Legionellosis: County and Population in Ohio

David Hibler

December 3, 2019

# Data

## Obtaining and Processing the Data

pop\_2010 = Ohio county population from [2010 census](https://www.census.gov/quickfacts/OH).  
pop\_2018 = [Estimated Ohio county population](http://worldpopulationreview.com/states/ohio/). Retrieved 2019-01-24, from <http://worldpopulationreview.com/states/ohio/>  
pop\_dens = Population density calculated from 2018 population and county area in sq miles.  
ln\_2018\_pop = The calculated natural log of the 2018 population.  
ln\_pop\_density = The calculated natural log of the 2018 population density.  
pop\_dif = The calculated change in population from 2010 to 2018.  
abs\_pop\_dif = The absolute value of change in population from 2010 to 2018.  
l\_2017 = Reported Legionellosis infections in 2017 gathered from the [Ohio Department of Health Infections Diseases Report](https://odh.ohio.gov/wps/portal/gov/odh/know-our-programs/infectious-diseases/Reports/2017-annual-summary).  
l\_2018 = Reported Legionellosis infections in 2018 that was specially disclosed by the Ohio Department of Health and is not yet available online, so was not used in this project.

Data set calculations (such as population densities and natural logs) were done in excell and then the data set was saved as a csv file to be analyzed in R.

## Analyzing the Data

Data sets were ploted for visual examination, and correlational tests (such as Pearson’s, Kendall’s, and Spearman’s) were run to check for signifigance.

set.seed(37)  
  
setwd("C:/Users/hible/Documents/School/Pubhealth/PUBHLTH 8802 - GISD sem")   
  
my\_data <- read.csv("ohio\_ld\_2018.csv", header=TRUE)  
  
l\_2017\_v <- my\_data$l\_2017  
pop\_2018\_v <- my\_data$pop\_2018  
pop\_density\_v <- my\_data$pop\_density  
ln\_2018\_pop\_v <- my\_data$ln\_2018\_pop  
ln\_pop\_density\_v <- my\_data$ln\_pop\_density  
  
cor\_p\_values <- matrix(nrow=3, ncol=4)  
colnames(cor\_p\_values) <- c("Infections vs Population", "Infections vs Pop\_density", "Infections vs Ln\_Pop", "Infections vs Ln\_Pop\_density")  
rownames(cor\_p\_values) <- c("Pearson", "Kendall", "Spearman")  
  
i\_p\_p <- cor.test(l\_2017\_v,pop\_2018\_v, method = "pearson")  
cor\_p\_values[1,1] <- i\_p\_p$p.value  
i\_p\_k <- cor.test(l\_2017\_v,pop\_2018\_v, method = "kendall")  
cor\_p\_values[2,1] <- i\_p\_k$p.value  
i\_p\_s <- cor.test(l\_2017\_v,pop\_2018\_v, method = "spearman")

## Warning in cor.test.default(l\_2017\_v, pop\_2018\_v, method = "spearman"):  
## Cannot compute exact p-value with ties

cor\_p\_values[3,1] <- i\_p\_s$p.value  
  
i\_p\_p <- cor.test(l\_2017\_v,pop\_density\_v, method = "pearson")  
cor\_p\_values[1,2] <- i\_p\_p$p.value  
i\_p\_k <- cor.test(l\_2017\_v,pop\_density\_v, method = "kendall")  
cor\_p\_values[2,2] <- i\_p\_k$p.value  
i\_p\_s <- cor.test(l\_2017\_v,pop\_density\_v, method = "spearman")

## Warning in cor.test.default(l\_2017\_v, pop\_density\_v, method = "spearman"):  
## Cannot compute exact p-value with ties

cor\_p\_values[3,2] <- i\_p\_s$p.value  
  
i\_p\_p <- cor.test(l\_2017\_v,ln\_2018\_pop\_v, method = "pearson")  
cor\_p\_values[1,3] <- i\_p\_p$p.value  
i\_p\_k <- cor.test(l\_2017\_v,ln\_2018\_pop\_v, method = "kendall")  
cor\_p\_values[2,3] <- i\_p\_k$p.value  
i\_p\_s <- cor.test(l\_2017\_v,ln\_2018\_pop\_v, method = "spearman")

## Warning in cor.test.default(l\_2017\_v, ln\_2018\_pop\_v, method = "spearman"):  
## Cannot compute exact p-value with ties

cor\_p\_values[3,3] <- i\_p\_s$p.value  
  
cor.test(l\_2017\_v,ln\_pop\_density\_v, method = "pearson")

##   
## Pearson's product-moment correlation  
##   
## data: l\_2017\_v and ln\_pop\_density\_v  
## t = 7.8433, df = 86, p-value = 1.096e-11  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.5045721 0.7533242  
## sample estimates:  
## cor   
## 0.6457702

i\_p\_p <- cor.test(l\_2017\_v,ln\_pop\_density\_v, method = "pearson")  
cor\_p\_values[1,4] <- i\_p\_p$p.value  
i\_p\_k <- cor.test(l\_2017\_v,ln\_pop\_density\_v, method = "kendall")  
cor\_p\_values[2,4] <- i\_p\_k$p.value  
i\_p\_s <- cor.test(l\_2017\_v,ln\_pop\_density\_v, method = "spearman")

## Warning in cor.test.default(l\_2017\_v, ln\_pop\_density\_v, method =  
## "spearman"): Cannot compute exact p-value with ties

cor\_p\_values[3,4] <- i\_p\_s$p.value  
  
print(cor\_p\_values)

## Infections vs Population Infections vs Pop\_density  
## Pearson 1.425785e-39 2.109105e-30  
## Kendall 4.891875e-14 1.331731e-13  
## Spearman 1.727944e-15 1.073629e-14  
## Infections vs Ln\_Pop Infections vs Ln\_Pop\_density  
## Pearson 5.435431e-12 1.096061e-11  
## Kendall 4.891875e-14 1.331731e-13  
## Spearman 1.727944e-15 1.073629e-14

Infections were found to be highly significantly correlated with population size, population density,the natural log of the population size, and the natuarl log of the population density.

# Summary stats

# summarize the data  
summary(my\_data)

## Ã¯..county area l\_2018 pop\_2010   
## Adams : 1 Min. :228.2 Min. : 0.00 Min. : 13319   
## Allen : 1 1st Qu.:409.5 1st Qu.: 1.00 1st Qu.: 37312   
## Ashland : 1 Median :439.1 Median : 3.00 Median : 57733   
## Ashtabula: 1 Mean :464.4 Mean : 10.57 Mean : 131368   
## Athens : 1 3rd Qu.:510.5 3rd Qu.: 8.00 3rd Qu.: 124145   
## Auglaize : 1 Max. :702.4 Max. :208.00 Max. :1267513   
## (Other) :82   
## pop\_2018 pop\_density ln\_2018\_pop ln\_pop\_density   
## Min. : 13142 Min. : 31.26 Min. : 9.484 Min. :3.442   
## 1st Qu.: 36946 1st Qu.: 83.94 1st Qu.:10.517 1st Qu.:4.429   
## Median : 57786 Median : 119.11 Median :10.964 Median :4.780   
## Mean : 131929 Mean : 292.84 Mean :11.169 Mean :5.047   
## 3rd Qu.: 123526 3rd Qu.: 256.16 3rd Qu.:11.724 3rd Qu.:5.546   
## Max. :1257401 Max. :2742.48 Max. :14.045 Max. :7.917   
##   
## pop\_dif abs\_pop\_dif l\_2017 inf\_dif   
## Min. :-10112.0 Min. : 2 Min. : 0.000 Min. :-6.000   
## 1st Qu.: -633.5 1st Qu.: 244 1st Qu.: 1.000 1st Qu.: 0.000   
## Median : -256.5 Median : 583 Median : 2.000 Median : 1.000   
## Mean : 561.2 Mean : 1880 Mean : 6.602 Mean : 3.966   
## 3rd Qu.: 97.5 3rd Qu.: 1375 3rd Qu.: 5.000 3rd Qu.: 4.250   
## Max. : 55915.0 Max. :55915 Max. :123.000 Max. :85.000   
##

mean(my\_data$l\_2017)

## [1] 6.602273

mean(my\_data$pop\_2018)

## [1] 131929

mean(my\_data$pop\_density)

## [1] 292.8386

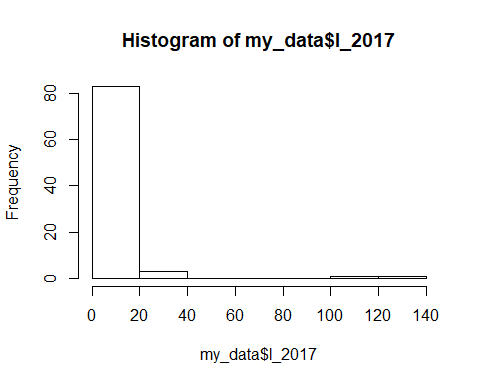
mean(my\_data$ln\_2018\_pop)

## [1] 11.16865

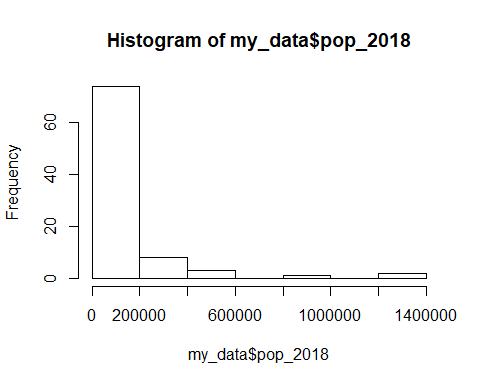
mean(my\_data$ln\_pop\_density)

## [1] 5.046705

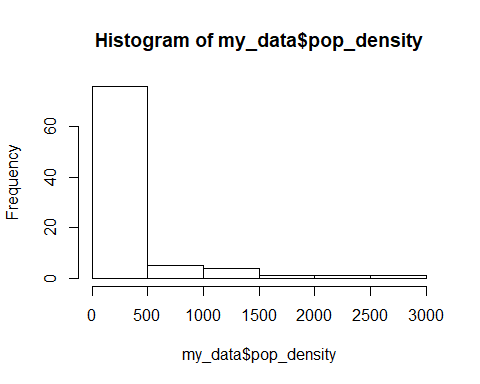
# histograms  
hist(my\_data$l\_2017)



hist(my\_data$pop\_2018)



hist(my\_data$pop\_density)



# variances, these should match what we supplied in the covariance matrix  
var(my\_data$l\_2017)

## [1] 317.7825

var(my\_data$pop\_2018)

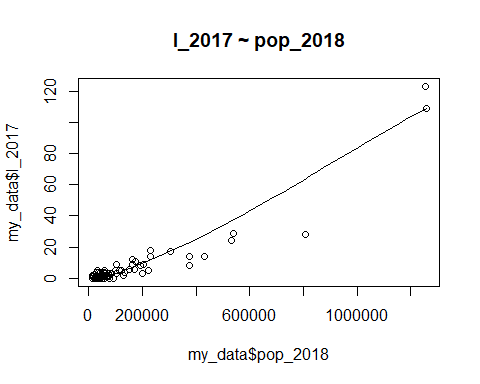
## [1] 46858858822

var(my\_data$pop\_density)

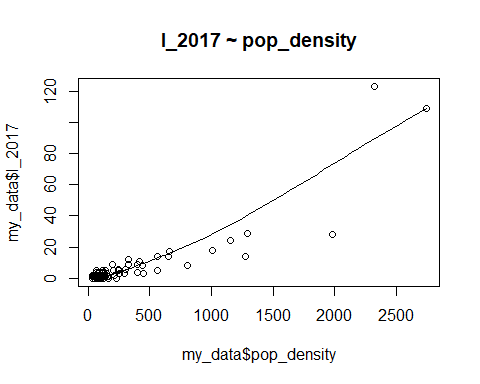
## [1] 222635.3

# Plot

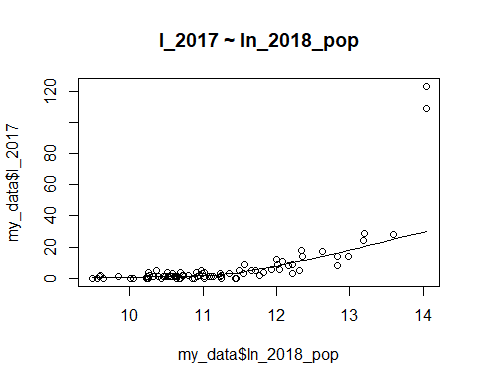
scatter.smooth(x=my\_data$pop\_2018, y=my\_data$l\_2017, main="l\_2017 ~ pop\_2018")



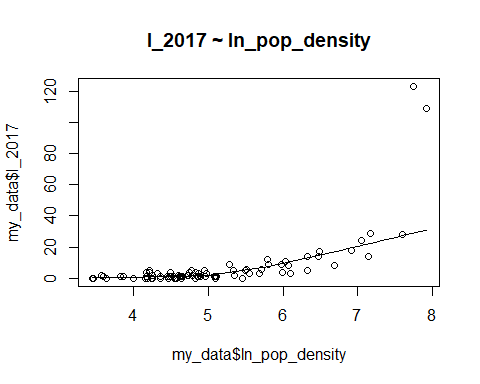
scatter.smooth(x=my\_data$pop\_density, y=my\_data$l\_2017, main="l\_2017 ~ pop\_density")



scatter.smooth(x=my\_data$ln\_2018\_pop, y=my\_data$l\_2017, main="l\_2017 ~ ln\_2018\_pop")



scatter.smooth(x=my\_data$ln\_pop\_density, y=my\_data$l\_2017, main="l\_2017 ~ ln\_pop\_density")

 # Linear regression

linearMod\_pop <- lm(l\_2017 ~ pop\_2018, data=my\_data) # build linear regression model on full data  
print(linearMod\_pop)

##   
## Call:  
## lm(formula = l\_2017 ~ pop\_2018, data = my\_data)  
##   
## Coefficients:  
## (Intercept) pop\_2018   
## -3.520e+00 7.672e-05

linearMod\_popd <- lm(l\_2017 ~ pop\_density, data=my\_data) # build linear regression model on full data  
print(linearMod\_popd)

##   
## Call:  
## lm(formula = l\_2017 ~ pop\_density, data = my\_data)  
##   
## Coefficients:  
## (Intercept) pop\_density   
## -3.19690 0.03346

linearMod\_lnpop <- lm(l\_2017 ~ ln\_2018\_pop, data=my\_data) # build linear regression model on full data  
print(linearMod\_lnpop)

##   
## Call:  
## lm(formula = l\_2017 ~ ln\_2018\_pop, data = my\_data)  
##   
## Coefficients:  
## (Intercept) ln\_2018\_pop   
## -123.59 11.66

linearMod\_lnpopd <- lm(l\_2017 ~ ln\_pop\_density, data=my\_data) # build linear regression model on full data  
print(linearMod\_lnpopd)

##   
## Call:  
## lm(formula = l\_2017 ~ ln\_pop\_density, data = my\_data)  
##   
## Coefficients:  
## (Intercept) ln\_pop\_density   
## -51.54 11.52