



## Version 0.6

### YouTube Playlist:

[https://www.youtube.com/playlist?list=PLlnEW8MeJ4z6Du\\_cbY6o08KsU6hNDkt4k](https://www.youtube.com/playlist?list=PLlnEW8MeJ4z6Du_cbY6o08KsU6hNDkt4k)

### Work in Progress:

Monday, January 29, 2018 Version

For a more comprehensive view of R's spatial and mapping capabilities, check our Roger Bivand's list: <https://cran.r-project.org/web/views/Spatial.html>

Most of these commands rely on the *spdep* library, and you will need to install and load it `library(spdep)` beforehand. Other libraries will be specifically mentioned when called for. e.g., *rgeos*, *rgdal*

### Brief R overview:

Free! Not fast for huge data sets, but few have one ☺  
# is R's comment character, ignores words after #  
R is case sensitive, so *myvar* is not same as *Myvar*. You can name things *my.var* or *my.var1* or *my.var.1*  
*myvar=3* or *myvar<-3* assign value 3 to *myvar*

For creating a SHP spatial data set using QGIS, see <https://www.youtube.com/watch?v=FHBDAGc9-8E>

Brief commentary on my method: I advise adding your data to the SHP file as in my videos, then reading it in and creating spatial weights (*W*) matrices in R (or creating weights in GeoDa from the same SHP file, and reading those in). While you can read in your data separately, be careful- it is possible that the ordering of the data and the weights will not match, and then Garbage in, Garbage out.

### Reading in Shapefile with (or without) Data:

```
library(rgdal)
spat.data=readOGR(dsn=".",
layer="shpfilename")
•dsn is directory (. is current)
•layer is shp file name without .SHP
names(spat.data) show variable names
spplot(spat.data,"var") make map
```

### Spatial Weights

•Reading in external weights files:  
`nb.wts=read.gal("gal.file.gal")`  
`nb.wts=read.gwt2nb("gwt.file.gwt")`  
•Convert nb to listw (needed for regressions, etc.)  
`listw.wts=nb2listw(nb.wts)`  
•Creating Spatial Weights in R  
`queen.nb=poly2nb(spat.data)`  
`rook.nb=poly2nb(spat.data,queen=FALSE)`  
*Nearest neighbors requires a matrix of coords, longitude in left column. If you don't have coordinates (centroids):*  
`library(rgeos)`  
`cent=gCentroid(spat.data,byid=TRUE)`  
`dnearneigh(coords matrix, d lower bound, d upper bound or all.linked)`  
Distance-based nearest neighbors  
`knearneigh(coords matrix, k=4):4`  
nearest neighbors

### Tools for Working with Weights

`is.symmetric.nb` is structure symmetric?  
`diffnb(nb1,nb2)` differences in neighbor lists  
`knn2nb` convert knn to nb format  
`mat2listw, nb2mat`: convert to/from matrix  
Others: `droplinks`, `summary.nb`, `plot.nb`  
`eigenw(listw)`: give eigenvalues of listw

More details: <https://cran.r-project.org/web/packages/spdep/vignettes/nb.pdf>

### Estimating regressions

Save model to a name: `reg.eq1=y~x1+x2`

OLS:  $y = X\beta + \varepsilon$

```
reg1=lm(reg.eq1,data=spat.data)
```

SLX Spatially Lagged X  $y = X\beta + Wx\theta + \varepsilon$

```
reg2=lmSLX(reg.eq1,data=spat.data,
listw)
```

SAR Spatial Lag Model  $y = \rho Wy + X\beta + \varepsilon$

```
reg3=lagsarlm(reg.eq1,data=
spat.data, listw)
```

### SEM Spatial Error Model

$y = X\beta + u, u = \lambda Wu + \varepsilon$

```
reg4=errorsarlm(reg.eq1,data=spat.d
ata, listw)
```

### SDEM Spatial Durbin Error Model (add lag X to SEM)

$y = X\beta + Wx\theta + u, u = \lambda Wu + \varepsilon$

```
reg5=errorsarlm(reg.eq1,
data=spat.data, listw,
etype="emixed")
```

### SDM Spatial Durbin Model (add lag X to SAR)

$y = \rho Wy + X\beta + WX\theta + \varepsilon$

```
reg6=lagsarlm(reg.eq1,
data=spat.data,listw, type="mixed")
```

### Test Model Restrictions:

`LR.sarlm(reg6, reg4)` likelihood ratio test to see if SDM should be restricted to the SEM. You can test any model to see if it should be restricted to a simpler, **nested** model. SDM and SDEM are not nested, as one cannot be *simplified* into the other.

### Marginal Effects of Models:

When using any model with a  $\rho Wy$ , you cannot interpret the coefficients as marginal effects from `summary(reg1)` command. You must calculate the direct, indirect, and total marginal effects using the `impacts` command. (Also useful for total impacts for models with SLX)

```
impacts(reg1,listw=listw.wts)
summary(impacts(reg1,listw=listw.wt
s,R=500),zstats=TRUE)#for p values
```



## Version 0.6

### Handout Back Side

#### Controversial? Models: But here they are!

**Manski** All-inclusive Model:  $y = \rho W y + X\beta + WX\theta + u$ ,  $u = \lambda W u + \varepsilon$  (not recommended)  
`reg6=sacsarlm(reg.eq1, data=spat.data, listw, {listw2=,} type="sacmixed")`  
*listw2 allows for a different weights matrix for the error structure if desired*

**SARAR** a.k.a. *Kelejian-Prucha, Cliff-Ord, or SAC*  
If all  $\theta = 0$ ,  $y = \rho W y + X\beta + u$ ,  $u = \lambda W u + \varepsilon$   
`reg7=sacsarlm(reg.eq1, data=spat.data, listw, {listw2=,} type="sac")`

**SARMA** (like SARAR, but more local error structure)  
 $y = \rho W y + X\beta + u$ ,  $u = I - \lambda W \varepsilon$  or  
 $y = \rho W y + X\beta + u$ ,  $u = \lambda W \varepsilon + \varepsilon$   
`reg8=spautolm(reg.eq1, data=spat.data, listw, family="SMA")`  
*Note on SARAR and SARMA: Using the same W matrix for lag and error terms can lead to identification problems. Probably best to avoid these models. HOWEVER, sometimes people mention a SARMA(p,q), allowing for p different  $\rho W y$  W matrices, and q different  $\theta W \varepsilon$  terms likewise (normally W's of different orders)*

#### A note on Model Selection Methodology:

Rule #1 is to always think about the spatial structure your data ought to take logically. If you don't know, or if several structures are possible, **then** think about statistical specification searches.

#### Lagrange Multiplier Tests on OLS to see if a SEM or SAR model might be needed:

`lm.LMtests(reg1, listw, test="all")`  
**"Anselin Method"**: If only one of LMerr or LMlag are significant, try that regression. If both are significant, then look at the Robust (to false positives for the other kind of spatial relationship) versions (RLMerr and RLMLag). If only one of those are significant, run that model. Ignore the SARMA result. My advice: If both RLMerr and RLMLag are significant, try the

#### LeSage and Pace Method (more or less, this is just a brief overview! 😊):

- 1) Run Spatial Durbin Model
- 2) Do LR tests to see if Spatial Durbin should be restricted to either the spatial Error or Spatial Lag Models
- 3) If the Spatial Error seems promising, consider the Spatial Durbin Error Model (adds Lagged X's to Spatial Error)

#### Other Spatial Stats Commands

`moran`, `moran.test`, `moran.plot`  
`lm.morantest` (designed for residuals)  
`bptest.sarlm(reg1, studentize=TRUE)`  
`moran.plot(variable, listw, labels=as.character(???), pch=19)`  
`Hausman.test(reg4)` #tests to see if OLS and SEM coefficients differ a lot; if so, Pace & Lesage (2008) suggest that neither SEM nor OLS are appropriate. Run on SEM model only.

`lagY=lag.listw(nb2listw(nb), Y)` #create lag Y object, avg of nb Y

**A note on terminology:** As you can see, many models go by many different names. Additionally, some argue that "SAR" should not refer to the lag model, but ean "Simultaneous AutoRegressive", as opposed to CAR for "Conditional AutoRegressive". In Spatial Econometrics we generally use SAR models, where spatial dependence is more global, but CAR models are more local in nature (used more in epidemiology).