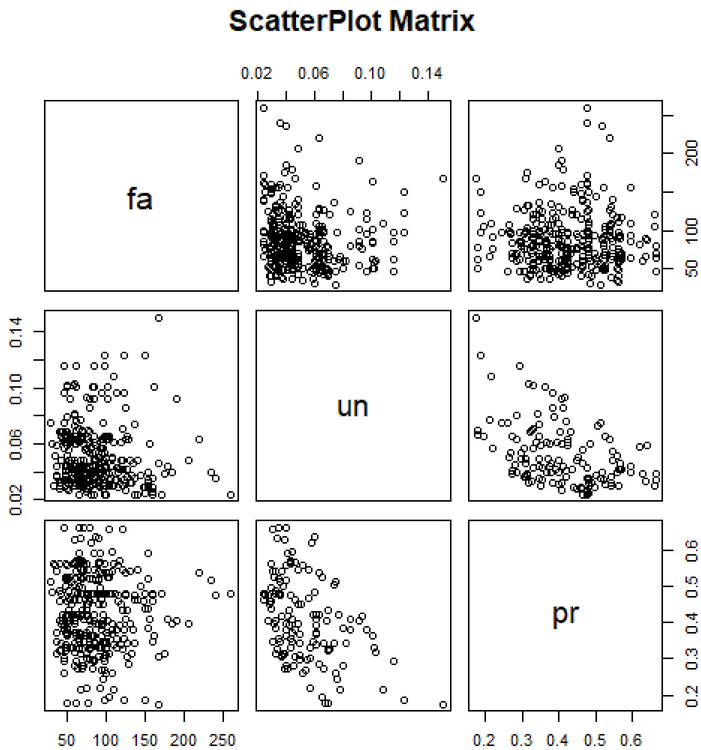
Jake Kruse

Lab 2

Geog 560

(1) (2 pts) Which combination of the three explanatory variables can achieve the best goodness of fit for the house purchase price estimation? (Hint: all-possible-subsets regression for selecting regression variables.)

Scatterplot matrix (with fa=FlOORSZ, un=UNEMPLOY, pr=PROF variables) to check for relationships. PROF and UNEMPLOY vary together.



Variables:

pp<- londonhp$PURCHASE

fa <- londonhp$FLOORSZ

un <- londonhp$UNEMPLOY

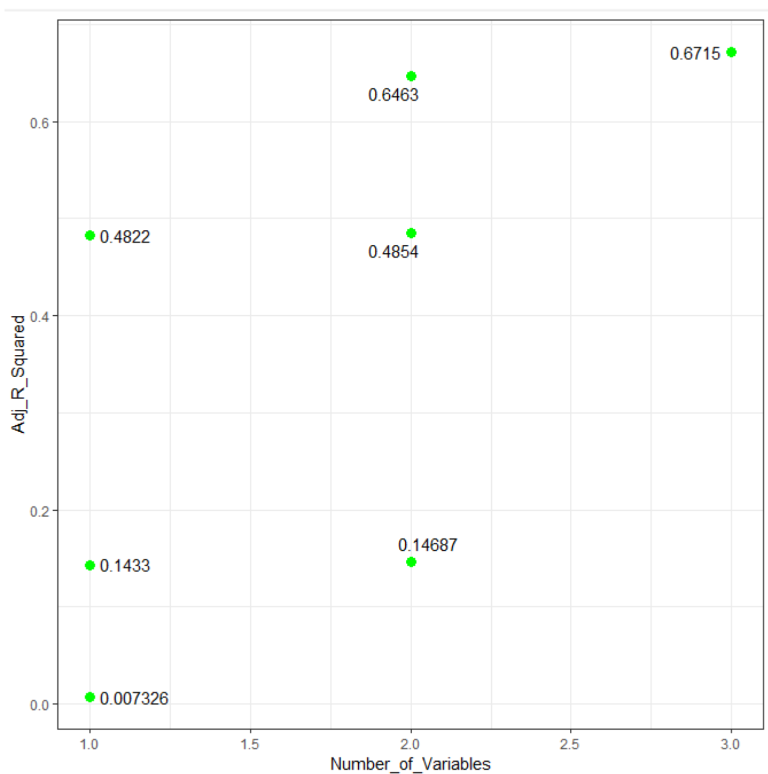
pr <- londonhp$PROF

Adjusted R^2 per model:

a <- lm(pp~fa) #0.4844

```
b <- lm(pp~un) #0.007326
c <- lm(pp~pr) #0.1433
d <- lm(pp~fa+un) #0.4854
e <- lm(pp~fa+pr) #0.6463
f <- lm(pp~un+pr) #0.14687
g <- lm(pp~fa+un+pr) #0.6715
```

Number of Variables in MLR vs Adjusted R^2 Value



All three variables together produced the highest adj-R^2 value (0.6715), but FLOORSZ and PROF were able to explain almost as much of the variability while using one less variable (R^2 of 0.6463).

Find the predictor that has the largest positive influence on the house price. (Hint: Standardized regression coefficients) fa (FLOORSZ) has the highest standardized coefficient value (0.725), which is much more than the other standardized coefficients of 0.491 and 0.185 (PROF and UNEMPLOY, respectively).

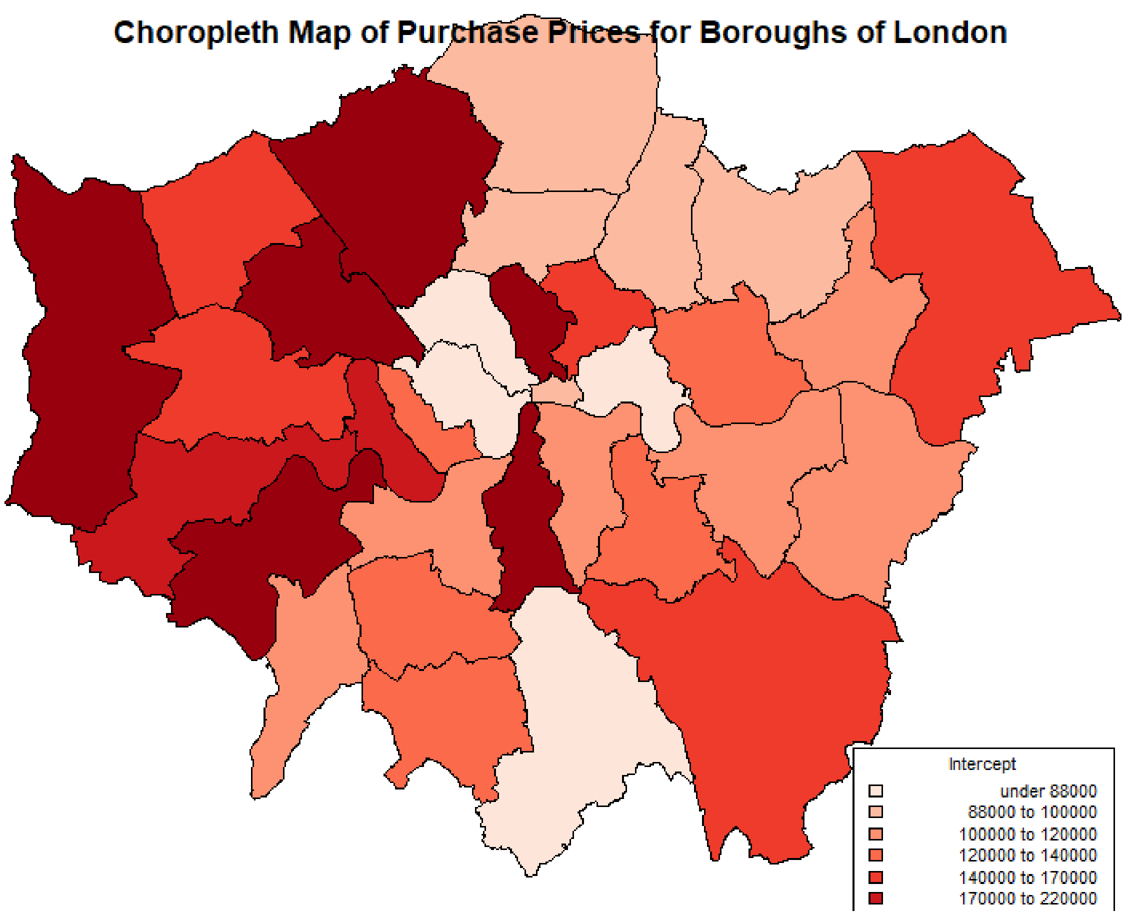
(2) (2 pts) Is the spatial distribution of house purchase prices spatially dependent? Which of the following spatial autoregressive model fits the data better? the spatial lag model or the spatial error model?

Spatial Lag Model Log likelihood: AIC: 7634.3

Spatial Error Model AIC: 7627.4

Spatial Error Model performed better for this dataset, as AIC was lower.

The following map shows that purchase prices do indeed vary spatially in London (units of Purchase Price not specified in package).

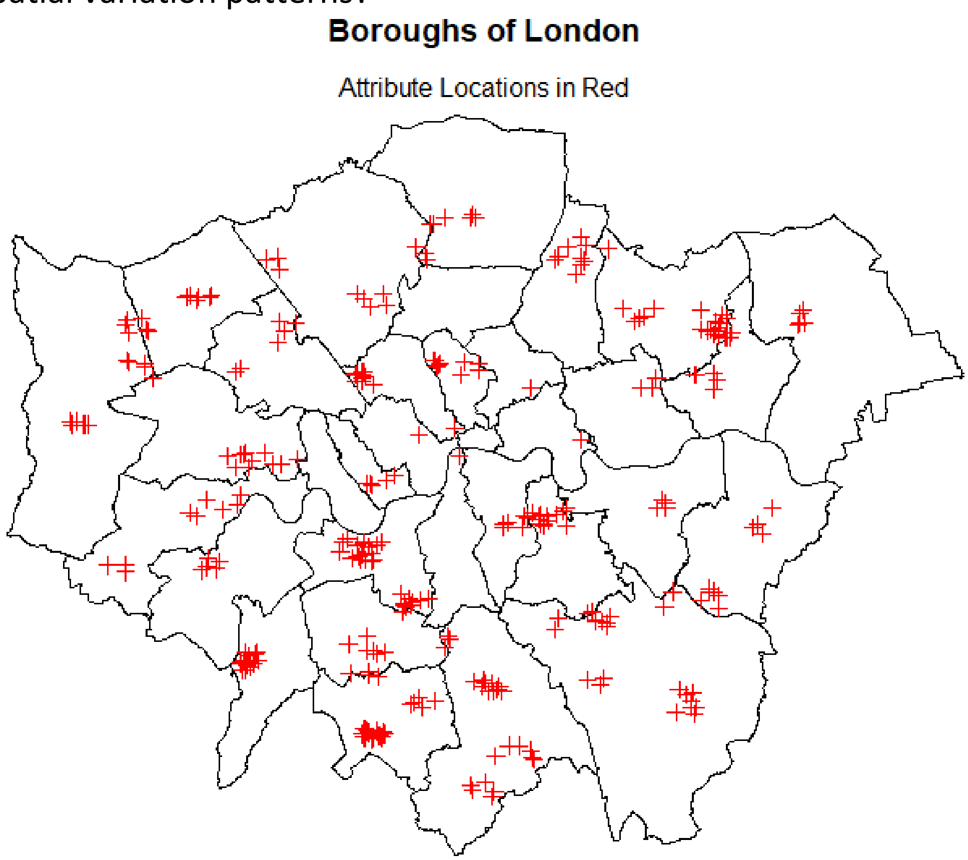


(3) (2 pts) If a Gaussian kernel is selected in the GWR including all above three predictors, what is the optimal bandwidth that minimizes the least squared

errors (cross-validation approach) between the observations and the estimated values. (Hint: help(bw.gwr))

Fixed bandwidth: 3362.161 CV score: 469722722823

(4) (4 pts) Choose the Gaussian kernel and use the optimal bandwidth from the previous question and conduct a GWR analysis. Does the GWR model improve the goodness of fit score and reduce the standard error comparing with the global regression model for the estimation of house price? Exploring the maps for the GWR coefficient estimations, do you find any spatial variation patterns?

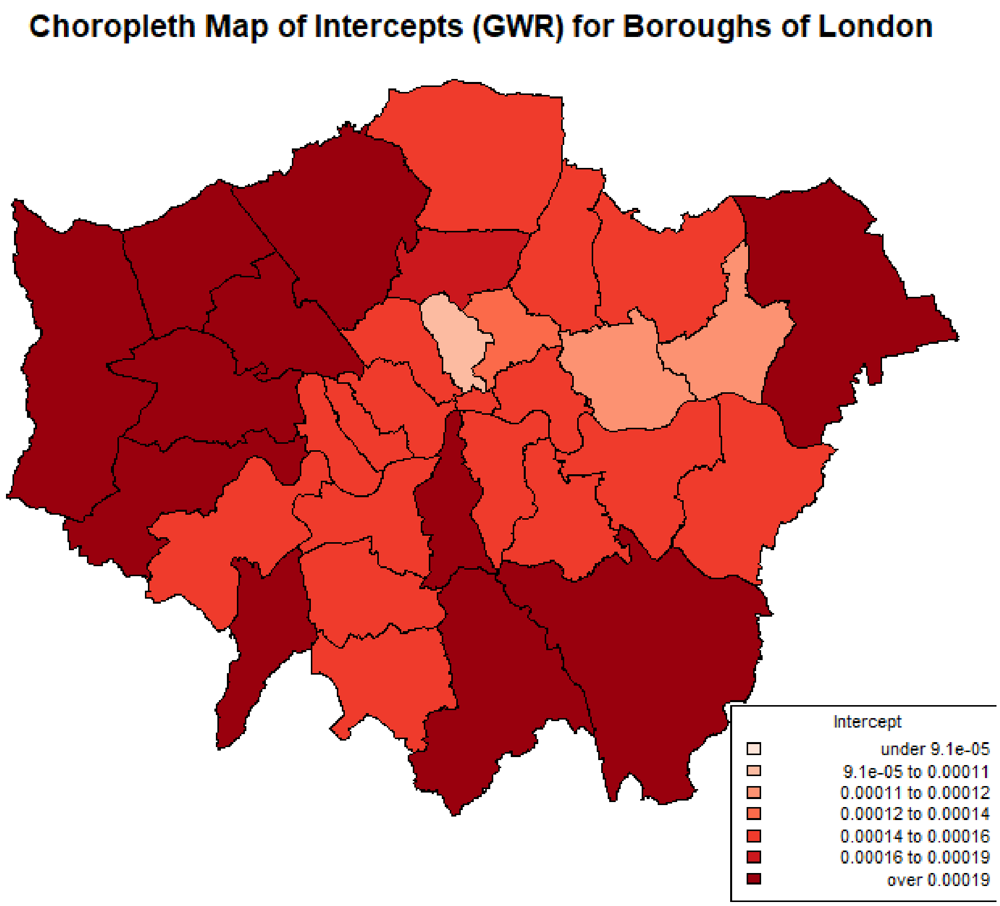


Global LR: Residual standard error: 43220 on 312 degrees of freedom;
Multiple R-Squared: 0.6746

GWR: RMSE/RSE for GWR: 3967.042   Multiple R-Squared: 0.8554501

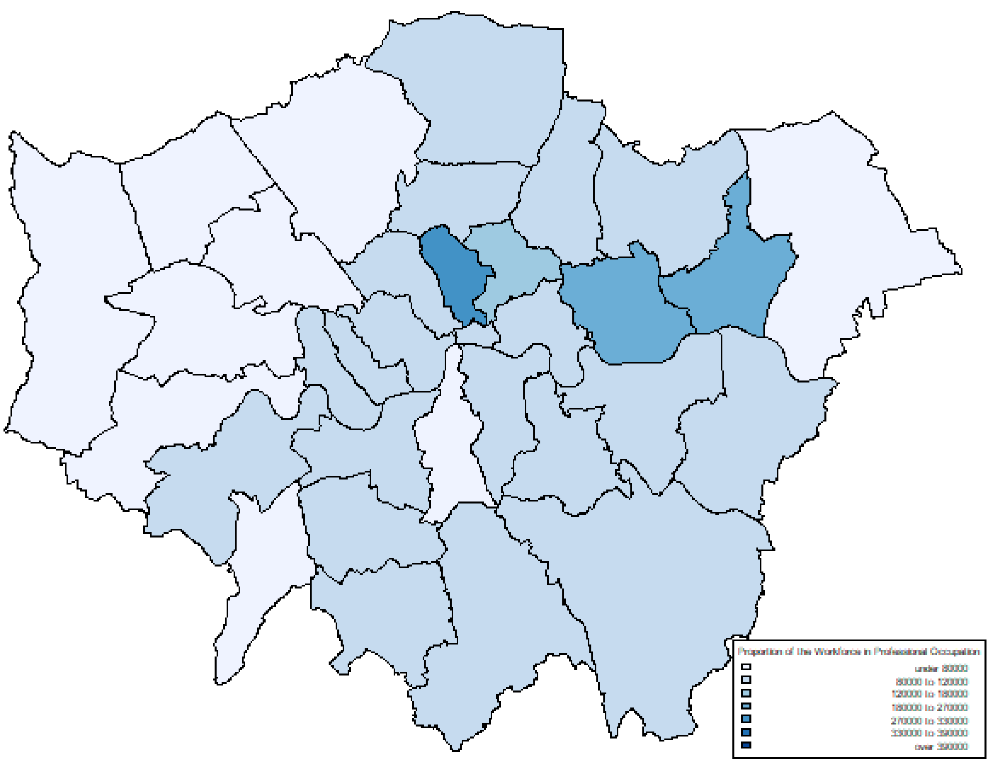
GWR significantly improved the goodness of fit and reduced the standard error.

For the intercepts, the outer boroughs tended to haver higher values.

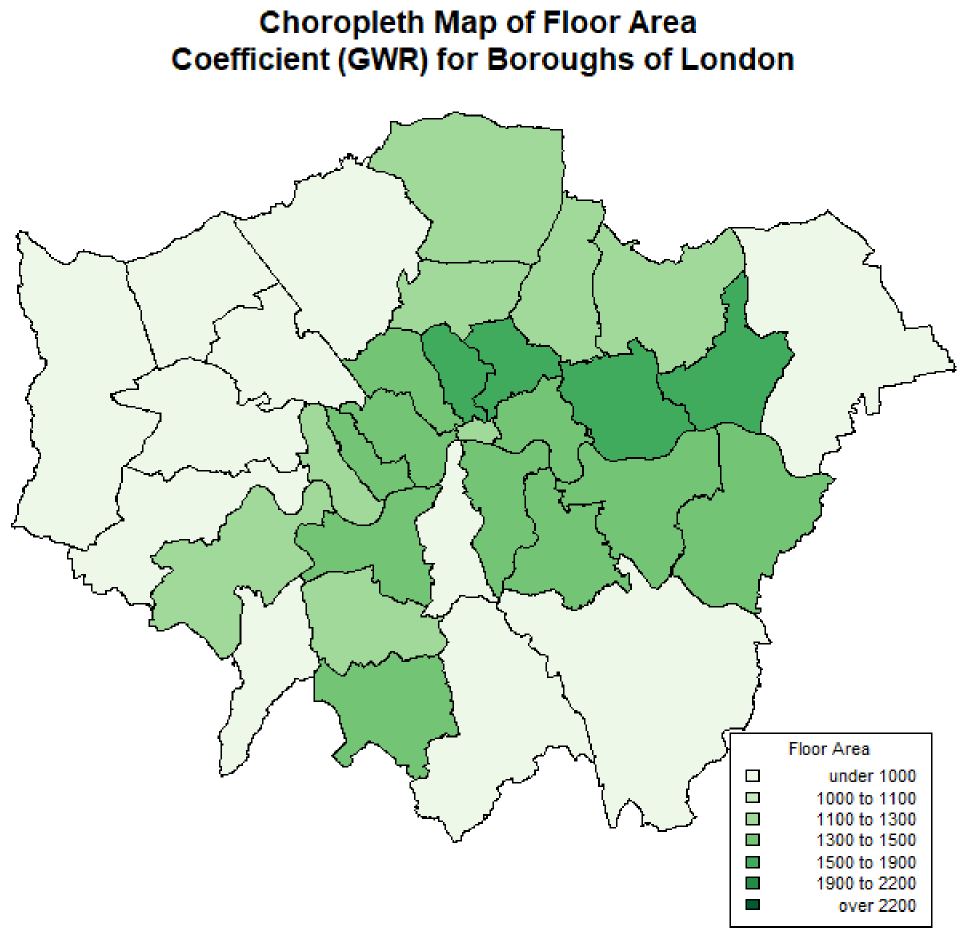


Coefficients for proportions of professional/managerial workforce northwest of the center of London are higher, with the outer boroughs having the lowest values.

Choropleth Map of Proportion of the Workforce in Professional Occupation Coefficient (GWR) for Boroughs of London

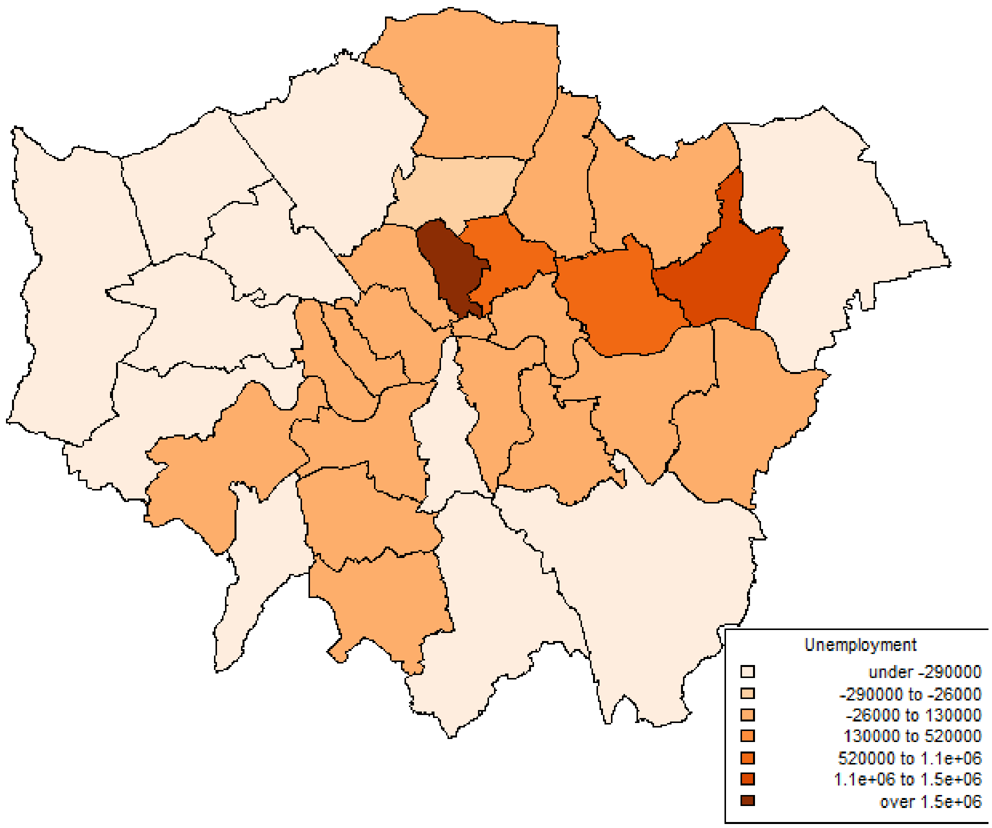


GWR coefficients for floor area tended to be highest in the center of London.



GWR coefficients for unemployment are highest in the city center, in almost the same distribution as coefficients for Floor Area and Proportion of Workfor in Profession occupatiosn.

Choropleth Map of Unemployment Coefficient (GWR) for Boroughs of London




```
#Jake Kruse
##Geog 560, Lab 2


#load GWmodel package, which has LondonHP data
if (!require("GWmodel")) install.packages('GWmodel')
library('GWmodel')
if (!require("GISTools")) install.packages('GISTools')
library('GISTools')
if(!require("ggrepel")) install.packages('ggrepel')
```

```
library('ggrepel')
library(ggplot2)
if (!require("rgdal"))install.packages("rgdal")
library(rgdal)
if (!require("raster"))install.packages("raster")
library(raster)

library(maptools)
library(RColorBrewer)
library(spdep)
##Some useful data from the package
##londonhp$PURCHASE: the purchase price of the property (Independent
Variable)
##londonhp$FLOORSZ: floor area of the property in square metres
##londonhp$UNEMPLOY: the rate of unemployment in the census ward in
which the house is located
##londonhp$PROF: the proportion of the workforce in professional or
managerial occupations in the census ward in which the house is located

#1
data(LondonHP) #Load in the spatialploygonsdf

names(londonhp) #Column names
summary(londonhp)#Summary stats

#2^k different MLR equations that can be constructed; 3 explanatory
variables = 2^3=8 eq.
#One is with no  variables, so essentially 7 eqs
pp<- londonhp$PURCHASE
fa <- londonhp$FLOORSZ
un <- londonhp$UNEMPLOY
pr <- londonhp$PROF


#model determinants of purchase price in London
pairs(~fa+un+pr, main = "ScatterPlot Matrix") #pr and un have a
relationship
```

```
a <- lm(pp~fa)
b <- lm(pp~un)
c <- lm(pp~pr)
d <- lm(pp~fa+un)
e <- lm(pp~fa+pr)
f <- lm(pp~un+pr)
g <- lm(pp~fa+un+pr)

#Adj R^2 to right
summary(a)#0.4844
summary(b)#0.007326
summary(c)#0.1433
summary(d)#0.4854
summary(e)#0.6463
summary(f)#0.14687
summary(g)#0.6715

#AdjR^2 plot
Adj_R_Squared <-
c(0.4822,0.007326,0.1433,0.4854,0.6463,0.14687,0.6715)
Number_of_Variables <- c(1,1,1,2,2,2,3)
Combination_of_Variables <- c(a,b,c,d,e,f,g)
df <- data.frame(x=Adj_R_Squared, y=Number_of_Variables, z =
Combination_of_Variables)

ggplot(data=df, aes(x=Number_of_Variables, y = Adj_R_Squared)) +
theme_bw() +
geom_text_repel(aes(label = Adj_R_Squared), box.padding = unit(0.45,
"lines")) + geom_point(colour = "green", size = 3)

#2 Is the data spatially dependent? Which fits better, the spatial lag model
or the spatial error model?
#spatial distance based on attribute points (used as reference location())
#overlay <- over(londonborough,londonhp)
#par(mar=c(10,10,10,10))
#plot(overlay)
  #Use distance to points to build weights list
```

```
#lag model
#k nearest neighbors; not overlayed on polygons
k <- 3
nn <- knearneigh(londonhp, k)
londonhp.neighbor.knn <- knn2nb(nn)
london.b <- nb2listw(londonhp.neighbor.knn)

lagmod <- lagsarlm(pp~fa+un+pr, data = londonhp, listw=london.b, type =
"lag")
summary(lagmod)

#spatial error model
elm=errorsarlm(g,data=londonhp, listw = london.b)

#variable map
Shading1 <- auto.shading(pp, n=7, cols = brewer.pal(n=7, "Reds")) #set a
shading style
par(mar=c(0,0,0,0))
choropleth(londonborough, pp, shading = Shading1) # create a chroploeth
map
choro.legend(548000.5, 162592.4,Shading1, cex=.7,title="Intercept") # add
a legend
title(main = "Choropleth Map of Purchase Prices for Boroughs of London",
line=-2)

#3
#multilinearMod <- lm(georgia$PctBach ~ georgia$MedInc+ georgia$PctEld
+ georgia$PctFB + georgia$PctPov, data=georgia)
#g <- lm(pp~fa+un+pr)
coefficients_lm <- coefficients(g)
coefficients_lm[1] #intercept
coefficients_lm[2] #fa
coefficients_lm[3] #un
coefficients_lm[4] #pr

std_coef_fa <- coefficients_lm[2] * sd(fa) / sd(pp)#first var
print(std_coef_fa)
```

```
std_coef_un <- coefficients_lm[3] * sd(un) / sd(pp)#second var
print(std_coef_un)

std_coef_pr <- coefficients_lm[4] * sd(pr) / sd(pp)#third var
print(std_coef_pr)

#3,4
# Compute the distances between the data points and the reference
location i
DM <-
gw.dist(dp.locat=coordinates(londonhp),rp.locat=coordinates(londonhp),p
=2,longlat=FALSE)

# Automatic bandwidth selection to calibrate a basic GWR model
BW <- bw.gwr(pp~fa+un+pr, data=londonhp,
approach="CV",kernel="gaussian", adaptive=FALSE, p=2, theta=0,
longlat=FALSE)#CV=cross validation ; fixed

# Fixed Distance Kernel
gwr.res <- gwr.basic(pp~fa+un+pr, data=londonhp,
regression.points=londonhp,kernel='gaussian', adaptive=FALSE,
bw=BW,longlat=FALSE, cv=TRUE, dMat=DM)#Assumed londlat=FALSE, since
using coordinates?

# Adaptive Kernel with fixed number of nearest neighbors
#gwr.res <- gwr.basic(pp~fa+un+pr, data=londonhp,kernel='gaussian',
adaptive=TRUE, BW)
#gwr.res


#plot geometry and add borough labels
par(mar=c(0,0,0,0))
plot(londonborough)
plot(londonmap, add = TRUE, col="red",bg="grey") # draw the polygons
title('Boroughs of London', line=-1) #add the title
mtext(text="Attribute Locations in Red", side=3, line=-3)
```

```
#Lat <- londonhp$X
#Lon <- londonhp$Y
#Name <- londonborough$NAME
#pl<- pointLabel(x=Lat,y=Lon,labels=Name,offset=0,cex=0.6) #add the point
name labels
#pts <- points(Lon,Lat,col="purple",pch=20) #add the centroids of counties;
need to calculate centroids?
#"pch=optional number" see below link for more information
#https://www.statmethods.net/advgraphs/parameters.html

#Chloropleth maps of intercept, coefficients for gwr.sdf
#use locator to find coordiantes for legend placement
leg_locator <- locator(2)
print(leg_locator)


gwr.sdf <- gwr.res$SDF
head(gwr.res$SDF)

Shading1 <- auto.shading(gwr.sdf$Intercept, n=7, cols = brewer.pal(n=7,
"Reds")) #set a shading style
par(mar=c(0,0,0,0))
choropleth(londonborough, gwr.sdf$Intercept, shading = Shading1) #
create a chroploeth map
choro.legend(548000.5, 162592.4,Shading1, cex=.7,title="Intercept") # add
a legend
title(main = "Choropleth Map of Intercepts (GWR) for Boroughs of London",
line=-2)

Shading2 <- auto.shading(gwr.sdf$pr, n=7, cols = brewer.pal(n=7, "Blues"))
#set a shading style
choropleth(londonborough, gwr.sdf$pr, shading = Shading2) # create a
chroploeth map
choro.legend(548000.5, 162592.4,Shading2, cex=.4, title="Proportion of
the Workforce in Professional Occupation") # add a legend
```

```
title(main = "Choropleth Map of Proportion of the Workforce in
Professional\n Occupation Coefficient (GWR) for Boroughs of London",
line=-2)

Shading3 <- auto.shading(gwr.sdf$fa, n=7, cols = brewer.pal(n=7,
"Greens")) #set a shading style
choropleth(londonborough, gwr.sdf$fa, shading = Shading3) # create a
chroploeth map
choro.legend(548000.5, 162592.4,Shading3, cex=.5,title="Floor Area") #
add a legend
title(main = "Choropleth Map of Floor Area \nCoefficient (GWR) for
Boroughs of London", line=-2)

Shading4 <- auto.shading(gwr.sdf$un, n=7, cols = brewer.pal(n=7,
"Oranges")) #set a shading style
choropleth(londonborough, gwr.sdf$un, shading = Shading4) # create a
chroploeth map
choro.legend(548000.5, 162592.4,Shading4, cex=.7,title="Unemployment")
# add a legend
title(main = "Choropleth Map of Unemployment \nCoefficient (GWR) for
Boroughs of London", line=-2)

## Goodness of Fit
pred_pp <- gwr.sdf$Intercept +
  fa*gwr.sdf$fa +
  un*gwr.sdf$un +
  pr*gwr.sdf$pr


gwr.residuals <- pp - pred_pp # actual obervation - prediction value
ESS <- sum(gwr.residuals*gwr.residuals) #error sum of squared
pp_var <- pp - mean(pp)
TSS <- sum(pp_var*pp_var)# total sum of squared variation
RSS <- TSS - ESS #get the regression sum of squared variation
gwr.RSquared <- RSS/TSS
print(gwr.RSquared)
print(gwr.res)
```

```
#Global LR: R-Squared 0.6746
#GWR: R-Squared 0.8554501

# Root of Mean Squared Error (RMSE) or Regression Standard Error
# RMSE = Squared Root of the Regression Sum of Squared / Degree of Freedom
gwr.RMSE <- sqrt(RSS) / (length(pp) - length(coefficients(g)))
print(gwr.RMSE)
#Global LR: Residual standard error: 3.784 on 154 degrees of freedom
#GWR: Residual standard error: 0.4102733 on 154 degrees of freedom
```