ECON 326: Economics of Developing Countries
TA Session 1

Vaidehi Parameswaran (Northwestern Econ)

April 2025

Today

Today

- Introductions
- Stata:
  1. The basics
  2. Data management
  3. Data visualisation
  4. Data analysis (OLS, Binary variables)

Introductions

Introductions

- Me: a second-year grad student in the econ department
  - I hope to study labour markets, monopsony power in India
  - Email: vaidehiparameswaran2029@u.northwestern.edu
  - Office hours: Monday 2-3pm in KGH 3411, Friday 2-3pm over zoom

Stata Basics

Why Stata?

Advantages of Stata:

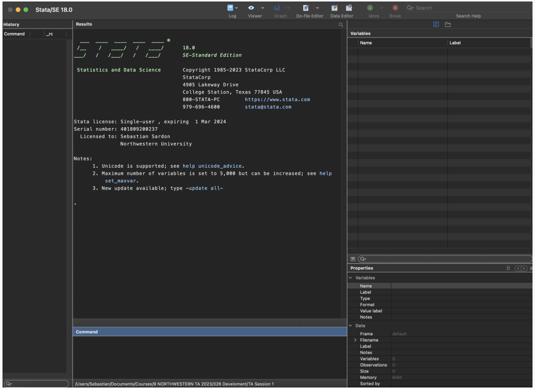
- **Easy to use**: you won't need to spend much time updating it, installing packages, or debugging your code (relative to, e.g., R and Python)
- **Well documented**: you can figure out how any command works writing `help` (or just `h`) into the prompt, followed by the command's name
- **Widely used**: Stata has a lot of built-in commands for the kind of econometrics you'll be doing in this class and is widely used in applied micro research

Disadvantages of Stata

- **Cost**: Stata is not free, but you can access it through the university
- **Not open source**: Needs a license, can't see the source code
- **Limited for tasks like ML**: If you're interested in machine learning, you might want to learn R or Python

The Interface

- Open Stata to see the **Results Window**



```
  ___  ____  ____  ____  ____ ®
 /__    /   ____/   /   ____/      18.0
___/   /   /___/   /   /___/       SE—Standard Edition

 Statistics and Data Science       Copyright 1985-2023 StataCorp LLC
                                   StataCorp
                                   4905 Lakeway Drive
                                   College Station, Texas 77845 USA
                                   800-STATA-PC        https://www.stata.com
                                   979-696-4600        stata@stata.com

Stata license: Single-user , expiring  1 Mar 2024
Serial number: 401809200237
  Licensed to: Sebastian Sardon
               Northwestern University

Notes:
      1. Unicode is supported; see help unicode_advice.
      2. Maximum number of variables is set to 5,000 but can be increased; see help
          set_maxvar.
      3. New update available; type -update all-

.
```

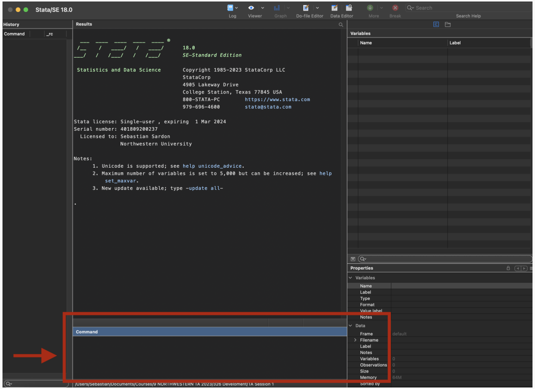
Variables
Name
Label
Type
Format
Value label
Notes

Data
Frame default
Filename
Label
Notes
Variables 0
Observations 0
Size 0
Memory 64M
Sorted by

/Users/Sebastian/Documents/Courses/9 NORTHWESTERN TA 2023/326 Develoment/TA Session 1

The Interface

- **Command Prompt:** allows you to type and run commands (try "`di`" followed by what you want Stata to display)



```
  ___  ____  ____  ____  ____ ®
 /__    /   ____/   /   ____/      18.0
___/   /   /___/   /   /___/       SE—Standard Edition

 Statistics and Data Science       Copyright 1985-2023 StataCorp LLC
                                   StataCorp
                                   4905 Lakeway Drive
                                   College Station, Texas 77845 USA
                                   800-STATA-PC        https://www.stata.com
                                   979-696-4600        stata@stata.com

Stata license: Single-user , expiring  1 Mar 2024
Serial number: 401809200237
  Licensed to: Sebastian Sardon
               Northwestern University

Notes:
      1. Unicode is supported; see help unicode_advice.
      2. Maximum number of variables is set to 5,000 but can be increased; see help
          set_maxvar.
      3. New update available; type -update all-

.
```

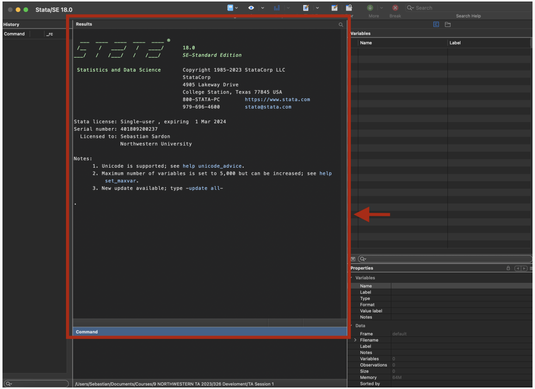
Variables
- Name
- Label
- Type
- Format
- Value label
- Notes

Data
- Frame: default
- Filename
- Label
- Notes
- Variables: 0
- Observations: 0
- Size: 0
- Memory: 64M
- Sorted by

/Users/Sebastian/Documents/Courses/9 NORTHWESTERN TA 2023/326 Develoment/TA Session 1

The Interface

- **Results** panel will contain the output of all executed commands



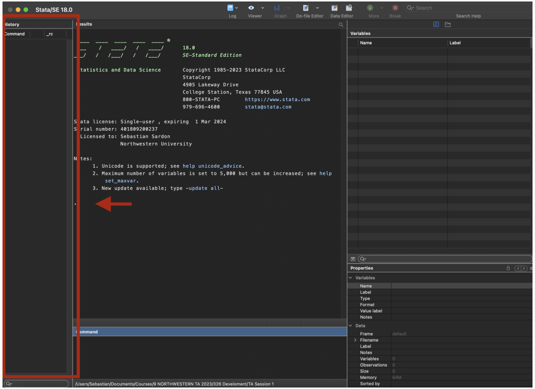
```
  ___  ____  ____  ____  ____ ®
 /__    /   ____/   /   ____/      18.0
___/   /   /___/   /   /___/       SE—Standard Edition

 Statistics and Data Science       Copyright 1985-2023 StataCorp LLC
                                   StataCorp
                                   4905 Lakeway Drive
                                   College Station, Texas 77845 USA
                                   800-STATA-PC        https://www.stata.com
                                   979-696-4600        stata@stata.com

Stata license: Single-user , expiring  1 Mar 2024
Serial number: 401809200237
  Licensed to: Sebastian Sardon
               Northwestern University

Notes:
      1. Unicode is supported; see help unicode_advice.
      2. Maximum number of variables is set to 5,000 but can be increased; see help
          set_maxvar.
      3. New update available; type -update all-

.
```

Frame default
Variables 0
Observations 0
Size 0
Memory 64M

/Users/Sebastian/Documents/Courses/9 NORTHWESTERN TA 2023/326 Develoment/TA Session 1

The Interface

- **History** panel will list all commands sent to the prompt



```
                                      18.0
                                      SE—Standard Edition

 tatistics and Data Science          Copyright 1985-2023 StataCorp LLC
                                      StataCorp
                                      4905 Lakeway Drive
                                      College Station, Texas 77845 USA
                                      800-STATA-PC        https://www.stata.com
                                      979-696-4600        stata@stata.com

S ata license: Single-user , expiring  1 Mar 2024
S rial number: 401809200237
  Licensed to: Sebastian Sardon
               Northwestern University

N tes:
      1. Unicode is supported; see help unicode_advice.
      2. Maximum number of variables is set to 5,000 but can be increased; see help
          set_maxvar.
      3. New update available; type -update all-

.
```

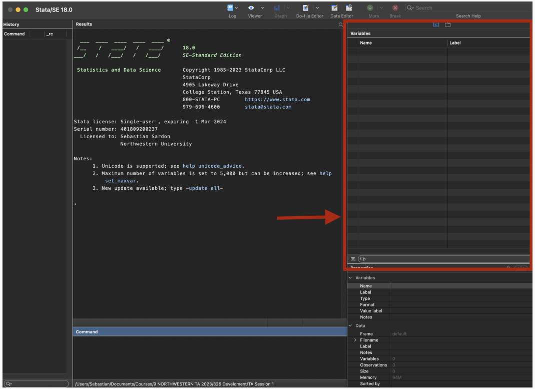
Variables
- Name
- Label
- Type
- Format
- Value label
- Notes

Data
- Frame default
- Filename
- Label
- Notes
- Variables 0
- Observations 0
- Size 0
- Memory 64M
- Sorted by

/Users/Sebastian/Documents/Courses/9 NORTHWESTERN TA 2023/326 Develoment/TA Session 1

The Interface

- **Variables** panel will show you all variables in the dataset that's currently open



```
  ___  ____  ____  ____  ____ ®
 /__    /   ____/   /   ____/      18.0
___/   /   /___/   /   /___/       SE—Standard Edition

 Statistics and Data Science       Copyright 1985-2023 StataCorp LLC
                                   StataCorp
                                   4905 Lakeway Drive
                                   College Station, Texas 77845 USA
                                   800-STATA-PC        https://www.stata.com
                                   979-696-4600        stata@stata.com

Stata license: Single-user , expiring  1 Mar 2024
Serial number: 401809200237
  Licensed to: Sebastian Sardon
               Northwestern University

Notes:
      1. Unicode is supported; see help unicode_advice.
      2. Maximum number of variables is set to 5,000 but can be increased; see help
          set_maxvar.
      3. New update available; type -update all-

.
```

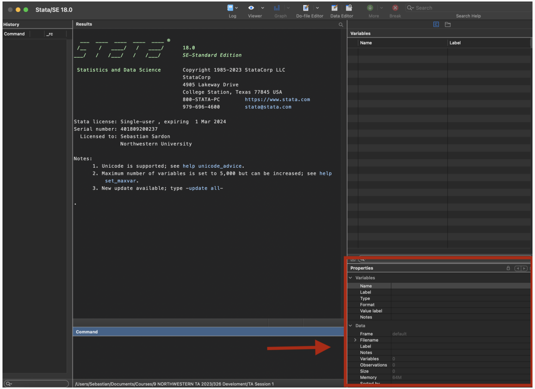
Variables
Name
Label
Type
Format
Value label
Notes

Data
Frame: default
Filename
Label
Notes
Variables: 0
Observations: 0
Size: 0
Memory: 64M
Sorted by

/Users/Sebastian/Documents/Courses/9 NORTHWESTERN TA 2023/326 Develoment/TA Session 1

The Interface

- **Properties** panel will show additional information about the currently open dataset



```
  ___  ____  ____  ____  ____ ®
 /__    /   ____/   /   ____/      18.0
___/   /   /___/   /   /___/       SE—Standard Edition

 Statistics and Data Science       Copyright 1985-2023 StataCorp LLC
                                   StataCorp
                                   4905 Lakeway Drive
                                   College Station, Texas 77845 USA
                                   800-STATA-PC        https://www.stata.com
                                   979-696-4600        stata@stata.com

Stata license: Single-user , expiring  1 Mar 2024
Serial number: 401809200237
  Licensed to: Sebastian Sardon
               Northwestern University

Notes:
      1. Unicode is supported; see help unicode_advice.
      2. Maximum number of variables is set to 5,000 but can be increased; see help
          set_maxvar.
      3. New update available; type -update all-

.
```

Properties

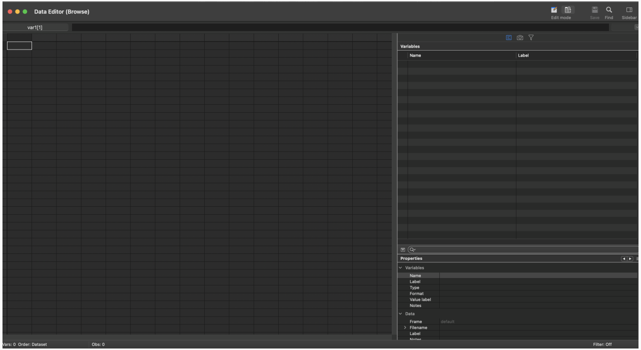
Variables
- Name
- Label
- Type
- Format
- Value label
- Notes

Data
- Frame: default
- Filename
- Label
- Notes
- Variables: 0
- Observations: 0
- Size: 0
- Memory: 64M
- Sorted by

/Users/Sebastian/Documents/Courses/9 NORTHWESTERN TA 2023/326 Develoment/TA Session 1

The Interface

- There’s also a **Data Editor** window that allows you to visualize the currently open dataset
- To open it, use the `br` command



The Interface

- The **Do-File Editor** is where we do most of our work
- A highly optimized text editor for writing Stata code
- You open it by opening any `.do` file, or in the menu bar via **File**→ **New** → **Do-file**.
- You can select the lines you want to run, then run them with **Shift**+**CMD**+**D** (Mac) or **CTRL**+**D** (Windows)
- Use the same shortcut keys without selection to run the entire do-file

Writing code in Stata

- Usual Preamble: typically start any do-file with these 4 commands
  1. `clear all` → Drops all variables
  2. `cd "/Users/vaidehiparameswaran/Desktop/teaching/econ326-sp/"` → Change Directory: this indicates where you will open from and save your files. All paths in the dofile will automatically get this prefix.
  3. `capture log close` → *If* a log file was opened, this closes it. `capture` suppresses output and potential error messages which would terminate the dofile. Here, we'd get an error the first time we run the dofile: no log would be open.
  4. `log using TA1.txt, replace` → Opens a log file (called `TA1`) that will record the contents of the Results window. This lets others verify your code actually worked.

Comment your code

- This means writing notes that won't run as commands, useful to annotate do-file and divide into sections
- Two ways to comment your code:
  - Typing * at the beginning of a line allows you to comment
  - You can also insert a comment after a command line by adding // after the command
  - Example:

```
display("Hello World!") //This line prints statement in results window
```

Stata does math

- Operators: `display` (or equivalently, `di`) also works as a calculator
  - `display` $2 * 3 + 3/2 - 2^\wedge 3$ → This will give you -0.5. Notice Stata follows the standard order of operations so you don't need brackets unless you want to change their order.
- Mathematical functions
  - `display ln(sqrt(abs(-2)))` → This will give you 0.34657359, the result of $\log(\sqrt{|-2|})$.

String Functions

- **string variables** are those containing letters (and other characters) instead of numbers (e.g. country names). You use string functions to manipulate them:
  - `display substr("abc",1,2)` → This tells Stata that, starting from position 1, in the string "abc", i.e. starting from "a", it should keep 2 characters. The result would be "ab".
  - `display subinstr("abc","b","X",1)` → This tells Stata to replace a substring ("b") within a string ("abc") with another substring("X"). The last argument, "1", indicates how many times such replacements are going to be made. Here "b" only occurs once within the string so this does not matter. The result would be "aXc".
  - If we replace the "1" from above with a period, we are telling Stata to do the replacement as many times as possible. For example: `display subinstr("abcbb","b","X",.)` yields "aXcXX".

Data Management

- We will work with NLS data on wage and education
- To begin, run the preamble commands
- Then open the data:

```
use "rawdata.dta", clear
```

- The `use` command opens the data set from the directory you set earlier in the preamble with `cd`
- `clear` is an option that specifies that it is okay to replace the dataset in memory. You need to include this to avoid errors in case you already had a dataset loaded into Stata
- **In general, a comma will separate a Stata command and its options**

Understanding your data

- `describe` → Overview of all your variables (types, formats and labels)
- `codebook` → Prints the codebook of your data with a description of each variable
- `browse` or `br` → Navigate dataset as if it was an Excel spreadsheet
- `count` → Total number of observations in your data

Understanding the data

- `tabulate x` → This command gives you a frequency table for the variable specified
- `tabulate x y` → A 2 by 2 frequency table. The first variable will be displayed in the rows and the second in columns
- `sum x` → summarize statistics of the variable `x` (short version: number of observations, mean, standard deviation, min, max)
- `sum x, d` → detailed version of `sum`, useful to inspect the distribution of a variable

Generating Variables

- `gen lwage = ln (wage)` → The `gen` command generates new variables
- `drop lwage` → The `drop` command deletes variables
- `gen lwage = ln(wage) if educ >= 13`
- `replace lwage = ln(wage) if educ < 13`
- `label var lwage "log wages"` → The `label var` command labels variables

Saving your work!

- `cap mkdir output` → create a folder (directory) called `output` – this goes inside the folder we are working within as set with `cd` above
- `save "output/nls88.dta", replace` → save the modified dataset; `replace` tells Stata to overwrite its previous version (if any)

Scatter Plots

- Suppose we want to create a scatterplot of hourly wages and total experience.
- This is easy to do with the command `twoway scatter wage ttl_exp`

Other Graphs

- `hist X` → histogram to visualize the distribution of a variable
- `kdensity X` → estimate probability density function of a variable

OLS

- Suppose we want to estimate the linear regression model relating wage with education, age, and experience.

$$\log lwage_i = \alpha + \beta_1 educ_i + \beta_2 age_i + \beta_3 exp_i + \mathbf{X_i}'\gamma + \epsilon_i$$

- $\mathbf{X_i}$ is a vector of controls (ignore for now)

OLS

- $$\log lwage_i = \alpha + \beta_1 educ_i + \beta_2 age_i + \beta_3 exp_i + \mathbf{X_i}'\gamma + \epsilon_i$$
- We want to estimate β_1: the effect of education on wages
- We can estimate α with the ordinary least squares (OLS) estimator as follows

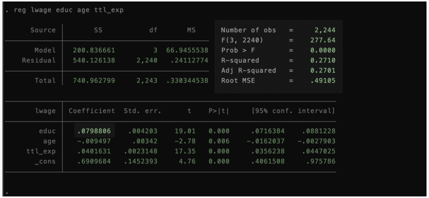
```
reg lwage educ age ttl_exp
```

- Additional controls (vector $\mathbf{X}$) can be added. For example,

```
reg lwage educ age ttl_exp tenure industry
```

OLS

- By running the simple regression command `reg lwage educ age ttl_exp` we get:

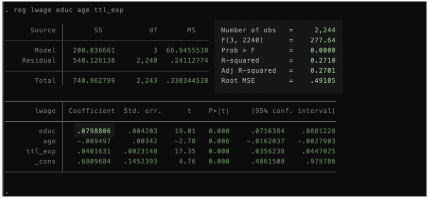


```
. reg lwage educ age ttl_exp

      Source |       SS           df       MS      Number of obs   =     2,244
-------------+----------------------------------   F(3, 2240)      =    277.64
       Model |  200.836661         3  66.9455538   Prob > F        =    0.0000
    Residual |  540.126138     2,240  .24112774    R-squared       =    0.2710
-------------+----------------------------------   Adj R-squared   =    0.2701
       Total |  740.962799     2,243  .330344538   Root MSE        =    .49105

------------------------------------------------------------------------------
       lwage | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
        educ |   .0798806    .004203    19.01   0.000     .0716384    .0881228
         age |   -.009497     .00342    -2.78   0.006    -.0162037   -.0027903
     ttl_exp |   .0401631   .0023148    17.35   0.000     .0356238    .0447025
       _cons |   .6909684   .1452393     4.76   0.000     .4061508     .975786
------------------------------------------------------------------------------

.
```

- We also get
  - The OLS estimator's standard error (square root of its variance, SE)
  - Its $t-$statistic (ratio of coefficient and SE)
  - Its $p-$value (Probability of obtaining an estimate at least as big as ours, and with the same sign, in a world where $\beta_1 = 0$ (i.e., "under the null hypothesis"))
  - A confidence interval where the true value of β_1 lies with 95% probability
  - An estimate of the intercept α (see the _cons statistic below the highlighted one)

OLS

- By running the simple regression command `reg lwage educ age ttl_exp` we get:



```
. reg lwage educ age ttl_exp

      Source |       SS           df       MS      Number of obs   =     2,244
-------------+----------------------------------   F(3, 2240)      =    277.64
       Model |  200.836661         3  66.9455538   Prob > F        =    0.0000
    Residual |  540.126138     2,240  .24112774    R-squared       =    0.2710
-------------+----------------------------------   Adj R-squared   =    0.2701
       Total |  740.962799     2,243  .330344538   Root MSE        =    .49105

------------------------------------------------------------------------------
       lwage | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
        educ |   .0798806    .004203    19.01   0.000     .0716384    .0881228
         age |   -.009497     .00342    -2.78   0.006    -.0162037   -.0027903
     ttl_exp |   .0401631   .0023148    17.35   0.000     .0356238    .0447025
       _cons |   .6909684   .1452393     4.76   0.000     .4061508     .975786
------------------------------------------------------------------------------

.
```

- In addition, Stata reports the regression's R^2
- This indicates the share of the variation in `lwage` that is explained by `educ`
- But correlation $\neq$ causation.

Binary Variables

- Binary variables (that take only two values, for e.g., 0 or 1) are common in economics
- Constructing binary variables in Stata is easy:

```
gen collgrad = 1 if educ >= 13
```

```
replace collgrad = 0 if educ < 13
```

- Or you can do it in one line:

```
gen collgrad = (educ >= 13)
```

Binary Variables

- Regressing a dependent variable on a binary variable gives us the average difference in the dependent variable between the two groups
- To see this, let's regress the log of wages on a binary variable indicating whether the individual is a college graduate
- We can do this with the command `reg lwage collgrad`
- Now let's also get the mean wages for college graduates and non-college graduates

See you next time!