Irrigated Agriculture

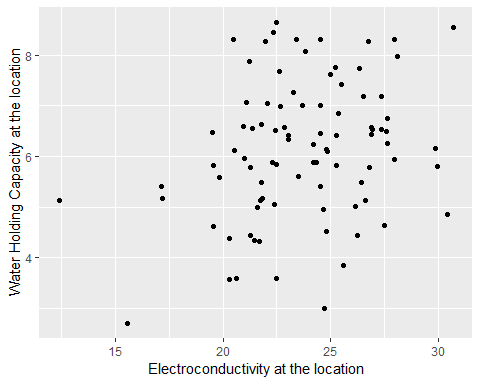
Elise Gertsch and Spencer Wadsworth

April 4, 2019

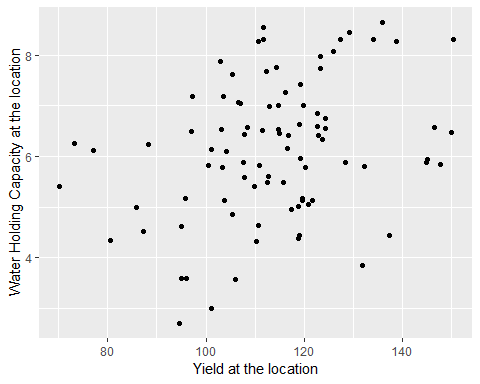
agr <- read.table("https://mheaton.byu.edu/Courses/Stat469/Topics/3%20-%20SpatialCorrelation/1%20-%20PointReference/HWCaseStudy/Data/WaterHoldingCapacity.txt", header = TRUE)  
  
library(ggplot2)  
library(gridExtra)  
library(tidyverse)  
library(car)  
library(geoR)  
source("https://raw.githubusercontent.com/MJHeaton/glstools/master/stdres.gls.R")  
source("https://raw.githubusercontent.com/MJHeaton/glstools/master/predictgls.R")  
library(multcomp)  
library(nlme)  
library(ggmap)  
  
pred <- agr %>% filter(is.na(WHC))   
obs <- agr %>% filter(!is.na(WHC))

1. Create exploratory plots of the data by looking at the relationship between WHC (the response variable) and Yield and EC. Comment on any general relationships you see from the data.

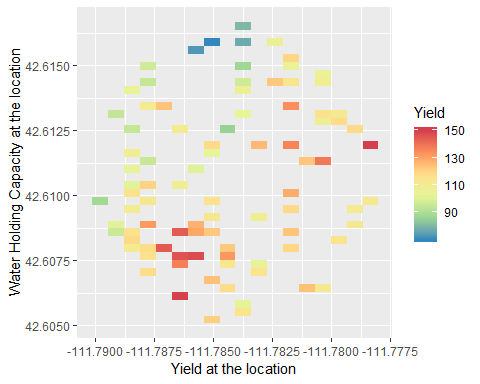
ggplot(data = obs, aes(x = EC, y = WHC)) + geom\_point() + xlab("Electroconductivity at the location") + ylab("Water Holding Capacity at the location")



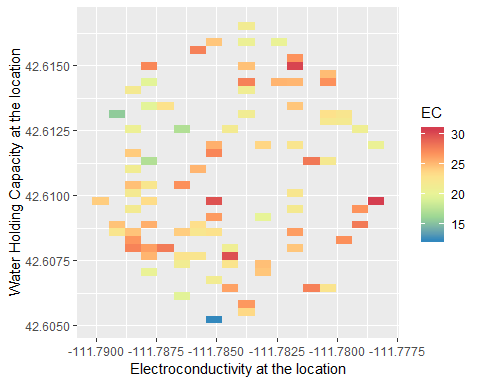
#The data appears to be linear with WHC increasing as EC increases  
  
ggplot(data = obs, aes(x = Yield, y = WHC)) + geom\_point() + xlab("Yield at the location") + ylab("Water Holding Capacity at the location")



#The data appears to be linear with WHC increasing as Yield increases  
  
# These plots are just for our benefit and curiosity  
ggplot(data = obs, aes(x = Lon, y = Lat, fill = Yield)) + geom\_raster() + scale\_fill\_distiller(palette = "Spectral", na.value = NA) + xlab("Yield at the location") + ylab("Water Holding Capacity at the location")



ggplot(data = obs, aes(x = Lon, y = Lat, fill = EC)) + geom\_raster() + scale\_fill\_distiller(palette = "Spectral", na.value = NA) + xlab("Electroconductivity at the location") + ylab("Water Holding Capacity at the location")

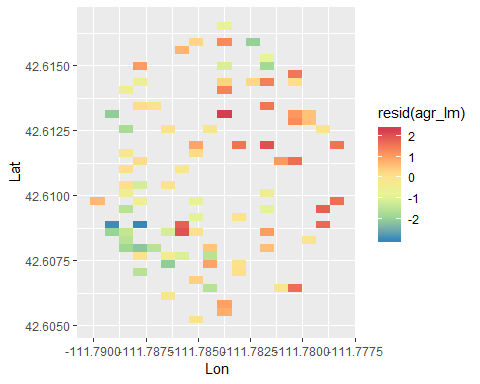


1. Fit an independent MLR model with a linear effect between Yield, EC and the response variable WHC. Explore the residuals to see if there is evidence of spatial correlation by mapping the residuals and plotting the variogram of the residuals.

agr\_lm <- lm(WHC ~ EC + Yield, data = obs)  
summary(agr\_lm)

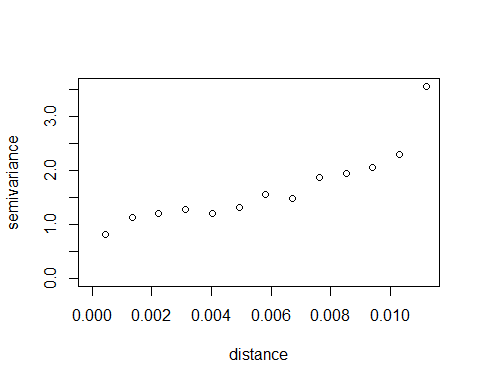
Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.524577 1.337510 0.392 0.69588   
## EC 0.111584 0.040941 2.725 0.00778 \*\*  
## Yield 0.026072 0.008103 3.218 0.00182 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.247 on 86 degrees of freedom  
## Multiple R-squared: 0.173, Adjusted R-squared: 0.1538   
## F-statistic: 8.997 on 2 and 86 DF, p-value: 0.0002833

ggplot(data = obs, aes(x = Lon, y = Lat, fill = resid(agr\_lm))) + geom\_raster() + scale\_fill\_distiller(palette = "Spectral", na.value = NA)



#certain sections of the map have related colors. This suggests spatial correlation  
  
vario <- variog(coords=obs[,1:2], data=resid(agr\_lm))

plot(vario)



#variogram appears to be increasing, this suggesting correlation in the residuals

1. To determine an appropriate correlation structure to use, fit a spatial model using exponential, spherical and Gaussian correlation functions with a nugget effect (don’t forget to filter out the missing observations). Compare the model fits using AIC and use the best fit model for the remainder of the analysis.

exp.mod <- gls(model = WHC ~ Yield + EC, data = obs, correlation = corExp(form =~Lon + Lat, nugget = TRUE), method = "ML")  
sph.mod <- gls(model = WHC ~ EC + Yield, data = obs, correlation = corSpher(form =~Lon + Lat, nugget = TRUE), method = "ML")  
gauss.mod <- gls(model = WHC ~ EC + Yield, data = obs, correlation = corGaus(form =~Lon + Lat, nugget = TRUE), method = "ML")  
  
AIC(exp.mod)

## [1] 272.3653

AIC(sph.mod)

## [1] 272.9623

AIC(gauss.mod)

## [1] 273.4355

#the exponential model yields the smallest AIC value. We will use that for the rest of the analysis  
  
#just to see the constrained variables  
coef(exp.mod$modelStruct$corStruct, unconstrained = FALSE)

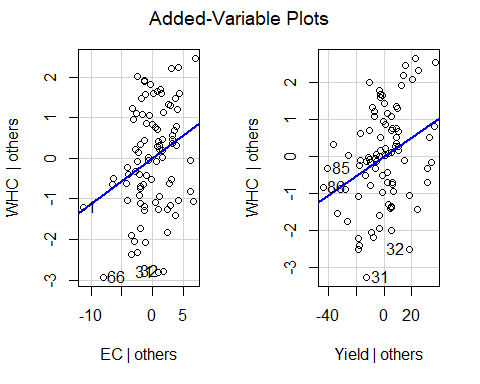
## range nugget   
## 0.00223684 0.36755325

1. Write out your model for analyzing the agriculture data in terms of parameters. Explain and interpret any parameters associated with the model.

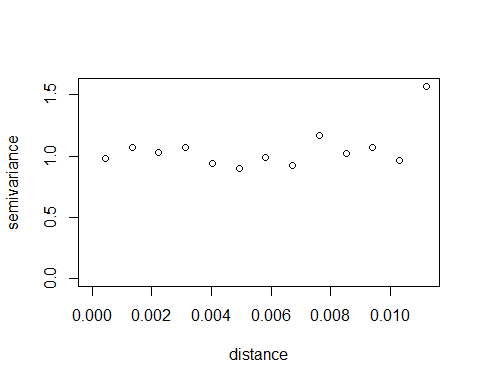
# y ~ N(X\*beta, sigma^2\*R)  
# y is the response variable, in this case WHC  
# X is the matrix with the various parameters plus an intercept. The parameters include EC, Yield, and the interaction between EC and Yield  
# beta is the vector of coefficients for each of the variables.  
# sigma^2\*R is a covariance matrix.   
#R is the correlation matrix that has the correlation values between different the various coordinates. In our case the correlation follows the exponential model so rho(si, sj) = exp{-||si - sj||/ phi}  
# we used a variance nugget to capture same-location variability. if || si - sj|| = 0, then cor(si, sj) = omega, else cor(si, sj) = (1 - omega)\*rho(si, sj), where omega is between 0 and 1

1. Fit your spatial MLR model and validate any assumptions you made to fit the model.

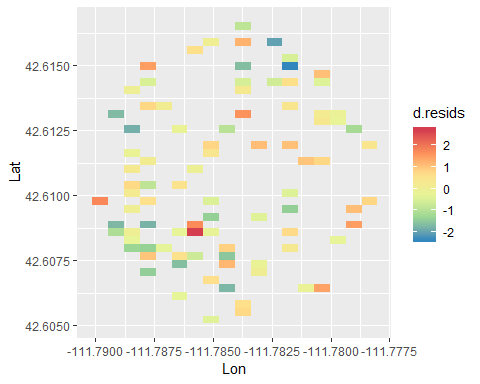
# 1. linearity  
avPlots(lm(WHC ~ EC + Yield, data = obs))



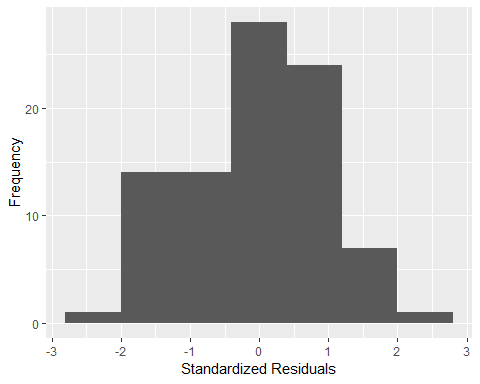
# the AV plots show that our data is linear  
  
# 2. independence  
d.resids <- stdres.gls(exp.mod)  
coords <- as.matrix(obs[,1:2], ncol = 2)  
plot(variog(coords = coords, data = d.resids))



# the variogram of decorrelated residuals is a fairly straight line, so we can say that it shows independence  
  
ggplot(data = obs, aes(x = Lon, y = Lat, fill = d.resids)) + geom\_raster() + scale\_fill\_distiller(palette = "Spectral", na.value = NA)

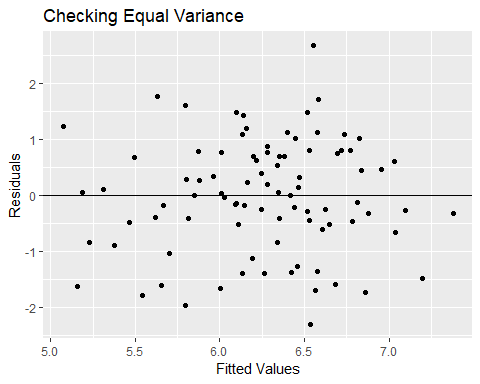


#appears at least slightly more random than the last one  
  
# 3. normality of residuals  
ggplot(data = obs, aes(x = d.resids)) + geom\_histogram(binwidth = .8) + xlab("Standardized Residuals") + ylab("Frequency")



ks.test(d.resids, 'pnorm')  
## D = 0.070803, p-value = 0.7366  
## alternative hypothesis: two-sided

# the histogram is slightly skewed left  
#the high p-value on the ks test helps us feel more comfortable about the normality  
  
# 4. equal variance  
ggplot(data = obs, aes(x = fitted(exp.mod), y = d.resids)) + geom\_point() + geom\_abline(intercept = 0, slope = 0) + labs(x = "Fitted Values", y = "Residuals") + ggtitle("Checking Equal Variance")



# we can say that the condition of equal variance is met because the points follow a random distribution about 0

1. Carry out a hypothesis test that locations with higher yield had higher WHC (which would make sense because more water would be available for the plant to use). Include a confidence interval for the effect of Yield on WHC and interpret this interval.

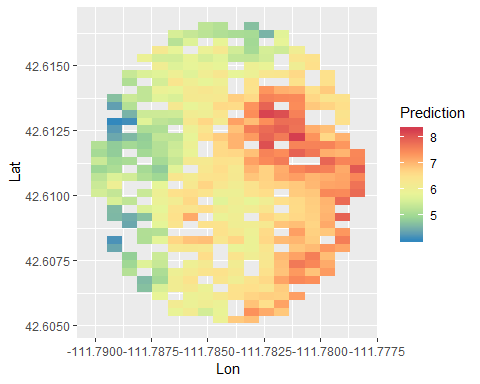
summary(exp.mod)  
## Coefficients:  
## Value Std.Error t-value p-value  
## (Intercept) 1.5663358 1.3950884 1.122750 0.2647  
## Yield 0.0257820 0.0094601 2.725329 0.0078  
## EC 0.0739945 0.0362603 2.040648 0.0444  
##   
## Correlation:   
## (Intr) Yield   
## Yield -0.745   
## EC -0.605 -0.003  
##   
## Standardized residuals:  
## Min Q1 Med Q3 Max   
## -2.46170693 -0.76133851 -0.03222967 0.72664078 1.69219614   
##   
## Residual standard error: 1.226917   
## Degrees of freedom: 89 total; 86 residual

#the p-value associated with an increasing WHC with yield is 0.0078. this is statistically significant and suggests that locations with higher yield have higher WHC  
  
intervals(exp.mod)$coef

## lower est. upper  
## (Intercept) -1.207008138 1.56633579 4.33967972  
## Yield 0.006975860 0.02578200 0.04458814  
## EC 0.001911404 0.07399449 0.14607758  
#the confidence interval for the effect of Yield is (0.007, 0.045), which does not contain 0, so we conclude that a one-unit increase in Yield increases WHC by 0.026

1. Predict WHC at all the locations where WHC is missing. Provide a plot of your predictions.

preds <- predictgls(exp.mod, newdframe = pred)  
  
ggplot(data = preds, aes(x = Lon, y = Lat, fill = Prediction)) + geom\_raster() + scale\_fill\_distiller(palette = "Spectral", na.value = NA)



1. (Extra Credit) Add a google map of the satellite terrain to your prediction map in #7.

bb <- make\_bbox(lon = Lon, lat = Lat, data = preds)  
mymap <- get\_map(location = bb, zoom = 15, maptype = "satellite")  
ggmap(mymap) + geom\_raster(data = preds, aes(x = Lon, y = Lat, fill = Prediction), alpha = 0.8) + scale\_fill\_distiller(palette = "Spectral", na.value = NA) + coord\_cartesian()

