

Introducing the Stata linter

A tool to produce clear and transparent
Stata code

Stata Conference 2023

Presented by Luis Eduardo San Martin

DIME Analytics – World Bank

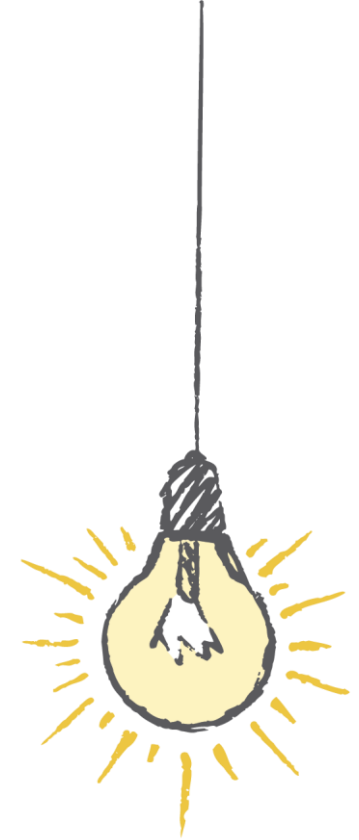


The problem



*“When your code works,
you are only half-done”*

Ancient proverb



The problem

- Good code: code that is correct and *comprehensible*.
- What is *comprehensible*, though?

The problem

- Style guides solve the problem, partially
- But Stata doesn't have widely adopted Style guides nor tools or commands to apply style guides to code

Stata linter

- The Stata linter is an automated detector that reads a do-file and flags bad coding practices

1

```
master_dofile.do* x
29 * ****
30 * PART 0: Standardize settings and install packages
31 * ****
32
33
34 *Stata Settings
35
36 clear all
37 set more off, perm
38 cap log close
39 estimates clear
40 window manage close graph _all
41 cap graph drop _all
42
43 version 16
44 sysdir set PLUS "C:\WBG\Rep-checks\FY23\FGI\fgi_replication_package_20220923\ado"
45
46
47
48 * Install all packages that this project requires:
49 local user_commands rittest lassopack winsor estout nmatch tsfill leebounds coefplot
50 distplot winsor pdslasso egenmore //Fill this list
51
52 foreach command of local user_commands {
53     cap which `command'
54     if _rc == 111 {
55         ssc install `command'
56     }
57 }
```

2

Command

```
lint master_dofile.do
```

3

```
. lint master_dofile.do
```

```
Linting file: master_dofile.do
```

```
-----
```

Bad practice	Occurrences
Hard tabs used instead of soft tabs:	Yes
One-letter local name in for-loop:	0
Non-standard indentation in { } code block:	2
No indentation on line following ///:	0
Use of . where missing() is appropriate:	0
Missing whitespaces around operators:	4
Implicit logic in if-condition:	0
Delimiter changed:	0
Working directory changed:	1
Lines too long:	13
Global macro reference without { }:	1
Potential omission of missing values in expression:	0
Backslash detected in potential file path:	8
Tilde (~) used instead of bang (!) in expression:	0

```
-----
```

```
For more information about coding guidelines visit the Stata linter wiki.
```

```
.
```


How to use the Stata linter



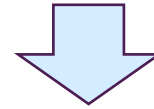
Requirements

- Stata 16 or higher
- Python (the command uses it in the backend)
- Pandas (Python library)
- ❖ Links to Python-Stata integration instructions [here](#)

Installation

Command

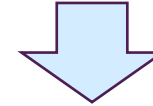
```
ssc install stata_linter|
```



```
. ssc install stata_linter  
checking stata_linter consistency and verifying not already installed...  
installing into ado\...  
installation complete.
```

Command

```
lint "test/bad.do"
```



```
. lint "test/bad.do"
```

```
Linting file: test/bad.do
```

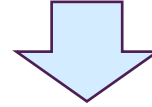
Bad practice	Occurrences
Hard tabs used instead of soft tabs:	Yes
One-letter local name in for-loop:	3
Non-standard indentation in { } code block:	7
No indentation on line following ///:	1
Use of . where missing() is appropriate:	5
Missing whitespaces around operators:	0
Implicit logic in if-condition:	1
Delimiter changed:	1
Working directory changed:	0
Lines too long:	5
Global macro reference without { }:	0
Potential omission of missing values in expression:	1
Backslash detected in potential file path:	0
Tilde (~) used instead of bang (!) in expression:	5

For more information about coding guidelines visit the [Stata linter wiki](#).

Basic use,
detection
feature

Command

```
lint "test/bad.do", verbose
```



```
(line 48): After declaring for loop statement or if-else statement, add indentation (4 whitespaces).
(line 48): Use "!missing(var)" instead of "var < ." or "var != ." or "var ~=" ."
(line 53): Use "!missing(var)" instead of "var < ." or "var != ." or "var ~=" ."
(line 60): Use "!missing(var)" instead of "var < ." or "var != ." or "var ~=" ."
(line 69): In for loops, index names should describe what the code is looping over. Do not use an abstract index such as "i
> ".
(line 69): After declaring for loop statement or if-else statement, add indentation (4 whitespaces).
(line 34): Are you using tilde (~) for negation? If so, for negation, use bang (!) instead of tilde (~).
(line 42): Are you using tilde (~) for negation? If so, for negation, use bang (!) instead of tilde (~).
(line 48): Are you using tilde (~) for negation? If so, for negation, use bang (!) instead of tilde (~).
(line 53): Are you using tilde (~) for negation? If so, for negation, use bang (!) instead of tilde (~).
(line 60): Are you using tilde (~) for negation? If so, for negation, use bang (!) instead of tilde (~).
(line 73): Are you taking missing values into account properly? (Remember that "a != 0" or "a > 0" include cases where a is
> missing.)
```

Bad practice	Occurrences
Hard tabs used instead of soft tabs:	Yes
One-letter local name in for-loop:	3
Non-standard indentation in { } code block:	7
No indentation on line following ///:	1
Use of . where missing() is appropriate:	5
Missing whitespaces around operators:	0
Implicit logic in if-condition:	1
Delimiter changed:	1
Working directory changed:	0
Lines too long:	5
Global macro reference without { }:	0
Potential omission of missing values in expression:	1
Backslash detected in potential file path:	0
Tilde (~) used instead of bang (!) in expression:	5

For more information about coding guidelines visit the [Stata linter wiki](#).

Verbose use, detection feature

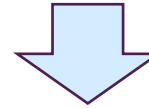
(line-by-line issues)

Correction feature

(generates new do-file, but can't
correct all issues)

Command

```
lint "test/bad.do" using "test/bad_corrected.do", automatic
```



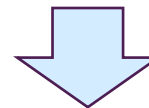
```
. lint "test/bad.do" using "test/bad_corrected.do", automatic
```

Bad practice	Occurrences
Hard tabs used instead of soft tabs:	Yes
One-letter local name in for-loop:	3
Non-standard indentation in { } code block:	7
No indentation on line following ///:	1
Use of . where missing() is appropriate:	5
Missing whitespaces around operators:	0
Implicit logic in if-condition:	1
Delimiter changed:	1
Working directory changed:	0
Lines too long:	5
Global macro reference without { }:	0
Potential omission of missing values in expression:	1
Backslash detected in potential file path:	0
Tilde (~) used instead of bang (!) in expression:	5


For more information about coding guidelines visit the [Stata linter wiki](#).


```
Correcting do-file
```

```
Corrected do-file saved to test/bad_corrected.do.
```



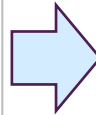
Name

 bad.do

 bad_corrected.do

Correction feature

```
bad.do x bad_corrected.do
10 * Stata codes to be corrected =====
11
12 * All hard tabs are replaced with soft tabs (= whitespaces)
13
14 * delimiter is corrected and three forward slashes will be used instead
15 #delimiter ;
16
17 foreach something in something something something something something something
18 something something { ; // some comment
19 do something ;
20 } ;
21
22 #delimiter cr
23
24 * Add indentation in brackets
25 if something {
26 do something
27 if another == 1 {
28 do that
29 }
30 }
31
32 foreach ii in potato potato cassava maize potato ///
33 cassava maize potato cassava maize potato cassava maize potato cassava maize potato
34 cassava maize potato cassava maize potato cassava maize {
35 if something ~= 1 & something != . {
36 do something
37 }
38 }
```



```
bad.do bad_corrected.do x
10 * Stata codes to be corrected =====
11
12 * All hard tabs are replaced with soft tabs (= whitespaces)
13
14 * delimiter is corrected and three forward slashes will be used instead
15
16 foreach something in something something something something something something ///
17 something ///
18 something something { // some comment
19 do something
20 }
21
22 * Add indentation in brackets
23 if something {
24 do something
25 if another == 1 {
26 do that
27 }
28 }
29
30 foreach ii in potato potato cassava maize potato ///
31 cassava maize potato cassava maize potato cassava maize potato ///
32 cassava maize potato cassava maize potato cassava maize potato ///
33 cassava maize {
34 if something ~= 1 & something != . {
35 do something
36 }
37 }
38 }
```

Recommended use

1. Use basic detection feature to get of the number of issues
2. If there only a few, use verbose mode and correct them manually
3. If there are many, use correction feature
4. Check that new do-file still works correctly
5. Fix the rest of issues manually in the new do-file

Bad practices flagged



Bad practices detected

(as of v1.02)

1. Use of soft tabs instead of hard tabs
2. Use of a one-letter local in a loop:

```
foreach i of `variables' {  
    tab `i'  
}
```

3. No indentation in lines following ///:

```
reg var_x var_y1 var_y2 ///  
var_y3 var_y4
```

Bad practices detected

(as v1.02)

4. Use of `.` instead of `missing()`
5. Missing whitespaces around math operators:

```
gen var3=var2+var1
```
6. Implicit logic in condition:

```
count if var1
```
7. Line delimiter changed
8. Working directory changed

Bad practices detected

(as of v1.02)

9. Long lines

10. Reference to global without curly brackets:

```
reg var_x $controls
```

11. Potential omission of missing value condition in expression:

```
count if var1 > 100 // note that this includes missings
```

12. Backslash in file paths

13. Use of tilde (~) instead of bang (!) in expression

Bad practices corrected

(as of v1.02)

1. `delimit#` is replaced by `///`
2. Hard tabs are replaced by soft tabs
3. Adds indentation inside loops
4. Long lines are split into two or more using `///`

Bad practices corrected

(as of v1.02)

5. If missing, adds whitespaces before curly brackets (except in globals)
6. Removes empty lines before closing a loop
7. Removed duplicated empty lines

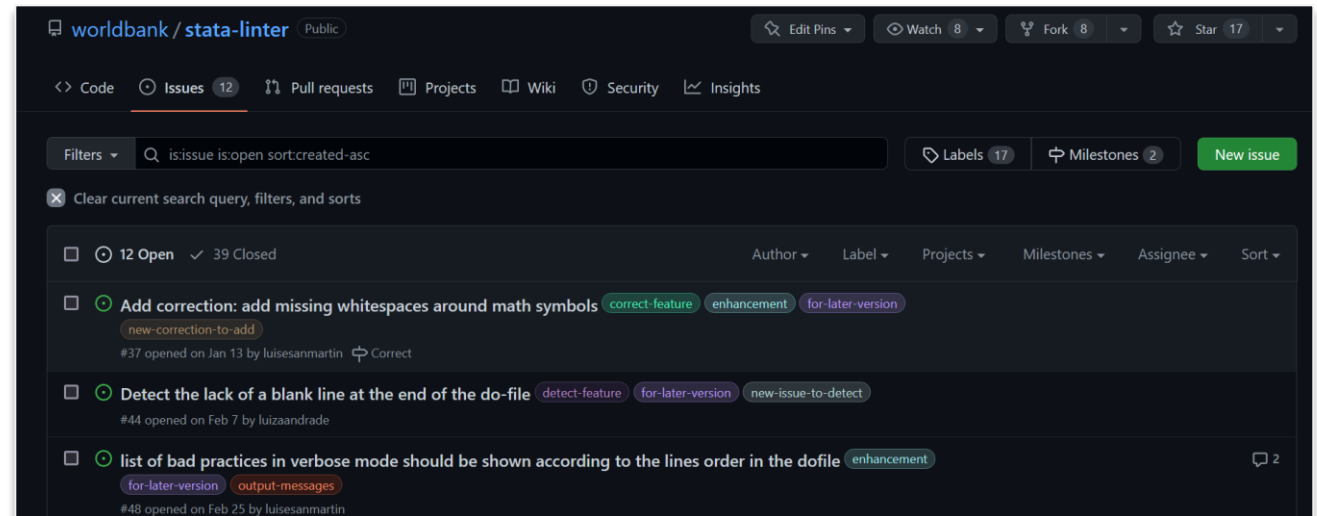
Collaborating with the project



Collaboration

Feedback, bugs detected, and suggestions:

- Submit an [issue on GH](#), or
- Send us an email to dimeanalytics@worldbank.org with the subject “Stata linter”



Collaboration

- The [project repository](#) is public
- Anyone can add their contributions and submit a pull request on GH

master 2 branches 2 tags Go to file Add file Code

luisesanmartin added ssc checklist bde52fe yesterday 223 commits

.github/workflows	only trigger on master pushes and prs	17 months ago
admin	added ssc checklist	yesterday
run	Small changes	4 months ago
src-py	moved python src and test files	yesterday
src	updated dates	yesterday
test	moved python src and test files	yesterday
.gitattributes	Initial commit	2 years ago
.gitignore	ssc meta info	yesterday
LICENSE	Update LICENSE	yesterday
README.md	replaced "inprep" option with "force"	3 days ago
stata.toc	updated version numbers in stata.toc and pkg	9 months ago
stata_linter.pkg	updated version numbers in stata.toc and pkg	9 months ago

Gracias!

Co-authors:

- Rony Rodriguez Ramirez
- Luis Eduardo San Martin
- Mizuhiro Suzuki

Contributors:

- Kristoffer Bjarkefur
- Luiza Cardoso de Andrade
- Benjamin Daniels
- Maria Jones

