Estimation of Normal Mixtures in a Nested Error Model with an Application to Small Area Estimation of Welfare

Chris Elbers (VU University Amsterdam)
Roy van der Weide (World Bank)

SAE 2013 Conference, Bangkok
3 September 2013

The income equation

Consider the income equation underlying poverty mapping

$$y_{ah} = x_{ah}\beta + e_{ah} = x_{ah}\beta + u_a + \varepsilon_{ah}$$

with a: area (or cluster), h: household, x_{ah}: household and area characteristics, u_a: area r.e., ε_{ah}: hh-level effect. The error components u_a and ε_{ah} are independent.

ELL involves massive out-of-sample imputation of this equation, using the (then) new availability of census information at unit record level:

$$\{x_{ah}\} \rightarrow \{\hat{y}_{ah}\} \cdots \rightarrow \hat{W}_a(\{\hat{y}_{ah}\})$$

Some history

Motivation: do better than ad hoc setting of welfare indicators.

Some history

Motivation: do better than ad hoc setting of welfare indicators.

We thought this procedure would be more convincing if

- minimal assumptions on u and ε
- allow for heteroskedasticity, at least for ε_{ah}
  and (not pooling too much across survey strata)

while

Some history

Motivation: do better than ad hoc setting of welfare indicators.

We thought this procedure would be more convincing if

- minimal assumptions on u and ε
- allow for heteroskedasticity, at least for ε_{ah}
  and (not pooling too much across survey strata)

while

- ignoring that samples have information on u_a for some areas,
- not being very familiar with the SAE world

Introducing EB in ELL

New developments led us to reconsider this setup:

- EU project of poverty mapping for new EU members (specifically requesting EB)
- Molina/Rao article showing the potential for EB in poverty mapping

So the main question to answer was

How implement EB in poverty mapping, and avoid assuming too much about distributions of u and ε?

Introducing EB in ELL

New developments led us to reconsider this setup:

- EU project of poverty mapping for new EU members (specifically requesting EB)
- Molina/Rao article showing the potential for EB in poverty mapping

So the main question to answer was

> How implement EB in poverty mapping, and avoid assuming too much about distributions of u and ε?

A first stab at parameter-free EB

Implementing EB is not difficult once p_ε and p_u are known. For the average disturbance:

$$\bar{e}_a = u_a + \bar{\varepsilon}_a,$$

so that if (target area a is in the sample), by Bayes rule:

$$p(u_a|\bar{e}_a) \propto p(\bar{e}_a|u_a)p_u(u_a) = p_\varepsilon(\bar{e}_a - u_a)p_u(u_a).$$

With distributions on the RHS known, MC techniques are available to draw u_a from its conditional distribution.

A first stab at parameter-free EB

Implementing EB is not difficult once p_ε and p_u are known. For the average disturbance:

$$\bar{e}_a = u_a + \bar{\varepsilon}_a,$$

so that if (target area a is in the sample), by Bayes rule:

$$p(u_a|\bar{e}_a) \propto p(\bar{e}_a|u_a)p_u(u_a) = p_\varepsilon(\bar{e}_a - u_a)p_u(u_a).$$

With distributions on the RHS known, MC techniques are available to draw u_a from its conditional distribution.

In a typical sample survey $\bar{\varepsilon}_a$ is approximately normal (CLT) so to recover u_a we need only filter 'noise' $\bar{\varepsilon}_a$ from 'data' $\bar{e}_a$. With p_u known, p_ε can be recovered. Note that variance of $\bar{\varepsilon}$ can be estimated easily.

No FFT for ELL

This filtering is routinely done in engineering (e.g. FFT) but requires many data points.
Solution: yield somewhat on the goal to be parameter free:

- Assume (reluctantly) homoskedasticity of ε_{ah}
- Assume that distributions of ε and u can be described well by normal mixtures

No FFT for ELL

This filtering is routinely done in engineering (e.g. FFT) but requires many data points.
Solution: yield somewhat on the goal to be parameter free:

- Assume (reluctantly) homoskedasticity of ε_{ah}
- Assume that distributions of ε and u can be described well by normal mixtures

Then:

- Estimate NM parameters for u using average cluster disturbances ($\bar{e}_a$), assuming that $\bar{\varepsilon}_a$ is normal (with variance that can be estimated; e.g. Henderson-3)
- With distribution of u known, estimate distribution of ε_{ah}
- Conditional distribution $p(u_a|\bar{e}_a)$ can now be obtained as outlined above, but can in fact be done much easier.

Normal mixtures

- If v is distributed as normal mixture its density function is of the form

$$f(x) = \pi_1 \varphi(x; \mu_1, \sigma_1) + \cdots + \pi_k \varphi(x; \mu_k, \sigma_k),$$

  with k the number of components of the NM, and $\varphi(x; \mu, \sigma)$ the normal density function
- Distributions can be approximated arbitrarily closely by normal mixtures
- Smooth distributions need only few components for good approximation
- *Big Bonus*: if x and y are jointly mixed normal, then $x|y$ and $y|x$ are also mixed normal and the parameters of these are closed expressions in terms of the parameters of x and y's distributions. This helps tremendously in estimation and in EB application.

Normal mixtures, continued

- Estimating normal mixtures is a very common task, mostly using the EM algorithm. However, the 'noise' $\bar{\varepsilon}_a$ (when deriving the distribution of u) or u (when deriving the distribution of ε) is a complication
- Cordy and Thomas (1997) derive EM algorithm for this situation assuming that component distributions are known
- Extension to include estimation of component parameters (μ_c, σ_c) is straightforward but leads to convergence problems

The EM algorithm for the deconvolution problem

If (for our case) random variable $\bar{e}_a$ is distributed as a normal mixture it can be represented as

$$\bar{e}_a = z_{a1} y_{a1} + \cdots + z_{ak} y_{ak},$$

where the z_{ai} are mutually exclusive binary indicators (over i) with $P[z_{ai} = 1] = \pi_i$ and the y_{ai} (independent, and independent of the z indicators) have density

$$\varphi(y; \mu_i, \sqrt{\sigma_i^2 + s_a^2}).$$

s_a^2 is the variance of $\bar{\varepsilon}_a$ (and could depend on a).

The EM algorithm alternates between (i) integrating out the indicator variables from the loglikelihood of $(z_{11}, ..., z_{Ak}; e_1, ..., e_A)$, using the current parameter values (de π, μ, σs); and (ii) maximizing the result with respect to these parameters to get the next update. (Until convergence.)

Adjusted EM algorithm for the deconvolution problem

- We think that it is the presence of s_a^2 in $\varphi(y; \mu_i, \sqrt{\sigma_i^2 + s_a^2})$ which complicates this scheme considerably
- We dealt with this problem by treating the $\bar{\varepsilon}_a$ as latent variables like the z_{ai}, but only integrating them out after deriving the first order conditions of the likelihood optimization
- Effectively, the maximization step of the EM algorithm is split in two parts. The full cycle is
  (i) integrate out zs from LL given current parameter values;
  (iia) compute expected FOCs (by integrating out $\bar{\varepsilon}$ given current parameter values); (iib) solving 'expected' FOCs with respect to parameters

Adjusted EM algorithm, cont'd

- Bonus: for the second problem of estimating the normal mixture distribution of ε_{ah}, once the distribution of u is known, the same method solves the problem of dependency of the e_{ah} from the same area (which makes it impossible to derive the LL)
- Cheating: for this second problem we make the further assumption that $p(u_a|e_{a1}, ..., e_{a,n_a}) \approx p(u_a|\bar{e}_a)$

Adjusted EM algorithm, cont'd

- Bonus: for the second problem of estimating the normal mixture distribution of ε_{ah}, once the distribution of u is known, the same method solves the problem of dependency of the e_{ah} from the same area (which makes it impossible to derive the LL)
- Cheating: for this second problem we make the further assumption that $p(u_a|e_{a1},...,e_{a,n_a}) \approx p(u_a|\bar{e}_a)$
- To do: *proof formally that after convergence the procedure leads to ML estimators*

The Adjusted EM algorithm: from the paper

The fixed-point solution to the following set of iterative equations yields the estimator $(\hat{\pi}_i, \hat{\mu}_i, \hat{\sigma}_i^2)$ for $(\pi_i, \mu_i, \sigma_i^2)$ for $i = 1, \ldots, m_u$:

$$\hat{\pi}_i^{(k+1)} = E\left[\sum_a \hat{\tau}_{ai}^{(k)}(\bar{\varepsilon}_a)|\bar{e}_a; \hat{p}^{(k)}(\bar{\varepsilon}_a|\bar{e}_a)\right]/A \quad (1)$$

$$\hat{\mu}_i^{(k+1)} = \frac{E\left[\sum_a \hat{\tau}_{ai}^{(k)}(\bar{\varepsilon}_a)(\bar{e}_a - \bar{\varepsilon}_a)|\bar{e}_a; \hat{p}^{(k)}(\bar{\varepsilon}_a|\bar{e}_a)\right]}{E\left[\sum_a \hat{\tau}_{ai}^{(k)}(\bar{\varepsilon}_a)|\bar{e}_a; \hat{p}^{(k)}(\bar{\varepsilon}_a|\bar{e}_a)\right]} \quad (2)$$

$$\hat{\sigma}_i^{2(k+1)} = \frac{E\left[\sum_a \hat{\tau}_{ai}^{(k)}(\bar{\varepsilon}_a)\left(\bar{e}_a - \bar{\varepsilon}_a - \hat{\mu}_i^{(k+1)}\right)^2 |\bar{e}_a; \hat{p}^{(k)}(\bar{\varepsilon}_a|\bar{e}_a)\right]}{E\left[\sum_a \hat{\tau}_{ai}^{(k)}(\bar{\varepsilon}_a)|\bar{e}_a; \hat{p}^{(k)}(\bar{\varepsilon}_a|\bar{e}_a)\right]}, \quad (3)$$

with:

$$\hat{\tau}_{ai}^{(k)}(\bar{\varepsilon}_a) = \frac{\hat{\pi}_i^{(k)}\varphi\left(\bar{e}_a - \bar{\varepsilon}_a; \hat{\mu}_i^{(k)}, \hat{\sigma}_i^{2(k)}\right)}{\sum_i \hat{\pi}_i^{(k)}\varphi\left(\bar{e}_a - \bar{\varepsilon}_a; \hat{\mu}_i^{(k)}, \hat{\sigma}_i^{2(k)}\right)}, \quad (4)$$

where φ denotes the normal probability density function, and where the expectations are taken over $\bar{\varepsilon}$ conditional on $\bar{e}_a$ using the iteration-k estimate of the conditional density function $\hat{p}^{(k)}(\bar{\varepsilon}_a|\bar{e}_a)$.

Adjusted EM algorithm, experience

- Still needs some computational short cuts (interpolation) to make this work
- In simulations, recovering the distribution of ε works very well (perhaps because of generally good signal-to-noise ratio, $\mathrm{var}(\varepsilon)/\mathrm{var}(u)$ and the many data points: the sample size)
- Recovering distribution of u is not as striking, but still OK (perhaps because of less favorable signal-to-noise ratio, $\mathrm{var}(u)/\mathrm{var}(\bar{\varepsilon})$ and much less data points: the number of areas/clusters)
- Identification problems when estimating normal mixtures (number of components) are less important in this context

Adjusted EM algorithm, experience

- Still needs some computational short cuts (interpolation) to make this work
- In simulations, recovering the distribution of ε works very well (perhaps because of generally good signal-to-noise ratio, $\mathrm{var}(\varepsilon)/\mathrm{var}(u)$ and the many data points: the sample size)
- Recovering distribution of u is not as striking, but still OK (perhaps because of less favorable signal-to-noise ratio, $\mathrm{var}(u)/\mathrm{var}(\bar{\varepsilon})$ and much less data points: the number of areas/clusters)
- Identification problems when estimating normal mixtures (number of components) are less important in this context
- We are close to answering the original research question

EB application

Since normal mixtures are closed under taking conditional expectations implementing EB in poverty mapping is particular fast and straightforward. A comparison with EB assuming normality (Molina/Rao) suggests however that the gains are minor (see Roy's presentation yesterday).

To get appreciable improvement an atypically high (relative) variance $\mathrm{var} u$ is required. This may be a common situation in SAE, but in poverty mapping we never find it. The reason could be that we tend to include many variables in our regression that seem to capture location effects quite effectively.

Conclusions

- Using normal mixtures for modeling error components make it possible to implement EB in poverty mapping, without making distributional assumptions on error components
- Experience so far suggests that mixture distributions can be estimated using a modified EM algorithm
- Given that location effects are relatively small in poverty mapping (conditional on regressors) scope for improving on 'normal' EB is not big; (and EB in general is likely to play only a modest part in poverty mapping)
- Deviations from normality of ε distribution could turn out to be more important
- The procedure can in principle be extended to multiple levels of nesting but becomes the computational burden increases fast (curse of dimensionality)

Conclusions, cont'd

Getting back to the original equation

$$y_{ah} = x_{ah}\beta + e_{ah} = x_{ah}\beta + u_a + \varepsilon_{ah},$$

if we want to improve poverty maps we should focus on the systematic part $x_{ah}\beta$, rather than the disturbance part $u_a + \varepsilon_{ah}$. There are plenty of new data sources and prediction techniques that could be explored for that.

Poverty mappers should focus more on improving the information base for the structural part of income regressions