/her odds of being a Democrat (coding of 1) relative to an Independent (coding of 2) will increase by a factor of 0.1356 (or an 86.44% reduction in relative odds) compared to if he/she was a minority.
- If an individual is white, his/her odds of being a Republican (coding of 3) relative to a Democrat (coding of 1) will increase by a factor of 9.1498 compared to if he/she was a minority.
- If an individual is white, as opposed to being a minority, his/her odds of belonging to an “Other” political party (coding of 4) relative to being an Independent (coding of 2) will increase by a factor of 0.2723 (or a 72.77% reduction in relative odds).

#### STATA COMMAND 14.6:

*Code:* “**mlogit var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**“listcoef, var1”**

*Output produced:* Calculates the factor changes associated with an independent variable for all pair-wise combinations of the outcome categories.

We can also calculate predicted probabilities of our categorical outcome variables in multinomial logistic regression via the “prvalue” and “prgen” commands.<sup>29</sup> Predicted probabilities are very convenient in multinomial logit as they do not require pair-wise comparisons of two outcome categories. Rather, we can see how a one unit change in an independent variable influences the probability of being in one outcome category without the consideration of a baseline outcome category. As we discussed in earlier lessons in logit and ordinal logit, “prvalue” computes the predicted probability of obtaining a categorical outcome for a specific value of an independent variable. It is especially convenient in the examining the influence of dummy variables on the probability of witnessing a certain categorical outcomes, as these types of independent variables can only take on two values. Let’s use the “prvalue” command to examine how gender influences an individual’s probability of belonging to any of the four political party affiliations. Like ordinal logit, if you want to determine how an independent variable changes the probability of witnessing a dependent variable outcome in multinomial logit, you should NEVER interpret probability output for one value of  $X_1$  in isolation. Rather, multinomial logistic output should be compared and interpreted ACROSS MULTIPLE values of  $X_1$ . We can do this via the “save” and “diff” syntax modifications of the “prvalue” command, just like we did in the ordinal logistic lesson. Use the

<sup>29</sup> Note: These commands may not work for multinomial logit in earlier versions of STATA (version 9 or 10). They work in versions 11 and higher.

same multinomial logistic regression as above (“mlogit partyid female age realincome white”). In the STATA command box, enter “prvalue, x(female=0) save” after the regression output, and then “prvalue, x(female=1) diff”. You should be presented with the following output:

```
. prvalue, x(female=0) save
mlogit: Predictions for partyid
Confidence intervals by delta method
```

		95% Conf. Interval
Pr(y=1 x):	0.2838	[ 0.2629, 0.3046]
Pr(y=3 x):	0.2398	[ 0.2197, 0.2600]
Pr(y=4 x):	0.0188	[ 0.0122, 0.0254]
Pr(y=2 x):	0.4576	[ 0.4346, 0.4806]

```

x=      white      female      age      realincome
    .72881356          0      47.29438      28189.124

. prvalue, x(female=1) diff
mlogit: Change in Predictions for partyid
Confidence intervals by delta method
```

	Current	Saved	Change	95% CI for Change
Pr(y=1 x):	0.3593	0.2838	0.0756	[ 0.0471, 0.1040]
Pr(y=3 x):	0.2202	0.2398	-0.0196	[-0.0448, 0.0055]
Pr(y=4 x):	0.0080	0.0188	-0.0108	[-0.0177, -0.0040]
Pr(y=2 x):	0.4125	0.4576	-0.0451	[-0.0752, -0.0150]

```

Current=      white      female      age      realincome
         .72881356          1      47.29438      28189.124
Saved=      .72881356          0      47.29438      28189.124
Diff=          0          1          0          0
.
```

**CONGRATULATIONS! You have just conducted an assessment of fitted probabilities for multinomial logit in STATA!** The first “prvalue” command provides predicted probabilities of men in terms of their political affiliations (highlighted in yellow). Men are most likely to identify as “Independents” (coding of 2). They are second most likely to identify as “Democrat” (coding of 1); you will note that this probability is significantly higher than identifying as “Republican” (coding of 3) as the 95% confidence intervals of the predicted probabilities do not overlap.

The change in predictions (the output provided after the “prvalue, ... diff” command), provides a better assessment for how gender influences the probability of belonging to a certain political party. Notice that for “current” output (i.e. output for females, highlighted in orange), there is a *significantly lower probability that women will belong an “Other” political party (coding of 4), or will classify as an “Independent” (coding of 2) relative to men.* We can assess this significance based upon the 95% confidence interval for the change in probabilities between men and women (far-right column,

highlighted in blue). If this column were straddling 0 (i.e. the upper and lower bound held alternative signs), we could NOT claim with significance that women were less likely to belong to a political party than men (this is the case for Republican party affiliation – coding of 3). Moreover, you may notice that women are significantly more likely to be Democrats (coding of 1) than men, as indicated by the fact that the change between a female coding of 0 and 1 is significantly different from zero.

We can examine how a continuous variable influences the probability of belong to a categorical outcome via the “prgen” command. As with ordinal logit, the “prgen” command not only tells us how changes in a continuous variable influence the probability of belonging to one outcome category in the isolation of others, but it also helps to determine whether significance is transitive across categories (in other words, it will help us detect whether  $X_1$  significantly increases/decreases the probability of a certain outcome across  $X_1$ ’s entire range). To demonstrate, let’s start by graphing the fitted probabilities of belonging to a certain political party across all values of age. Within this dataset, age ranges from 18 to 99 (you can verify this by using the “summarize” command). Using the “prgen” command, type the following syntax into the STATA command box IMMEDIATELY after the multinomial logit regression output above: “prgen age, from(18) to(99) generate(prAge) rest(mean) ci”. You should see the following added independent variables:

Variables			
Name	Label	Type	Form
graddeg	GradDeg	byte	%8.0c
prAgex		float	%9.0c
prAgep1	pr(1)	float	%9.0c
prAgep3	pr(3)	float	%9.0c
prAgep4	pr(4)	float	%9.0c
prAgep2	pr(2)	float	%9.0c
prAgep1lb	LB pr(1)	float	%9.0c
prAgep3lb	LB pr(3)	float	%9.0c
prAgep4lb	LB pr(4)	float	%9.0c
prAgep2lb	LB pr(2)	float	%9.0c
prAgep1ub	UB pr(1)	float	%9.0c
prAgep3ub	UB pr(3)	float	%9.0c
prAgep4ub	UB pr(4)	float	%9.0c
prAgep2ub	UB pr(2)	float	%9.0c
prAges1	pr(y<=1)	float	%9.0c
prAges3	pr(y<=3)	float	%9.0c
prAges4	pr(y<=4)	float	%9.0c

female	.018347	.0784321	0.23	0.815	-.1353771
age	.0126519	.0023212	5.45	0.000	.0081024
realincome	.0000106	1.24e-06	8.51	0.000	8.12e-06
_cons	-2.489897	.1574104	-15.82	0.000	-2.798416

white	1.516299	.4730888	3.21	0.001	.5890625
female	-.7547588	.267944	-2.82	0.005	-1.279919
age	-.0014001	.0078025	-0.18	0.858	-.0166927
realincome	8.16e-06	3.60e-06	2.27	0.023	1.10e-06
_cons	-4.461366	.5598167	-7.97	0.000	-5.558587

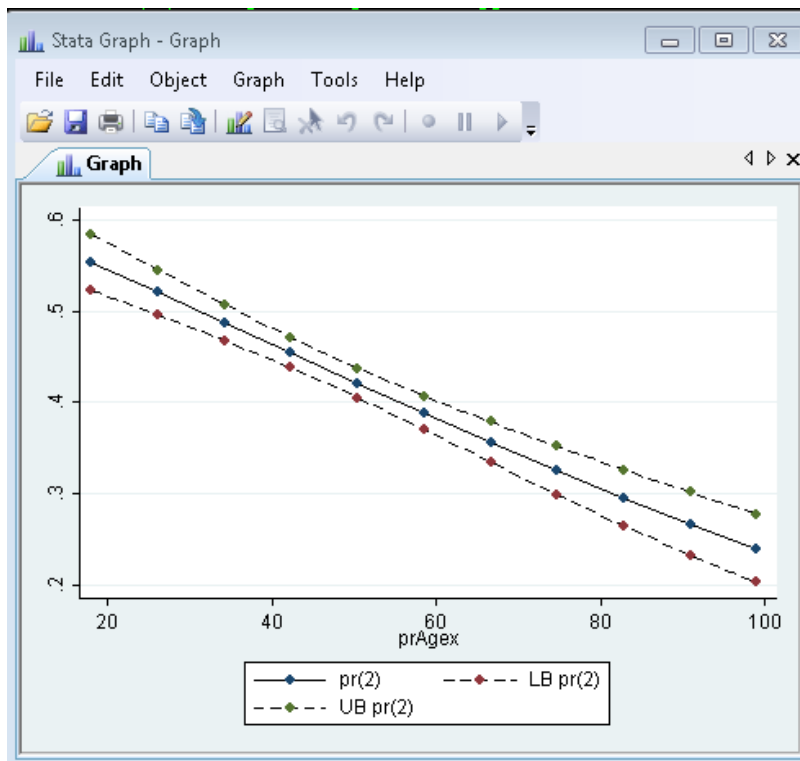
  

```

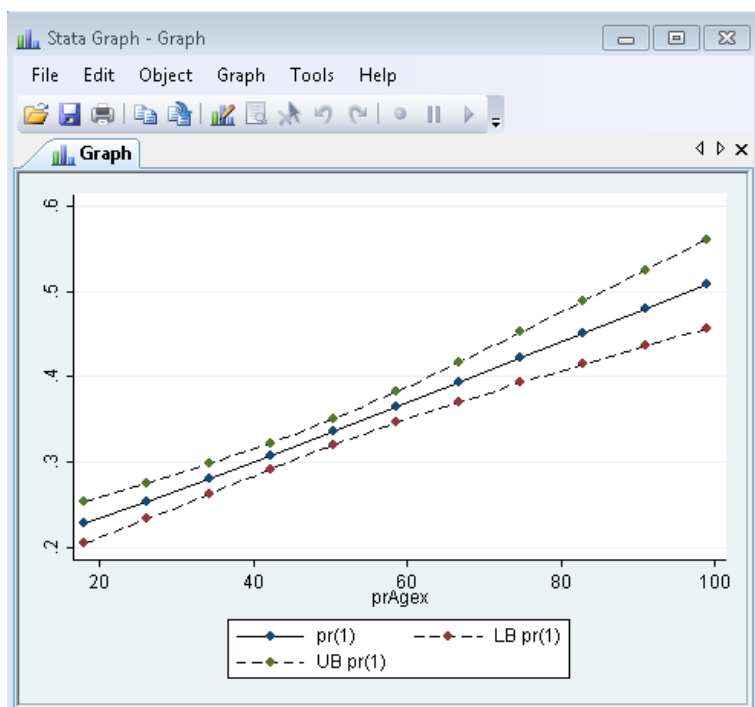
. prgen age, from(18) to(99) generate(prAge) ci
mlogit: Predicted values as age varies from 18 to 99.
x=   white   female   age   realincome
    .72881356   .55575379   47.29438   28189.124

```

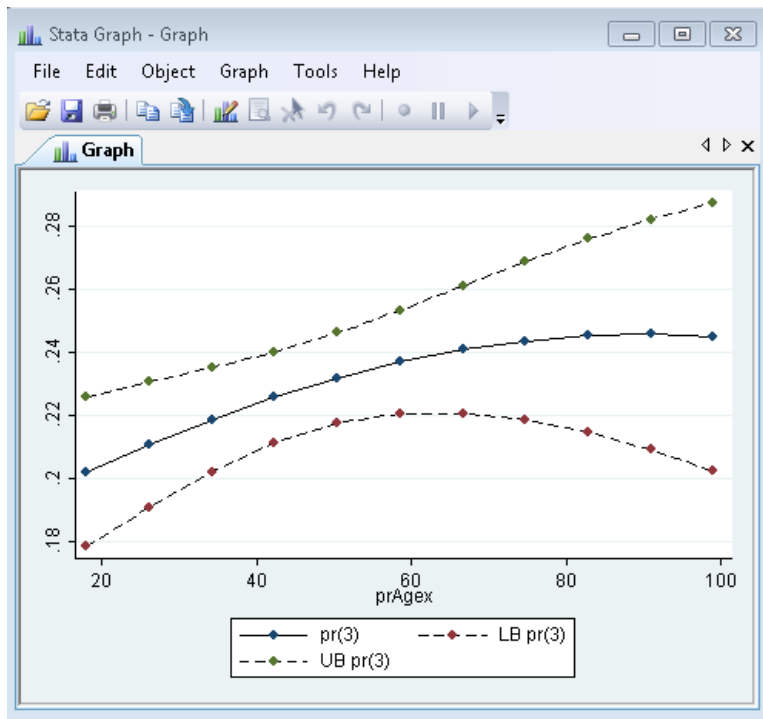
Like the previous logit and ordinal logit lessons, the “prgen” command has added new variables to our variable box. The probabilities of the four categories and their upper and lower bounds are represented, in accordance to the four political parties represented within the dependent variable coding. We can use the same syntax as we did for logit and ordinal logit to graph the fitted probabilities across all values of age. Let’s start with calculating the fitted probability of politically identifying as an “Independent” (coding of 2). Type the following code into the STATA command box: “graph twoway (connected prAgep2 prAgex, clcolor(black) clpat(solid)) (connected prAgep2ub prAgex, clcolor(black) clpat(dash)) (connected prAgep2lb prAgex, clcolor(black) clpat(dash))”. You should be presented with the following output:



Notice that as age increases, the probability of being an Independent decreases. Moreover, this decline is persistent across all values of age. Now change the above syntax to graph the fitted probability of an individual identifying as a “Democrat” (coding of 1) by typing the following code into the STATA command box: “graph twoway (connected prAgep1 prAgex, clcolor(black) clpat(solid)) (connected prAgeplub prAgex, clcolor(black) clpat(dash)) (connected prAgep1lb prAgex, clcolor(black) clpat(dash))”. You should be presented with the following output:



A surprising result emerges here; as individuals become older, they are more likely to identify as “Democrat”. As with the previous graphic, this dynamic is consistently significant across age. Let’s produce one more graphic, however, examining the influence of age on identifying as a “Republican” (coding of 3). Type in the following syntax into the STATA command editor: “graph twoway (connected prAgep3 prAge, clcolor(black) clpat(solid)) (connected prAgep3ub prAge, clcolor(black) clpat(dash)) (connected prAgep3lb prAge, clcolor(black) clpat(dash))”. You should see the following output:



Notice that though identifying as a Republican increases with age, this increase only happens prior to the age of 60. After 60, our confidence intervals significantly widen (we cannot distinguish a significant difference in an 80 year old being a Republican compared to a 40 year old). One factor which may be driving this is the distribution of the data across age. If older individuals are sparsely distributed within our “Republican” category, our confidence intervals within our graphics will likely widen, meaning that significance of age is NOT retained over its entire range for the “Republican” outcome category.

We end this lesson with the discussion of the independence of irrelevant alternatives (IIA) assumption. This assumption is crucial for multinomial logit; if it is violated, pair-wise beta results, and their associated significance levels, may be inaccurate. For multinomial logistic regression, we do not assume that outcomes can be ranked. However, once we determine the odds of witnessing one outcome versus another, IIA assumes that these odds will NOT change if we introduce an alternative outcome category. In terms of our analysis above, say we were conducting a multinomial logistic regression for three political affiliations; Democrats, Republicans and Libertarians. Suppose the odds that an individual was a Libertarian, relative to a Democrat, was 1 in 3 and the odds that an individual was a Libertarian relative to a Republican was 1 in 2. Under IIA, if we introduced a fourth political party affiliation (say “Green party”) these relative odds SHOULD NOT change; they should maintain their consistency regardless of the introduction of “irrelevant” outcomes.

One example of when IIA may be violated is in the case of strategic voting. Again, say two candidates are running in an election; a Green candidate with strong preferences for pro-environmental policies, and a Democratic candidate with milder preferences for pro-environmental policies. If a voter is concerned about environmental issues and must select between these two candidates, he/she will prefer the Green Party candidate. But suppose a Republican, who is largely apathetic to environmental policy, enters the race and is likely to win. Our environmentally concerned voter may switch his/her vote to the Democratic candidate, even though the candidate does not align completely with his/her environmental policy

preferences, in order to thwart the Republican's changes of winning. This example of strategic voting is a clear violation of IIA; in introducing another outcome, the individual has changed his/her preference. This violation of IIA will make the beta coefficients associated with an individual's (or group of individual's) characteristics inconsistent when changing baseline outcome categories.

We can test whether IIA is fulfilled within our multinomial logistic regression model via two tests: a Hausman test, which we will emphasize in this lab, and a Small-Hsiao test, which we will not emphasize given that its results are heavily contingent (and hence variable) upon how the data sample is divided into sub-samples for the test. A Hausman test of IIA first fits the regression model with all outcome categories, and then compares these results to another regression model with the same independent variable, but with one of the outcome categories dropped. If the regression results are not consistent between the two models, IIA is violated; if they are consistent, IIA is upheld. The IIA Hausman test, like the Brant test, is a post-estimation command and is unique to a regression's specification. Using the same regression as above (where political affiliation is explained by age, gender, real income, and the white racial dummy), type the following command immediately after your regression: "mlogtest, iia base".<sup>30</sup> You should be presented with the following output:

```

4
      female   -.7547588   .2679439   -2.82   0.005   -1.279919   -.22
      age      -.0014001   .0078025   -0.18   0.858   -.0166927   .01
    realincome   8.16e-06   3.60e-06    2.27   0.023   1.10e-06    .00
      white     1.516307   .4730855    3.21   0.001   .589076     2.4
      _cons    -4.461373   .5598139   -7.97   0.000   -5.558588   -3.3

(partyid==2 is the base outcome)

. mlogtest, iia base

Problem determining number of categories.

**** Hausman tests of IIA assumption

Ho: Odds(Outcome-J vs Outcome-K) are independent of other alternatives.

Omitted |      chi2   df   P>chi2   evidence
-----|-----
      1 |    -19.434    8    1.000   for Ho
      2 |   1252.804    8    0.000  against Ho
      3 |    -5.460    8    1.000   for Ho
      4 |    -0.381    8    1.000   for Ho

**** Small-Hsiao tests of IIA assumption

Ho: Odds(Outcome-J vs Outcome-K) are independent of other alternatives.
equation 2 not found
r(111);

```

<sup>30</sup> Newer versions of STATA may produce two Hausman tests if the asymptotic assumption required for Chi-squared tests are violated (i.e. one of the tails of the distribution of the Chi-Squared test statistic fails to converge to zero). If this is the case, use the test which produces corresponding p-values with the Chi-squared test statistics.

**CONGRATULATIONS! You have just conducted a Hausman test (and Small-Hsiao test) of the independence or irrelevant alternatives assumption in STATA!** The null hypothesis for a Hausman test is that the IIA assumption continues to remain fulfilled if the listed outcome category is removed from the regression. You'll notice that the associated Chi-squared statistic for the Democrat, Republican and "Other" political party affiliations are highly insignificant (and produce associated p-values that are well above 0.1). However, the Chi-squared statistic for the "Independent" political category is highly significant, violating the null hypothesis of IIA. This implies that when the "Independent" outcome category is removed from the regression, relative log odds results significantly change for the remaining pair-wise outcome combinations (in other words – the relative log odds for Democrats vs Republicans is significantly different when the "Independent" category is excluded, compared to when it is included). If only one outcome category violates the IIA assumption, then it is violated in full.

#### **STATA COMMAND 14.7:**

*Code:* **"*mlogit var1 var2 var3 ...*"**, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**"*mlogtest, iia base*"**

*Output produced:* Conducts a Hausman and Small-Hsiao test for the independent of irrelevant alternatives assumption in multinomial logit. If you are presented with a significant chi-squared statistic (i.e. with a p-value below 0.10) for at least one of your outcome categories, you must reject the null hypothesis that the assumption is satisfied.

There are several steps you can take when you find yourself presented with the IIA violation. Firstly, you should attempt to better specify your model and determine if this changes the test's results. Hausman test results are contingent on the model's specification. If you add new variables these results will change (although they may not completely eliminate the problem). To observe this, insert the educational dummies as controls into the multinomial logistic regression and re-conduct the test for IIA. You should obtain the following result:

```

      _cons      -3.027081   .1915013   -15.81    0.000   -3.402417   -2.651745
4      female      -.7589224   .268556    -2.83    0.005   -1.285282   -.2325623
      age        -.0004159   .0077999    -0.05    0.957   -.0157035   .0148717
      realincome    7.27e-06   3.82e-06     1.90    0.057   -2.25e-07   .0000148
      white        1.505982   .4777024     3.15    0.002   .5697026    2.442262
      hsdiploma     .0243456   .3921783     0.06    0.951   -.7443098    .793001
      assocdeg     -.5714197   .6831221    -0.84    0.403   -1.910314    .767475
      bachdeg       .6132433   .4436282     1.38    0.167   -.256252    1.482738
      graddeg      -.8051818   .6976657    -1.15    0.248   -2.172581    .5622177
      _cons      -4.513441   .6263959    -7.21    0.000   -5.741154   -3.285727

(partyid==2 is the base outcome)

. mlogtest, iia
Problem determining number of categories.

**** Hausman tests of IIA assumption
Ho: Odds(Outcome-J vs Outcome-K) are independent of other alternatives.

Omitted |      chi2   df   P>chi2   evidence
-----|-----
1        -23.873   16   1.000   for Ho
2        537.303   16   0.000   against Ho
3        -7.019   16   1.000   for Ho
4         0.308   16   1.000   for Ho

**** Small-Hsiao tests of IIA assumption
Ho: Odds(Outcome-J vs Outcome-K) are independent of other alternatives.
equation 2 not found
r(111);

```

Notice that while the Chi-squared statistic associated with excluding outcome 2 (“Independent”) is still significant, it’s value drops relative to the previous regression model. However, these added controls have not completely rectified the problem. If you discover this is the case, you can attempt to model your regression using multinomial probit (you can do this via the “mprobit” command). This regression model is more flexible in its approach to categorical analysis and does not rest on the restrictive assumption of IIA. However, multinomial probit is not without its problems. It is subject to estimation problems which may produce arbitrary or misleading results, and further problems can arise with heteroskedasticity. Given these empirical difficulties, some empirical modelers ignore the IIA assumption and continue to use multinomial logit (in their simulations on American and French electoral data, Dow and Endersby (2004) find that the violation of IIA may not skew results as badly as anticipated).<sup>31</sup>

<sup>31</sup> Dow, J. and Endersby, J. (2004) “Multinomial probit and multinomial logit: a comparison of choice models for voting research” *Electoral Studies*. Vol 23: 107-122.

---

### Practice Problems:

Practice Problem 1: Run a multinomial logit model which predicts the effect of the five following independent variables on an individual's political party affiliation: gender, age, income, (white) ethnicity and the degree to which they believe government should redistribute wealth (eqwlth). How would you broadly interpret the influence of eqwlth (note, you should have three pair-wise interpretations for your answer).

Practice Problem 2: Using the model from above, conduct an LR test to determine whether an individual's belief in wealth distribution influences their political affiliation. What do you discover? Are the other independent variable significant for the entirety of the model?

Practice Problem 3: Using the regression from Practice Problem 1, are race and gender jointly significant in influencing an individual's political affiliation? Make sure to provide proof with your answer.

Practice Problem 4: Using the model from Practice Problem 1, how would you interpret the influence of an individual's belief in wealth redistribution as a factor change for the following pair-wise outcome comparisons: being a Democrat compared to being a Republican; being an Independent compared to being a Democrat; belonging to an "other" political party relative to being a Republican?

Practice Problem 5: Using the "prvalue" command on the regression output above, for which political parties are whites significantly more likely to belong to relative to non-whites? Make sure to provide proof with your answer.

Practice Problem 6: Using the "prgen" command, explain the influence of the eqwlth scalar variable on the probability of belonging to the Democratic Party and the Republican Party. Does an individual's preferences for wealth distribution exhibit consistent significance across its entire range (1-7) for both party outcomes?

Practice Problem 7: What is the independence of irrelevant alternatives assumption and how does it influence results in multinomial logistic regression? Does the model used in Practice Problem 1 satisfy the independence of irrelevant alternatives assumption? Which categories are problematic? Do your results change when you add the four educational dummies to your model?

*Learning Objective 1: Understanding why Poisson and negative binomial regression are more optimal estimators for counts data than OLS.*

*Learning Objective 2: Estimating and interpreting the output of Poisson and negative binomial regression in STATA*

*Learning Objective 3: Testing the equi-dispersion assumption for Poisson regression via the alpha likelihood ratio test*

*Learning Objective 4: Understanding under what conditions negative binomial regression is a more ideal estimator than Poisson regression*

*Learning Objective 5: Comparing estimated and actual count outcomes of Poisson and negative binomial regression via the “prcounts” command*

The final lesson of this manual examines regression models used for count dependent variables. Counts, also known as frequencies, are integer data (i.e. their value has numerical meaning) but are discrete rather than continuous. Examples of dependent variables that are counts include: any type of events outcomes (i.e. the number of events that exist within a particular unit), the number of alcoholic units consumed by an individual within a week, the number of inmates in a prison, the number arrests that occur within a police jurisdiction, the number of riots that occur within a city, etc. Because counts are integer data, it is possible to use OLS as an estimator for these dependent variables. However, OLS runs into several problems with counts data. First, if the mean (or expected value) of the count variable is low - less than ten – OLS will produce biased results, namely the under-estimation of low count outcomes and the over-estimation of high count outcomes. Low count means are especially common if the dependent variable is rare, for example the occurrence of a war or a terrorist attack. Second, OLS commits similar problems with counts data as it does with binary dependent variables; results frequently violate the heteroskedasticity assumption, residuals do not conform to a normal distribution, and unfeasible (negative) count outcomes may be produced by the linear model. Because of these problems, researchers who analyze count or frequency data regularly resort to Poisson or negative binomial regression.

We begin discussion of count modeling with Poisson regression. Like logistic models, Poisson models are transformed in order to correct for problems with a standard linear model. Unlike logit, however, which is *log* transformed, Poisson (and negative binomial regression) models are *exponentially* transformed. The six central assumptions of Poisson regression are:

11. Observations are randomly sampled from a population where  $y_i$  and its residuals possess a Poisson distribution,  $\Pr(y_i | \mu) = \frac{e^{-\mu} \mu^{y_i}}{y_i!}$

---

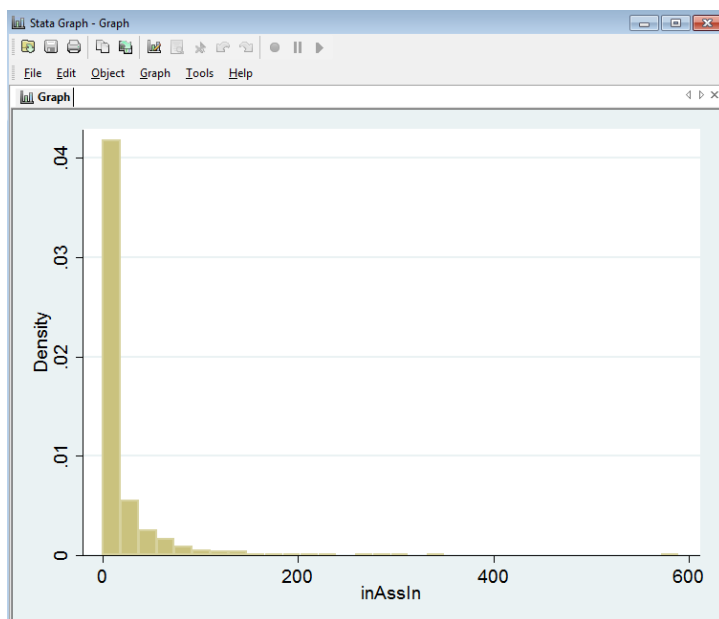
<sup>32</sup> This chapter was co-authored with Brett Burkhardt. The data used in this lesson originates from his research.

12. The Poisson distribution of  $y$  has an expected value/mean of  $E(y_i|x_i) = \mu$  (this is simply the average of the dependent variable)
13. The outcome,  $y_i$ , is discrete and can only embody non-negative values
14. The Poisson regression model has an **exponential transformation**,  $E(y_i|x_i) = e^{(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_n x_{in})}$ . It is important to note two things here. One, this transformation is slightly different from the log transformation of logistic models, but is done for roughly the same reason (to eliminate infeasible, i.e. negative, count outcomes). Two, the dependent variable for Poisson regression is expressed in terms of the expected value/average count, NOT the absolute count (this will be explained below).
15. The variance of  $y_i$  given  $x_i$  is equal to the mean of  $y_i$ ,  $\mu$  ( $\text{Var}(y_i|x_i) = \mu$ , the equidispersion assumption)
16. No (perfect) multi-collinearity

This lesson starts with a discussion of why Poisson regression is more suitable than OLS for counts data and how to interpret its results. Then, negative binomial regression, an alternative counts estimator, is discussed and it will be explained under what conditions this is a more optimal estimator. The data for this lesson are from the Bureau of Justice Statistics' 2005 *Census of State and Federal Adult Correctional Facilities* series. It contains data on all American adult correctional facilities in the country that hold primarily state or federal prisoners. It covers a wide variety of types of facilities, ranging from very high security confinement facilities to low security rehabilitation facilities that allow inmates to enter the community. In this lesson, we will focus on factors which influence the number of inmate assaults on other inmates (inassin"). The dataset includes the following variables, with **count variables in bold**:

- **inassin: Number of reported inmate assaults on other inmates**
- **inassstaffn: Number of reported inmate assaults on staff**
- minSec: Facility security (dummy): Minimum
- medSec: Facility security (dummy): Medium
- maxOrSuperSec: Facility security (dummy): Maximum or supermax
- private: 1 if privately operated; 0 if publicly operated
- crowd: Level of overcrowding
- age: Age of facility

Count dependent variables are problematic for OLS for three reasons. First, OLS produces non-normally distributed errors (violation of Assumption 7). Second, OLS produces errors without constant variance (i.e., heteroskedasticity; violation of Assumption 5). Third, OLS may produce negative predicted counts, which are logically impossible. Let's illustrate each of these problems by applying OLS to the dependent variable inassin. We'll start by looking at the distribution of the dependent variable. Create a histogram of the variable by typing "histogram inassin" in the STATA command box. You should be presented with the histogram below.



The variable is clearly skewed to the right, with an overabundance of zeros. More than 40% of facilities reported zero inmate-on-inmate assaults. You'll also (barely) notice the presence of some outliers in the sample; one facility reported 590 inmate assaults on other inmates. These outliers will cause our residuals to be skewed, which will violate the OLS normal distribution assumption that is required for t-testing.

In order to plot the distribution of residuals, we need to estimate an OLS model. Run a linear regression with inmate-on-inmate assaults (*inassin*) as the dependent variable and medium security (*medsec*) and maximum/supermax security (*maxorsupersec*) dummies (note, this means minimum security facilities are the baseline category) and crowding (*crowd*) as the independent variables. You should be presented with the following output:

```
. reg inassin medsec maxorsupersec crowd
```

Source	SS	df	MS
Model	276879.176	3	92293.0588
Residual	1682898.57	1643	1024.28398
Total	1959777.75	1646	1190.63047

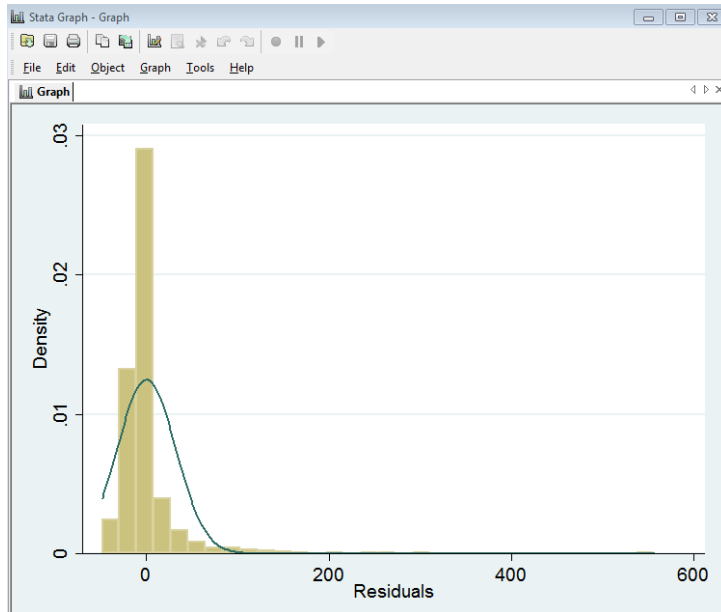
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
<i>medsec</i>	17.6068	1.893859	9.30	0.000	13.89217 21.32144
<i>maxorsuper~c</i>	27.1645	2.05899	13.19	0.000	23.12598 31.20303
<i>crowd</i>	16.03109	3.036913	5.28	0.000	10.07446 21.98772
<i>_cons</i>	-11.74235	3.146112	-3.73	0.000	-17.91316 -5.571537

Number of obs =	1647
F( 3, 1643) =	90.10
Prob > F =	0.0000
R-squared =	0.1413
Adj R-squared =	0.1397
Root MSE =	32.004

There appears to be significant and positive relationships between the number of inmate assaults and medium security facilities (relative to minimum security), maximum/supermax security facilities (relative to minimum), and overcrowded facilities. This is all plausible. However, this model may violate OLS assumptions which may lead to biased betas and/or inaccurate significance testing. To see whether the residuals of the model are normally distributed (Assumption 7), generate residuals by typing "predict

residual,  $r$ " (note, you are computing pure residuals here, not studentized residuals that you generated in Lesson 9). You will notice STATA creates a new variable, your residuals, from the linear model. Visually inspect the shape of the errors by producing a histogram of the residuals. You can add option "normal" to compare the residuals' distribution (what we have) to a normal distribution (what we want): type "histogram residual, normal" into the STATA command editor. You should be presented with the following graphic:



Notice that the distribution as a whole is asymmetric and skewed to the right; most of our count data congregates towards zero, and there are far fewer negative residuals than positive residuals (under a normal distribution we would expect these to be balanced). Also, the presence of several outliers skews the distribution of our residuals. Consequently, the normal distribution assumption is violated, which means the results of our t-tests for our beta coefficients may be skewed.

We can also test whether the heteroskedasticity assumption holds with this OLS model via a White test. Recall that the presence of heteroskedasticity does not bias our beta coefficients, but it does understate the values of the standard errors, leading to overly-optimistic t-statistics. To test the model above for heteroskedasticity, type "estat imtest, white" into the STATA command box (you can also check the "rvfplot" (residual versus fitted plot) to graphically examine the shape of the residuals). You should be presented with the following:

```

. predict residual, r
(174 missing values generated)

. histogram residual, normal
(bin=32, start=-48.34901, width=18.902871)

. estat imtest, white

white's test for Ho: homoskedasticity
  against Ha: unrestricted heteroskedasticity

      chi2(6)      =      30.23
      Prob > chi2   =      0.0000

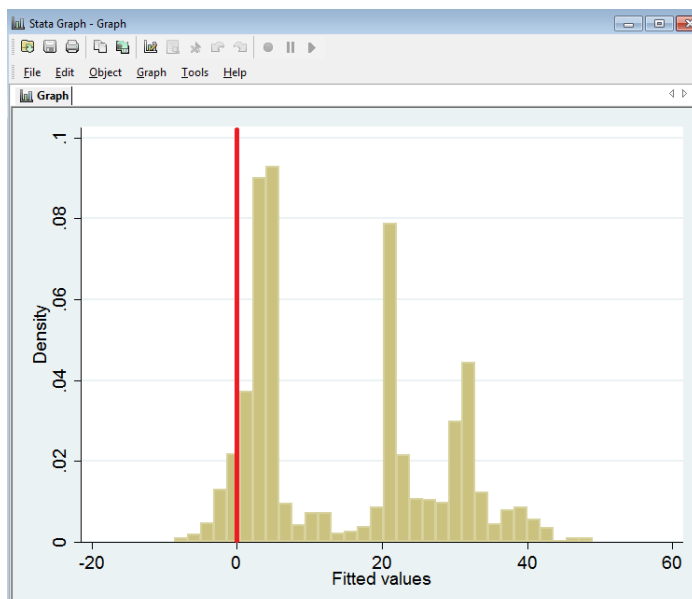
Cameron & Trivedi's decomposition of IM-test

```

Source	chi2	df	p
Heteroskedasticity	30.23	6	0.0000
Skewness	10.73	3	0.0133
Kurtosis	1.71	1	0.1913
Total	42.66	10	0.0000

The significant chi-square value tells us to reject the null hypothesis of homoscedasticity – hence Assumption 5 of OLS is also violated. This result is not significantly problematic. As was discussed in Lesson 10, heteroskedasticity can be rectified via the use of robust standard errors, or alternative model specification. For this particular model, however, we have to interpret the significance of the beta coefficients cautiously, as these results are likely inflated in the presence of heteroskedasticity.

A final risk in using OLS for count data, as with logit regression, is the prediction of infeasible outcomes, namely negative counts. In order to determine whether this OLS regression has produced any negative predictions of inmate assaults, type "predict yhat\_ols" into the STATA command box. This will generate a new variable, which is the predicted number of counts for each observation based on the regression output. We can track whether any of these predicted counts are below zero via the "sum yhat\_ols, d" command, which will present the minimum values of the predicted outcomes, or by producing a simple histogram. Type "histogram yhat\_ols" into the STATA command box. You should be presented with the following graphic (without the reference line):



Notice that some of our predicted values are less than zero (those that are to the left of the red reference line). This indicates that our model has produced infeasible count outcomes. While OLS's violation of heteroskedasticity is not particularly problematic in its use for count outcomes, as it can easily be rectified within the linear model, the production negative count outcomes cannot be rectified within a linear model. In order to correct this problem, Poisson exponentially transforms the linear model

$(E(y_i|x_i) = e^{(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n)})$  so that the production of negative counts is impossible (mathematically, the exponential transformation of any number produces a positive outcome). However, this means, like with logistic regression, that our Poisson regression output requires some care in its interpretation – we cannot interpret Poisson beta coefficients in terms of how a marginal change in X influences y.

To conduct a Poisson regression, click on “Statistics” in the top tool bar, then click on “count outcomes”, followed by “Poisson regression”. You should be presented with the following box:

The screenshot shows a software dialog box titled "poisson - Poisson regression". It has several tabs: "Model", "by/f/in", "Weights", "SE/Robust", "Reporting", and "Max options". The "Model" tab is active. Inside, there are fields for "Dependent variable:" (set to "inassin") and "Independent variables:". Below these is a checkbox for "Suppress constant term". An "Options" section contains radio buttons for "Exposure variable:" (selected) and "Offset variable:". Below that is a "Constraints:" section with a dropdown menu and a "Manage..." button. At the bottom of the dialog is a checkbox for "Keep collinear variables (rarely used)". The bottom of the window features a toolbar with icons for help, a printer, and a document, along with "OK", "Cancel", and "Submit" buttons.

Let's start our discussion of Poisson models by running a Poisson regression with only a constant term and no independent variables (we do this to demonstrate why we interpret Poisson output relative to the “expected value” or the “mean/average” count). Enter inassin as the dependent variable, and leave the independent variable box blank. Click “Ok”. You should be presented with the following output:

```
. poisson inassin
Iteration 0:   log likelihood = -35182.171
Iteration 1:   log likelihood = -35182.171

Poisson regression              Number of obs   =       1694
                                LR chi2(0)        =         0.00
                                Prob > chi2       =         .
                                Pseudo R2         =       0.0000

Log likelihood = -35182.171
```

	inassin	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
	_cons	2.74612	.006155	446.16	0.000	2.734056 2.758184

**CONGRATUALTIONS! You have just estimated a Poisson regression in STATA!** Before we dissect the nuts and bolts of Poisson, there are two things to emphasize regarding its output. First, though Poisson regression models are exponentially transformed ( $E(y_i|x_i) = e^{(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n)}$ ) to prevent negative count outcomes, output provided in STATA, is provided in terms of the natural log of  $y_i$  so as to eliminate the exponentiated model ( $\ln(E(y_i|x_i)) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n$ ). Therefore, beta coefficients in the STATA output express changes in the *natural log of the average count*, NOT the count or the exponentiated value of the dependent variable. In order to draw conclusions about the impact of an independent variable one must exponentially transform these beta coefficients (we will discuss this in more detail below).

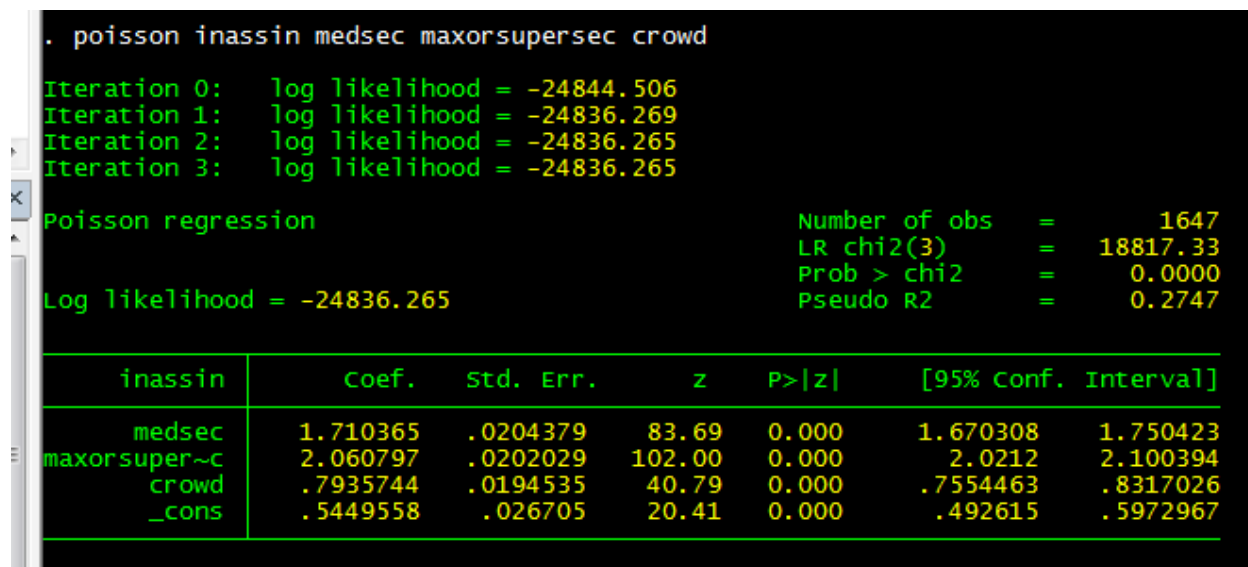
The second point to emphasize about Poisson output is that all results MUST be expressed in relation to the expected/mean/average count. In OLS, we would state the results of a regression coefficient as follows: “an increase in X changes y by  $\hat{\beta}$ ”. For Poisson, we would have to revise this statement to the following: “an increase in X changes the *expected value* of y (or the *average* count) by  $\hat{\beta}$ .” The reason why we interpret Poisson findings in relation to the expected/average count is because this model, in its most basic form (i.e. only consisting of a constant term), originates from the average count of the dependent variable. To test this, mathematically compute the exponentiated value of the constant term above (you should obtain  $e^{2.74612} = 15.582056\dots$ ). Then, compare this value to the average of inassin by typing “mean inassin” into the STATA command box. You should be presented with the following number:

```
. mean inassin
Mean estimation              Number of obs   =       1694
```

	Mean	Std. Err.	[95% Conf. Interval]
inassin	15.58205	.8344927	13.94531 17.2188

Notice that the average number of inmate assaults is identical to that of the exponentially transformed constant term. This is not a coincidence; Poisson models use the average/expected/mean count as the base result. Hence, results interpretation must be made in respect to the average/mean/expected count.

Let's estimate the OLS model above using Poisson regression. Add medsec, maxorsupersec, and crowd as independent variables in the Poisson regression box and click "Ok". You should be presented with the following results:



```
. poisson inassin medsec maxorsupersec crowd
```

Iteration 0: log likelihood = -24844.506  
Iteration 1: log likelihood = -24836.269  
Iteration 2: log likelihood = -24836.265  
Iteration 3: log likelihood = -24836.265

Poisson regression

Log likelihood = -24836.265

Number of obs = 1647  
LR chi2(3) = 18817.33  
Prob > chi2 = 0.0000  
Pseudo R2 = 0.2747

	inassin	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
medsec		1.710365	.0204379	83.69	0.000	1.670308 1.750423
maxorsuper~c		2.060797	.0202029	102.00	0.000	2.0212 2.100394
crowd		.7935744	.0194535	40.79	0.000	.7554463 .8317026
_cons		.5449558	.026705	20.41	0.000	.492615 .5972967

### CONGRATUALTIONS! You have just estimated a multivariate Poisson regression in STATA!

Like logit models, Poisson (and negative binomial regression) is a maximum likelihood estimator, and hence uses iterations to estimate results. You will also notice several similarities with logit. Poisson output produces a LR Chi-squared statistic, which determines the significance of the entire model. Similar to our F-statistic in OLS, or our LR Chi-squared statistic in logit, if we are presented with an insignificant Chi-squared statistic, we cannot reject the null hypothesis that our entire model is insignificant (that all beta coefficients of our dependent variables are equal to zero). Poisson also produces a psuedo R-squared value as a proxy of goodness of fit (although this is not a meaningful measure of goodness of fit, like R-squared in OLS, and should only be used when comparing different models, if at all). Also notice that significance testing in Poisson is conducted via z-statistics, identical to logit, and these can be interpreted in a similar manner as t-statistics in OLS (although the critical value for 90%, 95% and 99% significance differs slightly between t- and z-statistics). Finally, Poisson's beta coefficients should also be interpreted in the same general manner as those in logit, as they are in natural log values and not actual count values. The **ONLY** conclusions we can make from the results above are the direction of X's influence on the average/mean/expected count; we can say nothing about its magnitude. If the beta coefficient is significantly positive, higher values of X will increase the expected/average/mean count. If the beta coefficient is significantly negative, higher values of X will decrease the expected/average/mean count. Because all our coefficients above are positive, we can make the general claim that *increased crowding, and medium and maximum security prisons, relative to minimum security facilities, **increase** the average/expected/mean number of inmate-on-inmate assaults.* We can make no assertions about the magnitude of these increases from the output above.

## STATA COMMAND 15.1:

*Code:* “**poisson var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

*Output produced:* Generates Poisson regression output, where beta coefficients of independent variables are expressed in terms of the natural log of the expected/average/mean count.

In logistic regression we used odds ratios and fitted probabilities to provide meaningful values to our transformed beta coefficients. It is also possible to use fitted probabilities for Poisson estimates with the “prgen” and “prvalue” commands (we will not discuss this here, but the procedures are identical to those used in the logistic, ordinal logit, and multinomial logit lessons)<sup>33</sup>. For Poisson, we use factor changes to provide meaning to the beta magnitudes. Factor changes are similar to odds ratios in that they represent a multiplicative change of X on y, rather than an additive change, which is how we interpreted our beta coefficients for OLS. In order to calculate factor changes, type “listcoef” immediately after the Poisson regression. You should be presented with the following:

```
. poisson inassin medsec maxorsupersec crowd

Iteration 0:  log likelihood = -24844.506
Iteration 1:  log likelihood = -24836.269
Iteration 2:  log likelihood = -24836.265
Iteration 3:  log likelihood = -24836.265

Poisson regression                               Number of obs   =       1647
                                                LR chi2(3)      =    18817.33
                                                Prob > chi2     =       0.0000
Log likelihood = -24836.265                     Pseudo R2       =       0.2747
```

	inassin	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
	medsec	1.710365	.0204379	83.69	0.000	1.670308	1.750423
	maxorsuper~c	2.060797	.0202029	102.00	0.000	2.0212	2.100394
	crowd	.7935744	.0194535	40.79	0.000	.7554463	.8317026
	_cons	.5449558	.026705	20.41	0.000	.492615	.5972967

```
. listcoef

poisson (N=1647): Factor Change in Expected Count
Observed SD: 34.505514
```

	inassin	b	z	P> z	e^b	e^bstdx	SDofx
	medsec	1.71037	83.686	0.000	5.5310	2.1546	0.4488
	maxorsuper~c	2.06080	102.005	0.000	7.8522	2.3342	0.4113
	crowd	0.79357	40.793	0.000	2.2113	1.2353	0.2663

<sup>33</sup> For the “prgen” command in Poisson, one additional variable will be generated, labeled the predicted rate mu. This is simply the estimated average/mean/expected count across all possible values of the X variable specified in the “prgen” command.

**CONGRATUALTIONS! You have just calculated factor changes for Poisson output using STATA!** The “listcoef” command provides the beta coefficients, z-statistics, and their associated p-value, similar to the Poisson regression output. It also provides factor changes of the beta coefficients (highlighted in yellow) and the standardized factor change (highlighted in light blue). Factor changes operate like odds ratios; they are *multiplicative* changes in the expected/average/mean count for a 1-unit change in X. Like odds ratios, if a factor change is larger than 1, this indicates that the expected/average/mean count increases as X assumes a higher numerical value. If a factor change is smaller than 1, this indicates that the expected/average/mean count decreases as X assumes a higher numerical value. Based on the results above, we can interpret the factor change of medsec in the following manner: *if a facility is a medium security prison, relative to a minimum security prison, the expected/average/mean count of inmate assaults increases by a factor of 5.531*. Likewise, for the continuous crowding variable, we could make the following claim: *for every 1 unit increase in crowding, the expected/average/mean count of inmate assaults increases by a factor of 2.211*. Standardized factor changes may be more meaningful to use for continuous variables, as it provides the means to compare (continuous) independent variables that are measured in different manners. Standardized factor changes also may be more appropriate for continuous variables with small ranges, where a one unit increase would represent a dramatic increase. In regards to the crowding variable, we could also make the following claim: *for every standard deviation increase in crowding, the expected/average/mean count of inmate assaults increases by a factor of 1.235*.

#### **STATA COMMAND 15.2:**

*Code:* “**poisson var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.  
**“listcoef”**

*Output produced:* Produces factor changes for Poisson beta coefficients (note, “listcoef” can also be used with “nbreg”).

While Poisson regression overcomes problems associated with modeling count outcomes with OLS, it is subject to two additional problems of its own. The first problem associated with Poisson is that it can under-predict zero count outcomes and over-predict non-zero count outcomes for counts data with low means and considerable zero-clustering. Highlighted in the first histogram of this lesson, there were a considerable number of facilities (more than 40%) which reported no inmate-on-inmate assaults. To see whether Poisson over or under-predicted this high number of zero-assaults, we can use the “prcounts” command. The “prcounts” command is very versatile and, among other things, estimates the predicted probabilities of achieving a specific count value. We will use this command to compare the distribution of predicted outcomes of a Poisson model with the distribution of actual count outcomes. Like “prvalue” and “prgen”, the “prcounts” command is a post-estimation command whose results are uniquely tied to the previously estimated model. Below, we will use this command to plot the distribution of predicted inmate assaults – while the range of counts one can examine extends to 99, we will examine how predicted and actual counts are distributed for facilities with 0-50 assaults, as this is where the majority of the data is distributed. Type “prcounts prpois, plot max(50)” into the STATA command editor immediately after the Poisson regression specified above. You should see the following screen:

```

32 poisson inassin medsec maxorsupersec crowd
33 prcounts prpois, plot max(50)

```

Name	Label
prpoiscu40	Pr(y<=40) from poisson
prpoiscu41	Pr(y<=41) from poisson
prpoiscu42	Pr(y<=42) from poisson
prpoiscu43	Pr(y<=43) from poisson
prpoiscu44	Pr(y<=44) from poisson
prpoiscu45	Pr(y<=45) from poisson
prpoiscu46	Pr(y<=46) from poisson
prpoiscu47	Pr(y<=47) from poisson
prpoiscu48	Pr(y<=48) from poisson
prpoiscu49	Pr(y<=49) from poisson
prpoiscu50	Pr(y<=50) from poisson
prpoisprgt	Pr(y>50) from poisson
prpoisval	Count
prpoisobeq	Observed Pr(y=k) from poisson
prpoispreq	Predicted Pr(y=k) from poisson
prpoisoble	Observed Pr(y<=k) from poisson
prpoisprle	Predicted Pr(y<=k) from poisson

```

Iteration 2: log likelihood = -24836.265
Iteration 3: log likelihood = -24836.265

Poisson regression
Log likelihood = -24836.265

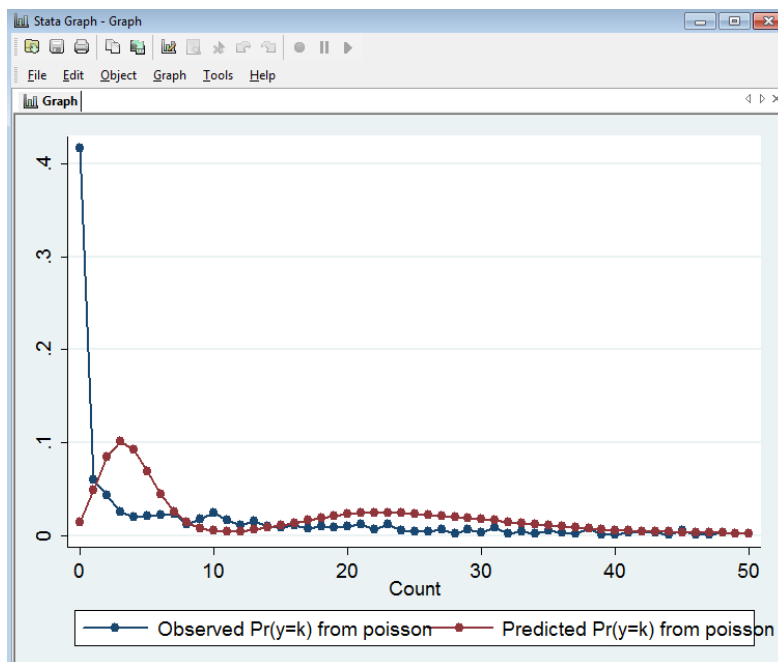
Number of obs   = 1647
LR chi2(3)      = 18817.33
Prob > chi2     = 0.0000
Pseudo R2       = 0.2747

+-----+-----+-----+-----+-----+-----+
| inassin | Coef. | Std. Err. | z    | P>|z| | [95% Conf. Interval] |
+-----+-----+-----+-----+-----+-----+
| medsec | 1.710365 | .0204379 | 83.69 | 0.000 | 1.670308 | 1.750423 |
| maxorsuper~C | 2.060797 | .0202029 | 102.00 | 0.000 | 2.0212 | 2.100394 |
| crowd  | .7935744 | .0194535 | 40.79 | 0.000 | .7554463 | .8317026 |
| _cons  | .5449558 | .026705 | 20.41 | 0.000 | .492615 | .5972967 |
+-----+-----+-----+-----+-----+-----+

. prcounts prpois, plot max(50)
(52 missing values generated)

```

You will notice that “prcounts” has created two types of values in our variables box: the first (ranging from 0 to 50, highlighted in blue) is the probability distribution of witnessing that particular count (which we’ve delienated as “prpois”). The second group of variables, highlighted in red, is what we will focus on for this lesson; these are the predicted and observed count outcomes for the Poisson regression model. In order to graph how the distribution of the predicted values (prpoispreq) compare to the distribution of acutal counts (prpoisobeq), type the following syntax into the STATA command box: “graph twoway connected prpoisobeq prpoispreq prpoisval”. You should be presented with the following graphic:



**CONGRATUALTIONS!** You have just created a graphic which compares the distribution of the predicted values from a Poisson model to the distribution of the actual count outcomes in STATA! Notice from the above graphic that the distritution of the Poisson model’s predicted counts (red line) is

significantly below the distribution of actual counts (blue line) for zero counts. What this means is that the Poisson model is severely under-estimating zero counts. Likewise, Poisson over-predicts counts above 0 but less than 10, compared to the actual observed values, indicated by the fact that the red line is much higher than the blue line. This graphic which compares Poisson's predicted counts to our actual counts value highlights an important problem when using Poisson for zero-clustered count data: Poisson may severely under-predict zero count outcomes, and over-predict non-zero count outcomes.

### STATA COMMAND 15.3:

*Code:* “***poisson var1 var2 var3 ...***”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

“***prcounts varname, plot max(#)***” where varname is the name assigned to the estimations of predicted count values, and # is the maximum of the range of counts the programmer wishes to consider (STATA sets a default at 9 if no maximum is specified)

“***graph twoway connected varnameobeq varnamepreq varnameval***”

*Output produced:* Graphs the distribution of predicted counts from Poisson and the distribution of actual count values (note, “prcounts” can also be used with “nbreg”).

A second problem associated with Poisson regression revolves around its equi-dispersion assumption. According to this assumption, the mean of the count variable must also be equal to its variance, which rarely occurs in practice. Equi-dispersion can be violated because the conditional variance is less than the conditional mean (*under-dispersion*) or because it is greater than the conditional mean (*over-dispersion*); the latter is much more prevalent in counts data than the former, and is especially present in counts data where significant outliers exist. Like heteroskedasticity, the violation of the equi-dispersion assumption yields heavily understated standard errors, and heavily inflated z-statistics increasing the risk of Type I errors (notice that significance of the Poisson model was exceptionally high, compared to OLS). One common approach for accounting for over-dispersion in Poisson is the use of negative binomial regression. Given that the equi-dispersion assumption is so frequently violated, scholars analyzing counts data generally rely upon negative binomial regression, rather than Poisson, as their default model.

In order to determine whether equi-dispersion is violated in the Poisson model above, one must first conduct a negative binomial regression. Click on the “Statistics” tab in the top tool bar, then click on “count outcomes”, and click “negative binomial regression”. You should be presented with the following box:

nbreg - Negative binomial regression

Model by/if/in Weights SE/Robust Reporting Max options

Dependent variable: inassin  
Independent variables: medsec maxorsupersec crowd

☐ Suppress constant term

Parameterization of dispersion  
☒ A function of the expected mean ☐ A constant

Options  
☒ Exposure variable:   
☐ Offset variable:

Constraints:  Manage...

☐ Keep collinear variables (rarely used)

OK Cancel Submit

Insert inassin as the dependent variable, and medsec, maxorsupersec and crowd as the independent variables (the same model used above). Click “Ok” and you should be presented with the following output:

```
. nbreg inassin medsec maxorsupersec crowd, dispersion(mean)
Fitting Poisson model:
Iteration 0:  log likelihood = -24844.506
Iteration 1:  log likelihood = -24836.269
Iteration 2:  log likelihood = -24836.265
Iteration 3:  log likelihood = -24836.265
Fitting constant-only model:
Iteration 0:  log likelihood = -6213.0917
Iteration 1:  log likelihood = -5114.8439
Iteration 2:  log likelihood = -5113.2282
Iteration 3:  log likelihood = -5113.2275
Iteration 4:  log likelihood = -5113.2275
Fitting full model:
Iteration 0:  log likelihood = -4987.1886
Iteration 1:  log likelihood = -4975.334
Iteration 2:  log likelihood = -4933.8359
Iteration 3:  log likelihood = -4933.6366
Iteration 4:  log likelihood = -4933.6365
Negative binomial regression
Dispersion = mean
Log likelihood = -4933.6365
Number of obs = 1647
LR chi2(3) = 359.18
Prob > chi2 = 0.0000
Pseudo R2 = 0.0351
```

	inassin	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
medsec		1.716907	.1133628	15.15	0.000	1.494719 1.939094
maxorsuper~c		2.047404	.1237507	16.54	0.000	1.804857 2.289951
crowd		.9290526	.1993804	4.66	0.000	.5382743 1.319831
_cons		.4019525	.2070715	1.94	0.052	-.0039001 .8078052
/lnalpha		1.27773	.0411946			1.19699 1.35847
alpha		3.588485	.147826			3.310139 3.890236

likelihood-ratio test of alpha=0: chibar2(01) = 4.0e+04 Prob>=chibar2 = 0.000

**CONGRATUALTIONS! You have just estimated a negative binomial regression in STATA!** The negative binomial output reveals if over-dispersion exists via the alpha likelihood ratio test. Alpha represents the scalar by which the variance of the count variable must be changed in order to equal the mean (if  $\text{Var}(y_i|x_i) = \mu_i + \alpha\mu_i$ ,  $\alpha$  must equal zero in order for equi-dispersion to exist. If  $\alpha$  is greater than 0, over-dispersion exists, whereas if it is less than zero, under-dispersion exists). In the output above, the likelihood-ratio test of alpha (outlined in yellow) tells us that we can fail to reject the null hypothesis that alpha equals zero. Because alpha is significantly positive, over-dispersion exists in the data.

#### STATA COMMAND 15.4:

*Code:* “**nbreg var1 var2 var3 ...**”, where var1 is your dependent variable, and var2, var3, ... are your independent variables.

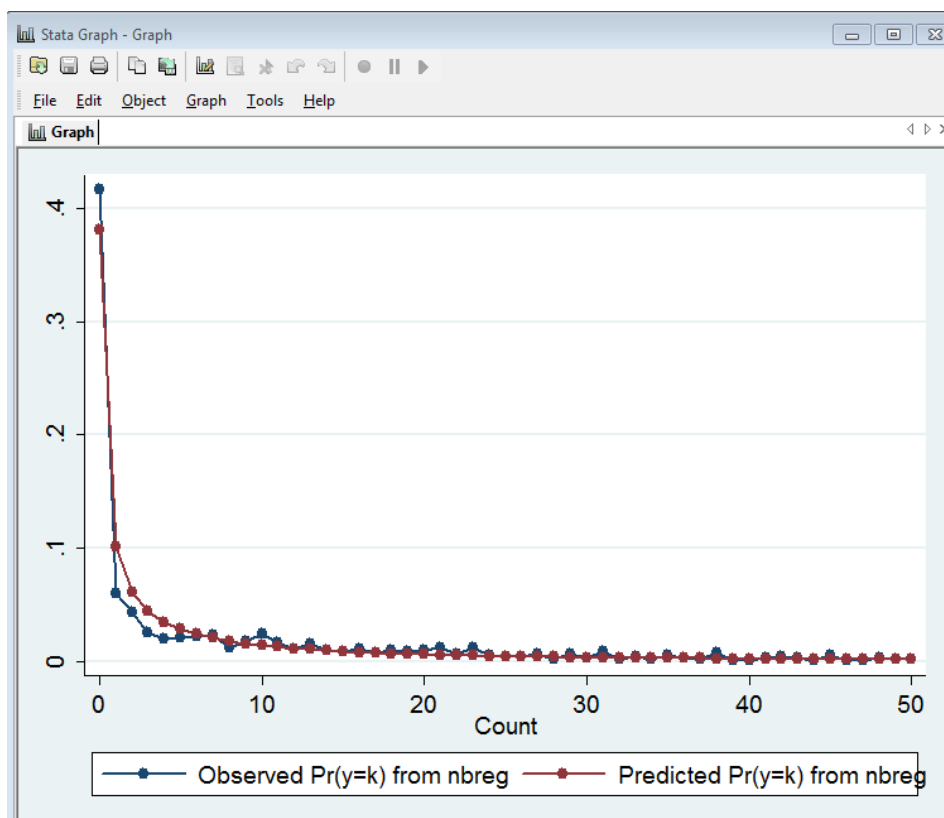
*Output produced:* Generates negative binomial regression output, where beta coefficients of independent variables are expressed in terms of the natural log of the expected/average/mean count.

Negative binomial regression adjusts the variance of the dependent variable by alpha in order to satisfy the equi-dispersion assumption. With this adjustment, the standard errors of the former Poisson model are adjusted upwards, producing lower z-statistics and reducing the risk of a Type 1 error. In terms of interpreting its beta coefficients, negative binomial regression’s output can be treated identically to Poisson’s; beta coefficients are expressed in terms of the natural log of the expected/average/mean count, and only general claims can be made about the direction of the effect (i.e. higher values of X increases/decreases the expected/average/mean count). In order to attach meaning to these beta coefficients, one can either calculate predicted values (via “prvalue” or “prgen”, not discussed in this lesson, although their use in “nbreg” is identical to that in “poisson” and “logit”) or factor changes. Factor changes can be computed in an identical manner as that used for Poisson. Immediately after the negative binomial regression, type “listcoef”. You should be presented with the following output:

/lnalpha	1.27773	.0411946		1.19699	1.	
alpha	3.588485	.147826		3.310139	3.8	
Likelihood-ratio test of alpha=0: chibar2(01) = 4.0e+04 Prob>=chibar2 =						
. listcoef						
nbreg (N=1647): Factor Change in Expected Count						
observed SD: 34.505514						
inassin	b	z	P> z	e^b	e^bstdx	SDofx
medsec	1.71691	15.145	0.000	5.5673	2.1609	0.4488
maxorsuper~c	2.04740	16.545	0.000	7.7478	2.3213	0.4113
crowd	0.92905	4.660	0.000	2.5321	1.2807	0.2663
ln alpha	1.27773					
alpha	3.58848	SE(alpha) = 0.14783				
LR test of alpha=0: 4.0e+04 Prob>=LRX2 = 0.000						

**CONGRATULATIONS! You have just calculated factor changes for negative binomial regression output using STATA!** As with Poisson, factor changes that are larger than one indicates that an increase in X leads to a multiplicative increase in the expected/average/mean count, while a factor change that is less than one indicates that an increase in X leads to a multiplicative decrease in the expected/average/mean count. If we wanted to interpret the factor change for the maxorsupersec variable, we would claim the following: *if a facility is a maximum or supermax security prison, relative to a minimum security prison, the expected/average/mean count of inmate assaults will increase by a factor of 7.748.*

An added benefit negative binomial regression, besides its correction of over/under-dispersion, is that this regression method also more accurately predicts counts outcomes for data that is heavily skewed towards zero. We can observe this by using the “prcounts” command. Immediately after you run the negative binomial regression above (notice, we’re using the same model specification as we used for the multivariate Poisson regression), type “prcounts pnbreg, plot max(50)”. STATA should create 50 new variables associated with our negative binomial regression model that display the predicted probabilities of witnessing a specific count outcome, as well as the predicted (pnbregpred) and observed (pnbregobeq) count outcomes for the negative binomial regression model. In order to determine whether negative binomial regression more accurately predicts zero counts, type the following command into the STATA command box: “graph twoway connected pnbregobeq pnbregpred pnbregval”. You should be presented with the following graphic:



**CONGRATUALTIONS! You have just created a graphic which compares the distribution of the predicted values from a negative binomial regression model to the distribution of the actual count**

**outcomes in STATA!** Unlike Poisson, the distribution of negative binomial regression's predicted values better match the distribution of the actual count values, especially in regards to the representation of zeros. Because negative binomial regression both corrects for over/under-dispersion and more accurately predicts counts that have zero-skew, it is more commonly employed for counts modeling than Poisson.

---

### Practice Problems:

Practice Problem 1: Run an OLS regression model using the number of reported inmate assaults on staff (*inassstaffn*) as the dependent variable, and the following independent variables: the medium and maximum (or supermax) security dummies, the private prison dummy, the degree of crowding in the prison and the age of the facility. Which variables are significant predictors of inmate assaults on staff? Is the model suitable for an analysis on the determinants of inmate assaults on staff (make sure to provide evidence with your claim)?

Practice Problem 2: Run a Poisson model using the regression from Problem 1 above. Which variables are significant predictors of inmate assaults on staff according to the model? How would you generally interpret the results of the age variable? The max or supermax security dummy?

Practice Problem 3: Using the model from Problem 2, interpret the results of the age variable and the medium security prison dummy in terms of factor changes. How would you interpret age in standardized factor changes?

Practice Problem 4: *Selecting 20 as the maximum count value*, does the Poisson model in Problem 2 accurately predict the distribution of inmate assaults on prison staff? Why or why not?

Practice Problem 5: What are the benefits of using negative binomial regression over Poisson? Using the model from Problem 2, is the equi-dispersion assumption violated? If so, what does this mean about your Poisson estimates? How do your estimates from the negative binomial regression model compare to those from the Poisson model?

Practice Problem 6: Using the negative binomial regression output from Problem 5, interpret the influence of crowding, the age of the facility and whether it is a medium security prison on the number of assaults on prison staff in terms of factor changes.

Practice Problem 7: Does the negative binomial model do a better job of predicting the distribution of inmate assaults on prison staff (use 20 as the maximum count value)? Why or why not?

## Appendix I: Helpful Commands for Data Cleaning/Management

---

One thing you will discover in data analysis, especially if using datasets that were coded by other researchers, is that datasets are not ready for analysis once you open them. Often they need to be cleaned and properly coded, especially if variables still exist within string form (i.e. are coded as words, which STATA does not recognize, rather than numbers, which STATA does recognize). One great feature about STATA is that it enables researchers to manage and recode their data. In this lesson, you will learn the six following major commands which are most commonly used in data management.

**STATA Command A.1:** *destring*

**STATA Command A.2:** *sort*

**STATA Command A.3:** *drop/keep*

A.3.1 Dropping observations

A.3.2 Dropping variables

A.3.3 Keeping observations

**STATA Command A.4:** *encode*

A.4.1 Codifying nominal categorical variables

A.4.2 Codifying ordinal categorical variables

**STATA Command A.5:** *replace*

A.5.1 Replacing numerical values with missing values

A.5.2 Replacing numerical values with other numerical values

**STATA Command A.6:** *generate*

A.6.1 Replicating a variable

A.6.2 Creating a variable that is conditional on the values of another variable

A.6.3 Creating a variable that is a mathematical function of one or more variables

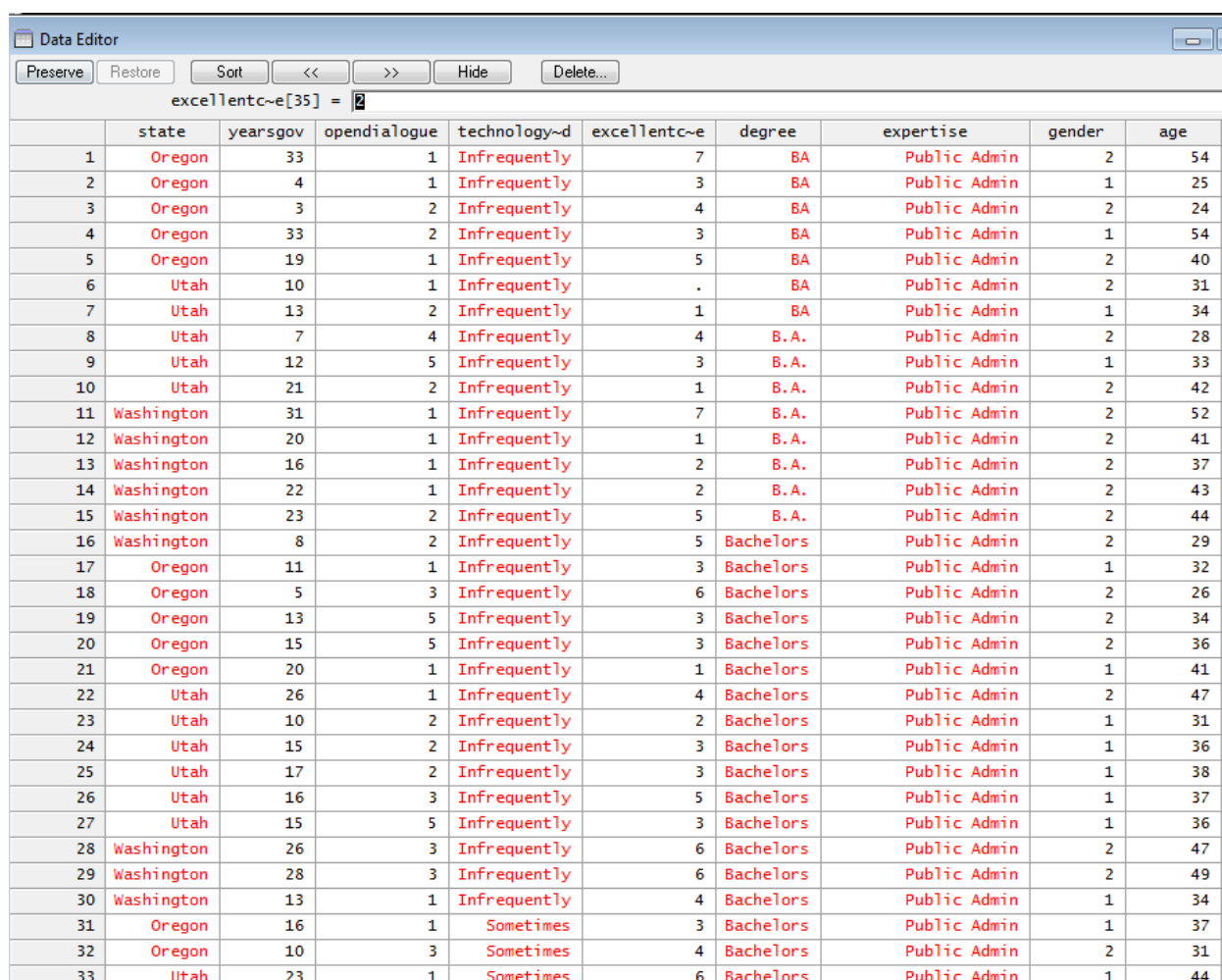
We will use two datasets to practice these commands on: 1.) The Civil Servant Survey administered by Professor Brent Steel at Oregon State University, and 2.) an elections statements database, created and used by MPP student Michael Nash for his MPP thesis, which documents which electoral candidates had public statements judged by the fact checking agency Politifact. The first dataset will be used for all exercises within the lesson, while the second will be the subject of the practice problems at the end. The variables included in the excel file are:

- **state:** The state where the civil servant works
- **yearsgov:** The number of years in government the civil servant has worked
- **opendialogue:** The degree to which the respondent believes the department in which he/she works encourages open dialogue (1 for “strongly agree”, 2 for “agree”, 3 for “somewhat agree”, 4 for “don’t know”, 5 for “somewhat disagree”, 6 for “disagree”, and 7 for “strongly disagree”)
- **technologyused:** The frequency of which advanced technology the respondent believes is used within his/her office (responses ranging from “infrequently”, “sometimes”, “always” and “NA”)
- **excellentservice:** The degree to which the respondent believes his/her office has an excellent civil service system (1 for “strongly agree”, 2 for “agree”, 3 for “somewhat agree”,

4 for “don’t know”, 5 for “somewhat disagree”, 6 for “disagree”, 7 for “strongly disagree”, and “NR” for “not recorded”)

- degree: The highest level of education received by the respondent (NOTE: this variable was fabricated for this exercise to preserve the identity of the respondent)
- expertise: The primary field of expertise of the respondent (NOTE: this variable was fabricated for purposes of the module)
- gender: The gender of the respondent (1 for “male”, 2 for “female”, and “#NULL!” for “not recorded”)
- age: The age of the respondent (NOTE: this variable was fabricated for this exercise to preserve the identity of the respondent)

Start by copying/pasting the data in the excel spreadsheet into the data editor in STATA. You may see a box that states “*You attempted to paste one or more string values into numeric variables. The contents of these cells, if any, are unchanged*”. Click “OK”. Your data should be presented as follows:



	state	yearsgov	opendialogue	technology~d	excellentc~e	degree	expertise	gender	age
1	Oregon	33	1	Infrequently	7	BA	Public Admin	2	54
2	Oregon	4	1	Infrequently	3	BA	Public Admin	1	25
3	Oregon	3	2	Infrequently	4	BA	Public Admin	2	24
4	Oregon	33	2	Infrequently	3	BA	Public Admin	1	54
5	Oregon	19	1	Infrequently	5	BA	Public Admin	2	40
6	Utah	10	1	Infrequently	.	BA	Public Admin	2	31
7	Utah	13	2	Infrequently	1	BA	Public Admin	1	34
8	Utah	7	4	Infrequently	4	B.A.	Public Admin	2	28
9	Utah	12	5	Infrequently	3	B.A.	Public Admin	1	33
10	Utah	21	2	Infrequently	1	B.A.	Public Admin	2	42
11	Washington	31	1	Infrequently	7	B.A.	Public Admin	2	52
12	Washington	20	1	Infrequently	1	B.A.	Public Admin	2	41
13	Washington	16	1	Infrequently	2	B.A.	Public Admin	2	37
14	Washington	22	1	Infrequently	2	B.A.	Public Admin	2	43
15	Washington	23	2	Infrequently	5	B.A.	Public Admin	2	44
16	Washington	8	2	Infrequently	5	Bachelors	Public Admin	2	29
17	Oregon	11	1	Infrequently	3	Bachelors	Public Admin	1	32
18	Oregon	5	3	Infrequently	6	Bachelors	Public Admin	2	26
19	Oregon	13	5	Infrequently	3	Bachelors	Public Admin	2	34
20	Oregon	15	5	Infrequently	3	Bachelors	Public Admin	2	36
21	Oregon	20	1	Infrequently	1	Bachelors	Public Admin	1	41
22	Utah	26	1	Infrequently	4	Bachelors	Public Admin	2	47
23	Utah	10	2	Infrequently	2	Bachelors	Public Admin	1	31
24	Utah	15	2	Infrequently	3	Bachelors	Public Admin	1	36
25	Utah	17	2	Infrequently	3	Bachelors	Public Admin	1	38
26	Utah	16	3	Infrequently	5	Bachelors	Public Admin	1	37
27	Utah	15	5	Infrequently	3	Bachelors	Public Admin	1	36
28	Washington	26	3	Infrequently	6	Bachelors	Public Admin	2	47
29	Washington	28	3	Infrequently	6	Bachelors	Public Admin	2	49
30	Washington	13	1	Infrequently	4	Bachelors	Public Admin	1	34
31	Oregon	16	1	Sometimes	3	Bachelors	Public Admin	1	37
32	Oregon	10	3	Sometimes	4	Bachelors	Public Admin	2	31
33	Utah	23	1	Sometimes	6	Bachelors	Public Admin	1	44

Note that with this given dataset, only five variables (those in black) are analyzable: “yearsgov”, “opendialogue”, “excellentcivilservice”, “gender”, and “age”. This is because these values are numeric.

STATA does not recognize string (word) variables – hence, anytime a variable is highlighted in red, you will not be able to quantitatively analyze. Also note that for variables which had some string responses for a numerical variable (i.e. “NR” for “excellentcivilservice”, or “#NULL!” for “gender”), STATA automatically replaced these values with periods, “.”, the symbol for missing values in STATA (you will learn how to replace missing values with periods below). This may not always happen however – you will notice this in your practice problems. If you find that one of your numerical variables is presented in red, due to the presence of several cells which have string variables within them, you must use the “destring” command to replace these. For example, if STATA had not removed the “NR” responses from our “excellentcivilservice” variable, you would have to type the following command into your STATA command editor: “destring excellentcivilservice, replace ignore(NR)”.

**STATA COMMAND A.1:**

*Code: “destring var1, replace ignore(word)”*

*Output produced:* Expels worded values (word) from an otherwise numerical variable, var1.

The civil service survey was administered across three states (Washington, Oregon and Utah) and contains 1,339 observations. Let’s say that you are only interested in analyzing data for the Pacific Northwest and hence wish to expunge Utah from your datafile (if you download datasets from ICPSR, you may find datafiles with over 100,000 observations, most of which you may prefer to delete!). In order to know which variables to drop, it is helpful to use the “sort” command first, which orders your data either alphabetically if it is a string variable, or in numerical order if it is a numerical variable. In the STATA command box type “sort state” and enter. You may find that only the command is presented in the output window, nothing else. However, re-open your data editor and you should see the following:

Data Editor

Preserve Restore Sort << >> Hide Delete...

state[35] = Oregon

	state	yearsgov	opendialogue	technology~d	excellents~e	degree	expertise	gender	age
1	Oregon	4	1	Infrequently	7	BA	Environmental Policy	2	22
2	Oregon	32	1	Infrequently	4	Bachelors	Public Admin	1	59
3	Oregon	21	2	Sometimes	4	MPA	Legal Issues	2	51
4	Oregon	36	3	Sometimes	3	MPA	Legal Issues	1	66
5	Oregon	9	3	Sometimes	4	PhD	Legal Issues	2	39
6	Oregon	38	2	Infrequently	1	Bachelors	Environmental Policy	2	56
7	Oregon	8	4	Always	4	Doctorate	Energy Policy	1	37
8	Oregon	24	3	Sometimes	3	PhD	Economic Policy	1	53
9	Oregon	6	2	Sometimes	4	MPA	Economic Policy	1	35
10	Oregon	16	2	Infrequently	3	MPP	Economic Policy	1	45
11	Oregon	16	3	Sometimes	3	Doctorate	Environmental Policy	2	46
12	Oregon	27	2	Infrequently	4	MPP	Economic Policy	1	56
13	Oregon	3	1	Sometimes	3	BS	Legal Issues	2	30
14	Oregon	16	3	Sometimes	5	MPA	Economic Policy	2	45
15	Oregon	32	3	Always	5	Doctorate	Environmental Policy	1	50
16	Oregon	19	2	Sometimes	5	MPA	Economic Policy	2	48
17	Oregon	7	5	Always	4	Doctorate	Energy Policy	1	36
18	Oregon	33	3	Always	4	Doctorate	Environmental Policy	2	51
19	Oregon	24	2	Always	6	Masters	Legal Issues	1	51
20	Oregon	8	4	Sometimes	5	Doctorate	Energy Policy	1	37
21	Oregon	29	7	Infrequently	5	Diploma	Public Admin	1	54
22	Oregon	19	3	Sometimes	3	Doctorate	Energy Policy	1	48
23	Oregon	31	2	Always	4	Doctorate	Public Admin	2	56
24	Oregon	22	2	Always	5	Doctorate	Environmental Policy	2	40
25	Oregon	18	1	Always	5	Masters	Public Admin	2	43
26	Oregon	36	1	Infrequently	2	Bachelors	Environmental Policy	2	54
27	Oregon	23	3	Sometimes	3	PhD	Economic Policy	1	52
28	Oregon	44	1	Always	3	Doctorate	Energy Policy	1	73
29	Oregon	17	2	Sometimes	3	MPA	Economic Policy	1	46
30	Oregon	12	2	Sometimes	3	MPA	Economic Policy	1	41
31	Oregon	30	1	Always	6	Bachelors	Public Admin	2	51
32	Oregon	45	4	Always	6	Doctorate	Environmental Policy	2	63
33	Oregon	29	3	Sometimes	3	PhD	Economic Policy	1	58
34	Oregon	20	3	Infrequently	5	MPP	Public Admin	1	45

**Congratulations! You have just sorted your data by state in STATA!**

## STATA COMMAND A.2:

*Code: “sort var1”*

*Output produced:* Sorts your data either alphabetically or numerically according to var1.

As you scroll down your data editor, you will see that your entire data is ordered alphabetically by state. This makes it easier to determine which observations (in our case, those associated with Utah civil servants) to drop, which we turn to next. However, before we implement the drop and keep commands, it is **VITAL** that you save a copy of the full dataset first! Once you drop observations, you cannot re-obtain them unless you saved a master copy. To save the data file, click on the “File” tab and then “Save As” and call the file “Masterdataset”.

To drop observations you need to specifically identify to STATA which observations you wish to expel. Open up your data editor and scroll down to the beginning of the Utah entries – you should note that all respondents from Utah occupy observation 454 to 887. Close the data editor and type in the command box “drop in 454/887” (the later piece of coding indicates the range of observations you wish to delete, inclusive of the first and final value – this is why it is helpful to sort your data first). Like the sort command, you should see the command in white, accompanied by green text which identifies how many observations were dropped (in this case 434). Open up the data editor and scroll down to the end of the Oregon entries. You should see the following:

	state	yearsgov	opendialogue	technology~d	excellentc~e	degree	expertise	gender	age
451	Oregon	15	3	Always	7	Bachelors	Public Admin	2	40
452	Oregon	25	3	Sometimes	3	Doctorate	Energy Policy	1	54
453	Oregon	16	3	Sometimes	4	PhD	Economic Policy	1	45
454	Washington	9	3	Sometimes	3	Doctorate	Energy Policy	2	38
455	Washington	15	2	Infrequently	.	BS	Public Admin	1	36
456	Washington	31	5	Infrequently	4	BS	Environmental Policy	2	49
457	Washington	8	2	Sometimes	3	Bachelors	Public Admin	2	33

### **Congratulations! You have just dropped multiple observations in STATA!**

You can also drop variables from your dataset. Say we wanted to drop the variable age from our dataset, as we no longer needed it for our analysis. Simply type “drop age” into the STATA command box. Similar to the results above, you should only see the command replicated in the black/white STATA output box. If you open the data editor, however, you will notice that age is no longer in the dataset.

As emphasized above, once you drop observations in STATA you cannot retrieve them. Hence, you should ONLY consider dropping values if your dataset is too big to manage in its complete form (i.e. those with hundreds of thousands of observations), and ONLY after you save a master copy for your file.

#### **STATA COMMAND A.3.1:**

*Code:* “**drop in value1/value2**”, where value1 is the first number of your observation range and value2 is the final number of the observation range.

*Output produced:* Drops multiple observations in STATA.

*Caveat:* Once you drop observations, you CAN NOT retrieve them. Make sure to save a master file before you do so.

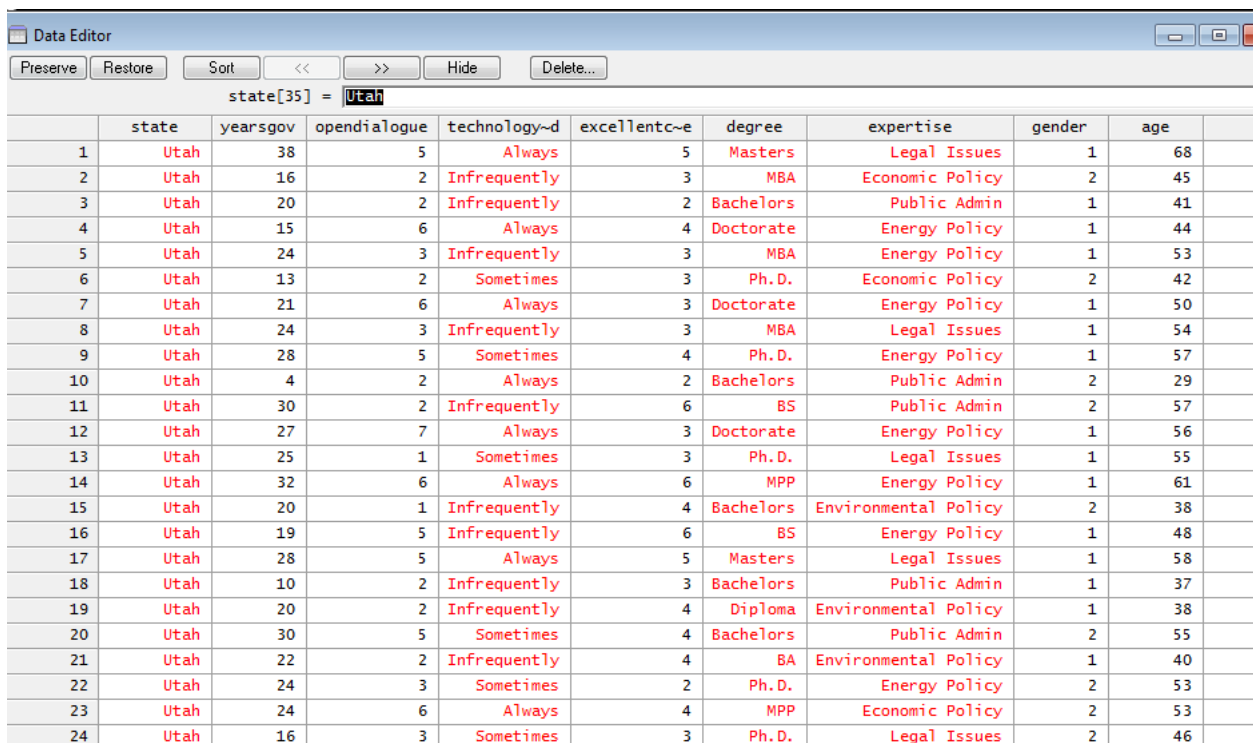
### STATA COMMAND A.3.2:

*Code:* “**drop var1**”, where var1 is the variable within the dataset you wish to drop.

*Output produced:* Removes the variable from your dataset.

*Caveat:* Once you drop variables, you CAN NOT retrieve them. Make sure to save a master file before you do so.

The drop command is a convenient one to use if you wish to expunge a small subset of data from your dataset. If you wish to expunge a large subset, however, the keep command may be more appropriate. Close down the datafile (do not save it) and reopen the Master file with observations from all three states. Now, say instead of only analyzing data for the Pacific Northwest respondents, you are interested in analyzing data for civil servants working only in Utah. In the command box, type “keep in 454/887”: you should see the command as well as the number of observations expunged (905) in the output box. Open the data editor and you should see the following:



	state	yearsgov	opendialogue	technology~d	excellentc~e	degree	expertise	gender	age
1	Utah	38	5	Always	5	Masters	Legal Issues	1	68
2	Utah	16	2	Infrequently	3	MBA	Economic Policy	2	45
3	Utah	20	2	Infrequently	2	Bachelors	Public Admin	1	41
4	Utah	15	6	Always	4	Doctorate	Energy Policy	1	44
5	Utah	24	3	Infrequently	3	MBA	Energy Policy	1	53
6	Utah	13	2	Sometimes	3	Ph.D.	Economic Policy	2	42
7	Utah	21	6	Always	3	Doctorate	Energy Policy	1	50
8	Utah	24	3	Infrequently	3	MBA	Legal Issues	1	54
9	Utah	28	5	Sometimes	4	Ph.D.	Energy Policy	1	57
10	Utah	4	2	Always	2	Bachelors	Public Admin	2	29
11	Utah	30	2	Infrequently	6	BS	Public Admin	2	57
12	Utah	27	7	Always	3	Doctorate	Energy Policy	1	56
13	Utah	25	1	Sometimes	3	Ph.D.	Legal Issues	1	55
14	Utah	32	6	Always	6	MPP	Energy Policy	1	61
15	Utah	20	1	Infrequently	4	Bachelors	Environmental Policy	2	38
16	Utah	19	5	Infrequently	6	BS	Energy Policy	1	48
17	Utah	28	5	Always	5	Masters	Legal Issues	1	58
18	Utah	10	2	Infrequently	3	Bachelors	Public Admin	1	37
19	Utah	20	2	Infrequently	4	Diploma	Environmental Policy	1	38
20	Utah	30	5	Sometimes	4	Bachelors	Public Admin	2	55
21	Utah	22	2	Infrequently	4	BA	Environmental Policy	1	40
22	Utah	24	3	Sometimes	2	Ph.D.	Energy Policy	2	53
23	Utah	24	6	Always	4	MPP	Economic Policy	2	53
24	Utah	16	3	Sometimes	3	Ph.D.	Legal Issues	2	46

**Congratulations! You have just kept multiple observations in STATA!**

Both the “drop” and “keep” commands should not be used with small datasets that are easy to manage for the primary reason that once they are implemented, expunged data are not recoverable. If you wish to analyze a subset of data within a manageable dataset, there are other ways to do so without dropping values (namely using the “if” add-on, which we will come to later).

### STATA COMMAND A.3.3:

*Code:* “**keep in value1/value2**”, where value1 is the first number of your observation range and value2 is the final number of the observation range.

*Output produced:* Keeps multiple observations in STATA.

*Caveat:* Once you keep observations, you CAN NOT retrieve those you expunge. Make sure to save a master file before you do so.

Once you define which observations you desire to keep, you can then move on to coding string variables which STATA is unable to quantitatively analyze. Most data management can be done using some variation of the last three commands of this module: “encode”, “replace”, and “generate”. The first command codifies string variables (i.e. makes words numeric), the second replaces numerical/missing values with values specified by the researcher, and the third creates numerical variables from those existing within your dataset. Going back to the master file, begin with the encode command. Say we are interested in numerically coding a civil servant’s field of expertise. To do so, you must specify a new variable name to STATA that you wish to create (let’s call the codified version of “expertise” “expertisevalue”). Type the following command into the STATA command box: “encode expertise, generate(expertisevalue)”. Open up your data editor, you should see the following:

expertisevalue[1] = 3						
frequency	percentage	degree	expertise	gender	age	expertisevalue
equently	7	BA	Environmental Policy	2	33	Environmental Policy
equently	4	Bachelors	Public Admin	1	59	Public Admin
ometimes	4	MPA	Legal Issues	2	51	Legal Issues
ometimes	3	MPA	Legal Issues	1	66	Legal Issues
ometimes	4	PhD	Legal Issues	2	39	Legal Issues
equently	1	Bachelors	Environmental Policy	2	56	Environmental Policy
Always	4	Doctorate	Energy Policy	1	37	Energy Policy
ometimes	3	PhD	Economic Policy	1	53	Economic Policy
ometimes	4	MPA	Economic Policy	1	35	Economic Policy
equently	3	MPP	Economic Policy	1	45	Economic Policy
ometimes	3	Doctorate	Environmental Policy	2	46	Environmental Policy
equently	4	MPP	Economic Policy	1	56	Economic Policy
ometimes	3	BS	Legal Issues	2	30	Legal Issues
ometimes	5	MPA	Economic Policy	2	45	Economic Policy
Always	5	Doctorate	Environmental Policy	1	50	Environmental Policy
ometimes	5	MPA	Economic Policy	2	48	Economic Policy
Always	4	Doctorate	Energy Policy	1	36	Energy Policy
Always	4	Doctorate	Environmental Policy	2	51	Environmental Policy
Always	6	Masters	Legal Issues	1	51	Legal Issues
ometimes	5	Doctorate	Energy Policy	1	37	Energy Policy
equently	5	Diploma	Public Admin	1	54	Public Admin

**Congratulations! You have just numerically codified a nominal string variable in STATA!**

Notice that in your new variable column, the same labels are attached to each cell, yet when you highlight the blue cell, a numerical value emerges. This means that STATA treats this category as the specified numerical value (in this case, the expertise category of environmental policy has received a numerical value of 3). Once you have encoded a string variable, it is possible to conduct relevant regression analysis with your created data (i.e. multinomial logistic regression) or create dummy variables for further data analysis.

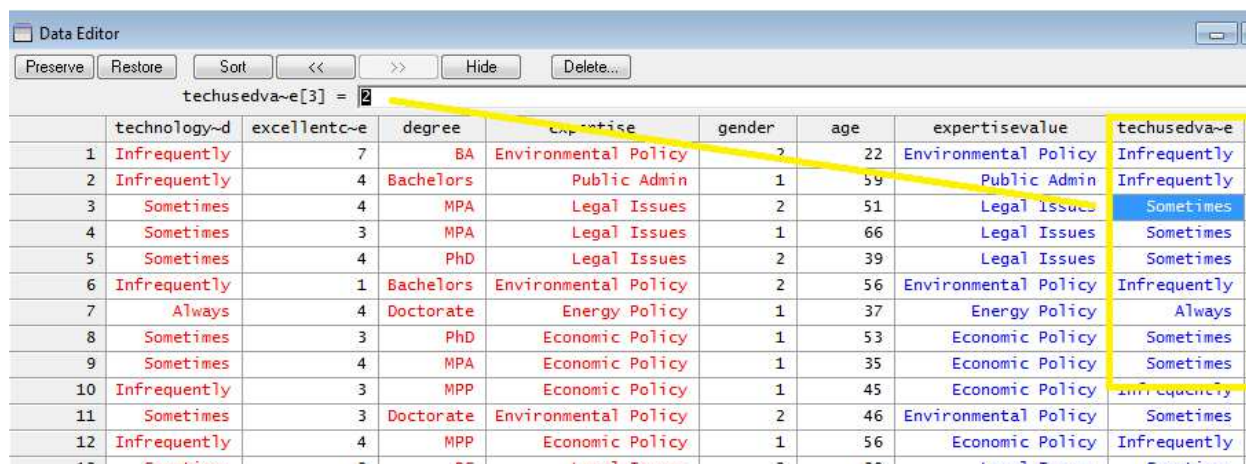
#### STATA COMMAND A.4.1:

*Code:* “**encode var1, generate(newvar1)**”, where var1 is a string variable, and newvar1 is the numerically coded version of var1.

*Output produced:* Numerically codifies string variables in STATA.

*Caveat:* Unless specified otherwise, string data is assigned values based on alphabetical order.

The one caveat about the encode command is that, if used in isolation, it will assign numerical values based upon alphabetical order. This is problematic if your string variable is an ordinal one (i.e. one where numerical ordering matters). For example, if we encoding the “technologyuse” variable, rather than assigning a coding of 1, 2, and 3 to “Always”, “Sometimes” and “Infrequently”, in their proper order, STATA would assign a value of 1 to “Always”, 2 to “Infrequently” and 3 to “Sometimes”, as this is the alphabetical order of these categories. Lucky, the encode command comes with a pre-command that enables you to specify which values you want assigned to which categories of a string variable, if relevant. To create an ordinal ranking scale for “technologyuse”, first type the following command into the STATA command box: “label define techusedvalue 1 Always 2 Sometimes 3 Infrequently” (note: STATA is case sensitive so you must type the categorical values verbatim, i.e. including capital letters, in order for the code values to register). This will NOT create a variable, but it will specify to STATA that the variable you create afterward (which MUST be named “techusedvalue”) will have these coding values for the specified category. Immediately after you type in the code above, type the following command into the STATA command editor: “encode technologyuse, generate(techusedvalue)”. Open the data editor, you should see the following new variable:



The screenshot shows the STATA Data Editor window. At the top, there's a toolbar with buttons: Preserve, Restore, Sort, <<, >>, Hide, and Delete... Below the toolbar, the command line shows 'techusedvalue[3] = 2'. The main area is a table with 13 rows and 9 columns. The columns are: technology~d, excellentc~e, degree, expertise, gender, age, expertisevalue, and techusedva~e. The 'techusedva~e' column is highlighted in yellow. A yellow arrow points from the command line to the value '2' in the 'techusedva~e' column of the first row.

	technology~d	excellentc~e	degree	expertise	gender	age	expertisevalue	techusedva~e
1	Infrequently	7	BA	Environmental Policy	2	22	Environmental Policy	Infrequently
2	Infrequently	4	Bachelors	Public Admin	1	59	Public Admin	Infrequently
3	Sometimes	4	MPA	Legal Issues	2	51	Legal Issues	Sometimes
4	Sometimes	3	MPA	Legal Issues	1	66	Legal Issues	Sometimes
5	Sometimes	4	PhD	Legal Issues	2	39	Legal Issues	Sometimes
6	Infrequently	1	Bachelors	Environmental Policy	2	56	Environmental Policy	Infrequently
7	Always	4	Doctorate	Energy Policy	1	37	Energy Policy	Always
8	Sometimes	3	PhD	Economic Policy	1	53	Economic Policy	Sometimes
9	Sometimes	4	MPA	Economic Policy	1	35	Economic Policy	Sometimes
10	Infrequently	3	MPP	Economic Policy	1	45	Economic Policy	Infrequently
11	Sometimes	3	Doctorate	Environmental Policy	2	46	Environmental Policy	Sometimes
12	Infrequently	4	MPP	Economic Policy	1	56	Economic Policy	Infrequently
13	Sometimes	3	PhD	Legal Issues	2	30	Legal Issues	Sometimes

**Congratulations! You have just numerically codified an ordinal string variable in STATA!**

As you click on the cell contents of your new variable, you should notice the values of the categories overlap with those you created in the previous label command. One word of caution however; once you create a label command for a new variable, those values will stick with the new variable even if you drop it from the dataset. Hence, if you type in the wrong coding, you will have to specify a different variable name doing it the second (correct) time around.

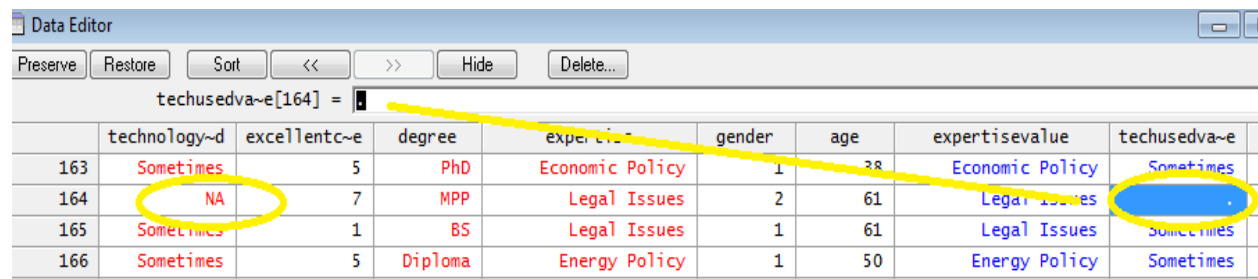
## STATA COMMAND A.4.2:

*Code:* “**label define newvar 1 category1 2 category2 3 category3...**”, where “newvar” is the new variable to be created, 1 is the numerical value you wish to assign to category1, 2 is the numerical value you wish to assign to category2, and so on.

“**encode var1, generate(newvar)**”, where var1 is a string variable, and newvar1 is the ordinally coded version of var1.

*Output produced:* Codifies string variables in STATA using a specified ordinal scale.

Now that you have codified your “technologyuse” variable, you may notice that a value of 4 was assigned to the “NA” category. As you will learn in the ordinal logistic regression less, non-responses/not-applicable/don’t-know responses should be treated like a missing value – no useful information is conveyed via these types of categories. You therefore want to replace these responses with periods, STATA’s code for missing values with the “replace” command. To recode the “NA” category for the “techusedvalue” variable, type the following command into the command box: “replace techusedvalue=. if techusedvalue==4”. Under the command in the output box, STATA should tell you how many observations of “techusedvalue” were replaced by “.” (in this case 17). Open up the data editor and scroll down to a cell that previously had an “NA” coding in it. You should see the following:



	technology~d	excellentc~e	degree	expertise	gender	age	expertisevalue	techusedva~e
163	Sometimes	5	PhD	Economic Policy	1	38	Economic Policy	Sometimes
164	NA	7	MPP	Legal Issues	2	61	Legal Issues	.
165	Sometimes	1	BS	Legal Issues	1	61	Legal Issues	Sometimes
166	Sometimes	5	Diploma	Energy Policy	1	50	Energy Policy	Sometimes

**Congratulations! You have just replaced a numerical value with a missing value in STATA!**

The replace command can also be used to recode numerical values with other numerical values. Taking the “gender” variable, let’s recode this into a dummy variable coding of 0 and 1 rather than 1 and 2 (you may need to “destring” gender beforehand).<sup>34</sup> Let’s replace the female coding of 2 in the gender variable with that of 0 – hence our gender variable will embody the value of 1 for “male” and 0 for “female”. Type “replace gender=0 if gender==2”. After you type in the command to STATA you should note that 580 cells have been replaced with 0 (indicating that there are 580 women in the sample). Open your data editor, and you should see the following coding for gender:

<sup>34</sup> Dummy variables are those whose outcome is binary (i.e. can only embody two values such as “yes/no”, “male/female”, etc.). When including dummy variables in regression analysis, it is common practice to code them as 0/1.

Data Editor

Preserve

Restore

Sort

<<

>>

Hide

Delete...

years

gov

[1]

=

4

	yearsgov	opendialogue	technology~d	excellents~e	degree	expertise	gender
1	4	1	Infrequently	7	BA	Environmental Policy	0
2	32	1	Infrequently	4	Bachelors	Public Admin	1
3	21	2	Sometimes	4	MPA	Legal Issues	0
4	36	3	Sometimes	3	MPA	Legal Issues	1
5	9	3	Sometimes	4	PhD	Legal Issues	0
6	38	2	Infrequently	1	Bachelors	Environmental Policy	0
7	8	4	Always	4	Doctorate	Energy Policy	1
8	24	3	Sometimes	3	PhD	Economic Policy	1
9	6	2	Sometimes	4	MPA	Economic Policy	1
10	16	2	Infrequently	3	MPP	Economic Policy	1
11	16	3	Sometimes	3	Doctorate	Environmental Policy	0
12	27	2	Infrequently	4	MPP	Economic Policy	1
13	3	1	Sometimes	3	BS	Legal Issues	0
14	16	3	Sometimes	5	MPA	Economic Policy	0
15	32	3	Always	5	Doctorate	Environmental Policy	1
16	19	2	Sometimes	5	MPA	Economic Policy	0

**Congratulations! You have just replaced a numerical value with another numerical value in STATA!**

#### STATA COMMAND A.5.1:

*Code:* “**replace var1=. if var1==#**”, where # is the value of var1 you wish to replace with a missing observation.

*Output produced:* Replaces a numerical value with a missing value.

#### STATA COMMAND A.5.2:

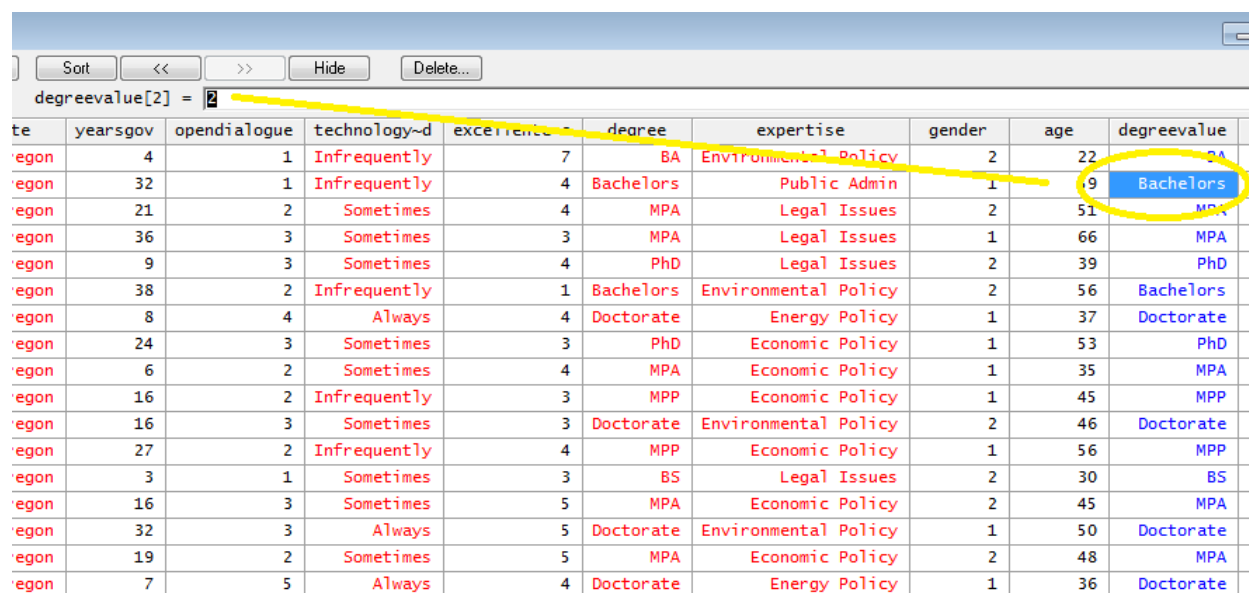
*Code:* “**replace var1=#1 if var1==#2**”, where #1 is the original value of var1, and #2 is the value you wish to replace it with.

*Output produced:* Replaces a numerical value with another numerical value.

The replace command is helpful for removing empty responses from both numerical and coded string variables, as well as replacing numerical values with other numerical ones. If used with the encode command, it can also be a helpful way to condense string values with multiple categories into fewer categories. Take the degree variable for example. Given how responses were written into the survey, there are multiple categories for a given degree. For the bachelor’s degree category, there are four different types of coding – “Bachelors”, “BA”, “B.A.” and “BS”. If you used the encode command, however, rather than condensing this into one category, STATA would codify each category separately given the different spelling (even BA and B.A.). You can use the “label”, “encode” and “replace”

command to condense these degree categories. Rather than having 12 different degree names, let's create four general categories: Diploma, Bachelors, Masters, and Doctorate.

To start the condensing process, we first want to specify to STATA the specific coding we want for our broad categories. Let's specify that we want the diploma category to have a coding of 1, the Bachelors category to have a coding of 2, the Masters category to have a coding of 3, and the Doctorate category to have a coding of four. Type the following command into the command box: "label define degreevalue 1 Diploma 2 Bachelors 3 Masters 4 Doctorate" (do not worry about creating labels for the other categories – STATA will automatically do this based on alphabetical order). Then type "encode degree, generate(degreevalue)" into the command box. Open the data editor and you should see the following:



te	yearsgov	opendialogue	technology~d	experience	degree	expertise	gender	age	degreevalue
egon	4	1	Infrequently	7	BA	Environmental Policy	2	22	2
egon	32	1	Infrequently	4	Bachelors	Public Admin	1	9	2
egon	21	2	Sometimes	4	MPA	Legal Issues	2	51	3
egon	36	3	Sometimes	3	MPA	Legal Issues	1	66	3
egon	9	3	Sometimes	4	PhD	Legal Issues	2	39	4
egon	38	2	Infrequently	1	Bachelors	Environmental Policy	2	56	2
egon	8	4	Always	4	Doctorate	Energy Policy	1	37	4
egon	24	3	Sometimes	3	PhD	Economic Policy	1	53	4
egon	6	2	Sometimes	4	MPA	Economic Policy	1	35	3
egon	16	2	Infrequently	3	MPP	Economic Policy	1	45	3
egon	16	3	Sometimes	3	Doctorate	Environmental Policy	2	46	4
egon	27	2	Infrequently	4	MPP	Economic Policy	1	56	3
egon	3	1	Sometimes	3	BS	Legal Issues	2	30	5
egon	16	3	Sometimes	5	MPA	Economic Policy	2	45	3
egon	32	3	Always	5	Doctorate	Environmental Policy	1	50	4
egon	19	2	Sometimes	5	MPA	Economic Policy	2	48	3
egon	7	5	Always	4	Doctorate	Energy Policy	1	36	4

Notice that STATA has assigned the 1, 2, 3, and 4 coding to the Diploma, Bachelors, Masters and Doctorate categories, while assigning values of 5 and higher to the other categories based upon their alphabetical order. To condense all bachelor degree categories into the "2" category, click over each category cell to determine their numerical value (hint, using the sort command on the degree variable may make this easier): B.A. should have a value of 5, BA a value of 6, and BS a value of 7. Starting with the bachelor degree category, type the following three "replace" commands into the command box to replace their values with the general Bachelors value of 2:

"replace degreevalue=2 if degreevalue==5"

"replace degreevalue=2 if degreevalue==6"

"replace degreevalue=2 if degreevalue==7"

Open your data editor and you should see the following:

store

Sort

<<

>>

Hide

Delete...

degreevalue[13] = 2

state	yearsgov	opendialogue	technology~d	excellents~e	degree	expertise	gender	age	degreevalue
Oregon	4	1	Infrequently	7	BA	Environmental Policy	2	33	Bachelors
Oregon	32	1	Infrequently	4	Bachelors	Public Admin	1	59	Bachelors
Oregon	21	2	Sometimes	4	MPA	Legal Issues	2	51	MPA
Oregon	36	3	Sometimes	3	MPA	Legal Issues	1	66	MPA
Oregon	9	3	Sometimes	4	PhD	Legal Issues	2	39	PhD
Oregon	38	2	Infrequently	1	Bachelors	Environmental Policy	2	56	Bachelors
Oregon	8	4	Always	4	Doctorate	Energy Policy	1	37	Doctorate
Oregon	24	3	Sometimes	3	PhD	Economic Policy	1	53	PhD
Oregon	6	2	Sometimes	4	MPA	Economic Policy	1	35	MPA
Oregon	16	2	Infrequently	3	MPP	Economic Policy	1	45	MPP
Oregon	16	3	Sometimes	3	Doctorate	Environmental Policy	2	46	Doctorate
Oregon	27	2	Infrequently	4	MPP	Economic Policy	1	56	MPP
Oregon	3	1	Sometimes	3	BS	Legal Issues	2	30	Bachelors
Oregon	16	3	Sometimes	5	MPA	Economic Policy	2	45	MPA
Oregon	32	3	Always	5	Doctorate	Environmental Policy	1	50	Doctorate
Oregon	19	2	Sometimes	5	MPA	Economic Policy	2	48	MPA
Oregon	7	5	Always	4	Doctorate	Energy Policy	1	36	Doctorate
Oregon	33	3	Always	4	Doctorate	Environmental Policy	2	51	Doctorate
Oregon	24	2	Always	6	Masters	Legal Issues	1	51	Masters
Oregon	8	4	Sometimes	5	Doctorate	Energy Policy	1	37	Doctorate
Oregon	29	7	Infrequently	5	Diploma	Public Admin	1	54	Diploma
Oregon	19	3	Sometimes	3	Doctorate	Energy Policy	1	48	Doctorate
Oregon	31	2	Always	4	Doctorate	Public Admin	2	56	Doctorate
Oregon	22	2	Always	5	Doctorate	Environmental Policy	2	40	Doctorate
Oregon	18	1	Always	5	Masters	Public Admin	2	43	Masters

Notice that the BA, BS and B.A. degrees in the “degreevalue” column have now been replaced with the general “Bachelors” category with a coding of 2. You can repeat this process, condensing the MBA, MPP and MPA categories into the general Masters category, and condensing the Ph.D. and PhD categories into the general doctorate category. To condense all Masters degrees into the general “Masters” category (coding of 3), type the following commands into the command box:

“replace degreevalue=3 if degreevalue==8” (to recode the MBA degree)  
 “replace degreevalue=3 if degreevalue==9” (to recode the MPA degree)  
 “replace degreevalue=3 if degreevalue==10” (to recode the MPP degree)

To condense the doctoral degrees into the general “Doctorate” category (coding of 4), type the following commands into the command box:

“replace degreevalue=4 if degreevalue==11” (to recode the Ph.D. degree)  
 “replace degreevalue=4 if degreevalue==12” (to recode the PhD degree)

After typing these commands, your data editor should contain ONLY the four general degree categories:

ort << >> Hide Delete...

state[7] = Oregon

yearsgov	opendialogue	technology~d	excellents~e	degree	expertise	gender	age	degreevalue
4	1	Infrequently	7	BA	Environmental Policy	2	22	Bachelors
32	1	Infrequently	4	Bachelors	Public Admin	1	59	Bachelors
21	2	Sometimes	4	MPA	Legal Issues	2	51	Masters
36	3	Sometimes	3	MPA	Legal Issues	1	66	Masters
9	3	Sometimes	4	PhD	Legal Issues	2	39	Doctorate
38	2	Infrequently	1	Bachelors	Environmental Policy	2	56	Bachelors
8	4	Always	4	Doctorate	Energy Policy	1	37	Doctorate
24	3	Sometimes	3	PhD	Economic Policy	1	53	Doctorate
6	2	Sometimes	4	MPA	Economic Policy	1	35	Masters
16	2	Infrequently	3	MPP	Economic Policy	1	45	Masters
16	3	Sometimes	3	Doctorate	Environmental Policy	2	46	Doctorate
27	2	Infrequently	4	MPP	Economic Policy	1	56	Masters
3	1	Sometimes	3	BS	Legal Issues	2	30	Bachelors
16	3	Sometimes	5	MPA	Economic Policy	2	45	Masters
32	3	Always	5	Doctorate	Environmental Policy	1	50	Doctorate
19	2	Sometimes	5	MPA	Economic Policy	2	48	Masters
7	5	Always	4	Doctorate	Energy Policy	1	36	Doctorate

The final data management command we will use today is the “generate” command. This command is very versatile; you can replicate variables, creating new coding for categorical variables, as well as creating new variables that are functions of one or more variables. Let’s start with replicating a variable. If you intend to manipulate a coded variable with the replace command, it may be beneficial to preserve a copy of its original version. Say we were interested in recoding our opendialogue variable, but wanted to create a “back-up” copy in case we entered our coding incorrectly. To replicate a variable, you must create a new name for the copy – we will call this “opendialogue2”. Type the following command into the STATA command box: “generate opendialogue2=opendialogue”. Open the data editor and you should see the following:

Data Editor


Preserve Restore Sort << >> Hide Delete...

opendialogue[1] = 1

	opendialogue	technology~d	excellents~e	degree	expertise	gender	age	degreevalue	opendialog~2
1	1	Infrequently	7	BA	Environmental Policy	2	22	Bachelors	1
2	1	Infrequently	4	Bachelors	Public Admin	1	59	Bachelors	1
3	2	Sometimes	4	MPA	Legal Issues	2	51	Masters	2
4	3	Sometimes	3	MPA	Legal Issues	1	66	Masters	3
5	3	Sometimes	4	PhD	Legal Issues	2	39	Doctorate	3
6	2	Infrequently	1	Bachelors	Environmental Policy	2	56	Bachelors	2
7	4	Always	4	Doctorate	Energy Policy	1	37	Doctorate	4
8	3	Sometimes	3	PhD	Economic Policy	1	53	Doctorate	3
9	2	Sometimes	4	MPA	Economic Policy	1	35	Masters	2
10	2	Infrequently	3	MPP	Economic Policy	1	45	Masters	2
11	3	Sometimes	3	Doctorate	Environmental Policy	2	46	Doctorate	3
12	2	Infrequently	4	MPP	Economic Policy	1	56	Masters	2
13	1	Sometimes	3	BS	Legal Issues	2	30	Bachelors	1
14	3	Sometimes	5	MPA	Economic Policy	2	45	Masters	3
15	3	Always	5	Doctorate	Environmental Policy	1	50	Doctorate	3
16	2	Sometimes	5	MPA	Economic Policy	2	48	Masters	2
17	5	Always	4	Doctorate	Energy Policy	1	36	Doctorate	5

**Congratulations! You have just replicated a variable in STATA!**

STATA can also create new codifications of pre-existing variables. Say rather than having a 1-4 coding of degree obtained, we would rather express a respondent's education in a 1-3 fashion: 1 for possessing a Diploma, 2 for possessing a Bachelor's degree and 3 for possessing an advanced degree (Masters or a Doctorate). When you create an alternative coding for a variable, you should have multiple "generate" commands that are conditional on the original variable's value. To make commands in STATA conditional, you must type "if" and then the specification after it. In the case of our new degree variable, call this "degreevalue2", type in the following conditional command into the STATA editor: "generate degreevalue2=1 if degreevalue==1". This will generate the new variable only for the "Diploma" category. Open your data editor and you should see the following:



xcellentc~e	degree	expertise	gender	age	degreevalue	opendialog~2	degreevalue2
7	BA	Environmental Policy	2	22	Bachelors	1	.
4	Bachelors	Public Admin	1	59	Bachelors	1	.
4	MPA	Legal Issues	2	51	Masters	2	.
3	MPA	Legal Issues	1	66	Masters	3	.
4	PhD	Legal Issues	2	39	Doctorate	3	.
1	Bachelors	Environmental Policy	2	56	Bachelors	2	.
4	Doctorate	Energy Policy	1	37	Doctorate	4	.
3	PhD	Economic Policy	1	53	Doctorate	3	.
4	MPA	Economic Policy	1	35	Masters	2	.
3	MPP	Economic Policy	1	45	Masters	2	.
3	Doctorate	Environmental Policy	2	46	Doctorate	3	.
4	MPP	Economic Policy	1	56	Masters	2	.
3	BS	Legal Issues	2	30	Bachelors	1	.
5	MPA	Economic Policy	2	45	Masters	3	.
5	Doctorate	Environmental Policy	1	50	Doctorate	3	.
5	MPA	Economic Policy	2	48	Masters	2	.
4	Doctorate	Energy Policy	1	36	Doctorate	5	.
4	Doctorate	Environmental Policy	2	51	Doctorate	3	.
6	Masters	Legal Issues	1	51	Masters	2	.
5	Doctorate	Energy Policy	1	37	Doctorate	4	.
5	Diploma	Public Admin	1	54	Diploma	7	1
3	Doctorate	Energy Policy	1	48	Doctorate	3	.
4	Doctorate	Public Admin	2	56	Doctorate	2	.
5	Doctorate	Environmental Policy	2	40	Doctorate	2	.

Notice that the new variable only has a value for the category of "degreevalue" which we conditionally specified within the command – 1. In order to complete the creation of this new variable, we must use the replace command. Type the following two commands into the STATA command box:

"replace degreevalue2=2 if degreevalue==2" (this codifies our new variable for the Bachelors category)  
 "replace degreevalue2=3 if degreevalue==3 | degreevalue==4" (this codifies our new variable for the Advanced degree category).

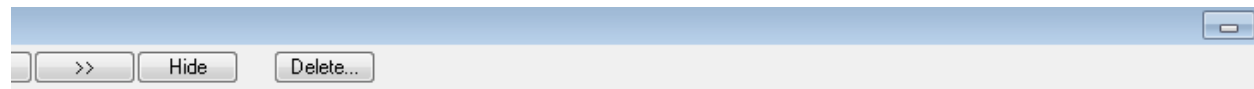
Open the data editor and you should see the following:



its square root or quadratic value), or multiple variables. Commonly used mathematical functions in STATA include:

- *addition*: +
- *subtraction*: -
- *product*: \*
- *quotient*: /
- *raise to the x power*: ^x
- *square root*: sqrt( )
- *absolute power*: abs( )
- *natural log*: ln( )

Let's start by creating a new variable, age2, which is the squared value of a civil servant's age. To do so, type the following command in the command box: "generate age2 = age^2". Open the data editor, you should see the following:



se	gender	age	degreevalue	opendialog~2	degreevalue2	under30adv~d	age2
l Policy	2	22	Bachelors	1	2	0	484
ic Admin	1	59	Bachelors	1	2	0	3481
l Issues	2	51	Masters	2	3	0	2601
l Issues	1	66	Masters	3	3	0	4356
l Issues	2	39	Doctorate	3	3	0	1521
l Policy	2	56	Bachelors	2	2	0	3136
y Policy	1	37	Doctorate	4	3	0	1369
c Policy	1	53	Doctorate	3	3	0	2809
c Policy	1	35	Masters	2	3	0	1225
c Policy	1	45	Masters	2	3	0	2025
l Policy	2	46	Doctorate	3	3	0	2116
c Policy	1	56	Masters	2	3	0	3136
l Issues	2	30	Bachelors	1	2	0	900
c Policy	2	45	Masters	3	3	0	2025
l Policy	1	50	Doctorate	3	3	0	2500
c Policy	2	48	Masters	2	3	0	2304
y Policy	1	36	Doctorate	5	3	0	1296
l Policy	2	51	Doctorate	3	3	0	2601
l Issues	1	51	Masters	2	3	0	2601
y Policy	1	37	Doctorate	4	3	0	1369
ic Admin	1	54	Diploma	7	1	0	2916
y Policy	1	48	Doctorate	3	3	0	2304

**Congratulations! You have just created a new variable in STATA!**

New variables can also be functions of multiple variables. Say we are interested in determining what proportion of each respondent's life has been spent working in government. To do so, we must divide "yearsgov" by "age" using the following command: "generate lifeingov = yearsgov/age". Open the data editor and you should see the following:

Data Editor

Preserve Restore Sort << >> Hide Delete...

degree[1] =

	degree	expertise	gender	age	degreevalue	opendialog~2	degreevalue2	under30adv~d	age2	lifeingov
1	BA	Environmental Policy	2	22	Bachelors	1	2	0	484	.1818182
2	Bachelors	Public Admin	1	59	Bachelors	1	2	0	3481	.5423729
3	MPA	Legal Issues	2	51	Masters	2	3	0	2601	.4117647
4	MPA	Legal Issues	1	66	Masters	3	3	0	4356	.5454546
5	PhD	Legal Issues	2	39	Doctorate	3	3	0	1521	.2307692
6	Bachelors	Environmental Policy	2	56	Bachelors	2	2	0	3136	.6785714
7	Doctorate	Energy Policy	1	37	Doctorate	4	3	0	1369	.2162162
8	PhD	Economic Policy	1	53	Doctorate	3	3	0	2809	.4528302
9	MPA	Economic Policy	1	35	Masters	2	3	0	1225	.1714286
10	MPP	Economic Policy	1	45	Masters	2	3	0	2025	.3555556
11	Doctorate	Environmental Policy	2	46	Doctorate	3	3	0	2116	.3478261
12	MPP	Economic Policy	1	56	Masters	2	3	0	3136	.4821429
13	BS	Legal Issues	2	30	Bachelors	1	2	0	900	.1
14	MPA	Economic Policy	2	45	Masters	3	3	0	2025	.3555556
15	Doctorate	Environmental Policy	1	50	Doctorate	3	3	0	2500	.64
16	MPA	Economic Policy	2	48	Masters	2	3	0	2304	.3958333
17	Doctorate	Energy Policy	1	36	Doctorate	5	3	0	1296	.1944444
18	Doctorate	Environmental Policy	2	51	Doctorate	3	3	0	2601	.6470588
19	Masters	Legal Issues	1	51	Masters	2	3	0	2601	.4705882
20	Doctorate	Energy Policy	1	37	Doctorate	4	3	0	1369	.2162162
21	Diploma	Public Admin	1	54	Diploma	7	1	0	2916	.537037
22	Doctorate	Energy Policy	1	48	Doctorate	3	3	0	2304	.3958333
23	Doctorate	Public Admin	2	56	Doctorate	2	3	0	3136	.5535714
24	Doctorate	Environmental Policy	2	40	Doctorate	2	3	0	1600	.55
25	Masters	Public Admin	2	43	Masters	1	3	0	1849	.4186046
26	Bachelors	Environmental Policy	2	54	Bachelors	1	2	0	2916	.6666667
27	PhD	Economic Policy	1	52	Doctorate	3	3	0	2704	.4423077
28	Doctorate	Energy Policy	1	73	Doctorate	1	3	0	5329	.6027398

Notice the creation of your new variable in the final column. When you create this variable, you can also multiply this value by 100 if you want to express is as percent on a 0-100 scale, but make sure to put parentheses around the entire quotient else it will multiply only the denominator by 100: “generate lifeingov = (yearsgov/age)\*100”.

## STATA COMMAND A.6.1:

*Code:* “**generate newvar1=var1**”

*Output produced:* Replicates the variable var1.

### STATA COMMAND A.6.2:

Code: “**generate newvar1=var1 if var1==#1**”  
“**replace newvar1=#2 if var1==#2 ...**”

*Supplementary Code (after “if”):*

- or: |
- and: &
- equals to: ==
- not equals to: !=
- greater than: >
- less than: <

*Output produced:* Creates a new coding of an/multiple old variable(s) based upon conditional specification using the supplementary coding.

### STATA COMMAND A.6.3:

Code: “**generate newvar1=f(var1, var2, ... )**”.

*Mathematical codes:*

- addition: +
- subtraction: -
- product: \*
- quotient: /
- raise to the  $x$  power:  $^x$
- square root:  $\text{sqrt}( )$
- absolute power:  $\text{abs}( )$
- natural log:  $\ln( )$

*Output produced:* Creates a new variable that is a specified function of var1, var2, etc.

---

### Practice Problems:

You are presented with a dataset elections statements database, created and used by MPP student Michael Nash, with information on political candidates that had public statements judged by the fact checking agency Politifact. Each observation is a recorded statement by the politician that was selected and judged for factual accuracy by Politifact (hence politicians with more than one observation had multiple statements judged by the independent organizations). The information you are presented with in the excel spreadsheet includes:

- name: The name of the candidate.
- margin: The margin of victory for the candidate in the 2009/2010 election.
- leadership: Whether the political candidate held a leadership position in Congress, 1 for yes, 0 for no.
- state: The state where the candidate sought office.
- office\_sought: The political office the candidate was running for, S for Senate, H for House and Gov for Governor.
- racestatus: The type of race the candidate was entering, either as an incumbent, challenger, or an open office (i.e. one where an incumbent has decided against running again).
- politifact\_judgement: Politifact's rating of the candidate's public statement, ranging from "Pants onFire" (an extreme lie) to "True".
- year\_of\_birth: The year the candidate was born.
- religion: The candidate's religion (note that "NA" implies that the candidate is Non-Affiliated to a church. "#NULL!" indicates a non-recorded value).
- gender: the gender of the candidate.
- result: The result of the race, W for win, L for loss, #NULL! if unrecorded.

Lab Practice Problem 1: Copy/paste the spreadsheet into excel. Destrung the "margin" variable, so all non-numerical values (i.e. #NULL!) are removed from this variable.

Lab Practice Problem 2: Sort the spreadsheet by the state in which the candidate is running. How many statements are made by Florida politicians?

Lab Practice Problem 3: SAVE A COPY OF THIS DATASET BEFORE YOU COMPLETE THIS PROBLEM. Drop all statements that were made by political leaders from the database.

Lab Practice Problem 4: Codify the state which the candidate is running (note: order does not matter)

Lab Practice Problem 5: Codify the office for which the candidate is running (note: order does not matter)

Lab Practice Problem 6: Codify Politifact's judgment of the statement, ranging from 1 (True) to 6 (PantsonFire), on an ordinal scale (remember, STATA is extremely sensitive to letter case, so make sure you type in the category exactly as it appears in the data editor – the sort code may be of help).

Lab Practice Problem 7: Codify the political party of the candidate. Recode the “republican” category as a 0 rather than a 2.

Lab Practice Problem 8: Codify the race status of the candidate on a 1/0 binary scale (1 for win and 0 for loss). Expunge data that do not conform to either of these categories.

Lab Practice Problem 9: Create a new variable that summarizes the religious views of the candidates according to the following broad categories: Jewish (coding of 1), Catholic (coding of 2), Protestant (coding of 3), Mormonism (coding of 4), and Non-Affiliated (coding of 5). *Hint: the sort command will help you and #NULL! responses should NOT receive a numerical coding.*

Lab Practice Problem 10: Using the candidate's year of birth, create a variable that approximates his/her age in the 2010 election.

Lab Practice Problem 11: Create a new dummy variable where a coding of 1 indicates a female running for the Senate, 0 if otherwise.

## Appendix II: Useful Links

---

In case you are having difficulties locating a command within STATA, or want to learn more about a command that you have learned, you can consult the following sources:

1. Princeton University Data and Statistical Services:  
[http://dss.princeton.edu/online\\_help/stats\\_packages/stata/](http://dss.princeton.edu/online_help/stats_packages/stata/)
2. UCLA Resources to Help You Learn and Use STATA:  
<http://www.ats.ucla.edu/stat/stata/>

3. Google Search

Given its popularity, STATA has multiple online forums to help you with specific commands and coding. If there is something you want to do in the program, but you do not know how, simply type what you want to do and “STATA” into the Google search engine, and you will be provided with numerous webpages that will provide assistance.