

# Development Data Boot Camp

## Intro to Stata: Quick Start

Ge Sun

University of Notre Dame

May 15, 2023

# Outline

Stata's Interface

Browse the Data

Basic Commands and Syntax

Importing Data from External Resources

Data Types in Stata

Explore the Data

Directory and Path

# Outline

## Stata's Interface

Browse the Data

Basic Commands and Syntax

Importing Data from External Resources

Data Types in Stata

Explore the Data

Directory and Path

# Stata's interface

- Double-clicking on the Stata's icon, and you can see the interface of Stata:

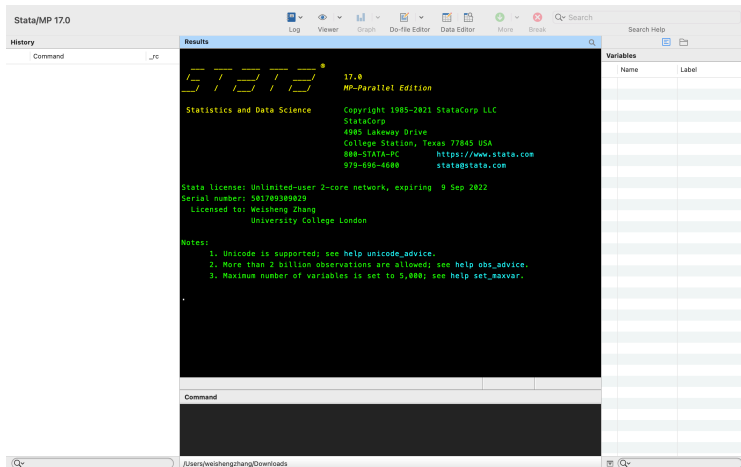
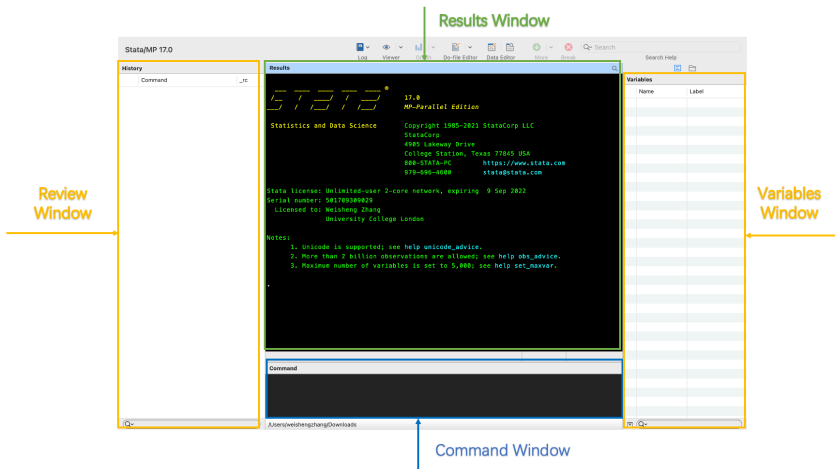


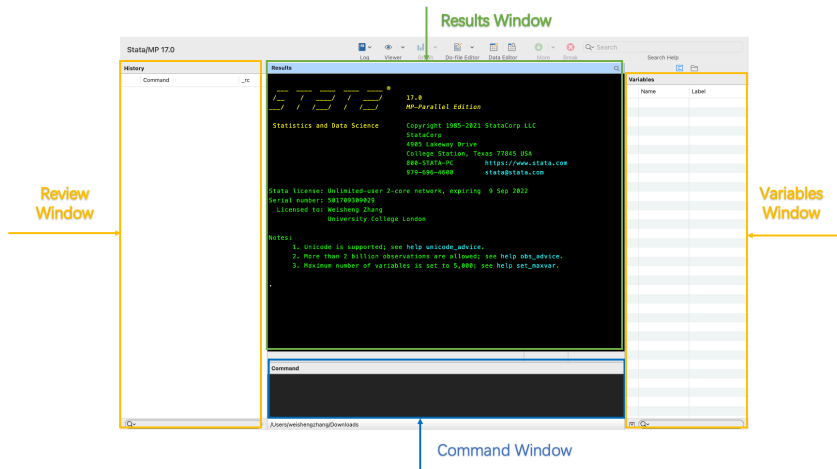
Figure 1: Stata's Interface

# Four Windows



- Review window: displays previous Stata commands entered in the current session.
- Results window: displays all the commands and relative results or feedbacks.

# Four Windows



- Stata command window: this is where you type commands.
- Variables window: lists all the variables in the currently open dataset

# Stata's interface

## A Quick Exercise

Please type the following command in the command window (command line) and then press the enter key.

```
sysuse auto.dta, clear
```

Can you tell me what happens in each of the four windows?

- \* *The meaning of this command:*

Use the datasets named *auto.dta* shipped with Stata, and if there is an old dataset in the memory, clear it before importing *auto.dta*.

# Stata's interface

## A Quick Exercise

Please type the following command in the command window (command line) and then press the enter key.

```
sysuse auto.dta, clear
```

Can you tell me what happens in each of the four windows?

- \* *The meaning of this command:*

Use the datasets named *auto.dta* shipped with Stata, and if there is an old dataset in the memory, clear it before importing *auto.dta*.

Where the data imported are?



# Outline

Stata's Interface

**Browse the Data**

Basic Commands and Syntax

Importing Data from External Resources

Data Types in Stata

Explore the Data

Directory and Path

# Browse the Data

Two ways to browse the data in the memory:

- ▶ Type *browse* in the command window
- ▶ Or Click the icon *Data Editor (Browse)*.
  - ▶ Notice there are two similar icons, *Data Editor (Browse)* and *Data Editor (Edit)*. We can not edit dataset under *Data Editor (Browse)*, while we can edit the dataset under *Data Editor (Edit)*, such as change the name of variable or delete some samples.
  - ▶ *Data Editor (Browse)* is recommended as in most cases, we do not want to directly change the dataset in the data editor.

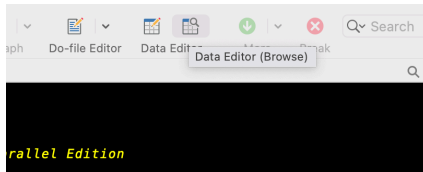


Figure 2: Click Data Editor (Browse)

# Data Editor in Stata

Data Editor (Browse) — auto.dta

Stata/MP 17.0 — auto.dta

Data Editor (Browse) — auto.dta

make[1] AMC Concord

	make	price	mpg	rep78	headroom	trunk	weight	length	turn	displacement	gear_
1	AMC Concord	4,099	22	3	2.5	11	2,930	186	40	121	
2	AMC Pacer	4,749	17	3	3.0	11	3,350	173	40	258	
3	AMC Spirit	3,799	22	.	3.0	12	2,640	168	35	121	
4	Buick Century	4,816	20	3	4.5	16	3,250	196	40	196	
5	Buick Electra	7,827	15	4	4.0	20	4,080	222	43	350	
6	Buick LeSabre	5,788	18	3	4.0	21	3,670	218	43	231	
7	Buick Opel	4,453	26	.	3.0	10	2,230	170	34	304	
8	Buick Regal	5,189	20	3	2.0	16	3,280	200	42	196	
9	Buick Riviera	10,372	16	3	3.5	17	3,880	207	43	231	
10	Buick Skylark	4,082	19	3	3.5	13	3,400	200	42	231	
11	Cad. Deville	11,385	14	3	4.0	20	4,330	221	44	425	
12	Cad. Eldorado	14,500	14	2	3.5	16	3,900	204	43	350	
13	Cad. Seville	15,906	21	3	3.0	13	4,290	204	45	350	
14	Chev. Chevette	3,299	29	3	2.5	9	2,110	163	34	231	
15	Chev. Impala	5,705	16	4	4.0	20	3,690	212	43	250	
16	Chev. Malibu	4,504	22	3	3.5	17	3,180	193	31	200	
17	Chev. Monte Carlo	5,104	22	2	2.0	16	3,220	200	41	200	
18	Chev. Monza	3,667	24	2	2.0	7	2,750	179	40	151	
19	Chev. Nova	3,955	19	3	3.5	13	3,430	197	43	250	
20	Dodge Colt	3,984	30	5	2.0	8	2,120	163	35	98	
21	Dodge Diplomat	4,010	18	2	4.0	17	3,600	206	46	318	
22	Dodge Magnum	5,886	16	2	4.0	17	3,600	206	46	318	
23	Dodge St. Regis	6,342	17	2	4.5	21	3,740	220	46	225	
24	Ford Fiesta	4,389	28	4	1.5	9	1,800	147	33	98	
25	Ford Mustang	4,187	21	3	2.0	10	2,650	179	43	140	
26	Linc. Continental	11,497	12	3	3.5	22	4,840	233	51	400	
27	Linc. Mark V	13,594	12	3	2.5	18	4,720	230	48	400	
28	Linc. Versailles	13,466	14	3	3.5	15	3,830	201	41	302	
29	Merc. Bobcat	3,829	22	4	3.0	9	2,580	169	39	140	
30	Merc. Cougar	5,379	14	4	3.5	16	4,060	221	48	302	

Vars: 12 Order: Dataset Obs: 74

Variables

Name	Label
<input checked="" type="checkbox"/> make	Make and model
<input checked="" type="checkbox"/> price	Price
<input checked="" type="checkbox"/> mpg	Mileage (mpg)
<input checked="" type="checkbox"/> rep78	Repair record 1978
<input checked="" type="checkbox"/> headroom	Headroom (in.)
<input checked="" type="checkbox"/> trunk	Trunk space (cu. ft.)
<input checked="" type="checkbox"/> weight	Weight (lbs.)
<input checked="" type="checkbox"/> length	Length (in.)
<input checked="" type="checkbox"/> turn	Turn circle (ft.)
<input checked="" type="checkbox"/> displacement	Displacement (cu. in.)
<input checked="" type="checkbox"/> gear_ratio	Gear ratio
<input checked="" type="checkbox"/> foreign	Car origin

Properties

Variables

Name	make
Label	Make and model
Type	str18
Format	%-18s
Value label	
Notes	

Data

Frame	default
> Filename	auto.dta
Label	1978 automobiles data
Length	18
Filter	Off

Figure 3: Data Browse

# Data Structure in Stata

## Data Structure in Stata:

- ▶ Column represents the variable, like the make and model of each car in this dataset, “make”, the price of cars, “price”, etc.
- ▶ Row represents the observation, containing the information of a given observation.
  - \* *What are the observations in this dataset?*
- ▶ Each cell contains a particular variable for a particular observation.
- ▶ Not every cell is filled with a value, some values are missing, which represented as .

# Data Structure in Stata

## Data Structure in Stata:

- ▶ Column represents the variable, like the make and model of each car in this dataset, “make”, the price of cars, “price”, etc.
- ▶ Row represents the observation, containing the information of a given observation.

*\* What are the observations in this dataset?*

- ▶ Each cell contains a particular variable for a particular observation.
- ▶ Not every cell is filled with a value, some values are missing, which represented as .

## Quick Question:

The 48th row documents a car's information, could you please tell me how many repair records it had in 1978?

# Outline

Stata's Interface

Browse the Data

**Basic Commands and Syntax**

Importing Data from External Resources

Data Types in Stata

Explore the Data

Directory and Path

# Variable Label

- ▶ Variable label is an attribute attaching to a variable. It is very important as it provides a short description of the variable.
- ▶ You are encouraged to add a label on the important variable because it can remind you of the meaning of the variable later. It also helps other people understand your dataset.

# Variable Label

- ▶ Variable label is an attribute attaching to a variable. It is very important as it provides a short description of the variable.
- ▶ You are encouraged to add a label on the important variable because it can remind you of the meaning of the variable later. It also helps other people understand your dataset.
- ▶ Two commands for label reading:

```
des [var]  
codebook [var]
```

- ▶ To create label for variables:

```
label variable [var] "var_label"
```



## Generate New Variables

- ▶ Command “generate” is used to generate new variables, we often use its abbreviation “gen”. From example:
  - ▶ `gen new_var = 1`: generate a new variable `new_var` which value takes 1.
  - ▶ `gen wtsq = weight * weight`: generate a new variable `wtsq` which value is squared of variable `weight`.

### Exercise

Generate `wtsq` variable, and label it with something that makes sense to you.

## Generate New Variables

- ▶ Command “generate” is used to generate new variables, we often use its abbreviation “gen”. From example:
  - ▶ *gen new\_var = 1*: generate a new variable *new\_var* which value takes 1.
  - ▶ *gen wtsq = weight \* weight*: generate a new variable *wtsq* which value is squared of variable *weight*.

### Exercise

Generate *wtsq* variable, and label it with something that makes sense to you.

- ▶ When we use *generate*, the variable generated can not be defined before. For example, if we type

```
gen new_var = 1
```

```
gen new_var = 2
```

there would be an error message says

variable new\_var already defined

## Editing Existing Variables: *replace* and *rename*

- ▶ We use command *replace* to edit the name of existing variables. For example: We have already generated a variable *new\_var* which value is 1, and now we want to change the value of *new\_var* from 1 to 2:

```
replace new_var = 2
```

- ▶ Command *rename* allows us to change variable names. For example, we want to change the name for variable *wtsg* , then

```
rename wtsg weight_square
```

implies that *wtsg* is now called *weight\_square*, and now it is legal to generated a new variable named *wtsg* by command *generate*.

## Command: *drop* and *keep*

- ▶ Command *drop* allows us to remove variables or observations from the dataset in memory. For example, if we type

```
drop weight_square
```

the variable *weight\_square* will be removed.

- ▶ Command *keep* is like another way to drop variables because after you execute this command, only the variables that you specify will continue to be “kept”.

```
keep make price mpg rep78 headroom
```

Also, we can keep data by the number/order of observations:

```
keep in 1/5
```

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

```
sysuse auto.dta, clear
keep make price mpg rep78
keep in 1/5
```

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

- ▶ The default operation in Stata is “column by column” not “cell-by-cell”.
  - \* *This gives Stata the power to do regression analysis efficiently but also makes it not a true “programming” language because we can not handle single cell (element) as freely as we wish (Stata can not support element by element operation).*

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

`gen mpg_2 = mpg + 1`

- Where would variable `mpg_2` appear?



## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

`gen mpg_2 = mpg + 1`

- ▶ Where would variable `mpg_2` appear?
- ▶ What are the values in `mpg_2`?

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

`replace rep78 = 0`

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

```
replace rep78 = 0  
drop rep78
```

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

```
replace rep78 = 0  
drop rep78  
keep make
```

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

```
replace rep78 = 0  
drop rep78  
keep make  
egen mean_mpg = mean(mpg)
```

## Digression but Important: Understanding Columns as the Analysis Unit in Stata

	make	price	mpg	rep78	headroom	
1	AMC Concord	4,099	22	3	2.5	
2	AMC Pacer	4,749	17	3	3.0	
3	AMC Spirit	3,799	22	.	3.0	
4	Buick Century	4,816	20	3	4.5	
5	Buick Electra	7,827	15	4	4.0	

```
replace rep78 = 0
drop rep78
keep make
egen mean_mpg = mean(mpg)
egen mean_rep78 = mean(rep78)
```

## Command: help

- ▶ *help* command displays detailed information about the specified. For example, type

help rename

and there will be a window displaying the document of how to use *rename*.

- ▶ Sometimes we do not remember the name of the Stata command, then we can use command *search* by keyword. For example, type

search label

and Stata will display a list of commands and resources involving keyword “label”.

# Basic Stata Grammar

The basic structure of grammar is:

*[prefix] : command [varlist] [= exp ] [if] [in] [weight] [using filename], [ options]*

- ▶ *prefix* denotes a command that repeats execution of *command* or modifies the input or output of *command*.

bysort county: egen average\_income\_country = mean(income)

- ▶ *command* denotes a Stata command,
- ▶ *varlist* denotes a list of variable names,
- ▶ *exp* is a mathematical expression,
- ▶ *weight* denotes a weighting expression,
- ▶ *filename* is a filename, and
- ▶ *options* denotes one or more options that apply to *command*.



# Outline

Stata's Interface

Browse the Data

Basic Commands and Syntax

Importing Data from External Resources

Data Types in Stata

Explore the Data

Directory and Path

# Importing Data from External Sources

- ▶ Stata can read other types of data from external sources including:
  - ▶ Excel spreadsheet (\*.xls, \*.xlsx)
  - ▶ text data (\*.csv)
  - ▶ Stata data (\*.dta)
  - ▶ DCT file (\*.dct)
- ▶ Stata has various commands for importing different kinds of data from external databases:
  - ▶ The *import excel* command can read Microsoft Excel files.
  - ▶ The *import delimited* or *insheet using* command can read \*.csv files.
  - ▶ The *use* command can read \*.dta files. (sysuse auto.dta)
  - ▶ The *infile* command can read \*.dct file.

# Importing data from external sources

The basic syntax of importing external data is:

```
import file_type using file_name, specific_options_wrt_file_type
```

- ▶ Replace *file\_type* with the type of your file, like *excel* or *spss*.
- ▶ Replace *file\_name* with the path and the name of your file, like */Users/sybil/Documents/school\_gps\_map.csv*.
- ▶ There are various options with respect to your type of file. For example, some widely used options for Excel spreadsheet are:
  - ▶ *firstrow*: treat first row of Excel data as variable names.
  - ▶ *sheet("sheetname")*: decide which Excel worksheet will be imported into Stata.
  - ▶ *clear*: replace data currently used in Stata (memory).

# Importing data from external sources

- ▶ The syntax of importing external data into Stata can be complex and hard to remember. Fortunately, we do not need to remember it with the help of GUI.
- ▶ As a beginner, we can use the following steps: (1) Import the data using the GUI, and (2) Copy the code that Stata generates automatically and paste it into your do-file, so that the process is reproducible.

# Importing data from external sources.

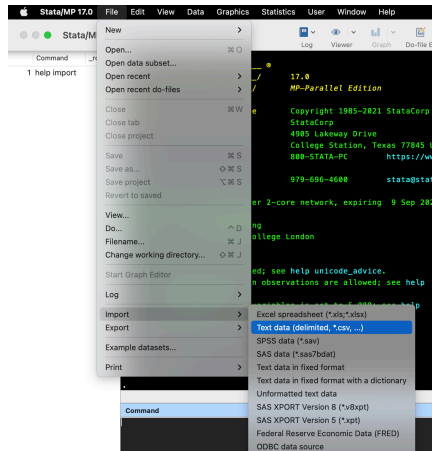


Figure 4: Choose the corresponding command

# Importing data from external sources.

import delimited - Import delimited text data

File to import:  
/Users/weishengzhang/Documents/Courses/TA/01data/school\_gps\_map.csv

Delimiter: Automatic ,

First row as variable names: Automatic

Variable name case: Lower

Floating point precision: Use default float

Text encoding: Automatic ISO-8859-1

Quote stripping: Automatic

Quote binding: Loose

☐ Treat sequential delimiters as one

Override row and column ranges...

Numeric parsing rules

☒ Use locale based parsing for numbers

Locale: ccp

Override decimal separator:

Override grouping separator:

Preview:

#	latitude	longitude	altitude	accuracy	treatment
2	29.454182	76.981697	192.19443	4	0
3	29.41946	76.943199	180.81078	4	1
4	29.375132	76.969536	211.7562	8	0
5	29.355429	76.972504	192.44884	4	0
6	29.413313	76.94236	197.31406	6	0
7	29.389912	77.010605	181.05901	4	0
8	29.459633	77.063843	191.328	4	1
9	29.373199	76.945946	186.53212	4	0
10	29.368961	76.911125	183.79817	4	1
11	29.336014	76.981819	183.77258	10	1
12	29.360317	76.944191	189.23714	10	0
13	29.393215	76.964195	178.95555	4	1
14	29.378622	77.032768	182.98137	4	1
15	29.40078	76.984535	187.43506	4	1
16	29.457056	77.028839	183.28719	4	1

Variable Type

latitude	numeric
longitude	numeric
altitude	numeric
accuracy	numeric
treatment	numeric

To change the data type for a column, right-click on the selected column and choose the appropriate type.  
Note: importing string data as numeric can result in loss of data.

? ↺ 📄

Submit Cancel OK

Figure 5: Choose the file and options

# Outline

Stata's Interface

Browse the Data

Basic Commands and Syntax

Importing Data from External Resources

**Data Types in Stata**

Explore the Data

Directory and Path

# Types of Data

- ▶ Every variable in Stata is either type **numeric** or type **string**
- ▶ Numeric:
  - \* Numeric type refers to variables that are stored as numbers.
  - \* You can use  $>$  and  $<$  to compare them.
  - \* You can enter them into arithmetic calculations.
  - \* You can get the mean, the quantiles for numeric variables.
- ▶ String:
  - \* String type refers to variables that are stored as words/sentences.
  - \* It is easy to extract different pieces of information from one sentence.
  - \* There are some magic commands that will help us achieve that goal.



# Numeric type data

## What can numeric type data be?

- ▶ **Int** Variables that are integers.
  - \* For example: year, age, and number of siblings.
- ▶ **Float (or double)** Variables that can take continuous values.
  - \* For example: income, consumption, and assets.
- ▶ **Byte** Variables with type byte are categories, i.e., variables that can only have finite values.
  - \* dummy variables (gender, whether having an insurance, whether living in rural areas).
  - \* education level (primary school, high school, college, etc.)
  - \* race (black, white, Asian, Hispanic, etc.)

*In most cases we do not need to pay much attention to the difference between int and byte, float and double.*

# String type data

## What can string type data be?

- ▶ country (USA, UK, etc.), state (Indiana, California, etc.)
- ▶ date (like 1994/01/01, which is a string instead of a number, we will deal with that later)
- ▶ Frequently used string types are:
  - \* strX, where X maximum length of the string. For examples, str5 refers to string variables with maximum length 5.
  - \* strL. strL indicates that variables can hold strings of arbitrary lengths.

*In most cases we do not need to distinguish str3, str5, and strL.*

## Extract useful information: `split`

- ▶ Sometimes we need to contract useful information from string variables.
- ▶ Consider variable `date` in the format `1940/12/20`, we want to contract information in this variable and attribute it into three variables, `year`, `month`, and `day`.
- ▶ Command `split` can help us. The usage of is

```
split date, gen(new_var), parse(/)
```

- ▶ Command `regexm` is more powerful, you can use help document to explore it whenever you need it.

## Convert one type to the other

- ▶ Sometimes we need to convert string variables to numeric variables. It often happens when we import excel or csv data into Stata, as in this case, Stata treat every variable as strings, even the variable only contains numbers.
- ▶ Command *destring* can help us. The usage of *destring* is

`destring var, replace`

or

`destring var, generate(new_var)`

## Convert one type to the other

- ▶ Sometimes we need to convert string variables to numeric variables. It often happens when we import excel or csv data into Stata, as in this case, Stata treat every variable as strings, even the variable only contains numbers.
- ▶ Command *destring* can help us. The usage of *destring* is

`destring var, replace`

or

`destring var, generate(new_var)`

- ▶ Numeric to string: *tostring*

# Outline

Stata's Interface

Browse the Data

Basic Commands and Syntax

Importing Data from External Resources

Data Types in Stata

**Explore the Data**

Directory and Path

## Commands: describe and codebook

- ▶ Two commonly used commands to explore the dataset:
  - ▶ *describe*. Provide an overview of the dataset:
    - ▶ Number of observations in the dataset.
    - ▶ Number of variables in the dataset.
    - ▶ Other basic information about the variables in our dataset, like types, labels, etc.
  - ▶ *codebook*. Provide more detailed information of variables in the dataset, like range and mean.
- ▶ Commands *describe* and *codebook* can also be used as exploring specific variables.

## Commands: lookfor and count

- ▶ *lookfor* helps you find variables by searching for string among all variable names and labels.
  - \* It helps find the relevant information you need, especially in a large dataset with tons of variables.
- ▶ *count* displays the number of observations in the data.
  - \* You can add conditions to ask for the number of observations that satisfy the specified conditions.  
count if rep78 > 3
  - \* It is very useful in debugging the codes!



## Commands: `tabulate` and `summarize`

- ▶ *tabulate* is an extremely useful command in exploring data.
  - ▶ When only one variable is specified, i.e., *tab var\_name*, *tabulate* gives count of times and percentage of each value appears in data.
  - ▶ When two variable is specified, i.e., *tab var\_name\_1, var\_name\_2*, *tabulate* gives the relationship between these two variables, where *var\_name\_1* appears as row, *var\_name\_2* appears as column.
- ▶ *summarize* is useful when we want to know basic summary statistics of some variables, i.e., mean, standard error, and median.
- ▶ *tabulate* might not work if there are too many distinct values of the variable, and it is often applied on category variables, like gender and occupation. *summarize* is often applied on continuous variables, such as incomes, as it is meaningless to calculate the mean of people's occupations.

## Commands: histogram

- ▶ When we are faced with a continuous variable, it is more convenient to look at a graph than reading tedious numbers.
- ▶ *hist* is a useful command to help us understand the big picture of continuous variables, it gives us a histogram of the distribution of a specific variable.
- ▶ Graph often tells us more than summary statistics and reveals some subtle problems. My recommendation is, always make histograms of critical variables in your research.

# Missing Values

- ▶ A missing value is . in numeric format, and it is "" in string format.
- ▶ We can use

```
gen miss_var_name = missing(var_name)
```

to take on the value 1 for true if its argument is missing and the value 0 for false, otherwise. It can be abbreviated `mi()`.

- ▶ Use `tab var_name, m` to explore how many observations have missing `var_name`. Command `tabulate` will omit missing values without option `m`.
- ▶ Regression analysis will omit all observations having a missing value for any one of the variables used in the model. (specification 1 v.s. specification 2)

# Outline

Stata's Interface

Browse the Data

Basic Commands and Syntax

Importing Data from External Resources

Data Types in Stata

Explore the Data

Directory and Path

## Where do you work

- ▶ **Directory:** A directory is a unique type of file that contains only the information needed to access files or other directories. (what is the folder you are working on)
  - Question: if you save some outputs in Stata now, where are you going to find them?

# Where do you work

- ▶ **Directory:** A directory is a unique type of file that contains only the information needed to access files or other directories. (what is the folder you are working on)
  - Question: if you save some outputs in Stata now, where are you going to find them?
- ▶ Showing the current directory:  
pwd (short for “print working directory”)

## Where do you work

- ▶ **Directory:** A directory is a unique type of file that contains only the information needed to access files or other directories. (what is the folder you are working on)
  - Question: if you save some outputs in Stata now, where are you going to find them?
- ▶ Showing the current directory:  
`pwd` (short for “print working directory”)
- ▶ Change the directory and save the data there:  
`cd “/Users/gesun/Desktop/Bootcamp2023/02_Intro_to_Stata/Class_Exercise”`  
`save auto_exercise.dta, replace`

# Where to find your work

**Path:** an address or location to help your computer locate the file you are searching for. (it is like the mail address to send you a postcard)

- ▶ Absolute path: a location in a file system relative to the root directory
  - \* Root directory: the top-level directory in a folder structure.
    - C:\ for Windows
    - / for Mac
- ▶ Relative path: a location in a file system relative to the current directory you are working on



# Where to find your work

## Example

```
cd "/Users/gesun/Desktop/Bootcamp2023/02_Intro_to_Stata/Class_Exercise"  
use auto_exercise.dta,clear
```

# Where to find your work

## Example

```
cd "/Users/gesun/Desktop/Bootcamp2023/02_Intro_to_Stata/Class_Exercise"  
use auto_exercise.dta,clear
```

```
cd "/Users/gesun/Desktop/Bootcamp2023/02_Intro_to_Stata"  
use "./Class_Exercise/auto_exercise.dta", clear
```

- ▶ `“./”` meaning: sub-folder
- ▶ *for Window users, replace `“/”` with `“\”`*

# Where to find your work

- ▶ Another reason why fold structure is important

```
cd "your_project_folder"  
use "./01_raw_data/data_file_name.dta",clear  
do "./02_do_files/clean_data.do"  
save "./03_temp/cleaned_data.dta",replace
```