gretl working papers

Random effects estimators for unbalanced panel data: a Monte Carlo analysis

Allin Cottrell[1]

working paper #4

September 2017

[1]Department of Economics, Wake Forest University; email cottrell@wfu.edu.

1 Introduction

Random-effects estimators for panel-data models have, of course, been much discussed in the econometrics literature. However, relatively little attention has been devoted to the performance and possible optimization of such estimators in the (common) case of unbalanced panels, the primary exception being the work of Badi Baltagi and Young-Jae Chang (Baltagi and Chang, 1994; Baltagi, 2005). In this paper we revisit the work of Baltagi and Chang, which has not always been correctly interpreted. We confine ourselves to one-way random effects (by individual and not by time) and concentrate on two of the most popular methods, those of Swamy and Arora (1972) and Nerlove (1971). Our perspective is practical and software-oriented; we are interested in developing appropriate options, and appropriate default procedures, for random-effects estimation. Our focus is on the open-source econometrics software gretl (Cottrell and Lucchetti, 2017) but we also make reference to Stata and R.

We may write the random-effects model as

$$y_{it} = X_{it}\beta + v_i + \varepsilon_{it}$$

where X_{it} is a $1 \times K$ vector of regressors, v_i is the "individual effect" (with variance σ_v^2) and ε_{it} is the "idiosyncratic" error (with variance σ_ε^2). Since the composite error $v_i + \varepsilon_{it}$ is not i.i.d., we can define a GLS estimator which is more efficient than either pooled OLS or the fixed-effects estimator—provided that GLS is consistent, which requires that the regressors are uncorrelated with the individual effects. This estimator can be implemented via OLS on a dataset in which all variables are subject to quasi-demeaning, as in

$$\tilde{y}_{it} = y_{it} - \theta_i \bar{y}_i$$

where $\bar{y}_i$ indicates the mean of the observations on y for individual i and $0 \leq \theta_i \leq 1$. In balanced panels θ is in common across individuals and is given by

$$\theta = 1 - \left(\frac{\sigma_\varepsilon^2}{\sigma_\varepsilon^2 + T\sigma_v^2} \right)^{\frac{1}{2}}$$

In an unbalanced panel θ differs by individual, depending on the number of available observations per individual, T_i:

$$\theta_i = 1 - \left(\frac{\sigma_\varepsilon^2}{\sigma_\varepsilon^2 + T_i\sigma_v^2} \right)^{\frac{1}{2}}$$

This point in itself is straightforward and does not stand in need of elaboration; our concern is with estimation of the error variance components, σ_v^2 and σ_ε^2, the former in particular.

As is well known, the method of Swamy and Arora (1972) is to infer σ_v^2 from estimates of the Within and Between variances (obtained from the residual variance of the fixed-effects model and the Between model, respectively). Writing σ_ε^2 and σ_b^2 for the Within and Between variances respectively, the formula for a balanced panel is

$$\hat{\sigma}_v^2 = \hat{\sigma}_b^2 - \hat{\sigma}_\varepsilon^2 / T \tag{1}$$

where T is the (common) number of observations per individual. The method of Nerlove (1971) is to infer σ_v^2 from the variance of the fixed effects (that is, the estimated per-individual intercepts). Writing the fixed effects as α_i, the formula is

$$\hat{\sigma}_v^2 = \frac{1}{N-1} \sum_{i=1}^{N} (\hat{\alpha}_i - \bar{\hat{\alpha}})^2 \tag{2}$$

where N is the number of included individuals and $\bar{\hat{\alpha}}$ is the mean of the intercepts.

Both of these methods were originally developed with balanced samples (uniform T) in mind. Baltagi and Chang (1994) put forward a version of the Swamy–Arora method for unbalanced panels, further discussion of which can be found in Baltagi (2005). To our knowledge, nobody has published a corresponding unbalanced-panel version of Nerlove's method, though one can envisage such a thing by analogy with Baltagi and Chang's modification of Swamy–Arora.

The purpose of this paper is, first, to explicate the proposal of Baltagi and Chang (which is quite easily misinterpreted) and to test two possible readings of their proposal against the "standard" (minimally modified, as explained below) Swamy–Arora method in the context of unbalanced panels, and second, to construct and assess an analogous modification to Nerlove's method.

2 Understanding Baltagi–Chang

The equation given by Baltagi and Chang for an unbalanced-panel version of Swamy–Arora is (using our notation in the subscripts of the variance components) as follows:

$$\hat{\sigma}_v^2 = \frac{\hat{u}^{b\prime} P \hat{u}^b - (N-K)\hat{\sigma}_\varepsilon^2}{n - \mathrm{tr}((X'PX)^{-1}X'ZZ'X)} \tag{3}$$

The matrix P is block-diagonal, the blocks being matrices of dimension T_i each of whose elements are $1/T_i$ (so that multiplication by this matrix has the effect of obtaining T_i copies of the individual mean of the multiplicand). N is as defined above, and K is the number of regressors including the intercept.

The matrix whose trace figures in the denominator in (3) need not concern us here; its definition is clear in context.[1] The term that calls for clarification (in that it may be liable to misinterpretation) is the leftmost element in the numerator, namely $\hat{u}^{b\prime} P \hat{u}^b$. Baltagi and Chang mean by $\hat{u}^b$ the residuals from the Between regression, but it is important to note that in their presentation this regression uses $n = \sum_{i=1}^{N} T_i$ observations, *not* just N (the number of included individuals) as is common in panel-data software. On page 70 of their article they state that the Between estimator is obtained as

$$\hat{\beta}_\mathrm{b} = (X'PX)^{-1}X'Py \tag{4}$$

Here X and y are of full length n, and the $n \times n$ matrix P is as described above.[2] Since the individual-mean values each appear T_i times in this regression, it is numerically identical to weighted least squares, with weights T_i, on a dataset in which the individual-mean values appear just once. This explains their comment (p. 73) that "$\hat{u}^{b\prime} P \hat{u}^b$ can be obtained as the OLS residual sum of squares from the regression involving $\sqrt{T_i}\bar{y}_{i.}$ on $\sqrt{T_i}\bar{X}_{i.}$" If we were to think of the residuals $\hat{u}^b$ as derived from a regression using N observations[3] this comment would be wrong (or correct only in the balanced case); but if the Between regression is defined as in (4) it is quite correct. In case a demonstration of this point is required, it is provided in Listing 1 in the Appendix.

The proprietary econometric software Stata is in the enviable position of being regarded as canonical by many practitioners. Nonetheless, the interpretation of the Baltagi–Chang unbalanced variant of Swamy–Arora embedded in Stata's `xtreg` command (when invoked

[1] Briefly, Z is an $n \times N$ matrix defined as $\mathrm{diag}(\iota_{T_i})$, where ι_{T_i} is a column vector of 1s of length T_i. When multiplied into an N-vector u it produces an n-vector in which element i of u occurs T_i times. In a balanced panel the trace reduces to TK.

[2] This is not specific to the 1994 Baltagi–Chang article; it is also the presentation given in Baltagi (2005) (and other editions of Baltagi's panel-data book).

[3] This would require that P be an $N \times N$ diagonal matrix with the T_i's on the diagonal.

with its sa option) is mistaken. What Stata in fact calculates in place of $\hat{u}^{b\prime} P \hat{u}^b$ is

$$\mathrm{SSR}_b^* = \sum_{i=1}^{N} T_i \tilde{u}_i^2 \tag{5}$$

where $\tilde{u}_i$ ($i = 1, \ldots, N$) denotes the residual from a Between regression of length N. This is therefore a T_i-weighted sum of squared residuals from an unweighted regression, as opposed to the sum of squared residuals from a T_i-weighted regression, as per Baltagi–Chang: these quantities are in general not equal (unless the T_i values are all the same, which *ex hypothesi* is not the case here). That (5) truly represents what Stata does was determined by emulating the results from `xtreg` with its `sa` option in gretl: if and only if we use this expression do we get exact numerical agreement.[4]

By contrast, the `plm` package for panel-data modeling in R (Croissant, 2016) implements just what Baltagi and Chang specified when the Swamy–Arora estimator is selected for an unbalanced panel. However, we cannot simply take for granted that what Baltagi and Chang recommend is "right" (and what Stata does is "wrong"). That remains to be determined.

3 The simulation dataset

Our artificial dataset comprises a maximum of $T = 20$ observations on each of $N = 10$, 20 or 40 individuals. In each case the individuals are divided into five classes of equal size, having T_i values of 1, 4, 10, 15 and 20 respectively. The mean T_i is therefore 10, so the total number of usable observations is $n = 10N$ (that is, 100, 200 or 400).

The data-generating process is

$$y_{it} = \alpha + \beta x_{it} + v_i + \varepsilon_{it}$$

where $x_{it} \sim N(0, 1)$, $v_i \sim N(0, \sigma_v^2)$ and $\varepsilon_{it} \sim N(0, \sigma_\varepsilon^2)$. These three variates are both mutually and serially independent.

For each run of the simulation the x_{it} values are generated once while the v_i and ε_{it} are generated anew at each of K = 250,000 iterations. The runs differ in two dimensions: (a) the number of individuals, as mentioned above, and (b) the relative size of the individual and idiosyncratic variances, namely

1. $\sigma_v^2 = \sigma_\varepsilon^2 = 1.0$;
2. $\sigma_v^2 = 1.6$, $\sigma_\varepsilon^2 = 0.4$; and
3. $\sigma_v^2 = 0.4$, $\sigma_\varepsilon^2 = 1.6$.

These choices conserve the variance of the composite error $v_i + \varepsilon_{it}$ at 2.0 while enabling us to explore the effect of the relative size of its components. The Tables below therefore show 9 "cases": three sample sizes $\times$ three ratios of σ_v^2 to σ_ε^2.

4 Simulation and results

In each run we employ four estimators, as follows.

1. What might be called the "standard" estimator. This employs a minimally modified version of equation (1), in which T (undefined for an unbalanced panel) is replaced

[4] In addition this point is now clear in the documentation for version 15 of Stata (`xtreg`, "Methods and formulas"). In previous versions the account given was inaccurate.

by some sort of average of the T_i's. This is used by `Stata` when the `re` option is given with `xtreg`, and it is the default in gretl. Stata replaces T with the harmonic mean of the T_i's; since the value in question appears as a divisor in the formula for $\hat{\sigma}_v^2$ this choice seems reasonable and gretl follows suit.[5] We give this estimator the label "HMT" (for harmonic mean of T).

2. Stata's take on Baltagi–Chang, which, as noted above, uses in (3) the T_i-weighted sum of squared residuals from an unweighted Between regression of length N. We label this "SBC."

3. The estimator actually specified by Baltagi and Chang, which involves T_i-weighted estimation of the Between model (labeled "BC").

4. By way of benchmark, "Infeasible GLS" (IGLS): we can implement this since we know the population values of σ_v^2 and σ_ε^2 and so can use these values in place of estimates.

We consider three figures of merit for the estimators: the mean error and MSE of $\hat{\sigma}_v$ along with the MSE of $\hat{\beta}$, expressed as a ratio to that produced by IGLS. To be explicit, we mean by the MSE of $\hat{\beta}$ the mean value of $(\hat{\beta} - \beta)^2$, or the mean square error of estimate, and similarly for the MSE of $\hat{\sigma}_v$.

Results are shown in Table 1.[6] The best values for each "case" (minimum absolute value of mean error, minima of the MSEs) are highlighted in red. These results enable us to address two questions: (1) Is one of the Baltagi–Chang variants clearly "correct" and the other ill advised? (2) Does either of these variants offer a demonstrable efficiency gain over simple HMT?

On the first question—comparison of BC and SBC—it is clear that BC is a more efficient estimator of the variance of the individual effects than SBC. The figures of merit for $\hat{\sigma}_v^2$ are always substantially better for BC than SBC (although MSE($\hat{\sigma}_v^2$) for BC is in some cases inferior to that for the HMT method). However, this does not translate into greater efficiency in estimation of the slope coefficient, β, which is presumably of greater inherent interest in practice; in fact, SBC is slightly better in that regard. In all nine cases in Table 1 SBC gives a lesser MSE($\hat{\beta}$) than BC, although the differences are small (and SBC is in two cases marginally inferior to HMT in this respect).

Cameron and Trivedi (2005) remark of the random effects model that "more efficient estimators of the variance components... will not necessarily increase the efficiency of $\hat{\beta}_{\text{RE}}$" (p. 734). We're seeing a noteworthy instance of their point here: a clearly *inefficient* estimator of σ_v^2, SBC, turns out to yield the most efficient estimator of β. In each of the runs reported in Table 1 we initialized gretl's random number generator with the same seed, for the sake of consistency across the cases. However, in view of the somewhat surprising results we performed additional runs with different seeds (and also with different patterns of imbalance in the data). On that basis we're able to say that the reported results are not flukey but quite typical.

There appears to be a problem of the "second best" type here. Reassuringly, if we use the true population value of σ_v^2 (IGLS) we always get the minimal MSE for the β estimate (as shown in Table 1: the MSE ratio to IGLS for $\hat{\beta}$ is always greater than 1.0 for the feasible estimators), but picking a better feasible estimator of σ_v^2 does not necessarily yield better estimates of β.

A further perspective on our simulation results is provided by Figure 1. Here we restrict the comparison to two figures of merit—namely, the MSEs of $\hat{\sigma}_v^2$ and $\hat{\beta}$—but add two elements. First, we include a sample with $N = 80$ individuals (and so $n = 800$ total observations), to help gauge the consistency of the estimators. Second, we construct a balanced panel with

[5] We have not been able to find a source in the panel-data literature for this recommendation.

[6] The script that produced this output is shown in Listing 2 in the Appendix.

		ME($\hat{\sigma}_v^2$)	MSE($\hat{\sigma}_v^2$)	MSE($\hat{\beta}$)/IGLS
$N = 10, n = 100$				
$\sigma_v^2 = \sigma_\varepsilon^2$	HMT	-0.0458	0.4049	1.01099
	SBC	+0.0746	0.4435	1.00515
	BC	+0.0006	0.3936	1.00600
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 > \sigma_\varepsilon^2$	HMT	-0.0183	0.7200	1.00449
	SB	+0.1062	1.0134	1.00366
	BC	+0.0007	0.9054	1.00428
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 < \sigma_\varepsilon^2$	HMT	-0.0404	0.1827	1.01665
	SBC	+0.0440	0.1066	1.00522
	BC	+0.0019	0.0919	1.00596
	IGLS	0.0000	0.0000	1.00000
$N = 20, n = 200$				
$\sigma_v^2 = \sigma_\varepsilon^2$	HMT	-0.0259	0.1898	1.00283
	SBC	+0.0589	0.1994	1.00164
	BC	-0.0008	0.1798	1.00190
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 > \sigma_\varepsilon^2$	HMT	-0.0107	0.3246	1.00107
	SBC	+0.0837	0.4549	1.00107
	BC	-0.0011	0.4137	1.00126
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 < \sigma_\varepsilon^2$	HMT	-0.0320	0.1006	1.01608
	SBC	+0.0342	0.0487	1.00249
	BC	-0.0003	0.0426	1.00308
	IGLS	0.0000	0.0000	1.00000
$N = 40, n = 400$				
$\sigma_v^2 = \sigma_\varepsilon^2$	HMT	-0.0138	0.0913	1.00146
	SBC	+0.0367	0.0917	1.00104
	BC	+0.0003	0.0862	1.00118
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 > \sigma_\varepsilon^2$	HMT	-0.0049	0.1542	1.00060
	SBC	+0.0491	0.2096	1.00064
	BC	+0.0007	0.1989	1.00072
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 < \sigma_\varepsilon^2$	HMT	-0.0214	0.0525	1.01126
	SBC	+0.0243	0.0224	1.00164
	BC	+0.0000	0.0203	1.00195
	IGLS	0.0000	0.0000	1.00000

Table 1: Errors of estimation, Swamy–Arora (ME = mean error, MSE = mean square error)

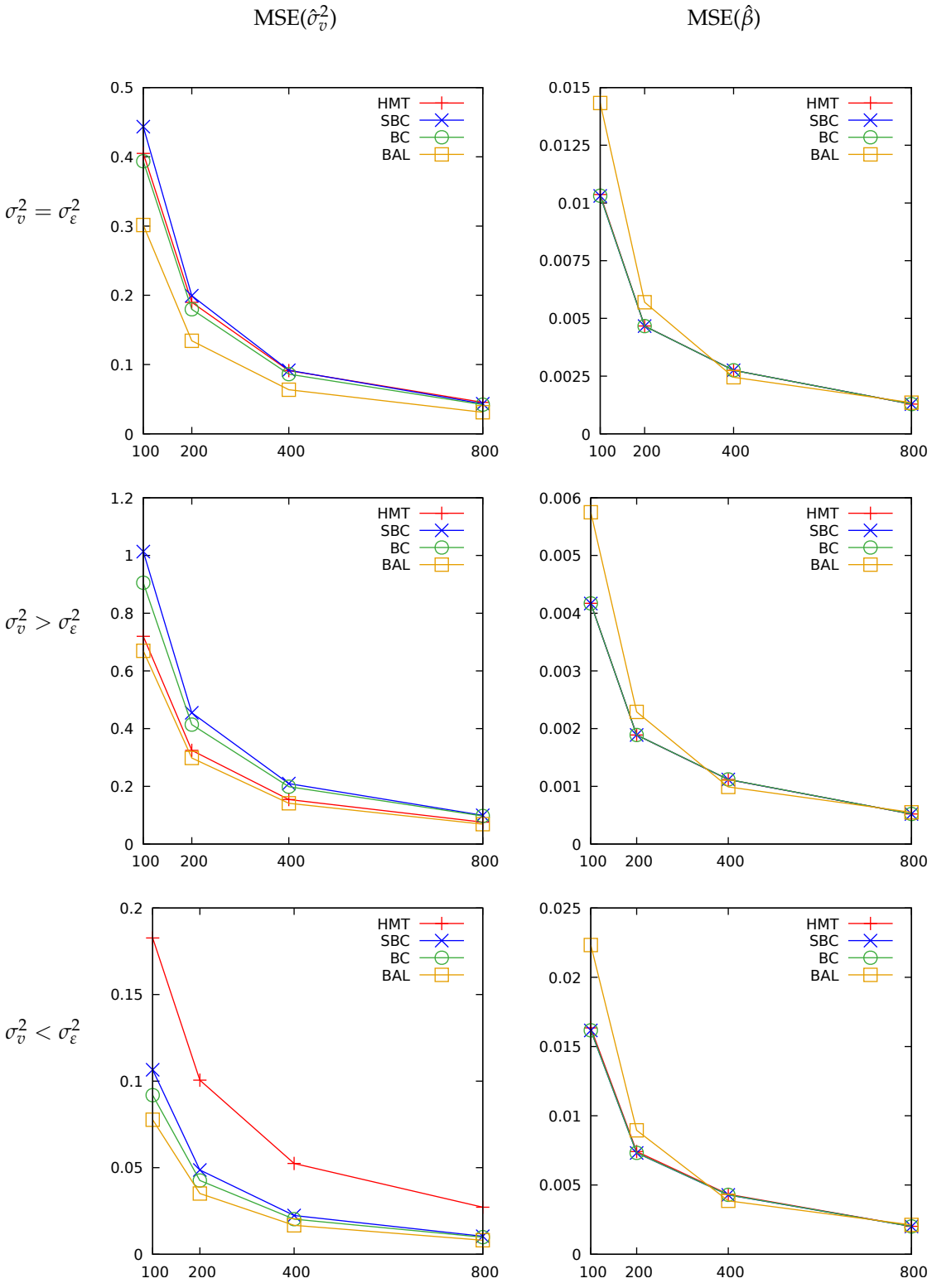

Figure 1: Consistency check: Swamy–Arora variants, including a balanced panel with $T = 10$ (BAL); $n = \sum T_i$ on x-axis.

the same n but a uniform $T = 10$ observations per individual; on this sample we deploy only the standard balanced-panel Swamy–Arora estimator. The following points emerge from this exercise:

- All of the estimators under consideration clearly exhibit consistency with respect to N.
- While the performance of the various formulae considered as estimators of σ_v^2 differs quite substantially, their performance as estimators of the slope coefficient, β, is almost indistinguishable. This is particularly true of the three unbalanced-panel methods. And to the extent that estimators of β *are* distinguishable, it is noteworthy that the balanced-panel estimator applied to a balanced panel of the same overall length n as the unbalanced ones is actually inferior for $n < 400$.
- A variant of the somewhat suprising result noted above—namely, that a more efficient estimator of σ_v^2 may, in the Swamy–Arora context, yield a less efficient estimator of β—is exhibited in the balanced/unbalanced comparison. A balanced sample permits a sharper estimate of σ_v^2 at all n, but at smaller n the *un*balanced samples support sharper estimation of β.

Our second question concerned the gain from using either BC or SBC relative to the minimally adjusted version of Swamy–Arora. Our results suggest that one can gain efficiency in estimation of σ_v^2 by using BC (but *not* SBC, unless σ_v^2 is small relative to σ_ε^2). Both SBC and (to a slightly lesser extent) BC appear to offer greater efficiency in estimation of β, though the gain is marginal and not guaranteed. In particular, when σ_v^2 is large relative to σ_ε^2 the "standard" estimator may work best. Conversely, the greatest gains from SBC and BC occur when σ_v^2 is relatively small, but even then they are small.

5 Nerlove's method

The modification to Swamy–Arora proposed by Baltagi and Chang involves weighting by the T_i values in an unbalanced panel. In the case of Nerlove's estimator, a "natural" analogy would be to compute the T_i-weighted variance of the estimated fixed effects in place of the unweighted variance. That is, we replace equation (2) with

$$\hat{\sigma}_v^2 = \frac{N}{N-1} \sum_{i=1}^{N} w_i (\hat{\alpha}_i - \bar{\hat{\alpha}}_w)^2$$

where

$$w_i = T_i / n \quad \text{and} \quad \bar{\hat{\alpha}}_w = \sum_{i=1}^{N} w_i \hat{\alpha}_i$$

Table 2 shows results from simulations on the same pattern as those performed in relation to Swamy–Arora. Here "STD" means the standard unweighted formula and "WTD" the T_i-weighted one. Our findings are broadly similar to those for Swamy–Arora: efficiency *may* be gained in estimation of both σ_v^2 and β by weighting, but to the greatest extent when σ_v^2 is relatively small, while the unweighted estimator may be superior when σ_v^2 is large.

As with Swamy–Arora, Figure 2 extends the analysis in Table 2. Here we see some similarities with the Swamy–Arora exercise above but also some differences.

- On the similarities side, substantial efficiency gains in estimation of σ_v^2 via T_i-weighting (in the cases $\sigma_v^2 \leq \sigma_\varepsilon^2$) fail to translate into gains in efficiency of estimation of β. Here, again, the respective MSE($\hat{\beta}$) values for the weighted and unweighted estimators are virtually indistinguishable.

- On the differences side, it appears that, given the overall n, a balanced sample is always favorable for estimation of both σ_v^2 and β when the Nerlove approach is taken. Of course, this contrast—balanced versus unbalanced samples of the same overall size—is hypothetical; it does not correspond to a choice typically faced by a researcher.

6 Practical conclusions

We embarked on this exercise hoping to draw definite conclusions regarding the best policy for econometric software to adopt with regard to random-effects panel-data models estimated on unbalanced data. That turns out to be more difficult than we anticipated. Nonetheless we have made some modifications to gretl, which we hope will be useful to practitioners, in light of our results.

First, a little background: the basic command for panel-data estimation in gretl is `panel`. The default estimator under `panel` is fixed effects but random effects can be invoked via the `--random-effects` option flag. Under random effects the default procedure is Swamy–Arora (with HMT for unbalanced panels) but the additional `--nerlove` option is available.

So here are the modifications. A new option, `--unbalanced`, is available when `panel` is invoked in random-effects mode. The requirements of the new option differ depending on whether the `--nerlove` option is also given, as follows.

- When Swamy–Arora is used, the new option requires a parameter, either `stata` or `bc`. These invoke, respectively, the variants labeled "SBC" and "BC" above. For example,

  ```
  panel y 0 X --random-effects --unbalanced=bc
  ```

  will invoke the Baltagi–Chang unbalanced-panel version of Swamy–Arora, if the dataset is in fact unbalanced for the regression in question.

- When the Nerlove estimator is selected, no parameter is required; the effect of `--unbalanced` is just to employ the T_i-weighted estimator of the variance of the fixed effects, as described above.

We hope it is clear from the the discussion above that marvels should not be expected from gretl's `--unbalanced` option. If the user's focus is firmly on the regression slope coefficients, little gain should be expected.

However, given the consistency displayed in Figures 1 and 2, it appears that the practitioner need not be too worried about using an unbalanced sample in random-effects estimation, with or without a putative "correction" for the unbalanced nature of the sample, provided that the overall sample size is adequate. This suggests a comment on the `plm` panel-data package for R (Croissant, 2016): `plm` refuses to produce estimates for unbalanced panels except when the Swamy–Arora estimator is specified (in which case the package produces Baltagi–Chang estimates). In light of our results this seems unnecessarily austere: balance or lack thereof in the panel makes less difference than one might think to the efficiency of estimation of the slope coefficient(s).

References

Baltagi, B. H. (2005) *Econometric Analysis of Panel Data, 3e*, Chichester: Wiley.

Baltagi, B. H. and Y.-J. Chang (1994) 'Incomplete panels: A comparative study of alternative estimators for the unbalanced one-way error component regression model', *Journal of Econometrics* 62: 67–89.

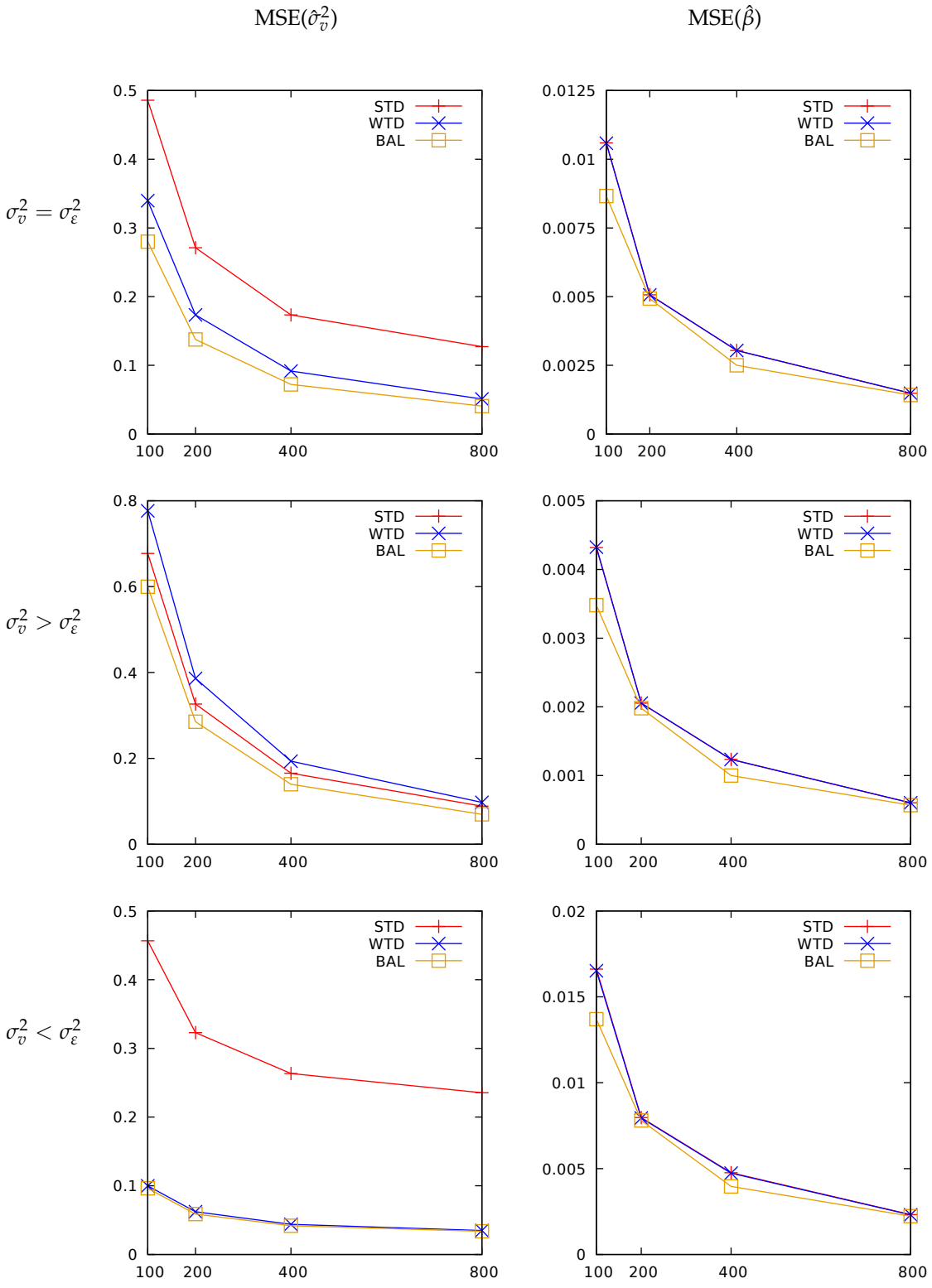

Figure 2: Consistency check: Nerlove variants; $n = \sum T_i$ on x-axis.

Cameron, A. C. and P. K. Trivedi (2005) *Microeconometrics, Methods and Applications*, Cambridge: Cambridge University Press.

Cottrell, A. and R. Lucchetti (2017) *Gretl User's Guide*, gretl documentation. URL `http://sourceforge.net/projects/gretl/files/manual/`.

Croissant, Y. (2016) 'Package 'plm''. R package documentation. URL `https://cran.r-project.org/package=plm`.

Nerlove, M. (1971) 'Further evidence on the estimation of dynamic economic relations from a time series of cross sections', *Econometrica* 39: 359–382.

Swamy, P. A. V. B. and S. S. Arora (1972) 'The exact finite sample properties of the estimators of coefficients in the error components regression models', *Econometrica* 40: 261–275.

		ME($\hat{\sigma}_v^2$)	MSE($\hat{\sigma}_v^2$)	MSE($\hat{\beta}$)/IGLS
$N = 10, n = 100$				
$\sigma_v^2 = \sigma_\varepsilon^2$	STD	+0.2917	0.4858	1.00371
	WTD	+0.0510	0.3397	1.00238
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 > \sigma_\varepsilon^2$	STD	+0.1174	0.6772	1.00155
	WTD	−0.0393	0.7766	1.00326
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 < \sigma_\varepsilon^2$	STD	+0.4663	0.4566	1.00816
	WTD	+0.1415	0.0998	1.00226
	IGLS	0.0000	0.0000	1.00000
$N = 20, n = 200$				
$\sigma_v^2 = \sigma_\varepsilon^2$	STD	+0.2876	0.2711	1.00189
	WTD	+0.0773	0.1734	1.00117
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 > \sigma_\varepsilon^2$	STD	+0.1155	0.3264	1.00067
	WTD	+0.0032	0.3863	1.00089
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 < \sigma_\varepsilon^2$	STD	+0.4593	0.3230	1.00559
	WTD	+0.1514	0.0622	1.00203
	IGLS	0.0000	0.0000	1.00000
$N = 40, n = 400$				
$\sigma_v^2 = \sigma_\varepsilon^2$	STD	+0.2856	0.1733	1.00167
	WTD	+0.0897	0.0916	1.00096
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 > \sigma_\varepsilon^2$	STD	+0.1148	0.1655	1.00044
	WTD	+0.0231	0.1935	1.00051
	IGLS	0.0000	0.0000	1.00000
$\sigma_v^2 < \sigma_\varepsilon^2$	STD	+0.4567	0.2634	1.00829
	WTD	+0.1563	0.0440	1.00308
	IGLS	0.0000	0.0000	1.00000

Table 2: Errors of estimation, Nerlove (ME = mean error, MSE = mean square error)

Appendix: hansl scripts

Example 1: Demonstration of equivalence

This script demonstrates the equivalence of the two ways given by Baltagi and Chang to obtain the term they notate as $\hat{u}^{b\prime} P \hat{u}^{b}$: (a) quite literally, where $\hat{u}^{b}$ is the residual from a full-length Between regression and P is a block-diagonal matrix as described in the text, and (b) as the sum of squared residuals from a Between regression of length N, where all the (individual-mean) data are multiplied by $\sqrt{T_i}$. The script also demonstrates that this magnitude differs from the T_i-weighted sum of squared residuals from an unweighted Between regression of length N, as employed by Stata.

```
set verbose off

nulldata 120
# panel structure: max(Ti) = 4, 30 individuals
setobs 4 1:1 --stacked-time-series
series x = normal()
series vi = pexpand(mnormal(30,1))
series eit = normal()
series y = 10 + x + vi + eit

# apply mask to reduce Ti to 2 for the first 15 individuals
scalar j=1
loop i=1..15 -q
  y[j] = NA
  x[j] = NA
  y[j+1] = NA
  x[j+1] = NA
  j += 4
endloop

# run "standard" between regression with N=30 observations
panel y 0 x --between
matrix ub = $uhat
ub[1:15] *= sqrt(2)
ub[16:30] *= sqrt(4)
printf "Ti-weighted SSR (a la Stata): %g\n", ub'ub

/*
# compare stata (same results as gretl: uncomment if you have stata)
genr unit
genr time
foreign language=stata --send-data
  xtset unit time
  xtreg y x, be
end foreign
*/
```

Continued on next page.

Example 1: Demonstration of equivalence, continued

```
# build the Baltagi-Chang P matrix (** = Kronecker product)
matrix P1 = I(15) ** ones(2,2)/2
matrix P2 = I(15) ** ones(4,4)/4
matrix P = (P1 ~ zeros(30,60)) | (zeros(60,30) ~ P2)

# calculate ub'P*ub as per Baltagi-Chang
matrix my = {y}
matrix X = ones(90,1) ~ {x}
matrix bb1 = inv(X'P*X) * X'P*my
printf "\nBaltagi-Chang between beta-hat (regression length n)"
printf "\n%#.6g", bb1
matrix ub = my - X*bb1
printf "ub'P*ub = %g\n", ub'P*ub

# construct individual-means dataset of length N, and run
# Ti-weighted between regression
matrix ybar = pshrink(pmean(y))
matrix Xbar = ones(30,1) ~ pshrink(pmean(x))
ybar[1:15] *= sqrt(2)
ybar[16:30] *= sqrt(4)
Xbar[1:15,] *= sqrt(2)
Xbar[16:30,] *= sqrt(4)
matrix bb2 = mols(ybar, Xbar)
printf "\nbeta-hat from weighted between regression (length N)"
printf "\n%#.6g", bb2
matrix ubw = ybar - Xbar*bb2
printf "ubw'ubw = %g\n", ubw'ubw
```

Example 2: Simulation script

This is the script used to populate Table 1 in the text. It is designed to be called by the program `gretlcli` with the `--scriptopt` command-line option set to 1, 2, or 3 (for $N = 10$), 21, 22 or 23 ($N = 20$), or 41, 42, or 43 ($N = 40$). The Nerlove table is produced by a very similar script.

```
set verbose off

scalar parms = scriptopt
scalar N = 10
if parms > 40
  N = 40
  parms -= 40
elif parms > 20
  N = 20
  parms -= 20
endif

scalar T = 20   # max time-series length
scalar NT = N*T # total length of dataset

nulldata NT --preserve
setobs T 1:1 --stacked-time-series

set seed 777333119 # comment this out for a runtime-based seed
matrix Ti = {1, 5, 10, 14, 20}
series x = normal()
scalar K = 250000

# set the relative variances based on @parms
if parms == 1
  true_s2v = 1.0
  true_s2e = 1.0
elif parms == 2
  true_s2v = 1.6
  true_s2e = 0.4
else
  true_s2v = 0.4
  true_s2e = 1.6
endif

scalar true_sv = sqrt(true_s2v)
scalar true_se = sqrt(true_s2e)
scalar true_b2 = 1.0

printf "\ntrue_s2v = %.1f\n", true_s2v
printf "true_s2e = %.1f\n", true_s2e
printf "true_b2 = %.1f\n", true_b2
printf "NT = %d\n", NT
printf "K  = %d\n", K

matrix s2v = zeros(K,4)
matrix b2 =  zeros(K,4)
```

Continued on next page.

Example 2: Simulation script, continued

```
# the number of distinct Ti values
scalar nTi = nelem(Ti)
# the number of individuals with each Ti value
scalar indiv_per = N/nTi

# write matrix of population variances for IGLS
string vname = sprintf("reV%d.mat", parms)
mwrite({true_s2v, true_s2e}, vname)

loop k=1..K -q
    series v = pexpand(mnormal(N,1) * true_sv)
    series e = normal() * true_se
    series y = 10 + true_b2*x + v + e

    # unbalance the data!
    loop j=1..nTi -q
        scalar tmax = T - Ti[j]
        loop i=1..indiv_per -q
            scalar ij = 1 + (j-1)*T*indiv_per + (i-1) * T
            loop t=1..tmax -q
                y[ij+t-1] = NA
            endloop
        endloop
    endloop

    panel y 0 x --random --quiet
    s2v[k,1] = $model.s2v
    b2[k,1] = $coeff[2]
    panel y 0 x --random --unbalanced=stata --quiet
    s2v[k,2] = $model.s2v
    b2[k,2] = $coeff[2]
    panel y 0 x --random --unbalanced=bc --quiet
    s2v[k,3] = $model.s2v
    b2[k,3] = $coeff[2]
    panel y 0 x --random --unbalanced=@vname --quiet
    s2v[k,4] = $model.s2v
    b2[k,4] = $coeff[2]
endloop

matrix ME = meanc(s2v - true_s2v)
matrix MSE = meanc((s2v - true_s2v).^2)
matrix bMSE = meanc((b2 - true_b2).^2)
printf "\nMEs, MSE of s2v estimate, b2 MSE ratio\n\n"

printf " &    &   ME    & MSE   & ratio\n"
printf " & HMT  & %+.4f & %.4f & %.5f \\\\\n", ME[1], MSE[1], bMSE[1]/bMSE[4]
printf " & SBC  & %+.4f & %.4f & %.5f \\\\\n", ME[2], MSE[2], bMSE[2]/bMSE[4]
printf " & BC   & %+.4f & %.4f & %.5f \\\\\n", ME[3], MSE[3], bMSE[3]/bMSE[4]
printf " & IGLS & %4f   & %.4f & %.5f \\\\\n", ME[4], MSE[4], bMSE[4]/bMSE[4]
```