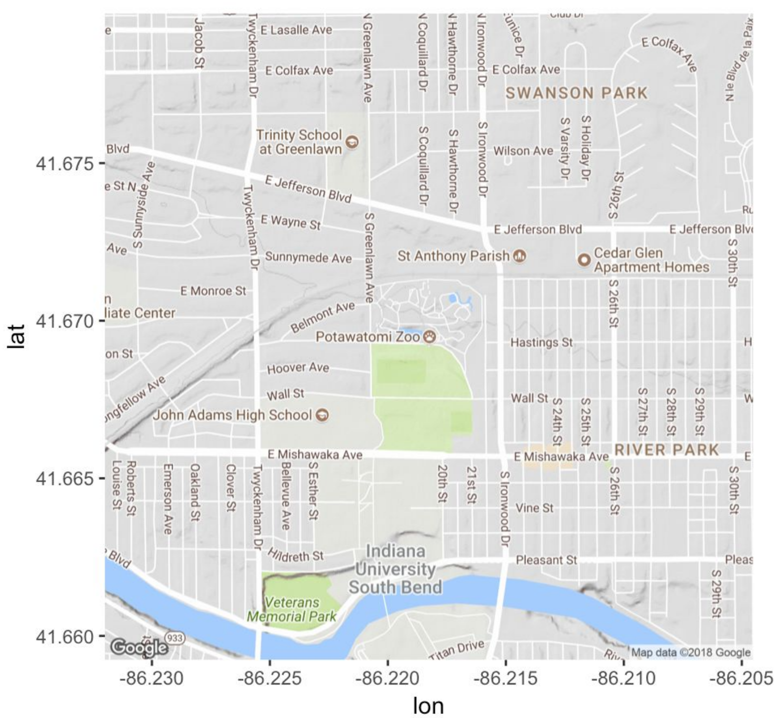
Patrick Tinsley
ACMS 60855 HW #3
Due Date: 3/29

Problem 1

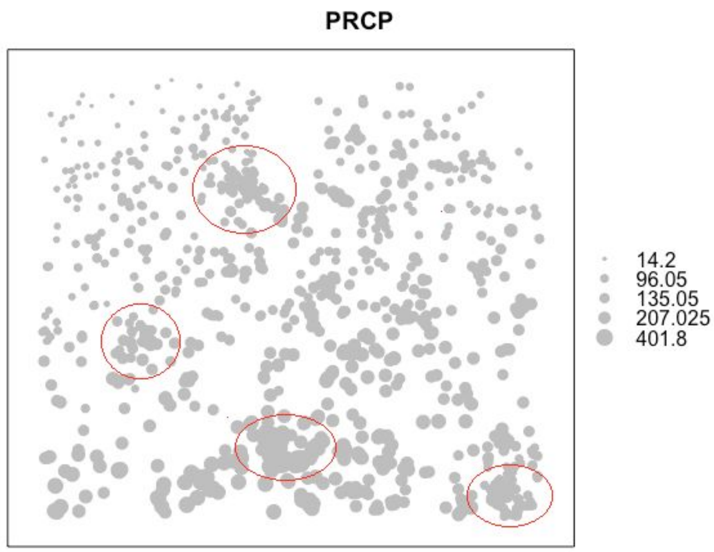
The Potawatomi Zoo is situated at 500 S Greenlawn Ave, South Bend, IN 46615. A map of the zoo's location can be seen below. The distances and travel times from the zoo to O'Hare International Airport are listed under the map. Since ggmap and mapdist rely on Google data, we must ask how Google estimates distance and travel time. The distances are different between mode of transport because the exact path taken differs from mode to mode. The estimated travel times are computed using official and recommended speed limits for a road, historical average speed data, and real-time traffic information.



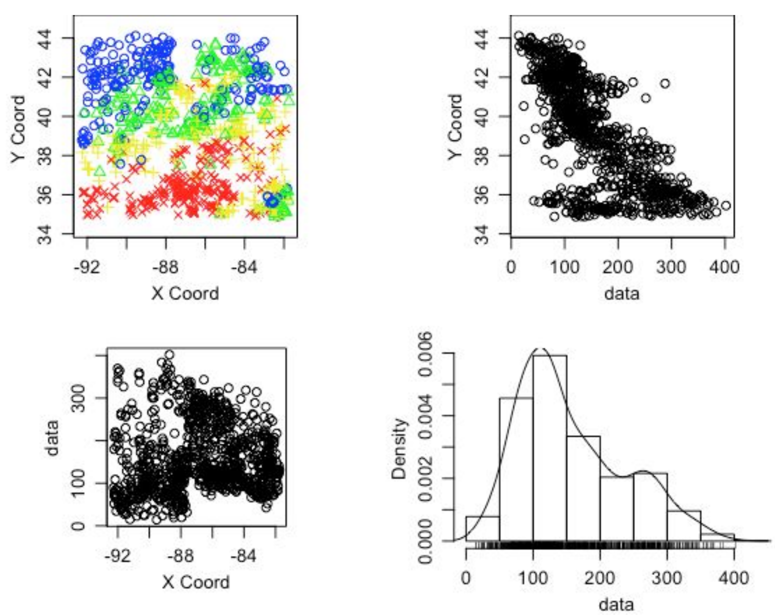

```
                                   from                                   to      m      km
1 500 S Greenlawn Ave, South Bend, IN 46615 10000 W O'Hare Ave, Chicago, IL 60666 194938 194.938
2 500 S Greenlawn Ave, South Bend, IN 46615 10000 W O'Hare Ave, Chicago, IL 60666 195767 195.767
3 500 S Greenlawn Ave, South Bend, IN 46615 10000 W O'Hare Ave, Chicago, IL 60666 170433 170.433
     miles seconds   minutes     hours  method
1 121.1345    7538  125.6333  2.093889 driving
2 121.6496   37243  620.7167 10.345278  biking
3 105.9071  125989 2099.8167 34.996944 walking
```

Problem 2

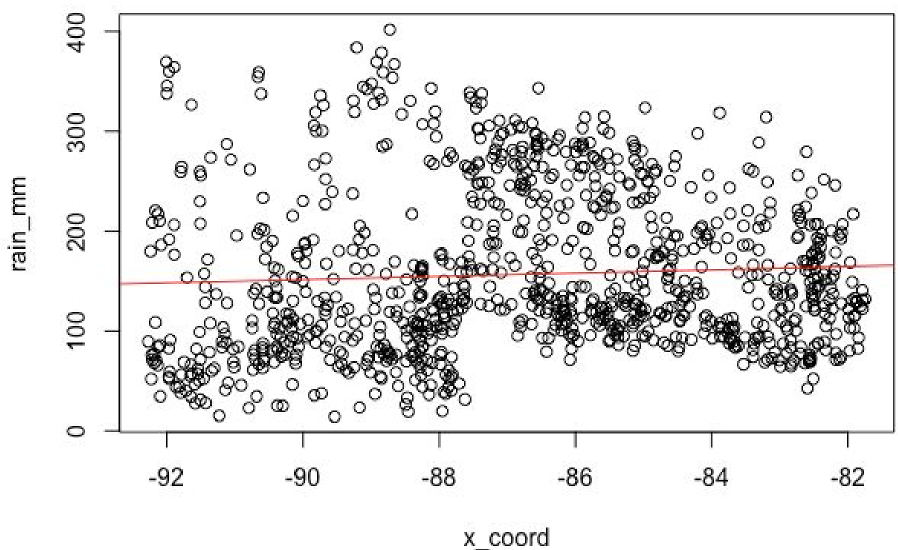
The bubble plot for the midwest region can be seen below. Worthy of noting is the vertical trend in the data; it appears that the lower parts of the region on the y-axis collected more rainfall in millimeters. Additionally, there appears to be some clustering in the data, especially in the red circles seen in the plot. Perhaps more data is available in these parts because they are potentially more popular locales, such cities.

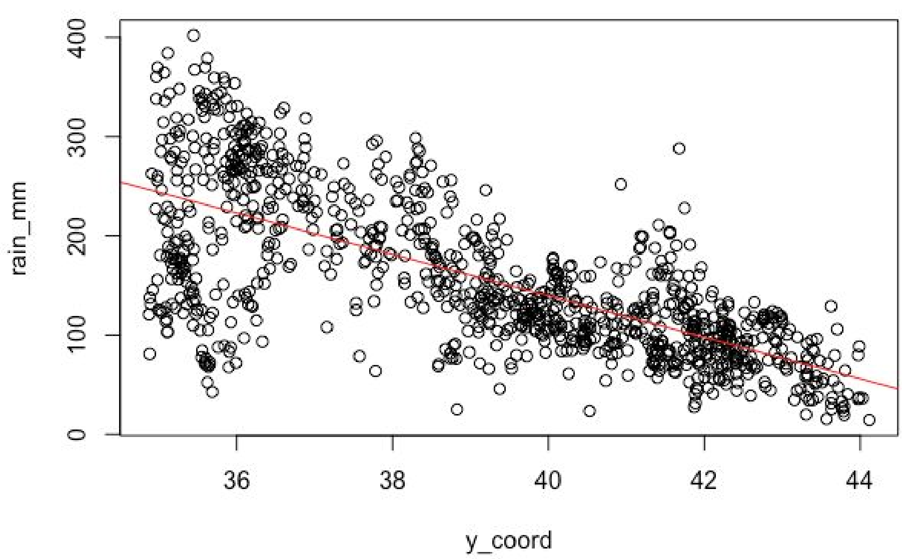


Exploratory Analysis: Below is a set of plots describing the data set. Details are investigated further on the next page.

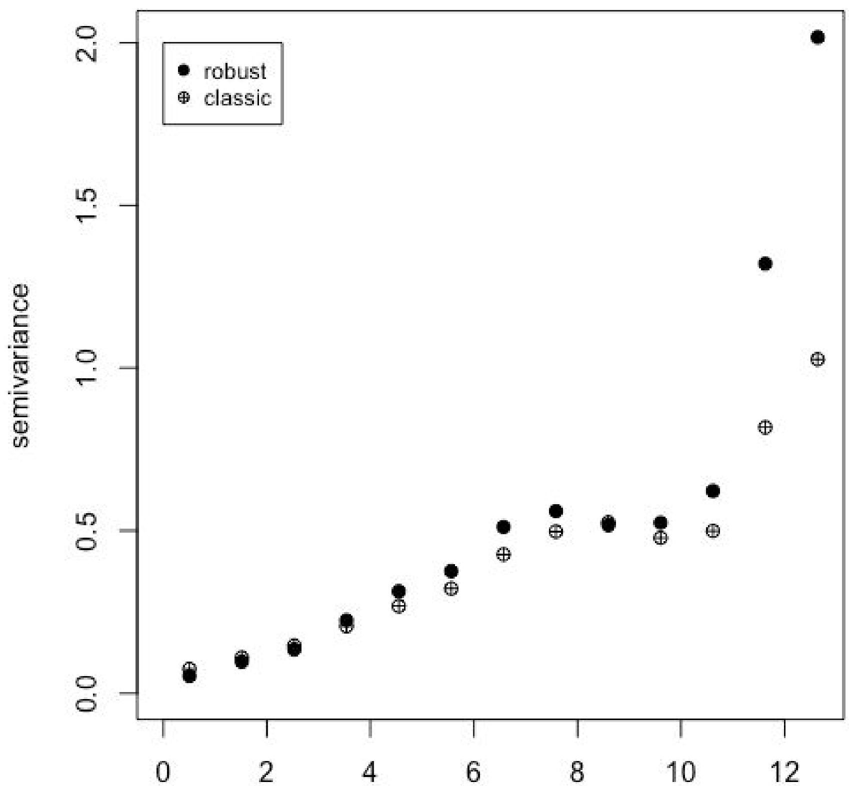


EA (continued): In the second and third graphs (bottom-left and top-right, respectively) we see how the longitude and latitude relate to the rainfall in the midwest region. With longitude, we fit a linear model to estimate the relationship. However, the coefficient is only significant at the .1 level, which is not very strong. Conversely, we fit a similar model to estimate the relationship between latitude and rainfall. In this case, we see a strongly negative correlation between the two at a significance level of .001; the plot (a rotation of the top-right plot above) can be seen below. Following intuition, we can conclude that the regions in the south receive more rainfall.





The variograms for the data set in question can be seen below. For both cases (classic and robust), the same general trend is captured by the variog function. As expected, the rainfall becomes more uncorrelated (exhibits more variance) between locations that grow farther apart, with close destinations exhibiting similar rainfall collection. What's interesting in the plot below is the slight dip around lags/distances 9 and 10. A trough (or at least a constant slope) at this location suggests that there might exist a distance at which two locations are actually more likely to receive similar rainfall; again, this might be attributed to data skewed towards more popular locations, such as cities, but it is interesting nonetheless. Both variograms show that at a distance of more than 10, the data has high variance, approaching the sill very rapidly.



Appendix

```
# Problem 1
rm(list=ls())

# install.packages('ggmap')
library(ggmap)

zoo_addr = '500 S Greenlawn Ave, South Bend, IN 46615'
ohare_addr = '10000 W O\'Hare Ave, Chicago, IL 60666'

zoo_map <- get_map(zoo_addr, zoom=15)
ggmap(zoo_map)

from <- zoo_addr
to <- ohare_addr

driving = mapdist(from, to, mode='driving')
driving$method = 'driving'
biking = mapdist(from, to, mode='bicycling')
biking$method = 'biking'
walking = mapdist(from, to, mode='walking')
walking$method = 'walking'

distances = data.frame()
distances = rbind(distances, driving)
distances = rbind(distances, biking)
distances = rbind(distances, walking)
distances

# Problem 2
rm(list=ls())

library(geoR)
library(sp)
library(geoR)

df = read.csv('precip.csv')
head(df)

coordinates(df) = ~LONGITUDE+LATITUDE
bubble(df, 'PRCP', col='grey', maxsize=1.5, pch=19, fill=T)
head(df)

df.geoR = as.geodata(df, data.col=1)

points(df.geoR, ylab="", xlab="", col="gray")
```

```
plot(df.geoR)
df.geoR

###

x_coord = df.geoR[[1]][, 1]
y_coord = df.geoR[[1]][, 2]
rain_mm = df.geoR[[2]]

plot(x_coord, rain_mm)
x_fit = lm(rain_mm ~ x_coord)
summary(x_fit)
abline(x_fit, col='red')

plot(y_coord, rain_mm)
y_fit = lm(rain_mm ~ y_coord)
summary(y_fit)
abline(y_fit, col='red')

###

lz.geoR = df.geoR
lz.geoR$data = log(df.geoR$data)

# example of classical variogram
lz.v = variog(lz.geoR, max.dist=1500)

# example of robust variogram
lz.v2 = variog(lz.geoR, max.dist=1500, estimator.type = c("modulus"))

# comparison of classical vs robust variogram
plot(lz.v2,pch=19)
lines(lz.v, type="p", pch=10)
legend(0, 2.0, pch=c(19,10), c("robust","classic"), cex=0.8)

# x=lz.v$u
# y1=lz.v$v
# y2=lz.v2$v
# for_plotting <- data.frame(x,y1,y2)
# require(ggplot2)
# ggplot(for_plotting, aes(x)) +
#   geom_line(aes(y=y1), colour="red") +
#   geom_line(aes(y=y2), colour="green")
```