Final Project Quantitative Economics

Juan Martin Morelli

June 2, 2016

1 Introduction

Empirical evidence shows that spreads are highly synchronized bewteen countries. Can that be explain by a standard model of default risk with synchronized outputs, where the synchronization is estimated from the data? Out prior is that it cannot. The expected contribution of the paper is to explain contagion effects among Emerging Markets, due to negative shocks to common international intermediaries[1].

The first step is to study if the comovement observed in the data between credit spread in EMs can be explained by synchronized income processes between these countries. Standard default models such as Arellano (2008) assume that credit risk is induced by countries' fundamentals, measured by income and the level of debt. Default probabilities increase with the level of debt, and decrease with income. Thus, if inome is synchronized among countries, then we should see a significant correlation among spreads.

The way to analyze this is by modelling GDP growth with both, a common systemic component and an idiosyncratic component. This systemic component is common to all countries in the sample, hence it is an aggregate shock. For this, we run a panel regression that allows us to estimate the following equation for both, GDP and spreads:

$$x_{it} = \alpha_i^x + \Delta_t^x + \epsilon_{it}^x, \quad x = y, sp$$

where α_i^x is country fixed effects, Δ_t^x is time fixed effects and ϵ_{it}^x is an idiosyncratic shock.

The previous regression over GDP allows us to obtain estimated series for the time dummies $\left\{\hat{\Delta}_t^y\right\}$ (systemic process), and for the idiosyncratic components $\hat{\epsilon}_{it}^y$.

Then we can estimate an AR(1) for the systemic process:

$$\hat{\Delta}_t^y = \rho_y \hat{\Delta}_{t-1}^y + \epsilon_t^y$$

and through some dynamic panel estimator, we can run a regression for

[1]The paper is being elaborated jointly with Pablo Ottonello (University of Michigan) and Diego Perez (NYU)

$$\hat{\epsilon}_{it} = \rho_\epsilon \hat{\epsilon}_{it-1} + \nu_{it}^{\epsilon}$$

Then we have the estimators $\hat{\rho}_y$, $\hat{\rho}_\epsilon$ and $\hat{\sigma}_v^2$. It is important to note that these are second step estimators, so we have to make corrections if we want to get their standard errors. Also note that we are assuming a common ρ_ϵ and σ_ν. With the triplet $\left(\hat{\rho}_y, \hat{\rho}_\epsilon, \hat{\sigma}_v^2\right)$ we can now generate simulations for the GDP process.

We plan to solve Arellano (2008) accounting for 3 states instead of 2: $(B_{it}, \Delta_t, \epsilon_{it})$. We also plan to improve the solution algorithm that is currently used, in order to obtain smooth policy and pricing functions. Splines or Chebychev polinomials should work. As a matter of fact, Hatchondo, Martinez and Sapriza (RED, 2008) have shown that a discrete state space (DSS) technique with evenly spaced grid points needs a large number of grid points to avoid generating spurious interest rate movements. The DSS method converges to the collocation method for a large number of gridpoints. In particular, the dispersion of spreads is significantly reduced when the number of enwodment gridpoints is increased, but not so when the number of gridpoints for assets is increased. Interestingly, the authors argue that Arellano's algorithm can be improved in terms of speed by updating simultaneously both, the value function and the prices, just as it is done in QuantEcon.

We plan to simulate Δ_t^y for some T and ϵ_{it} for the same T and $N = 53$ countries. Then we get the spreads and effective GDP as an output, and compute a panel regression with time fixed effects. If the model is correct, then we should see syncronization in spreads. We expect to find that this is not the case, contradicting the empirical evidence, and motivating the modelling of financial intermediaries and the existence of contagion effects.

In what follows we first try to replicate Arellano (2008) and improve the solution method. In particular, we replicate Table 4 from the Arellano (2008) paper, try the collocations solution method, then use VFI with HIS and parallelization. Once the paper is replicated, we extend the model to three states with what we found to be the best solution method in the original Arellano (2008).

2 Arellano (2008) with Two States

2.1 Replication of Arellano (2008)

The approach of the current project was to first replicate the Arellano (2008) paper. The purpose is to further understand the model, and to make sure that the base model works correctly, before stating our findings. The main equations of the Arellano model are:

$$\begin{aligned} V_c(b,y) &= \max_{b' \geq -Z} \left\{ u(c(b')) + \beta \int V(b',y')P(y,y')dy' \right\} \\ V_d &= u(h(y)) + \beta \int \left[\theta V(0,y') + (1-\theta)V_d(y')\right] P(y,y')dy' \\ V(b,y) &= \max\left\{V_c(b,y); V_d(y)\right\} \end{aligned} \tag{1}$$

The government is going to default when $V_c(b,y) < V_d(y)$, and the probability of having default next period is:

$$\delta(b', y) \equiv \int \mathbf{1}\left\{V_c(b', y') < V_d(y')\right\} P(y, y') dy' \tag{2}$$

Finally, the zero profit condition implies:

$$q(b', y) = \frac{1 - \delta(b', y)}{1 + r} \tag{3}$$

There already is a code on QuantEcon webpage that does most of the job. It improves the Arellano code by updating the prices at each iterative step, instead of doing it after the value function converges (and updating the value function once again)[2]. As in an extension in the webpage, we added more nodes than the original Arellano: 551 for bonds and 51 for output. Then we replicated figures 3 and 4, as well as table 4. The jupyter file "Arellano_original" has the output. With the caveat of having more nodes, the current code seems to be replicating quite satisfactorily bonds prices, the savings function and the value functions.

However, the interest rates cannot be replicated. In particular, we get that interest rates in the high state can never reach the same high values as rates in the low state do (we do not mean here curves crossings, just values). Furthermore, our (annualized) interest rate reaches a much higher value under the low state than what is presented in Arellano (2008). This can be because the non-default set is broader under the current code than what Arellano (2008) has. Note that the plots are trimmed because of default events occurring. As will be mentioned in the HIS section, interest rates are highly sensitive to the choice of nodes.

Later on we used the optimal policies and consequent equilibrium prices to simulate the economy. The first part of the simulation section of "Arellano_original" shows a single realization, just to check behaviour. The following part follows Arellano (2008) methodology. The authors simulates the economy for some horizon T, captures 100 default episodes such that the history between defaults is of 74 periods of more, and pre-default statistics are computed using those 74 periods.

Following Table 4, we computed the spread, trade balance, fall in consumption and fall in output one period just before default, and took the mean among those 100 default samples. We also computed their standard deviation and correlations with output and interest rates.

In addition, we computed the mean debt in percentage of output and the mean spread for the pre-default 74 periods samples. Finally, we calculated the output deviation in default by computing the deviation of the mean output during default from the mean output pre-default for each default sample, and took the mean of those.

2.2 Arellano (2008) with Collocations

The next step was to try using collocation techniques in the Arellano (2008) model. The reason is to avoid adding a large number of gridpoints, specially since we will later add state variables. Recall the mentionting on

[2]See Hatchondo, Martinez and Sapriza (RED, 2008) for further discussion. They argue that without interpolations a large number of nodes is needed for the model to perform well. Indeed much more nodes are needed than those used originally in Arellano (2008).

the ill-behaviour of the model with few nodes for the VFI solution method. The translation to the collocations version implies the following system of equations:

$$\begin{aligned}
\Phi(b,y;\omega_c) &= \max_{b'\geq -Z}\{u(c(b')) + \beta\Phi(b',y;\omega_e)\} \\
\Phi(y;\omega_d) &= u(h(y)) + \beta\sum_{y'\in Y}\Pi[y,y']\left[\theta\Phi(0,y';\omega_c) + (1-\theta)\Phi(y';\omega_d)\right] \\
\Phi(b,y;\omega_e) &= (\Pi\otimes \mathbf{I_{N_b}})\max\{\Phi(b,y';\omega_c);\Phi(y';\omega_d)\} \\
V(b,y) &= \max\{\Phi(b,y;\omega_c);\Phi(y;\omega_d)\}
\end{aligned} \tag{4}$$

Note that $\Phi(b,y;\omega)$ is an abbreviation for $\sum_{j\in N_y N_b}\phi(b,y)_j\omega_j$, where $\phi(b,y)$ is a row vector.

The advantage of separating the value function from its expected value is that it greatly increases the computation speed, since we do not have to compute the expecation for each optimization iteration. We can solve V and V_e separately and then solve the system.

We are going to choose $N_y = 10$ collocation nodes for the income process and $N_b = 10$ collocation nodes for the level of debt. Then the set of collocation nodes is $\mathbf{s} = [\mathbf{1}_{N_y}\otimes \mathbf{b}, \mathbf{y}\otimes \mathbf{1}_{N_b}]$, so that $\mathbf{s}$ is of dimension $N\times 2$ with $N = N_y\times N_b$.

We chose linear splines for both, income and bonds. For better understanding, we write down the problem in matrix form. All the variables are now vectors. Let Φ_x be the basis structure for variable x.

$$\begin{aligned}
\Phi(s)\omega_c &= \max_{b'\in Z}\{u(c(b')) + \beta\Phi([b',y])\omega_e\} \\
\Phi_d(y)\omega_d &= u(h(y)) + \beta\Pi\left[\theta\Phi([0,y'])\omega_c + (1-\theta)\Phi_d(y')\omega_d\right] \\
\Phi(s)\omega_e &= (\Pi\otimes \mathbf{I_{N_b}})\max\{\Phi([b,y'])\omega_c;\mathbf{1_{N_b}}\otimes\Phi_d(y)\omega_d\}
\end{aligned} \tag{5}$$

For the second equation, note that $[0,y]$ is just the block of the state vector in which $s = (b=0,y)$, which is of size $N_y\times 1$. Then, $\Phi([0,y])$ is of size $N_y\times N$ and $\Phi([0,y])\omega_c$ is of size $N_y\times 1$. Then L is of size $N_y\times 1$. Note that we also had to use "Interpolations.jl" since the pricing kernel depends on tomorrow's bonds b'.

The way to solve by iteration is:Given (ωc,ωd), get ωd' and ωe' and with ωe' get ωc'

1. Guess some initial weights (ω_c,ω_d)
2. Get ω_e' from the third equation of collocations.
3. With ω_c and ω_d, get ω_d' from the second collocations equation.
4. With ω_e', get ω_c' from the first collocations equation.
5. Use output as a new guess.

With 15 knots for y and 15 for B, the computation time was 8 seconds (the same as the original Arellano). However, there was no significant improvement over the original Arellano (2008) code[3]. On the contrary,

[3] We also tried using the initial guess (ω_c,ω_d) to get ω_d'. With ω_d' and ω_c get ω_e', and with the latter get ω_c'. There were no significant changes.

results are worse. The pricing kernel is too discrete, and if we try adding more knots, then we do not get convergence. There must be something wrong in this solution by collocations. The code is in the jupyter file "Arellano_collocation_approach2". If we try quadratic or cubic splines for bonds, then there is no convergence.

2.3 Arellano (2008) with HIS and Parallelization

Since we could not get nice results with the collocations method, we decided to go on and try the VFI method with HIS algorithm. The original Arellano code (QuantEcon version) was improved in terms of speed. For instance, when using 51 gridpoints for output and 551 for bonds, it took 24 seconds with the HIS algorithm and 86 without it.

Keeping the HIS, we tried different specifications for the gridpoints, and ran simulations in order to replicate table 4 from Arellano (2008). As in the paper, we kept 100 samples of default to compute the statistics. If we used Arellano's suggestion (21 for output, 200 for bonds), then we get spreads in default of 58 and falls in output of consumption and output of 7-8 percent. The standard deviations are matched for all but spreads, which are too high (16 instead of 6). The correlation between output and consumption and trade balance is matched, but the correlation between spreads and output is -51% instead of -29%. The correlation between spreads and consumption is overestimated and that between spreads and trade balance is underestimated, when comparing with Arellano. The rest of the statistics in table 4 are more or less the same, except for the mean spread, which is twice as high as that from Arellano.

Next we tried adding gridpoints to the income process. In particular, we set $n_y = 102$ and $n_b = 200$. The pricing functions became smoother, but spreads expode. The spread in default is 448 and the standard deviation is 67. This is strange since in Hatchondo et al they argue that adding gridpoints to income reduces the standard deviation of spreads! What we infer is that adding income gridpoints increases the mass in lower values of income, thus increasing the probability of reaching states where bond prices are near zero. In the original Arellano paper the price function q does reach values near to zero, but it seems that in her simulations those states were hardly ever visited.

We then tried adding more gridpoints in bonds instead (i.e. $n_y = 21$ and $n_b = 600$). Hatchondo et al argue that will have no significant effects in results. We found the same results. Note that we wrote down a parallelized version for the cases where we have a large number of gridpoints (for a small number of gridpoints it is inefficient to parallelize). Also, for the original Arellano model we used the Tauchen from QuantEcon but for all other variations we computed out own Tauchen function.

Following Hatchondo et al, when using additional gridpoints we tried adding samples of default. In particular, we took 1500 samples (Nsamples). Results are improved in the sense that spreads are reduced, but so is the fall in output and consumption. When we try 1000 gridpoints for output and 2000 for bonds, spreads in default are 90 (annualized), fall in output and consumption is 3%-4%.

All in all, we are able to replicate Arellano (2008) following her methodology, but we cannot get good results when adding more gridpoints.

3 Arellano Economy with Three States

In this stage we decomposed the income process between a systemic component Δ_t and an idiosyncratic component ϵ_{it}. Hence we have three states $(b_t, \Delta_t, \epsilon_{it})$, as described in the introduction. We first had to provide the autocorrelation coefficients and standard deviations for the Tauchen of the processes Δ_t and ϵ_{it}. In both cases we assumed the process follows an AR(1).

For the systemic component Δ_t we got the estimations of the time fixed effects from a panel regression $(\hat{\Delta}_t)$, and then just performed a second stage AR(1) estimation to get the autocorrelation of $\hat{\Delta}_t$. This wat we got $\hat{\rho}_\Delta$. Then we computed the standard deviation of the process by computing the standard deviation of the estimated residuals of the AR(1) regression.

As for the idiosyncratic component ε_{it}, we had to run a dynamic panel regression (Arellano-Bond). Hence, apart from assuming an AR(1) process, we assumed the innovations to ε_{it} are common among all countries. Hence our assumption is that, filtering the countries fixed effects, output in all countries come from the same distribution. In addition, we assume that the idiosyncratic and the systemic processes are independent. Then we can compute the Tauchen for each of them, get the marginal Markov transition matrices and then the joint matrix is just the kroenecker product between the two matrices.

We solved the model with and without HIS (to check speed and robustness). Once again the HIS improved speed while being robust. So we kept it for the next stage, and increased the number of grid points to have 100 on income and 551 on bonds.

Given the increased number of grid points, we parallelized the code by two different approaches (outer loop on output vs inner loop on current bonds). The fastest was on the outer loop on income. Adding procs from 4 to 8 did not improve speed. We increased the number of grid points on income to 12x12 = 144 and of bonds to 800. It took similar time with 4 and 8 procs, so we kept it to 4. The final code is "Arellano_3states_parallelized.ipynb".

The next step would be to simulate and compute the statistics under this new specification, in the same fashion as Arellano (2008). Of course, this time we are not targeting Argentina but a pool of countries. So we would need to compute the same empirical statistics as in Table 1, but this time for the average country. If we get similar results in the simulations, then we could procede to run simulations for 53 countries, and get the series for output and spreads. With those we plan to run the same panel regressions as in the data, and check the goodness of fit. We expect spreads not to be syncronized even when all countries are subject to the same realization of Δ_t (i.e. low R2), suggesting the existence of contagion of risk.

Since we still did not compute the empirical statistics for the representative country, we just set up the structure to run the simulations and save the output for the panel regression.

4 Conclusion

In this project we managed to replicate Arellano (2008). We also tried to repliate results form Hatchondo et al (2008). Following the latter we tried to solve the model by collocation methods, but results were not satisfactory. Then we tried to add gridpoints to the discrete method, which required implementing an HIS algorithm and parallelization to increase speed. Results were not convincing when adding gridpoints. Furthermore, collocation methods are deseired, so we need to solve this problem.

Next we changed the state space of Arellano (2008) to allow for both, a systemic component and an idiosyncratic one. Even though the code is set up, we need to do some fine tuning on the empiricical panel regression and compute average statistics to later compare with average statistics coming out from the simulations.

Once these issues are resolved, we need to expand the model to allow for intermediaries risk premia as an additional state variable.