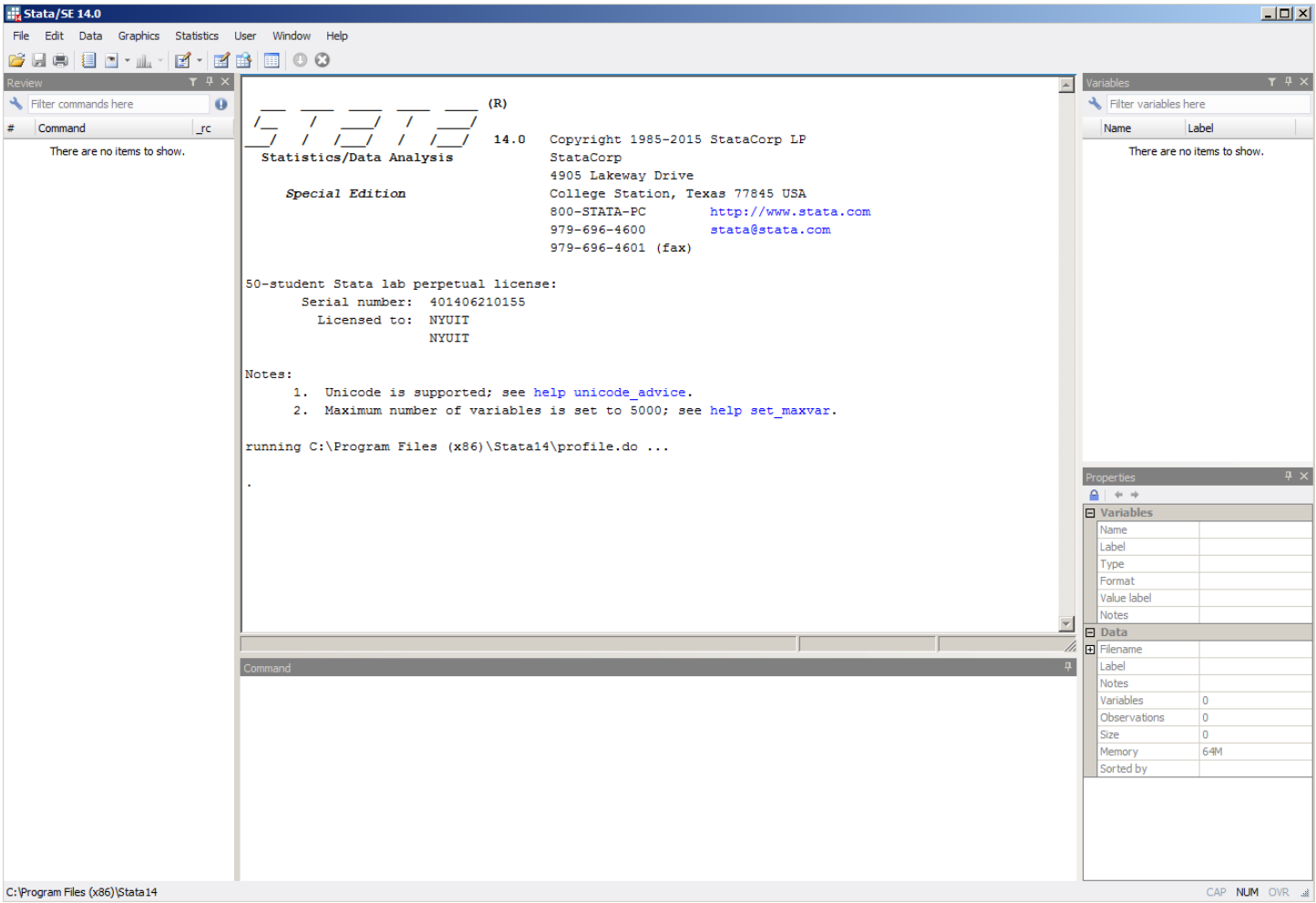
# Introduction to Stata

Table of Contents

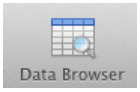
## Stata Environment

### *Stata Interface*

- When you open up Stata, the main window that opens up contains five different boxes
- The Results/Console will display your results and any errors that occur when you run your commands
- The Command box allows you to interactively run you Stata commands
- The Review box will display a list of previous commands you have run
- The Variables box will display all the variables contained in your dataset
- The Properties box will provide you with more information about your dataset, including number of observations, variable types, labels that are applied to your values etc...



*Browser*

- Use the `browse` command to open up the browser
- Alternatively, you can press the Data Browser Button at the top of the console 


Data Editor (Browse) — PizzaData.dta

Edit | Browse | Filter | Variables | Properties | Snapshots

ID[1] Jane

| | ID | pizza | female | hs | college | grad | income | age |
|---|---|---|---|---|---|---|---|---|
| 1 | Jane | 109 | 1 | 0 | 0 | 0 | 15000 | 25 |
| 2 | Cate | 0 | 1 | 0 | 0 | 0 | 30000 | 45 |
| 3 | Sue | 0 | 1 | 0 | 0 | 0 | 12000 | 20 |
| 4 | Bev | 108 | 1 | 0 | 0 | 0 | 20000 | 28 |
| 5 | Mich | 220 | 1 | 1 | 0 | 0 | 15000 | 25 |
| 6 | Ella | 189 | 1 | 1 | 0 | 0 | 30000 | 35 |
| 7 | Joan | 64 | 1 | 1 | 0 | 0 | 12000 | 40 |
| 8 | Suki | 262 | 1 | 1 | 0 | 0 | 12000 | 22 |
| 9 | Jen | 64 | 1 | 1 | 0 | 0 | 28000 | 30 |
| 10 | Lila | 35 | 1 | 1 | 0 | 0 | 22000 | 21 |
| 11 | Kit | 94 | 1 | 1 | 0 | 0 | 44000 | -1 |
| 12 | Lara | 71 | 1 | 0 | 1 | 0 | 10000 | 21 |
| 13 | Beth | 403 | 1 | 0 | 1 | 0 | 222000 | 45 |
| 14 | Judy | 41 | 1 | 0 | 1 | 0 | 32000 | 36 |
| 15 | Sue1 | 10 | 1 | 0 | 1 | 0 | 45000 | 36 |
| 16 | Sue2 | 110 | 1 | 0 | 1 | 0 | 55000 | 40 |
| 17 | Teri | 239 | 1 | 0 | 1 | 0 | 29000 | 23 |
| 18 | June | 63 | 1 | 0 | 1 | 0 | 39000 | 32 |
| 19 | Barb | 0 | 1 | 0 | 1 | 0 | 70000 | 52 |
| 20 | Bula | 106 | 1 | 0 | 0 | 1 | 55000 | 30 |
| 21 | Lily | 0 | 1 | 0 | 0 | 1 | 90000 | 45 |
| 22 | Joe | 141 | 0 | 0 | 0 | 0 | 6000 | 32 |
| 23 | Jon | 299 | 0 | 0 | 0 | 0 | 18000 | 20 |
| 24 | Joe2 | 148 | 0 | 0 | 0 | 0 | 55000 | -1 |
| 25 | John | 424 | 0 | 1 | 0 | 0 | 10000 | 18 |
| 26 | John2 | 242 | 0 | 1 | 0 | 0 | 23000 | 30 |
| 27 | Dave | 119 | 0 | 1 | 0 | 0 | 35000 | 45 |

Variables

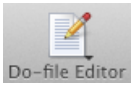
| Name | Label |
|---|---|
| ID | |
| pizza | annual pizza expen... |
| female | =1 if female |
| hs | =1 if highest degre... |
| college | =1 if highest degre... |
| grad | =1 if highest degre... |
| income | annual income, $ |
| age | age in years |

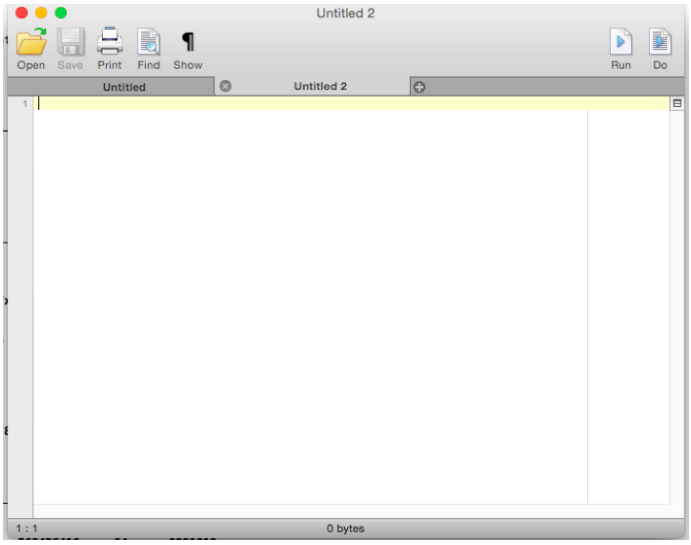
Properties

| | |
|---|---|
| **Variables** | |
| Name | ID |
| Label | |
| Type | str12 |
| Format | %12s |
| Value label | |
| Notes | |
| **Data** | |
| Filename | PizzaData.dta |
| Label | |
| Notes | |
| Variables | 8 |
| Observations | 40 |
| Size | 1.56K |
| Memory | 64M |
| Sorted by | |

Vars: 8 Order: Dataset | Obs: 40 | Length: 12 | Filter: Off

*Do File Editor*

- A .do file is a text file which contains all of the commands you use
- **File** → **New** → **Do-file** or you can choose the Do-file Editor button 

*Stata Syntax*

- Stata commands are generally one or two words
- In the help file, you will see the beginning of the word underlined, this means you can use the shorter syntax as a shortcut for the command
- For example, the `histogram` command has the first four letters underlined, so you can use the shortcut `hist`

## Title

**stata.com**

**histogram** — Histograms for continuous and categorical variables

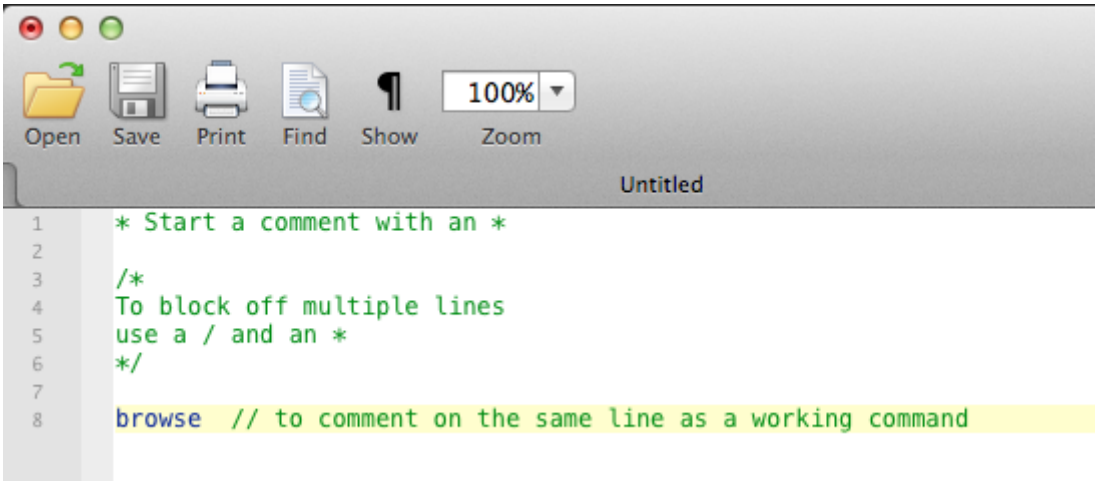


## Syntax

**histogram** *varname* [*if*] [*in*] [*weight*] [, [*continuous_opts* | *discrete_opts*] *options*]

*Comments*

- A comment can be a note to yourself or other readers
- Comments will appear green in your .do file and will not execute in the Console
- Three ways to write comments
    - Start a line with an `*`
    - Start a line/block with `/*` and end line/block with `*/`
    - After a line of code use `//` to create a comment

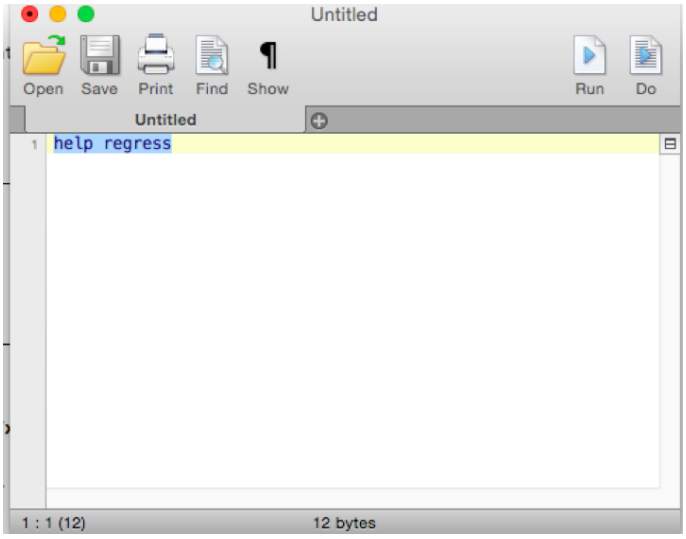


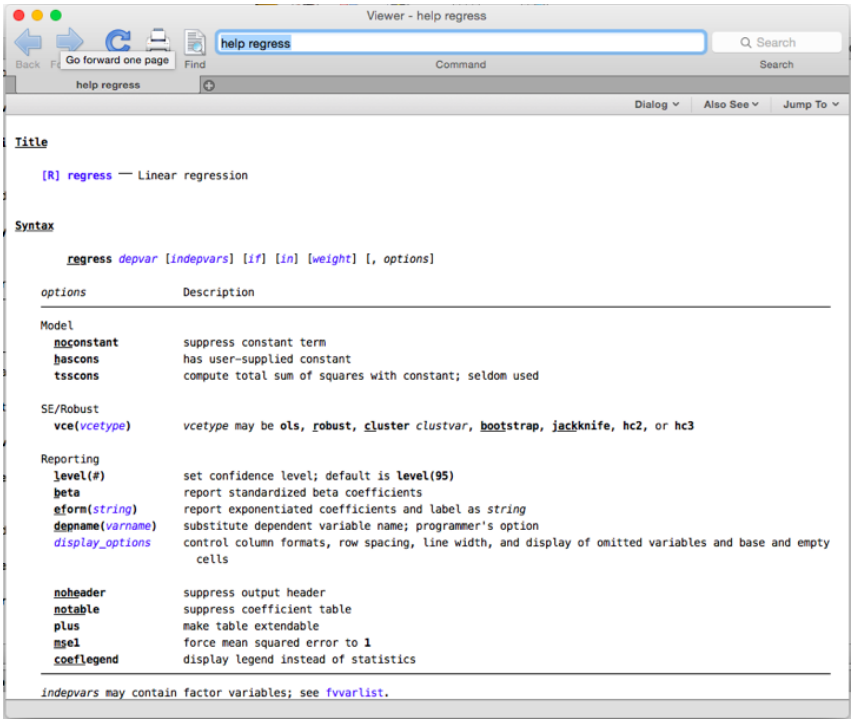
*How to Get Help inside Stata*

- Use the `help` command followed by the command which you seek assistance about

- You can also access help by choosing the Viewer button 

and typing your command in the Command bar at the top of the Viewer



**Working Directory**

- A working directory is the path to the folder in which you are importing data from and saving data to
- Use the `pwd` command to print out where your working directory is currently set to
- Use the `cd` command to change your directory
- Use the `dir` command to list the files in your working directory

Example:

a. Mac OS:

```
cd "/Users/NYU User/Desktop"
```

b. Windows:

```
cd "C:\Users\NYU User\Desktop"
```

c. NYU Virtual Computing Lab (VCL):
Start the path with `"\\client\c$` followed by your computer path

```
cd "\\client\c$\C:\Users\NYU User\Desktop"
```

## Importing Data

*Import Stata (.dta) File*

- A Stata dataset has a .dta extension
- If the dataset is in your working directory, use the `use` command to import a Stata dataset
- `use "Dataset"`

*Import Excel (.xls) File*

- Excel data comes with either an .xls or .xlsx extension
- If the dataset is in your working directory, use the `import excel` command to import an Excel file
- `import excel "Dataset.xls"`

*Import Text (.csv) File*

- A .csv is a comma separated value file
- If the dataset is in your working directory, use the `import delimited` command to import a .csv file
- `import delimited "Dataset.csv"`

## Creating Variables

- To create a new variable (column) in your dataset, use the `generate` command
- `generate MonthlyPizza = (pizza / 12)`
- The `egen` command can also be used to create new variables
- `egen avgSalary = mean(income), by(Education)`

*Managing Variables*

- The `keep` and `drop` commands can be used to keep and drop certain variables from your dataset command can also be used to create new variables

## Labels

*Variable Labels*

- Stata has specific rules when you name a variable
  - Variable names are case-sensitive
  - The name cannot contain more than 32 characters
  - Names can contain only letters, numbers and underscores
  - Spaces and other special characters are not allowed
  - The first character must be a character or an underscore and not a number
- Variable labels have more flexibility, they can be longer and can contain spaces and special characters
- To attach a label to the variable use the `label variable` command
- `label variable MonthlyPizza "Monthly Expenditure on Pizza"`

*Value Labels*

- To attach a label to the values of a variable use the `label define` command to create the variable
- Then use the `label values` command to apply the labels to the variable
- Use the `codebook` command to see the mappings of values and their labels
- `label define eduLabel 0 "Less than High School" 1 "High School" 2 "Undergraduate Degree" 3 "Graduate Degree"`
- `label values Education edulabel`

```
-------------------------------------------------------------------------------
Education                                       Highest Education Achieved
-------------------------------------------------------------------------------

                 type:  numeric (float)
                label:  eduLabel

                range:  [0,3]                        units:  1
        unique values:  4                    missing .:  0/40

           tabulation:  Freq.   Numeric  Label
                           7         0  Less than High School
                          15         1  High School
                          15         2  Undergraduate Degree
                           3         3  Graduate Degree
```

## Descriptive Statistics

- Use the `summarize` command to see the mappings of values and their labels
- `summarize income`

```
. summarize income
```

| Variable | Obs | Mean | Std. Dev. | Min | Max |
|---|---|---|---|---|---|
| income | 40 | 42925 | 39358.57 | 6000 | 222000 |

- Use the `detail` option to expand the summary statistics

```
. summarize income, d
```

annual income, $

| | Percentiles | Smallest | | |
|---|---|---|---|---|
| 1% | 6000 | 6000 | | |
| 5% | 10000 | 10000 | | |
| 10% | 12000 | 10000 | Obs | 40 |
| 25% | 21000 | 12000 | Sum of Wgt. | 40 |
| 50% | 30000 | | Mean | 42925 |
| | | Largest | Std. Dev. | 39358.57 |
| 75% | 55000 | 85000 | | |
| 90% | 82500 | 90000 | Variance | 1.55e+09 |
| 95% | 111000 | 132000 | Skewness | 2.735044 |
| 99% | 222000 | 222000 | Kurtosis | 12.20313 |

- Use the `mean` command to obtain more information about the mean

```
. mean income

Mean estimation                 Number of obs    =         40
```

| | Mean | Std. Err. | [95% Conf. | Interval] |
|---|---|---|---|---|
| income | 42925 | 6223.136 | 30337.52 | 55512.48 |

- Use the `tabstat` command to build custom tables

```
. tabstat income, stats(mean sd) by(Education)

Summary for variables: income
     by categories of: Education (Highest Education Achieved)
```

| Education | mean | sd |
|---|---|---|
| Less than High S | 22285.71 | 16214.34 |
| High School | 29733.33 | 18922.65 |
| Undergraduate De | 59333.33 | 53756.95 |
| Graduate Degree | 75000 | 18027.76 |
| Total | 42925 | 39358.57 |

## Frequency Tables

- `tabulate variable_name`
- `tab Education`

```
. tab Education
```

| Highest Education Achieved | Freq. | Percent | Cum. |
|---|---|---|---|
| Less than High School | 7 | 17.50 | 17.50 |
| High School | 15 | 37.50 | 55.00 |
| Undergraduate Degree | 15 | 37.50 | 92.50 |
| Graduate Degree | 3 | 7.50 | 100.00 |
| Total | 40 | 100.00 | |

- To create a cross tabulation, you can report two variable names
- `tab Education female`

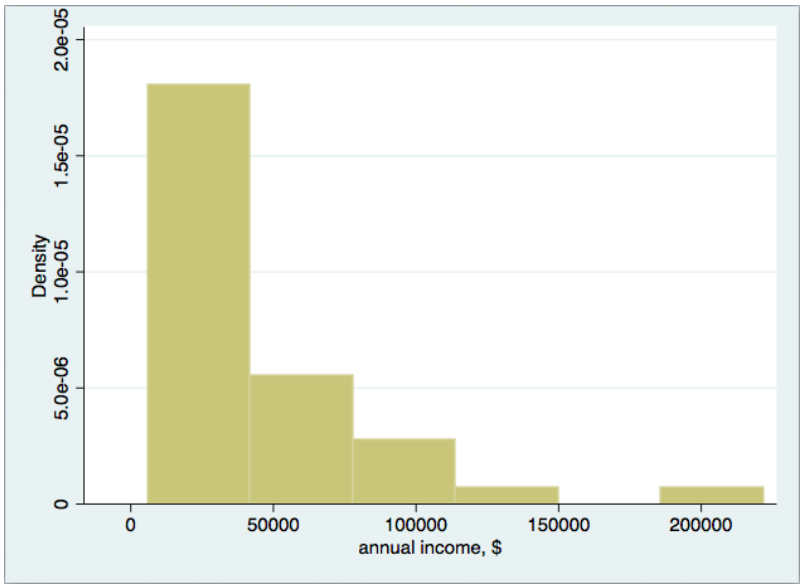
```
. tab Education female
```

| Highest Education Achieved | =1 if female | | Total |
|---|---|---|---|
| | 0 | 1 | |
| Less than High School | 3 | 4 | 7 |
| High School | 8 | 7 | 15 |
| Undergraduate Degree | 7 | 8 | 15 |
| Graduate Degree | 1 | 2 | 3 |
| Total | 19 | 21 | 40 |

## Graphs

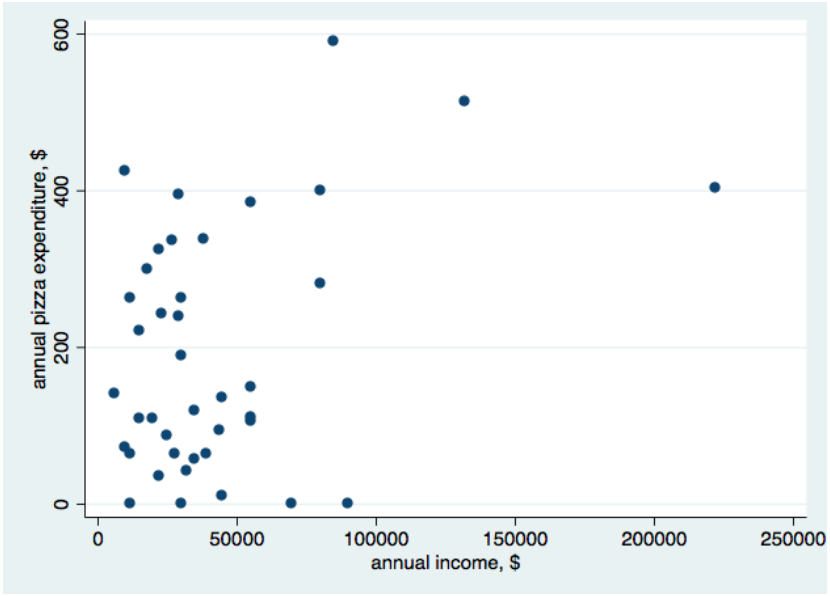
*Histogram*

- Create a histogram to see the distribution of a continuous variable
- `histogram income`




*Scatterplot*

- To examine the relationship between two continuous variables using a scatterplot the command `twoway scatter` can be used, where the dependent variable is listed first and the independent variable is listed second
- `twoway scatter pizza income`



## Basic Analysis

*Correlation*

- Use the `correlate` command to create a correlation matrix
- `correlate income pizza`

```
. correlate income pizza
(obs=40)

             |   income    pizza
-------------+------------------
      income |   1.0000
       pizza |   0.3680   1.0000
```

*T-Test*

- Using “pizza” as the continuous variable and “female” as the categorical variable (with 2 categories), run an independent samples ttest with the `ttest` command
- `ttest pizza, by(female)`

```
. ttest pizza, by(female)

Two-sample t test with equal variances

   Group |     Obs        Mean    Std. Err.   Std. Dev.   [95% Conf. Interval]
---------+--------------------------------------------------------------------
       0 |      19    288.1053    33.76907    147.196    217.1591    359.0515
       1 |      21    104.1905    22.89995   104.9408    56.42202    151.9589
---------+--------------------------------------------------------------------
combined |      40      191.55    24.64689   155.8806      141.697     241.403
---------+--------------------------------------------------------------------
    diff |            183.9148    40.12417               102.6877    265.1419
------------------------------------------------------------------------------
    diff = mean(0) - mean(1)                                      t =   4.5836
Ho: diff = 0                     degrees of freedom =       38

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

*Chi-Square Test*

- Using the two categorical variables “female” and “Education”, use `tab` command with the `chi2` option to perform a chi-square test
- `tab female Education, chi2`

```
. tab female Education
```

| =1 if female | Highest Education Achieved | | | | Total |
|---|---|---|---|---|---|
| | Less than | High Scho | Undergrad | Graduate | |
| 0 | 3 | 8 | 7 | 1 | 19 |
| 1 | 4 | 7 | 8 | 2 | 21 |
| Total | 7 | 15 | 15 | 3 | 40 |

*Regression*

- Using “price” as the dependent variable, and “mpg” as the predictor:
- `regress pizza income`

```
. regress pizza income
```

| Source | SS | df | MS | Number of obs | = | 40 |
|---|---|---|---|---|---|---|
| | | | | F(1, 38) | = | 5.95 |
| Model | 128366.057 | 1 | 128366.057 | Prob > F | = | 0.0195 |
| Residual | 819285.843 | 38 | 21560.1538 | R-squared | = | 0.1355 |
| | | | | Adj R-squared | = | 0.1127 |
| Total | 947651.9 | 39 | 24298.7667 | Root MSE | = | 146.83 |

| pizza | Coef. | Std. Err. | t | P>\|t\| | [95% Conf. | Interval] |
|---|---|---|---|---|---|---|
| income | .0014577 | .0005974 | 2.44 | 0.019 | .0002483 | .002667 |
| _cons | 128.9803 | 34.59125 | 3.73 | 0.001 | 58.95401 | 199.0067 |

- `regress pizza income age female`

```
. regress pizza income age female
```

| Source | SS | df | MS | Number of obs | = | 40 |
|---|---|---|---|---|---|---|
| | | | | F(3, 36) | = | 13.72 |
| Model | 505479.049 | 3 | 168493.016 | Prob > F | = | 0.0000 |
| Residual | 442172.851 | 36 | 12282.5792 | R-squared | = | 0.5334 |
| | | | | Adj R-squared | = | 0.4945 |
| Total | 947651.9 | 39 | 24298.7667 | Root MSE | = | 110.83 |

| pizza | Coef. | Std. Err. | t | P>\|t\| | [95% Conf. | Interval] |
|---|---|---|---|---|---|---|
| income | .0017427 | .000481 | 3.62 | 0.001 | .0007671 | .0027183 |
| age | -2.83686 | 1.445795 | -1.96 | 0.058 | -5.769068 | .0953477 |
| female | -178.245 | 35.14504 | -5.07 | 0.000 | -249.5224 | -106.9675 |
| _cons | 296.6359 | 48.15155 | 6.16 | 0.000 | 198.98 | 394.2917 |

## Exporting Results

*Exporting Regression*

- First install the esstab package
- `ssc install esttab`
- Then run your regression models, and store them after the estimation, by using the command `estimates store M1` (M1 is an arbitrary name)
- Use esttab to create a single table with all the coefficients. This can be a .csv or .rtf file
- `esttab M1 M2 M3 using output.rtf`

Example:

```
ssc install esttab
reg pizza income
estimate store M1
reg pizza income age
estimate store M2
reg pizza income age female
estimate store M3
reg pizza income age female i.Education
estimate store M4

esttab M1 M2 M3 M4 using output.rtf
```

| | (1) pizza | (2) pizza | (3) pizza | (4) pizza |
|---|---|---|---|---|
| income | 0.00146* | 0.00183** | 0.00174*** | 0.00190*** |
| | (2.44) | (2.95) | (3.62) | (3.63) |
| age | | -3.222 | -2.837 | -3.106* |
| | | (-1.73) | (-1.96) | (-2.13) |
| female | | | -178.2*** | -169.7*** |
| | | | (-5.07) | (-4.88) |
| 0.Education | | | | 0 |
| | | | | (.) |
| 1.Education | | | | 87.43 |
| | | | | (1.72) |
| 2.Education | | | | 47.15 |
| | | | | (0.89) |
| 3.Education | | | | -20.92 |
| | | | | (-0.26) |
| _cons | 129.0*** | 211.0*** | 296.6*** | 244.6*** |
| | (3.73) | (3.62) | (6.16) | (4.35) |
| *N* | 40 | 40 | 40 | 40 |

*t* statistics in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

*Using Logs*

- A log allows you to record everything that happens during your Stata session that appears in the Results/Console window
- `log using "my_log"`
- `log close`

*Exporting Data*

- To export your data as a Stata file (.dta) use the `save` command
- `save pizzaData`
- To export your data as an Excel file (.xls) use the `export excel` command
- `export excel using "pizzaData"`