Lecture 11: Spatial Statistics II

Calculating a Spatial Covariance Function Using Stata

- Goal is to determine the spatial extent of dependence in the data
- The standard way to do this is to estimate the following spatial covariance function:

$$C(d) = \frac{1}{N_{ij}} \sum_{i,j,d_{ij}=d} (z_i - \bar{z})(z_j - \bar{z})$$

- Make sure your coordinate points are calculated in meters
- Demean the data
- Create a data set that is all pairwise combinations of all observations**
- Calculate distances and products of ε from the two data sets
- Keep relevant distance ranges to calculate covariance function
- Collapse on distance intervals

** If your data set is too big such that you can't pairwise join the full data set at once, do it in chunks

Fixing the Standard Deviation of an Estimated Mean

Recall that if using iid data, $\hat{V}(\bar{Z}) = \frac{\sum_i \varepsilon^2{}_i}{n}$ where $\varepsilon_i = Z_i - \bar{Z}$

if using dependent data, $\hat{V}(\bar{Z}) = \frac{1}{n^2} \sum_i \sum_j \varepsilon_i \varepsilon_j$

and $\hat{se}(\bar{Z}) = \sqrt{\hat{V}(\bar{Z})}$

Sometimes it's useful to parameterize the Covariance Function.

For example, maybe $\text{cov}(Z_i, Z_j) = \sigma_0{}^2 e^{-\alpha d_{ij}}$, $0 < \rho < 1$, d_{ij} = distance

We can estimate α with a linear regression

$$\ln[\text{cov}(Z_i, Z_j)] = \underbrace{\ln(\sigma_0{}^2)}_{\text{constant}} - \alpha d_{ij} + u_{ij}$$

Notice that we can also estimate $\sigma_0{}^2 = \frac{1}{N} \sum_i (Z_i - \bar{Z})^2$

Which is a better method because it doesn't depend on functional form

Example With Data on Fraction Riding Transit

```
. gen fride = trvlpbn/trvlpbd
. sum fride
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       fride |       633    .0651358    .1043463          0   .7605634
. gen eps = fride-r(mean)
. global sqrn = sqrt(r(N))
. gen sd = r(sd)/$sqrn
. tab sd

*** This is standard deviation of the estimated mean absent spatial autocorrelation

         sd |      Freq.     Percent        Cum.
------------+-----------------------------------
   .0041474 |        633      100.00      100.00
------------+-----------------------------------
      Total |        633      100.00

. gen coveps = epsfr*epsto

. collapse (sum) coveps

.
. gen adj_sd = sqrt(coveps)/($sqrn^2)

. list

*** This is the sd of estimated mean accounting for spatial autocorrelation
     +---------------------+
     |   coveps     adj_sd |
     |---------------------|
  1. | 424.9993   .0325679 |
     +---------------------+
```

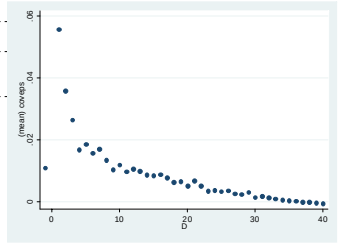
Analyzing the Covariance Function

- Determine the extent of spatial autocorrelation in the data
- Construct weights matrix for calculation of Moran's I or for use in regression

```
. replace dis = dis/1000
(92084 real changes made)
. gen ln_coveps = log(coveps)
(33128 missing values generated)
. reg ln_coveps dis

      Source |       SS       df       MS              Number of obs =   59589
-------------+------------------------------           F(  1, 59587) = 2564.60
       Model |  6632.17667     1  6632.17667            Prob > F      =  0.0000
    Residual |  154094.962 59587  2.58605001           R-squared     =  0.0413
-------------+------------------------------           Adj R-squared =  0.0412
       Total |  160727.138 59588  2.69730715           Root MSE      =  1.6081

------------------------------------------------------------------------------
   ln_coveps |      Coef.   Std. Err.      t
-------------+----------------------------------------------------------------
         dis |  -.0638939   .0012617   -50.64
       _cons |  -5.085842    .016373  -310.62
------------------------------------------------------------------------------
```



Regression Analysis With Spatial Data

Suppose that we have the model

$y_i = X_i\beta_i + \varepsilon_i$

and we want to estimate the β_i parameters

The only difference between this setup and a standard regression setup is that observations have locations

- All the standard caveats still apply:
  - X variables are "exogenous" or E(Xε)=0
  - There are no omitted variables correlated with any variables in X that predict y
  - y does not cause any variables in X
  - The sample used for the regression does not suffer from selection bias
- May want to weight by number of people represented by each observation

Estimating the Parameters

$\min_{\beta}[\sum_i (y_i - X_i\beta)^2] = \min_{\beta}[(y - X\beta)'(y - X\beta)]$

produces $\hat{\beta} = [X'X]^{-1}X'y$

which is the Ordinary Least Squares (OLS) estimator

$V(\hat{\beta}) = [X'X]^{-1}X'V(\varepsilon\varepsilon')X[X'X]^{-1} = [X'[V(\varepsilon\varepsilon')]^{-1}X]^{-1}$

If the observations i are iid,

$= [X'X]^{-1}X'\sigma_\varepsilon{}^2 IX[X'X]^{-1} = [X'X]^{-1}\sigma_\varepsilon{}^2$

Otherwise, we need a way to calculate $V(\varepsilon\varepsilon')$.

- As a rule of thumb, the more 0s you can put in the variance-covariance matrix of the errors V(εε'), the smaller your standard errors will be

Options for Fixing Standard Errors

- If you have data that can be grouped into categories across which it is reasonable to assume no spatial dependence in the error term, you can use "cluster" – but you need to have many clusters
  - "Cluster" assumes a general covariance structure within category
  - $V(\varepsilon\varepsilon') = \begin{bmatrix} \Sigma_1 & 0 & 0 \\ 0 & \Sigma_2 & 0 \\ 0 & 0 & \Sigma_3 \end{bmatrix}$
  - Generally overestimates standard error of parameters because you know some of those covariances in Σ_i should be 0 or constrained
  - Example: census tract data for multiple metropolitan areas
  - Need the number of clusters to be large
  - Stata Command: `reg y x1 x2 x3 x4, cluster(msa)`

Options for Fixing Standard Errors

• If this is not possible because, for example, you do not have many clusters, you can define fewer clusters that have weak dependence and many observations each and adjust the t-statistic critical value. (Conley, Hansen, 2009)

  • Define G clusters

  • Run the regression using this set of clusters

  • For testing whether a coefficient is not equal to 0 with 95% confidence, use the critical value from the student t-distribution with G-1 degrees of freedom instead of the reported p-value in the regression table

Student t-distribution Critical Values for 2-Tailed Tests - 95% Confidence

G-1	Value	G-1	Value
3	3.18	15	2.13
4	2.78	20	2.09
5	2.57	25	2.06
6	2.45	30	2.04
7	2.36	35	2.03
8	2.31	40	2.02
9	2.26	45	2.01
10	2.23	50	2.01

Options for Fixing Standard Errors

• If that doesn't work – because it gives you standard errors that are still too large – you can put more structure on the error term using a Feasible Generalized Least Squares type procedure.

• Run OLS, capture estimate of errors ε

• Estimate a covariance function on the errors and plug fitted values in for Σ in the formula

$$V(\hat{\beta}) = (X'X)^{-1}X'\Sigma X(X'X)^{-1}$$

$$\text{For example, might plug in } \hat{\Sigma}_{ij} = \hat{\varepsilon}_i\hat{\varepsilon}_j(1-\frac{d_{ij}}{\overline{D}}) \text{ for } d_{ij} < \overline{D}$$

• You can implement this with spatreg in Stata

  • Define an NXN weights matrix that is calculated as above w_{ii}=1 and w_{ij} declines with the distance between observations i and j

Spatial Lag Models

• Want to know the effect of neighbors' actions on your own actions

• Neighbors can be defined in different ways and can be a continuous measure

  • Similarity in demand for living in different states
  • Relative spatial location

$$y = \alpha Wy + X\beta + \varepsilon$$

where W is the weights matrix

• This is a complicated econometric equation to estimate, but essentially what you do is solve for y from the above equation

$$y = (I - aW)^{-1}X\beta + (I - \alpha W)^{-1}\varepsilon$$

where W is the weights matrix

• You still want to have "exogenous" variation in predictor variables to achieve consistent estimates of α

Implementation in Stata

• Set of add-on spatial econometric commands exist

  • package sg162 from http://www.stata.com/stb/stb60

```
Clear
set mem 200m
set matsize 1000

*** Calculate potential weights matrices
spatwmat, name(W1) xcoord(x_coord) ycoord(y_coord) band(0 3000) binary eigenval(E1)
spatwmat, name(W2) xcoord(x_coord) ycoord(y_coord) band(0 3000) standardize eigenval(E2)

*** Calculate Moran statistic
spatgsa hoval income crime, weights(W) moran twotail

*** Run spatial error model
spatreg y x1 x2, weights(W1) eigenval(E1) model(error) robust

*** Run spatial lag model
spatreg y x1 x2, weights(W2) eigenval(E2) model(lag) robust
```

Geographically Weighted Regression

- Run OLS but allowing coefficients to vary across space
- Implement this by doing weighted least squares for every point on the map, with the weights declining with distance to the point
- Produces a surface of parameters, predicted values and errors
- Geographers sometimes find this method useful for model selection
    - Allows you to check whether parameter estimates change a lot over space. If they are unstable, that is a good indication that the model being used is not well specified
    - Is a good way of creating a smoothed version of an outcome variable, by taking predicted values from this regression – akin to Kriging