The Stata Journal (*yyyy*) *vv*, Number *ii*, pp. 1–31

Inference for imputed latent classes using multiple imputation

Stas Kolenikov
NORC
Columbia, Missouri, USA
kolenikov-stas@norc.org

Abstract. I introduce a command to multiply impute latent classes following `gsem, lclass()` latent class analysis. This allows properly propagating uncertainty in class membership to downstream analysis that may characterize the demographic composition of the classes, or use the class as a predictor variable in statitsical models.

Keywords: st0001, postlca_class_predpute, latent class analysis, multiple imputation

1 Latent class analysis

Latent class analysis (LCA) is a commonly used statistical and quantitative social science technique of modeling counts in high dimensional contingency tables, or tables of associations of categorical variables Hagenaars and McCutcheon (2002); McCutcheon (1987). LCA is a form of loglinear modeling, so let us explain that first. If the researcher has several categorical variables $X_1, X_2, \ldots, X_p$ with categories 1 through $m_j, j = 1, \ldots, p$, at their disposal, and can produce counts $n_{k_1 k_2 \ldots k_p}$ in a complete p-dimensional table, the first step could be modeling in main effects:

$$\mathbb{E} \log n_{k_1 k_2 \ldots k_p} = \text{offset} + \sum_{j=1}^{p} \sum_{k=1}^{m_j} \beta_{j k_j} \tag{1}$$

with applicable identification constraints (such as the sum of the coefficients of a single variable is zero, or the coefficient for the first category of a variable is zero). Parameter estimates can be obtained by maximum likelihood, as equation (1) is a Poisson regression model. This model can be denoted as $X_1 + X_2 + \ldots + X_p$ main effects model. The fit of the model is assessed by the Pearson χ^2 test comparing the expected vs. observed cell counts, or the likelihood ratio test against a saturated model where each cell has its own coefficient. If the model were to be found inadequate, the researcher can entertain adding interactions, e.g. the interaction of X_1 and X_2 would have $m_1 \times m_2$ terms for each pair of values of these variables, rather than $m_1 + m_2$ main effects:

$$\mathbb{E} \log n_{k_1 k_2 \ldots k_p} = \text{offset} + \sum_{k_1=1}^{m_1} \sum_{k_2=1}^{m_2} \beta_{12, k_1 k_2} + \sum_{j=3}^{p} \sum_{k=1}^{m_j} \beta_{j k_j} \tag{2}$$

 st0001

This model can be denoted as $X_1\#X_2 + X_3 + \ldots + X_p$.

In the loglinear model notation, the latent class models are models of the form $C\#(X_1 + X_2 + \ldots + X_p)$. Categorical latent variable C is the latent class. The model is now a mixture of Poisson regressions, and maximum likelihood estimation additionally involves estimating the prevalence of each class of C.

Further extensions of latent class analysis may include:

1. Analysis with interactions of the observed variables;
2. Analysis with complex survey data (in which case estimation proceeds with `svy` prefix, and the counts are the weighted estimates of the population totals in cells);
3. Constrained analyses with structural zeroes or ones (e.g. that every member of class $C = 1$ must have the value $X_1 = 1$);
4. Constrained analyses where some variables have identical coefficients across classes.

1.1 Official Stata implementation

Official Stata `gsem, lclass()` implements the *main effects* LCA. The syntax is that of the SEM families, with the variables that the arrow points to interpreted as the outcome variables, and the latent class variable being the source of the arrow:

```
. webuse gsem_lca1
. gsem (accident play insurance stock <- ), logit lclass(C 2)
```

The goodness of fit test against the free tabulation counts is provided by `estat gof` (not available after the complex survey data analysis.)

As LCA is implemented through `gsem`, all the link functions and generalized linear model families are supported, extending the "mainstream" LCA.

1.2 Hopefully useful examples

LCA has found use in analyses of complicated economic concepts from (measured with error) survey data.

Kolenikov and Daley analyzed the latent classes of employees using the U.S. Department of Labor Worker Classification Survey. The observed variables were (composite) self-report of the employment status (are you an employee at your job; do you refer to your work as your business, your client, your job, etc.); tax status (the forms that the worker receives from their firm: W-2, 1099, K-1, etc.); behavioral control (functions the worker performs and the degree of control over these functions, such as direct reporting to somebody, schedule, permission to leave, etc.); and non-control composite (hired for fixed time or specific project). They found the best fitting model to contain three classes: employees-and-they-know-it (59%), nonemployees-and-they-know-it (24%), and confused (17%) who classify themselves as employees but their tax documentation is unclear, and other variables tend to place them into non-employee status.

1.3 Scope for this package

Researchers are often interested in describing the latent classes or using these classes in analysis as predictors or as moderators. The official [SEM] **gsem postestimation** commands provide limited possibilities, namely reporting of the means of the dependent variables by class via `estat lcmean`. For nearly all meaningful applications of LCA, this is insufficient.

The program distributed with the current package, `postlca_class_predpute`, provides a pathway for the appropriate statistical inference that would account for uncertainty in class prediction. This is achieved through the mechanics of multiple imputation (van Buuren 2018). The name is supposed to convey that

1. it is supposed to be run after LCA as a post-estimation command;
2. it predicts / imputes the latent classes.

2 The new command

Imputation of latent classes, a `gsem` postestimation command:

```
\begin{stsyntax}
    postlca\_class\_predpute,
    lcimpute(\varname)
    addm(\num)
    \optional{ seed(\num) }
\end{stsyntax}
```

`lcimpute(`*varname*`)` specifies the name of the latent class variable to be imputed. This option is required.

`addm(`#`)` specifies the number of imputations to be created. This option is required.

`seed(`#`)` specifies the random number seed.

3 Examples

3.1 Stata manual data set example

The LCA capabilities of Stata are exemplified in [SEM] **Example 50g**:

```
. frame change default
. cap frame gsem_lca1: clear
. cap frame drop gsem_lca1
. frame create gsem_lca1
```

```
. frame change gsem_lca1
.
. webuse gsem_lca1.dta, clear
(Latent class analysis)

. describe

Contains data from https://www.stata-press.com/data/r18/gsem_lca1.dta
 Observations:           216                  Latent class analysis
    Variables:             4                  17 Jan 2023 12:52
                                              (_dta has notes)
-------------------------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------------------------
accident        byte    %9.0g                 Would testify against friend in accident case
play            byte    %9.0g                 Would give negative review of friend´s play
insurance       byte    %9.0g                 Would disclose health concerns to friend´s
                                                insurance company
stock           byte    %9.0g                 Would keep company secret from friend
-------------------------------------------------------------------------------------------------
Sorted by: accident  play  insurance  stock

. gsem (accident play insurance stock <-), logit lclass(C 2)
  (output omitted)
Generalized structural equation model                     Number of obs = 216
Log likelihood = -504.46767

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
1.C          |  (base outcome)
-------------+----------------------------------------------------------------
2.C          |
       _cons |  -.9482041   .2886333    -3.29   0.001    -1.513915   -.3824933
------------------------------------------------------------------------------

Class:    1

Response: accident
Family:   Bernoulli
Link:     Logit

Response: play
Family:   Bernoulli
Link:     Logit

Response: insurance
Family:   Bernoulli
Link:     Logit

Response: stock
Family:   Bernoulli
Link:     Logit

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
accident     |
       _cons |   .9128742   .1974695     4.62   0.000     .5258411    1.299907
-------------+----------------------------------------------------------------
play         |
       _cons |  -.7099072   .2249096    -3.16   0.002    -1.150722   -.2690926
-------------+----------------------------------------------------------------
insurance    |
```

```
       _cons |  -.6014307   .2123096    -2.83   0.005     -1.01755   -.1853115
-------------+----------------------------------------------------------------
stock        |
       _cons |  -1.880142   .3337665    -5.63   0.000    -2.534312   -1.225972
------------------------------------------------------------------------------

Class:    2
```

(*output omitted*)

```
------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
accident     |
       _cons |   4.983017   3.745987     1.33   0.183    -2.358982    12.32502
-------------+----------------------------------------------------------------
play         |
       _cons |   2.747366   1.165853     2.36   0.018     .4623372    5.032395
-------------+----------------------------------------------------------------
insurance    |
       _cons |   2.534582   .9644841     2.63   0.009     .6442279    4.424936
-------------+----------------------------------------------------------------
stock        |
       _cons |   1.203416   .5361735     2.24   0.025     .1525356    2.254297
------------------------------------------------------------------------------
```

One of the official post-estimation commands available after `gsem, lclass()` is the computation of the class-specific means of the outcome variables:

```
. set rmsg on
r; t=0.00 14:45:54
. estat lcprob
Latent class marginal probabilities                   Number of obs = 216

--------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.     [95% conf. interval]
-------------+------------------------------------------------
           C |
          1  |   .7207539   .0580926      .5944743    .8196407
          2  |   .2792461   .0580926      .1803593    .4055257
--------------------------------------------------------------
r; t=1.13 14:45:55
. estat lcmean
Latent class marginal means                           Number of obs = 216

--------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.     [95% conf. interval]
-------------+------------------------------------------------
1            |
    accident |   .7135879   .0403588      .6285126    .7858194
        play |   .3296193   .0496984      .2403573    .4331299
   insurance |   .3540164   .0485528      .2655049    .4538042
       stock |   .1323726   .0383331      .0734875    .2268872
-------------+------------------------------------------------
2            |
    accident |   .9931933   .0253243      .0863544    .9999956
        play |   .9397644   .0659957      .6135685    .9935191
   insurance |   .9265309   .0656538      .6557086    .9881667
       stock |    .769132   .0952072      .5380601    .9050206
--------------------------------------------------------------
```

```
r; t=4.48 14:46:00
. set rmsg off
```

The mutiple imputation version of this estimation task could look as follows:

```
. set rmsg on
r; t=0.00 14:46:00
. postlca_class_predpute, lcimpute(lclass) addm(10) seed(12345)
(216 missing values generated)
(10 imputations added; M = 10)
r; t=0.05 14:46:00
. mi estimate : prop lclass
Multiple-imputation estimates     Imputations     =         10
Proportion estimation             Number of obs   =        216
                                  Average RVI     =     0.4594
                                  Largest FMI     =     0.3319
                                  Complete DF     =        215
DF adjustment:   Small sample     DF:     min     =      55.99
                                          avg     =      55.99
Within VCE type:     Analytic             max     =      55.99

-------------------------------------------------------------
             |                                Normal
             |  Proportion   Std. err.     [95% conf. interval]
-------------+-----------------------------------------------
      lclass |
           1 |    .7236111   .0367281      .6500355    .7971867
           2 |    .2763889   .0367281      .2028133    .3499645
-------------------------------------------------------------
r; t=2.01 14:46:02
. mi estimate : mean accident, over(lclass)
Multiple-imputation estimates     Imputations     =         10
Mean estimation                   Number of obs   =        216
                                  Average RVI     =     0.3882
                                  Largest FMI     =     0.4485
                                  Complete DF     =        215
DF adjustment:   Small sample     DF:     min     =      35.59
                                          avg     =     116.62
Within VCE type:     Analytic             max     =     197.64

------------------------------------------------------------------
                 |       Mean   Std. err.     [95% conf. interval]
-----------------+------------------------------------------------
c.accident@lclass|
               1 |   .7144964   .0369935      .6415438    .7874491
               2 |   .9934973   .0135709      .9659633    1.021031
------------------------------------------------------------------
Note: Numbers of observations in e(_N) vary among imputations.
r; t=2.40 14:46:04
. set rmsg off
```

The name of the latent class variable (here, `lclass`) and the number of imputations are required. The seed is optional, but of course is strongly recommended for reproducibility of the results, as the underlying data are randomly simulated. The multiple imputation version is notably faster.

As one of many diagnostic outputs of MI, the increase in variances / standard errors due to imputations serves as an indication of how much of a problem would treating the singly imputed (e.g. modal probability) latent classes would have been. In the above output, the fraction of missing data (FMI) is 33% to 40%, and the relative variance increase (RVI) is the similar range from 39% to 45%. This means that the analysis with the deterministic (modal) imputation of the classes would have had standard errors that are about 20% too small.

```
. webuse gsem_lca1.dta, clear
(Latent class analysis)
. quietly gsem (accident play insurance stock <-), logit lclass(C 2)
. predict post_1, class(1) classposterior
. gen byte lclass_modal = 2 - (post_1 > 0.50)
. mean post_1 lclass_modal
Mean estimation                             Number of obs = 216

             |       Mean   Std. err.     [95% conf. interval]
-------------+------------------------------------------------
      post_1 |   .7207539   .0257112      .6700756    .7714321
lclass_modal |   1.328704   .0320361      1.265559    1.391849

. mean accident, over(lclass_modal)
Mean estimation                                     Number of obs = 216

                         |       Mean   Std. err.     [95% conf. interval]
-------------------------+------------------------------------------------
c.accident@lclass_modal  |
                       1 |   .6896552   .0385529      .6136651    .7656452
                       2 |          1          0             .           .
```

3.2 NHANES complex survey data example

In many important and realistic applications of LCA, including the case that necessitated the development of this package, the data come from complex survey designs that require setting the data up for the appropriate survey-design adjusted analyses. See [SVY] **svyset**, [MI] **mi svyset**, and Kolenikov and Pitblado (2014).

The standard data set for the [SVY] commands is an extract from the National Health and Nutrition Examination Survey, Round Two (NHANES II) data. I will use a handful of binary health outcomes and one ordinal outcome to demonstrate LCA; the ordinal outcome is arguably an extension that is not quite well covered in the "classical" social science LCA.

```
. frame change default
. cap frame nhanes2: clear
. cap frame drop nhanes2
. frame create nhanes2
. frame change nhanes2
.
```

```
. webuse nhanes2.dta, clear
. svyset
Sampling weights: finalwgt
             VCE: linearized
     Single unit: missing
        Strata 1: strata
 Sampling unit 1: psu
           FPC 1: <zero>
. svy , subpop(if hlthstat<8) : ///
>         gsem ///
>                 (heartatk diabetes highbp <-, logit) ///
>                 (hlthstat <-, ologit) ///
>         , lclass(C 2) nolog  startvalues(randomid, draws(5) seed(101))
(running gsem on estimation sample)
Survey: Generalized structural equation model
Number of strata = 31                            Number of obs   =      10,351
Number of PSUs   = 62                            Population size = 117,157,513
                                                 Subpop. no. obs =      10,335
                                                 Subpop. size    = 116,997,257
                                                 Design df       =          31
```

	Coefficient	Linearized std. err.	t	P>|t|	[95% conf.	interval]
1.C	(base outcome)					
2.C						
_cons	1.330043	.1259401	10.56	0.000	1.073186	1.586899

```
Class:    1
Response: heartatk                                      Number of obs = 10,335
Family:   Bernoulli
Link:     Logit
Response: diabetes                                      Number of obs = 10,335
Family:   Bernoulli
Link:     Logit
Response: highbp                                        Number of obs = 10,335
Family:   Bernoulli
Link:     Logit
Response: hlthstat                                      Number of obs = 10,335
Family:   Ordinal
Link:     Logit
```

	Coefficient	Linearized std. err.	t	P>|t|	[95% conf.	interval]
heartatk						
_cons	-1.874967	.1150791	-16.29	0.000	-2.109672	-1.640261
diabetes						
_cons	-1.785271	.0805057	-22.18	0.000	-1.949463	-1.621078
highbp						
_cons	.4244921	.076861	5.52	0.000	.2677332	.5812511
/hlthstat						

```
        cut1 |  -3.659014   .8903346                      -5.474863   -1.843165
        cut2 |  -2.272516   .4402984                       -3.17051   -1.374521
        cut3 |  -.2566588   .2032721                      -.671235    .1579173
        cut4 |   1.229244   .1951641                      .8312038    1.627283
--------------------------------------------------------------------------------

Class:    2

Response: heartatk                                  Number of obs = 10,335
Family:   Bernoulli
Link:     Logit

Response: diabetes                                  Number of obs = 10,335
Family:   Bernoulli
Link:     Logit

Response: highbp                                    Number of obs = 10,335
Family:   Bernoulli
Link:     Logit

Response: hlthstat                                  Number of obs = 10,335
Family:   Ordinal
Link:     Logit

--------------------------------------------------------------------------------
             |             Linearized
             | Coefficient  std. err.      t    P>|t|     [95% conf. interval]
-------------+------------------------------------------------------------------
heartatk     |
       _cons |  -6.081307   .6280801    -9.68   0.000    -7.362285   -4.800329
-------------+------------------------------------------------------------------
diabetes     |
       _cons |  -5.223215   .6044468    -8.64   0.000    -6.455993   -3.990438
-------------+------------------------------------------------------------------
highbp       |
       _cons |  -.8166105   .0750027   -10.89   0.000    -.9695795   -.6636415
-------------+------------------------------------------------------------------
/hlthstat    |
        cut1 |   -.657824   .0483113                     -.7563555   -.5592926
        cut2 |   .7123144   .0649814                      .5797839    .8448448
        cut3 |   2.647239   .1192958                      2.403934    2.890544
        cut4 |   24.64389    14.1421                     -4.199113    53.48689
--------------------------------------------------------------------------------

. set rmsg on
r; t=0.00 14:46:16

. estat lcprob

Latent class marginal probabilities

Number of strata = 31                          Number of obs   =      10,351
Number of PSUs   = 62                          Population size = 117,157,513
                                               Design df       =          31

--------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.     [95% conf. interval]
-------------+------------------------------------------------
           C |
           1 |   .2091523   .0208315      .1698206    .2547976
           2 |   .7908477   .0208315      .7452024    .8301794
--------------------------------------------------------------
r; t=10.03 14:46:26

. estat lcmean

Latent class marginal means
```

```
Number of strata = 31                          Number of obs   =      10,351
Number of PSUs   = 62                          Population size = 117,157,513
                                               Design df       =          31

--------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.     [95% conf. interval]
-------------+------------------------------------------------
1            |
    heartatk |   .1329681   .0132672      .1081603    .1624295
    diabetes |   .1436535   .0099036      .1246119    .1650562
      highbp |   .6045577    .018375      .5665363    .6413552
             |
    hlthstat |
  Excellent  |   .0251111   .0217959      .0041733    .1366775
  Very good  |   .0683138    .021455      .0355579     .127263
       Good  |   .3427603   .0254834      .2928195    .3964437
       Fair  |   .3375009   .0210993      .2958981    .3817814
       Poor  |   .2263139   .0341724      .1642029    .3033906
-------------+------------------------------------------------
2            |
    heartatk |     .00228   .0014287      .0006343    .0081599
    diabetes |   .0053611   .0032231      .0015686    .0181559
      highbp |   .3064836   .0159419      .2749643    .3399221
             |
    hlthstat |
  Excellent  |   .3412286     .01086       .319438    .3637112
  Very good  |   .3296838   .0082507      .3130796     .346724
       Good  |   .2629283   .0094265      .2441597    .2826002
       Fair  |   .0661594   .0073704       .052623    .0828732
       Poor  |   1.98e-11   2.81e-10      5.68e-24    .9857545
--------------------------------------------------------------
r; t=80.58 14:47:47

. set rmsg off
```

This analysis approximates breaking down the population into "generally healthy" and "unhealthy" groups, as e.g. the gradient of *hlthstat* variable between the classes shows. The official `gsem postestimation` commands take approximately forever to run (there is underlying `margins` implementation with iterations over the numeric derivatives step size used to compute the stadnard errors). There is an interaction of `svy` and `gsem` in that `svy` forces its own starting values that happen to be infeasible for LCA, hence the need to specify the initial random search. The use of the `postlca_class_predpute` command makes it possible to run the analylsis much faster, and to conduct complementary analyses, e.g. analysis of the racial composition of the two classes.

```
. set rmsg on
r; t=0.00 14:47:47

. postlca_class_predpute, lcimpute(lclass) addm(10) seed(5678)
(10,351 missing values generated)
(10 imputations added; M = 10)

Sampling weights: finalwgt
             VCE: linearized
     Single unit: missing
        Strata 1: strata
 Sampling unit 1: psu
```

```
           FPC 1: <zero>
r; t=0.22 14:47:47

. mi estimate : prop lclass

Multiple-imputation estimates     Imputations     =         10
Proportion estimation             Number of obs   =     10,351
                                  Average RVI     =     0.5318
                                  Largest FMI     =     0.3641
                                  Complete DF     =      10350
DF adjustment:   Small sample     DF:     min     =      73.85
                                          avg     =      73.85
Within VCE type:     Analytic             max     =      73.85

─────────────┬──────────────────────────────────────────────
             │                                   Normal
             │ Proportion   Std. err.     [95% conf. interval]
─────────────┼──────────────────────────────────────────────
      lclass │
           1 │   .2675394   .0053851       .256809    .2782698
           2 │   .7324606   .0053851      .7217302     .743191
─────────────┴──────────────────────────────────────────────
r; t=1.53 14:47:49

. mi estimate : prop hlthstat if hlthstat < 8, over(lclass)

Multiple-imputation estimates     Imputations     =         10
Proportion estimation             Number of obs   =     10,335
                                  Average RVI     =          .
                                  Largest FMI     =          .
                                  Complete DF     =      10334
DF adjustment:   Small sample     DF:     min     =      36.31
                                          avg     =          .
Within VCE type:     Analytic             max     =          .

────────────────┬──────────────────────────────────────────────
                │                                   Normal
                │ Proportion   Std. err.     [95% conf. interval]
────────────────┼──────────────────────────────────────────────
hlthstat@lclass │
   Excellent 1  │   .0196036   .0034636      .0126501    .0265572
   Excellent 2  │   .3104144   .0055292      .2995676    .3212613
   Very good 1  │   .0550625   .0061217      .0426506    .0674743
   Very good 2  │   .3217978   .0056545      .3106989    .3328966
        Good 1  │   .2971359   .0106787      .2758813    .3183905
        Good 2  │   .2795891   .0056354      .2685025    .2906756
        Fair 1  │   .3635286   .0106978      .3423564    .3847008
        Fair 2  │   .0881987    .003944      .0803607    .0960367
        Poor 1  │   .2646694   .0089821      .2470257     .282313
        Poor 2  │          0  (no observations)
────────────────┴──────────────────────────────────────────────
Note: Numbers of observations in e(_N) vary among imputations.
r; t=3.01 14:47:52

. mi estimate : prop race, over(lclass)

Multiple-imputation estimates     Imputations     =         10
Proportion estimation             Number of obs   =     10,351
                                  Average RVI     =     0.4677
                                  Largest FMI     =     0.4671
                                  Complete DF     =      10350
DF adjustment:   Small sample     DF:     min     =      45.27
                                          avg     =     119.43
Within VCE type:     Analytic             max     =     321.80

─────────────┬──────────────────────────────────────────────
             │
```

```
             |                                     Normal
             |  Proportion   Std. err.     [95% conf. interval]
-------------+--------------------------------------------------
race@lclass  |
     White 1 |     .839782   .0093485      .8209563    .8586078
     White 2 |    .8889046   .0042799      .8804196    .8973897
     Black 1 |    .1434614   .0085447      .1263568     .160566
     Black 2 |    .0908358   .0038489      .0832192    .0984524
     Other 1 |    .0167566   .0031195      .0105146    .0229986
     Other 2 |    .0202596   .0017697      .0167778    .0237413
----------------------------------------------------------------
Note: Numbers of observations in e(_N) vary among imputations.
r; t=2.98 14:47:55

. set rmsg off
```

3.3 Choosing the number of imputations

One "researcher's degrees of freedom" aspect of this analysis is the number of imputations M that need to be created. What this number affects the most is the stability of the standard errors obtained through the multiple imputation process. This stability is internally assessed with estimated degrees of freedom associated with the variance estimate (Barnard and Rubin 1999). With $M = 10$ imputations, the smaller "poor health" class have about 50 degrees of freedom:

```
. mi estimate : prop race, over(lclass)

Multiple-imputation estimates     Imputations     =         10
Proportion estimation             Number of obs   =     10,351
                                  Average RVI     =     0.4677
                                  Largest FMI     =     0.4671
                                  Complete DF     =      10350
DF adjustment:   Small sample     DF:     min     =      45.27
                                          avg     =     119.43
Within VCE type:     Analytic             max     =     321.80
----------------------------------------------------------------
             |                                     Normal
             |  Proportion   Std. err.     [95% conf. interval]
-------------+--------------------------------------------------
race@lclass  |
     White 1 |     .839782   .0093485      .8209563    .8586078
     White 2 |    .8889046   .0042799      .8804196    .8973897
     Black 1 |    .1434614   .0085447      .1263568     .160566
     Black 2 |    .0908358   .0038489      .0832192    .0984524
     Other 1 |    .0167566   .0031195      .0105146    .0229986
     Other 2 |    .0202596   .0017697      .0167778    .0237413
----------------------------------------------------------------
Note: Numbers of observations in e(_N) vary among imputations.

. mi estimate, dftable

Multiple-imputation estimates     Imputations     =         10
Proportion estimation             Number of obs   =     10,351
                                  Average RVI     =     0.4677
                                  Largest FMI     =     0.4671
                                  Complete DF     =      10350
DF adjustment:   Small sample     DF:     min     =      45.27
                                          avg     =     119.43
```

```
Within VCE type:     Analytic            max     =     321.80

─────────────┬──────────────────────────────────────────────
             │                                       Normal
             │  Proportion   Std. err.          df   std. err.
─────────────┼──────────────────────────────────────────────
 race@lclass │
     White 1 │     .839782   .0093485         45.3      34.13
     White 2 │    .8889046   .0042799        106.2      18.59
     Black 1 │    .1434614   .0085447         57.9      28.28
     Black 2 │    .0908358   .0038489        126.3      16.62
     Other 1 │    .0167566   .0031195         59.0      27.89
     Other 2 │    .0202596   .0017697        321.8       9.38
─────────────┴──────────────────────────────────────────────
Note: Numbers of observations in e(_N) vary among imputations.
```

The public use version of the NHANES II data uses the approximate design that has 62 PSU in 31 strata, resulting in 31 design degrees of freedom. The imputation degrees of freedom barely exceed that. Let us push the number of imputations up:

```
. webuse nhanes2.dta, clear
. qui svy , subpop(if hlthstat<8) : ///
>         gsem ///
>                 (heartatk diabetes highbp <-, logit) ///
>                 (hlthstat <-, ologit) ///
>         , lclass(C 2) nolog  startvalues(randomid, draws(5) seed(101))
. postlca_class_predpute, lcimpute(lclass) addm(62) seed(9752)
(10,351 missing values generated)
(62 imputations added; M = 62)
Sampling weights: finalwgt
             VCE: linearized
     Single unit: missing
        Strata 1: strata
 Sampling unit 1: psu
           FPC 1: <zero>
. mi estimate : prop race, over(lclass)
Multiple-imputation estimates     Imputations     =         62
Proportion estimation             Number of obs   =     10,351
                                  Average RVI     =     0.2483
                                  Largest FMI     =     0.2893
                                  Complete DF     =      10350
DF adjustment:   Small sample     DF:     min     =     671.70
                                          avg     =   1,786.62
Within VCE type:     Analytic             max     =   3,092.35

─────────────┬──────────────────────────────────────────────
             │                                  Normal
             │  Proportion   Std. err.     [95% conf. interval]
─────────────┼──────────────────────────────────────────────
 race@lclass │
     White 1 │    .8369206   .0079313      .8213584    .8524827
     White 2 │    .8900061   .0038498      .8824571     .897555
     Black 1 │    .1455849    .007628      .1306162    .1605536
     Black 2 │    .0900031   .0035541      .0830334    .0969728
     Other 1 │    .0174946   .0029465      .0117091      .02328
     Other 2 │    .0199908    .001709        .01664    .0233417
─────────────┴──────────────────────────────────────────────
Note: Numbers of observations in e(_N) vary among imputations.
```

```
. mi estimate, dftable

Multiple-imputation estimates     Imputations     =         62
Proportion estimation             Number of obs   =     10,351
                                  Average RVI     =     0.2483
                                  Largest FMI     =     0.2893
                                  Complete DF     =      10350
DF adjustment:   Small sample     DF:     min     =     671.70
                                          avg     =   1,786.62
Within VCE type:     Analytic             max     =   3,092.35

--------------------------------------------------------------
             |                                          Normal
             |  Proportion   Std. err.          df   std. err.
-------------+------------------------------------------------
race@lclass  |
     White 1 |    .8369206    .0079313      1100.3       13.14
     White 2 |    .8900061    .0038498      2639.5        7.08
     Black 1 |    .1455849     .007628      1004.8       13.98
     Black 2 |    .0900031    .0035541      2211.1        8.08
     Other 1 |    .0174946    .0029465       671.7       18.45
     Other 2 |    .0199908     .001709      3092.4        6.26
--------------------------------------------------------------
Note: Numbers of observations in e(_N) vary among imputations.
```

The MI degrees of freedom are now comfortably above 600. In many i.i.d. data situations, increasing the number of imputations to several dozens can often send the MI degrees of freedom to approximate infinity (the reported numbers are in hundreds of thousands). With complex survey designs that have limited degrees of freedom within each implicate, this may not materialize. Researchers are encouraged to adopt the workflow where, in parallel, they

1. start with a small number of imputations, like `addm(10)` in the example above, and develop the analysis code for all the substantive analyses, and
2. working with the key outcomes or analyses, experiment with several values of M to find a reasonable trade-off when degrees of freedom exceed the sample size for i.i.d. data, and/or exceed the design degrees of freedom for complex survey data by a factor of 3–5.

Then a chosen value of M can be used for all analyses in the paper.

Even a large number of replications may not protect the researcher from classes that may have structural zeroes. These produce zero standard errors and missing degrees of freedom and variance increase statistics:

```
.
. mi estimate , dftable : prop hlthstat if hlthstat < 8, over(lclass)

Multiple-imputation estimates     Imputations     =         62
Proportion estimation             Number of obs   =     10,335
                                  Average RVI     =          .
                                  Largest FMI     =          .
                                  Complete DF     =      10334
DF adjustment:   Small sample     DF:     min     =     234.37
                                          avg     =          .
```

```
Within VCE type:     Analytic            max     =          .
```

	Proportion	Std. err.	df	Normal std. err.
hlthstat@lclass				
Excellent 1	.0183993	.0033919	309.7	32.68
Excellent 2	.3111818	.0054853	6242.4	3.09
Very good 1	.0573944	.0062471	234.4	41.21
Very good 2	.3212476	.0056361	4001.1	5.03
Good 1	.2947403	.0104573	575.3	20.57
Good 2	.2804536	.0055871	2155.5	8.23
Fair 1	.3656239	.010418	1033.8	13.71
Fair 2	.087117	.00393	549.1	21.27
Poor 1	.2638421	.0087535	4603.6	4.40
Poor 2	0	(no observations)		

```
Note: Numbers of observations in e(_N) vary among imputations.
.
```

4 User's guide to sj.sty

The *Stata Journal* is produced using `statapress.cls` and `sj.sty`, a LaTeX 2_ε document class and package, respectively, each developed and maintained at StataCorp by the Stata Press staff. These files manage the look and feel of each article in the *Stata Journal*.

4.1 The title page

Each insert must begin with title-generating commands. For example,

```
\inserttype[st0001]{article}
\author{short author list}{%
  First author\\First affiliation\\City, State/Country\\Email address
  \and
  Second author\\Second affiliation\\City, State/Country\\Email address
}
\title[short toc title]{Long title for first page of journal insert}
\sjSetDOI{!!}
\maketitle
```

Here `\inserttype` identifies the tag (for example, st0001) associated with the journal insert and the insert type (for example, article). The default `\inserttype` is "notag", possibly with a number appended. `\author` identifies the short and long versions of the list of authors (that is, J. M. Doe for the short title and John Michael Doe for the long). The short author list is only the author initial(s) and last name, and the long author list is the author initial(s) and last name, author affiliation(s), and city and state or country (spelled out with accents applied as necessary). An email address should be included for, at least, the corresponding author. `\title` identifies the short (optional) and long (required) versions of the title of the journal insert. The optional argument to `\title` is used as the even-numbered page header. If the optional argument to `\title` is not given, the long title is used. The required argument to `\title` is placed in the table of contents with the short author list. Titles should not have any font changes or TeX macros in them. `\sjSetDOI{!!}` is filled in by Stata Press with a DOI. `\maketitle` must be the last command of this sequence; it uses the information given in the previous commands to generate the title for a new journal insert.

4.2 The abstract

The abstract is generated using the `abstract` environment. The abstract states the purpose of the article and area of research. Abstracts must be able to stand alone from the full-text article. For this reason, fully cite references rather than merely supplying the author and date. Also, avoid introduction of acronyms in the abstract. The `\keywords` are also appended to the abstract. Here is an example abstract with keywords:

```
\begin{abstract}
This is an example article.  You should change the \input{} line in
\texttt{main.tex} to point to your file.  If this is your first submission to
the {\sl Stata Journal}, please read the following ‘‘getting started’’
```

```
information.

\keywords{\inserttag, command name(s), keyword(s)}
\end{abstract}
```

`\inserttag` will be replaced automatically with the tag given in `\inserttype` (here st0001). The first keyword will be the article tag (assigned by Stata Press); other keywords for indexing purposes should be added by the author(s). Community-contributed command names should be listed after the article tag. Plural terms and multiple concepts should be avoided.

4.3 Sectioning

All sections are generated using the standard LaTeX sectioning commands:
`\section`, `\subsection`, ...

Sections in articles are numbered. If the optional short section title is given, it will be put into bookmarks for the electronic version of the journal; otherwise, the long section title is used. Like article titles, section titles should not have any font changes or TeX macros in them.

4.4 The bib option

BibTeX is a program that formats citations and references according to a bibliographic style. The following two commands load the bibliographic style file for the *Stata Journal* (`sj.bst`) and open the database of bibliographic entries (`sj.bib`):

```
\bibliographystyle{sj}
\bibliography{sj}
```

Here are some example citations: Akaike (1973), Ben-Akiva and Lerman (1985), Dyke and Patterson (1952), Greene (2003), Kendall and Stuart (1979), Hilbe (1993a), Hilbe (1994), Hilbe (1993b), Maddala (1983), and Goossens, Mittelbach, and Samarin (1994). They are generated by using the `\citet` and `\citet*` commands from the `natbib` package. Here we test `\citeb` and `\citebetal`: Akaike [1973], Ben-Akiva and Lerman [1985], Dyke and Patterson [1952], Greene [2003], Kendall and Stuart [1979], Hilbe [1993a], Hilbe [1994], Hilbe [1993b], Maddala [1983], and Goossens, Mittelbach, and Samarin [1994]. Sometimes using the `\cite` macros will result in an overfull line as shown above. The solution is to list the author names and the citation year separately, for example, `Ben-Akiva and Lerman [\citeyear{benAkivaLerman}]`.

The `bib` option of `statapress.sty` indicates that citations and references will be formatted using BibTeX and the `natbib` package. This option is the default (meaning that it need not be supplied), but there is no harm in supplying it to the `statapress` document class in the main LaTeX driver file (for example, `main.tex`).

```
\documentclass[bib]{sj}
```

If you choose not to use BibTeX, you can use the `nobib` option of `statapress.sty`.

```
\documentclass[nobib]{statapress}
```

BibTeX and bibliographic styles are described in Goossens, Mittelbach, and Samarin (1994).

4.5 Author information

The *About the authors* section is generated by using the `aboutauthors` environment. There is also an `aboutauthor` environment for journal inserts by one author. For example,

```
\begin{aboutauthor}

Text giving background about the author goes in here.

\end{aboutauthor}
```

5 User's guide to stata.sty

`stata.sty` is a LaTeX package containing macros and environments to help authors produce documents containing Stata output and syntax diagrams.

5.1 Citing the Stata manuals

The macros for generating references to the Stata manuals are given in table 1.

Table 1: Stata manual references

Example	Result
`\bayesref{bayes}`	[BAYES] **bayes**
`\cmref{cmchoiceset}`	[CM] **cmchoiceset**
`\dref{Data types}`	[D] **Data types**
`\dsgeref{dsge}`	[DSGE] **dsge**
`\ermref{eregress}`	[ERM] **eregress**
`\fnref{Statistical functions}`	[FN] **Statistical functions**
`\fmmref{fmm:~betareg}`	[FMM] **fmm: betareg**
`\grefa{Graph Editor}`	[G-1] **Graph Editor**
`\grefb{graph}`	[G-2] **graph**
`\grefci{line\_options}`	[G-3] ***line_options***
`\grefdi{connectstyle}`	[G-4] ***connectstyle***
`\gsref{6~Using the Data Editor}`	[GS] **6 Using the Data Editor**
`\irtref{irt}`	[IRT] **irt**
`\lassoref{Lasso intro}`	[LASSO] **Lasso intro**
`\metaref{meta}`	[META] **meta**
`\meref{me}`	[ME] **me**
`\mreff{Intro}`	[M-0] **Intro**
`\mrefa{Ado}`	[M-1] **Ado**
`\mrefb{Declarations}`	[M-2] **Declarations**
`\mrefc{mata clear}`	[M-3] **mata clear**
`\mrefd{Matrix}`	[M-4] **Matrix**
`\mrefe{st\_view($\,$)}`	[M-5] **st_view( )**
`\mrefg{Glossary}`	[M-6] **Glossary**
`\miref{mi impute}`	[MI] **mi impute**
`\mvref{cluster}`	[MV] **cluster**
`\pref{syntax}`	[P] **syntax**
`\pssrefa{Intro}`	[PSS-1] **Intro**
`\pssrefb{power}`	[PSS-2] **power**
`\pssrefc{ciwidth}`	[PSS-3] **ciwidth**
`\pssrefd{Unbalanced designs}`	[PSS-4] **Unbalanced designs**
`\pssrefe{Glossary}`	[PSS-5] **Glossary**
`\pssref{Subject and author index}`	[PSS] **Subject and author index**
`\rptref{Dynamic documents intro}`	[RPT] **Dynamic documents intro**
`\rref{regress}`	[R] **regress**
`\spref{Intro}`	[SP] **Intro**
`\stref{streg}`	[ST] **streg**
`\svyref{svy:~tabulate oneway}`	[SVY] **svy: tabulate oneway**
`\tsref{arima}`	[TS] **arima**
`\uref{1~Read this---it will help}`	[U] **1 Read this—it will help**
`\xtref{xtreg}`	[XT] **xtreg**

5.2 Stata syntax

Here is an example syntax display:

regress *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [, noconstant hascons tsscons vce(*vcetype*) level(#) beta eform(*string*) depname(*varname*) *display_options* noheader notable plus mse1 coeflegend]

This syntax is generated by

```
\begin{stsyntax}
\dunderbar{reg}ress
    \depvar\
    \optindepvars\
    \optif\
    \optin\
    \optweight\
    \optional{,
    \underbar{nocons}tant
    \underbar{h}ascons
    tsscons
    vce({\it vcetype\/})
    \underbar{l}evel(\num)
    \underbar{b}eta
    \underbar{ef}orm(\ststring)
    \dunderbar{dep}name(\varname)
    {\it display\_options}
    \underbar{nohe}ader
    \underbar{notab}le
    plus
    \underbar{ms}e1
    \underbar{coefl}egend}
\end{stsyntax}
```

Each command should be formatted using a separate `stsyntax` environment. Table 2 contains an example of each syntax macro provided in `stata.sty`.

Table 2: Stata syntax elements

Macro	Result	Macro	Result
`\LB`	[	`\ifexp`	`if`
`\RB`	]	`\optif`	[*if*]
`\varname`	*varname*	`\inrange`	`in`
`\optvarname`	[*varname*]	`\optin`	[*in*]
`\varlist`	*varlist*	`\eqexp`	`=`*exp*
`\optvarlist`	[*varlist*]	`\opteqexp`	[`=`*exp*]
`\newvarname`	*newvar*	`\byvarlist`	`by` *varlist*`:`
`\optnewvarname`	[*newvar*]	`\optby`	[`by` *varlist*`:`]
`\newvarlist`	*newvarlist*	`\optional{text}`	[`text`]
`\optnewvarlist`	[*newvarlist*]	`\optweight`	[*weight*]
`\depvar`	*depvar*	`\num`	#
`\optindepvars`	[*indepvars*]	`\ststring`	*string*
`\opttype`	[*type*]		

`\underbar` is a standard macro that generates underlines. The `\dunderbar` macro from `stata.sty` generates the underlines for words with descenders. For example,

- `{\tt \underbar{reg}ress}` generates `regress`
- `{\tt \dunderbar{reg}ress}` generates `regress`

The plain TeX macros `\it`, `\sl`, and `\tt` are also available. `\it` should be used to denote "replaceable" words, such as *varname*. `\sl` can be used for emphasis but should not be overused. `\tt` should be used to denote words that are to be typed, such as command names.

When describing the options of a new command, the `\hangpara` and `\morehang` commands provide a means to reproduce a paragraph style similar to that of the Stata reference manuals. For example,

`level(`#`)` specifies the confidence level, as a percentage, for confidence intervals. The default is `level(95)` or as set by `set level`; see [U] **20.8 Specifying the width of confidence intervals**.

was generated by

```
\hangpara
{\tt level(\num)} specifies the confidence level, as a percentage,
for confidence intervals.  The default is {\tt level(95)} or as set by {\tt
set level}; see \uref{20.8~Specifying the width of confidence intervals}.
```

5.3 Stata output

When submitting *Stata Journal* articles that contain Stata output, also submit a do-file and all relevant datasets that reproduce the output (do not forget to set the random-number seed when doing simulations). Results should be reproducible. Begin examples by loading the data. Code should be written to respect a linesize of 80 characters. The following is an example of the `stlog` environment containing output from simple linear regression analysis on two variables in `auto.dta`:

```
. sysuse auto
(1978 Automobile Data)

. regress mpg weight

      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  1,    72) =  134.62
       Model |   1591.9902     1   1591.9902           Prob > F      =  0.0000
    Residual |  851.469256    72  11.8259619           R-squared     =  0.6515
-------------+------------------------------           Adj R-squared =  0.6467
       Total |  2443.45946    73  33.4720474           Root MSE      =  3.4389

------------------------------------------------------------------------------
         mpg |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |  -.0060087   .0005179   -11.60   0.000    -.0070411   -.0049763
       _cons |   39.44028   1.614003    24.44   0.000     36.22283    42.65774
------------------------------------------------------------------------------
```

The above listing was included using

```
\begin{stlog}
\input{output1.log.tex}\nullskip
\end{stlog}
```

where `output1.log.tex` is a Stata log file converted to include TEX macros by using the `sjlog` command (more on `sjlog` shortly). `\nullskip` adjusts the spacing around the log file.

On occasion, it is convenient (maybe even necessary) to be able to omit some of the output or let it spill onto the next page. Here is a listing containing the details of the following discussion:

```
\begin{stlog}
. sysuse auto
(1978 Automobile Data)
{\smallskip}
. regress mpg weight
{\smallskip}
\oom
{\smallskip}
\clearpage
\end{stlog}
```

The `\oom` macro creates a short message indicating omitted output in the following example, and the `\clearpage` macro inserts a page break.

```
. sysuse auto
(1978 Automobile Data)
. regress mpg weight
```

(*output omitted*)

The output in `output1.log.tex` was generated from the following `output.do`:

```
* output.do
set more off
capture log close
sjlog using output1, replace
sysuse auto
regress mpg weight
sjlog close, replace
sort weight
predict yhat
set scheme sj
scatter mpg yhat weight, c(. l) s(x i)
graph export output1.eps, replace
exit
```

`output.do` generates a `.smcl` file, `.log` file, and `.log.tex` file using `sjlog`. The actual file used in the above listing was generated by

```
. sjlog type output.do
```

`sjlog.ado` is provided in the Stata package for `sjlatex`. `sjlog` is a Stata command that helps generate log output to be included in LaTeX documents using the `stlog` environment. If you have installed the `sjlatex` package, see the help file for `sjlog` for more details. The lines that make up the table output from `regress` are generated from line-drawing macros defined in `stata.sty`; these were macros written using some font metrics defined in Knuth (1986).

By default, `stlog` sets an 8-point font for the log. Use the `auto` option to turn this behavior off, allowing you to use the current font size, or change it by using `\fontsize{#}{#}\selectfont`. The call to `stlog` with the `auto` option looks like `\begin[auto]{stlog}`.

Here is an example where we are using a 12-point font.

```
. sjlog type output.do
```

5.4 About tables

Tables should be created using the standard LaTeX methods. See Lamport (1994) for a discussion and examples. Tables should be included in the main text rather than at the end of the document. Tables should be called out in the text prior to appearance.

There are many user-written commands that produce LaTeX output, including tables. Christopher F. Baum has written `outtable`, a Stata command for creating LaTeX tables from Stata matrices. Ben Jann's well-known `estout` command can also produce LaTeX output. To find other user-written commands that produce LaTeX output, try

```
. net search latex
```

Tables with notes

Table 3 shows the order and format to use for notes to tables.

Table 3: Industrial clusters

China		United States	
Core of cluster	Size (in # of units)	Core of cluster	Size (in # of units)
Construction	28[a]	Public administration and defense; compulsory social security	30[b]
Food, beverages, and tobacco	3	Food, beverages, and tobacco	2
Textiles and textile products	2	Chemicals and chemical products	1
Chemicals and chemical products	1	Basic metals and fabricated metal	1
Transport equipment	1	Transport equipment	1
$L_a = 0.602$***		$L_a = 0.567$	
$L_w = 0.828$**		$L_w = 0.837$	
$L_m = 0.335$*		$L_m = 0.287$	
$K^* = 5$		$K^* = 5$	
$K = 35$		$K = 35$	

SOURCE: Pew Research Center.
NOTE: U.S. industrial clusters based on U.S. input–output flows of goods expressed in millions of dollars between 35 ISIC industries from the WIOD data. The minimum number of clusters `k()` was set equal to five. The algorithm returns L_a, L_w, and L_m, which refer to the average of the internal relative flows, the population-weighted average of the internal relative flows, and the minimum of the internal relative flows, respectively. K^* and K refer to the number of defined regional clusters and the number of distinct starting units, respectively.
[a] This note pertains only to row 1 column 2.
[b] This note pertains only to row 1 column 4.
*** denotes $p < 0.01$; ** denotes $p < 0.05$; * denotes $p < 0.1$.

Order of notes should be

1. source notes
2. notes applying to the whole table
3. notes applying to specific parts of the table

4. notes on significance levels

Special notes:

- Use `\centering` because the `center` environment adds unnecessary vertical spacing.
- Place the `\begin{threeparttable}` line above the caption.

Tables should be included in the main text rather than at the end of the document. Tables should be called out in the text prior to appearance.

5.5 Encapsulated PostScript (EPS)

You can include figures by using either `\includegraphics` or `\epsfig`.

```
\begin{figure}[h!]
\begin{center}
\includegraphics{eps/output1.eps}
\end{center}
\caption{Scatterplot with simple linear regression line}
\label{fig}
\end{figure}
```

```
\begin{figure}[h!]
\begin{center}
\epsfig{file=output1}
\end{center}
\caption{Scatterplot with simple linear regression line}
\label{fig}
\end{figure}
```

Figure 1 is included using `\epsfig` from the `epsfig` package.

The graph was generated by running `output.do`, the do-file given in section 5.3. The `epsfig` package is described in Goossens, Mittelbach, and Samarin (1994).

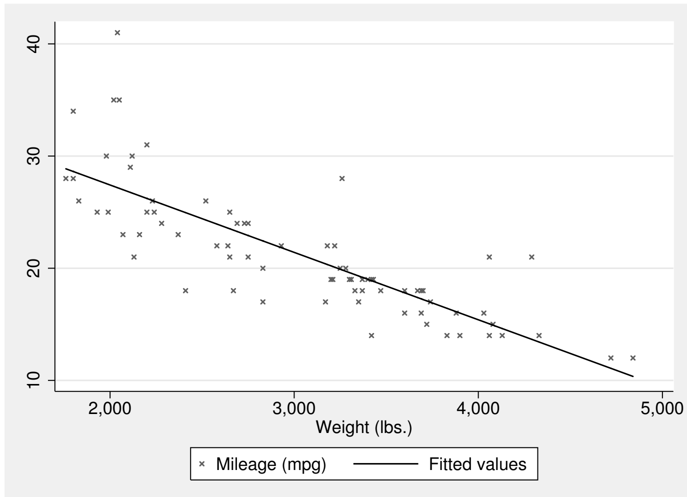

Figure 1: Scatterplot with simple linear regression line

EPS is the preferred format for graphs and line art. Figures should be included in the main text rather than at the end of the document and should be called out in the text prior to appearance. If your article is written in Word, you should submit your figures as separate EPS files. Rasterized-based files of at least 300 dpi (dots per inch) are acceptable. Avoid using bitmaps for figures and graphs, because even if images are outputted at 300 dpi, bitmaps can increase the size of the resulting file for printing. (However, bitmaps will be allowed for photographs, which are used in, for example, the *Stata Journal* Editors' prize announcement.) Images should be submitted in black and white (grayscale). We recommend that graphs created in Stata use the `sj` scheme.

5.6 Stored results

The `stresults` environment provides a table to describe the stored results of a Stata command. It consists of four columns: the first and third column are for Stata result identifiers (for example, `r(N)`, `e(cmd)`), and the second and fourth columns are for a brief description of the respective identifier. Each group of results is generated using the `\stresultsgroup` macro. The following is an example containing a brief description of the results that `regress` stored to `e()`:

Scalars

`e(N)`	number of observations	`e(F)`	F statistic
`e(mss)`	model sum of squares	`e(rmse)`	root mean squared error
`e(df_m)`	model degrees of freedom	`e(ll_r)`	log likelihood
`e(rss)`	residual sum of squares	`e(ll_r0)`	log likelihood, constant-only model
`e(df_r)`	residual degrees of freedom		
`e(r2)`	R-squared	`e(N_clust)`	number of clusters

Macros

`e(cmd)`	`regress`	`e(wexp)`	weight expression
`e(depvar)`	name of dependent variable	`e(clustvar)`	name of cluster variable
`e(model)`	`ols` or `iv`	`e(vcetype)`	title used to label Std. Err.
`e(wtype)`	weight type	`e(predict)`	program used to implement `predict`

Matrices

`e(b)`	coefficient vector	`e(V)`	variance–covariance matrix of the estimators

Functions

`e(sample)`	marks estimation sample

Alternatively, you can use the `stresults2` environment to create a two column table. This format works better if your descriptions are long.

5.7 Examples and notes

The following are environments for examples and notes similar to those given in the Stata reference manuals. They are generated using the `stexample` and `sttech` environments, respectively.

▷ Example

This is the default alignment for a Stata example.

◁

▷ Example

For this example, `\stexamplehskip` was set to `0.0pt` before beginning. This sentence is supposed to spill over to the next line, thus revealing that the first sentence was indented.

This sentence is supposed to show that new paragraphs are automatically indented (provided that `\parindent` is nonzero).

◁

❏ **Technical note**

For this note, `\sttechhskip` was set to `-13.90755pt` (the default) before beginning. This sentence is supposed to spill over to the next line, thus revealing that the first sentence was indented.

This sentence is supposed to show that new paragraphs are automatically indented (provided that `\parindent` is nonzero).

❏

5.8 Special characters

Table 4 contains macros that generate some useful characters in the typewriter (fixed width) font. The exceptions are `\stcaret` and `\sttilde`, which use the currently specified font; the strictly fixed-width versions are `\caret` and `\tytilde`, respectively.

Table 4: Special characters

Macro	Result	Macro	Result
`\stbackslash`	\	`\sttilde`	~
`\stforslash`	/	`\tytilde`	~
`\stcaret`	ˆ	`\lbr`	{
`\caret`	^	`\rbr`	}

5.9 Equations and formulas

In (3), $\overline{x}$ was generated using `\stbar{x}`. Here `\stbar` is equivalent to the TEX macro `\overline`.

$$E(\overline{x}) = \mu \tag{3}$$

In (4), $\widehat{\beta}$ was generated using `\sthat{\beta}`. Here `\sthat` is equivalent to the TEX macro `\widehat`.

$$V(\widehat{\beta}) = V\{(X'X)^{-1}X'y\} = (X'X)^{-1}X'V(y)X(X'X)^{-1} \tag{4}$$

Formulas should be defined and follow a concise style. Different disciplines adhere to different notation styles; however, if the notation cannot be clearly interpreted, you may be asked to make changes. The bolding and font selection guidelines are the following:

- Matrices are capitalized and bolded; for instance, $\mathbf{\Pi} + \mathbf{\Theta} + \mathbf{\Phi} - \mathbf{B}$.
- Vectors are lowercased and bolded; for instance, $\boldsymbol{\pi} + \boldsymbol{\theta} + \boldsymbol{\phi} - \mathbf{b}$.
- Scalars are lowercased and nonbolded; for instance, $r_2 + c_1 - c_2$.

Sentence punctuation should not be used in formulas set off from the text.

Formulas in line with the text should use the solidus (/) instead of a horizontal line for fractional terms.

Nesting of grouping is square brackets, curly braces, and then parentheses, or $[\{()\}]$.

Only those equations explicitly referred to in the text should be assigned an equation number.

6 References

Akaike, H. 1973. Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, ed. B. N. Petrov and F. Csaki, 267–281. Budapest, Hungary: Akademiai Kiado.

Barnard, J., and D. B. Rubin. 1999. Small-Sample Degrees of Freedom with Multiple Imputation. *Biometrika* 86: 948–955.

Ben-Akiva, M., and S. R. Lerman. 1985. *Discrete Choice Analysis: Theory and Application to Travel Demand*. Cambridge, MA: MIT Press.

Dyke, G. V., and H. D. Patterson. 1952. Analysis of factorial arrangements when the data are proportions. *Biometrics* 8: 1–12.

Goossens, M., F. Mittelbach, and A. Samarin. 1994. *The LaTeX Companion*. Reading, MA: Addison–Wesley.

Greene, W. H. 2003. *Econometric Analysis*. 5th ed. Upper Saddle River, NJ: Prentice Hall.

Hagenaars, J. A., and A. L. McCutcheon. 2002. *Applied latent class analysis*. Cambridge University Press.

Hilbe, J. 1993a. sg16: Generalized linear models. *Stata Technical Bulletin* 11: 20–28. Reprinted in *Stata Technical Bulletin Reprints*. Vol. 2, pp. 149–159. College Station, TX: Stata Press.

———. 1993b. Log Negative Binomial Regression as a Generalized Linear Model. *Graduate College Committee on Statistics* (Technical Report 26).

———. 1994. Generalized linear models. *American Statistician* 48: 255–265.

Kendall, M., and A. Stuart. 1979. *The Advanced Theory of Statistics*. Vol. 2. 4th ed. London: Griffin.

Knuth, D. E. 1986. *The TEX book*. Reading, MA: Addison–Wesley.

Kolenikov, S., and K. Daley. 2017. Latent Class Analysis of Worker Knowledge of Their Employment Status. 3898–3906. Alexandria, VA, USA. Available from www.asasrms.org/Proceedings/y2017/files/594102.pdf.

Kolenikov, S., and J. Pitblado. 2014. Analysis of complex health survey data. In *Handbook of Health Survey Methods*, ed. T. P. Johnson, chap. 29. Hoboken, NJ: Wiley.

Lamport, L. 1994. *LATEX: A Document Preparation System*. 2nd ed. Reading, MA: Addison–Wesley.

Maddala, G. S. 1983. *Limited-Dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press.

McCutcheon, A. L. 1987. *Latent class analysis*. No. 64 in Quantitative Applications in Social Sciences—"The Little Green Book", Sage.

van Buuren, S. 2018. *Flexible Imputation of Missing Data*. 2nd ed. Chapman & Hall/CRC.

About the authors

Stas Kolenikov is Principal Statistician at NORC who has been using Stata and writing Stata programs for about 25 years. He had worked on economic welfare and inequality, spatiotemporal environmental statistics, mixture models, missing data, multiple imputation, structural equations with latent variables, resampling methods, complex sampling designs, survey weights, Bayesian mixed models, combining probability and non-probability samples, latent class analysis, and likely some other stuff, too.