

# Basic Data Analysis Course on STATA

## Graphical presentation of data in Stata

Md Monowar Hossain

Institute of Applied Statistics and Data Science (IASDS)

University of Dhaka

`mmhossain1@isrt.ac.bd`

`mdmonowarhossain.netlify.app`

July 18, 2026

## Appending datasets

- You can combine different datasets into a single large dataset using the append and merge commands.
- First we create two separate data files by sub-setting based on age of respondents to obtain women aged less and greater than 18 years.

```
use bd_individual.dta, clear
keep if v012 >= 18
save adults.dta
count
```

```
use bd_individual.dta, clear
keep if v012 < 18
save nonadults.dta
count
```

# Appending datasets

- We can now use append to join these two datasets. The command append is used to add extra observations (rows) from another dataset to the existing dataset.

```
append using adults.dta  
count
```

- Note that, you have to have the **same number of variables in both data files** before you append them.

# Appending Datasets

| Dataset A |       |     |
|-----------|-------|-----|
| ID        | Name  | Age |
| 1         | Alice | 25  |
| 2         | Bob   | 30  |
| 3         | Carol | 35  |

| Dataset B |      |     |
|-----------|------|-----|
| ID        | Name | Age |
| 4         | Dave | 28  |
| 5         | Emma | 32  |



***Append***

| Appended Dataset |       |     |
|------------------|-------|-----|
| ID               | Name  | Age |
| 1                | Alice | 25  |
| 2                | Bob   | 30  |
| 3                | Carol | 35  |
| 4                | Dave  | 28  |
| 5                | Emma  | 32  |

# Merging datasets

- Now we create two datasets by sub-setting based on demographic and wealth information of the respondents.

```
use bd_individual.dta, clear
keep caseid v001 v002 v003 v010 v024 v025 v106
sort caseid
save demographic.dta
```

```
use bd_individual.dta, clear
keep caseid v119 v120 v121 v122 v123 v124 v125
sort caseid
save wealth.dta
```

- We have to **sort the two datasets based on a common variable** beforehand in order to merge them.

# Merging datasets

- We can now employ merge to join these two datasets.

```
merge 1:1 caseid using demographic.dta
```

- Note that, the last column produced indicates the case was present in both master and secondary data.
- The open dataset is called the master data.
- For cases that come from only master data will show 1 in the column produced. For cases from only secondary data it will show 2.

# Merging datasets

- Now, we perform 1:m merge as follows:

```
use data1.dta, clear  
merge 1:m household_id using data2.dta
```

- We can also perform m:1 merge using the following codes:

```
use data2.dta, clear  
merge m:1 household_id using data1.dta
```

# 1:1 Merge

| Dataset X |     |        |
|-----------|-----|--------|
| ID        | Age | Salary |
| 1         | 25  | 50000  |
| 2         | 30  | 60000  |

| Dataset Y |             |
|-----------|-------------|
| ID        | City        |
| 1         | New York    |
| 2         | Los Angeles |



| Merged Dataset |     |        |             |
|----------------|-----|--------|-------------|
| ID             | Age | Salary | City        |
| 1              | 25  | 50000  | New York    |
| 2              | 30  | 60000  | Los Angeles |

# 1:M Merge

| Dataset X |      |
|-----------|------|
| ID        | Name |
| 1         | John |

| Dataset Y |                |
|-----------|----------------|
| ID        | Product        |
| 1         | Laptop 1200    |
| 1         | Phone 600      |
| 1         | Headphones 150 |



| Merged Result |      |         |            |        |
|---------------|------|---------|------------|--------|
| ID            | Name | Country | Product    | Amount |
| 1             | John | USA     | Laptop     | 1200   |
| 1             | John | USA     | Phone      | 600    |
| 1             | John | USA     | Headphones | 150    |

## M:1 Merge

| Dataset X |       |         |
|-----------|-------|---------|
| ID        | Name  | Country |
| 1         | Alice | USA     |
| 2         | Bob   | USA     |
| 3         | Carol | USA     |

| Dataset Y |               |
|-----------|---------------|
| Country   | Continent     |
| USA       | North America |



| Merged Result |       |         |       |               |
|---------------|-------|---------|-------|---------------|
| ID            | Name  | Country | Score | Continent     |
| 1             | Alice | USA     | 85    | North America |
| 1             | Bob   | USA     | 90    | North America |
| 1             | Carol | USA     | 78    | North America |

# Available graphs

- For qualitative data
  - Bar chart
  - Pie chart
- For quantitative data
  - Histogram
  - Boxplot
  - Scatter diagram

# Graphical presentation of qualitative data

- Bar chart and pie chart are graphical presentation for summary of qualitative data.
- In bar chart comparisons are made based on original scale of measurement, while in pie chart comparisons are made in terms of transformed measurement scale (in degree)
- Therefore, bar chart is more preferable than pie chart.

# Bar chart

- A bar graph is another tool to represent frequencies or percentages of categories of a nominal or ordinal variable.
- In a bar graph, categories are displayed through rectangles of equal width where heights of the rectangles represent frequency/percentage of the respective categories.

# Types of bar chart

The bar charts can be classified as the following way

- Simple bar chart (involves with single variable)
- Multiple bar chart (involves with multiple variables)
- Component (or Stacked) bar chart (involves with multiple variables)

## Simple bar chart

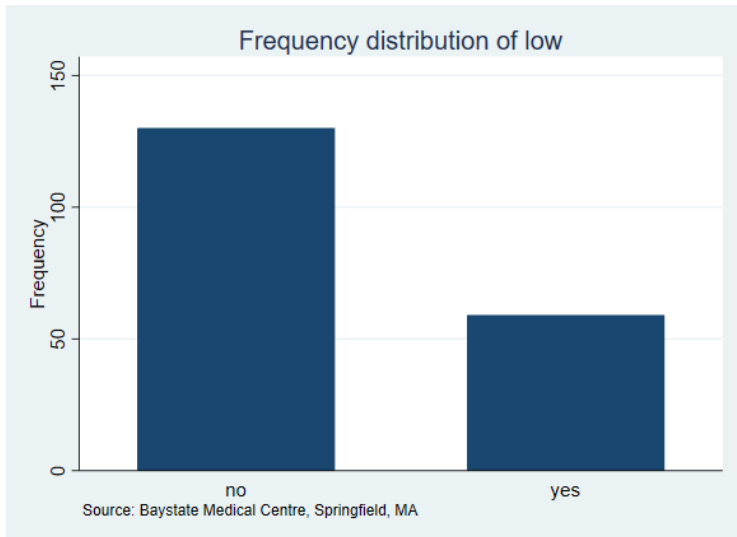
```
use birthwt.dta, clear

graph bar (count), over(low) ///
ytitle("Frequency") ///
title("Frequency distribution of low") ///
note("Source: Baystate Medical Centre, Springfield, MA")

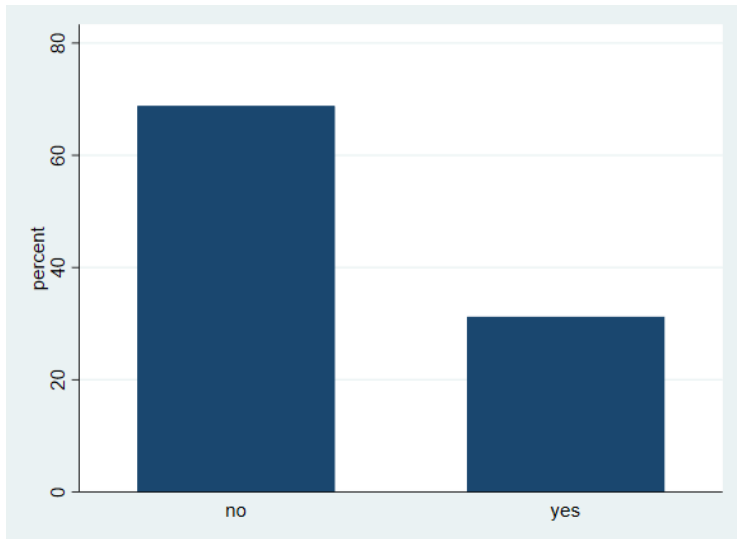
graph bar, over(low) /*gives a percentage bar graph*/
```

How to save these graphs?

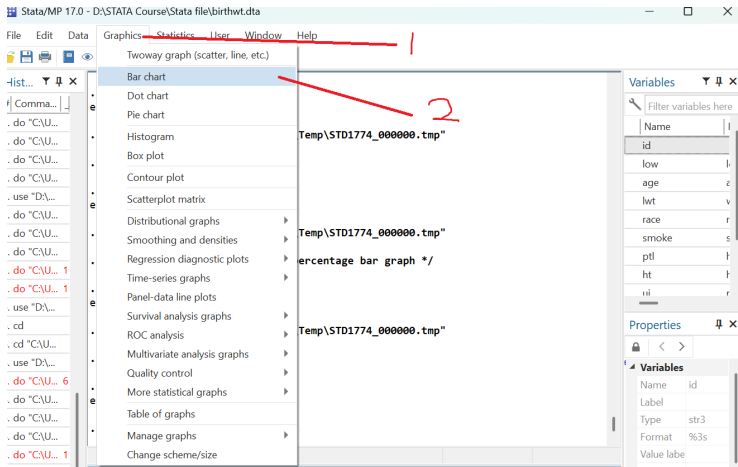
# Simple bar chart



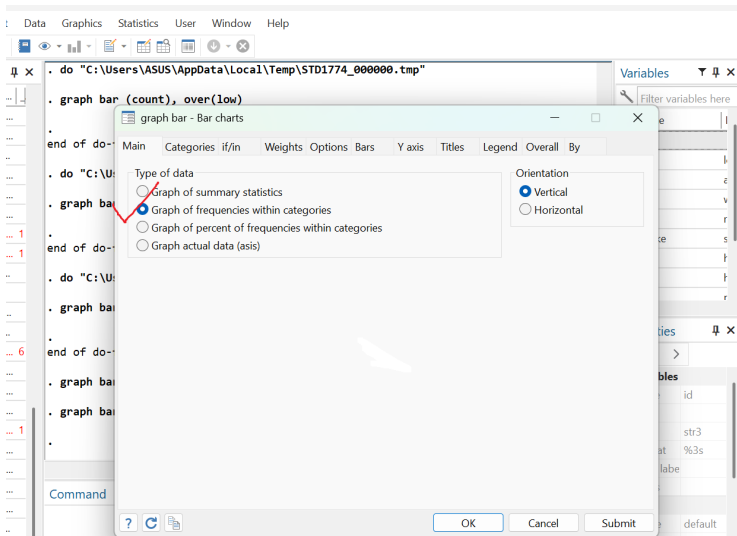
## Simple bar chart



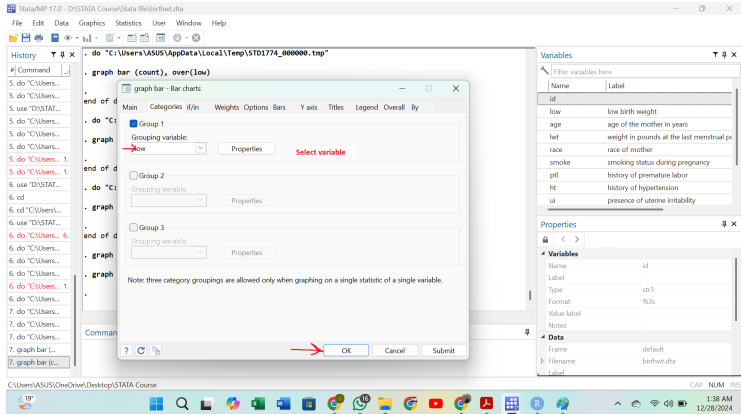
# Simple bar chart



# Simple bar chart



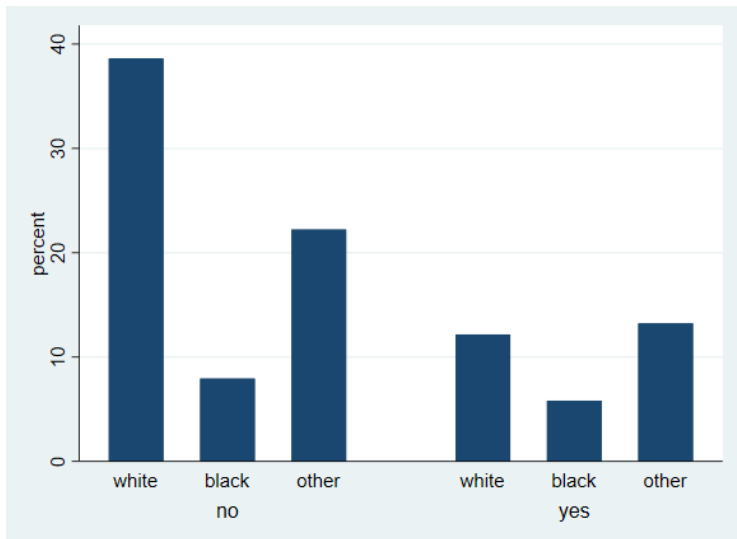
## Simple bar chart



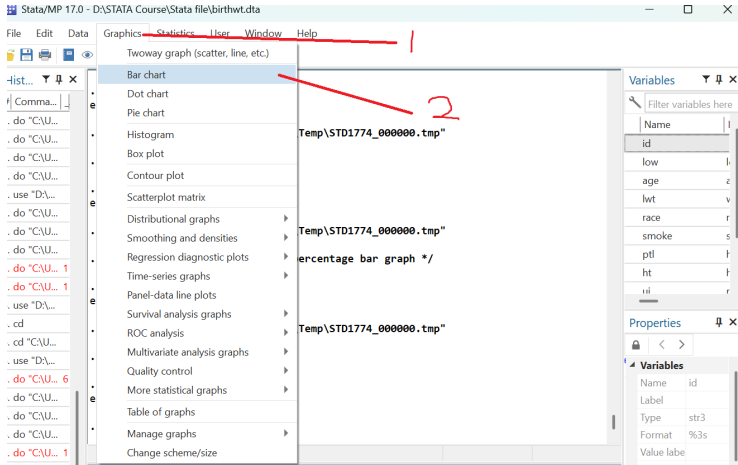
# Multiple bar charts

```
graph bar (count), over(race) over(low)  
graph bar, over(race) over(low)
```

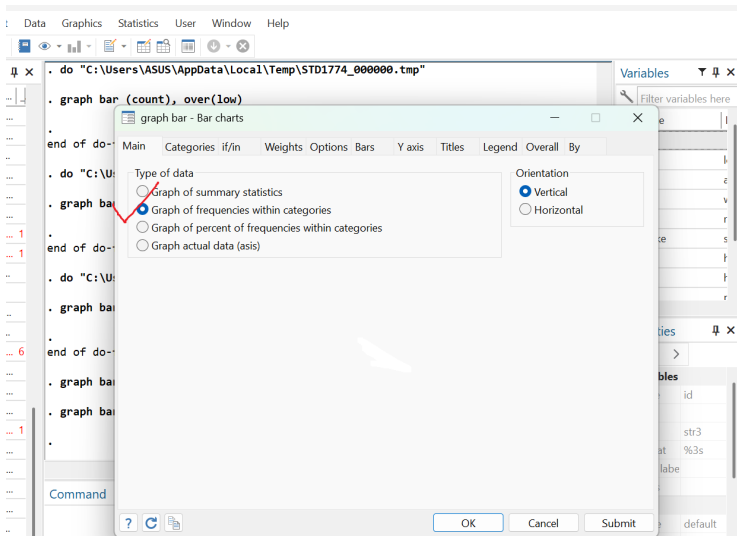
# Multiple bar charts



# Multiple bar chart



# Multiple bar chart



# Multiple bar chart

graph bar - Bar charts

Main Categories if/in Weights Options Bars Y axis Titles Legend Overall By

☒ Group 1

Grouping variable:  
race Properties Select 1st Variable

☒ Group 2

Grouping variable:  
low Properties Select 2nd Variable

☐ Group 3

Grouping variable:  
Properties

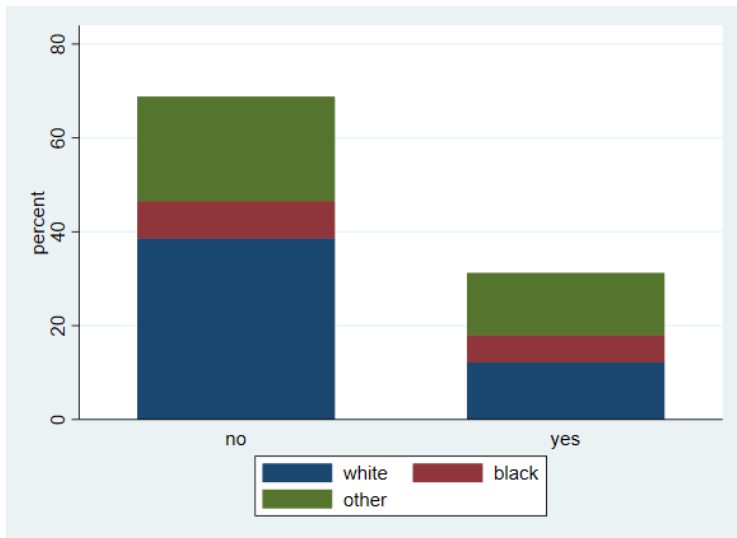
Note: three category groupings are allowed only when graphing on a single statistic of a single variable.

Then Ok OK Cancel Submit

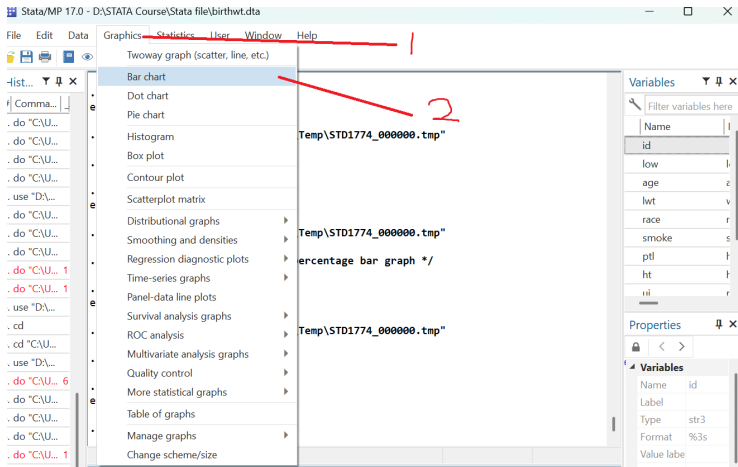
## stacked bar charts

```
graph bar, over(race) over(low) stack asyvars  
/*treat first over() group as yvars*/
```

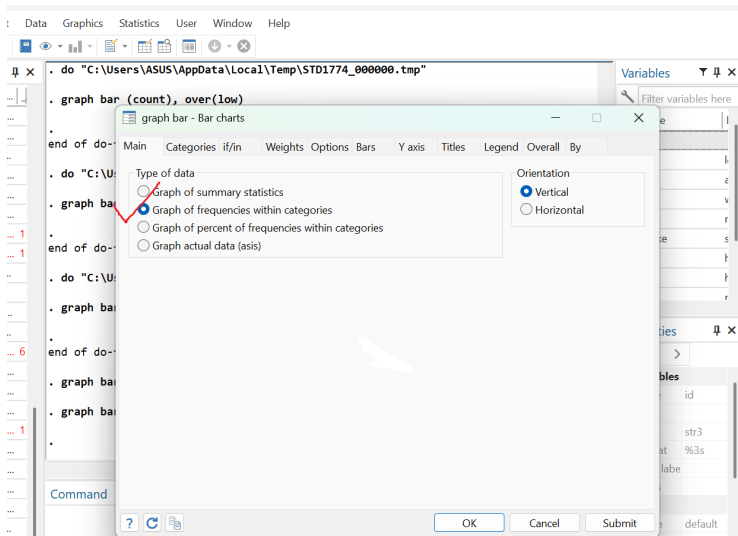
## stacked bar charts



# stacked bar chart



stacked bar chart



# stacked bar chart

graph bar - Bar charts

Main Categories if/in Weights Options Bars Y axis Titles Legend Overall By

☒ Group 1

Grouping variable:  
race ▼ Properties Select 1st Variable

☒ Group 2

Grouping variable:  
low ▼ Properties Select 2nd Variable

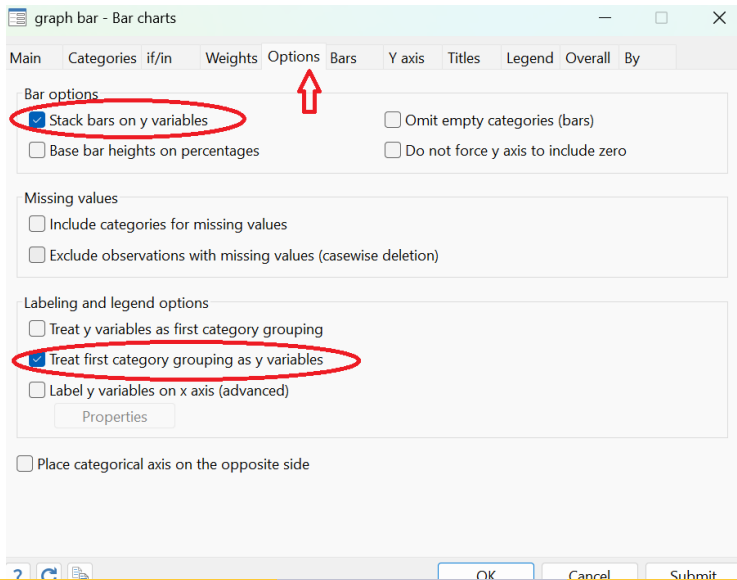
☐ Group 3

Grouping variable:  
▼ Properties

Note: three category groupings are allowed only when graphing on a single statistic of a single variable.

Then Ok OK Cancel Submit

# stacked bar chart



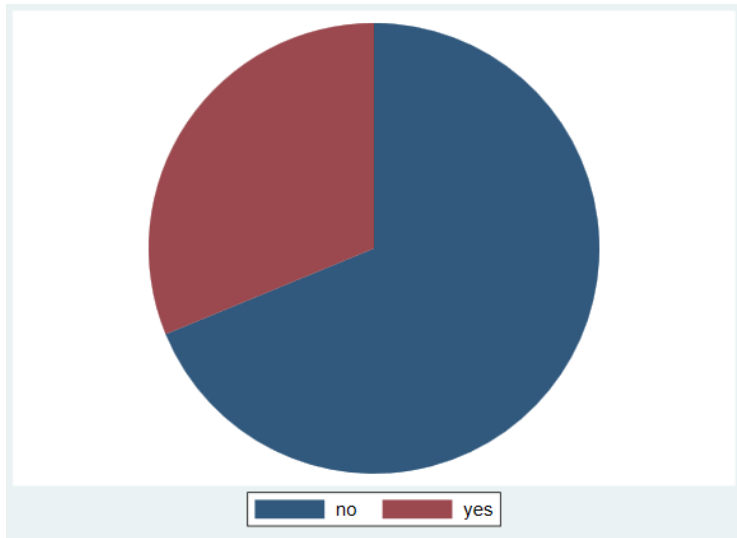
# Pie chart

- The pie diagram is intended to compare the distinct components which together constitute a whole.
- The whole is presented by a circle of arbitrary radius and segments of the circle represents the component parts.
- The whole corresponds to the  $360^\circ$  which is the total number of degree in a circle. This  $360^\circ$  is to be divided proportionately among the components for drawing a pie chart.

# Pie chart

graph pie, over(low)

# Pie chart



# Pie chart

The screenshot shows the Stata 17.0 interface. The 'Graphics' menu is open, and 'Pie chart' is selected. The command window shows the command: `graph pie, over(low)`. The 'Variables' panel on the right lists the following variables:

| Name  | Label                                         |
|-------|-----------------------------------------------|
| low   | low birth weight                              |
| age   | age of the mother in years                    |
| lwt   | weight in pounds at the last menstrual period |
| race  | race of mother                                |
| smoke | smoking status during pregnancy               |
| pft   | history of premature labor                    |
| ht    | history of hypertension                       |
| ui    | presence of uterine incontinuity              |

The 'Properties' panel on the right shows the following variables:

| Variable | Label                                         |
|----------|-----------------------------------------------|
| low      | low birth weight                              |
| age      | age of the mother in years                    |
| lwt      | weight in pounds at the last menstrual period |
| race     | race of mother                                |
| smoke    | smoking status during pregnancy               |
| pft      | history of premature labor                    |
| ht       | history of hypertension                       |
| ui       | presence of uterine incontinuity              |

# Pie chart

graph pie - Pie charts

Main

if/in

Weights

Options

Slices

Titles

Legend

Overall

By

Pie chart

☒ Graph by categories ✓

☐ Graph by variables

Category variable:

low

Variable: (optional)

Select Categorical Variable

# THANK YOU