```
. do who_wrote_this.do
```

Who Wrote This Do-File?

AI, Stata, and the Future of Applied Analysis and Teaching

Leslie-Anne Keown · University of Regina

Canadian Stata User Group Meeting · November 6, 2026

A career in do-files

Statistics Canada

National surveys, large files

Research Data Centre

Confidential microdata, vetted output

Correctional Service of Canada

Over a decade of applied corrections research

University teaching

Methods and statistics at four universities over 20 years

A lot of do-files. Most of them typed badly.

Where the hours used to go

BEFORE

- Cleaning and recoding, variable by variable
- Hunting for syntax in help files and Statalist
- An afternoon lost to one broken loop

NOW

- Describe the task, then review the code
- More time choosing and specifying models
- More time on what the results mean

The bottleneck moves from syntax to judgment.

From plain English to working Stata

THE ASK

“Recode every _q item from a 1–5 scale to agree / not agree, label the new variables, and tabulate each by province.”

THE DRAFT

```
* Recode Likert items to binary
foreach v of varlist *_q {
    recode `v' (1/3 = 0 "Not agree") ///
           (4/5 = 1 "Agree"), gen(`v'_bin)
    label variable `v'_bin "`: variable label `v'' (agree)"
}
foreach v of varlist *_q_bin {
    tab `v' province, col
}
```

Seconds to draft. Still worth a line-by-line read.

Macros for the rest of us

Students, new analysts, and the colleague who always meant to learn loops.

Loops and locals

`foreach · forvalues · local`

Do it once, correctly, for every variable or wave.

Programs

`program define · syntax`

Reusable routines instead of copy-and-paste.

Reproducible output

`collect · etable · putdocx`

Tables that build themselves, every rerun.

The features were always there. The on-ramp is new.

BUT…

If you can’t read the code, you can’t vouch for the results.

Syntax and data cleaning matter more now, not less: they’re how you check the work.

How AI-written Stata goes wrong

Invented options

Plausible syntax that doesn't exist, or belongs to a different command.

Missing values count as "big"

```
gen high = (income > 50000)
```

Missing income is coded as high: Stata treats . as larger than any number.

The wrong merge

```
merge m:m id using visits
```

Runs without complaint. Almost never what you want.

Quiet analytic choices

Dropped cases, missing weights, a different reference category. Nobody asked; nobody noticed.

Trust, but assert

- Check counts before and after every step
- Tabulate with missing values showing
- Make IDs and merges prove themselves
- Ask the AI to explain its code, then check the explanation

VERIFY.DO

```
* Make the data prove it
count
tab province, missing
isid person_id
merge 1:1 person_id using health, assert(match)

* Fix the missing-value trap, then check it
gen high = (income > 50000) if !missing(income)
assert missing(high) if missing(income)
```

What changes in the classroom

AI writes the first draft. Students are the reviewers.

First, specify

Tell the AI each variable's **level of measurement** and the **analysis you want**. Otherwise the syntax may run, but the results can be analytically unusable.

"satisfaction is ordinal (1–5), region is nominal, income is continuous: fit an ordered logit of satisfaction on income and region."

Read

Students annotate an AI-written do-file, line by line, in their own words.

Break

Find the planted bug: the missing-value trap, the bad merge, the dropped cases.

Defend

Explain every analytic choice, in writing or out loud.

A CONFESSION

The real breakthrough

WHAT I TYPED

“recod teh liekrt vars to agre/not agre and tabluate by provnce plz”

WHAT I GOT

```
foreach v of varlist *_q {
    recode `v' (1/3=0) (4/5=1), gen(`v'_bin)
    tab `v'_bin province, col
}
```

I’m still a lousy typist. Stata never has to know.

Four things to take home

01	Spend the hours on analysis and interpretation, not syntax.
02	Macros, loops and programs are now within reach for newer users.
03	Syntax and data-cleaning skills still matter: they’re how you verify.
04	Typing skills are now optional.

```
. help questions
```