Spatial Statistics and Spatial Econometrics
Prof. Gaurav Arora
Department of Social Sciences and Humanities
Indraprastha Institute of Information Technology, Delhi

Lecture - 19B
Spatially Lagged Variables in Regression Models

So, welcome back to the second part of the 19th lecture. We have now seen in detail the spatial lag model including its interpretations and impact on inference.

So, we are now going to move to this next popular form of a spatial regression model called the spatial error model, or even called the spatial autoregressive error model. Those of you who are used to time series notation will be able to connect with it. So, in short, it is called the SAR model.

So, I am going to start with motivation, I am going to motivate this model before I get into the technical details of it. So, first is that this SAR model represents a situation where spatial dependence in the error term arises due to omitted random factors.

So, some random factors explain why we omit them and they are now a part of the error term, that is why this type of phenomenon or this term is also defined as a nuisance. It is a nuisance spatial dependence term.

And, then, it is employed in situations that are characterized; this is an important point characterized by a mismatch in spatial process with the spatial scale of observation. So, if you are studying a process, a population process, or a natural process which is very high resolution in nature, the data that you are aggregated in nature, right? Then, such situations will be characterized by nuisance spatial random error.

Because there is no way for me to account for the spatial process at the resolution that it actually occurs. I have data for something at the district level, and at the state level, but the actual process for example, crime. Now, the spatial resolution of how crime is observed might be much finer in terms of the data that are actually recorded than the data that are in the public domain right.

In that situation, such a mismatch can never be accounted for in a model. So, it is going to enter into the error term as a random disturbance, a nuisance term right? And, you can imagine this type of categorization is quite general, I mean this is something that we encounter a lot of the time and that is by spatial error model happens to be a very popular form of the model.

So, I will give you an example, as an example, we can think about combining high resolution let us say land use data, something that we have seen that is that say the Bhuvan data, right? And, let us say we are combining that as a function of something like farmer education or farmer education disparity in terms of gender ratio things like the dropout rates of boys and girls, the education levels, the skill development levels, and so on.

The training that let us say women farmers have versus men farmers. So, in households where I have women heads versus men as heads or male members of the family as the head of the farmers. Then, you may have these systematic differences, then you might want to account for them in terms of how land uses are taken. But, such variables are usually only available at the district level or taluk level, at the block level, and state level whereas, land use let us say with Bhuvan is available at a very high resolution.

So, on the left-hand side, I have a variable that is of the resolution of 50 meters by 50 meters, and on the right-hand side, I have a variable that is at the district level. So, many many pixels of land use are contained within one district and this mismatch is going to cause a random nuisance in my model. And, hence will make the spatial error model really useful. Finally, I want to just say that a spatial error process impacts the efficiency of estimates.

So, earlier we saw that a spatial lag you know version of it, and it actually led to bias in the estimates. Here, I will not have the bias and we have seen the reasons earlier why there is no bias with the spatial error, and spatial dependence in the error term. We saw that with much more restricted form, now we have a much more general form with the spatial lag notation or the notion of spatial lags.

So, having understood this, let us move on to the mathematical treatment of a spatial error model. So, the SAR model can be then written as the following. So, we have a y equals x beta plus u, something that we are very used to. Now, here we have a variance of u given x is not equal to a situation which we are very fond of where the off-diagonal elements are 0 or diagonal elements are exactly the same across all $u_{i's}$.

Here, I am not going to have that situation rather what I am going to have is that covariance of u_i and u_j will not be equal to 0 for some i not equal to j, I wrote this even when we had you know earlier when we had just introduced the SAR model. Now, notation-wise, we can write a note, the variance of u given x, again u is an n by one term. So, the variance by itself is going to be N by N, that is why I have I is N by N, sigma squared is just a scalar so, it is just 1 by 1.

I am just checking y is N by 1, x is N by K, and beta is K by 1. So, I have N by 1 and u is N by 1 so, all consistent. Now, this variance of u given x can be written as expectation u u transpose given x. How is that possible? Well, u is N by 1, and u transpose is going to be 1 by N. So, overall I have a N by N which is the same as the matrix on the left-hand side. What is being used in this definition is the definition of the variance.

So, the variance of u given x, by definition so, little aside I am doing this for you, just in case people not have seen this stuff before. The variance of u given x is going to be expectation u u prime given x minus expectation of u given x the whole square dot product. Now, this term is equal to 0, by assumption 2 of my regression model.

So, you can go back and revise the regression model where we did. So, that is why I can write variance of u given x as just expectation u u prime given x. And, similarly, I can write the covariance as expectation u_i times u_j given x, it is not comma expectation times u_i u_j is not equal to 0 for some i not equal to j. Well, it could also be all ok, I mean it could also be non-zero for all i not equal to j, but at least for some i, some of the off-diagonal elements are non-zero.

So, what this situation, this second condition implies is that the off-diagonal elements of the variance-covariance matrix are non-zero. So, we are working with a non-spherical variance-covariance structure, right? And, I am just using these terms because earlier when I introduced the regression; we have done a review of it, where I discussed all of these.

What are non-spherical errors, and what do we do when we have non-spherical errors, how what is the impact, and how do we reconcile these situations; you have to go back and revisit these things again and again. So, in terms of specification, I am just going to say model specification; what we are saying is that we have y equals x beta plus u where such that u is equal to lambda W u plus e, where this e is distributed normal with 0 mean and a homoscedastic variance-covariance structure.

So, this e is what we are used to when there is no spatial dependence in errors, and the spatial dependence in errors is explained through the weights matrix. So, now, this W u is nothing, but the lag in u because w by itself is an N by N, u is N by 1 that is why this u_L is also N by 1. So, it is a lag just like we defined spatial lag for the outcome variable w y; now, we are working with this W u, right? So, this is the spatial lag of the error term in parent regression.

Why do I say parent, why do I use the term parent regression? Well, I have two, you know you can think of two regression models, one is on the outcome variable y and the second one is on the error term, right? So, u is the error in the first equation, in the second equation u itself is dependent on its lag.

And, what remains is now a random term e which is the homoscedastic error term, right? So, e this implies that e is homoscedastic, and u is heteroscedastic, right? I mean this implies u is heteroscedastic. So, this is all about the model specification.

So, let us rewrite this thing and move forward with our analysis. So, we have y equals x beta plus u such that u is equal to lambda W u plus e, where e is normal multivariate, normal 0 mean sigma squared I_N, very very quickly my dimensions are N by 1, y is N by 1, x is N by K, beta is K by 1. So, I have N by 1 and N by 1, u is N by 1, w is N by N, u is N by 1 so, I have an N by 1, lambda is a scalar, so it is a 1 by 1, e is an N by 1 so, e is N by 1.

So, its mean term, that is the distribution mean, the population mean is N by 1, and the variance-covariance; obviously will be an N by N. So, everything falls in line so far as I understand it. Now, we can rewrite the error structure as under. So, I can say u minus lambda W u equals e which implies I have I_N minus lambda times u equals e right, or as I did earlier, I could pre multiplied this by I_N minus lambda W inverse and I will be able to rewrite u.

So, now my variance-covariance matrix, now our let us see the variance-covariance matrix of u can be written as. So, I am going to now say expectation x so, I am putting the conditional subscript just to have a concise notation u u prime is equal to the expectation of u is I_N minus lambda W inverse e times I_N minus lambda w inverse e the whole thing transpose.

So, I have I_N minus lambda W inversed e and then multiplied by I_N inversed e transposed. So, I am going to rewrite this thing again now. So, I have this thing is, I am going to take the transpose of the second term. So, I am going to just do that. So, I have equal to the expectation I_N minus lambda W, this will be written as is e. Now, this transpose will become e transpose I_N minus lambda W inversed transposed.

So, I have done nothing, I have simply done this. Now, expectation remember is conditional on x. Now, this expectation operator is a linear operator so, it will start to enter matrix form inside. I, lambda and W all are independent of x. So, the expectation of the operator can start to go right in they are degenerate terms.

So, I have this is equal to x I_N minus lambda W inverse expectation e e transpose times I_N minus lambda W inverse transpose. Now, this expectation e e transpose, well first of all this step is because the expectation is a linear operator. And, expectation e e prime is nothing, but sigma squared I_N. This is the variance-covariance structure of the error term, right?

So, I can now move forward and say this is equal to I_N minus lambda W inverse sigma squared I_N minus lambda W inverse the transpose which can be written as sigma squared is a constant, it will come out. And, what I am looking at now is sigma squared times I_N minus lambda W inverse and I_N minus lambda W inverse the whole thing transpose.

So, let us work with its dimensions, I_N is an N by N, W is an N by N, and lambda is just a scalar. So, I am working with an N by N, the inverse of N by N is N by N, and the transpose of an N by N is again an N by N. So, overall it is as if I have sigma squared omega as the variance-covariance structure of the error term u and this omega by itself is an N by N matrix, right?

So, this entity I minus lambda W inverse times its own transpose is what is characterizing the off-diagonal elements and spatial dependence through off-diagonal elements using the weights matrix. So, I am going to write it down in a concise manner, that this term here characterizes spatial dependence through W and lambda. Lambda is to be estimated, but this is the term from where this idea of a spatial lag or notion of spatial lag is leading to a spatial dependence structure in the error variance-covariance matrix.

And, the basic idea is that because I do not have sigma squared I_N and I have sigma squared omega, I can say that errors are heteroscedastic. This is something I am very used to, by now I am very used to in terms of characterizing spatial error processes. So, I have a model, which is y equals x beta plus u such that u is equal to lambda W u which is the u_L term plus e and e is a homoscedastic error with expectation 0.

So, this structure here implied that the variance of conditional variance of u which is also an expectation of u u prime can be written as sigma squared omega with non-zero off-diagonal elements. And, these non-zero off-diagonal elements are characterizing spatial dependence through this interesting term I minus lambda W which is directly sourced from the model. I hope this is crystal clear to all of you.

So, with that, I am going to move to the next step which is the obvious next step by now, is the estimation. I want to now estimate the spatial autoregressive model or the spatial error model. So, my next agenda is to estimate the spatial error model. So obviously, from what I have learned till now, what I am going to do, I am going to apply the FGLS strategy. And, if you have followed through the nodes till now, the FGLS strategy usually works in steps. So, we are going to have steps here. I am going to pronounce this step slowly.

Step 1, what is step 1? Well, luckily very good for us because of the weights matrix characterization, we exactly know the variance-covariance structure. So, we are going to write that down. So, we say, we know the variance-covariance matrix for the error term alright? So, what is the variance-covariance matrix? It is the variance of u given x is equal to sigma squared omega, we have seen the exact form of omega, we have written it down previously.

This thing can also be notated as something that we have used earlier as a matrix sigma, right? We know that I am going to rewrite omega, now sigma inverse if I were to write sigma inverse that will be equal to sigma squared inverse omega inverse. And, you can write as I minus lambda W transpose because you know when we take an inverse, you know you we flip the transpose. So, you have I_N minus lambda W transpose times I_N minus lambda W and a sigma squared is sitting right in front.

Step 2, we are going to find a matrix T such that T prime times T is exactly equal to omega inverse. And, T is a square that is an N by N matrix, it is an arbitrary N by N matrix with this property that is T prime T is going to be omega. Now, it is not so, hard to see that T is simply equal to I_N minus lambda W perfect. So, because I have this readymade solution in step 3 I am going to define transformed dependent and independent variables.

These are very logical steps in FGLS now, by now we have looked at FGLSs in multiple settings. And, I believe that you will have a grip on why these steps are following the path that they are. If you do not, then you have to go back to previous lectures and revise FGLS.

So, I am going to define y^M which is my new; so, I can also instead of calling it M, I am just going to call it y new as T_y and x new as T_x.

Now of course, by definition T is I minus lambda W y and I minus lambda W times x. I have to always check dimensions, I am going to do that y is N by 1, T is N by N. So, y is N by 1 perfect, x is N by K, and T is N by N. So, x new is N by K. And, I just have to keep a note that this lambda we need to estimate this term. So, the straightforward first three steps of the FGLS strategy.

So, let us go and write down step 4. Well, in step 4 we regress y new on x new using the maximum likelihood estimation strategy, this is something we have seen as well in the maximum likelihood we declared the assumption. What was the assumption that was declared in terms of distribution, sorry we declared the distribution of the error of the random structure. What assumption did we start with? Well, we declared that e had a normal distribution.

So, we will start with my maximum likelihood estimation, we begin from there right? So, we know what? We know that e is multivariate normal 0 mean sigma squared I_N variance-covariance, right? Then, the log-likelihood function for estimation is written as ln L beta, lambda, sigma squared y X.

So, I have my parameters of interest, these are all my model parameters of interest. I want to estimate beta, lambda, and sigma squared of course, lambda is my important parameter, the parameter of focus here, right? But, for maximum likelihood estimation, I will be estimating all these parameters, and at my disposal what I have is the data. I have some data and I have an understanding of the population distribution.

I am trying to fit the data using these flexible parameters so, that the model fits the data in the best possible manner. So, through this, my algorithm says you have to maximize the likelihood of observing the sample given the population process. And, in the process, you will estimate beta, lambda, and sigma squared. Remember, beta is a K by 1 so, I have K different parameters. So, overall this means I have K plus 2 parameters to estimate. So, at least that is the size of data strength that I need.

So, if I have more parameters to estimate, more is the data, that I need to do that. And, you can go back and look at any typical standard normal distribution-based maximum likelihood estimation process for a regression. It has a standard analytical form which is the following y new minus x new beta times transpose y new minus x new beta.

Now, interestingly we have to always check dimensions we have the first term as N minus 2, which is a scalar, and N minus 2 log of sigma square is a scalar. So, scalar 1 by 1, scalar, here, I see something in matrix 1. So, y I know is N by 1, x is N by K, and beta is K by 1. So, I have an entity that is N by 1. The transpose of this entity is 1 by 1, 1 by N. It is multiplied by the entity which we have seen N by 1. So, overall I have a scalar. So, it works out well for me.

So, all I am doing is I have a scalar representation on the right-hand side, I have some unknowns that are betas, lambda, and sigma squared. And, I will simply write the first-order condition and just go through with this maximum likelihood estimation process. In return as a solution to the above, I will have beta hat FGLS, lambda hat FGLS and a sigma squared hat FGLS.

This is the solution, these are going to be efficient estimators, right? We have seen this earlier, you can go back to the theory and learn from there, but these are all efficient estimators. So, lambda hat FGLS is our parameter of focus, it tells us the extent and the form of the variance-covariance structure of spatial dependence representation in the error term of a spatial error model.

So, having understood the spatial error model, its form, and the motivation, why do we even study it and we have seen why it should be a popular model form right? And finally, coming to a point where we can figure out how to estimate this model; although of course, it is a highly computational exercise.

You will see upcoming very soon we will be starting hands-on tutorials. And, you will see that the statistical software, you will see that estimation of a spatial error model is scanned and so is the estimation of a spatial lag model.

But, it is good to know what is happening behind the scenes. So, the theory that we see here is going behind the estimation of the SAR model, right? How does the weights matrix enter, and how does the spatial dependence parameter lambda enter the entire story, right? So, these are very important steps in terms of understanding the results that you will see from our tutorial exercise.

What I want to discuss for a little bit now is what is the impact of ignoring spatial dependence in model error. If I ignore spatial dependence, what I am saying is I am assuming lambda equals 0 in previous discussions, right? So, I am saying as if my error term has a variance-covariance structure which is represented by an identity matrix, that is off-diagonal elements are 0, errors are spherical, and the diagonal elements are the same.

If I assume that and now my lambda is non-zero which is what Anselin is showing us through this simulation. So, now, I am looking at lambda as my parameter of interest and as lambda increases, what happens to my beta hat? You see that when lambda equals 0, my expectation of beta hat is somewhere around here and it remains there for all the cases of lambda. What is happening; however, is that for lambda equals 0.9, there is a very large error around the expectation of beta hat.

And, this error has gone up considerably from the case where lambda was equal to 0, right? So, what is happening is that it's so, lambda equals non-zero which is spatial dependence in error term bringing in inefficiency into the model estimators that is the beta hats, right? It is bringing the idea that the standard error of this beta hat will now become large, it is going to become an imprecise and inefficient estimator, right?

How do you go about reconciling it? The FGLS strategy that we have just seen before we saw this picture.

So, that is almost all about the spatial error model. I just want to talk about the variant, a variant of the spatial error model which is called the Spatial Moving Average model. So, this is called as also the SMA error model, right? So, I am going to briefly discuss it, the specification of this model is as follows. The specification is y equals x beta plus u, something I am very used to such that now u is equal to lambda W e plus e, where e is multivariate normal 0 comma sigma squared I_N.

Instead, of u being spatially dependent on itself in the neighborhood, now it is dependent on the lags of a standard normal process. So, it is a variant of how in the random structure these off-diagonal elements are correlated with each other. As far as interpretation is concerned, why is it different right?

A question comes to us, why is it different from what we were doing earlier? Well, in this setting, this particular model represents a very specific case when e represents innovation, and then W e is somehow the propagation of or diffusion of innovation, right?

So, diffusion of innovation, right? In social sciences or technology, people are very interested in why technology is adopted the way they are. Can we understand them formally, can we measure the diffusion of why people buy certain types of apps on the internet, or why they buy certain models of phones, computing devices, and so on; such a model can explain such things.

So, this diffusion of innovation can also innovation, can be termed the smoothing effect, the smoothing effect of innovation on spatial neighbors. So, it is a diffusion, right? So, we are looking at how the neighbors are going to get this effect or how is it going to spill over to the neighbors.

Here, to understand how we estimate these things, the critical entity was this variance-covariance matrix sigma. Once you know the sigma, you also know other things, right? So, sigma which is nothing, but the variance of u given x, in this case, is sigma squared I plus lambda W. Instead of I minus lambda W for SAR, for SMA it's I plus lambda W times I plus lambda W transpose.

So, I would just mark out, and notice this plus sign as how things are different. From here on everything will follow through exactly as same as the spatial error model, which we have just discussed in detail. So, that is all about specifying spatial dependence through the error model of a spatial regression framework. So, by now you have developed a good understanding, a good hold of how these spatial processes are specified formally, how we measure these effects, and how we estimate these from the data, right?

So, going forward we are coming very close to the end of this lecture series, and moving on to the hands-on exercises, but now, we have formally used this device called the weights matrix to specify spatial dependence in either the mean outcome or the error structure, something that we started this module with.

As a next step, we will look at some global measures of spatial autocorrelation using the weights matrix-based notion of spatial lags. And, then we will look at a couple of hypothesis tests which will be strictly time-permitting, but I will end this lecture here.

Thank you very much for your attention.