Lab Exercise #8

This week, we will explore different spatial regression methods, including spatial lag, error regression, as well as geographically weighted regression.

The data for this exercise are Boston housing data based on http://lib.stat.cmu.edu/datasets/boston_corrected.txt. Some of the variables are described in the table below.

TOWN
: a factor with levels given by town names

TOWNNO
: a numeric vector corresponding to TOWN

TRACT
: a numeric vector of tract ID numbers

LON
: a numeric vector of tract point longitudes in decimal degrees

LAT
: a numeric vector of tract point latitudes in decimal degrees

MEDV
: a numeric vector of median values of owner-occupied housing in USD 1000

CMEDV
: a numeric vector of corrected median values of owner-occupied housing in USD 1000

CRIM
: a numeric vector of per capita crime

ZN
: a numeric vector of proportions of residential land zoned for lots over 25000 sq. ft per town (constant for all Boston tracts)

INDUS
: a numeric vector of proportions of non-retail business acres per town (constant for all Boston tracts)

CHAS
: a factor with levels 1 if tract borders Charles River; 0 otherwise

NOX
: a numeric vector of nitric oxides concentration (parts per 10 million) per town

RM
: a numeric vector of average numbers of rooms per dwelling

AGE
: a numeric vector of proportions of owner-occupied units built prior to 1940

DIS
: a numeric vector of weighted distances to five Boston employment centres

RAD
: a numeric vector of an index of accessibility to radial highways per town (constant for all Boston tracts)

TAX
: a numeric vector full-value property-tax rate per USD 10,000 per town (constant for all Boston tracts)

PTRATIO
: a numeric vector of pupil-teacher ratios per town (constant for all Boston tracts)

B

	a numeric vector of `1000*(Bk - 0.63)^2` where Bk is the proportion of blacks
LSTAT	a numeric vector of percentage values of lower status population

Part 0: Install packages

```
install.packages("spdep")

install.packages("maptools")

install.packages("sp")

install.packages("lmtest")

install.packages("rgdal")

install.packages("spstat")

install.packages("car")

install.packages("spgwr")

install.packages("RColorBrewer")

install.packages("spatialreg")


# install other packages as necessary on your own RStudio.
```

Part 1: Spatial Regression

```
library(maptools)     ## Data management

library(sp)         ## Data management

library(spdep)       ## Spatial autocorrelation

library(lmtest)

library(spatialreg)

## read shapefile from the system

Boston <- readOGR(system.file("shapes/boston_tracts.shp", package="spData")[1])

class(Boston)

plot(Boston)


# creates a row-standardized "listw" object (default style = "W")
```

```
Boston_queen_nb <- poly2nb(Boston, queen=TRUE)

Boston_w <- nb2listw(Boston_queen_nb)


# creates a non row-standardized "listw" object

Boston_b <- nb2listw(Boston_queen_nb,style="B")

# displays the "data.frame" attribute of the shape file

attr(Boston,"data")

# using log of median housing value instead of housing value

Boston$LOGMEDV<-log(Boston$CMEDV)


# OLS for housing value

Boston_OLS <- lm(LOGMEDV~CRIM + CHAS + NOX + RM + AGE + DIS + B + LSTAT, data=Boston)

summary(Boston_OLS)

# saving residuals and fitted values

Boston$olsresid<-residuals(Boston_OLS)

Boston$ols_fitted <- fitted(Boston_OLS)

# Moran test for residuals

lm.morantest(Boston_OLS, Boston_w)
```

Q1. Describe the OLS regression model results and their significance. (10 points)

```
# Lagrange multiplier tests for residuals

lm.LMtests(Boston_OLS, Boston_w, test="all")

#Breusch Pagan test for heteroskedasticity

bptest(Boston_OLS)
```

Q2. Describe the diagnostics test results for spatial autocorrelation. (10 points)

```
# spatial lag regression

Boston_spatial_lag <- lagsarlm(LOGMEDV ~ CRIM + CHAS + NOX + RM + AGE + DIS + B + LSTAT,
data=Boston, Boston_w)
```

```
summary(Boston_spatial_lag)

# saving residuals and fitted values

Boston$lagresid<-residuals(Boston_spatial_lag)

Boston$lag_fitted <- fitted(Boston_spatial_lag)
```

Q3. Describe the spatial lag regression model results and its significance. (10 points)

```
# Error regression

Boston_spatial_error <- errorsarlm(LOGMEDV ~ CRIM + CHAS + NOX + RM + AGE + DIS + B + LSTAT,
data=Boston, Boston_w)

summary(Boston_spatial_error)

# saving residuals and fitted values

Boston$error_resid<-residuals(Boston_spatial_error)

Boston$error_fitted <- fitted(Boston_spatial_error)
```

Q4. Describe the spatial error regression model results and its significance. (10 points)

Part 2: GWR

```
library(classInt)

library(car) #install first

library(spdep)

library(spgwr) #install first

library(RColorBrewer)

require(spatstat)

# create a data frame

boston.df <- data.frame(Boston)

names(boston.df)     # gives us the column names

colnames(boston.df)  # alternative call; same thing

attach(boston.df) #Now you can call the variable name without using “$”

LOGCMEDV <- log(CMEDV)
```

We need a grid reference for the location of each observation.

we create this location using the coordinates of the centroids:

XYgridtable <- cbind(LON, LAT)

#using gwr.sel to find optimal parameters

adaptive.bw<-gwr.sel(LOGCMEDV~CRIM+CHAS+NOX+RM+AGE+DIS+B+LSTAT,adapt=TRUE, method="cv", coords=XYgridtable)

###gwr.Gauss() default

#get optimal proportion of observations to include in weighting scheme (k-nearest neighbours): approximately 0.01979116

each regression will expand or contract so as to include roughly 2% of sample

adaptive.bw

#number of observations (polygon)

dim(boston.df)[1]

how many points does each window take

print(dim(boston.df)[1]*adaptive.bw)

#run gwr using the optimal parameter found above

bos.gwr<-gwr(LOGCMEDV ~ CRIM + CHAS + NOX + RM + AGE + DIS + B + LSTAT, adapt = adaptive.bw, gweight = gwr.Gauss, hatmatrix = TRUE, coords = XYgridtable)

print(bos.gwr)

#Plot local R^2

classes_fx <- classIntervals(bos.gwrSDFlocalR2, n=5, style="quantile")

cols <- findColours(classes_fx,pal)

pal <- brewer.pal(5, "YlOrRd")

plot(Boston,col=cols, border="grey")

legend(x="topleft", cex=0.65, fill=attr(cols,"palette"), bty="n", legend=names(attr(cols, "table")), title="Local R^2",ncol=1)

#Plot Residual

```
res <- bos.gwr$SDF$gwr.e

classes_fx <- classIntervals(res, n=5, style="quantile")

pal <- brewer.pal(5, "YlOrRd")

cols <- findColours(classes_fx,pal)

plot(Boston,col=cols, border="transparent")

legend(x="topleft", cex=0.65, fill=attr(cols, "palette"), bty="n", legend=names(attr(cols, "table")),
title="Residual", ncol=1)
```

Q5. Take screenshots of the plots and describe the pattern of local R^2 and residual. (10 points)

```
###plot local coefficient (CRIM)

coef <- bos.gwr$SDF$CRIM

classes_fx <- classIntervals(coef, n=5, style="quantile")

cols <- findColours(classes_fx,pal)

plot(Boston,col=cols, border="transparent")

legend(x="topleft", cex=0.75, fill=attr(cols, "palette"), bty="n", legend=names(attr(cols, "table")), title =
"Local Coefficient Estimates (CRIM)", ncol=1)

###plot local coefficient (NOX)

coef <- bos.gwr$SDF$NOX

classes_fx <- classIntervals(coef, n=5, style="quantile")

cols <- findColours(classes_fx,pal)

plot(Boston,col=cols, border="transparent")

legend(x="topleft", cex=0.7, fill=attr(cols, "palette"), bty="n", legend=names(attr(cols, "table")),
title="Local Coefficient Estimates (NOX)", ncol = 1)
```

Q6. Take a screen shot of the plot and describe the patterns. Also plot other independent variables in the gwr and take a screenshot. (10 points)

```
#plot t values (CRIME)

bos.gwr$SDF$CRIM_t <- bos.gwr$SDF$CRIM/bos.gwr$SDF$CRIM_se

tCRIM<-bos.gwr$SDF$CRIM_t

display.brewer.pal(3, "RdBu")

colors <- brewer.pal(3, "RdBu")  # Stores colors in object color

color.category.reg <- findInterval(tCRIM,
```

```
                    c(min(tCRIM)-.0001, -1.96, -1.64,
                     max(tCRIM)+.0001),all.inside=TRUE)
#classes_fx <- classIntervals(tCRIM, n=5, style="quantile")
#cols <- findColours(classes_fx,pal)
plot(Boston,col=colors[color.category.reg], border="transparent")
labels <- c("<-1.96", " -1.96 to  -1.64", "Not significant")
legend("bottom", legend=labels, fill=colors, cex=0.8,
    y.intersp = 0.99, bty="n")
```

Q7. Take a screen shot of the plot and describe the pattern. (10 points)

Q8. Describe the GWR model results and their significance. How does it compare with the regression results in part 1? (10 points)

Q9. How could you use these results as a policymaker who is interested in environmental or social issues (pollution or affordable housing)? (20 points)