Irrigation HW5

Naomi Zubeldia and Drew Millane

2023-04-04

```
library(tidyverse)
library(gstat)
library(multcomp)
library(nlme)
```

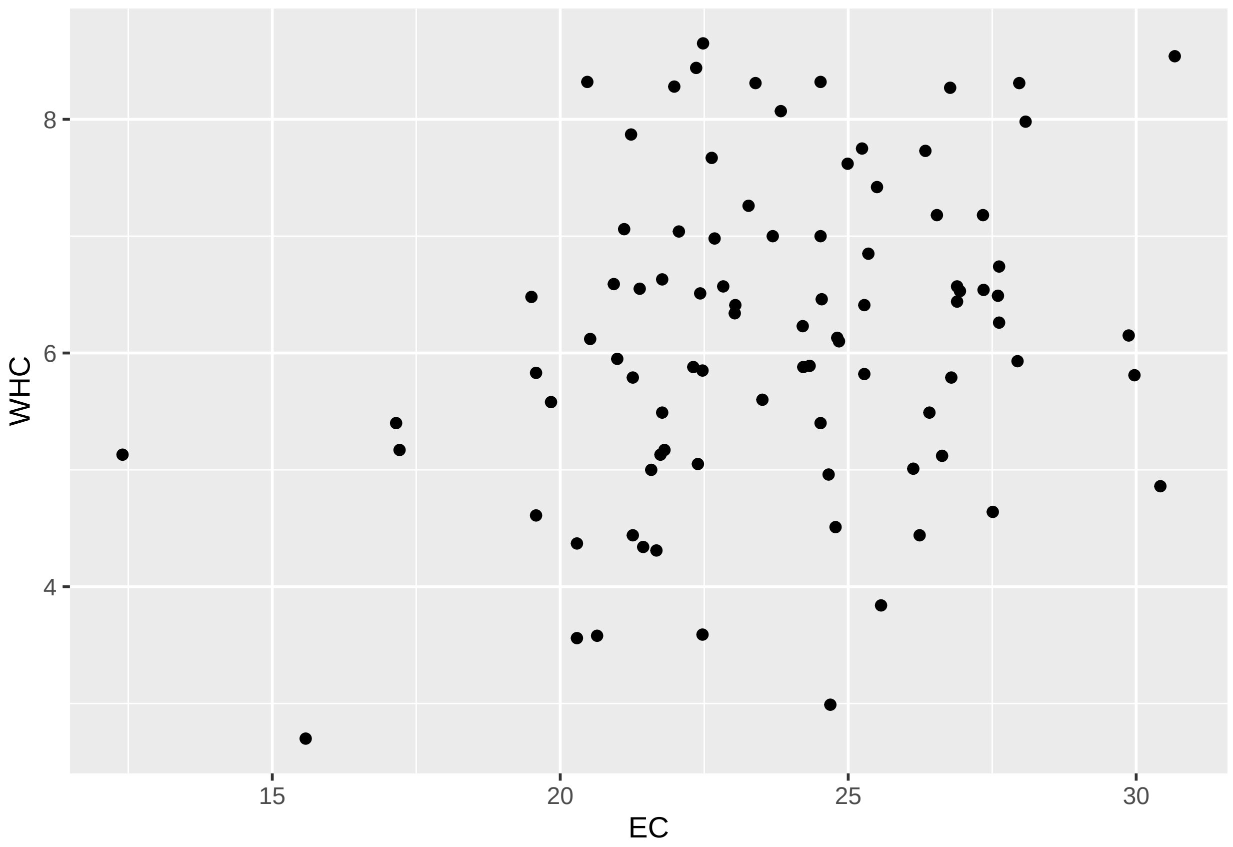
```
water<- read.table('WaterHoldingCapacity.txt',header=T)
df<- na.omit(water)
```

1. Exploratory Data Analysis

For our exploratory data analysis we created two scatter plot of EC and Yield against WHC. Our first plot indicates that there is a positive relationship between EC and WHC. We calculated a correlation of 0.27 so it is a weak positive relationship. Our second plot shows that Yield as well as has a positive relationship with WHC. So, as yield increases the volume of water increases as well. It's correlation is 0.32, which is still weak.

```
ggplot(data=water,mapping=aes(x=EC, y=WHC)) + geom_point()
```

```
## Warning: Removed 439 rows containing missing values (geom_point).
```

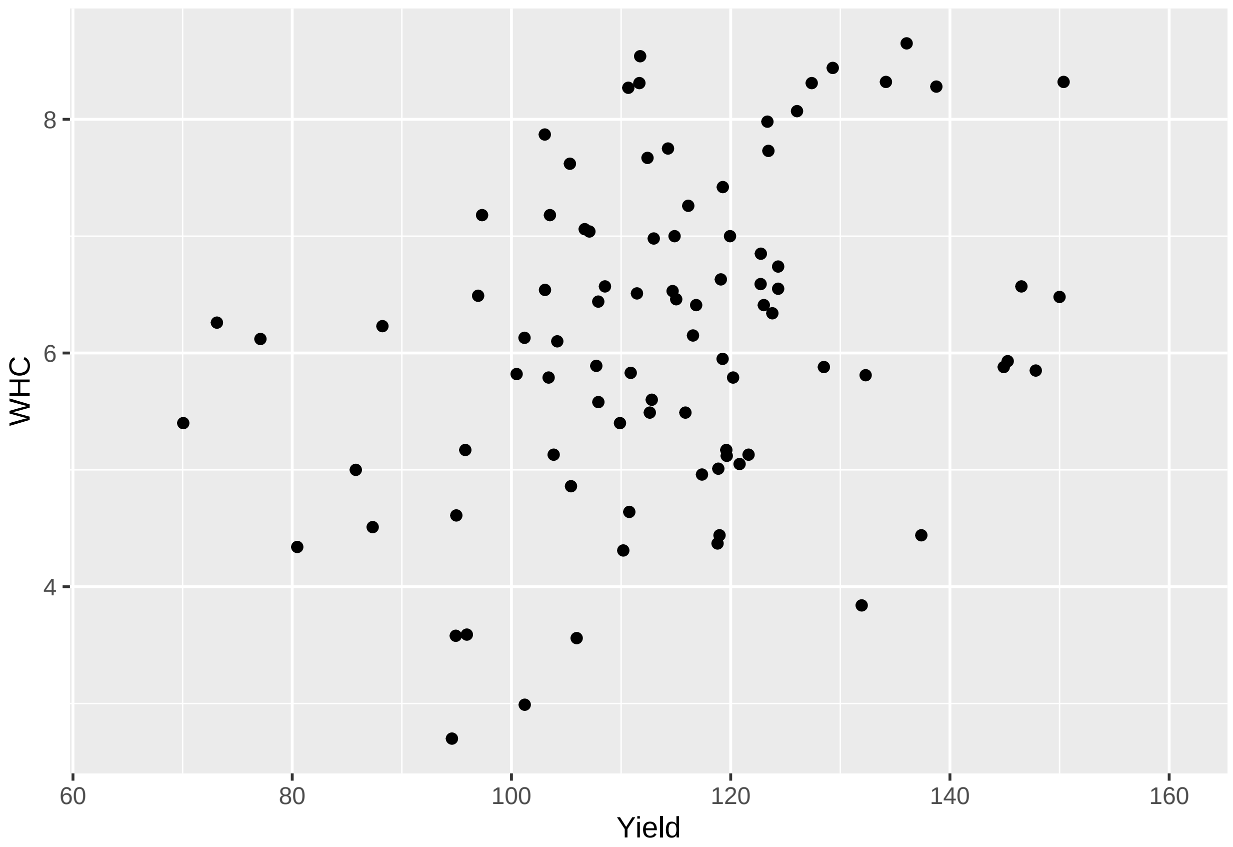


```
cor(df$EC,df$WHC)
```

```
## [1] 0.2710414
```

```
ggplot(data=water,mapping=aes(x=Yield, y=WHC)) + geom_point()
```

```
## Warning: Removed 439 rows containing missing values (geom_point).
```

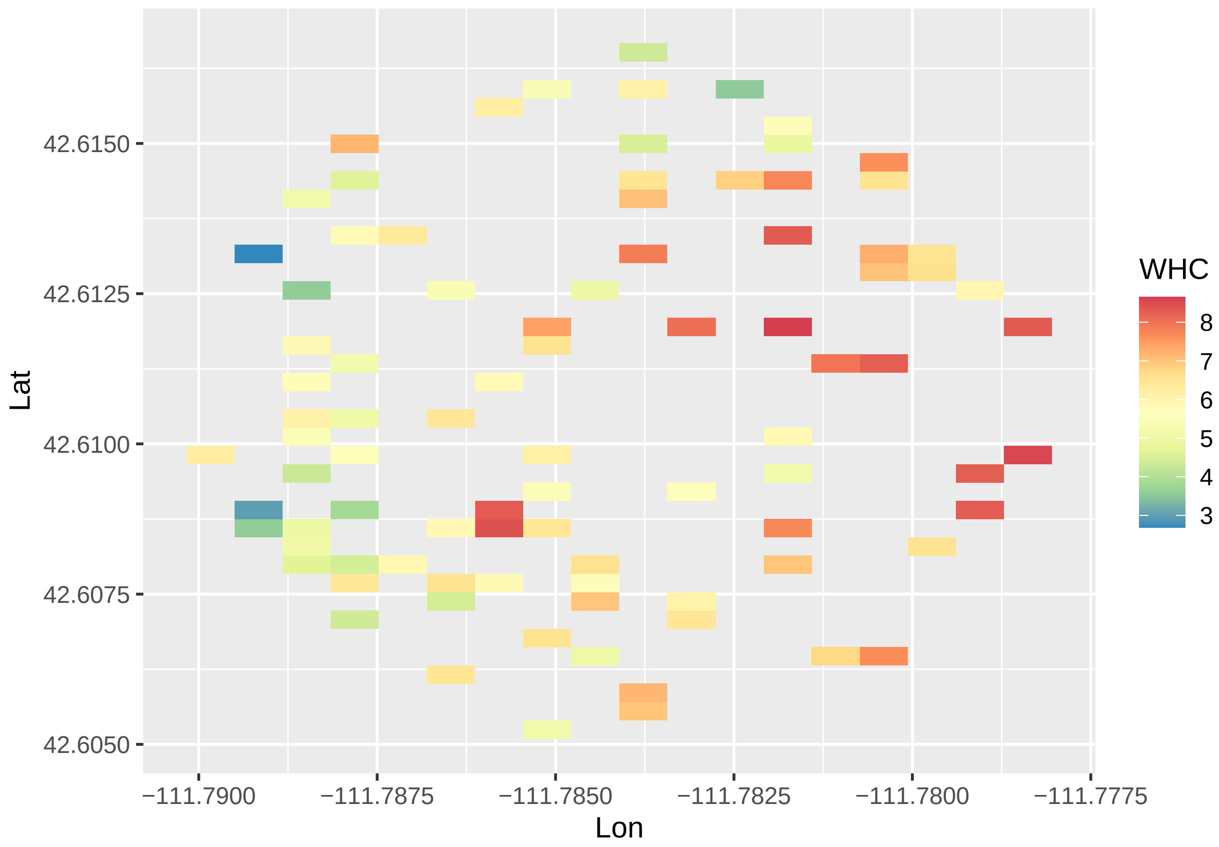


```
cor(df$Yield,df$WHC)
```

```
## [1] 0.3187329
```

We also created a plot that maps out the WHC values with the corresponding coordinates. As you can see, we have many missing values which we will fill in with our spatial model later in the analysis.

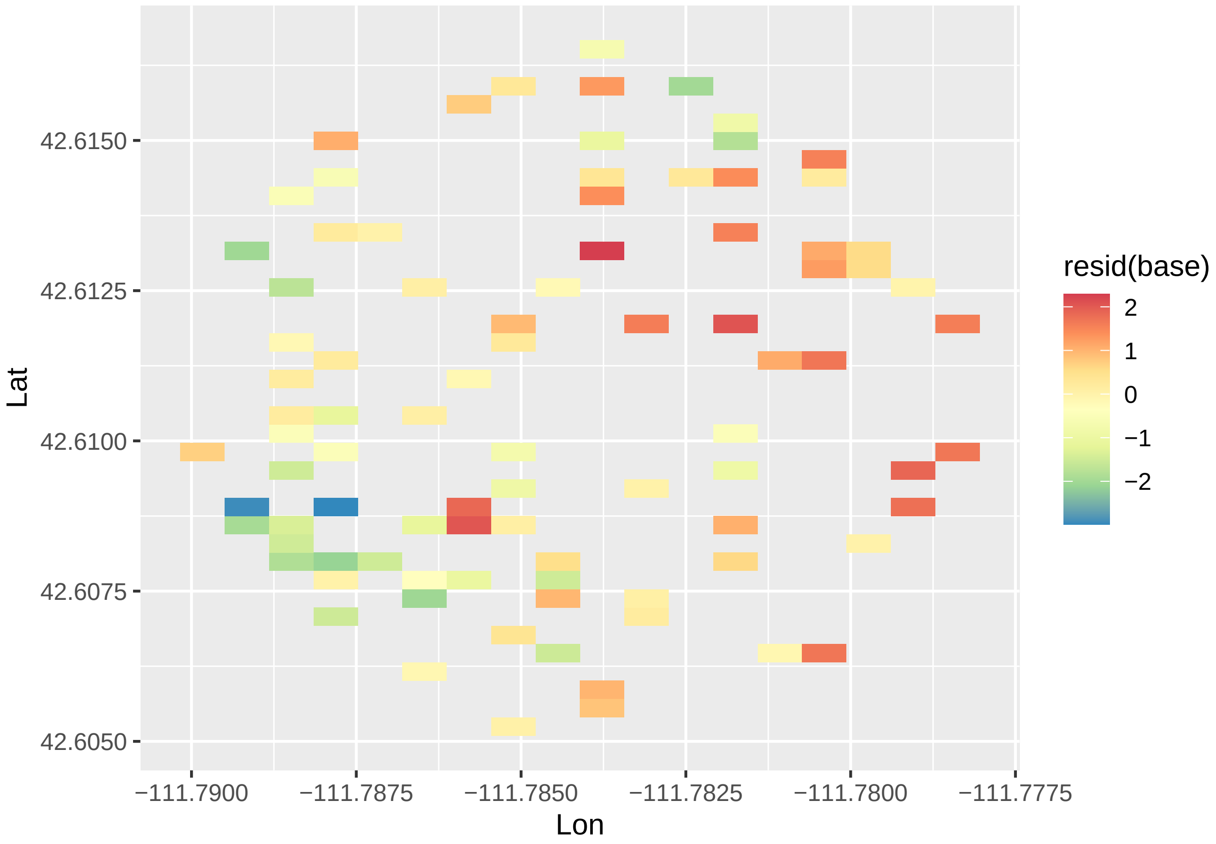
```
ggplot(data=water ,mapping=aes(x=Lon, y=Lat, fill=WHC)) + geom_tile() + scale_fill_distiller(palette="Sp
```



2. Independent Linear Model

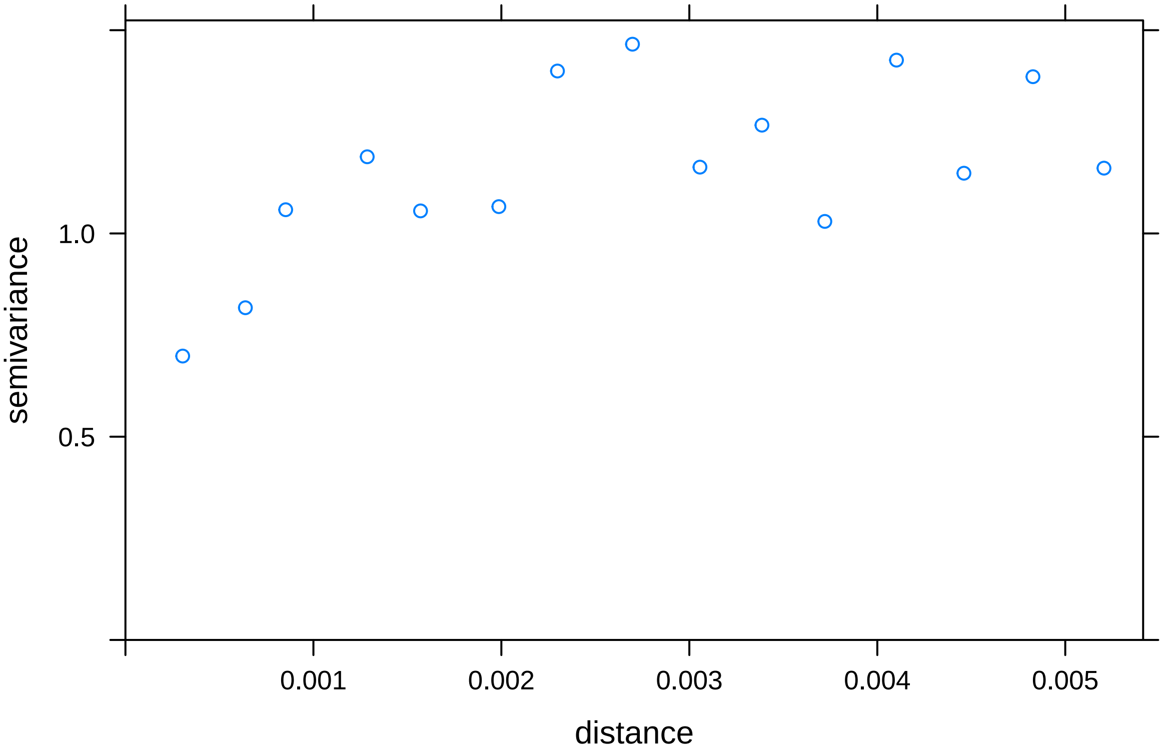
We created a linear model with no spatial correlation incorporated into the model. After calculating the residuals we plotted the residuals. It appears that there is still come spatial correlation left over after using the linear model.

```
#creating basic lm
base<- lm(WHC~Yield+EC,data=water)

#plot of map of the residuals
new=cbind(df,resid(base))
ggplot(data=new ,mapping=aes(x=Lon, y=Lat, fill=resid(base))) + geom_tile() + scale_fill_distiller(pale
```



We created a variogram of the residuals to see if there is evidence of spatial correlation. Because the line is variable and not straight near the top we can assume that there is evidence of spatial correlation.

```
#create variogram of residuals
myVariogram <- variogram(object= WHC~Yield+EC, locations=~Lon+Lat, data=df)
plot(myVariogram)
```



# 3. Determining the Correlation Structure

We created three correlation structures and calculated their AIC scores to see which would be the most appropriate to use. Overall, the exponential correlation structure with an AIC 272.36 is the best model. Therefore, we will use this model for the rest of our analysis.

```
#comparing different correlation gls models
ex<- gls(model=WHC~Yield+EC, data=df, correlation=corExp(form=~Lon+Lat, nugget=TRUE), method="ML")

sph<- gls(model=WHC~Yield+EC, data=df, correlation=corSpher(form=~Lon+Lat, nugget=TRUE), method="ML")

gs<-gls(model=WHC~Yield+EC, data=df, correlation=corGaus(form=~Lon+Lat, nugget=TRUE), method="ML")

#using AIC to find best model
AIC(ex)
```

```
## [1] 272.3653
```

```
AIC(sph)
```

```
## [1] 272.9623
```

```
AIC(gs)
```

```
## [1] 273.4355
```

4. Spatial MLR Model

The spatial model with exponetial correlation is explained below:

$$\boldsymbol{y} \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 \boldsymbol{R}(\boldsymbol{\phi}, \boldsymbol{\omega}))$$

$\boldsymbol{y}$: A nx1 vector of our response variable. In this case, this is the WHC of specific coordinate. Each observation in the data set has a row in this vector with the corresponding heart rate.

$\boldsymbol{X}$: A model matrix of explanatory variables (covariates). In this case, we first have a column of 1's for the intercept and an additional column for each explanatory variable. We have n rows in this matrix, where each row corresponds to a different observation.

$\boldsymbol{\beta}$: A vector of coefficients for each explanatory variable. In our case, these are the coefficients representing the intercept and the effect of the explanatory factors Yield and EC.

σ^2: This represents the variability of $\boldsymbol{y}$ about the regression line.

$\boldsymbol{R}$: This is a nxn matrix that represents spatial correlation between the locations. On the diagonal are 1's and on the off-diagonal is the exponential correlation between locations.

ϕ: This represents the range parameter, which determines how fast spatial correlation decreases between two locations as the distance increases.

ω: This represents the nugget parameter, which represents same-location variability

5.Assumptions

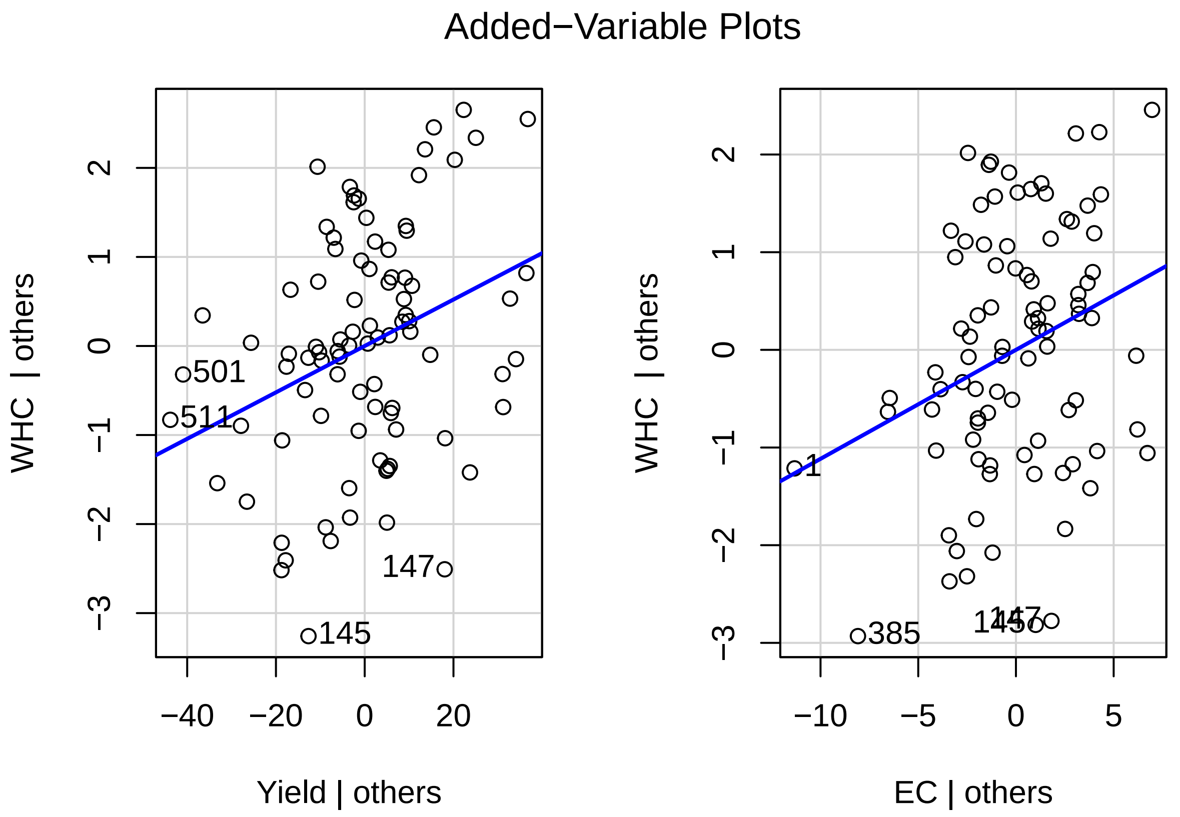
After fitting our model, we need to check the LiNE assumptions to make sure it is valid.

```r
ex <- gls(model=WHC~Yield+EC, data=df, correlation=corExp(form=~Lon+Lat, nugget=TRUE), method="ML")
```

Linearity

We created AV Plots of our base model and it appears that we have linearity for both EC and Yield.

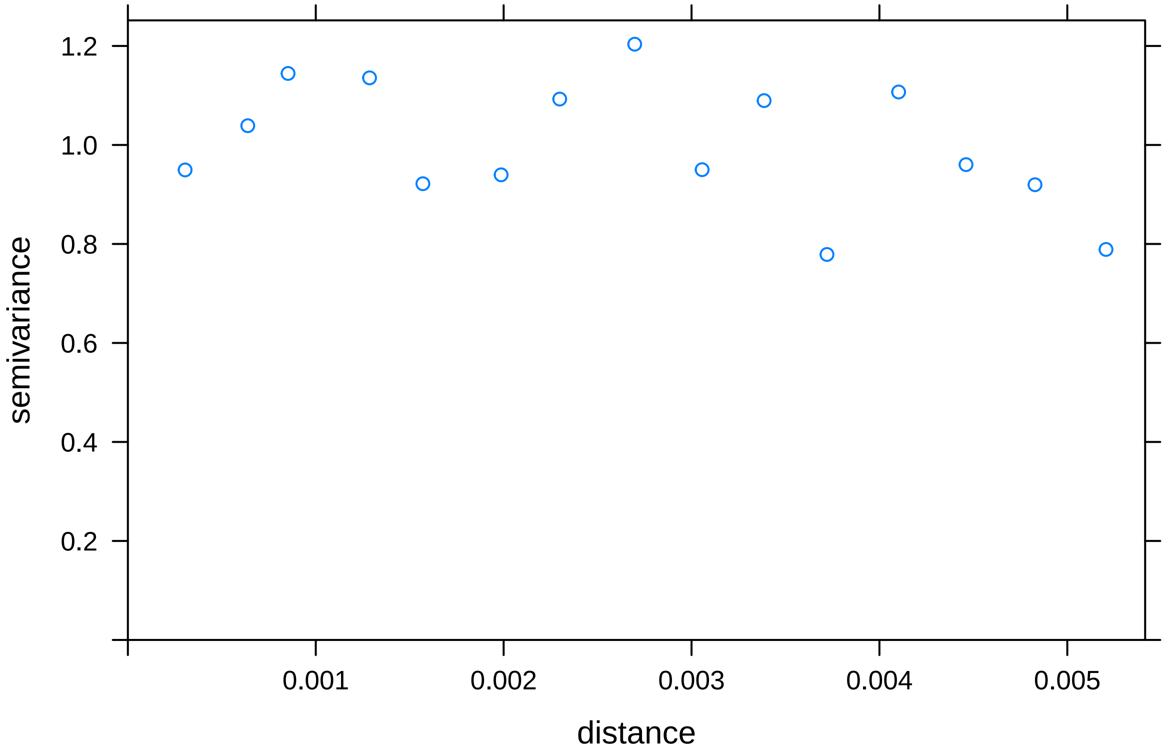
```r
#checking linearity with avplot
library(car)
avPlots(base)
```



Independence

For independence we created another variogram and compared to our previous it is less variable and is closer to one. Therefore, independence is met.

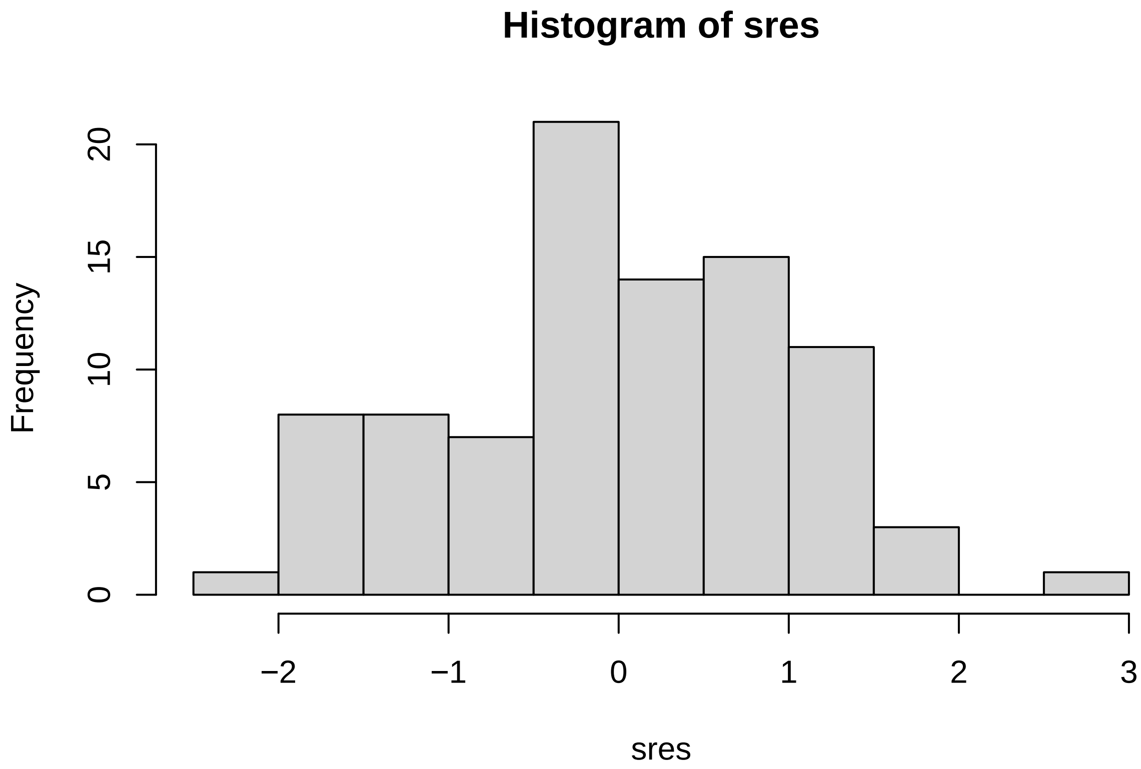
```
#checking independence with variogram of decorrelated resid

source("../STAT469/glstools-master/glstools-master/stdres.gls.R")
#residuals
sres <- stdres.gls(ex)

residDF <- data.frame(Lon=df$Lon, Lat=df$Lat, decorrResid=sres)
residVariogram <- variogram(object=decorrResid~1, locations=~Lon+Lat, data=residDF)
plot(residVariogram)
```



Normality

For normality we made a histogram of the residuals. From a glance, it seems to be normal and we conclude that normality is met.

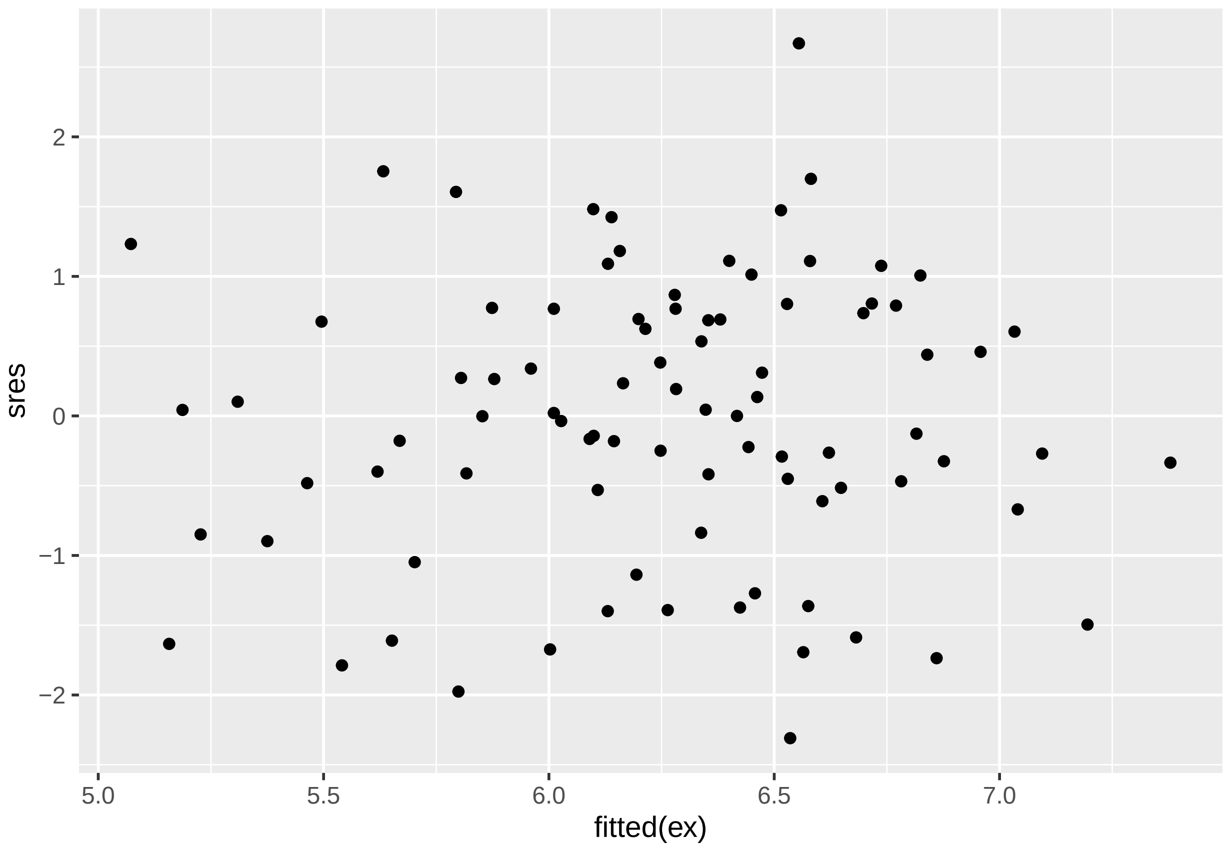
```
#hist of std. residuals
hist(sres)
```



## Equal Variance

For equal variance, we plotted our fitted values against our residuals. There doesn't seem to be any 'funneling', but they stay equally spaced. Therefore, equal variance is met.

```
#checking equal variance of std. resid vs fit
ggplot(mapping=aes(x=fitted(ex),y=sres))+geom_point()
```



6. Cross Validation

We found the our model of choice to be valid and need to determine if it has a good predictive accuracy. We ran 50 cross validations below to check the RPMSE, coverage, and width of prediction.

```
#running cross validations of spatial model

source("../STAT469/glstools-master/glstools-master/predictgls.R")

n.cv <- 50 #Number of CV studies to run
n.test <- 18 #Number of observations in a test set
rpmse <- rep(x=NA, times=n.cv)
bias <- rep(x=NA, times=n.cv)
wid <- rep(x=NA, times=n.cv)
cvg <- rep(x=NA, times=n.cv)

pb <- txtProgressBar(min = 0, max = n.cv, style = 3)
```

```
##   |                                                                               |
```

```
for(cv in 1:n.cv){
  ## Run the CV code
  ## Select test observations
  test.obs <- sample(x=1:nrow(df), size=n.test)

  ## Split into test and training sets
  test.set <- df[test.obs,]
  train.set <- df[-test.obs,]

  ## Fit a lm() using the training data
  train.gls <- ex<- gls(model=WHC~Yield+EC, data=df, correlation=corExp(form=~Lon+Lat, nugget=TRUE), me


  ## Generate predictions for the test set
  my.preds <- predictgls( glsobj=train.gls,newdframe=test.set,level=.99)

  ## Calculate bias
  bias[cv] <- mean(my.preds[,'Prediction']-test.set[['WHC']])

  ## Calculate RPMSE
  rpmse[cv] <- (test.set[['WHC']]-my.preds[,'Prediction'])^2 %>% mean() %>% sqrt()

  ## Calculate Coverage
  cvg[cv] <- ((test.set[['WHC']] > my.preds[,'lwr']) & (test.set[['WHC']] < my.preds[,'upr'])) %>% mean

  ## Calculate Width
  wid[cv] <- (my.preds[,'upr'] - my.preds[,'lwr']) %>% mean()

  ## Update the progress bar
  setTxtProgressBar(pb, cv)
}
```

```
##   |                                                                            |=
```

```
close(pb)
```

```
print(mean(bias))
```

```
## [1] 0.1613308
```

```
print(mean(rpmse))
```

```
## [1] 1.201508
```

```
print(mean(cvg))
```

```
## [1] 1
```

```
print(mean(wid))
```

```
## [1] 6.85338
```

The bias is very small, which is good, and as well as the rpmse. The coverage tells us that the mean of the data is covered in every interval done. The width is nice and large for full coverage. Overall, it appears that our model does well for predicting so we can go forward with our analysis.

7. Hypothesis Testing

We wanted to test if locations with higher yield have higher WHC as well. From the general linear hypothesis test we did below, we reject the null hypothesis due to a small pvalue and find that the locations with higher yield do have higher WHC.

To understand the effect the Yield has on WHC, we conducted a confidence interval. With 95% confidence, we conclude that as Yield increases by 1 WHC increases between 0.0072 to .044 on average.

```
#Hypothesis
a <- matrix(c(0,1,0), nrow = 1)
summary <- glht(ex,a,alternative = 'greater', rhs = 0) %>%
  summary()

summary
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Fit: gls(model = WHC ~ Yield + EC, data = df, correlation = corExp(form = ~Lon +
##     Lat, nugget = TRUE), method = "ML")
##
## Linear Hypotheses:
##        Estimate Std. Error z value  Pr(>z)
## 1 <= 0  0.02578    0.00946   2.725 0.00321 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## (Adjusted p values reported -- single-step method)
```

```
confint(ex)
```

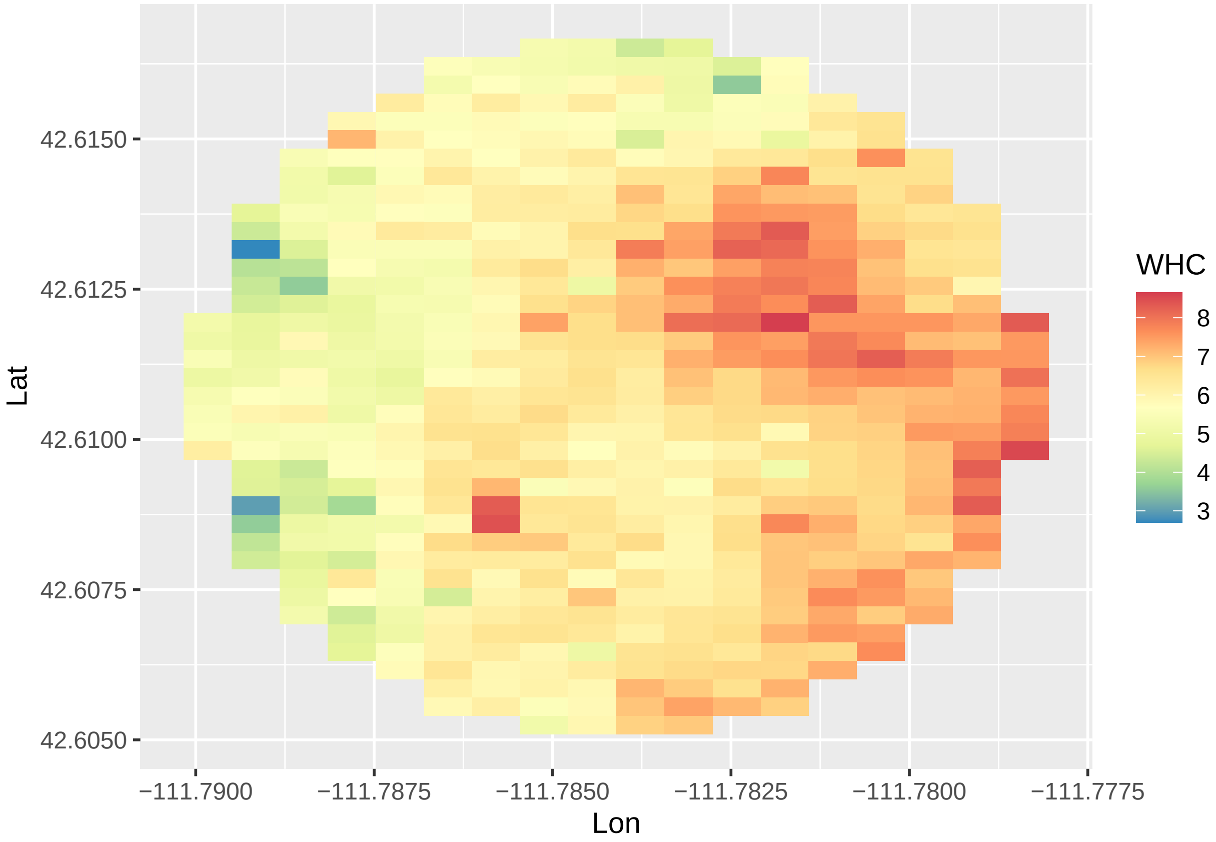
```
##                    2.5 %     97.5 %
## (Intercept) -1.167987207 4.30065879
## Yield        0.007240462 0.04432354
## EC           0.002925613 0.14506337
```

8. Predictions

Now that we have a good model, we predicted the WHC at all the locations it was missing due to cloud cover. We plotted those predictions in the map below, which is now filled unlike the ones earlier in the analysis.

```
#Dataset with NA's
test <- water %>%
  filter(is.na(WHC))

## Predictions
fit.vals <- predictgls(ex,test)

##Fill in NA's
test$WHC <- fit.vals$Prediction
```

```
## Combine datasets
preds <- rbind(test,df)


# Graphing the water content
ggplot(data=preds,mapping=aes(x=Lon, y=Lat, fill=WHC)) + geom_tile() +
  scale_fill_distiller(palette="Spectral",na.value=NA)
```


Lat
Lon
WHC