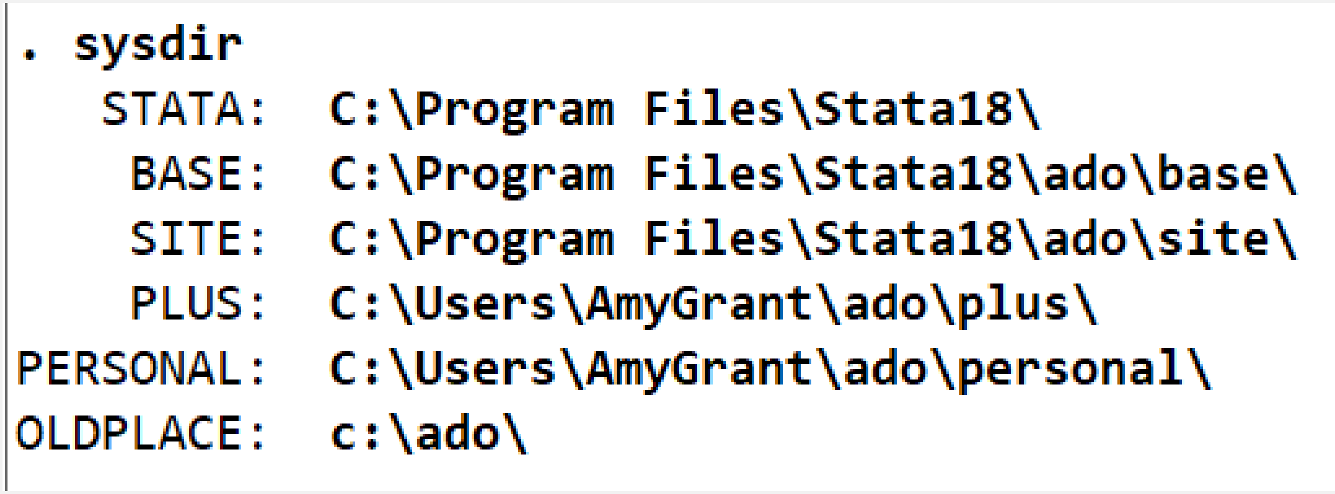
Be Bold

Editing Stata Commands to Make Your Own

An SDAS Presentation

Introducing your personal ado path



```
. sysdir
   STATA:   C:\Program Files\Stata18\
    BASE:   C:\Program Files\Stata18\ado\base\
    SITE:   C:\Program Files\Stata18\ado\site\
    PLUS:   C:\Users\AmyGrant\ado\plus\
PERSONAL:   C:\Users\AmyGrant\ado\personal\
OLDPLACE:   c:\ado\
```

www.techtips.surveydesign.com.au/post/the-sysdir-and-adopath-command

Part 1:
Altering Existing Commands (.ado)

Why Alter Existing Commands?

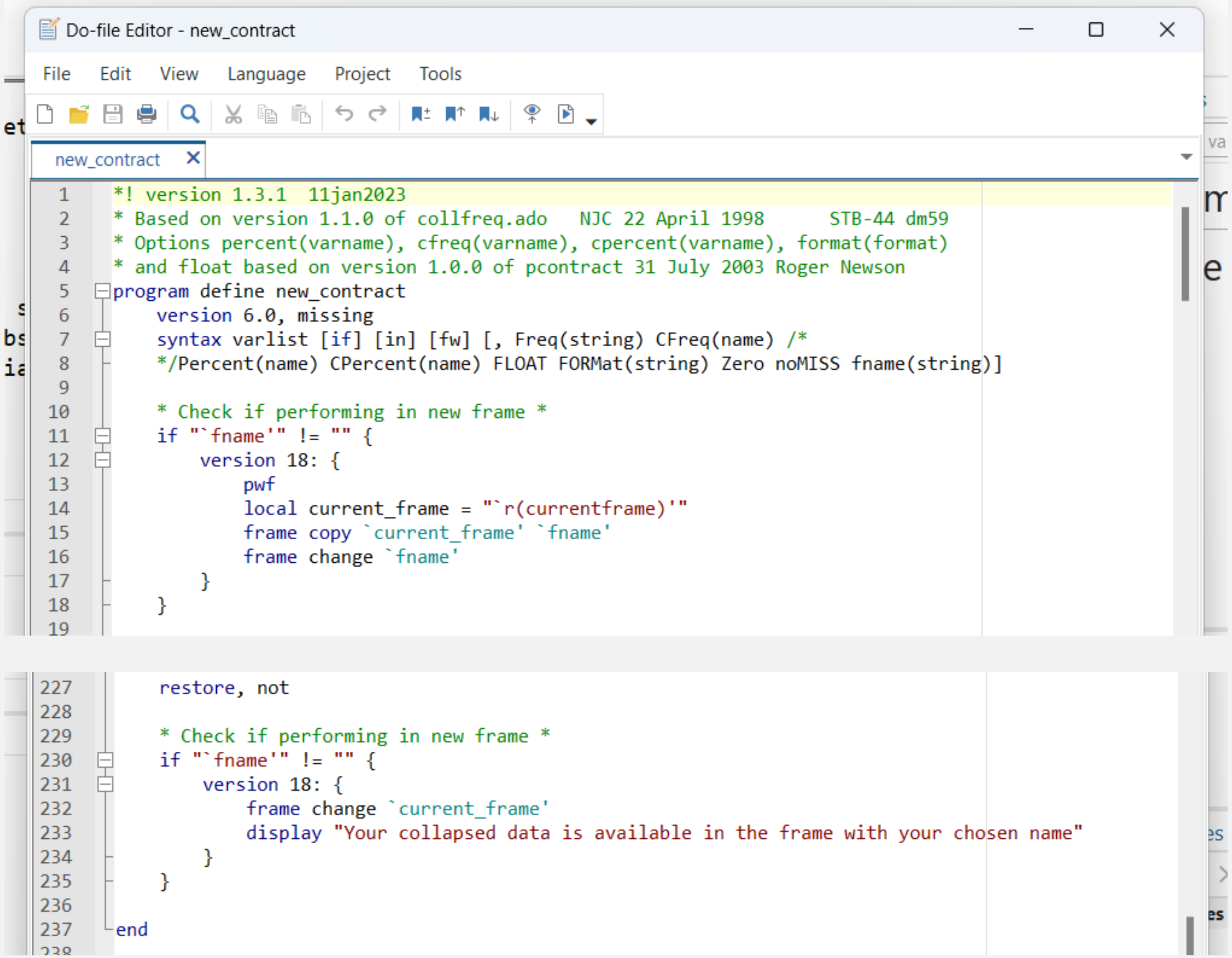
Benefits

- Tailor existing commands to your own purposes
- Increase functionality of commands

Dangers

- Impede the original functionality of the code
- Different commands were built in different versions of Stata

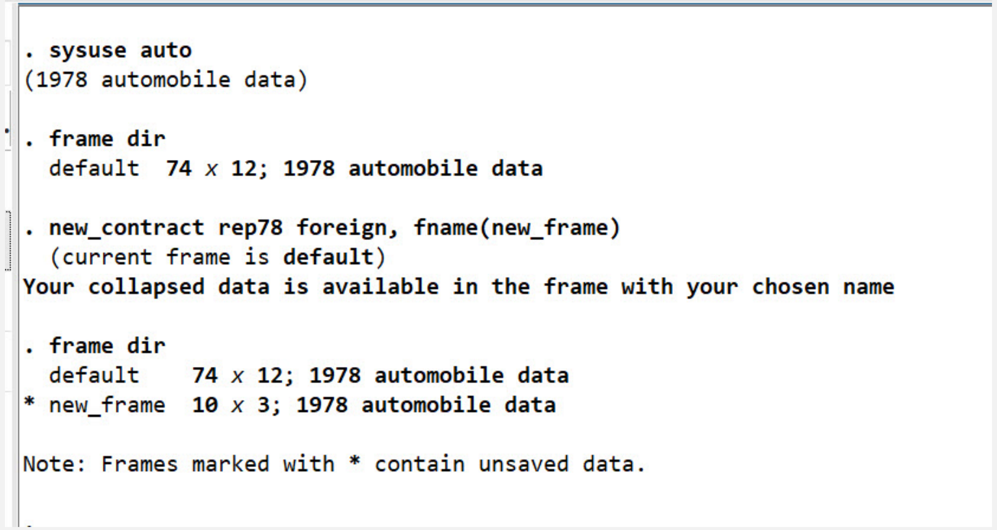
Example: Contract /Collapse Commands



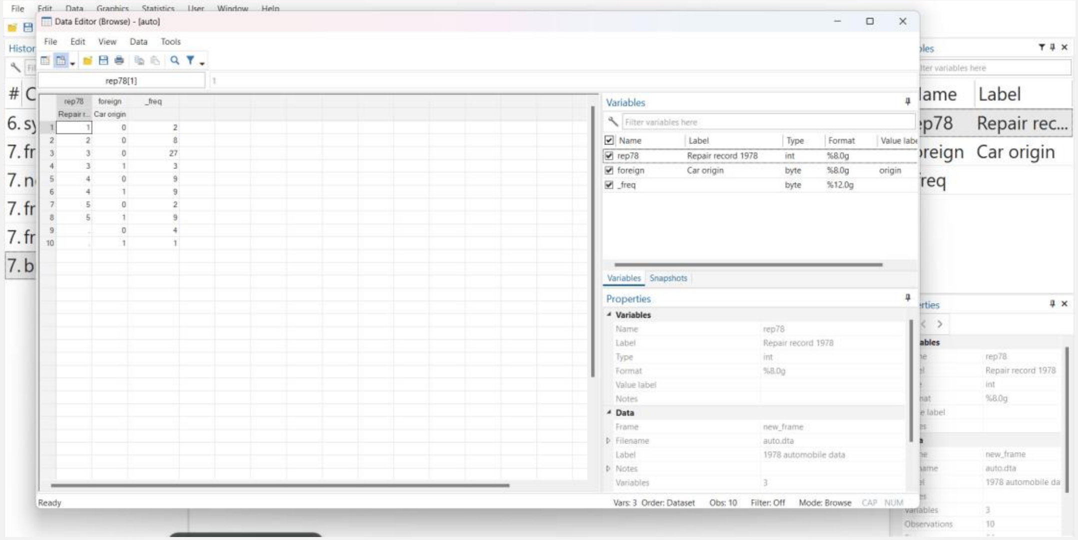
```
*! version 1.3.1  11jan2023
* Based on version 1.1.0 of collfreq.ado   NJC 22 April 1998      STB-44 dm59
* Options percent(varname), cfreq(varname), cpercent(varname), format(format)
* and float based on version 1.0.0 of pcontract 31 July 2003 Roger Newson
program define new_contract
    version 6.0, missing
    syntax varlist [if] [in] [fw] [, Freq(string) CFreq(name) /*
    */Percent(name) CPercent(name) FLOAT FORMat(string) Zero noMISS fname(string)]

    * Check if performing in new frame *
    if "`fname'" != "" {
        version 18: {
            pwf
            local current_frame = "`r(currentframe)'"
            frame copy `current_frame' `fname'
            frame change `fname'
        }
    }

```

```
    restore, not

    * Check if performing in new frame *
    if "`fname'" != "" {
        version 18: {
            frame change `current_frame'
            display "Your collapsed data is available in the frame with your chosen name"
        }
    }

end
```

Example: Contract/Collapse Commands



```
. sysuse auto
(1978 automobile data)

. frame dir
  default  74 x 12; 1978 automobile data

. new_contract rep78 foreign, fname(new_frame)
  (current frame is default)
Your collapsed data is available in the frame with your chosen name

. frame dir
  default    74 x 12; 1978 automobile data
* new_frame  10 x 3; 1978 automobile data

Note: Frames marked with * contain unsaved data.
```



	rep78 Repair r...	foreign Car origin	_freq
1	1	0	2
2	2	0	8
3	3	0	27
4	3	1	3
5	4	0	9
6	4	1	9
7	5	0	2
8	5	1	9
9	.	0	4
10	.	1	1

☑ Name	Label	Type	Format	Value labe
☑ rep78	Repair record 1978	int	%8.0g	
☑ foreign	Car origin	byte	%8.0g	origin
☑ _freq		byte	%12.0g	

Variables	
Name	rep78
Label	Repair record 1978
Type	int
Format	%8.0g
Value label	
Notes	
Data	
Frame	new_frame
Filename	auto.dta
Label	1978 automobile data
Notes	
Variables	3

Ready

Vars: 3 Order: Dataset Obs: 10 Filter: Off Mode: Browse CAP NUM

Part 2:
Creating New Commands (.ado)

Why User Written Commands?

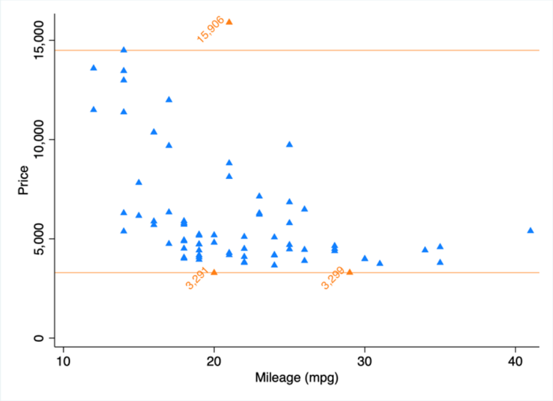
Benefits

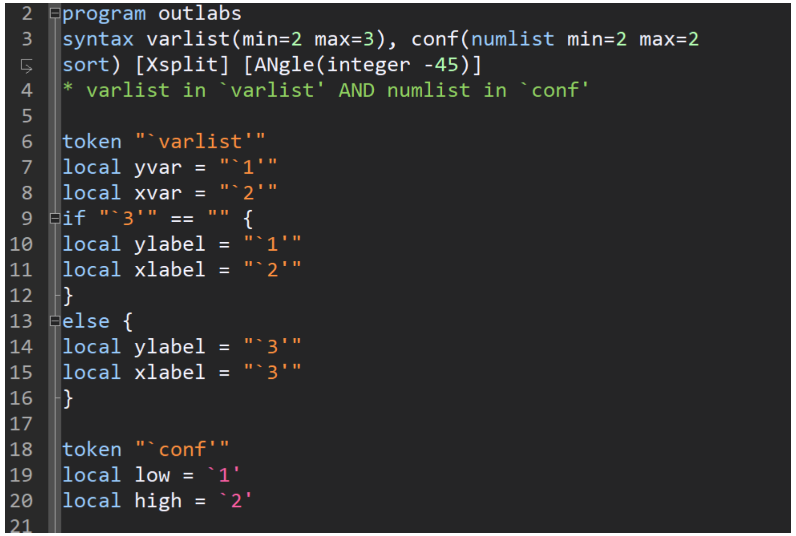
- Efficiency - Automating existing sets of frequently used commands
- Customisation - Automating niche applications
- Reproducibility

Dangers

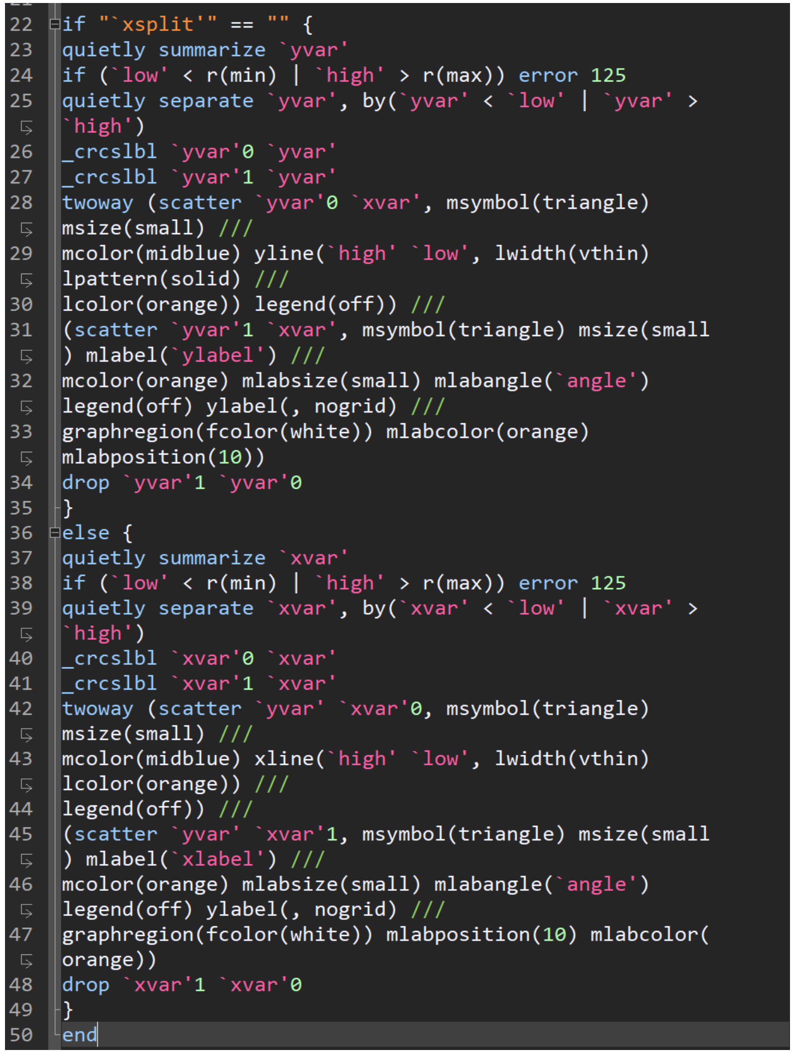
- Generalisability
- Requires a good base knowledge of programming in Stata
- More difficult to acquire support

Example: The Outlabs Command developed by Laura





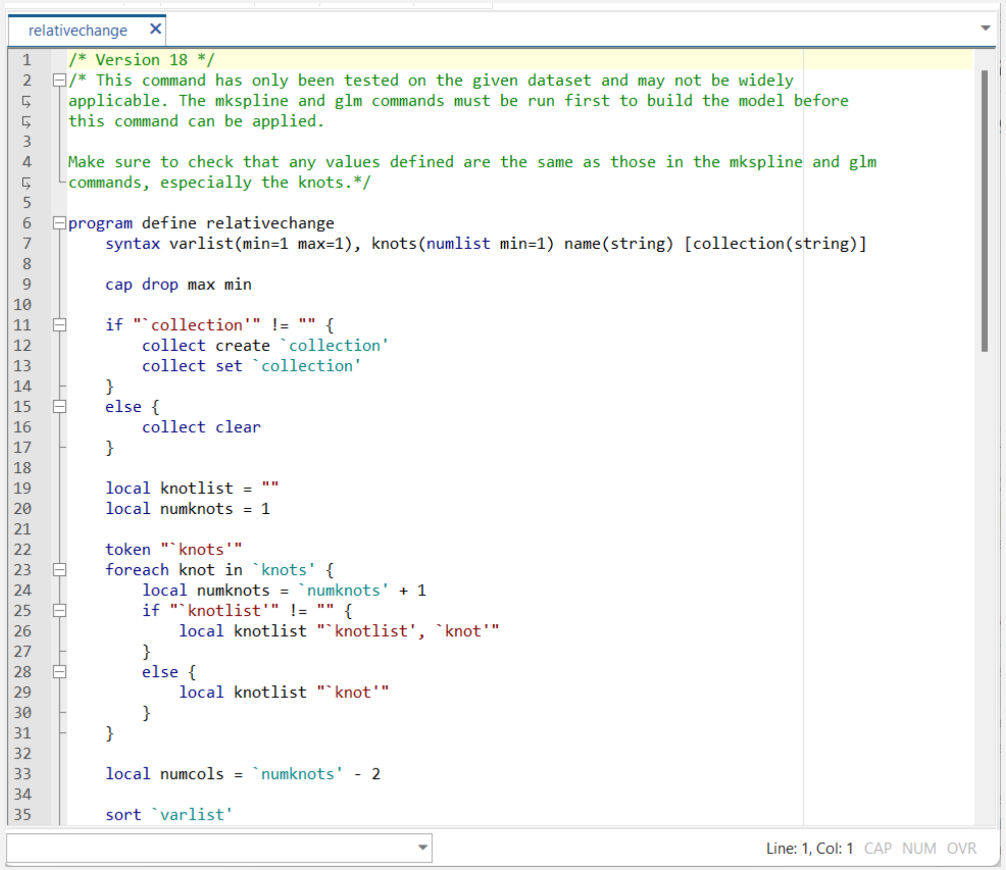
```
program outlabs
syntax varlist(min=2 max=3), conf(numlist min=2 max=2 sort) [Xsplit] [ANGle(integer -45)]
* varlist in `varlist' AND numlist in `conf'

token "`varlist'"
local yvar = "`1'"
local xvar = "`2'"
if "`3'" == "" {
local ylabel = "`1'"
local xlabel = "`2'"
}
else {
local ylabel = "`3'"
local xlabel = "`3'"
}

token "`conf'"
local low = `1'
local high = `2'
```



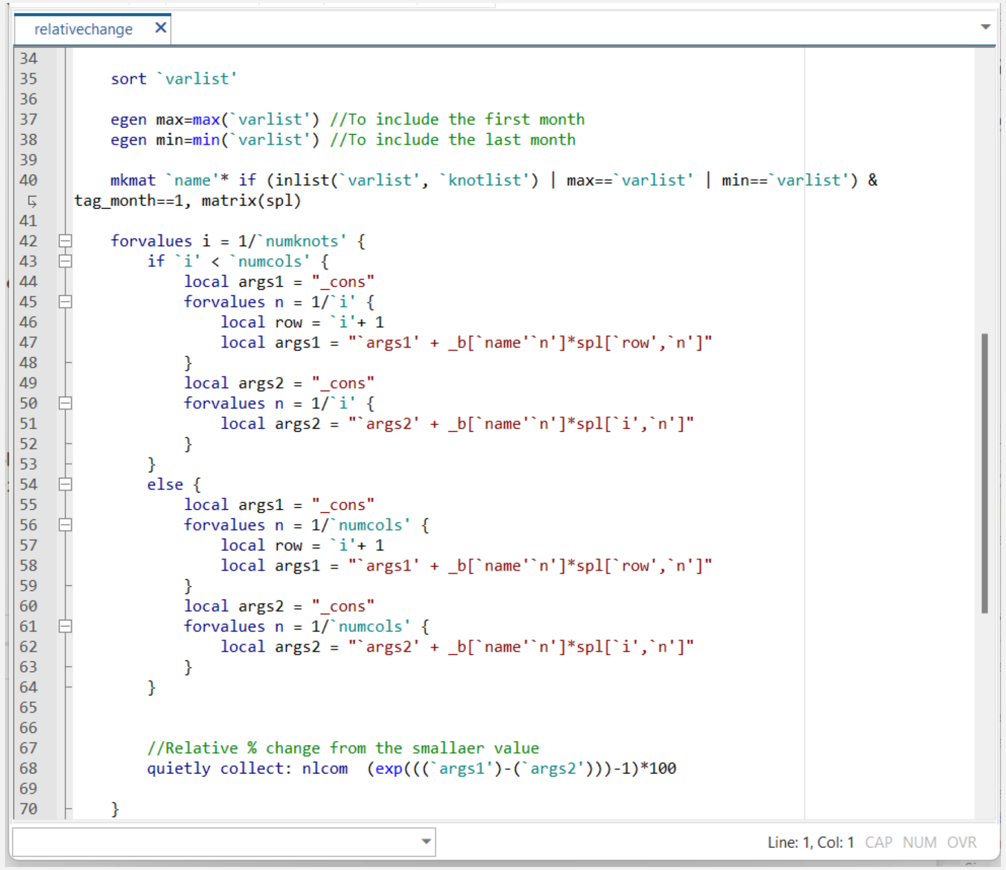
```
if "`xsplit'" == "" {
quietly summarize `yvar'
if (`low' < r(min) | `high' > r(max)) error 125
quietly separate `yvar', by(`yvar' < `low' | `yvar' > `high')
_crcslbl `yvar'0 `yvar'
_crcslbl `yvar'1 `yvar'
twoway (scatter `yvar'0 `xvar', msymbol(triangle) msize(small) ///
mcolor(midblue) yline(`high' `low', lwidth(vthin) lpattern(solid) ///
lcolor(orange)) legend(off)) ///
(scatter `yvar'1 `xvar', msymbol(triangle) msize(small) mlabel(`ylabel') ///
mcolor(orange) mlabsize(small) mlabangle(`angle') legend(off) ylabel(, nogrid) ///
graphregion(fcolor(white)) mlabcolor(orange) mlabposition(10))
drop `yvar'1 `yvar'0
}
else {
quietly summarize `xvar'
if (`low' < r(min) | `high' > r(max)) error 125
quietly separate `xvar', by(`xvar' < `low' | `xvar' > `high')
_crcslbl `xvar'0 `xvar'
_crcslbl `xvar'1 `xvar'
twoway (scatter `yvar' `xvar'0, msymbol(triangle) msize(small) ///
mcolor(midblue) xline(`high' `low', lwidth(vthin) lcolor(orange)) ///
legend(off)) ///
(scatter `yvar' `xvar'1, msymbol(triangle) msize(small) mlabel(`xlabel') ///
mcolor(orange) mlabsize(small) mlabangle(`angle') legend(off) ylabel(, nogrid) ///
graphregion(fcolor(white)) mlabposition(10) mlabcolor(orange))
drop `xvar'1 `xvar'0
}
end
```

https://www.techtips.surveydesign.com.au/post/the-outlabs-command.html

Example: relativechange.ado

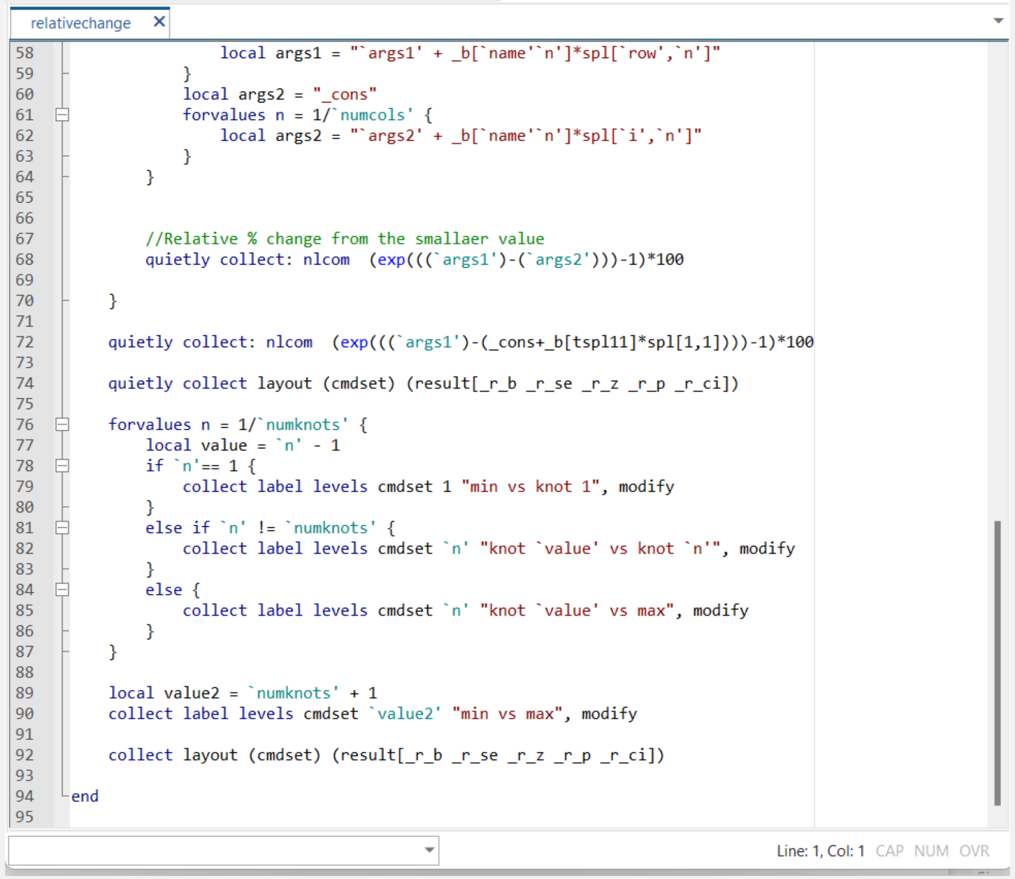


```
/* Version 18 */
/* This command has only been tested on the given dataset and may not be widely
applicable. The mkspline and glm commands must be run first to build the model before
this command can be applied.

Make sure to check that any values defined are the same as those in the mkspline and glm
commands, especially the knots.*/

program define relativechange
    syntax varlist(min=1 max=1), knots(numlist min=1) name(string) [collection(string)]

    cap drop max min

    if "`collection'" != "" {
        collect create `collection'
        collect set `collection'
    }
    else {
        collect clear
    }

    local knotlist = ""
    local numknots = 1

    token "`knots'"
    foreach knot in `knots' {
        local numknots = `numknots' + 1
        if "`knotlist'" != "" {
            local knotlist "`knotlist', `knot'"
        }
        else {
            local knotlist "`knot'"
        }
    }

    local numcols = `numknots' - 2

    sort `varlist'
```



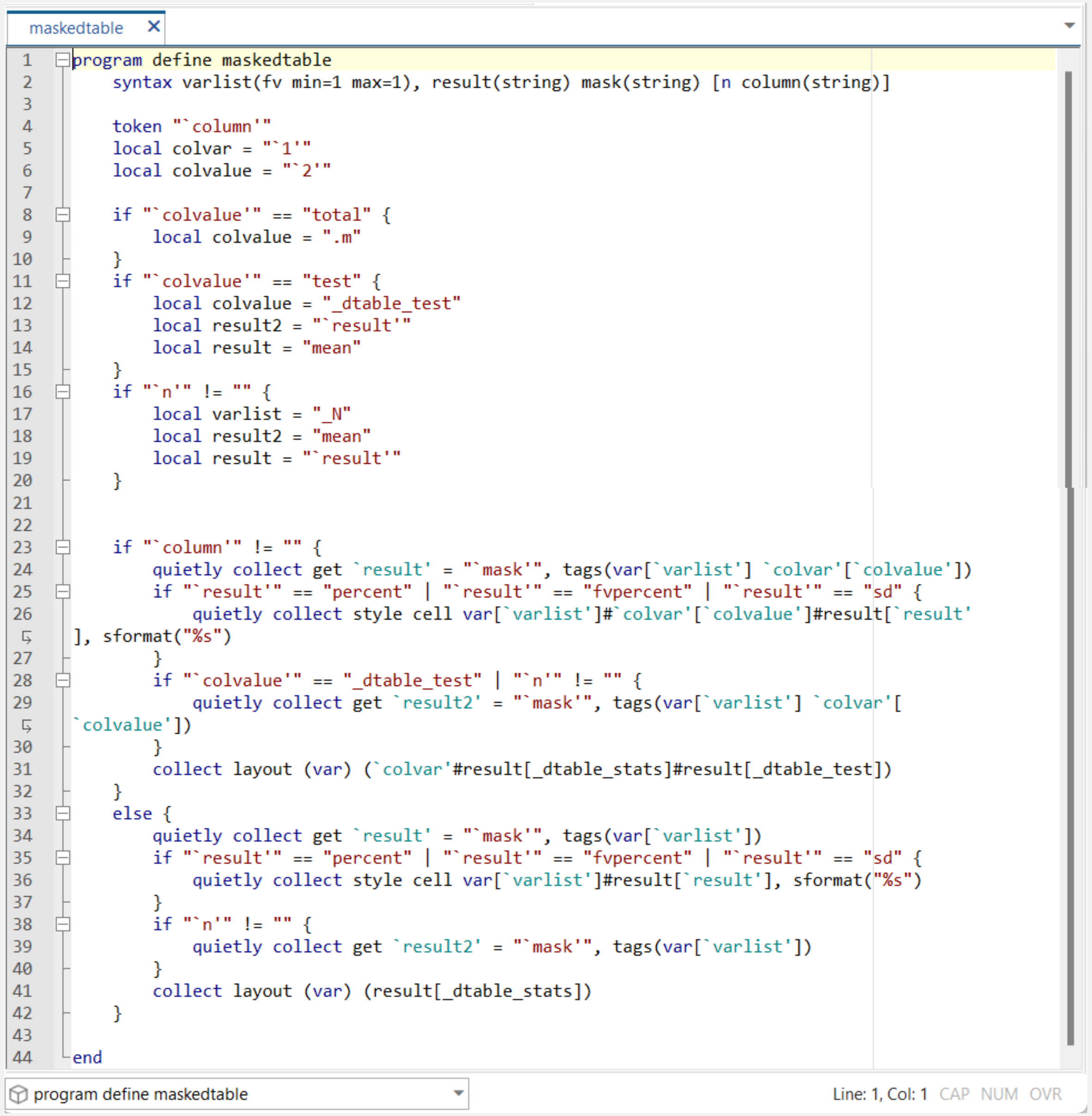
```

    sort `varlist'

    egen max=max(`varlist') //To include the first month
    egen min=min(`varlist') //To include the last month

    mkmat `name'* if (inlist(`varlist', `knotlist') | max==`varlist' | min==`varlist') &
tag_month==1, matrix(spl)

    forvalues i = 1/`numknots' {
        if `i' < `numcols' {
            local args1 = "_cons"
            forvalues n = 1/`i' {
                local row = `i'+ 1
                local args1 = "`args1' + _b[`name'`n']*spl[`row',`n']"
            }
            local args2 = "_cons"
            forvalues n = 1/`i' {
                local args2 = "`args2' + _b[`name'`n']*spl[`i',`n']"
            }
        }
        else {
            local args1 = "_cons"
            forvalues n = 1/`numcols' {
                local row = `i'+ 1
                local args1 = "`args1' + _b[`name'`n']*spl[`row',`n']"
            }
            local args2 = "_cons"
            forvalues n = 1/`numcols' {
                local args2 = "`args2' + _b[`name'`n']*spl[`i',`n']"
            }
        }


        //Relative % change from the smallaer value
        quietly collect: nlcom  (exp(((`args1')-(`args2')))-1)*100

    }
```

Example: relativechange.ado



```
                local args1 = "`args1' + _b[`name'`n']*spl[`row',`n']"
            }
            local args2 = "_cons"
            forvalues n = 1/`numcols' {
                local args2 = "`args2' + _b[`name'`n']*spl[`i',`n']"
            }
        }


        //Relative % change from the smallaer value
        quietly collect: nlcom  (exp(((`args1')-(`args2')))-1)*100

    }

    quietly collect: nlcom  (exp(((`args1')-(_cons+_b[tspl11]*spl[1,1])))-1)*100

    quietly collect layout (cmdset) (result[_r_b _r_se _r_z _r_p _r_ci])

    forvalues n = 1/`numknots' {
        local value = `n' - 1
        if `n'== 1 {
            collect label levels cmdset 1 "min vs knot 1", modify
        }
        else if `n' != `numknots' {
            collect label levels cmdset `n' "knot `value' vs knot `n'", modify
        }
        else {
            collect label levels cmdset `n' "knot `value' vs max", modify
        }
    }

    local value2 = `numknots' + 1
    collect label levels cmdset `value2' "min vs max", modify

    collect layout (cmdset) (result[_r_b _r_se _r_z _r_p _r_ci])

end

```

Example: Masking Tables



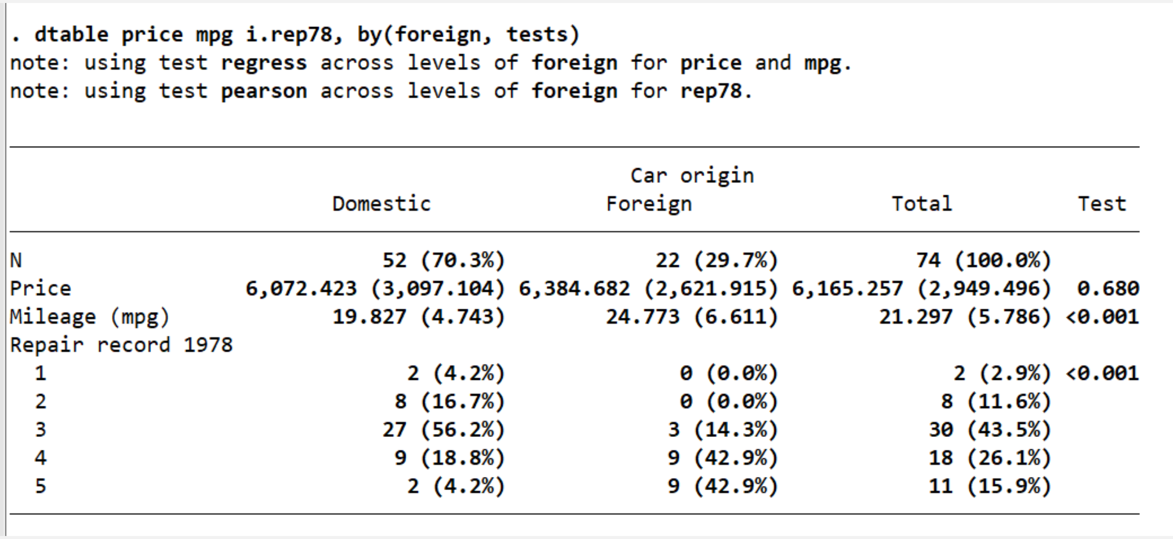
```
program define maskedtable
    syntax varlist(fv min=1 max=1), result(string) mask(string) [n column(string)]

    token "`column'"
    local colvar = "`1'"
    local colvalue = "`2'"

    if "`colvalue'" == "total" {
        local colvalue = ".m"
    }
    if "`colvalue'" == "test" {
        local colvalue = "_dtable_test"
        local result2 = "`result'"
        local result = "mean"
    }
    if "`n'" != "" {
        local varlist = "_N"
        local result2 = "mean"
        local result = "`result'"
    }


    if "`column'" != "" {
        quietly collect get `result' = "`mask'", tags(var[`varlist'] `colvar'[`colvalue'])
        if "`result'" == "percent" | "`result'" == "fvpercent" | "`result'" == "sd" {
            quietly collect style cell var[`varlist']#`colvar'[`colvalue']#result[`result'], sformat("%s")
        }
        if "`colvalue'" == "_dtable_test" | "`n'" != "" {
            quietly collect get `result2' = "`mask'", tags(var[`varlist'] `colvar'[`colvalue'])
        }
        collect layout (var) (`colvar'#result[_dtable_stats]#result[_dtable_test])
    }
    else {
        quietly collect get `result' = "`mask'", tags(var[`varlist'])
        if "`result'" == "percent" | "`result'" == "fvpercent" | "`result'" == "sd" {
            quietly collect style cell var[`varlist']#result[`result'], sformat("%s")
        }
        if "`n'" != "" {
            quietly collect get `result2' = "`mask'", tags(var[`varlist'])
        }
        collect layout (var) (result[_dtable_stats])
    }

end
```

Example: Masking Tables

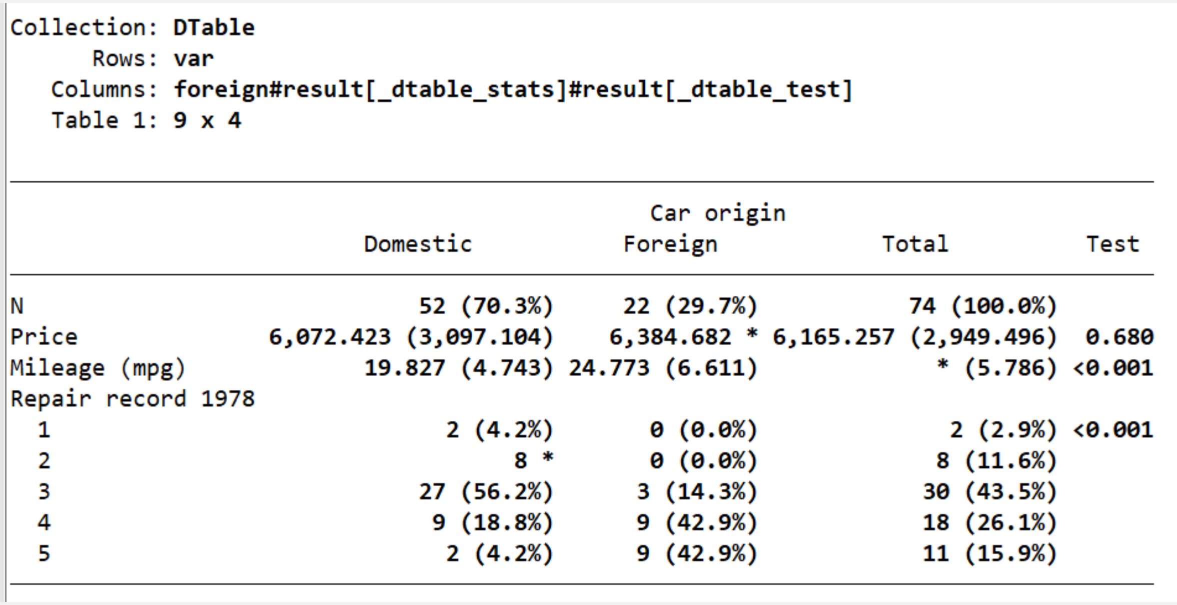
maskedtable price, result(sd) mask(*) column(foreign 1)

maskedtable mpg, result(mean) mask(*) column(foreign total)

maskedtable 2.rep78, result(fvpercent) mask(*) column(foreign 0)



```
. dtable price mpg i.rep78, by(foreign, tests)
note: using test regress across levels of foreign for price and mpg.
note: using test pearson across levels of foreign for rep78.

───────────────────────────────────────────────────────────────────────────────────────────
                                                  Car origin
                          Domestic              Foreign                Total         Test
───────────────────────────────────────────────────────────────────────────────────────────
N                             52 (70.3%)            22 (29.7%)            74 (100.0%)
Price              6,072.423 (3,097.104) 6,384.682 (2,621.915) 6,165.257 (2,949.496)  0.680
Mileage (mpg)             19.827 (4.743)        24.773 (6.611)        21.297 (5.786) <0.001
Repair record 1978
  1                             2 (4.2%)              0 (0.0%)              2 (2.9%) <0.001
  2                            8 (16.7%)              0 (0.0%)             8 (11.6%)
  3                           27 (56.2%)             3 (14.3%)            30 (43.5%)
  4                            9 (18.8%)             9 (42.9%)            18 (26.1%)
  5                             2 (4.2%)             9 (42.9%)            11 (15.9%)
───────────────────────────────────────────────────────────────────────────────────────────
```



```
Collection: DTable
      Rows: var
   Columns: foreign#result[_dtable_stats]#result[_dtable_test]
   Table 1: 9 x 4

────────────────────────────────────────────────────────────────────────────────────
                                               Car origin
                          Domestic           Foreign            Total          Test
────────────────────────────────────────────────────────────────────────────────────
N                             52 (70.3%)     22 (29.7%)           74 (100.0%)
Price              6,072.423 (3,097.104)    6,384.682 * 6,165.257 (2,949.496)  0.680
Mileage (mpg)             19.827 (4.743) 24.773 (6.611)             * (5.786) <0.001
Repair record 1978
  1                             2 (4.2%)       0 (0.0%)              2 (2.9%) <0.001
  2                                  8 *       0 (0.0%)             8 (11.6%)
  3                           27 (56.2%)      3 (14.3%)            30 (43.5%)
  4                            9 (18.8%)      9 (42.9%)            18 (26.1%)
  5                             2 (4.2%)      9 (42.9%)            11 (15.9%)
────────────────────────────────────────────────────────────────────────────────────
```

Where to Learn More

SDAS TechTips Website

www.techtips.surveydesign.com.au/index.html

SDAS Webinars – Two free webinars in March

https://www.surveydesign.com.au/webinars.html

CSA - Core Module 3 - Programming Foundations

https://www.surveydesign.com.au/certified/module.html?trainId=MC-Ob00000LK48E

Books - An Introduction to Stata Programming and

The Mata Book

https://www.surveydesign.com.au/stata/books.html

Browse existing ado files - in Stata find your adopath and look at existing command files for inspiration

Stata documentation/manuals

StataList – the Stata forum

https://www.statalist.org/forums/

The Stata Journal

Programming an estimation command in Stata – a series by David Drukker on the Stata blog

https://blog.stata.com/2016/01/15/programming-an-estimation-command-in-stata-a-map-to-posted-entries/

SSC Archive (Boston College Department of Economics, Statistical Software Components) – a database of user-written commands

https://ideas.repec.org/s/boc/bocode.html

GitHub – a platform that hosts code

https://github.com/