

Stata (Level 1 – Data) Workshop

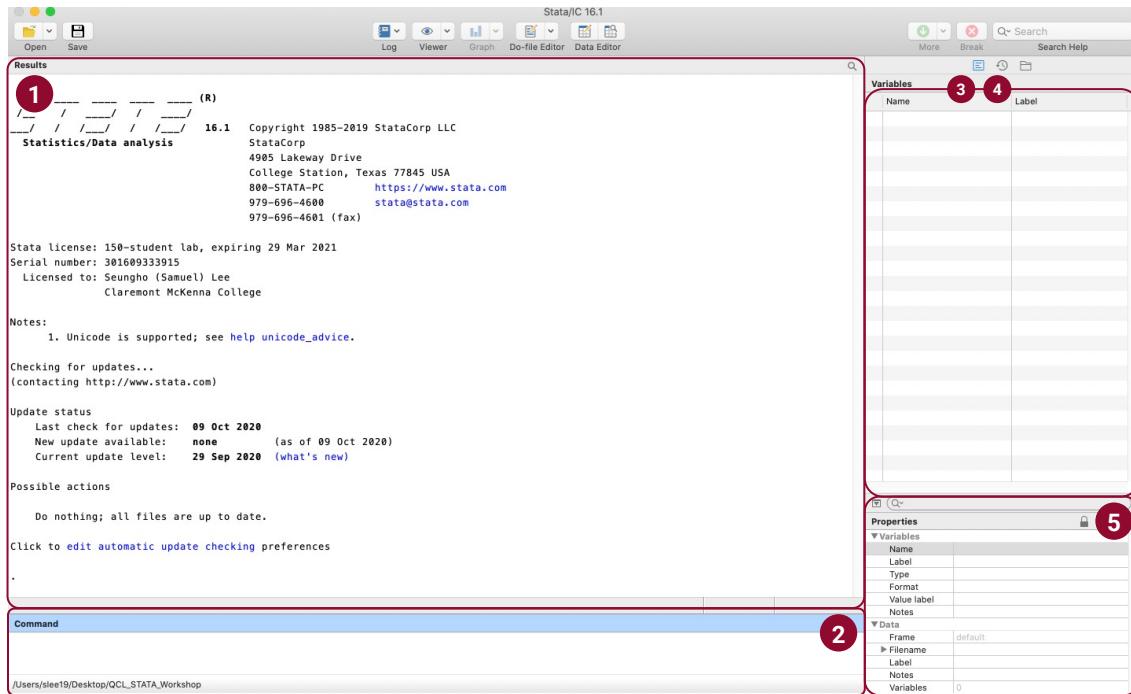
Quantitative and Computing Lab (QCL)

Before We Begin

1. Sign-in Link
2. Retrieve Workshop file at (https://github.com/CMC-QCL/STATA-L1-Workshop/blob/students/Stata_Workshop_Files.zip)
 - After you have downloaded “Workshop_Files” compressed file, extract the folder on “Desktop”
3. Learning Objective: Ensure that all participants gain some level of confidence using Stata and statistical analysis
4. Tip: Open .do, PDF, and any other files that I am using throughout the workshop to make sure that you are following along.

Stata Console – Main Window

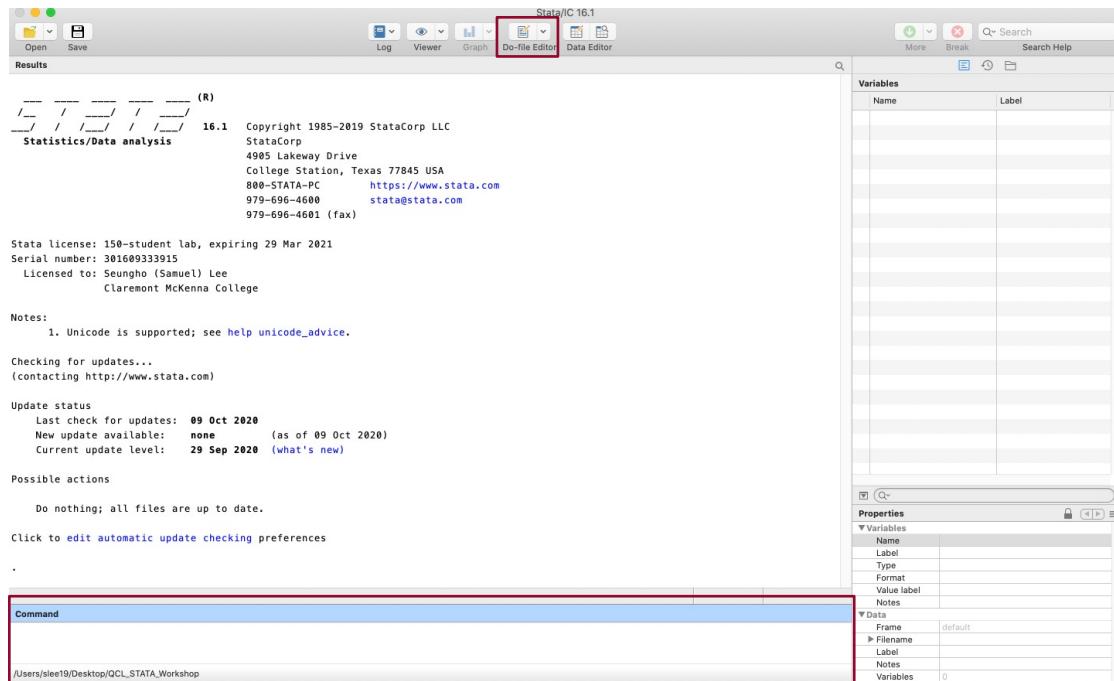
Stata window largely consists of command history, command line, output window, variable list, and data format.



- 1 Results:** displays commands and resulting outputs from current session
- 2 Command Line:** a window where a user enters a command
- 3 Variable List:** lists all variables specified in active session
- 4 Command History:** shows every command performed in active session
- 5 Data Format:** detailed description of highlighted variable (e.g., type)

Stata Console – How It Works

While Stata is truly “interactive,” users can also run a program as a “batch” mode (running commands listed on a file)

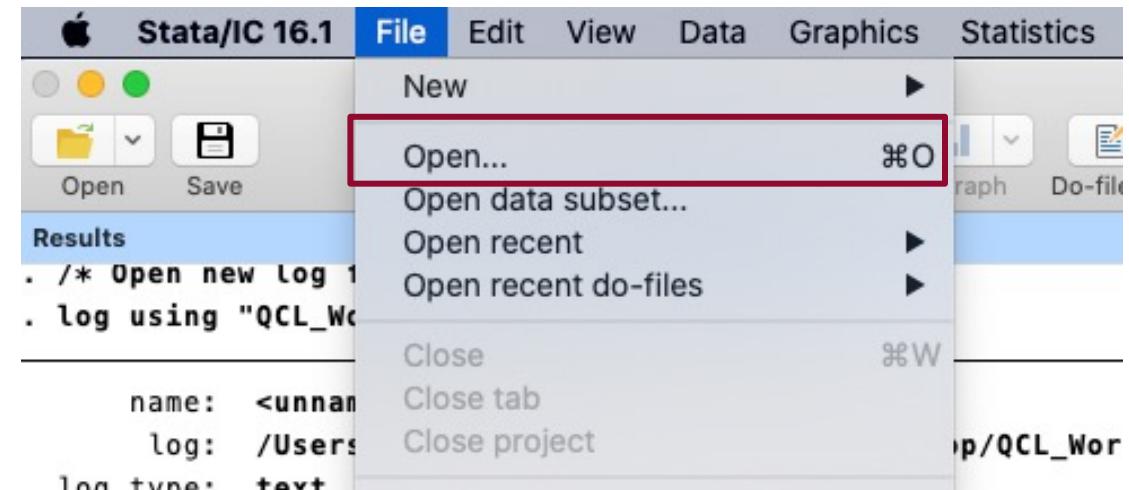


- 1. Interactive Use:** typing Stata commands *directly* on the Command window to produce results.
- 2. Batch Mode:** All commands are compiled in a file (called *Do-Files*), which Stata reads and executes.

During this workshop, we are going to use Do-File “.do” to import and explore data and conduct relevant analysis

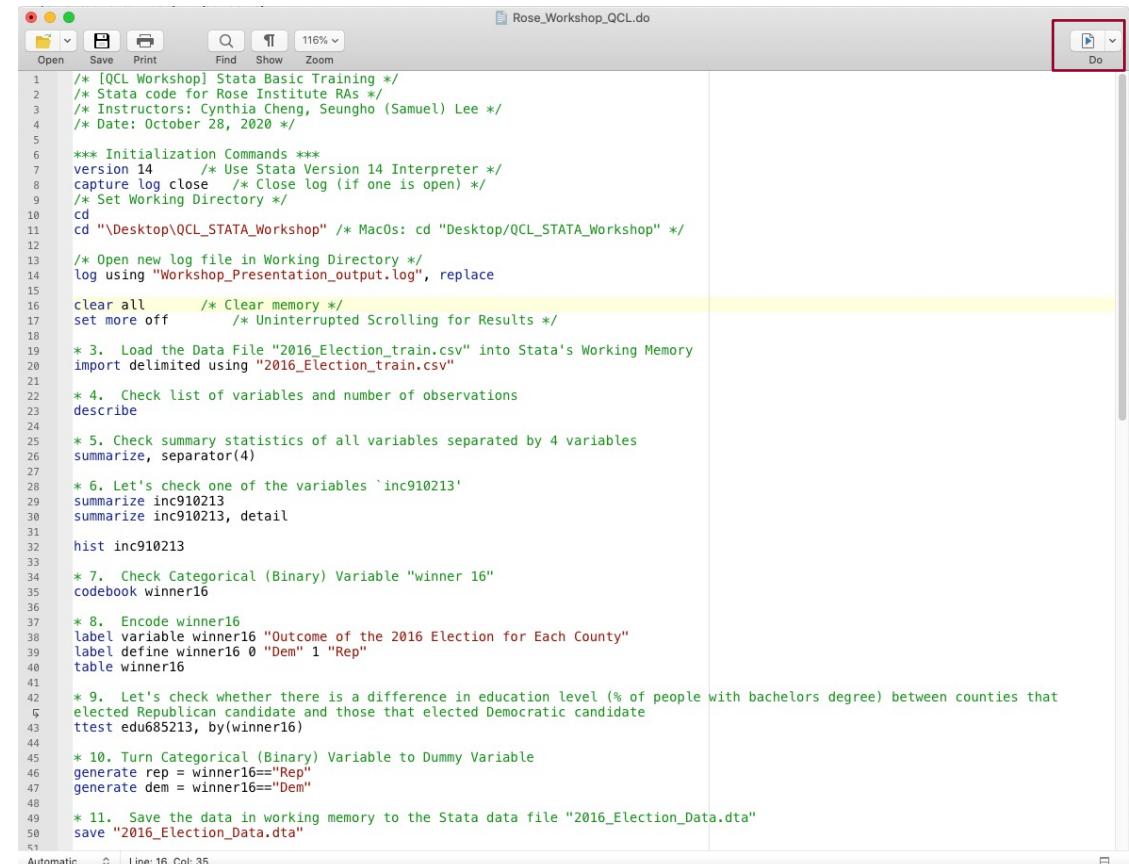
Stata Console – Do-Files (Open)

1. Press “File” on a top menu bar
2. Select “Open...”
3. Go to “Workshop_Files” folder we downloaded on “Desktop” folder
4. Open “Stata_Workshop_QCL.do” file



Stata Console – Do-Files

- ▶ Think of it as a set of *instructions* for Stata to conduct without manual input
- ▶ It is a good practice to compile *Do-File* since doing so allows others to **reproduce**
- ▶ **Comment:**
 - ▶ /* [INSERT COMMENT] */: comments a specified section
 - ▶ *: comments a whole line
- ▶ Press **boxed icon** shown on the screenshot to execute the file



```

1  /* [QCL Workshop] Stata Basic Training */
2  /* Stata code for Rose Institute RAs */
3  /* Instructors: Cynthia Cheng, Seungho (Samuel) Lee */
4  /* Date: October 28, 2020 */
5
6  *** Initialization Commands ***
7  version 14 /* Use Stata Version 14 Interpreter */
8  capture log close /* Close log (if one is open) */
9  /* Set Working Directory */
10 cd
11 cd "\Desktop\QCL_STATA_Workshop" /* MacOs: cd "Desktop/QCL_STATA_Workshop" */
12
13 /* Open new log file in Working Directory */
14 log using "Workshop_Presentation_output.log", replace
15
16 clear all /* Clear memory */
17 set more off /* Uninterrupted Scrolling for Results */
18
19 * 3. Load the Data File "2016_Election_train.csv" into Stata's Working Memory
20 import delimited using "2016_Election_train.csv"
21
22 * 4. Check list of variables and number of observations
23 describe
24
25 * 5. Check summary statistics of all variables separated by 4 variables
26 summarize, separator(4)
27
28 * 6. Let's check one of the variables `inc910213'
29 summarize inc910213
30 summarize inc910213, detail
31
32 hist inc910213
33
34 * 7. Check Categorical (Binary) Variable "winner_16"
35 codebook winner16
36
37 * 8. Encode winner16
38 label variable winner16 "Outcome of the 2016 Election for Each County"
39 label define winner16 0 "Dem" 1 "Rep"
40 table winner16
41
42 * 9. Let's check whether there is a difference in education level (% of people with bachelors degree) between counties that
43 elected Republican candidate and those that elected Democratic candidate
44 ttest edu685213, by(winner16)
45
46 * 10. Turn Categorical (Binary) Variable to Dummy Variable
47 generate rep = winner16=="Rep"
48 generate dem = winner16=="Dem"
49
50 * 11. Save the data in working memory to the Stata data file "2016_Election_Data.dta"
51 save "2016_Election_Data.dta"
52

```

Automatic Line: 16, Col: 35

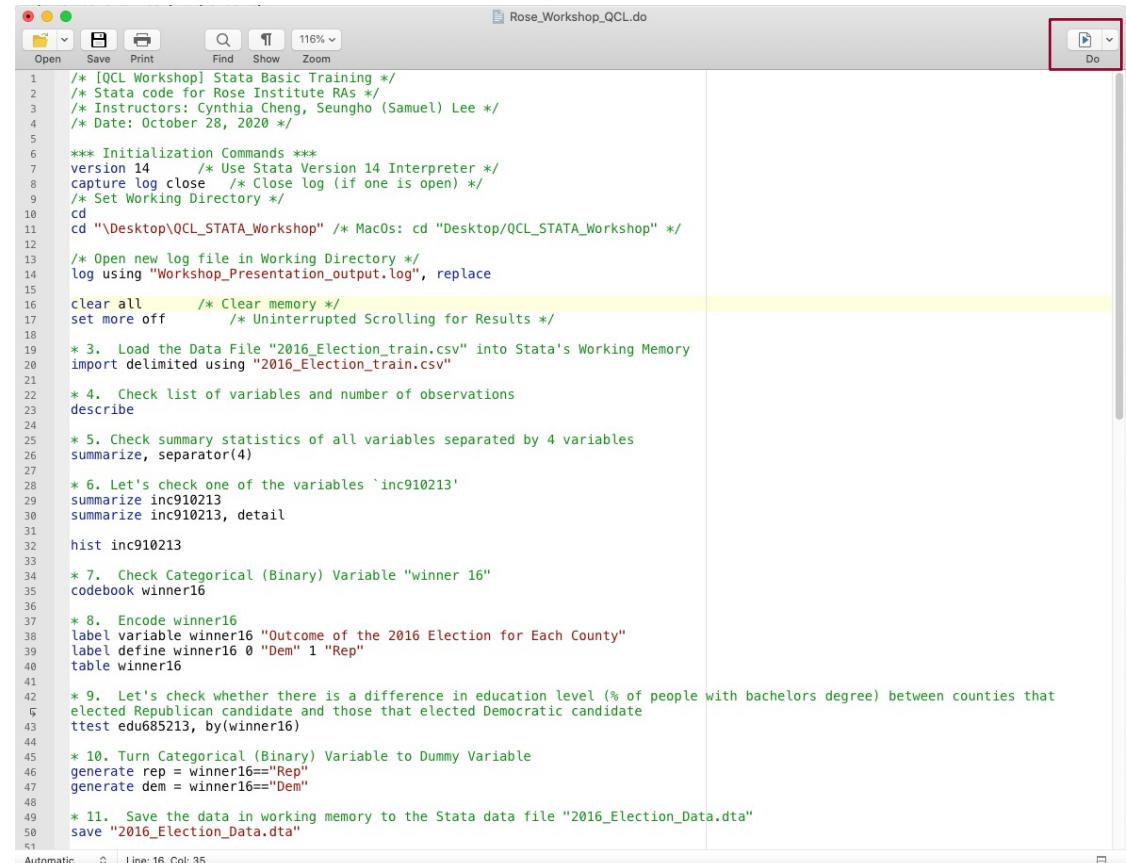
Data Import

► Setting Working Directory:

- “cd” command sets which folder you are going to be working on
- Make sure to include data files in the folder

► Importing Data

- Run import delimited using “filename.csv” command to import data files
- You can also import Excel files (and many others) as well as direct URL link!



```
/* [QCL Workshop] Stata Basic Training */
/* Stata code for Rose Institute RAs */
/* Instructors: Cynthia Cheng, Seungho (Samuel) Lee */
/* Date: October 28, 2020 */

*** Initialization Commands ***
version 14 /* Use Stata Version 14 Interpreter */
capture log close /* Close log (if one is open) */
/* Set Working Directory */
cd "\Desktop\QCL_STATA_Workshop" /* MacOs: cd "Desktop/QCL_STATA_Workshop" */

/* Open new log file in Working Directory */
log using "Workshop_Presentation_output.log", replace

clear all /* Clear memory */
set more off /* Uninterrupted Scrolling for Results */

* 3. Load the Data File "2016_Election_train.csv" into Stata's Working Memory
import delimited using "2016_Election_train.csv"

* 4. Check list of variables and number of observations
describe

* 5. Check summary statistics of all variables separated by 4 variables
summarize, separator(4)

* 6. Let's check one of the variables `inc910213'
summarize inc910213
summarize inc910213, detail

hist inc910213

* 7. Check Categorical (Binary) Variable "winner_16"
codebook winner16

* 8. Encode winner16
label variable winner16 "Outcome of the 2016 Election for Each County"
label define winner16 0 "Dem" 1 "Rep"
table winner16

* 9. Let's check whether there is a difference in education level (% of people with bachelors degree) between counties that elected Republican candidate and those that elected Democratic candidate
ttest edu685213, by(winner16)

* 10. Turn Categorical (Binary) Variable to Dummy Variable
generate rep = winner16=="Rep"
generate dem = winner16=="Dem"

* 11. Save the data in working memory to the Stata data file "2016_Election_Data.dta"
save "2016_Election_Data.dta"
```

Define data

Whenever you use *import* function, it outputs a message that indicates the numbers of variables and observations in the dataset. For more details, use *describe*

```
. * 3. Load the Data File "2016_Election_train.csv" into Stata's Working Memory
. import delimited using "2016_Election_train.csv"
(52 vars, 2,489 obs)

.
. * 4. Check list of variables and number of observations
. describe

Contains data
obs:           2,489
vars:          52

variable name  storage  display    value
          type      format   label     variable label
-----  
pst045214    long     %12.0g    PST045214
pst040210    long     %12.0g    PST040210
pst120214    float    %9.0g     PST120214
pop010210    long     %12.0g    POP010210
age135214    float    %9.0g     AGE135214
age295214    float    %9.0g     AGE295214
age775214    float    %9.0g     AGE775214
sex255214    float    %9.0g     SEX255214
rhi125214    float    %9.0g     RHI125214
rhi225214    float    %9.0g     RHI225214
```

- ▶ Data: Sampled 2016 Presidential Election Data by Counties (ECON122)
- ▶ ***describe* function can be used see a more detailed information of the imported data:**
 - ▶ Observations, Variables
 - ▶ Variable Name, Storage Type (e.g., long, float), Display format, value label, variable label
 - ▶ On Stata, you can label values and variables, which are helpful references
(we will look at these functions during the Hands-on Exercise)

Define data

Whenever you use *import* function, it outputs a message that indicates the numbers of variables and observations in the dataset. For more details, use *describe*

```
. * 3. Load the Data File "2016_Election_train.csv" into Stata's Working Memory
. import delimited using "2016_Election_train.csv"
(52 vars, 2,489 obs)

.

. * 4. Check list of variables and number of observations
. describe

Contains data
obs:           2,489
vars:          52

variable name  storage   display    value
           type      format   label     variable label
-----
```

variable name	storage type	display format	value label	variable label
pst045214	long	%12.0g	PST045214	PST045214
pst040210	long	%12.0g	PST040210	PST040210
pst120214	float	%9.0g	PST120214	PST120214
pop010210	long	%12.0g	POP010210	POP010210
age135214	float	%9.0g	AGE135214	AGE135214
age295214	float	%9.0g	AGE295214	AGE295214
age775214	float	%9.0g	AGE775214	AGE775214
sex255214	float	%9.0g	SEX255214	SEX255214
rhi125214	float	%9.0g	RHI125214	RHI125214
rhi225214	float	%9.0g	RHI225214	RHI225214

► Common Storage Types

- ▶ byte: integer values between -127 and 100
- ▶ int: integer values between -32,767 and 32,740
- ▶ long: integer values between -2,147,483,647 and 2,147,483,620
- ▶ float: real numbers (i.e., numbers with decimal points) with about 8 digits of accuracy
- ▶ double: real numbers (i.e., numbers with decimal points) with about 16 digits of accuracy
- ▶ str3: string values with a maximum length of 3
- ▶ What does having string values imply about the variable? (winner16 is a string variable!)

Summary Statistics

- ▶ **summarize** function can be used see a more detailed information of the imported data:
 - ▶ Observations: number of observations in the variable
 - ▶ Mean: Mean (Average) Value of the variable
 - ▶ Standard Deviation
 - ▶ Min
 - ▶ Max

```
. summarize inc910213
```

Variable	Obs	Mean	Std. Dev.	Min	Max
inc910213	2,489	23558.73	5382.698	11818	62498

Summary Statistics

- Including *detail* option in **summarize** allows users to check more specific statistics:
 - Percentiles:** a value of the variable at a given percentile (50th Percentile = Median)
 - Smallest/Largest:** 4 lowest/highest values
 - Skewness:** degree of distortion in our distribution (from normal) and direction
 - Positive:** skewed to the right
 - Negative:** skewed to the left
 - Zero:** Normal
 - Kurtosis:** how “fat” the tails are in the distribution, which shows whether there are *extreme outliers* in the data
 - High deviation from 3** indicates that there is *high kurtosis*

```
. summarize inc910213, detail
```

INC910213		
Percentiles		
1%	13954	
5%	16540	
10%	17842	
25%	19929	
50%	22888	
75%	26187	
90%	29905	
95%	33170	
99%	42210	
Smallest		
	11818	
	12042	
	12113	
	12177	
Largest		
	54608	
	56791	
	62018	
	62498	
Obs		2,489
Sum of Wgt.		2,489
Mean		23558.73
Std. Dev.		5382.698
Variance		2.90e+07
Skewness		1.437638
Kurtosis		7.982021

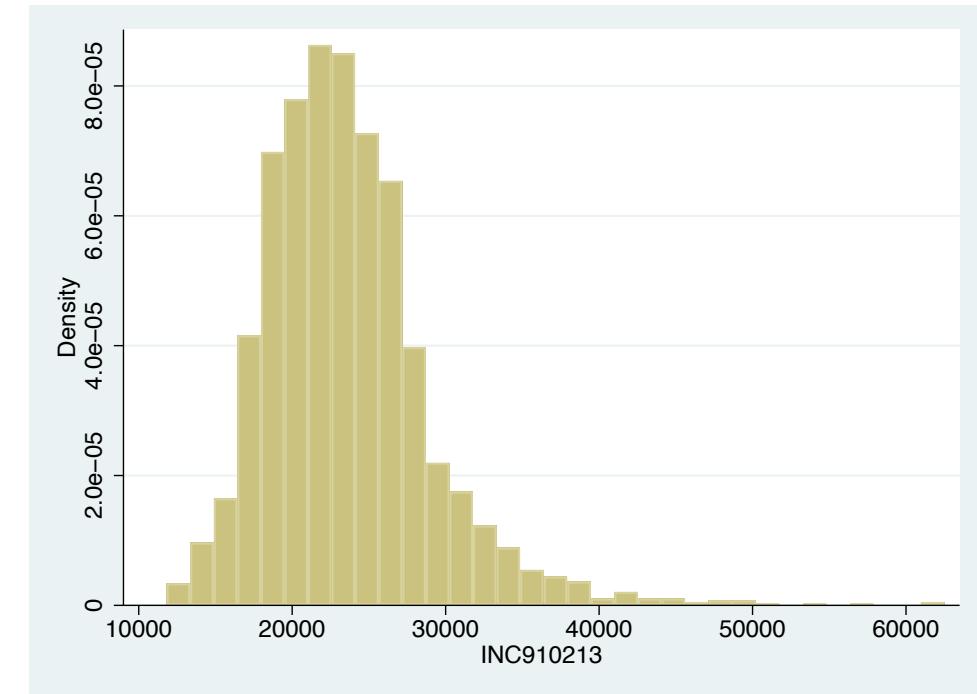
Mean cannot be captured correctly,
leading to wrong interpretation!

Summary Statistics

```
. summarize inc910213, detail
```

INC910213		
Percentiles		
1%	13954	
5%	16540	
10%	17842	
25%	19929	
50%	22888	
75%	26187	
90%	29905	
95%	33170	
99%	42210	
Smallest		
	11818	
	12042	
	12113	
	12177	
Largest		
	54608	
	56791	
	62018	
	62498	
Obs		2,489
Sum of Wgt.		2,489
Mean		23558.73
Std. Dev.		5382.698
Variance		2.90e+07
Skewness		1.437638
Kurtosis		7.982021

Mean cannot be captured correctly,
leading to wrong interpretation!



Summary Statistics

Summarize function can be used to see more detailed information about each variable. This can be done with all at once or on an individual basis

```
. summarize, separator(4)
```

Variable	Obs	Mean	Std. Dev.	Min	Max
pst045214	2,489	105939.7	351141.8	262	1.01e+07
pst040210	2,489	102615.3	337237.6	286	9818664
pst120214	2,489	.4666131	4.212294	-17	72.9
pop010210	2,489	102609.2	337229.8	286	9818605
age135214	2,489	5.88188	1.166596	1.5	12.2
age295214	2,489	22.52294	3.291917	7.4	37.5
age775214	2,489	17.66199	4.405271	4.1	52.9
sex255214	2,489	49.99574	2.167879	30.2	56.8
rhi125214	2,489	85.68112	15.331	12.8	99.3
rhi225214	2,489	9.123423	14.22124	0	84.1
rhi325214	2,489	1.916191	5.948727	0	82.2
rhi425214	2,489	1.322981	2.427037	0	42.4
rhi525214	2,489	.0992768	.3537572	0	12.7
rhi625214	2,489	1.849538	1.278885	0	29.4
rhi725214	2,489	9.007553	13.41946	.2	95.2
rhi825214	2,489	77.68726	19.34094	3.1	98.6
pop715213	2,489	86.40723	4.401603	50.8	99.8
pop645213	2,489	4.461511	5.482427	0	47.8
pop815213	2,489	9.077139	11.31973	0	94.2
edu635213	2,489	84.56774	6.806949	54	99

- ▶ For a summary statistics output for all variables, following values are produced for each variable:
 - ▶ Observations
 - ▶ Mean
 - ▶ Standard Deviation
 - ▶ Min
 - ▶ Max
- ▶ What can we know about *winner16* variable?

Summary Statistics

Summarize function can be used to see more detailed information about each variable. This can be done with all at once or on an individual basis

Variable	Obs	Mean	Std. Dev.	Min	Max
winner16	0				

- ▶ Why does *winner16* have 0 observation?
 - ▶ As we mentioned before, it is stored as a *string* type, which needs to be recoded
 - ▶ Let's try **codebook** function to check what string inputs are recorded in the variable

Summary Statistics

```
. codebook winner16
```

```
winner16
```

```
(unlabeled)
```

```
type: string (str3)

unique values: 2 missing "": 0/2,489

tabulation: Freq. Value
            378 "Dem"
            2,111 "Rep"
```

- ▶ We can see that there are 378 occurrences of “Dem” and 2,111 occurrences of “Rep”
 - ▶ As we can see from a boxed corner, our binary variable is not labeled
 - ▶ By labeling / encoding our data, we are able to assess statistical significance of differences between different groups (or *string* values), which is done with *t-test*
- ▶ Let’s label *winner16* variable and run *t-test*

Categorical Data – Processing

Labeling is useful in analyzing variables from different observations based on their *string values*

- ▶ Categorical Variables

- ▶ **Binary**, Nominal, Ordinal
- ▶ Can be used for classifying different categories, predicting categorical events, or explaining differences among categorical values

- ▶ Numerical Variables

- ▶ Continuous (infinite interval) or Discrete (finite)
- ▶ Take on any value within a finite or infinite interval
- ▶ Can be used for finding relationships and identifying characteristics



```
. * 8. Turn Categorical/Binary Variable "winner16"  
. generate rep = winner16=="Rep"  
  
. label variable winner16 "Outcome of the 2016 Elect  
  
. label define winner16 0 "Dem" 1 "Rep"  
. label value rep winner16
```

- ① Creates *rep* variable with 1 = "Rep" (all else are 0)
- ② Set *winner16*'s label with a description
- ③ Define numerical values to *winner16*'s string values
- ④ Assign *winner16* string values as *rep* labels

Summary Statistics (cont.)

- ▶ This is *t*-test of % of county residents with bachelor's degree (*edu685213*) on two *string* groups
- ▶ Shows *t*-test result on ***difference between the two groups*** with summary statistics of each group and a complete dataset
- ▶ Running *t*-test without specifying by condition produces the test on whether the variable is statistically significantly different from 0

▶ Results

- ▶ We find the difference ***statistically significant*** at almost ***0% confidence level***
- ▶ We also find that the counties that elected a democratic candidate have ***higher proportion of college educated residents***

```
. ttest edu685213, by(winner16)

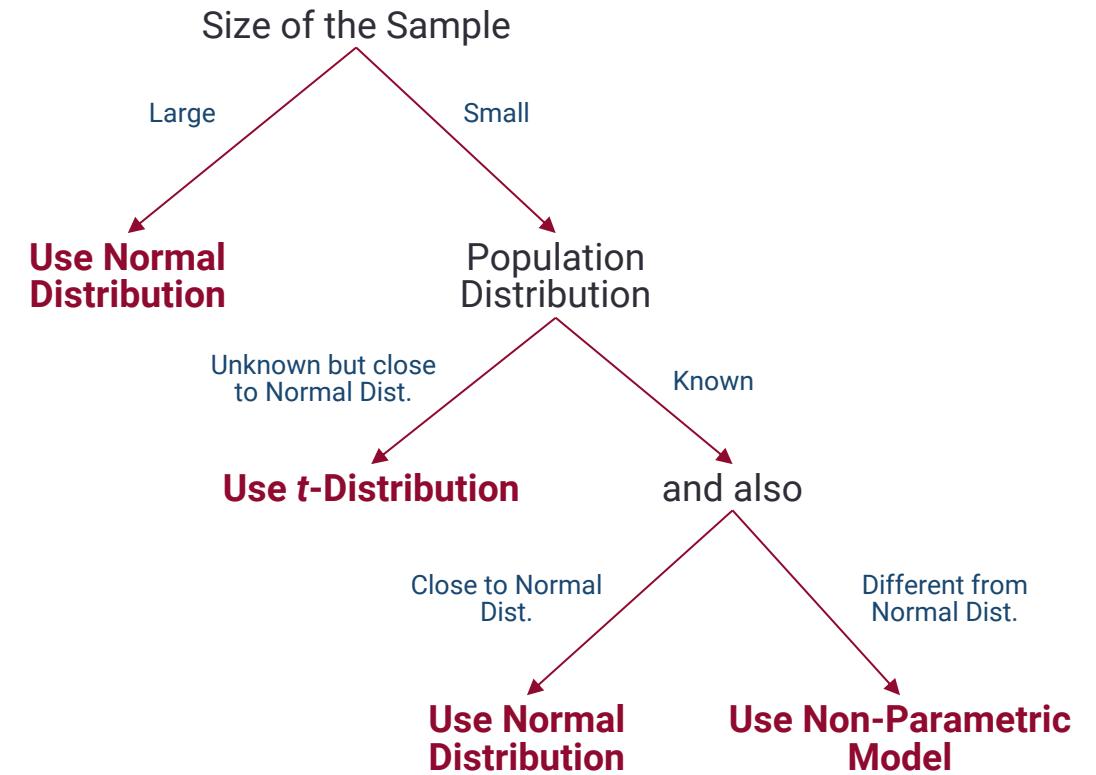
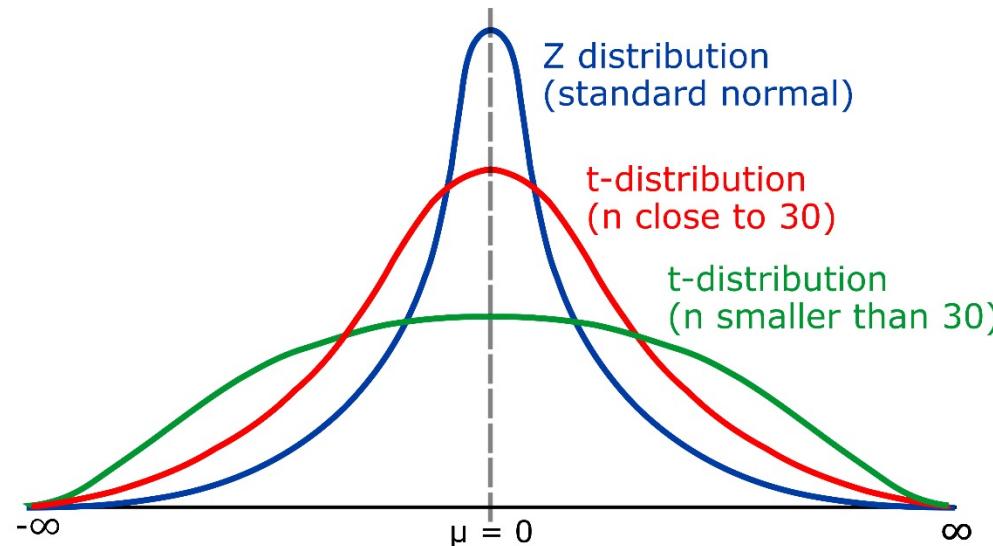
Two-sample t test with equal variances

      Group          Obs        Mean       Std. Err.      Std. Dev. [95% Conf. Interval]
      Dem           378     28.08862     .6283082    12.21571    26.8532    29.32405
      Rep          2,111    18.1973     .1462483    6.719467   17.91049   18.48411
combined          2,489    19.69948     .1718544    8.573794   19.36249   20.03647
diff            9.891324    .4359419                 9.036478   10.74617

      diff = mean(Dem) - mean(Rep)          t = 22.6895
      Ho: diff = 0                         degrees of freedom = 2487

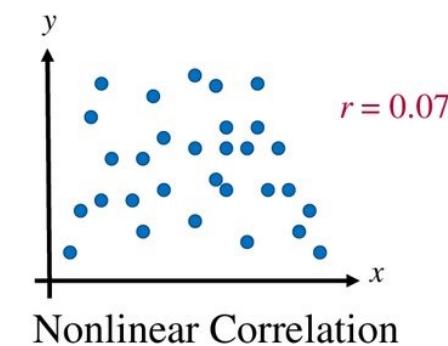
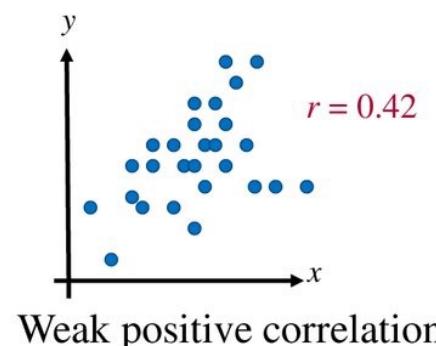
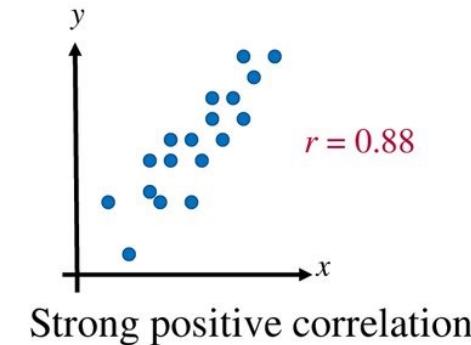
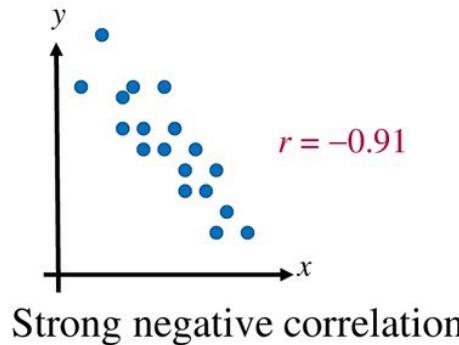
      Ha: diff < 0          Ha: diff != 0          Ha: diff > 0
      Pr(T < t) = 1.0000    Pr(|T| > |t|) = 0.0000    Pr(T > t) = 0.0000
```

Summary Statistics



Regression – Background

Before jumping into analyzing `winner16`, let's take a look at how population size impacts the number of firms in counties



- ▶ **Correlation Analysis:** evaluates the strength of relationship between two numerical variables
 - ▶ If the coefficient is **close to ± 1** , a relationship between the two are **strongly correlated**
 - ▶ **Strongly Positive:** two variables move along the same direction
 - ▶ **Strongly Negative:** two variables move along the opposite direction
- ▶ Note that **Correlation \neq Causation**

Regression – Background

Before jumping into analyzing *winner16*, let's take a look at how population size impacts the number of firms in counties

		s~001207 p~045214
sbo001207		1.0000
pst045214	0.9845	1.0000

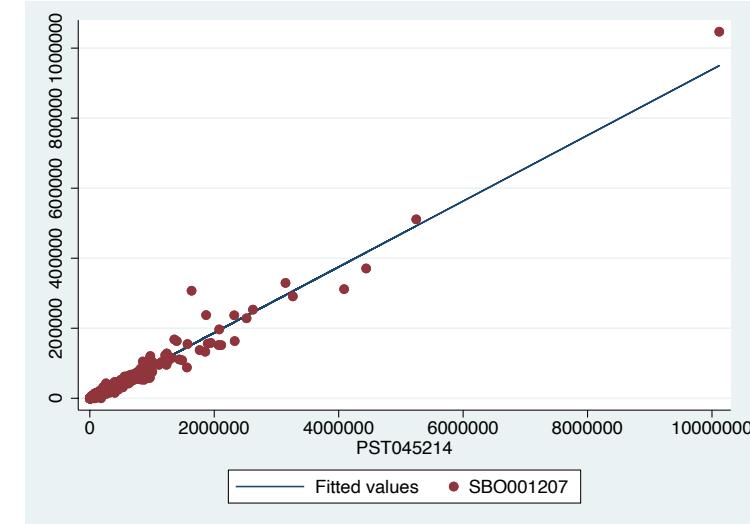
- ▶ **Correlation Analysis:** evaluates the strength of relationship between two numerical variables
- ▶ If the coefficient is **close to ± 1** , a relationship between the two are **strongly correlated**
- ▶ **Strongly Positive:** two variables move along the same direction
- ▶ **Strongly Negative:** two variables move along the opposite direction
- ▶ If the two variables have corr. coefficient of 0.9845, what would the regression look like?

Regression – Background

Regression Output

. reg sbo001207 pst045214						
Source	SS	df	MS	Number of obs	=	2,489
Model	2.7093e+12	1	2.7093e+12	F(1, 2487)	=	78154.33
Residual	8.6213e+10	2,487	34665466.3	Prob > F	=	0.0000
Total	2.7955e+12	2,488	1.1236e+09	R-squared	=	0.9692
				Adj R-squared	=	0.9691
				Root MSE	=	5887.7
<hr/>						
sbo001207	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pst045214	.0939761	.0003362	279.56	0.000	.0933169	.0946353
_cons	-849.4545	123.2708	-6.89	0.000	-1091.179	-607.7304

Regression Plot



- ▶ **R²** value is 0.9692, meaning that **96.92%** of variation is explained
- ▶ This is squared value of correlation coefficient that we saw earlier

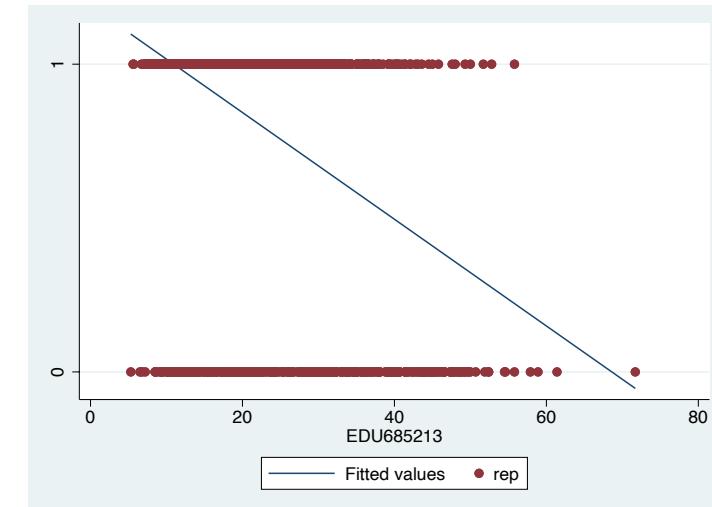
- ▶ Commands required to plot the regression:
 - ▶ **predict yhat** (right after running the regression)
 - ▶ **twoway scatter yhat y x, connect(l.) symbol(i 0)**
- ▶ We will now regress *Rep* binary variable on % of college educated to see the relationship

Regression – Simple Regression

```
. reg rep edu685213
```

Source	SS	df	MS	Number of obs	=	2,489
Model	54.9822781	1	54.9822781	F(1, 2487)	=	514.82
Residual	265.611535	2,487	.106799974	Prob > F	=	0.0000
Total	320.593813	2,488	.128856034	R-squared	=	0.1715
				Adj R-squared	=	0.1712
				Root MSE	=	.3268

rep	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
edu685213	-.0173386	.0007642	-22.69	0.000	-.018837 -.0158401
_cons	1.189692	.0164171	72.47	0.000	1.1575 1.221885



- ▶ This is a simple linear regression model, $Rep = \beta_0 + \beta_1 \cdot edu685213 + \epsilon$.
 - ▶ We can see from R^2 that there is a lot of room to improve (82.85% of variation is still not explained)
 - ▶ Let's try to add more explanatory (independent) variables
- ▶ We can also see that coefficient of the % of college graduates is statistically significant (large t -statistics / low p -value)

Regression – Multiple Regression

- ▶ Take a look at a correlation between % of residents with bachelor's degree and those with high school diploma (**edu635213**)

- ▶ **Rules of Thumb:**

No linear relationship = 0

Perfect linear relationship = ± 1

Weak linear relationship = $|0 - 0.3|$

Moderate linear relationship = $|0.3 - 0.7|$

Strong linear relationship = $|0.7 - 1.0|$

- ▶ Generally, adding two or more variables with $R < 0.7$ does not increase a presence of *Multicollinearity*

```
. corr edu685213 edu635213  
(obs=2,489)
```

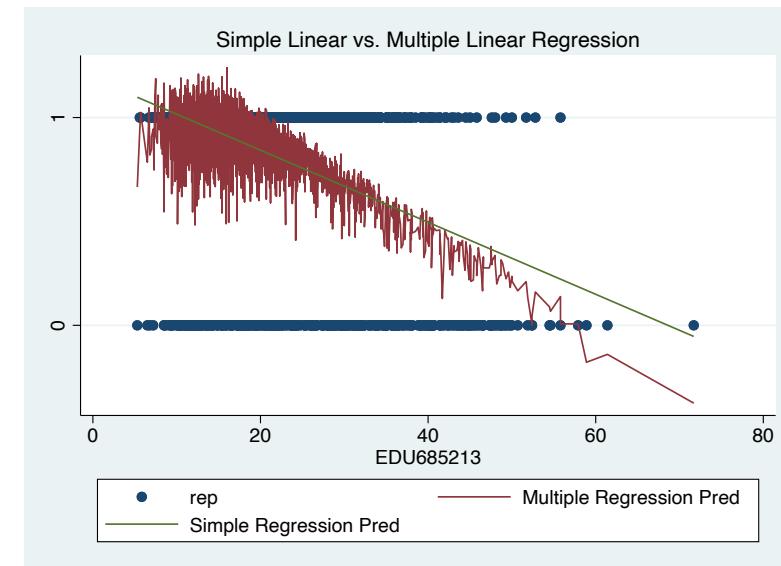
	e~685213	e~635213
edu685213	1.0000	
edu635213	0.5958	1.0000

Regression – Multiple Regression

```
. reg rep edu685213 edu635213
```

Source	SS	df	MS	Number of obs	=	2,489
Model	85.4182128	2	42.7091064	F(2, 2486)	=	451.47
Residual	235.1756	2,486	.0946	Prob > F	=	0.0000
Total	320.593813	2,488	.128856034	R-squared	=	0.2664
				Adj R-squared	=	0.2658
				Root MSE	=	.30757

rep	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
edu685213	-.0269097	.0008955	-30.05	0.000	-.0286657 -.0251536
edu635213	.0202324	.001128	17.94	0.000	.0180205 .0224443
_cons	-.3327694	.0862735	-3.86	0.000	-.5019448 -.163594



- ▶ This is a multiple linear regression model, $y = \beta_0 + \beta_1 \cdot \text{edu685213} + \beta_2 \cdot \text{edu635213} + \epsilon$.
- ▶ There is a recognizable improvement from the model
- ▶ Also, notice that the coefficients have low p -values

Regression – Multiple Regression

- Our final model exhibits relatively high performance with coefficients statistically significant
 - Adjusted R-Squared: 0.4324
 - Each coefficient represents a percentage point difference on selecting Trump by a unit change in the variable
(e.g., having 1% higher proportion of Persons under 18 years lead to 1.66 p.p increase in chances of selecting Trump)
- BONUS: Instead of Linear Regression model, what can we use to address this *classification problem*?

```
. reg rep edu685213 age295214 rhi825214 inc910213 edu685_inc910
```

Source	SS	df	MS	Number of obs	=	2,489
Model	139.003014	5	27.8006027	F(5, 2483)	=	380.13
Residual	181.590799	2,483	.073133628	Prob > F	=	0.0000
Total	320.593813	2,488	.128856034	R-squared	=	0.4336
				Adj R-squared	=	0.4324
				Root MSE	=	.27043

rep	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
edu685213	-.0124817	.0020331	-6.14	0.000	-.0164685 -.0084949
age295214	.0166106	.0017648	9.41	0.000	.0131499 .0200713
rhi825214	.0092258	.000322	28.65	0.000	.0085945 .0098572
inc910213	.0000142	2.61e-06	5.46	0.000	9.12e-06 .0000193
edu685_inc910	-2.72e-07	6.99e-08	-3.90	0.000	-4.09e-07 -1.35e-07
_cons	-.1959171	.0678765	-2.89	0.004	-.3290174 -.0628168

Regression – Logistic Regression

- **Logistic Regression:** a binary classification model, with a $0 \leq h_{\theta}(x) \leq 1$ range, that outputs probability of an observation to be either of the binary values, using:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 x_1 + \dots + \beta_x x_x$$

- Utilizes Maximum Likelihood Estimation (MLE)
- Good for analyzing binary variables since it is bound between 0 and 1
- Coefficients: the expected change in log odds for one-unit increase in one of the independent variables (all held constant)
- Can be used as a prediction model or as a model for Propensity Score Matching in experimental designs

```
. logit rep edu635213 edu685213 age295214 rhi825214
```

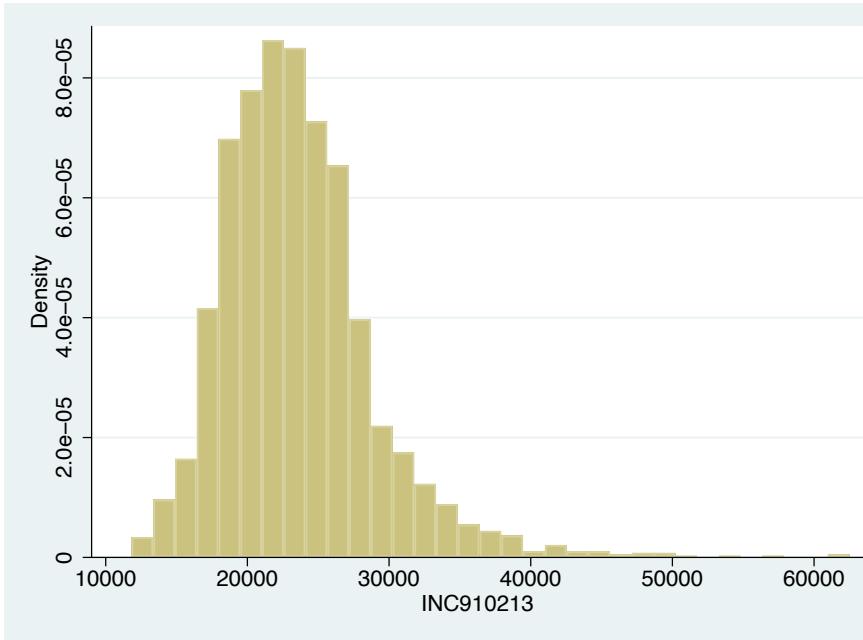
```
Iteration 0:  log likelihood = -1060.1549
Iteration 1:  log likelihood = -613.33484
Iteration 2:  log likelihood = -497.25696
Iteration 3:  log likelihood = -488.21436
Iteration 4:  log likelihood = -488.11949
Iteration 5:  log likelihood = -488.11942
Iteration 6:  log likelihood = -488.11942
```

```
Logistic regression                                         Number of obs      =     2,489
                                                               LR chi2(4)       =    1144.07
                                                               Prob > chi2      =     0.0000
                                                               Pseudo R2        =     0.5396
Log likelihood = -488.11942
```

rep	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
edu635213	-.0726455	.0223479	-3.25	0.001	-.1164466 -.0288445
edu685213	-.1565054	.0132512	-11.81	0.000	-.1824772 -.1305336
age295214	.2281572	.0251916	9.06	0.000	.1787825 .2775319
rhi825214	.1311172	.0077747	16.86	0.000	.1158791 .1463552
_cons	-2.670578	1.508954	-1.77	0.077	-5.628074 .2869181

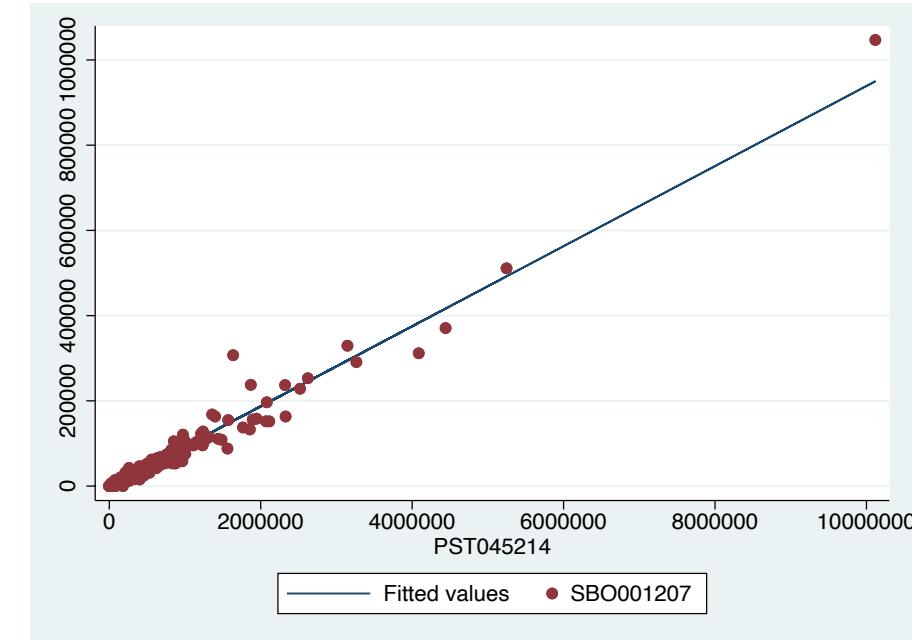
Charts – Syntax and Examples

Histogram of INC910213



Command: `hist inc910213`

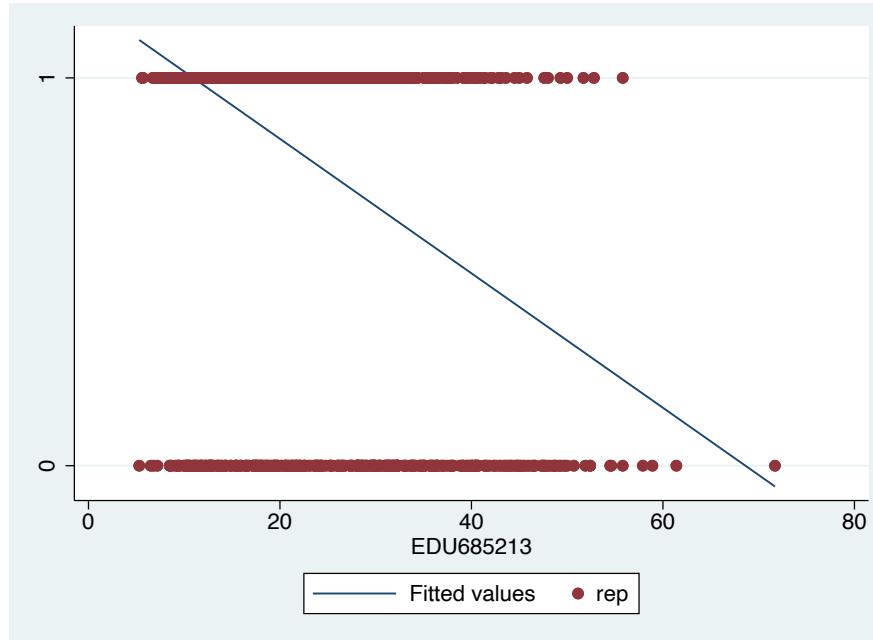
Regression Plot of Simple Linear Regression



`twoway scatter yhat_ex sb001207
pst045214, connect(l .) symbol(i 0)`

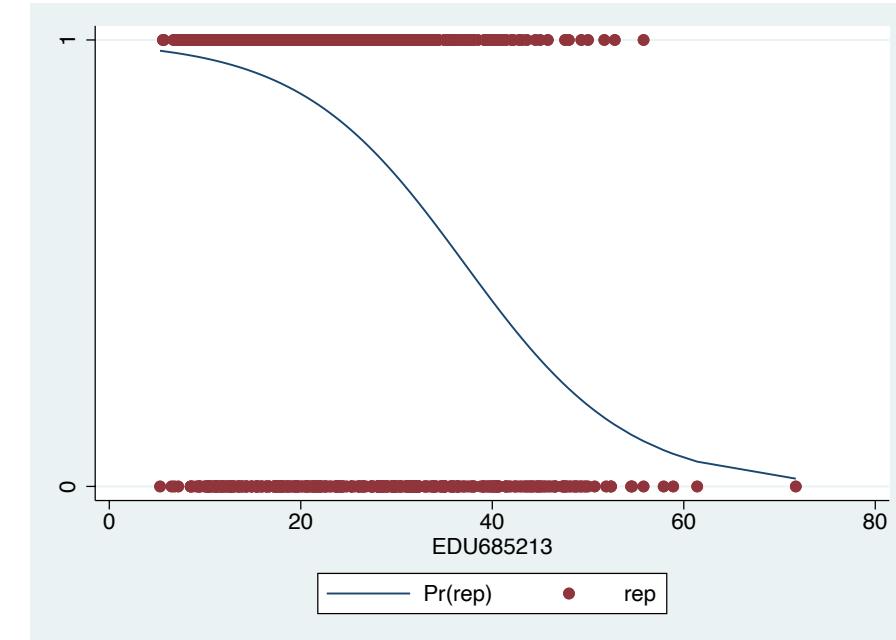
Charts – Syntax and Examples

Simple Linear Regression Plot



```
twoway scatter yhat_linear rep edu685213,  
connect(l .) symbol(i 0) sort ylabel(0 1)
```

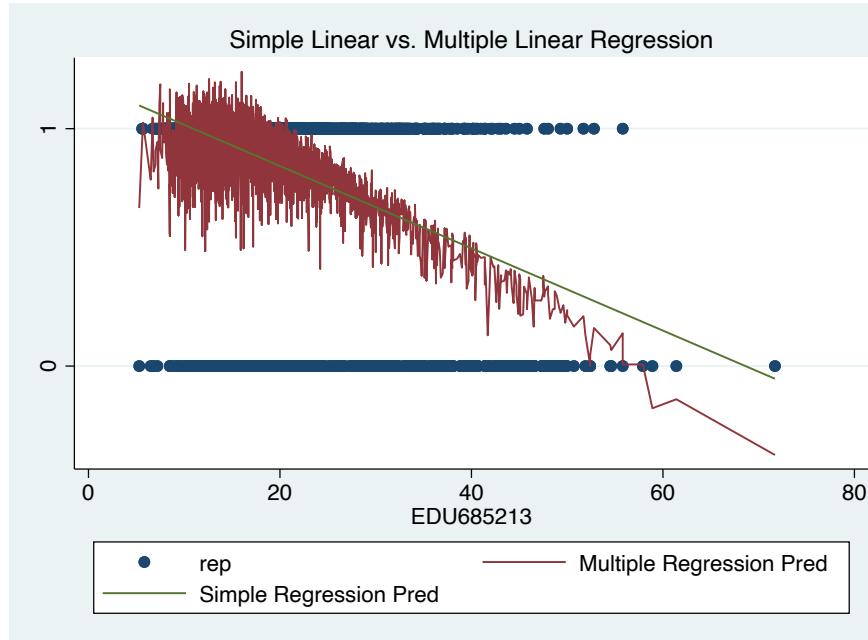
Regression Plot of Simple Logistic Regression (Logit)



```
twoway scatter yhat_logit rep edu685213,  
connect(l i) msymbol(i 0) sort ylabel(0 1)
```

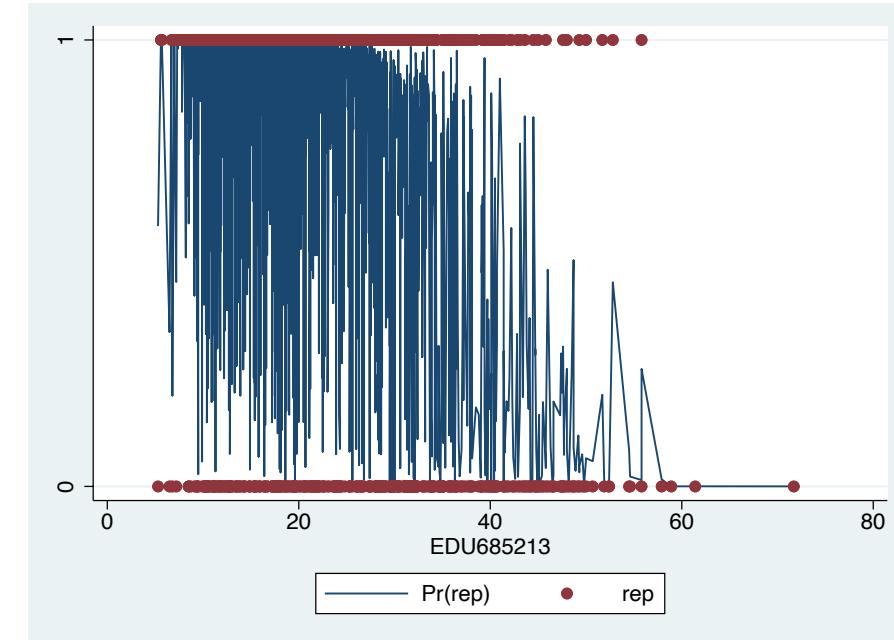
Charts – Syntax and Examples

Simple Linear vs. Multiple Linear Regression Plot



```
twoway scatter rep MLR yhat_linear edu685213,  
connect(. l -) msymbol(. i i) sort ylabel(0 1)
```

Regression Plot of Multiple Logistic Regression (Logit)



```
twoway scatter yhat_all rep edu685213,  
connect(l i) msymbol(i 0) sort ylabel(0 1)
```

Summary / Q&A

- ▶ Key Takeaways:
 - ▶ User Interface (Do-File)
 - ▶ Data Import and Exploration
 - ▶ Summary Statistics
 - ▶ Regression Analysis
- ▶ Tips:
 - ▶ Stata/R Useful Packages: https://geocenter.github.io/StataTraining/portfolio/06_resource/
 - ▶ Internet Guide to Stata: <http://wlm.userweb.mwn.de/Stata/>
 - ▶ UCLA IDRE Guide: <https://stats.idre.ucla.edu/stata/>
 - ▶ Princeton DSS: <https://www.princeton.edu/~otorres/Stata/StataTutorial.pdf>
- ▶ Contact: qcl@cmc.edu (with a title: “Re: QCL Stata Workshop”)
- ▶ Feel free to contact me (slee19@students.cmc.edu) if you have any questions on PowerPoint and Stata materials

Appendix

Regression – Multiple Regression

```
. reg rep edu685213 edu635213 age295214
```

Source	SS	df	MS	Number of obs	=	2,489
Model	85.9024426	3	28.6341475	F(3, 2485)	=	303.19
Residual	234.69137	2,485	.094443207	Prob > F	=	0.0000
Total	320.593813	2,488	.128856034	R-squared	=	0.2679
				Adj R-squared	=	0.2671
				Root MSE	=	.30732

rep	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
edu685213	-.0269092	.0008948	-30.07	0.000	-.0286638 -.0251546
edu635213	.0206581	.0011426	18.08	0.000	.0184175 .0228986
age295214	.0043285	.0019116	2.26	0.024	.00058 .008077
_cons	-.4662677	.1044352	-4.46	0.000	-.6710568 -.2614787

- ▶ Let's try to add more explanatory variable: *age295214* (persons under 18 years)
 - ▶ We can see from Adjusted R^2 that there is a lot of room to improve (73.29% of variation is still not explained)
 - ▶ Also, note that *age295214* has a relatively high *p*-value but within 95% Confidence Level
 - ▶ Can adding *rhi825214* (White alone, Not Hispanic) help to increase our model's performance?

Regression – Multiple Regression

- ▶ As expected, including race variable leads to better performance (42.69% of variation explained)
- ▶ We also see that a coefficient of the high school graduates is not statistically significant. Why?
- ▶ How should we proceed?

```
. reg rep edu635213 edu685213 age295214 rhi825214
```

Source	SS	df	MS	Number of obs	=	2,489
Model	136.867873	4	34.2169683	F(4, 2484)	=	462.62
Residual	183.72594	2,484	.073963744	Prob > F	=	0.0000
Total	320.593813	2,488	.128856034	R-squared	=	0.4269
				Adj R-squared	=	0.4260
				Root MSE	=	.27196

	rep	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
edu635213	.001145	.001255	0.91	0.362	-.001316	.0036059
edu685213	-.0166986	.0008822	-18.93	0.000	-.0184286	-.0149686
age295214	.0176026	.0017657	9.97	0.000	.0141403	.0210649
rhi825214	.009711	.0003699	26.25	0.000	.0089856	.0104365
_cons	-.0706276	.093642	-0.75	0.451	-.254252	.1129969

Regression – Multiple Regression

- ▶ Is *edu635213* variable duplicitous in our model? Should we exclude them?
- ▶ What other variables do you think we can include in our model?
(Hint: Think about how policies impact people based on their background)

```
. corr edu635213 edu685213 age295214 rhi825214  
(obs=2,489)
```

	e~635213	e~685213	a~295214	r~825214
edu635213	1.0000			
edu685213	0.5958	1.0000		
age295214	-0.2035	-0.1214	1.0000	
rhi825214	0.4768	-0.0165	-0.3200	1.0000

Regression – Multiple Regression

- ▶ How about Income variable?
 - ▶ *INC910213*: Per Capita Income
- ▶ From checking the correlation table, what can we infer about the relationship between the level of education and per capital income level?
- ▶ If we assume that the impact of income on political leanings differ by the education level, we can use two methods:
 - 1) **Interaction Terms**
 - 2) Instrumental Variable (IV)

```
. corr edu685213 inc910213  
(obs=2,489)
```

		e~685213 i~910213
edu685213	1.0000	1.0000
	0.7714	

Regression – Multiple Regression

- ▶ Let's try to make *interaction term from the two variables*:

$\text{EDU685_INC910} = \text{EDU685213} \cdot \text{INC910213}$

- ▶ Use **gen edu685_inc910 = edu685213 * inc910213** command to generate the interaction variable
- ▶ Regress the following:
reg rep edu685213 age295214 rhi825214 inc910213 edu685_inc910

```
. corr edu685213 inc910213  
(obs=2,489)
```

		e~685213 i~910213
edu685213	1.0000	1.0000
	0.7714	