Spatial Regression Models, Version 2

Chapter 4

Michael D. Ward and Kristian Skrede Gleditsch

2018-01-21

Introduction

These are lecture/lab notes for Chapter 5 in Spatial Regression Models. You will find all necessary information, including $\mathcal{R}$-code, to replicate the figures and maps found in this chapter. A few of the maps were obtained from public repositories (Wiki and others), but what you find here will allow replication of the bulk of chapter 5. This is intended to provide code as well as a didactic repository.

Code appears below as it is in the unadorned code files. Comments are shown with one poundsign; messages from the execution of this code is flagged with two poundsigns.

There are two goals of this material:

1. Provide replication code for the materials in Spatial Regression Models, version 2. This is hosted on httpps://srmbook.com
2. To elaborate a bit on what the code does to provide instruction on develop code in $\mathcal{R}$ that replications the material in the published volume.

This was developed on the following platform. Earlier (and later) platforms should work (as should windows and linux flavored platforms). However, some libraries may not be available for all platforms. Such is life.

If you have any feature requests or find bugs herein, please do not hesitate to pass them along to the authors (michael.don.ward at gmail.com and ksg at essex.ac.uk).

Chapter 5 deals with the so-called spatial error model. This model is appropriate when the values the error term are spatially correlated. These results may differ slightly from what is printed in the book.

Table 5.1, OLS, Spatial Lag, and Spatial Error Estimates

```
# Chapter 5 Code for Ward & Gleditsch, Spatial Regression Models, Sage, 2018.

# R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
# Copyright (C) 2017 The R Foundation for Statistical Computing
# Platform: x86_64-apple-darwin15.6.0 (64-bit)
rm(list = ls())
library(cshapes)
library(wbstats)
p5<-read.csv("rawdata/p5.csv")
wmap <- cshp(date = as.Date("2015-01-01"),useGW=TRUE)  # get map from CSHAPES
wmap$ISO1AL3 <- as.character(wmap$ISO1AL3)
wmap$ISO1AL3[wmap$CNTRY_NAME == "Kosovo"] <- c("RKS")
wmap$ISO1AL3 <- as.factor(wmap$ISO1AL3)
wmap@data$id <- as.character(rownames(wmap@data))
wmap.df <- wmap@data
wmap@data$democ <- p5$democ[match(wmap@data$ISO1AL3, p5$scode)]
wmap@data$autoc <- p5$autoc[match(wmap@data$ISO1AL3, p5$scode)]
wmap@data$polity <- wmap@data$democ - wmap@data$autoc

# now get GDP per capita data
```

```
library(wbstats)
gdp <- wb(indicator = c("NY.GDP.PCAP.CD"),
          startdate = 2014, enddate = 2014,country="countries_only")
gdp$value <- log(gdp$value)
# join the GDP data to the map data.frame
wmap@data$gdp <- gdp$value[match(wmap.df$ISO1AL2, gdp$iso2c)]

# make wmap@data a small data matrix
a.df<-subset.data.frame(wmap@data,select=c("polity","gdp"))
dim(a.df)
```

[1] 195 2

```
# now impute missing data in gdp and polity with sbgcop
library(sbgcop)
new.df<-sbgcop.mcmc(a.df,seed=123456,nsamp=10000,verb = FALSE)
non.miss.df<-data.frame(new.df$Y.pmean)
row.names(non.miss.df)<-wmap@data$GWCODE
non.miss.df$country<-wmap@data$CNTRY_NAME

# now get distance matrix
wmap.dist <- distmatrix(date = as.Date("2015-01-01"))

# recode distance data to be binary, for distance < 400km
w.dist<-wmap.dist
w.dist[w.dist<=400]<-1
w.dist[w.dist>400]<-0
# row standardize
w.dist.rstd<-w.dist/rowSums(w.dist)
# 0 on diagonal
diag(w.dist.rstd)<-0

# create spatial lag of polity. round result
non.miss.df$sp.lag.polity<- round((w.dist.rstd)%*%(non.miss.df$polity))
# order map dataframe on GWCODE
nm.df<-non.miss.df[order(wmap@data$GWCODE),]

# get rid of isolates
omits<-(which(rowSums(w.dist.rstd)==0))

# get rid of isolates
nm.df<-nm.df[-omits,]
w.d.rstd<-w.dist.rstd[-omits,-omits]
dim(w.d.rstd)
```

[1] 180 180

```
dim(nm.df)
```

[1] 180 4

```
# spatial model
library(spdep)
listw.obj<-mat2listw(w.d.rstd)
# spatial lag
sar.fit<-lagsarlm(polity ~ gdp, data=nm.df, listw.obj,type="lag",
                  method="eigen")
# spatial error
sem.fit <- errorsarlm(polity ~ gdp, data=nm.df,
```

```
                      listw.obj, method="eigen", quiet=TRUE)
# OLS
ols.fit <- glm(polity ~ gdp, data=nm.df)

library(stargazer)
stargazer(ols.fit,sar.fit,sem.fit,type = "latex",star.char = NULL,nobs=TRUE,digits=2,
          star.cutoffs = NA,notes.append = FALSE,
          notes= "",dep.var.caption  = "OLS, Spatial Lag, and Spatial Error Estimates"
          ,header=FALSE
)
```

Table 1:

	OLS, Spatial Lag, and Spatial Error Estimates		
	polity		
	normal	*spatial autoregressive*	*spatial error*
	(1)	(2)	(3)
gdp	1.29	0.77	1.18
	(0.31)	(0.28)	(0.39)
Constant	−7.36	−4.83	−5.74
	(2.73)	(2.36)	(3.49)
Observations	180	180	180
Log Likelihood	−583.29	−559.52	−557.57
σ^2		27.59	26.72
Akaike Inf. Crit.	1,170.57	1,127.04	1,123.14
Wald Test (df = 1)		54.11	63.46
LR Test (df = 1)		45.53	49.44

Note:

```
#SLIGHTLY DIFFERENT THAN 5.1 in Book owing to using imputed data.
```

Table 5.2, Exports, Europe: T_{i→j}

```
# Chapter 4 Code for Ward & Gleditsch, Spatial Regression Models
# Sage, 2018.

# R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
# Copyright (C) 2017 The R Foundation for Statistical Computing
# Platform: x86_64-apple-darwin15.6.0 (64-bit)

rm(list = ls())
# Matrix with reverse dyads
cmat98BA <- read.table("rawdata/tradedata/dyadmat_europe98_BA-no origins 3+.csv",sep=",",header=T)

# Matrix without  with reverse dyads
cmat98noBA <- read.table("rawdata/tradedata/dyadmat_europe98_no BA-no origins 3+.csv",sep=",",header=T)

# Data
```

```
logdat98 <- read.table("rawdata/tradedata//dyaddata_europe98_log-no origins 3+.csv",sep=",",header=T)

# Make distance matrix a list
dlist <- vector(mode="list",length=dim(logdat98)[1])
attr(dlist,"region.id") <- logdat98$id
attr(dlist,"class") <- "nb"
ids <- unique(cmat98BA$c1)
for(i in 1:length(ids)){
  # Find their numbers in the data
  dat1 <- cmat98BA[cmat98BA$c1==ids[i],]
  dat2 <- cmat98noBA[cmat98noBA$c1==ids[i],]
  input <- sort(c(dat1$c2,dat2$c2))
  dlist[[i]] <- input
}

# OLS
tab2.ols <- glm(logtrade ~ logdem + logapop + logbpop +
                  logargdppc + logbrgdppc + logs + logdist + logmid,
                  data=logdat98,na.action=na.omit)
# logLik(tab2.ols)

# SEM
tab2.sem <- errorsarlm(logtrade ~ logdem + logapop + logbpop +
                      logargdppc + logbrgdppc + logs + logdist + logmid,
                      data=logdat98,na.action=na.omit,
                      nb2listw(dlist,style="W"), method="eigen", quiet=TRUE)
# summary(tab2.sem)
# logLik(tab2.sem)

stargazer(tab2.ols, tab2.sem,type = "latex",star.char = NULL,nobs=TRUE,digits=2,
          star.cutoffs = NA,notes.append = FALSE,
          notes = " ",dep.var.caption  = "Exports, Europe:",header=FALSE
)
```

Table 5.3, Exports, Africa: T_{i→j}

```
# Chapter 4 Code for Ward & Gleditsch, Spatial Regression Models
# Sage, 2018.

# R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
# Copyright (C) 2017 The R Foundation for Statistical Computing
# Platform: x86_64-apple-darwin15.6.0 (64-bit)

rm(list = ls())
#Afica trade example

# Read data
cmat98BA <- read.table("rawdata/tradedata/dyadmat_africa98_BA.csv",sep=",",header=T)
cmat98noBA <- read.table("rawdata/tradedata/dyadmat_africa98_no BA.csv",sep=",",header=T)
logdat98 <- read.table("rawdata/tradedata/dyaddata_africa98_log.csv",sep=",",header=T)

# Make weigth matrix lists
dlist <- vector(mode="list",length=dim(logdat98)[1])
attr(dlist,"region.id") <- logdat98$id
```

Table 2:

	Exports, Europe:	
	logtrade	
	normal	*spatial error*
	(1)	(2)
logdem	0.38	0.43
	(0.06)	(0.10)
logapop	0.86	0.89
	(0.02)	(0.03)
logbpop	0.75	0.77
	(0.02)	(0.03)
logargdppc	1.54	1.56
	(0.04)	(0.06)
logbrgdppc	1.01	1.03
	(0.04)	(0.06)
logs	0.33	0.35
	(0.05)	(0.05)
logdist	−0.34	−0.34
	(0.01)	(0.01)
logmid	−1.94	−1.48
	(0.27)	(0.29)
Constant	−32.70	−33.94
	(0.67)	(1.71)
Observations	1,500	1,500
Log Likelihood	−2,325.80	−2,239.67
σ^2		1.14
Akaike Inf. Crit.	4,669.60	4,501.34
Wald Test		5,435.86 (df = 1)
LR Test		170.26 (df = 1)

Note:

```
attr(dlist,"class") <- "nb"
ids <- unique(cmat98BA$c1)
for(i in 1:length(ids)){
  # Find their numbers in the data
  dat1 <- cmat98BA[cmat98BA$c1==ids[i],]
  dat2 <- cmat98noBA[cmat98noBA$c1==ids[i],]
  input <- sort(c(dat1$c2,dat2$c2))
  dlist[[i]] <- input
}


tab3.ols <- glm(logtrade ~ logdem + logapop + logbpop + logargdppc +
                logbrgdppc + logs + logdist + logmid,
                data=logdat98,na.action=na.omit,trace=FALSE)
# logLik(tab3.ols)

tab3.sem <- errorsarlm(logtrade ~ logdem + logapop + logbpop + logargdppc +
                  logbrgdppc + logs + logdist + logmid,
                  data=logdat98,na.action=na.omit,
                  nb2listw(dlist,style="W"), method="eigen", quiet=TRUE)
# logLik(tab3.sem)

stargazer(tab3.ols, tab3.sem,type = "latex",star.char = NULL,nobs=TRUE,digits=2,
          star.cutoffs = NA,notes.append = FALSE,
          notes = "",dep.var.caption  = "Exports, Africa:",header=FALSE
)
```

Table 3:

	Exports, Africa:	
	logtrade	
	normal	*spatial error*
	(1)	(2)
logdem	−0.04	−0.01
	(0.04)	(0.05)
logapop	0.26	0.26
	(0.01)	(0.02)
logbpop	0.23	0.23
	(0.01)	(0.02)
logargdppc	0.38	0.38
	(0.02)	(0.03)
logbrgdppc	0.31	0.31
	(0.02)	(0.03)
logs	3.41	3.43
	(0.40)	(0.47)
logdist	−0.17	−0.17
	(0.01)	(0.01)
logmid	−0.71	−0.42
	(0.18)	(0.18)
Constant	−7.41	−7.47
	(0.33)	(1.45)
Observations	2,550	2,550
Log Likelihood	−3,097.23	−2,945.94
σ^2		0.58
Akaike Inf. Crit.	6,212.47	5,913.88
Wald Test		15,425.39 (df = 1)
LR Test		300.58 (df = 1)

Note: