xtcluster: To pool or not to pool? A partially heterogeneous framework for short panel data models

Demetris Christodoulou (Sydney) and Vasilis Sarafidis (Monash)

Methodological and Empirical Advances in Financial Analysis (MEAFA)

September 25, 2015

1. Estimation problem

1. Estimation problem

2. `xtcluster` estimation command

Outline

Estimation question

- Slope parameter homogeneity in large-scale panel data analysis is an assumption that is often difficult to justify. On the other hand, imposing no structure on how parameters vary across individual units is rather an extreme and quite inefficient approach.

Estimation question

- Slope parameter homogeneity in large-scale panel data analysis is an assumption that is often difficult to justify. On the other hand, imposing no structure on how parameters vary across individual units is rather an extreme and quite inefficient approach.
- Sarafidis and Weber (2015, *Oxford Bulletin of Econ and Stat*) propose a middle ground, that of imposing partially heterogeneous restrictions with respect to the individuals, N. That is, individuals may behave in clusters with homogeneous slope parameters and the intra-cluster heterogeneity is attributed to unobserved fixed effects.

Estimation question

- Slope parameter homogeneity in large-scale panel data analysis is an assumption that is often difficult to justify. On the other hand, imposing no structure on how parameters vary across individual units is rather an extreme and quite inefficient approach.
- Sarafidis and Weber (2015, *Oxford Bulletin of Econ and Stat*) propose a middle ground, that of imposing partially heterogeneous restrictions with respect to the individuals, N. That is, individuals may behave in clusters with homogeneous slope parameters and the intra-cluster heterogeneity is attributed to unobserved fixed effects.
- The method is useful for exploring data in the absence of knowledge about parameter structures. It is also useful for examining the validity of *a priori* imposed structures, such as industry or risk classification, or some other economically-driven structure.

Estimation setting

- For a given linear short panel data model with exogenous regressors, the estimation problem is concerned with discovering potentially heterogeneous clusters of individuals, $i_\omega = 1, 2, \ldots, N_\omega$, each repeatedly observed over a fixed time period, $t = 1, 2, \ldots, T$.

Estimation setting

- For a given linear short panel data model with exogenous regressors, the estimation problem is concerned with discovering potentially heterogeneous clusters of individuals, $i_\omega = 1, 2, \ldots, N_\omega$, each repeatedly observed over a fixed time period, $t = 1, 2, \ldots, T$.

- Note that it is the individual that belongs to a cluster $\omega = 1, 2, \ldots, \Omega$ and not a single observation, hence no individual can be classified in more than one cluster.

Estimation setting

- For a given linear short panel data model with exogenous regressors, the estimation problem is concerned with discovering potentially heterogeneous clusters of individuals, $i_\omega = 1, 2, \ldots, N_\omega$, each repeatedly observed over a fixed time period, $t = 1, 2, \ldots, T$.
- Note that it is the individual that belongs to a cluster $\omega = 1, 2, \ldots, \Omega$ and not a single observation, hence no individual can be classified in more than one cluster.
- The focus is on the analysis of 'short panels' where $N >> T$, with $N \rightarrow \infty$ and T fixed. In practical applications, T can be unbalanced hence with individual-specific T_i in which case $\overline{T}$ is fixed.

Estimation problem

- The key estimation problem is that we neither know the size of Ω nor the individual classification to each cluster, i_ω. These need to be estimated from the data.

Estimation problem

- The key estimation problem is that we neither know the size of Ω nor the individual classification to each cluster, i_ω. These need to be estimated from the data.

- Sarafidis and Weber (2015) propose to estimate Ω and the corresponding classification to each cluster, i_ω, using a *partitional clustering* approach (e.g. Kaufman and Rousseeuw, 1990; Everitt 1993). They show through analytical work and simulation that their proposed solution yields strongly consistent estimates for estimating the size number of Ω and the true partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ with $Prob \to 1$ as $N \to \infty$, for any T that remains fixed.

Estimation problem

- The key estimation problem is that we neither know the size of Ω nor the individual classification to each cluster, i_ω. These need to be estimated from the data.

- Sarafidis and Weber (2015) propose to estimate Ω and the corresponding classification to each cluster, i_ω, using a *partitional clustering* approach (e.g. Kaufman and Rousseeuw, 1990; Everitt 1993). They show through analytical work and simulation that their proposed solution yields strongly consistent estimates for estimating the size number of Ω and the true partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ with $Prob \rightarrow 1$ as $N \rightarrow \infty$, for any T that remains fixed.

- The method has been originally developed for linear static panel data analysis with no endogenous regressors (i.e. can be applied with `xtreg`).

Estimation framework

- The generalised estimation framework for the linear static panel data regression is specified as follows:

$$y_{i_\omega t} = \beta'_\omega \mathbf{x}_{i_\omega t} + \epsilon_{i_\omega t}$$

Estimation framework

- The generalised estimation framework for the linear static panel data regression is specified as follows:

$$y_{i_\omega t} = \boldsymbol{\beta}'_\omega \mathbf{x}_{i_\omega t} + \epsilon_{i_\omega t}$$

- $y_{i_\omega t}$ is the dependent variable and $\mathbf{x}_{i_\omega t}$ a $M \times 1$ vector of covariates, which denote observations for the i^{th} individual that belongs to cluster ω. Observations are repeated over t.

Estimation framework

- The generalised estimation framework for the linear static panel data regression is specified as follows:

$$y_{i_\omega t} = \boldsymbol{\beta}'_\omega \mathbf{x}_{i_\omega t} + \epsilon_{i_\omega t}$$

- $y_{i_\omega t}$ is the dependent variable and $\mathbf{x}_{i_\omega t}$ a $M \times 1$ vector of covariates, which denote observations for the i^{th} individual that belongs to cluster ω. Observations are repeated over t.
- $\boldsymbol{\beta}_\omega$ is a $M \times 1$ vector of fixed parameters homogeneous to each cluster and heterogeneous across clusters.

Estimation framework

- The generalised estimation framework for the linear static panel data regression is specified as follows:

$$y_{i_\omega t} = \boldsymbol{\beta}'_\omega \mathbf{x}_{i_\omega t} + \epsilon_{i_\omega t}$$

- $y_{i_\omega t}$ is the dependent variable and $\mathbf{x}_{i_\omega t}$ a $M \times 1$ vector of covariates, which denote observations for the i^{th} individual that belongs to cluster ω. Observations are repeated over t.
- $\boldsymbol{\beta}_\omega$ is a $M \times 1$ vector of fixed parameters homogeneous to each cluster and heterogeneous across clusters.
- All remaining intra-cluster parameter heterogeneity is attributed to individual-specific and time-specific fixed effects, as part of the composite error term, $\epsilon_{i_\omega t} = u_{i_\omega} + t_s + v_{i_\omega t}$, where $v_{i_\omega t} \sim N\left(0, \sigma^2\right)$.

Estimation algorithm

The optimised partition and size of Ω is found as follows:

1. Specify a range for potential cluster size Ω (e.g. from 1 to 5).

Estimation algorithm

The optimised partition and size of Ω is found as follows:

1. Specify a range for potential cluster size Ω (e.g. from 1 to 5).
2. Given Ω, obtain an initial partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ using randomised classification, via predetermined classification, or through prespecified variation by applying the Calinski-Harabasz clustering criterion. Save the residual sum of squares for each cluster, RSS_ω, and calculate the total $RSS = \sum_{\omega=1}^{\Omega} RSS_\omega$.

Estimation algorithm

The optimised partition and size of Ω is found as follows:

1. Specify a range for potential cluster size Ω (e.g. from 1 to 5).
2. Given Ω, obtain an initial partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ using randomised classification, via predetermined classification, or through prespecified variation by applying the Calinski-Harabasz clustering criterion. Save the residual sum of squares for each cluster, RSS_ω, and calculate the total $RSS = \sum_{\omega=1}^{\Omega} RSS_\omega$.
3. Reclassify individual i_ω with all of its T_i observations to all remaining clusters each time obtaining the RSS. Assign individual i_ω as part of cluster ω that achieves the smallest RSS.

Estimation algorithm

The optimised partition and size of Ω is found as follows:

1. Specify a range for potential cluster size Ω (e.g. from 1 to 5).
2. Given Ω, obtain an initial partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ using randomised classification, via predetermined classification, or through prespecified variation by applying the Calinski-Harabasz clustering criterion. Save the residual sum of squares for each cluster, RSS_ω, and calculate the total $RSS = \sum_{\omega=1}^{\Omega} RSS_\omega$.
3. Reclassify individual i_ω with all of its T_i observations to all remaining clusters each time obtaining the RSS. Assign individual i_ω as part of cluster ω that achieves the smallest RSS.
4. Repeat step 2 for every other individual $j = 1, 2, \ldots, N-1$.

Estimation algorithm

The optimised partition and size of Ω is found as follows:

1. Specify a range for potential cluster size Ω (e.g. from 1 to 5).
2. Given Ω, obtain an initial partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ using randomised classification, via predetermined classification, or through prespecified variation by applying the Calinski-Harabasz clustering criterion. Save the residual sum of squares for each cluster, RSS_ω, and calculate the total $RSS = \sum_{\omega=1}^{\Omega} RSS_\omega$.
3. Reclassify individual i_ω with all of its T_i observations to all remaining clusters each time obtaining the RSS. Assign individual i_ω as part of cluster ω that achieves the smallest RSS.
4. Repeat step 2 for every other individual $j = 1, 2, \ldots, N-1$.
5. Iterate steps 2 and 3 until RSS cannot be minimised further.

Estimation algorithm

The optimised partition and size of Ω is found as follows:

1. Specify a range for potential cluster size Ω (e.g. from 1 to 5).
2. Given Ω, obtain an initial partition $N_\omega = N_1, N_2, \ldots, N_\Omega$ using randomised classification, via predetermined classification, or through prespecified variation by applying the Calinski-Harabasz clustering criterion. Save the residual sum of squares for each cluster, RSS_ω, and calculate the total $RSS = \sum_{\omega=1}^{\Omega} RSS_\omega$.
3. Reclassify individual i_ω with all of its T_i observations to all remaining clusters each time obtaining the RSS. Assign individual i_ω as part of cluster ω that achieves the smallest RSS.
4. Repeat step 2 for every other individual $j = 1, 2, \ldots, N-1$.
5. Iterate steps 2 and 3 until RSS cannot be minimised further.
6. Repeat steps 1 to 5 for different Ω sizes. The optimal Ω is the one which optimises the MIC.

Model Information Criterion (MIC)

- For linear models, the MIC is a function of the *RSS* subject to a penalty constraint:

$$MIC = N \log\left(\frac{RSS}{N\bar{T}}\right) + f\left(\Omega\right)\theta_N$$

Model Information Criterion (MIC)

- For linear models, the MIC is a function of the RSS subject to a penalty constraint:

$$MIC = N \log \left(\frac{RSS}{N\bar{T}} \right) + f(\Omega)\, \theta_N$$

- where $\bar{T} = \frac{1}{N}\sum_{i=1}^{N} T_i$ is the average time series length for panels with unequal time series. For panels with equal time-series it holds that $\bar{T} = T$ (i.e. strongly or weakly balanced panels).

Model Information Criterion (MIC)

- For linear models, the MIC is a function of the *RSS* subject to a penalty constraint:

$$MIC = N \log\left(\frac{RSS}{N\bar{T}}\right) + f(\Omega)\,\theta_N$$

- where $\bar{T} = \frac{1}{N}\sum_{i=1}^{N} T_i$ is the average time series length for panels with unequal time series. For panels with equal time-series it holds that $\bar{T} = T$ (i.e. strongly or weakly balanced panels).
- $f(\Omega)\,\theta_N$ is a penalty function for overfitting Ω. $f(\Omega)$ is a strictly increasing function, and the defaults are set to $f(\Omega) = \Omega$ and $\theta_N = 13\log N + 23N$.

Model Information Criterion (MIC)

- For linear models, the MIC is a function of the RSS subject to a penalty constraint:

$$MIC = N \log \left(\frac{RSS}{N\bar{T}} \right) + f(\Omega)\, \theta_N$$

- where $\bar{T} = \frac{1}{N} \sum_{i=1}^{N} T_i$ is the average time series length for panels with unequal time series. For panels with equal time-series it holds that $\bar{T} = T$ (i.e. strongly or weakly balanced panels).
- $f(\Omega)\, \theta_N$ is a penalty function for overfitting Ω. $f(\Omega)$ is a strictly increasing function, and the defaults are set to $f(\Omega) = \Omega$ and $\theta_N = 13 \log N + 23N$.
- Note that $1 \geq \Omega \geq \xi$ where $\xi << N$. Again, we do not know Ω but for $N \to \infty$ and T fixed, the MIC will point to the true Ω.

Outline

Stata estimation command

`xtcluster` requires the initialisation of cluster partition and then iterates the reclassification of individuals to all clusters up to the convergence of the RSS. The `eclass` command has the following syntax:

`xtcluster` *depvar indepvars* [if] [in] , [*options*]

Stata estimation command

`xtcluster` requires the initialisation of cluster partition and then iterates the reclassification of individuals to all clusters up to the convergence of the RSS. The `eclass` command has the following syntax:

`xtcluster` *depvar indepvars* `[if] [in] , [`*options*`]`

where *options*, with their default values, include:

- `name`(om)
- `iterate`(100)
- `theta`(default)
- `initpart`(*suboptions*)

```
random seed(123)
omega(3)
preclass(var)
ktype(kmeans)
prevars(varlist)
```

Stata estimation command

The program begins by obtaining an initial partition of all individuals into the Ω clusters. `initpart(default)` applies a randomized initial partition, which is equivalent to specifying:

- `initpart(random omega(3) seed(123) ktype(kmeans))`

Stata estimation command

The program begins by obtaining an initial partition of all individuals into the Ω clusters. `initpart(default)` applies a randomized initial partition, which is equivalent to specifying:

- `initpart(random omega(3) seed(123) ktype(kmeans))`

Alternatively, the user may specify a predetermined initial partition and the size of Ω on the basis of some theoretical classification:

- `initpart(preclass(`*indicator_var*`))`

Stata estimation command

The program begins by obtaining an initial partition of all individuals into the Ω clusters. `initpart(default)` applies a randomized initial partition, which is equivalent to specifying:

- `initpart(random omega(3) seed(123) ktype(kmeans))`

Alternatively, the user may specify a predetermined initial partition and the size of Ω on the basis of some theoretical classification:

- `initpart(preclass(`*indicator_var*`))`

Finally, the initial partition may be obtained on the basis of pre-specified variables, using the Calinski-Harabasz clustering criterion:

- `initpart(prevars(X) omega(3))`
- `initpart(prevars(`*varlist*`) omega(3) ktype(kmedians))`
- `initpart(prevars(b) omega(3))`

xtcluster display and ereturn list

- xtcluster prints iterations and the Total RSS at the end of every iteration up to convergence.

xtcluster display and ereturn list

- `xtcluster` prints iterations and the Total RSS at the end of every iteration up to convergence.
- It also prints the Calinski-Harabasz pseudo-F criterion for every number of clusters up to Ω if the option `kvars()` is specified.

xtcluster display and ereturn list

- `xtcluster` prints iterations and the Total RSS at the end of every iteration up to convergence.
- It also prints the Calinski-Harabasz pseudo-F criterion for every number of clusters up to Ω if the option `kvars()` is specified.
- `ereturn list` returns two scalars: the model information criterion as `e(mic)` and the specified size of Ω as `e(omega)`. It also returns a matrix: the *RSS* at every iteration in vector form as `e(rss)`.

xtcluster display and ereturn list

- `xtcluster` prints iterations and the Total RSS at the end of every iteration up to convergence.
- It also prints the Calinski-Harabasz pseudo-F criterion for every number of clusters up to Ω if the option `kvars()` is specified.
- `ereturn list` returns two scalars: the model information criterion as `e(mic)` and the specified size of Ω as `e(omega)`. It also returns a matrix: the *RSS* at every iteration in vector form as `e(rss)`.
- `xtcluster` generates an indicator variable as specified in option `name()` that takes the values $\omega = 1, 2, \ldots, \Omega$ (the default name is `om`). This can then be used for subsequent analysis, e.g.:

```
.  forvalues i = 1/`=e(omega)' {
2.  xtreg y x1 x2 if om==`i', fe vce(robust)
3.  }
```

Outline

Application 1: supporting evidence for homogeneity assumption

- `xtcluster` can be used to examine for the underlying hypothesis in `xtreg` that the slopes are homogeneous across all panels.

Application 1: supporting evidence for homogeneity assumption

- `xtcluster` can be used to examine for the underlying hypothesis in `xtreg` that the slopes are homogeneous across all panels.
- Munnell (1990) and Baltagi, Song and Jung (2001) apply a Cobb-Douglas production function for modelling the productivity of public capital at the state level, as a function of private capital stock, highway component, water component, building, and unemployment rate. The `productivity.dta` dataset contains observations on 48 U.S. states (panels) over 1970-1986. and the xtcluster command suggests that the slopes are homogeneous.
- The `productivity.dta` dataset is available from the StataPress website and is discussed in the manual entry of the `mixed` command.

Application 1: randomized initial partition

```
. use http://www.stata-press.com/data/r14/productivity.dta
. xtcluster gsp private emp hwy water other unemp,
> init(random omega(2) seed(123)) name(om2)

Initial partition through randomized classification

Partitional clustering iterations:

Iteration 1:   Total RSS = .7157044
Iteration 2:   Total RSS = .6881156
Iteration 3:   Total RSS = .6862053
Iteration 4:   Total RSS = .6744427
Iteration 5:   Total RSS = .6588415
Iteration 6:   Total RSS = .6588415
```

Specified Number of clusters	Total Residual Sum of Squares	Model Information Criterion
2	0.659	1738.298

Application 1: initial partition via prespecified variation

```
. xtcluster gsp private emp hwy water other unemp,
> init(prevars(X) omega(4))

Initial partition given prespecified variables:

Calinski & Harabasz pseudo-F for 1 groups:   .
Calinski & Harabasz pseudo-F for 2 groups:   1653.544
Calinski & Harabasz pseudo-F for 3 groups:   3419.6376
Calinski & Harabasz pseudo-F for 4 groups:   6893.3135

Partitional clustering iterations:

Iteration 1:   Total RSS = .4971343
Iteration 2:   Total RSS = .4665229
Iteration 3:   Total RSS = .4665229
```

Specified Number of clusters	Total Residual Sum of Squares	Model Information Criterion
4	0.467	4030.381

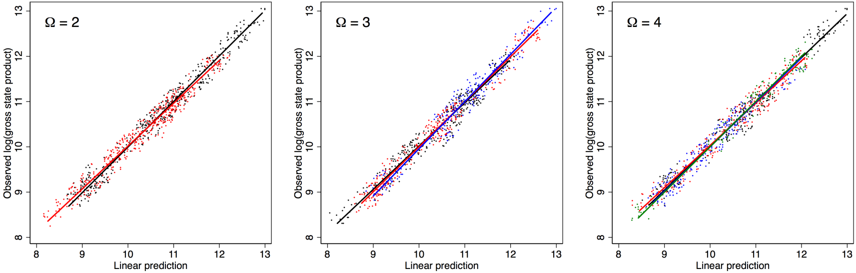
Application 1: initial partition via prespecified classes

```
. xtcluster gsp private emp hwy water other unemp,
> init(preclass(region)) name(om_region)

Initial partition given prespecified classification: region

Partitional clustering iterations:

Iteration 1:   Total RSS = .3527132
Iteration 2:   Total RSS = .3457506
Iteration 3:   Total RSS = .3428682
Iteration 4:   Total RSS = .3428682
```

Specified Number of clusters	Total Residual Sum of Squares	Model Information Criterion
9	0.343	9787.227

Application 1: different sizes for Ω

```
. forvalues i = 3/8 {
2. xtcluster gsp private emp hwy water other unemp,
>  init(random omega(`i') seed(123)) name(om`i')
3. }
```



- xtcluster pretty much confirms slope homogeneity across states.

Application 2: discovering potential heterogeneity

- The company audit costs (ln_af) are determined as a exponential function of the size of the audit in terms of company assets (ln_at), as well as the inherent risk of the audit in terms of rate of liquidity (cr) and leverage exposure (d2e).
- The dataset audit.dta holds hand-collected observations on S&P ASX200 company financials over 2000-2007 including audit fees plus other company characteristics. It is an unbalanced panel dataset, and is available from the MEAFA website:

```
. use http://meafa3.econ.usyd.edu.au/dta/audit.dta
```

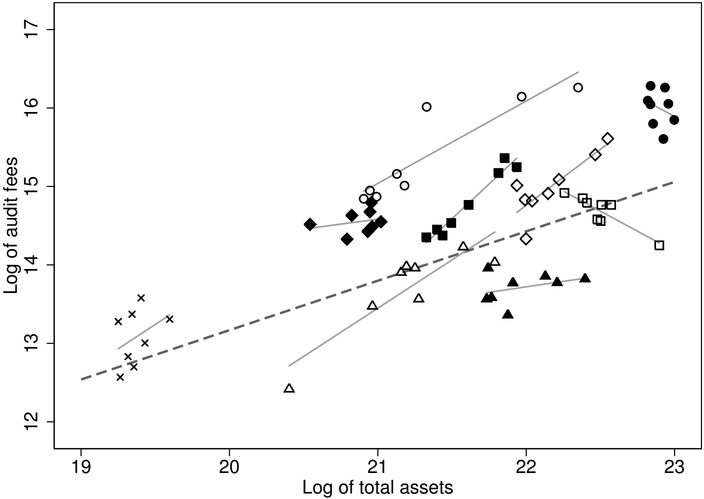
Application 2: discovering potential heterogeneity

- The company audit costs (`ln_af`) are determined as a exponential function of the size of the audit in terms of company assets (`ln_at`), as well as the inherent risk of the audit in terms of rate of liquidity (`cr`) and leverage exposure (`d2e`).
- The dataset `audit.dta` holds hand-collected observations on S&P ASX200 company financials over 2000-2007 including audit fees plus other company characteristics. It is an unbalanced panel dataset, and is available from the MEAFA website:

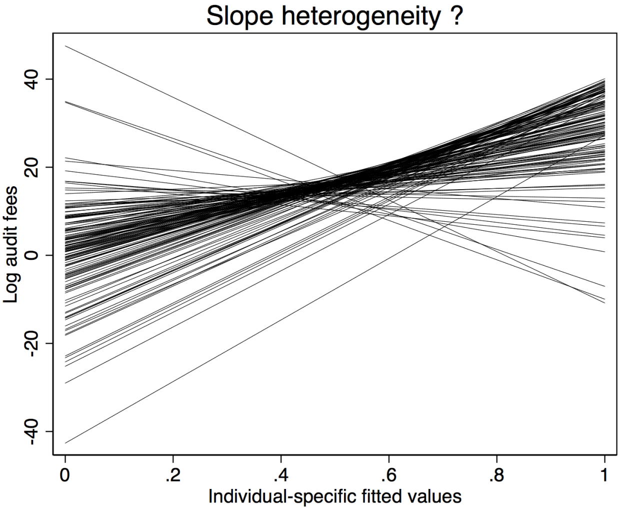
```
. use http://meafa3.econ.usyd.edu.au/dta/audit.dta
```

- An initial EDA investigation on parameter structure suggests that `xtcluster` may be well suited for discovering heterogeneity in this data. There is evident heterogeneity in slope coefficients plus remaining heterogeneity in intercepts within clusters of slopes.

Application 2: evidence for potential slope heterogeneity



Application 2: evidence for potential slope heterogeneity



Application 2: examining claimed heterogeneity

- In addition to exploring for potential slope heterogeneity, `xtcluster` can be applied to examine the claims made by the Australian company regulator, ASIC, offering financial reporting cost relief to companies by allowing them to redact disclosure for wholly-owned subsidiary companies. If the relief is effective then companies benefiting should be described by less steep slopes.

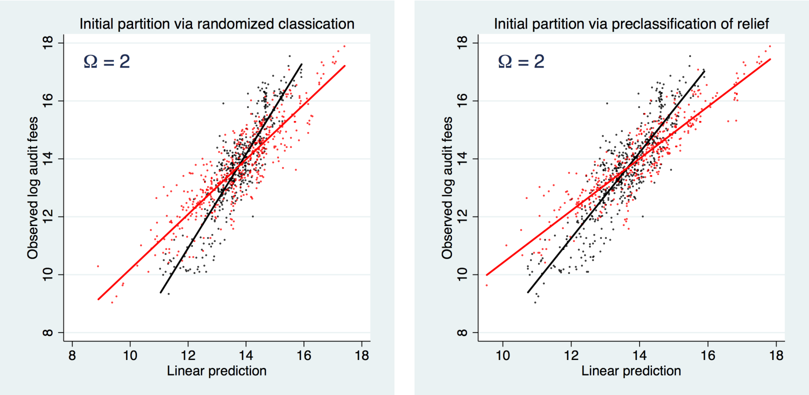
Application 2: examining claimed heterogeneity

- In addition to exploring for potential slope heterogeneity, `xtcluster` can be applied to examine the claims made by the Australian company regulator, ASIC, offering financial reporting cost relief to companies by allowing them to redact disclosure for wholly-owned subsidiary companies. If the relief is effective then companies benefiting should be described by less steep slopes.

```
. use http://meafa3.econ.usyd.edu.au/dta/audit.dta
. generate ln_af = ln(af)
. generate ln_at = ln(at)
. generate cr = lc/ac
. generate d2e = lt/(at-lt)
. xtcluster ln_af ln_at cr d2e, init(random omega(2)) name(om2)
. xtcluster ln_af ln_at cr d2e, init(preclass(relief))
```

Application 2: heterogeneous slopes

- The MIC suggests optimal partition at $\Omega = 2$.



Application 2: examining the assumption of relief

- Obtaining the initial partition through either randomised classification or preclassification seems to converge to the same final cluster partition. To check if they are the same individuals:

```
. egen tag = tag(id)
. tabulate om2 om_relief if tag
```

om2	om_relief 1	om_relief 2	Total
1	**70**	**7**	**77**
2	**7**	**52**	**59**
Total	**77**	**59**	**136**

Application 2: examining the assumption of relief

- More interestingly, to examine the assumption of `relief`, `tabulate` the optimal partition for $\Omega = 2$ with the binary preclassification of `relief`:

```
. tabulate om_relief relief if tag
. tabulate om2 relief if tag
```

om_relief	relief: Deed	relief: 2	Total
1	45	26	71
2	28	27	55
Total	73	53	126

Concluding remarks

- `xtcluster` can be used to either assess the assumption of slope homogeneity in linear short panel data models, or to discover potential heterogeneous clusters (an exploratory data approach).

Concluding remarks

- `xtcluster` can be used to either assess the assumption of slope homogeneity in linear short panel data models, or to discover potential heterogeneous clusters (an exploratory data approach).
- `xtcluster` is computationally intensive so it can be very slow with large data. We are working on increasing computational efficiency.

Concluding remarks

- `xtcluster` can be used to either assess the assumption of slope homogeneity in linear short panel data models, or to discover potential heterogeneous clusters (an exploratory data approach).
- `xtcluster` is computationally intensive so it can be very slow with large data. We are working on increasing computational efficiency.
- We also plan to do the following:
    - Incorporate other methods for obtaining the initial partition, such as on the basis of individual-specific estimated parameters.
    - Extend the application of `xtcluster` and the MIC to other estimators and other objective functions.
    - Allow for more complex factor structures in the error term.

MEAFA forthcoming Stata training

- Google the keyword '**MEAFA**' for forthcoming Stata workshops:

- 28 Sep 2015: **Text Analysis** by Normand Peladeau, Provalis Research.
- 14-16 Dec 2015: **Microeconometrics for Count Data** by Colin Cameron, University of California at Davis.
- 15-19 Feb 2016: **Panel Data Analysis: Linear, Dynamic, Nonlinear, Mixed, Hierarchical** by Vasilis Sarafidis, Monash University.

- Feb 2016: 2 more MEAFA PhD top-up research scholarships.