AMM Tutorial 5

Fix your randomness

Xinran Hu

August 21, 2024

1 Random Effects vs. Fixed Effects

These are two approaches you can use to deal with the unobserved individual heterogeneity a_i when there are more than two periods in the data. They basically both try to transform the original regression equation so that a_i gets cancelled out.

Let's start with the random effects model because it's the less realistic scenario. Random effects deal with unobserved heterogeneity ONLY WHEN a_i is completely uncorrelated with any of the X variables. This is usually a pretty strong assumption, as a_i refers to something we cannot observe and hence does not know very well about. In this scenario, you can more or less comprehend the situation as merging a_i with ε_{it} is problematic as the resulting $v_{it} = a_i + \varepsilon_{it}$ would be serially correlated as it has a time invariant component for each i.

To fix this, we can execute an *ad hoc* transformation on the data so that the transformed equivalent of error term v_{it} would no longer be correlated along the time dimension.

And you might realize why this is less realistic - Given that we cannot observe a_i anyway, how can we have the confidence of saying that a_i is not correlated with the X variables at all?

Relaxing this assumption requires the use of the fixed effects model, which fortunately is conceptually more straightforward than the random effects model. Fixed effects model can be interpreted as a generalization of the first difference model, which collapses a two-period dataset into a "cross-section" by taking the difference between the two periods. Fixed effects models do the same collapsing of data, but trying to use all periods at the same time - which is why we are subtracting means of variables from each observation.

One problem with the fixed effects estimator is that any variable that does not vary across time would be cancelled out in this transformation, which is unfortunate but inevitable. On the other hand, random effects model allows inclusion of such variables at the cost of a very strong assumption.

If this still doesn't sound straightforward enough for you, another way to comprehend the fixed effects model (which sounds slightly dumber but I somehow preferred this back when I was a student) is to envision it as we are trying to estimate a_i for all i as if it's a whole set of dummy

variables.

2 (Actual) Panel Regressions with Stata

We already know that we can let Stata know that it's dealing with a panel dataset with the `xtreg` command, which allows us to generate first differences automatically with the `D.` operator. There should have something else that we can do... Right?

Right! I did not go into the very detail of the maths Michelle should have covered in the lecture, but you probably remember that there are a lot of calculations to do if you want to estimate a fixed/random effects model. Luckily, we can avoid all these trouble by running regressions using the `xtreg` command on a dataset that has been declared to be panel. The inputs for `xtreg` is largely identical with `reg`, as long as you specify the model by including `fe` or `re` option all the time.

Once you run a panel regression, you'll notice that there are 3 extra statistics reported at the end of the output table: `sigma_u`, `sigma_e`, and `rho`. These terms are not super straightforward as we used a different set of notation than what Stata is implying:

$$Y_{it} = \beta_0 + \beta_1 x_{1,it} + \beta_2 x_{2,it} + \cdots + a_i + \varepsilon_{it}$$

Under this set of notation that we've been using, `sigma_u` refers to σ_a, `sigma_e` refers to σ_ε, and `rho` refers to $\sigma_a^2/(\sigma_a^2 + \sigma_\varepsilon^2)$. These things don't look super useful at first glance, but a rule of thumb we can use is that if `rho` is close to 1 then random effects and fixed effects models would give similar results.

A sketch of proof for the rule of thumb is as $\sigma_a^2/(\sigma_a^2 + \sigma_\varepsilon^2) + \sigma_\varepsilon^2/(\sigma_a^2 + \sigma_\varepsilon^2) = 1$, $\sigma_a^2/(\sigma_a^2 + \sigma_\varepsilon^2)$ being close to 1 means that $\sigma_\varepsilon^2/(\sigma_a^2 + \sigma_\varepsilon^2)$ is close to 0, which could only be that σ_ε^2 is really small compared with σ_a^2. Therefore, if we multiply σ_a^2 by $T > 1$, σ_ε^2 will take up an even smaller proportion in the sum between the two terms (i.e. The ratio would be even closer to 0), which means that λ on slide 29, week 5 would approach 1 and we'll end up doing almost the same transformation for both models.

3 A Petty Stata Trick

Note that 1) we want to run the regressions for this week with all the year dummies included; 2) the year dummies are named in a consistent manner. This sounds like a setting that would need a productivity hack - Is there a way to call several variables with a tiny bit of code?

Well, in Stata you can include all variables that share the same first several characters in their names by typing in the shared part of variable names, followed by a `*` immediately. As long as you are sure that no other irrelevant variables would be included by mistake, this trick can be game changing at times.

Another way of including several dummy variables in a compact manner, albeit less preferred, is by using the `i.` operator in Stata. Similar to the `D.` operator, adding `i.variable` to a regression would result in Stata generating the requested variables temporarily for the regression. In the case of `i.`, Stata would treat the variable specified as categorical and generate a dummy variable for each unique value taken.

You might start to see why `i.` is less preferred - including all scenarios for the dummy variable will definitely create a dummy variable trap, and we'll have no control over which value Stata decide to drop. Therefore, if you have a strong preference over which group should be set as the reference group then it's definitely better to do it manually.