Using SAS and STATA in Archival Accounting Research

Kai Chen

Dec 2, 2014

Overview

- SAS and STATA are most commonly used software in archival accounting research.
- SAS is harder to learn. STATA is much easier.
- At different empirical work stage, one is much more powerful than the other. Specifically,
  - ☑ At **sample selection** stage, the unquestionable winner is **SAS**.
  - ☑ At **data analysis** stage, the unquestionable winner is **STATA**.
- Both SAS and STATA have a great ability to add useful macros or commands developed by other users (STATA has an edge on SAS).

SAS is more powerful at sample selection stage

- Archival researchers often need to extract data from various databases on WRDS.

- SAS is much more efficient for such task (i.e., merging data) because:
  - ☑ WRDS is powered by SAS.
  - ☑ SAS fully supports SQL (Structured Query Language), a special-purpose programming language designed for merging data.

- STATA only has a “baby” merge function.

But STATA undoubtedly wins at data analysis stage

- Take several typical situations for example

- **Situation 1:** Calculate change in a variable, for example,

Firm	Year	Sales	ΔSales
A	2008	101	
A	2009	80	?
A	2010	95	
A	2011	110	
B	2008	1001	
B	2009	800	
B	2010	900	
B	2011	950	
C	2008	245	
C	2009	254	
C	2010	307	
C	2011	298	

But STATA undoubtedly wins at data analysis stage

- **Situation 1:** Calculate change in a variable, for example,

SAS	STATA

☑ **proc sql** is probably the most convenient procedure.
 	
`proc sql;`
`  create table temp`
`  as select a.*, b.sale as lagsale`
`  from dataset a left join dataset b`
`  on a.firm=b.firm and a.year=b.year+1;`
`quit;`
 	
☑ Alternatively, use **lag** function in a data step.	

But STATA undoubtedly wins at data analysis stage

- **Situation 1:** Calculate change in a variable, for example,

SAS	STATA

☑ **`proc sql`** is probably the most convenient procedure.
 	
`proc sql;`
`  create table temp`
`  as select a.*, b.sale as lagsale`
`  from dataset a left join dataset b`
`  on a.firm=b.firm and a.year=b.year+1;`
`quit;`
 	
☑ Alternatively, use **`lag`** function in data step.
	☑ Two-line commands:
	
	`tsset firm year, yearly`
	`generate chg_sale = D.sale`

But STATA undoubtedly wins at data analysis stage

- **Situation 1:** Calculate change in a variable, for example,

SAS	STATA

☑ **`proc sql`** is probably the most convenient procedure.
 	
`proc sql;`
`  create table temp`
`  as select a.*, b.sale as lagsale`
`  from dataset a left join dataset b`
`  on a.firm=b.firm and a.year=b.year+1;`
`quit;`
 	
☑ Alternatively, use **`lag`** function in data step.
	☑ Two-line commands:
	
	`tsset firm year, yearly`
	`generate chg_sale = D.sale`
	
	☑ Many useful variations, for example:
	
	`L.sale` sale t-1
	`L2.sale` sale t-2
	
	`F.sale` sale t+1
	`F2.sale` sale t+2
	
	`D.sale` sale t - sale t-1

But STATA undoubtedly wins at data analysis stage

- **Situation 2:** Fixed effects regression, for example,

DepVar = IndepVar + Year Effect

SAS	STATA

☑ **proc glm** is probably the most convenient procedure.
 	
`proc glm data=dataset;`
`class year;`
`model DepVar=IndepVar year /solution;`
`run;`
`quit;`
 	
☑ Alternatively, use **proc reg**, but time-consuming.
Step 1: Manually generate dummy variables for each sample year.
Step 2: Bring all DepVar, IndepVar, and year dummies into **proc reg** procedure.	

But STATA undoubtedly wins at data analysis stage

- **Situation 2:** Fixed effects regression, for example,

DepVar = IndepVar + Year Effect

SAS	STATA
☑ **proc glm** is probably the most convenient procedure. **proc glm** data=dataset; class year; model DepVar=IndepVar year /solution; **run**; **quit**; ☑ Alternatively, use **proc reg**, but time-consuming. Step 1: Manually generate dummy variables for each sample year. Step 2: Bring all DepVar, IndepVar, and year dummies into **proc reg** procedure.	☑ Single-line command: **regress DepVar IndepVar i.year**

But STATA undoubtedly wins at data analysis stage

Example dataset contains seven years data (from 2006 to 2012)

```
. regress depvar indepvar i.year
```

Source	SS	df	MS
Model	2.4489e+12	7	3.4985e+11
Residual	1.0614e+13	62215	170595756
Total	1.3063e+13	62222	209934363

Number of obs	= 62223
F(7, 62215)	= 2050.73
Prob > F	= 0.0000
R-squared	= 0.1875
Adj R-squared	= 0.1874
Root MSE	= 13061

depvar	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
indepvar	.0619532	.0005175	119.71	0.000	.0609389	.0629675
year						
2007	200.8505	192.4999	1.04	0.297	-176.4497	578.1508
2008	296.0818	193.7499	1.53	0.126	-83.66845	675.8321
2009	119.5906	194.5418	0.61	0.539	-261.7116	500.8929
2010	342.9851	194.8196	1.76	0.078	-38.86183	724.832
2011	587.7027	194.9924	3.01	0.003	205.5172	969.8882
2012	510.0516	192.4881	2.65	0.008	132.7745	887.3287
_cons	1965.11	134.99	14.56	0.000	1700.529	2229.69

But STATA undoubtedly wins at data analysis stage

- **Situation 3:** Clustered or Rogers standard errors, for example,

"All specifications include year and industry fixed effects and standard errors are heteroskedasticity robust, clustered at the firm level." (Costello, 2013)

SAS	STATA

☑ **proc surveyreg** is probably the most convenient procedure.
 	
`proc surveyreg data=dataset;`
`cluster firm;`
`model DepVar=IndepVar;`
`run;`
`quit;`	

But STATA undoubtedly wins at data analysis stage

- **Situation 3:** Clustered or Rogers standard errors, for example,

"All specifications include year and industry fixed effects and standard errors are heteroskedasticity robust, clustered at the firm level." (Costello, 2013)

SAS	STATA

☑ **proc surveyreg** is probably the most convenient procedure.
 	
`proc surveyreg data=dataset;`
`cluster firm;`
`model DepVar=IndepVar;`
`run;`
`quit;`
	☑ Single-line commands:
	
	`regress DepVar IndepVar, vce(cl firm)`

But STATA undoubtedly wins at data analysis stage

- A technical article concludes:

  "It is difficult to perform robust regression, or other kinds of robust methods in SAS. … STATA has a very nice array of robust methods that are very easy to use."

- STATA's estimation procedures are more additive. For example, if we have to handle both fixed effects and clustered standard errors:

  - ☑ STATA: single-line command:

    ```
    regress DepVar IndepVar i.year, vce(cl firm)
    ```

  - ☑ SAS: more complicated (**`proc surveyreg`** maybe the best)

But STATA undoubtedly wins at data analysis stage

- **<u>Situation 4:</u>** Interaction, for example,

$$DepVar = A + B + A*B$$

SAS	STATA
☑ Use **`proc reg`** <u>Step 1</u>: Manually generate a new variable equal to A*B. <u>Step 2</u>: Bring all variables into **`proc reg`** procedure. ☑ Alternatively, **`proc glm`** may be simpler.	

But STATA undoubtedly wins at data analysis stage

- **Situation 4:** Interaction, for example,

$$DepVar = A + B + A*B$$

SAS	STATA
☑ Use **`proc reg`** Step 1: Manually generate a new variable equal to A*B. Step 2: Bring all variables into **`proc reg`** procedure. ☑ Alternatively, **`proc glm`** may be simpler.	☑ One-step command: **`regress DepVar c.A##c.B`**

But STATA undoubtedly wins at data analysis stage

- **<u>Situation 5:</u>** 2SLS

2SLS is used when the model has endogenous independent variables (a common reason is omitted variables).

Once again:

- ☑ SAS: at least two regressions

- ☑ STATA: single-line command (**`ivregress`**) to complete 2 stages at once

But STATA undoubtedly wins at data analysis stage

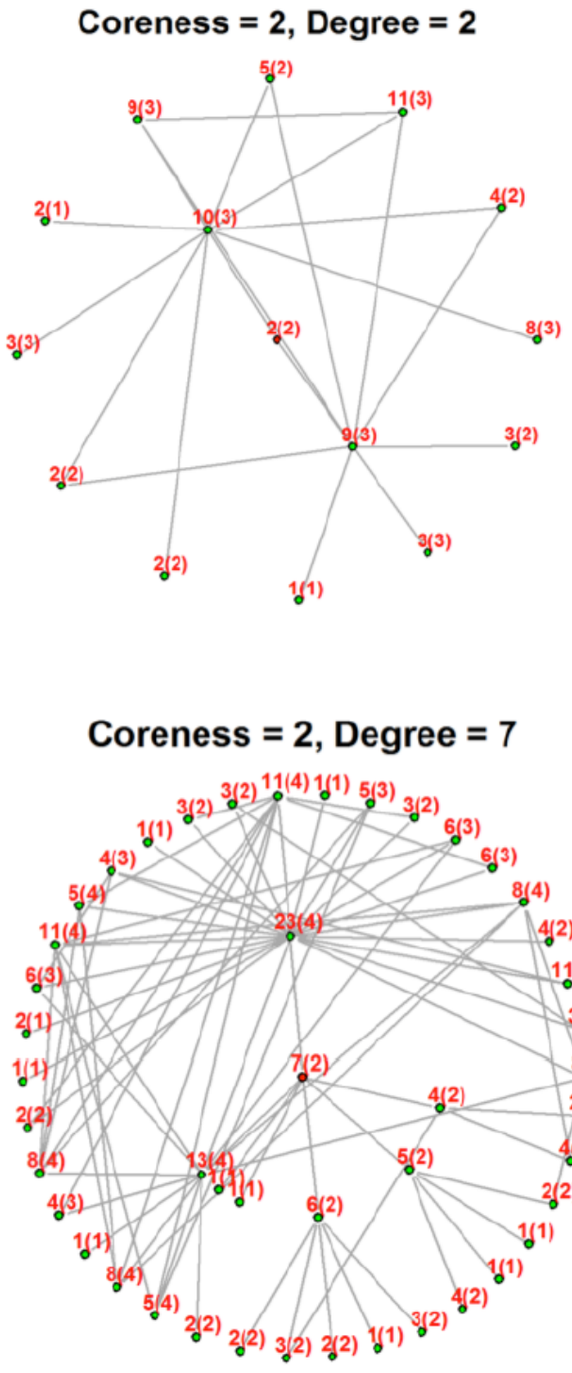
- **Situation 6:** Graphics. Remember Hollifield's paper two weeks ago?

SAS	STATA
☑ May have the most powerful graphic tools, but very technical and tricky to learn.	☑ Graph commands are very easy to use and also very powerful. ☑ Easily create publication quality graphs. ☑ Can be edited using a graph editor.

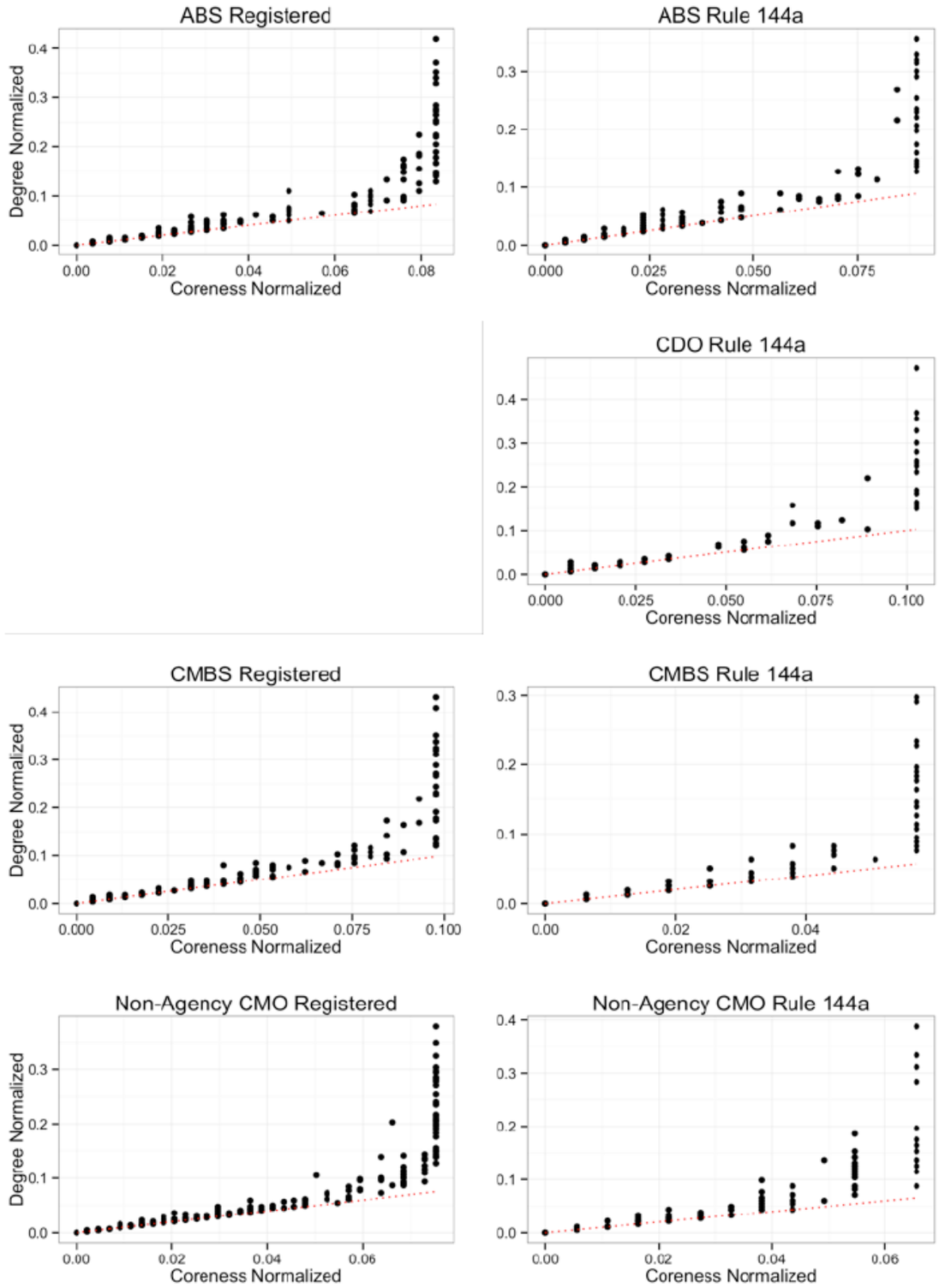
But STATA undoubtedly wins at data analysis stage

Figure 6: Non-Retail Dealers' Degree and Coreness

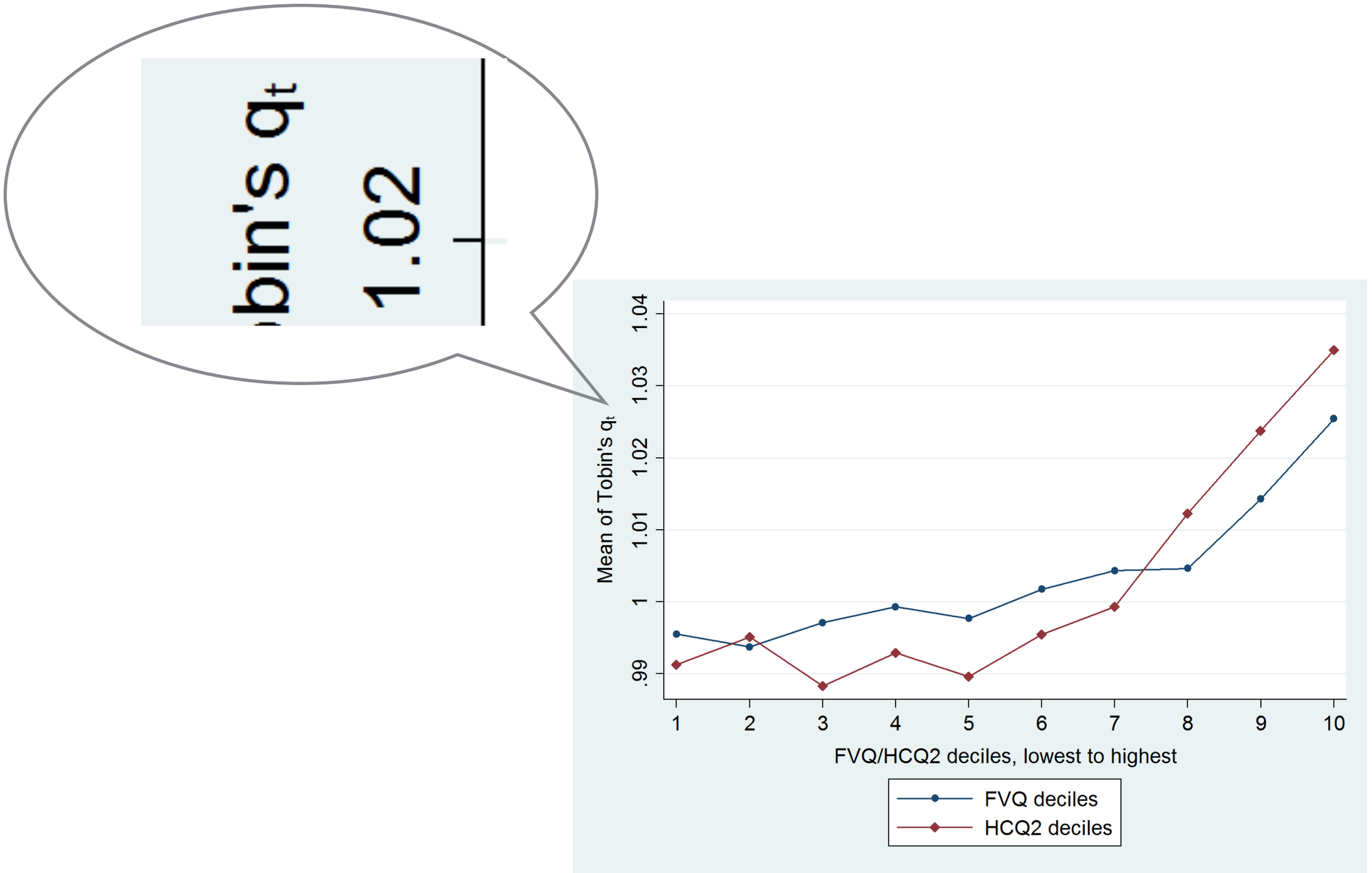
Panel A: Degree and Coreness for Two Dealers in ABS Reg.



Panel B: Degree and Coreness for All Dealers



But STATA undoubtedly wins at data analysis stage



But STATA undoubtedly wins at data analysis stage

- In short, for almost every single task in data analysis (sort, drop, group, summary statistics, regressions),

  - ☑ STATA code is shorter, more intuitive, and closer to natural language than SAS code.

  - ☑ STATA has more easy-to-use cutting-edge estimation procedures than SAS.

Top user-written macros and commands in SAS and STATA

- Both SAS and STATA users develop macros or commands for free download to enhance the software capability.
- To use macros developed by other users in SAS, we need a DIY spirit.
- Install a user-written command in STATA is much easier, thanks to Boston College Department of Economics and Christopher Baum.

Top user-written SAS macro

- **EVTSTUDY**

This macro calculates Cumulative Abnormal Returns:

- ☑ We tell the macro permno and event date
- ☑ The macro returns cumulative abnormal return within the event window (we can specify 3-day or 5-day or other).
- ☑ We can specify which model to use: market-adjusted model, standard market model, Fama-French 4-factor model.

Top user-written STATA commands

- **OUTREG** or **OUTREG2**

STATA command to write estimation tables to a Word or TeX file. For example, I run 5 regressions and each returns a table like this.

```
Linear regression                               Number of obs =     668
                                                F( 31,    40) =  730.60
                                                Prob > F      =  0.0000
                                                R-squared     =  0.4137
                                                Root MSE      =  .13863

                                    (Std. Err. adjusted for 41 clusters in rssd9001)
------------------------------------------------------------------------------
             |               Robust
         ret |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     s_nidva |    .4027016   .0608822     6.61   0.000      .279654    .5257491
       s_dva |     7.82537   2.086251     3.75   0.001       3.6089    12.04184
          uf |   -.0127315   .0398658    -0.32   0.751     -.0933031    .0678402
             |
c.s_dva#c.uf |   -14.69152   3.300949    -4.45   0.000     -21.36299   -8.020056
             |
       s_imp |    .223543   .2035808     1.10   0.279     -.1879092    .6349952
       s_oci |    1.059702   .4082858     2.60   0.013      .2345256    1.884878
```

Top user-written STATA commands

OUTREG can report all results in a more publishable table.

	RET	RET	RET	RET	RET
s_nidva	0.379	0.350	0.362	0.397	0.412
	(0.082)***	(0.081)***	(0.090)***	(0.060)***	(0.068)***
s_dva	-2.284	10.719	10.713	8.034	8.005
	(1.793)	(2.036)***	(2.052)***	(2.332)***	(2.381)***
imr	0.007	0.007	0.007	0.004	0.005
	(0.008)	(0.008)	(0.008)	(0.008)	(0.008)
uf		0.022	0.018	-0.006	-0.012
		(0.058)	(0.059)	(0.048)	(0.047)
c.s_dva#c.uf		-19.095	-19.128	-15.301	-15.314
		(3.298)***	(3.312)***	(3.663)***	(3.705)***
s_imp			0.179		0.230
			(0.204)		(0.202)
s_oci				1.056	1.064
				(0.411)**	(0.417)**
_cons	-0.112	-0.130	-0.127	-0.105	-0.101
	(0.025)***	(0.051)**	(0.052)**	(0.046)**	(0.046)**
R^2	0.38	0.39	0.39	0.41	0.41
N	668	668	668	668	668

* $p<0.1$; ** $p<0.05$; *** $p<0.01$

Top user-written STATA commands

- **`WINSOR`**

STATA command to winsorize a variable:

- ☑ We can specify the winsorization percentage (1% or 5% or other).
- ☑ We can do a one-sided winsorization.

Top user-written STATA commands

- **MDESC**

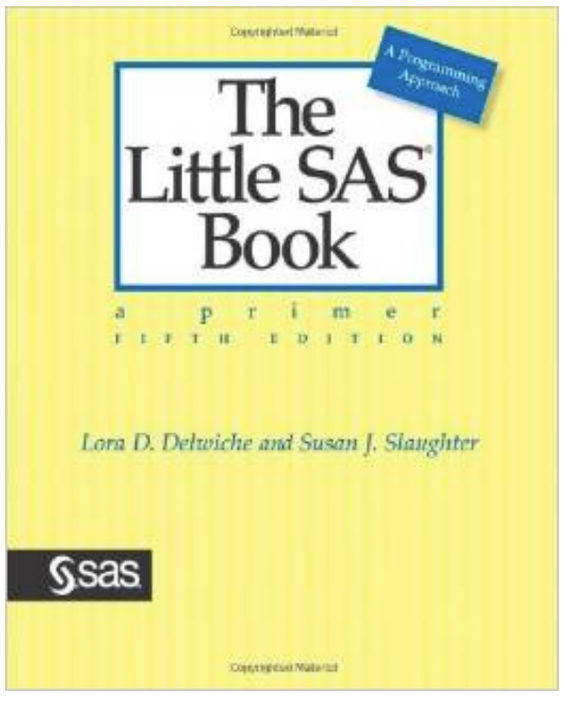
STATA command to tabulate prevalence of missing values.

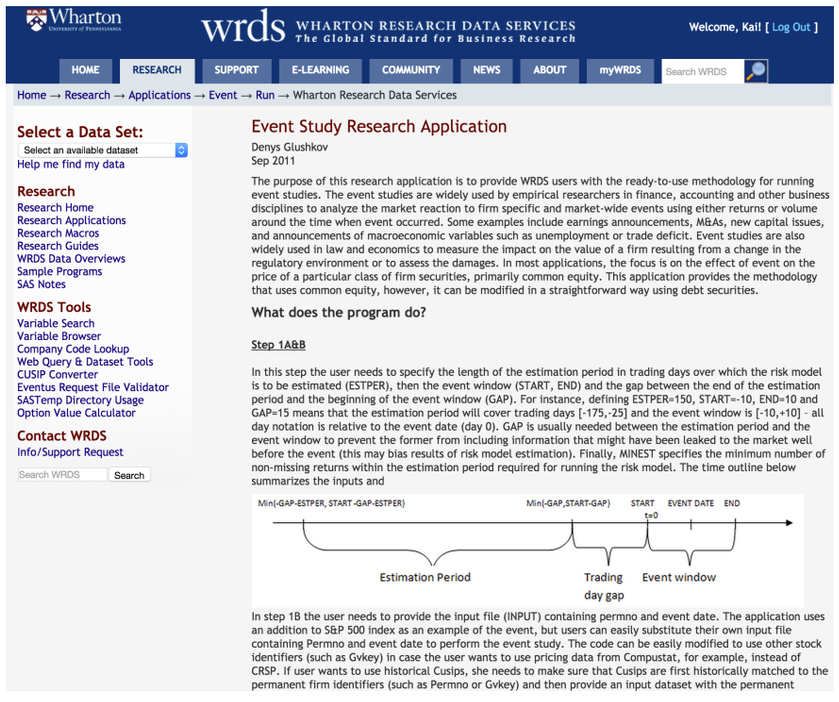
```
. mdesc
```

Variable	Missing	Total	Percent Missing
gvkey	0	78,270	0.00
datadate	0	78,270	0.00
fyear	338	78,270	0.43
tic	6	78,270	0.01
at	15,664	78,270	20.01
sale	16,032	78,270	20.48

In the end—which to choose, SAS or STATA?

- My suggestion is **both**, but for different tasks.
- If you often use WRDS and merge data, SAS SQL is almost a must and will greatly improve your work efficiency.
- Learning resources:





Event study application at WRDS.
http://wrds-web.wharton.upenn.edu/wrds/research/applications/event/run/

In the end—which to choose, SAS or STATA?

- Once you get all data and start to do data analysis, then STATA
- Learning resource: