spatial analysis

Annabelle

April 28, 2020

Set up file structure

load in data

## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:  
## cols(  
## X1 = col\_double(),  
## total = col\_double(),  
## area\_x = col\_double(),  
## percentpublic = col\_double(),  
## taxa = col\_double(),  
## nsumemploy = col\_double(),  
## total\_x\_x = col\_double()  
## )

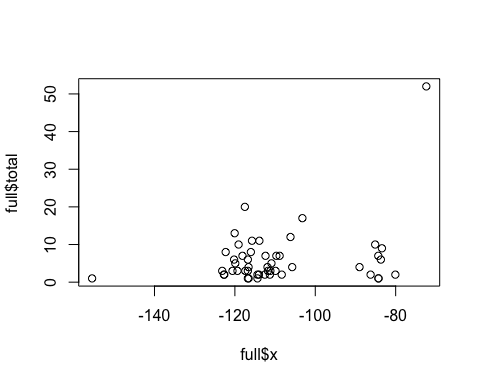
**Need to check is species in center got cleaned after Detailed\_methods was created - I think might have some species in here that no longer have partner information**

cleaning and combing with combo

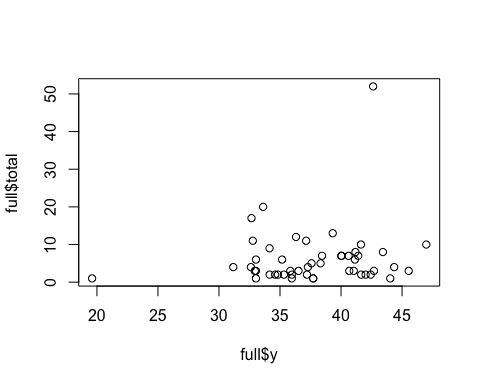
options for analysis

Add coords information to glmm (lecture 6)

#plot against the response variable   
plot(full$total ~ full$x) #almost definiety not independent on westcoast



plot(full$total ~ full$y)



numfull <- full[,c(3:4,9:14)] #with Coords   
  
  
length(numfull$total)

## [1] 49

choose(51,2)

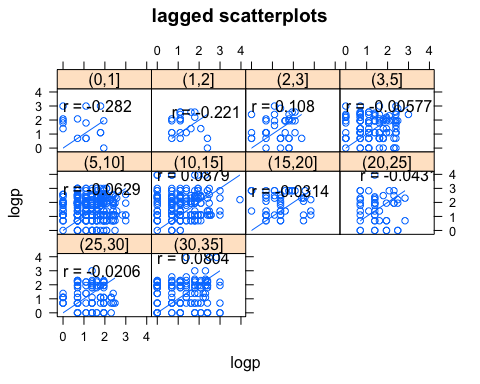
## [1] 1275

#plot(variogram(total~1,numfull,cloud=T))  
  
  
#variogram   
#?variogram  
#vario.Head =variogram(numfull~1,data=numfull) #error "NA values in coordinates"  
#vario.Head.fit=fit.variogram.reml(head~1,Head,model=vgm(60000,"Gau",5,0)) #code from lecture   
#plot(vario.Head,vario.Head.fit,pch=19) #code from lecture   
  
#plot(variogram(total~1,numfull,cloud=T)) #different function in sp package? Can't find  
  
#total#standard linear analysis   
##remove x effect and examine residuals   
  
## predictor ~ x +y (see if y is contributing something significant to the fit)  
 #then check vairogram   
  
  
# Q - more straight forward to just use coords in glmm regression? numFactor?

As part of regression, split into east coast v west values to see if could graph differences in part 1 of analysis (didn’t have time to use this in thesis)

Generate variogram with raw data - text from lecture 10

Head <- full #made in chunk above  
Head <- Head[,c(5,6,9)] #select only necessary columns   
Head <- as.data.frame(Head)  
#  
Head$logp = log(Head$total)  
#  
# remove NA values   
### Head <- Head[-which(is.na(Head$total)),]  
#  
library(sp)  
#data(meuse)  
#  
coordinates(Head) = c("x", "y")  
#  
library(RColorBrewer)  
library(classInt)  
library(gstat)  
 # hscat(logp~1,data=Head, breaks=c(0,5,10,15,20,25,30,200))  
hscat(logp~1,data=Head, breaks=c(0,1,2,3,5,10,15,20,25,30,35)) ## axes are wrong? Plots look square..



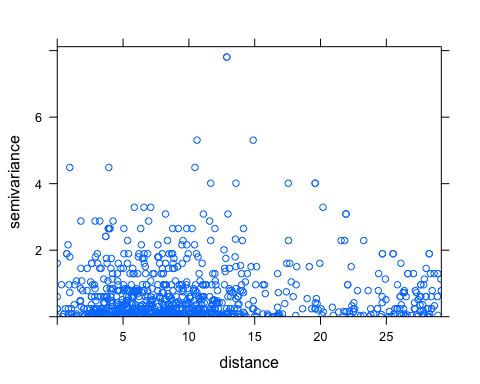
#  
# How many points are there?  
#  
length(Head$logp)

## [1] 49

#  
# How many pairs of points?  
#  
choose(49,2)

## [1] 1176

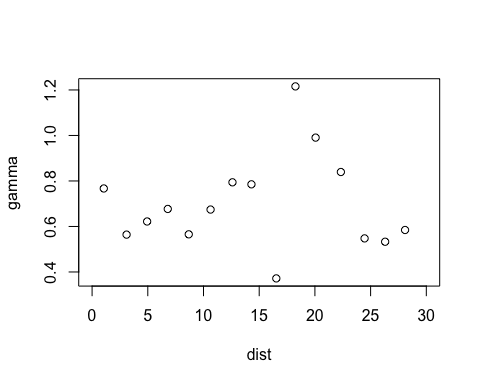
#  
# Scatter plot of squared differences  
#  
plot(variogram(logp~1,Head,cloud=T))



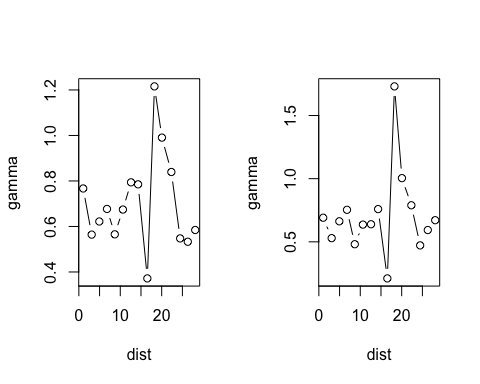
#  
# Differences averaged for distance increments  
# to show empirical variogram  
#  
logzinc.vario = variogram(logp~1,Head)  
#  
# Take a look at the contents of logzinc.vario  
#  
logzinc.vario

## np dist gamma dir.hor dir.ver id  
## 1 39 1.049629 0.7670071 0 0 var1  
## 2 91 3.095180 0.5641947 0 0 var1  
## 3 122 4.940534 0.6219304 0 0 var1  
## 4 115 6.792845 0.6769958 0 0 var1  
## 5 128 8.676215 0.5654667 0 0 var1  
## 6 104 10.634332 0.6743748 0 0 var1  
## 7 76 12.599758 0.7940539 0 0 var1  
## 8 34 14.303532 0.7852107 0 0 var1  
## 9 22 16.535099 0.3717494 0 0 var1  
## 10 12 18.252912 1.2156208 0 0 var1  
## 11 16 20.062575 0.9905548 0 0 var1  
## 12 23 22.331722 0.8393943 0 0 var1  
## 13 19 24.458810 0.5475721 0 0 var1  
## 14 34 26.304587 0.5328639 0 0 var1  
## 15 41 28.091918 0.5845019 0 0 var1

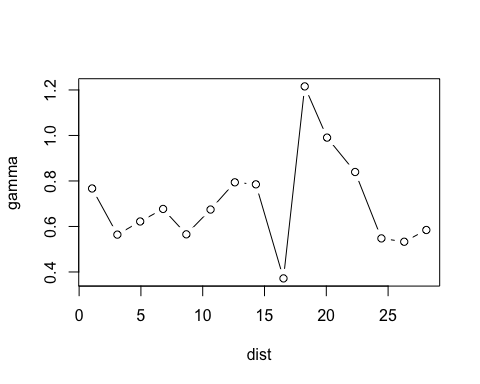
#  
# Now plot the empirical variogram  
# and identify the number of pairs of points  
# going into each estimate  
#  
plot(gamma~dist,data=logzinc.vario,xlim=c(0,30))



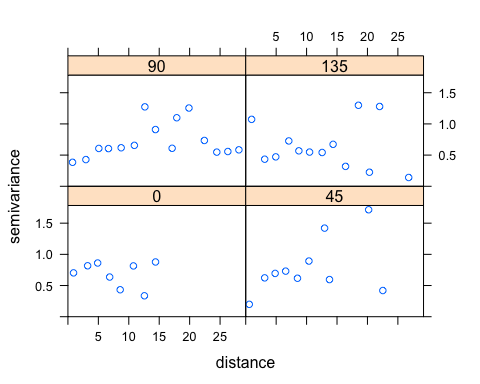
#text(logzinc.vario$dist+100,logzinc.vario$gamma,logzinc.vario$np)  
#  
# Sometimes outliers can have a big effect  
# Note that for this data set one outlier will go into the   
# computation of the squared difference 155 times.  
# Cressie suggests a robust measure of the variogram.  
#  
logzinc.vario.robust = variogram(logp~1,Head,cressie=T)  
#  
par(mfrow=c(1,2))  
plot(gamma~dist,data=logzinc.vario,type="b")  
#title("Classical Variogram")  
plot(gamma~dist,data=logzinc.vario.robust,type="b")



#title("Robust Variogram")  
par(mfrow=c(1,1))  
#  
#  
# \*\*\* this part of the code doesn't work right yet.. don't think the lines would be super relevant for this data anyway?   
#  
#  
# Do a permutation test by shifting the values associated with  
# each location around at random (i.e. permute them)  
# and compute the variogram and plot it each time this is done.  
#  
# First plot the original variogram estimate  
#  
plot(gamma~dist,data=logzinc.vario,type="n")  
#  
# Now do 100 permutations and plot the resulting "pure nugget" effect  
# Where our variogram lies outside this envelope show significant  
# correlation.  
#  
x=data.frame(Head)$x  
y=data.frame(Head)$y  
logzinc=Head$logp  
id0=seq(length(Head$logp))  
#for(i in 1:100)  
#{  
# id = sample(id0)  
# hold.perm = data.frame(x=x,y=y,logp=logp[id])  
# coordinates(hold.perm)=c("x","y")  
# hold.vario = variogram(logp~1,hold.perm)  
# lines(gamma~dist,data=hold.vario,col=2)  
#}  
lines(gamma~dist,data=logzinc.vario,col=1,type="b")

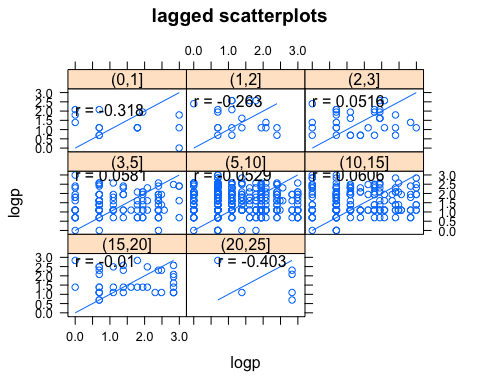


#  
# Examining for Anisotropy  
#  
logzinc.vario.dir = variogram(logp~1,Head,alpha=c(0,45,90,135))  
plot(logzinc.vario.dir)



Variogram for westcoast data \*\* Now having issues running code even thoguh it’s bug free - starting with coordinates line ..

Head <- fullwest #made in chunk above  
Head <- Head[,c(6,15,16)] #select only necessary columns   
#  
# might want to remove Weki bug in Hawaii row 37 #\*# see a big difference when this is removed  
Head <- Head[-c(37),]  
#  
Head$logp = log(Head$total)  
#  
# remove NA values   
### Head <- Head[-which(is.na(Head$total)),]  
#  
library(sp)  
#data(meuse)  
#  
coordinates(Head) = c("x", "y")  
#  
library(RColorBrewer)  
library(classInt)  
library(gstat)  
 # hscat(logp~1,data=Head, breaks=c(0,5,10,15,20,25,30,200))  
hscat(logp~1,data=Head, breaks=c(0,1,2,3,5,10,15,20,25,30,35)) ## axes are wrong? Plots look square..



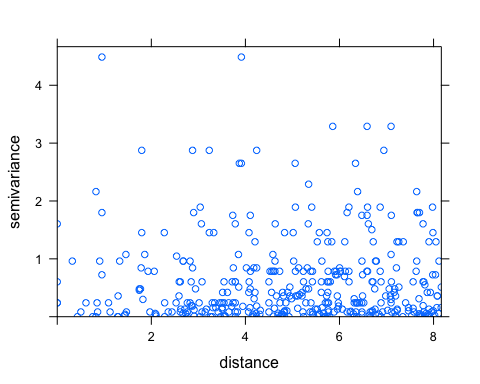
#  
# How many points are there?  
#  
length(Head$logp)

## [1] 39

#  
# How many pairs of points?  
#  
choose(40,2)

## [1] 780

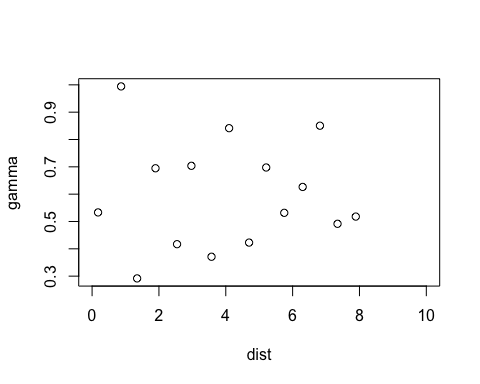
#  
# Scatter plot of squared differences  
#  
plot(variogram(logp~1,Head,cloud=T))



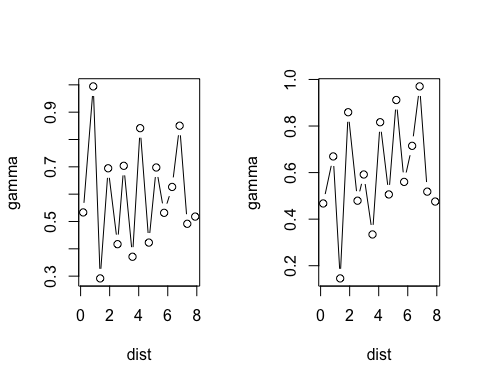
#  
# Differences averaged for distance increments  
# to show empirical variogram  
#  
logzinc.vario = variogram(logp~1,Head)  
#  
# Take a look at the contents of logzinc.vario  
#  
logzinc.vario

## np dist gamma dir.hor dir.ver id  
## 1 7 0.1786564 0.5331765 0 0 var1  
## 2 11 0.8683467 0.9943035 0 0 var1  
## 3 9 1.3473772 0.2917349 0 0 var1  
## 4 14 1.8969502 0.6947212 0 0 var1  
## 5 17 2.5419176 0.4169883 0 0 var1  
## 6 25 2.9710391 0.7037508 0 0 var1  
## 7 32 3.5730923 0.3709883 0 0 var1  
## 8 30 4.0992399 0.8414931 0 0 var1  
## 9 31 4.6972870 0.4227932 0 0 var1  
## 10 34 5.2094209 0.6975773 0 0 var1  
## 11 41 5.7494825 0.5316826 0 0 var1  
## 12 33 6.3004496 0.6263907 0 0 var1  
## 13 37 6.8175945 0.8504583 0 0 var1  
## 14 26 7.3433371 0.4915804 0 0 var1  
## 15 40 7.8895435 0.5179412 0 0 var1

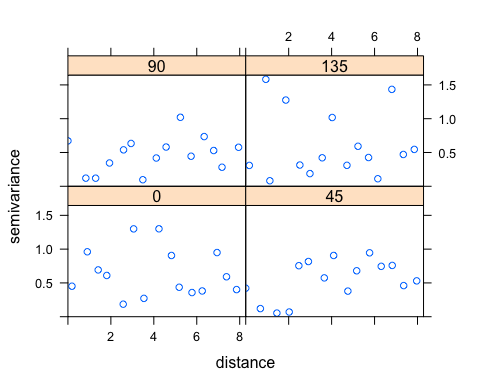
#  
# Now plot the empirical variogram  
# and identify the number of pairs of points  
# going into each estimate  
#  
plot(gamma~dist,data=logzinc.vario,xlim=c(0,10))



#text(logzinc.vario$dist+100,logzinc.vario$gamma,logzinc.vario$np)  
#  
# Sometimes outliers can have a big effect  
# Note that for this data set one outlier will go into the   
# computation of the squared difference 155 times.  
# Cressie suggests a robust measure of the variogram.  
#  
logzinc.vario.robust = variogram(logp~1,Head,cressie=T)  
#  
par(mfrow=c(1,2))  
plot(gamma~dist,data=logzinc.vario,type="b")  
#title("Classical Variogram")  
plot(gamma~dist,data=logzinc.vario.robust,type="b")



#title("Robust Variogram")  
par(mfrow=c(1,1))  
  
  
# Examining for Anisotropy  
#  
logzinc.vario.dir = variogram(logp~1,Head,alpha=c(0,45,90,135))  
plot(logzinc.vario.dir)



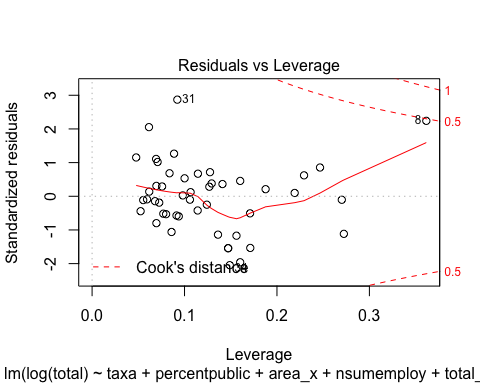
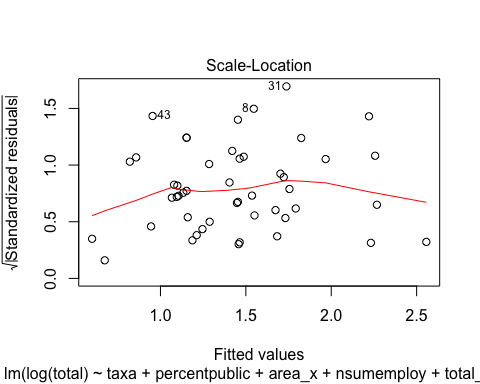
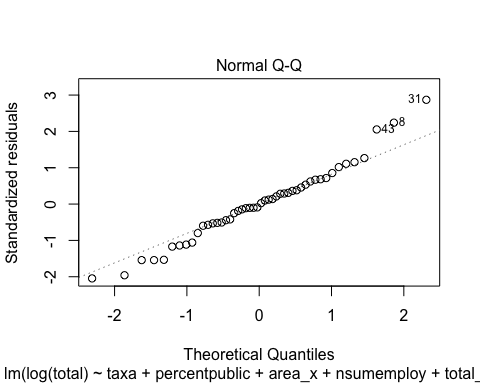
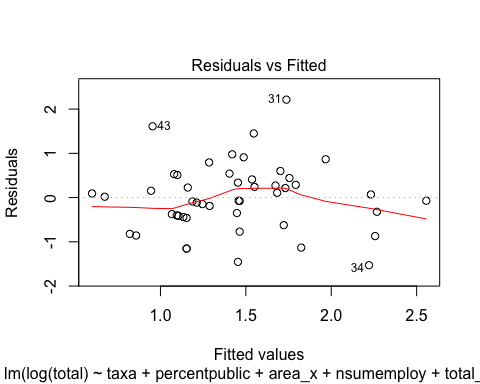
Linear regression with spatial components added

original model

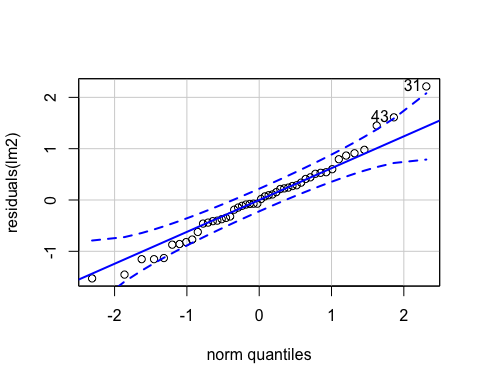
lm2 <- lm(log(total) ~ taxa + percentpublic + area\_x + nsumemploy + total\_x\_x, data=RegData)  
#withou log   
#lm2 <- lm((total) ~ taxa + percentpublic + area\_x + nsumemploy + total\_x\_x, data=RegData)  
#  
summary(lm2)

##   
## Call:  
## lm(formula = log(total) ~ taxa + percentpublic + area\_x + nsumemploy +   
## total\_x\_x, data = RegData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.5279 -0.4195 -0.0260 0.4178 2.2137   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.906e-01 3.600e-01 2.474 0.0175 \*  
## taxa -3.463e-01 2.647e-01 -1.308 0.1980   
## percentpublic 1.517e-02 4.471e-01 0.034 0.9731   
## area\_x 4.853e-12 3.607e-12 1.345 0.1857   
## nsumemploy 6.771e-07 3.084e-06 0.220 0.8273   
## total\_x\_x 2.543e-01 9.482e-02 2.682 0.0104 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8096 on 42 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.24, Adjusted R-squared: 0.1495   
## F-statistic: 2.653 on 5 and 42 DF, p-value: 0.03585

#  
#kable(table(lm2$coefficients))  
#  
par(mfrow = c(1, 1))  
plot(lm2)



#  
qqPlot(residuals(lm2))



## 31 43   
## 30 41

par(mfrow = c(1, 1))  
#  
#plot(predicted(lm2), residuals(lm2))  
#hist(residuals(lm2))  
# how add CI equivalent around q-q plot?   
AIC(lm2)

## [1] 123.5299

#check\_overdispersion(lm2)

put coordinates into model as predictors

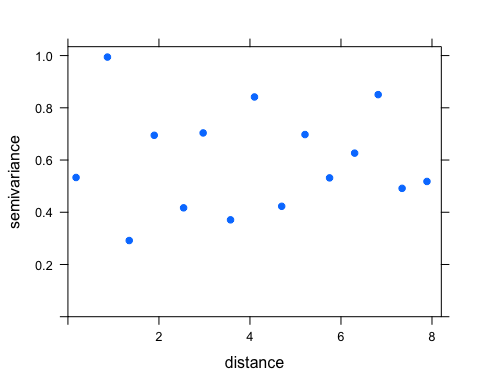
# Empirical variogram estimation  
#  
library(gstat)  
vario.Head =variogram(logp~1,data=Head)  
vario.Head.fit=fit.variogram.reml(logp~1,Head,model=vgm(60000,"Gau",5,0)) #Get a bunch of weird error messages here

## Warning in fit.variogram.reml(logp ~ 1, Head, model = vgm(60000, "Gau", :  
## singular V matrix in calc\_VinvIminAw  
  
## Warning in fit.variogram.reml(logp ~ 1, Head, model = vgm(60000, "Gau", :  
## singular V matrix in calc\_VinvIminAw

## Warning in fit.variogram.reml(logp ~ 1, Head, model = vgm(60000, "Gau", :  
## singular matrix in reml

## Warning in fit.variogram.reml(logp ~ 1, Head, model = vgm(60000, "Gau", :  
## no convergence while fitting variogram

plot(vario.Head,vario.Head.fit,pch=19)



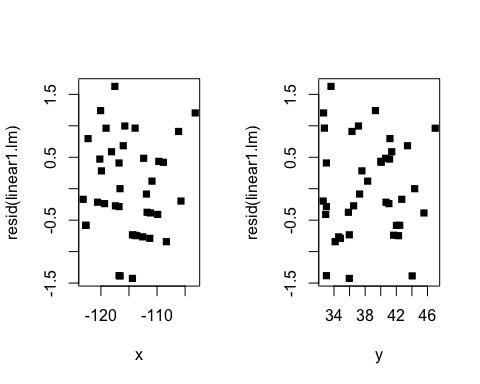
#  
### So there isn't much trend to remove?   
#  
#  
#  
# look for trend in data   
# Remove the trend  
#  
linear1.lm = lm(logp~x,data=Head)  
summary(linear1.lm)

##   
## Call:  
## lm(formula = logp ~ x, data = Head)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.4257 -0.5822 -0.1682 0.5358 1.6267   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 3.49876 2.93958 1.190 0.242  
## x 0.01812 0.02555 0.709 0.483  
##   
## Residual standard error: 0.7839 on 37 degrees of freedom  
## Multiple R-squared: 0.01341, Adjusted R-squared: -0.01325   
## F-statistic: 0.503 on 1 and 37 DF, p-value: 0.4826

AIC(linear1.lm)

## [1] 95.63307

#  
x=data.frame(Head)$x  
y=data.frame(Head)$y  
par(mfrow=c(1,2))  
plot(resid(linear1.lm)~x,pch=15) # clustering of points but no trend for each   
plot(resid(linear1.lm)~y,pch=15)



par(mfrow=c(1,1))  
#  
#  
#  
# Going to stop ploting residuals for each   
#  
#  
linear2.lm = lm(logp~x+y,data=Head)  
summary(linear2.lm)

##   
## Call:  
## lm(formula = logp ~ x + y, data = Head)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.4207 -0.5842 -0.1708 0.5303 1.6397   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 3.509555 2.984731 1.176 0.247  
## x 0.018965 0.029052 0.653 0.518  
## y 0.002237 0.034940 0.064 0.949  
##   
## Residual standard error: 0.7947 on 36 degrees of freedom  
## Multiple R-squared: 0.01353, Adjusted R-squared: -0.04128   
## F-statistic: 0.2468 on 2 and 36 DF, p-value: 0.7826

anova(linear1.lm,linear2.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x  
## Model 2: logp ~ x + y  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 37 22.736   
## 2 36 22.734 1 0.0025891 0.0041 0.9493

AIC(linear2.lm)

## [1] 97.62863

#  
#  
linear3.lm = lm(logp~x\*y,data=Head)  
summary(linear3.lm)

##   
## Call:  
## lm(formula = logp ~ x \* y, data = Head)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.4010 -0.6676 -0.2066 0.5542 1.7108   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 18.602534 30.609153 0.608 0.547  
## x 0.150603 0.267285 0.563 0.577  
## y -0.396148 0.804783 -0.492 0.626  
## x:y -0.003467 0.006998 -0.495 0.623  
##   
## Residual standard error: 0.8031 on 35 degrees of freedom  
## Multiple R-squared: 0.0204, Adjusted R-squared: -0.06357   
## F-statistic: 0.2429 on 3 and 35 DF, p-value: 0.8658

anova(linear2.lm,linear3.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x + y  
## Model 2: logp ~ x \* y  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 36 22.734   
## 2 35 22.576 1 0.15836 0.2455 0.6233

AIC(linear3.lm)

## [1] 99.35601

#  
linear4.lm = lm(logp~x\*y+I(x^2),data=Head)  
summary(linear4.lm)

##   
## Call:  
## lm(formula = logp ~ x \* y + I(x^2), data = Head)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.3648 -0.5411 -0.1478 0.4710 1.5659   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.141e+02 6.223e+01 1.833 0.0756 .  
## x 2.395e+00 1.311e+00 1.827 0.0765 .  
## y 1.307e+00 1.250e+00 1.046 0.3031   
## I(x^2) 1.230e-02 7.041e-03 1.746 0.0898 .  
## x:y 1.137e-02 1.088e-02 1.045 0.3035   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7806 on 34 degrees of freedom  
## Multiple R-squared: 0.101, Adjusted R-squared: -0.004715   
## F-statistic: 0.9554 on 4 and 34 DF, p-value: 0.4444

anova(linear3.lm,linear4.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x \* y  
## Model 2: logp ~ x \* y + I(x^2)  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 35 22.576   
## 2 34 20.717 1 1.8586 3.0502 0.08976 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

AIC(linear4.lm)

## [1] 98.00538

#  
linear5.lm = lm(logp~x\*y+I(y^2),data=Head)  
summary(linear5.lm)

##   
## Call:  
## lm(formula = logp ~ x \* y + I(y^2), data = Head)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.3989 -0.6688 -0.2027 0.5524 1.7136   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 18.5047351 31.4149033 0.589 0.560  
## x 0.1521113 0.2808472 0.542 0.592  
## y -0.3863363 0.9447213 -0.409 0.685  
## I(y^2) -0.0001818 0.0088055 -0.021 0.984  
## x:y -0.0035048 0.0073275 -0.478 0.635  
##   
## Residual standard error: 0.8148 on 34 degrees of freedom  
## Multiple R-squared: 0.02041, Adjusted R-squared: -0.09484   
## F-statistic: 0.1771 on 4 and 34 DF, p-value: 0.9487

anova(linear3.lm,linear5.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x \* y  
## Model 2: logp ~ x \* y + I(y^2)  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 35 22.576   
## 2 34 22.575 1 0.00028312 4e-04 0.9836

AIC(linear5.lm)

## [1] 101.3555

### AIC of lm 3 was lowest so will use that in model combining with other predictors

combine original model and coordinates

#switching to dataset full bc has coords and predictors   
  
linear3.lm = lm(logp~x\*y,data=Head)  
summary(linear3.lm)

##   
## Call:  
## lm(formula = logp ~ x \* y, data = Head)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.4010 -0.6676 -0.2066 0.5542 1.7108   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 18.602534 30.609153 0.608 0.547  
## x 0.150603 0.267285 0.563 0.577  
## y -0.396148 0.804783 -0.492 0.626  
## x:y -0.003467 0.006998 -0.495 0.623  
##   
## Residual standard error: 0.8031 on 35 degrees of freedom  
## Multiple R-squared: 0.0204, Adjusted R-squared: -0.06357   
## F-statistic: 0.2429 on 3 and 35 DF, p-value: 0.8658

anova(linear2.lm,linear3.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x + y  
## Model 2: logp ~ x \* y  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 36 22.734   
## 2 35 22.576 1 0.15836 0.2455 0.6233

AIC(linear3.lm)

## [1] 99.35601

lm2 <- lm(log(total) ~ taxa + percentpublic + area\_x + nsumemploy + total\_x\_x, data=RegData)  
summary(lm2)

##   
## Call:  
## lm(formula = log(total) ~ taxa + percentpublic + area\_x + nsumemploy +   
## total\_x\_x, data = RegData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.5279 -0.4195 -0.0260 0.4178 2.2137   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.906e-01 3.600e-01 2.474 0.0175 \*  
## taxa -3.463e-01 2.647e-01 -1.308 0.1980   
## percentpublic 1.517e-02 4.471e-01 0.034 0.9731   
## area\_x 4.853e-12 3.607e-12 1.345 0.1857   
## nsumemploy 6.771e-07 3.084e-06 0.220 0.8273   
## total\_x\_x 2.543e-01 9.482e-02 2.682 0.0104 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8096 on 42 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.24, Adjusted R-squared: 0.1495   
## F-statistic: 2.653 on 5 and 42 DF, p-value: 0.03585

AIC(lm2)

## [1] 123.5299

lmall1 <- lm(log(total) ~ taxa + percentpublic + area\_x + nsumemploy + total\_x\_x + x, data = full)  
summary(lmall1)

##   
## Call:  
## lm(formula = log(total) ~ taxa + percentpublic + area\_x + nsumemploy +   
## total\_x\_x + x, data = full)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.65884 -0.44054 -0.01578 0.42007 2.13597   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.987e-01 1.389e+00 0.719 0.47638   
## taxa -2.831e-01 2.727e-01 -1.038 0.30562   
## percentpublic 2.875e-03 7.385e-01 0.004 0.99691   
## area\_x 4.811e-12 3.708e-12 1.298 0.20206   
## nsumemploy 1.873e-06 3.589e-06 0.522 0.60468   
## total\_x\_x 2.799e-01 1.013e-01 2.764 0.00867 \*\*  
## x 1.620e-03 1.537e-02 0.105 0.91660   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8132 on 39 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2643, Adjusted R-squared: 0.1511   
## F-statistic: 2.335 on 6 and 39 DF, p-value: 0.05061

anova(linear2.lm,linear3.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x + y  
## Model 2: logp ~ x \* y  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 36 22.734   
## 2 35 22.576 1 0.15836 0.2455 0.6233

AIC(lmall1)

## [1] 119.9303

lmall2 <- lm(log(total) ~ taxa + percentpublic + area\_x + nsumemploy + total\_x\_x + x+y, data = full)  
summary(lmall2)

##   
## Call:  
## lm(formula = log(total) ~ taxa + percentpublic + area\_x + nsumemploy +   
## total\_x\_x + x + y, data = full)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.60256 -0.45931 0.02235 0.41967 1.80763   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.950e-01 1.573e+00 0.124 0.9020   
## taxa -3.025e-01 2.727e-01 -1.109 0.2742   
## percentpublic 2.411e-01 7.693e-01 0.313 0.7557   
## area\_x 5.393e-12 3.739e-12 1.442 0.1574   
## nsumemploy 4.300e-06 4.228e-06 1.017 0.3156   
## total\_x\_x 2.674e-01 1.017e-01 2.628 0.0123 \*  
## x 9.750e-03 1.708e-02 0.571 0.5715   
## y 3.974e-02 3.680e-02 1.080 0.2869   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8115 on 38 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2862, Adjusted R-squared: 0.1547   
## F-statistic: 2.177 on 7 and 38 DF, p-value: 0.05845

anova(linear2.lm,linear3.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x + y  
## Model 2: logp ~ x \* y  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 36 22.734   
## 2 35 22.576 1 0.15836 0.2455 0.6233

AIC(lmall2)

## [1] 120.5395

lmall3 <- lm(log(total) ~ taxa + percentpublic + area\_x + nsumemploy + total\_x\_x + x\*y, data = full)  
summary(lmall3)

##   
## Call:  
## lm(formula = log(total) ~ taxa + percentpublic + area\_x + nsumemploy +   
## total\_x\_x + x \* y, data = full)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.60459 -0.44482 0.00939 0.44044 1.54925   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.090e+01 1.192e+01 -0.915 0.3662   
## taxa -2.744e-01 2.748e-01 -0.999 0.3245   
## percentpublic 1.446e-01 7.773e-01 0.186 0.8535   
## area\_x 5.567e-12 3.749e-12 1.485 0.1461   
## nsumemploy 2.553e-06 4.625e-06 0.552 0.5843   
## total\_x\_x 2.494e-01 1.037e-01 2.406 0.0213 \*  
## x -9.542e-02 1.132e-01 -0.843 0.4049   
## y 3.258e-01 3.067e-01 1.062 0.2950   
## x:y 2.676e-03 2.849e-03 0.939 0.3536   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8128 on 37 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.3028, Adjusted R-squared: 0.1521   
## F-statistic: 2.009 on 8 and 37 DF, p-value: 0.07251

anova(linear2.lm,linear3.lm,test="F")

## Analysis of Variance Table  
##   
## Model 1: logp ~ x + y  
## Model 2: logp ~ x \* y  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 36 22.734   
## 2 35 22.576 1 0.15836 0.2455 0.6233

AIC(lmall3)

## [1] 121.4551

#result is a lower AIC and adjusted R squared value

Distance between points - done two ways and made two distance matrices