Lab 8: Spatial Regression and Poverty

Introduction

The goal of these exercises is to give you a chance to put the concepts we have been discussing in class into practice. Keep in mind we have only a limited amount of time (one week this time!), so our focus for this lab will be *breadth* rather than *depth*!

To get started, let's load some data. We're going to use the Southern Counties Data from Voss (2006), which we've been working with in class. There are two ways to get this data, the 'traditional' way (loading a shapefile), and the 'easy' way, loading a 'pre-packaged' dataset I have prepared for you:

```
library(rgdal)
soco = readOGR("Data/south00.shp", layer="south00")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "Data/south00.shp", layer: "south00"
## with 1387 features
## It has 17 fields
```

```
summary(soco@data)
```

If the above doesn't work, you can always get the data from here:

```
load("Data/soco.rdata")
summary(soco@data)  # You might want to check this out
```

The data contains the following variables/information (this info is also available in the `vardescription_south00.csv` file). This dataset was originally used to examine intercounty variation in child poverty rates in the US. In particular, the authors used it as a way to demonstate the utility of spatial regression analysis.

Variable	Description
CNTY_ST	County and state name
STUSAB	State abbreviation
FIPS	FIPS code
YCOORD	Y coordinate (meters)
XCOORD	X coordinate (meters)
SQYCORD	Y coordinate squared (for trend surface)
SQXCORD	X coordinate squared (for trend surface)
XYCOORD	X coordinate * Y coordinate (for trend surface)
PPOV	Proportion of children in poverty
PHSP	Proportion Hispanic
PFHH	Proportion female-headed households
PWKCO	Proportion work outside of county of residence
PHSLS	Proportion less than high school educated
PUNEM	Proportion unemployed
PUDEM	Proportion males underemployed, some work in 1999
PEXTR	Proportion employed in extractive industry
PPSRV	Proportion employed in professional services
PMSRV	Proportion employed in miscellaneous services

Variable	Description
PNDMFG	Proportion employed in non-durable manufacturing
PNHSPW	Proportion non-Hispanic white
PMNRTY	Proportion minority (total - non-Hispanic white)
LO_POV	Natural log proportion children in poverty
PFRN	Proportion foreign-born
PNAT	Proportion native-born
PBLK	Proportion African American/black, alone and including Hispanic
P65UP	Proportion 65 and older
PDSABL	Proportion disabled
METRO	Metro county
PERPOV	Persistent poverty, 1970-2000 (ERS)
OTMIG	Rate of out-migration
BINMIG	Rate of black in-migration from non-south
INCARC	Proportion males 18-64 in correctional institutions
BINCARC	Proportion black males 18-64 in correctional institutions
SQRTPPOV	Square root proportion children in poverty
SQRTUNEM	Square root proportion unemployed
SQRTPFHH	Square root proportion female-headed households
LOGHSPLS	Natural log proportion less than high school educated
PHSPLUS	Natural log proportion high school educated or more

Part I: Exploratory Spatial Data Analysis

The response variable we are interested in is `PPOV`, which describes the proportion of children living in poverty in each county. So first things first, create a choropleth map that shows the distribution of `PPOV` across the sothern states:

```
library(maptools)
library(RColorBrewer)
library(classInt)
# Consider using quantiles class breaks, or a 'ramp-style' palette
# Show R code to produce a choropleth map of PPOV
```

`R` doesn't have a ton of fancy tools for mapping, but it does a pretty good job of quickly giving us the information we will need to do our due diligence on the relationships in our data. If this were a real project then we would need to undertake the above steps for all of our independent variables, but for now we will just proceed with `PPOV` (our response; done already) and `PFHH`. In addition to the plots, note if there is any observable spatial co-variation in these variables:

```
# Show R code to produce choropleth map of PFHH variable
# Is there any notable co-variation in the above two variables
```

It is also always important to examine the underlying attribute distribution of your variables (i.e., via histograms, density plots, etc). Create histograms and any other relevant plots to give us an idea of the underlying distribution of the `PPOV` and `PFHH` variables from our dataset. What do you notice about these variables?

```
# Show R code to produce histograms of the `PPOV` and `PFHH` variables
```

Now that we have an idea of the spatial and aspatial distribution of these variables, it is time to look at their relationships with each other. Create a scatterplot of `PPOV` with `PFHH` and note any potentially important relationships you observe:

```
# Show R code to produce a scatterplot of `PPOV` with `PFHH`
```

Take the opportunity here to study the relationship between these two key variables. Now add a 'best fit' line to the plot (think `abline` function or `geom_abline` if you're using `ggplot`). What is the slope of this line?

```
# Show R code to produce a scatterplot of `PPOV` with `PFHH` with a best fit line
# What is the slope of the line? You'll need to fit a model here...
```

Obviously the two variables are positively and fairly strongly correlated. Does it look like there are any outliers here? How might you determine this (no need to show `R` code here, just make suggestions)?

```
# How might you determine if there are outliers in the previous scatterplot?
```

Bonus: try to take a look at the state-specific relationships between `PPOV` and `PFHH`. Does this seem to capture any spatial effects?

```
# Bonus: Show R code to produce conditional scatterplot of `PPOV` with `PFHH` for each state.
# You'll probably need ggplot for this (facet_wrap)
```

OLS and Residuals

At this point we should have a good feel for some of the descriptive characteristics in our data. There is obviously much more that could be done, but you should have the basic idea at this point. Since some of the things we should have done (Global and Local Spatial Autocorrelation) will be done later in the lab anyway, so we can safely skip over them here. For now though, `R` is just one more software program that will allow you to run a basic OLS Regression.

Spatial Weights

Before we get started we need to define a spatial weights matrix. We won't use it directly here (as we will when we run spatial regression), but `R`'s `spdep` allows us to compute Moran's I for our residuals as well as some other spatial diagnostics on the OLS so we do our matrix now rather than later.

```
library(spdep)
w_nb = poly2nb(soco, row.names=soco$FIPS, queen=TRUE)  # Use queen contiguity
```

Summarize the above neighbors list, and create a simple plot to show overall connectivity. Does this look about right? Which location is the most connected? Bonus: Can you tell me which county this is?

```
# Show R code to summarize and plot the above neighbors list
# Bonus: Which county is the most connected. Tell me the FIPS _and_ name...
```

This neighbors list is just one 'part' of weights matrix creation. Convert the above neighbors list into a proper weights object, using row standardised weights (`see ?nb2listw`):

```
# Show R code to convert w_nb to a weights matrix (call it w_mat)
```

OLS Regression

Now we'll fit an OLS Regression model (call it `mod1`) using the Southern Counties dataset. Firstly, the *dependent variable* will be the square root of the percentage of children living in poverty (`SQRTPPOV`), and your *independent variables* will include `PHSP`, `PFHH`, `PUNEM`, `PEXTR`, `P65UP`, `METRO`, `PHSPLUS`. Fow now, we'll ignore any possible interaction terms etc...

What is the R^2 for this model? Are the coefficients all significant?

```
# Show R code to fit an OLS Regression model with the above variables
# What is the R^2 value? Are the coefficients significant?
```

This leaves us to interpret the various output statistics. Some things to pay attention to: * Log likelihood: higher, better, (less negative) * Aikake info criterion (AIC): lower, better * Others?

These are all aspatial diagnostics and mostly they will give us information in a comparative sense. We should also look for multicollinearity; what is a good test for this (remember, multicolinearity inflates the standard errors (variance) of the coefficients)? Perform this test; is multicolinearity going to be a problem here?

```
library(car)
# Show R code to run a test for the effects (variance inflation) of multicolinearity
```

What about heteroscedasticity (non-constant variance of the residuals)? There is a nice test for this too (`ncvTest` from the `car` package), is it significant here? Note: this test is sometimes also called the Breusch-Pagan test.

```
# Show R code to run ncvTest and determine if it is significant
```

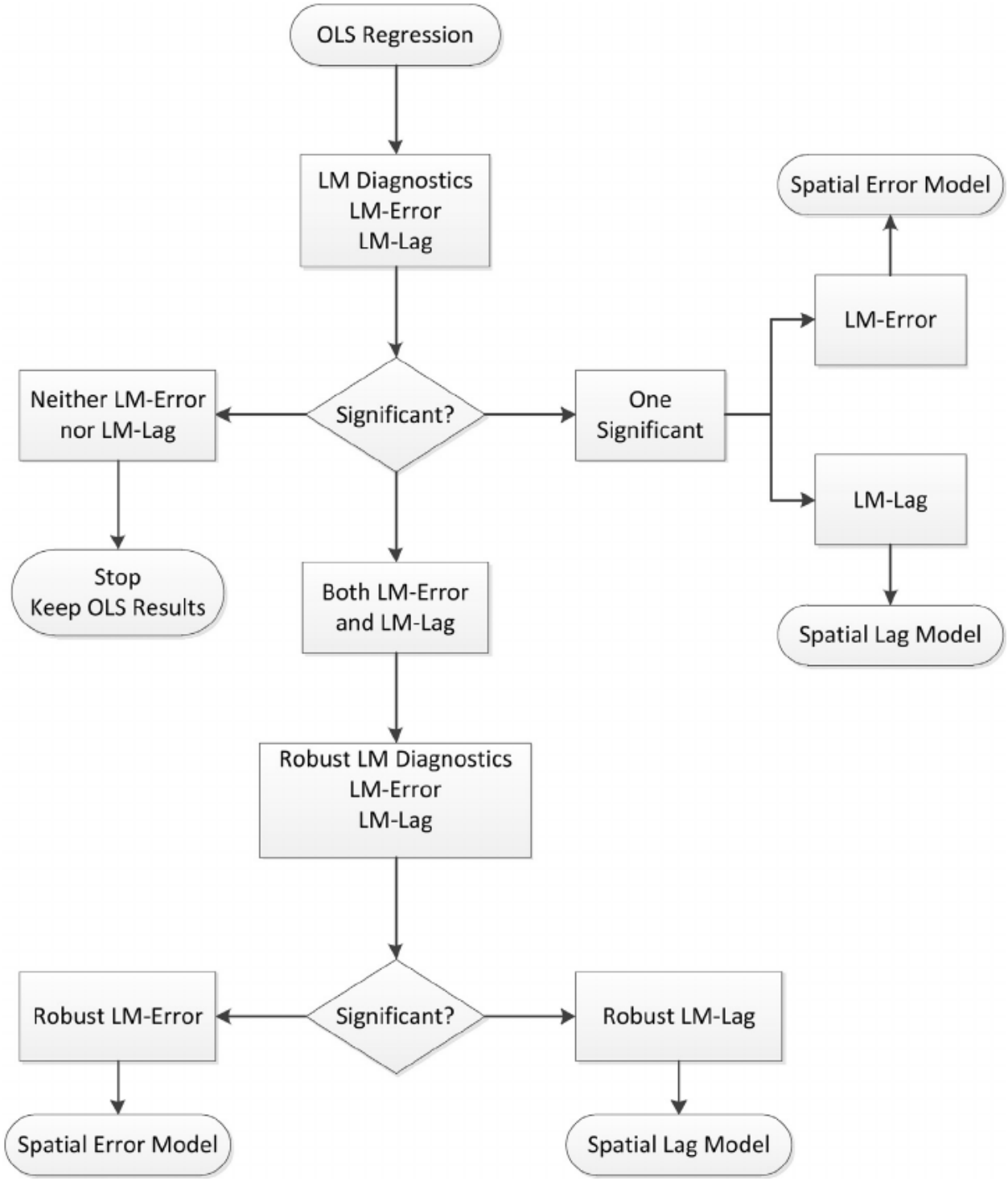
The potential problems we have with heteroskedasticity could be ameliorated by transforming some of our variables, and/or reducing the impact of certain outliers. Whether or not your discipline thinks this is appropriate statistical technique is up to you. For now we will leave this be, but recall from the lecture that heteroskedasticity is a violation of one of the key assumptions of OLS and its presence should throw into question the validity of your model.

What about the residuals, are they normally distributed? What are some good ways to test this: visually and statistically (the `moments` package has some useful tests for this)?

```
# R code to test for normally distributed errors (plot and/or test statistic)
```

Diagnostics

Now its time to decide if we need to pursue spatial regression models to counteract any of the issues we've observed so far. We can use Anselin's decision tree framework to help us here:



Firstly, we'll explore Moran's I calculated on the OLS residuals. In this example, what does this show?

```
# Show R code to run a Global Moran's I test for residual spatial autocorrelation
```

We can also perform the Lagrange Multiplier tests. These help us to calculate the 'effectiveness' of the two forms of spatial regression model, along with their *robust* forms. The way we read this is we look to see if the lag LM is significant. Then we look at the error LM. If only one is significant then the metrics point to that

type of model. If both are significant then we forget what we just read and pick the higher of the two robust scores. Run the `lm.LMtests` to help us decide which spatial regression model to use going forward:

```
# Show R code to run lm.LMtests
```

Residual Maps

The Mississippi Delta, Appalachia, and to a lesser extent the U.S. Mexican border are all home to clusters of high residuals. On the plus side, we don't have any extreme outliers, so our model isn't performing too badly! Produce a quick map to show this spatial distribution:

```
# Show R code to produce a basic map of residuals
```

A good way to *quantitatively* show clustering in the above residuals might be to plot the local spatial autocorrelation statistics (Gi* or Moran's Ii). For **bonus** points, compute the local spatial autocorrelation of the above residuals and map them:

```
# Bonus: compute local moran's Ii (or local getis and ord stat)
# Map results to get an idea of local clustering in residuals
```

Part II: Spatial Regression

In this segment we will run spatial lag and spatial error models and compare the results we will also work at interpreting the models.

We begin, as before with specifying our regression model design: `SQRTPPOV ~ PHSP + PFHH + PUNEM + PEXTR + P65UP + METRO + PHSPLUS`. We'll use the same weights matrix as before (`w_mat`), but this time we'll fit spatial lag and error models:

```
# Show R code to fit a spatial lag (lag_mod) and a spatial error (err_mod) model
```

Now comes the fun part, interpreting our results. We can print the `summary()` results of all three model runs and explore the output. We have already started with the first model (OLS) and explored some spatial dependence diagnostics. Indeed, we have already seen that the LM's and robust LM's indicate we should prefer a spatial *lag* model over the spatial *error* model.

Now let's compare the summary model diagnostics. The R^2 value is a bit 'iffy' with spatial models, so the log-likelihood and AIC are preferred. Which model appears to perform best in terms of model fit?

```
# Show some R code to highlight which model fits best in terms of R^2, AIC, and log-likelihood
```

Turn next to the spatial autoregressive coefficients (ρ, spatial lag, or λ, spatial error). What is the value of Rho? Is it significant? What is its sign (positive or negative spatial autocorrelation)?

```
# Show R code to get Rho (or just refer to earlier printout and state value)
# Is it significant? What is its sign? What does this mean?
```

Show the same for the error model. Which one seems to indicate stronger spatial autocorrelation? Is there a difference in the associated standar errors for these coefficients?

```
# Show R code to get Lambda (or just refer to earlier printout and state value)
# Is it significant? How does it compare to above value for Rho?
```

What about the LR and Wald tests? What do these suggest? How do they compare to ANOVA between the spatial regression models and the OLS model?

```
# Show R code to compare models' LR and Wald tests (or refer to earlier printout)
# How is this similar/dis-misimilar to an ANOVA test?
```

Look at other explanatory variables (signs and magnitudes). It looks like the `METRO` variable lost significance in the error model. What might this suggest? To get a better idea of the *impacts* that these models are capturing, take at look at the direct, indirect, and total impacts of the lag model:

```
# R code to extract 'impacts' of the lag model (caution, very slow!)
```

How do the above impacts relate to the values from the OLS and error model?

```
# Describe the key differences between the three models in
# terms of coefficients
```

Our model *still* has major problems with heteroskedasticity. We would have to deal with this going forward. Consider a scatterplot that compares predicted values against residuals:

```
# Show R code to produce a scatterplot to help us
# compare predicted with residuals values from our
# chosen model
```

What about a Breusch-Pagan test to look at heteroscedasticity? In addition to the measure we used earlier, we can use the `bptest.sarlm` function from the `spdep` package to look at heteroscedasticity in spatial models *specifically* (perform for both lag and error model and discuss significance):

```
# R code to perform BP test on spatial regression models.
# Are they significant?
```

For bonus points, take a look at the remaining residual spatial autocorrelation. Is it significant?

```
# Bonus: Compute global and local spatial autocorrelation in
# the error model residuals.
# Is the SA still significant?
# Consider plotting the local spatial autocorrelation also...
```

Stepping back a little, let's try and understand what these models have told us:

- First, there are spatial processes at play in our data that we need to be thinking about—ignoring spatial autocorrelation of over 0.3 is not good practice!
- Second, our diagnostics and output consistently favor a Spatial *Error* form as a means of capturing this spatial autocorrelation.

This suggests that our processes are varying consistently across small areas, but that we are not likely seeing an active process of counties interacting with one another—so we don't need to talk about the movement of individuals across county lines as the source of this relationship—but we *are* more likely to have some combination of large scale regional processes and regionally varying missing variables.

Reference

This lab has been adapted from a lab developed by Chris Fowler from the University of Washington.