Introduction to Stata:
Computation & Regression Analysis

Kory Kantenga

University of Pennsylvania

Today

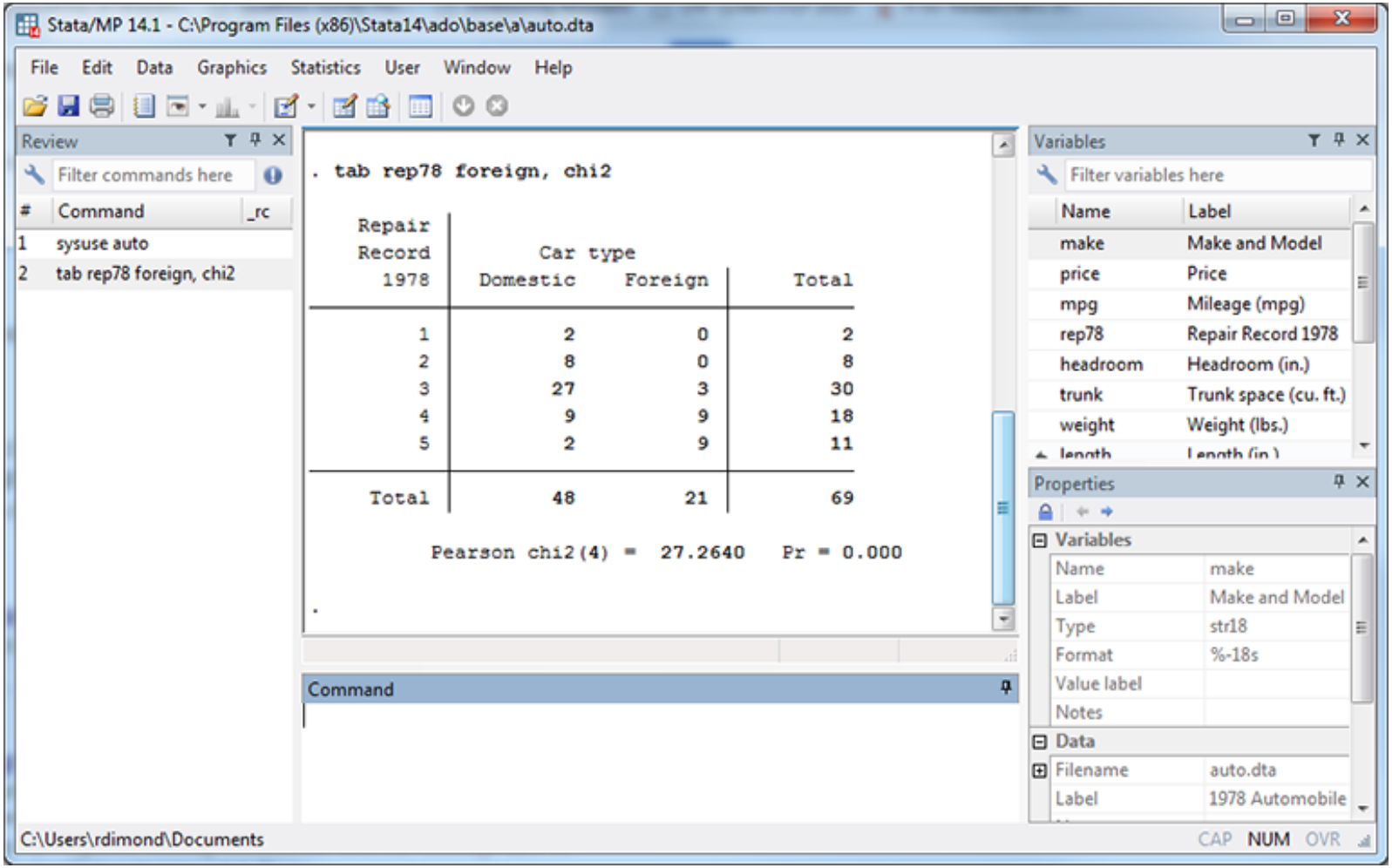
1. What is Stata?
2. Computing in Stata
3. Graphics in Stata
4. More in Stata

Stata is a both high-level language for programming and a user interface

Stata Windows

- Do-file Editor: Commands to execute, make comments
- Data Editor: Browse and edit data
- Command Line: Type in command and hit `Return`
- Variables: Variable names and types
- Graphs: Plots (to export)
- Review: List of past commands entered
- Results

Stata Interface



Why Stata?

- Designed for statistics
- User Friendly
- Standard tools + developer tools

Getting Started

- Stata is not freeware. You have to purchase it, but it is available on student discount at most universities and on the computing network.
- Download data from `https://github.com/korykantenga/Stata_R_Workshop`

Who am I? Where am I?

It is always a good idea to clear the environment before starting

```
clear all
```

Stata needs to know where to look for things so we need to set the directory

```
pwd
```

```
cd "FolderPathGoesHere"
```

What are commands? Where to find them?

- Commands tell Stata what to do
- Stata comes with a bunch of commands, but you can write your own or install someone else's using `ssc install CommandName`

Notable Examples:

- `su`: summarize variables and locally store their statistics
- `reg`: run a simple linear regression
- `xi`: create {0,1} variables from categorical variables

Type `help CommandName`

What is an Do-File? Where to find them?

- An Do-file is a text file (with the file extension `.do`) containing your Stata Code
- Stata Code is a series of comments, commands and functions that execute the tasks you require
- We can add comments using the (*) symbol or (//)

Where to find them?

You make your own :-)

Debugging

More than likely you will run into errors. These errors are usually bugs.

Bugs are problems in YOUR CODE.
Almost all errors will be syntax errors.

Syntax errors are the coding equivalent of misspellings.

The 5 Stages of Debugging

At some point in each of our lives, we must face errors in our code. Debugging is a natural healing process to help us through these times. It is important to recognize these common stages and realize that debugging will eventually come to an end.

Denial

This stage is often characterized by such phrases as "What? That's impossible," or "I know this is right." A strong sign of denial is recompiling without changing any code, "just in case."

Bargaining/Self-Blame

Several programming errors are uncovered and the programmer feels stupid and guilty for having made them. Bargaining is common: "If I fix this, will you please compile?" Also, "I only have 14 errors to go!"

Anger

Cryptic error messages send the programmer into a rage. This stage is accompanied by an hours-long and profanity-filled diatribe about the limitations of the language directed at whomever will listen.

Depression

Following the outburst, the programmer becomes aware that hours have gone by unproductively and there is still no solution in sight. The programmer becomes listless. Posture often deteriorates.

Acceptance

The programmer finally accepts the situation, declares the bug a "feature", and goes to play some Quake.

Getting Help

1. Use `help x` to see the help file for a command with name `x`
2. Use `search x` to search for a command with name `x`
3. Enter `help` into the Command Line and hit `Return` key
4. Try a tutorial series
5. GIFY (G-It-For-Yourself)

Today

1. What is Stata?
2. Computing in Stata
3. Graphics in Stata
4. More in Stata

Simple Computations

- Use = to assign values (e.g. `x=3.141593`)
- Standard arithmetic operators (e.g. `+,-,*,/`)
- Standard operators like
  - natural logarithm: `ln(x)`
  - e^x: `exp(x)`
  - a^x: `a^x`

Creating New Variables (gen)

Generating uniform random number between (0,2)

- set obs 20 // create 20 observations
- gen myVariable = 2*runiform()

Importing Data (insheet, use)

Import data from csv and save as .dta

- cd "myPath" // set directory where data is
- import delimited using "myData.csv"
- save "myData.dta"

Show all variables and memory

- memory // shows system memory
- describe // shows variable and memory usage

Modify (replace) and Edit (drop, keep)

Replace small observations and drop them

- replace myVariable = -1 if myVariable<=0.001
- drop if myVariable==-1 // logicals require double equals
- keep if myVariable<3 // drops observations >3

Summarize Variables (su)

pop5_17	long	%12.0gc	Pop, 5 to 17 years
pop18p	long	%12.0gc	Pop, 18 and older
pop65p	long	%12.0gc	Pop, 65 and older
popurban	long	%12.0gc	Urban population
medage	float	%9.2f	Median age
death	long	%12.0gc	Number of deaths
marriage	long	%12.0gc	Number of marriages
divorce	long	%12.0gc	Number of divorces
drate	float	%9.0g	Death Rate
age	float	%9.0g	

Sorted by:

```
. generate urbanized=popurban/pop

. summarize urbanized
```

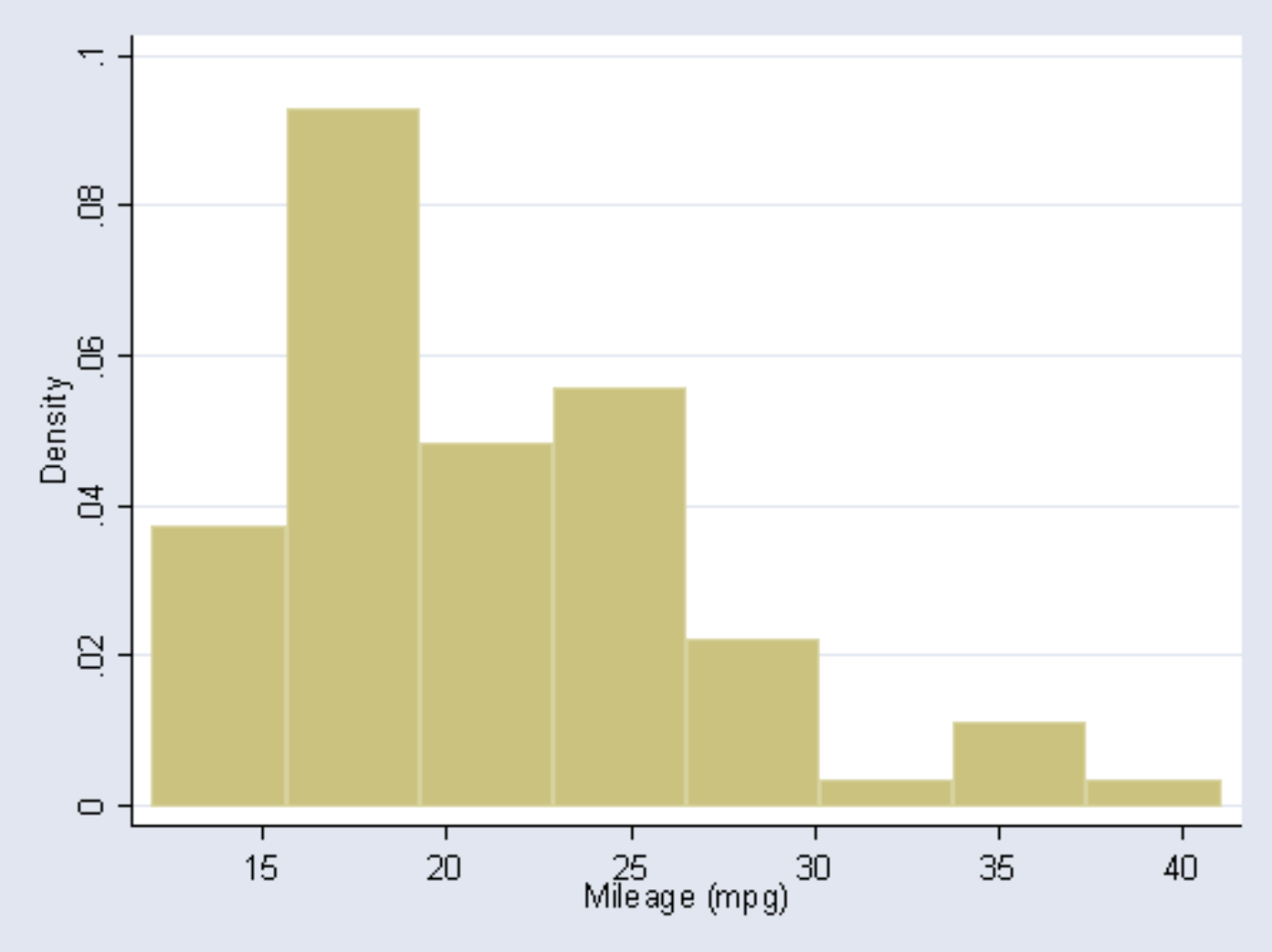
Variable	Obs	Mean	Std. Dev.	Min	Max
urbanized	50	.6694913	.1440956	.3377319	.9129498

Today

1. What is Stata?
2. Computing in Stata
3. Graphics in Stata
4. More in Stata

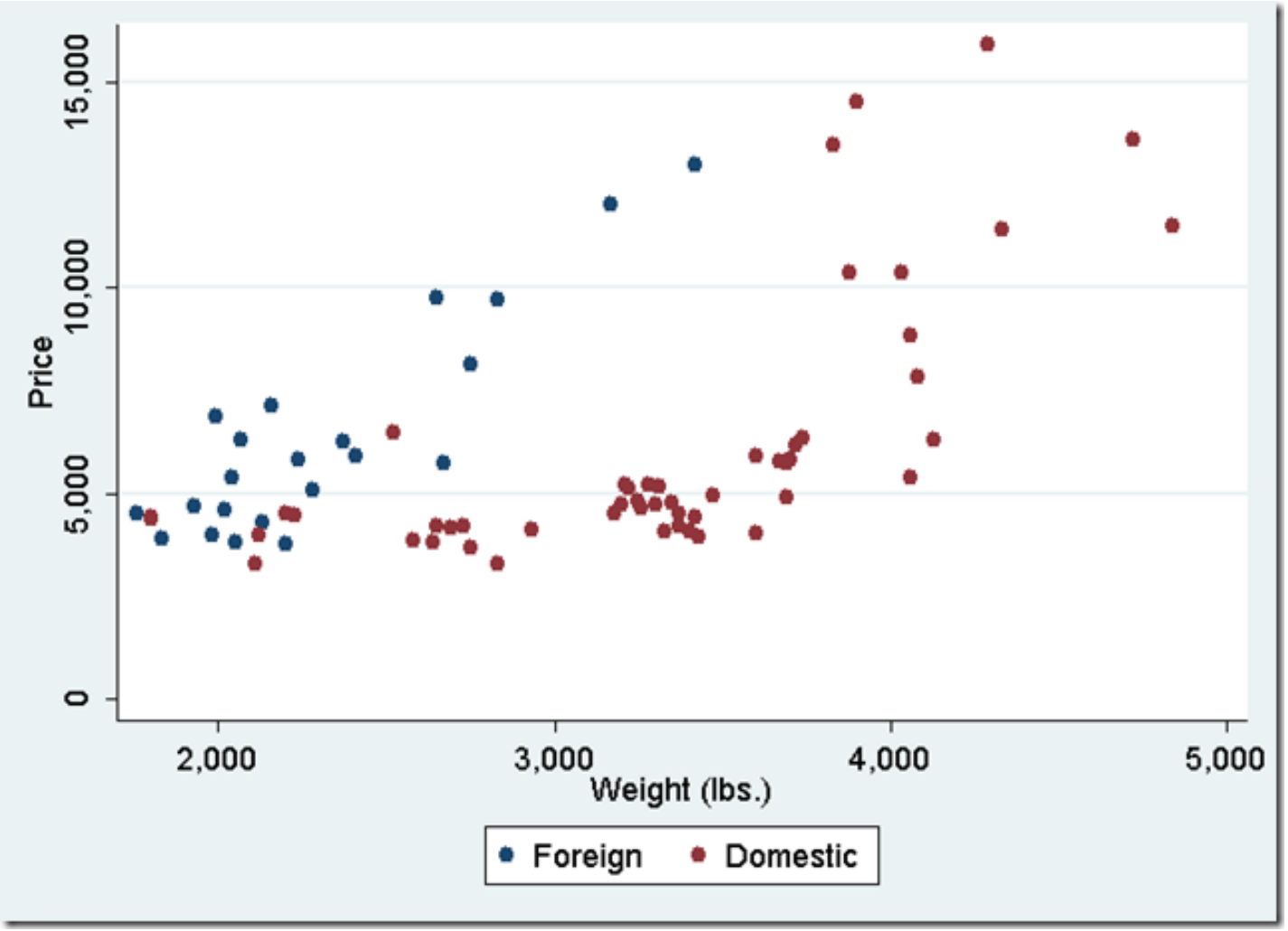
Histogram (hist)

```
hist mpg
```



Scatter Plot (twoway scatter)

twoway scatter price weight, legend(label(1 "Foreign") label(2 "Domestic"))



Simple Linear Regression (reg)

```
. regress cholesterol time_tv
```

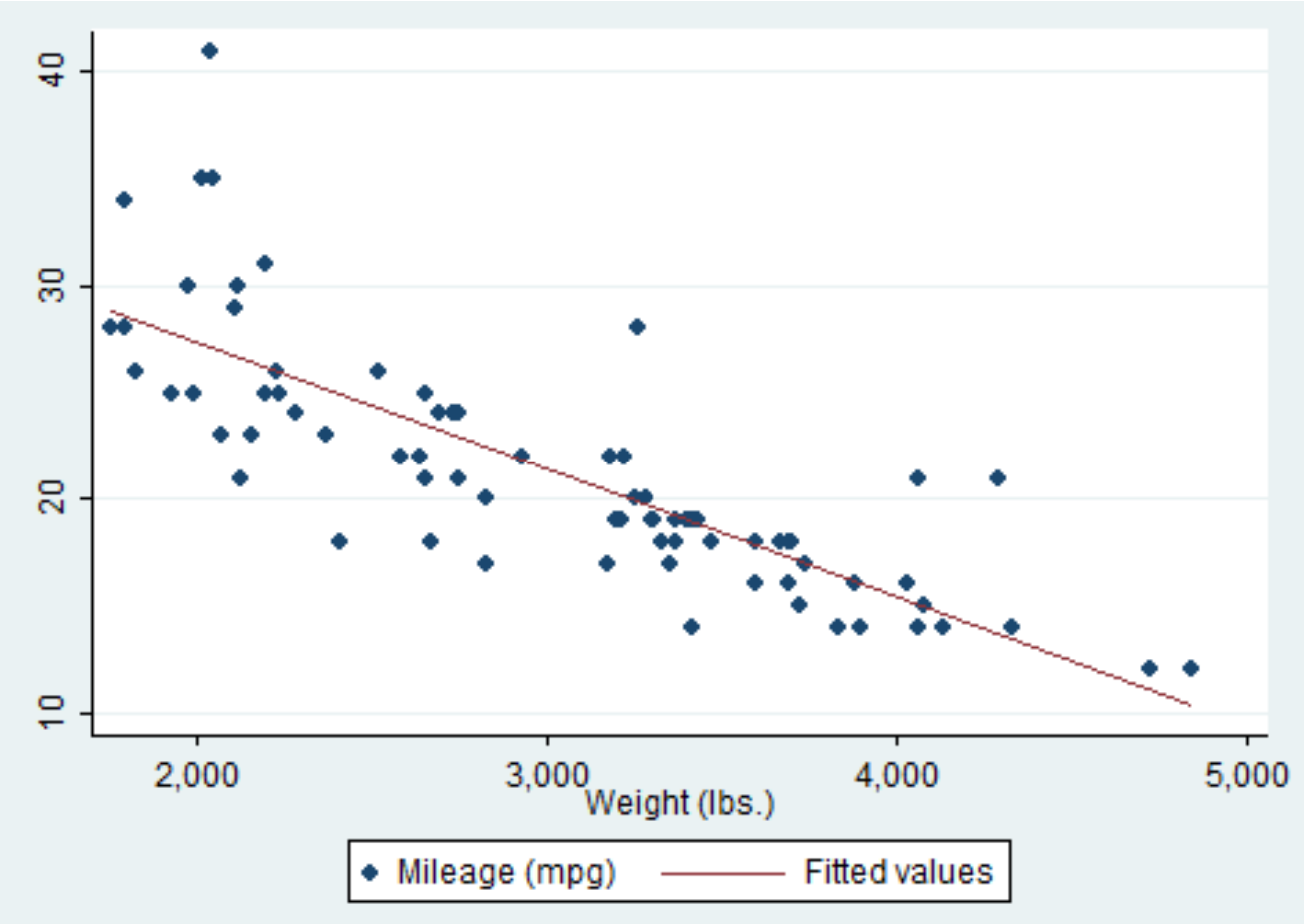
Source	SS	df	MS
Model	5.04902329	1	5.04902329
Residual	28.3220135	98	.289000137
Total	33.3710367	99	.337081179

Number of obs	= 100
F(1, 98)	= 17.47
Prob > F	= 0.0001
R-squared	= 0.1513
Adj R-squared	= 0.1426
Root MSE	= .53759

cholesterol	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
time_tv	.0440691	.0105434	4.18	0.000	.0231461	.0649921
_cons	-2.134777	1.813099	-1.18	0.242	-5.732812	1.463259

Fitted Line (twoway lfit)

twoway scatter mpg weight || lfit mpg weight



Today

1. What is Stata?
2. Computing in Stata
3. Graphics in Stata
4. More in Stata

Advanced Topics (beyond our scope)

- Functions
- Loops
- Low-level Coding (MATA)
- Astral Projection and Telekinesis

For all of these and more, try GIFY.

Topics in Statistics

1. Descriptive Statistics (e.g. mean, median, mode)
2. Probability Theory (e.g. Bayes Rule)
3. Random Variables (Discrete, Continuous)
4. Probability Distributions (e.g. CDF, PDF, PMF)
5. Expectations (e.g. $\mathbb{E}[aX] = a \cdot \mathbb{E}[X]$)
6. Point and Interval Estimation (e.g. confidence intervals)
7. Properties (e.g. unbiasedness, efficiency, consistency)
8. Sampling Distributions (e.g. $\bar{X} \sim N(\mu, \sigma^2/N)$)
9. Hypothesis Testing (e.g. $H_0 : \mu = 0, \ H_1 : \mu \neq 0$)
10. Simple Linear Regression (e.g. $y_i = \alpha + \beta x_i + \varepsilon_i$)

Topics in Regression Analysis

1. Linear Regression Models
   - Least Squares Estimation
   - Inference
2. Panel Data Models
   - First Difference
   - Fixed Effects
3. Simultaneous-Equations Models
   - Instrumental Variables
4. Discrete Choice Models
   - Maximum Likelihood Estimation
5. Time Series Models

Help Sites

- `https://www.statalist.org/forums/help`