ECON3500 Lab 5: Merging and hypothesis tests

Materials

- `acs2024_4pct.dta`
- Do-file template `econ3500_lab_template.do`
- BLS county unemployment data `laucnty24.xlsx` *(or download from BLS)*

Objectives

Today we're going to work with `acs2024_4pct.dta`, which contains information from the 2024 American Community Survey. *Note that this is a different version from what we have been using! It has a few more variables and also a larger sample.*

We're going to merge county-level unemployment rates from the Bureau of Labor Statistics.

By the end of this lab, you should be able to complete the following tasks in Stata:

- Import data from Excel
- Merge data sets
- Test hypotheses for linear combinations of coefficients
- Test the general significance of a regression

Data context

Each row in `acs2024_4pct.dta` is an individual from the 2024 ACS microdata. The file includes demographics, education, labor-force status, earnings, and geographic identifiers at the state and county level. The BLS county unemployment file (`laucnty24.xlsx`) contains 2024 annual average labor force statistics for every U.S. county.

We will merge the two datasets by county, matching on state and county FIPS codes.

Variables we'll use

ACS data (`acs2024_4pct.dta`)

variable	meaning	notes
`inctot`	total personal income	9999999 = N/A; replace before analysis
`educ`	educational attainment	numeric categories; check labels with `tab educ, nolabel`
`labforce`	labor force status	2 = in labor force (check with `tab labforce, nolabel`)
`age`	age	
`statefip`	state FIPS code	used for merging
`countyfip`	county FIPS code	0 = not identified; used for merging

BLS data (`laucnty24.xlsx`)

column	meaning	notes
State FIPS Code	2-digit state code	imported as string; needs `destring`
County FIPS Code	3-digit county code	imported as string; needs `destring`
County Name/State Abbreviation	county name	
Labor Force	total county labor force	
Employed	county employed count	
Unemployed	county unemployed count	
Unemployment Rate (%)	county unemployment rate	

Key commands

command	description
Importing data	
`import excel using "file.xlsx", firstrow clear`	Import an Excel file. `firstrow` uses row 1 as variable names. `clear` erases existing data.
`import excel using "file.xlsx", cellrange(A2) firstrow clear`	Same, but start reading from cell A2 (useful when row 1 is a title, not data).
Identifying duplicates	
`duplicates list var1 var2`	List any observations that are duplicates on the listed variables.
`duplicates tag var1 var2, gen(d1)`	Generate a new variable, `d1`, that indicates which observations are duplicates for `var1` and `var2`.
Merging datasets	
`merge 1:1 var1 var2 using file2`	One-to-one merge on `var1` and `var2`. No duplicates allowed in either dataset.
`merge m:1 var1 var2 using file2`	Many-to-one merge on `var1` and `var2`. Duplicates OK in master data (like merging county data into individual data) but not in using data.
Converting between string and numeric variables	
`destring var1, gen(newvar)`	Convert a string variable to numeric, saving as `newvar`.
`destring var1, replace`	Convert a string variable to numeric, replacing the original.
`tostring var2, gen(string_var)`	Convert a numeric variable to string, saving as `string_var`.
Statistical tests	
`test var1 = var2`	Run after a regression. Tests whether the coefficient on `var1` equals the coefficient on `var2`.
`testparm var1 var2 ...`	Run after a regression. Tests whether all listed variables are jointly equal to zero.

A note on temporary files (optional)

This exercise works by having two data sets stored on your hard drive, then running a `merge` command to unite them. You might notice that the workflow feels clunky and generates extra files — open a data set, save it, open another data set, then merge in the first data set.

You can use temporary files to speed things up! Basically, you can save files in your local memory, and call those files the same way we called local variables. Everything has to be run in the do-file for this to work.

A short example (you can paste this in a do-file and run it, as it uses built-in Stata files):

```
tempfile tempauto       // Declare tempfile (needs to run before you try to save)

webuse autosize, clear

save `tempauto', replace    // save to temp file

webuse autoexpense, clear

merge 1:1 make using `tempauto'   // call tempfile

tab _merge    // check out merge

list
```

Workflow overview

1. Import the BLS county unemployment data from Excel.
2. Clean variables and save as a Stata data file.
3. Open the ACS data and restrict the sample.
4. Merge in county-level unemployment by state and county FIPS codes.
5. Create education indicators and run regressions.
6. Conduct hypothesis tests.

Lab 5 Worksheet

What do I submit?

- Your written up answers to the exercise questions. This can be typed or written out then scanned (or photographed), in any reasonable format. *Note: Question 21 is optional.*
- The do-file you've created that runs this analysis
- A log file that contains the results from this exercise.

Exercises

Part 1: Import and prepare unemployment data

1. Visit https://www.bls.gov/lau/tables.htm to access 2024 annual **county-level** unemployment rates. Download the appropriate table as an Excel file.[1]
    a. Open the file in Excel or another spreadsheet program. Notice that the first row contains a title and the actual column headers start in the second row.
    b. You do not need to edit the file — we'll handle everything in Stata.
2. Open Stata and start a new do-file using the template. Update the file paths and add code to start (and end) a log.
3. Import your unemployment Excel file into Stata. Because the first row is a title (not column headers), use the `cellrange` option to start reading from row 2:

[1] If you have trouble accessing the BLS website, you can use the file provided in the lab materials above.

```
import excel using "laucnty24.xlsx", cellrange(A2) firstrow clear
```

Run `describe` to see the variable names Stata assigned. How many observations (counties) are there?

4. The FIPS code variables were imported as **strings** (text), not numbers. Convert them to numeric variables so they match the ACS data:

   ```
   destring StateFIPSCode, gen(statefip)
   destring CountyFIPSCode, gen(countyfip)
   ```

   (If Stata named your variables differently, check with `describe` and adjust accordingly.)

5. Check for duplicates on `statefip` and `countyfip`. Are there any? *(There shouldn't be — each county should appear exactly once.)*

6. Save your unemployment data as a Stata file:

   ```
   save "unemp_2024.dta", replace
   ```

Part 2: Merge with ACS data

7. Open `acs2024_4pct.dta` and restrict the sample to adults (age 18+).

8. Before merging, take a look at the county identifier in the ACS data. Tabulate `countyfip`. What do you notice about the value 0?[2]

9. Now, merge your unemployment data into the ACS by county:

   ```
   merge m:1 statefip countyfip using "unemp_2024.dta"
   ```

   a. Why do we use `m:1` (many-to-one) instead of `1:1`?

   b. Tabulate the `_merge` variable. What share of observations successfully merged?[3]

10. Drop any unmatched observations (you can use `drop if`) and drop the `_merge` variable. What is the average unemployment rate for the sample — why would this be different than taking an average of county unemployment rates from your Excel file?

Part 3: Education variables and regression

11. Why can't we use `educ` directly as a linear variable in a regression?

12. Generate three dummy variables. These three variables should be mutually exclusive, and they should not be missing for any observations.

    - `lesshs`, a variable equal to one if a person completed *less than* a high school diploma
    - `hsgrad`, a variable equal to one if a person completed at least a high school diploma but less than a Bachelor's degree
    - `colgrad`, a variable equal to one if a person completed a Bachelor's degree or higher

    Note: Education is coded with **labels,** which means that it is numeric data with a description of what each number means on top. These show up as blue in the Stata browser. To see the underlying codes: `tab educ, nolabel`.

13. What is the mean of `lesshs`, `hsgrad`, and `colgrad`?

[2] In IPUMS data, `countyfip = 0` means the county is **not identified** — the Census Bureau withholds county identifiers for small counties to protect confidentiality. These observations cannot be matched to BLS data.

[3] Expect roughly 40–60% of observations to match. The main reason for non-matches is that many ACS respondents have `countyfip = 0` (county not identified).

14. Before running a regression, check `inctot` (total personal income) for N/A codes. Replace any N/A values as missing.[4] Then estimate a regression of total personal income on education, using the binary variables you just created. Omit `lesshs`. Use robust standard errors.

Part 4: Hypothesis tests

15. Set up a hypothesis test for whether both `hsgrad` and `colgrad` are jointly significant. Report the null hypothesis, alternative hypothesis, test statistic, and conclusion.
16. Set up a hypothesis test for whether the returns to being a high-school graduate are the same as the returns to being a college graduate. Report the null hypothesis, alternative hypothesis, test statistic, and conclusion.
17. Is this regression significant overall? Explain how you know.

Part 5: Adding unemployment

18. Now add county-level unemployment rate to the previous equation. What is the interpretation of the coefficient on unemployment? Is it statistically significant?
19. Estimate the same equation by regressing total personal income on the education binary variables and county-level unemployment, restricting to those who are currently in the labor force. How does this change the coefficient on unemployment?
20. Identify three *state* or *county-level* variables that are likely to cause omitted variable bias if you want to know whether unemployment affects individual income.
21. *(Optional)* For *one* of the variables you listed above, find the data online, import into Stata, and merge it in. Regress total personal income on the education binary variables, county-level unemployment, and the new variable you found. Restrict your sample to those who are currently in the labor force. How does the inclusion of your new variable affect the coefficient on unemployment?

[4] Use `summarize inctot` to check for suspicious values. In IPUMS data, 9999999 typically means N/A.