Homework5Part1

Rafael Espinosa Castaneda

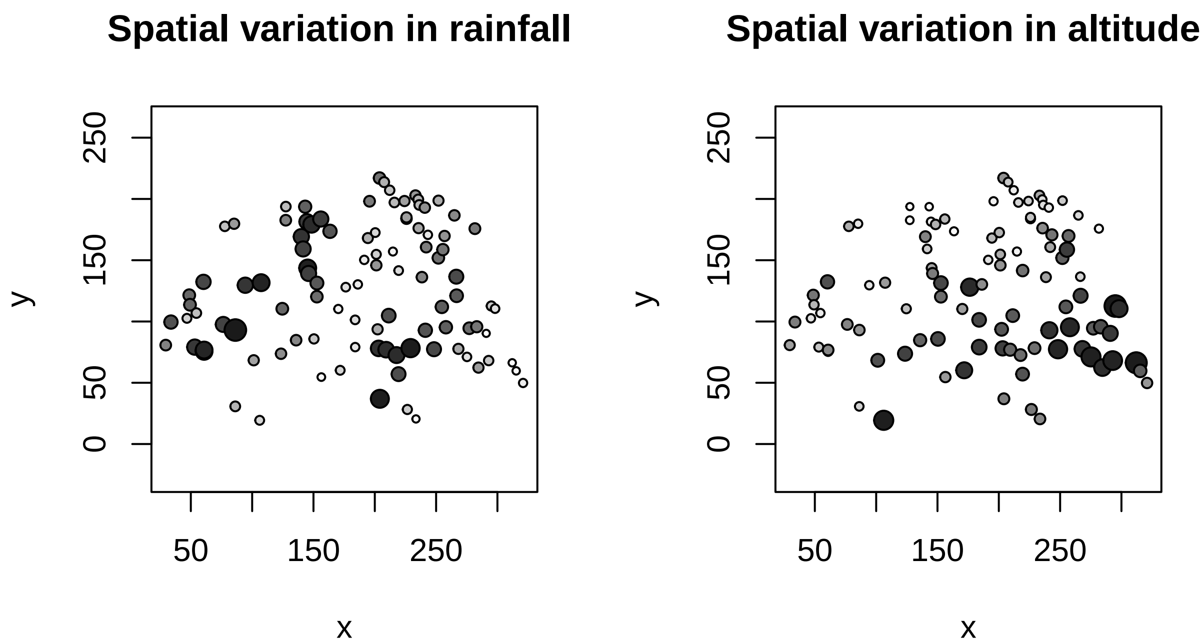
2023-02-18

```
# --------------------------Setting the parameters to be used------------------------
##################################################
## COMPUTING DEMO:  Kriging
##
## DSC 383: Advanced Predictive Models for Complex Data
## By:  Kate Calder, UT Austin
## Last updated: September 26, 2021
##################################################

## load libraries
library(ggplot2)
library(mapproj)
library(geoR)
library(leaflet)
library(gridExtra)
library(measurements)
library(dplyr)


# load scallops data
rain <- read.table("/Users/rafa/Documents/Master Austin/MAESTRIA_AUSTIN/Advanced Predictive Models/Hw5/

rain$altitude<-conv_unit(rain$altitude, "ft", "mi")
```

Part a

```
# create geodata (geoR) object
rain_geo_altitude <- as.geodata(
  cbind(rain$altitude,
        rain$x,
        rain$y),
  data.col = 1,
  coords.col = 2:3)



# create geodata (geoR) object
rain_geo_rainfall <- as.geodata(
  cbind(rain$rainfall,
        rain$x,
        rain$y),
  data.col = 1,
```

```
  coords.col = 2:3)


par(mfrow=c(1,2))

points.geodata(rain_geo_rainfall, col = gray(seq(1, 0.1, l = 100)),xlab = "x", ylab = "y")
title("Spatial variation in rainfall ")

points.geodata(rain_geo_altitude, col = gray(seq(1, 0.1, l = 100)),xlab = "x", ylab = "y")
title("Spatial variation in altitude ")
```



ANSWER

We can notice certain spatial dependence for the altitude. We can notice from the plot that in general the altitude is greater for coordinates $x >= 100$ and $y < 150$. For the rainfall variable we can notice a certain dependece such that greater values are around the center of the 2D coordinates.

Part b

```
lR_RA = lm(sqrt(rainfall)~altitude, data = rain)

summary(lR_RA)
```

```
##
## Call:
## lm(formula = sqrt(rainfall) ~ altitude, data = rain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9070 -2.9051 -0.9749  3.4518 11.2529
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  14.1273     0.9001  15.695   <2e-16 ***
## altitude     -8.3681     4.6910  -1.784   0.0775 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.275 on 98 degrees of freedom
## Multiple R-squared:  0.03145,    Adjusted R-squared:  0.02157
## F-statistic: 3.182 on 1 and 98 DF,  p-value: 0.07754
```

```r
print("The estimated error variance")
```

```
## [1] "The estimated error variance"
```

```r
print(4.275^2)
```

```
## [1] 18.27563
```

```r
print("The proportion of variance is ")
```

```
## [1] "The proportion of variance is "
```

```r
print(0.03145)
```

```
## [1] 0.03145
```

ANSWER

The equation is $\sqrt{rainfall} = 14.1273 - 8.3681 * Altitude$. The estimated error variance is 18.27563 and the proportion of variation in the square root of rainfall is explained by altitude is 0.03145

Part c

```r
N<-length(rain$rainfall)
#tot<-N*(N-1)

tot<-N*(N-1)/2
print(N)
```

```
## [1] 100
```

```r
print(tot)
```

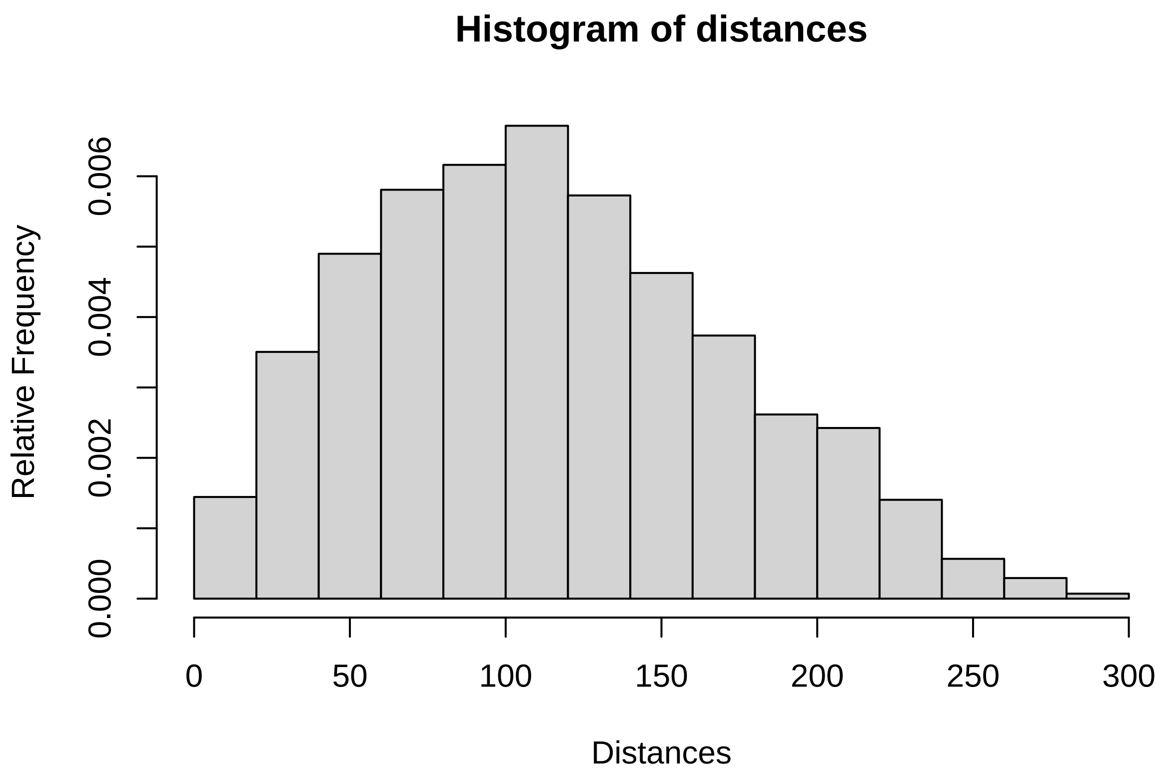
```
## [1] 4950
```

```r
distances<-rep(1,tot)
count<-0
for (i in 1:N)
     {
       for(j in (i+1):N)
       {
         if (i!=j)
         {
           count<-count+1
         distances[count]<- ( (rain$x[i]-rain$x[j])^2 +(rain$y[i]-rain$y[j])^2 )^(1/2)

         if (is.na(distances[count]))
         {
```

```
        print(paste("i",i,"j",j))
        print("The x_i")
        print(rain$x[i])
        print("The x_j")
        print(rain$x[j])
        print("The y_i")
        print(rain$y[i])
        print("The y_j")
        print(rain$y[j])
      }



      }

     }

   }
```

```
## [1] "i 100 j 101"
## [1] "The x_i"
## [1] 320.9114
## [1] "The x_j"
## [1] NA
## [1] "The y_i"
## [1] 49.82554
## [1] "The y_j"
## [1] NA
```

```
hist(distances, breaks=seq(from=0, to=300, by=20),freq=FALSE,xlab="Distances",ylab="Relative Frequency")
```



Part d

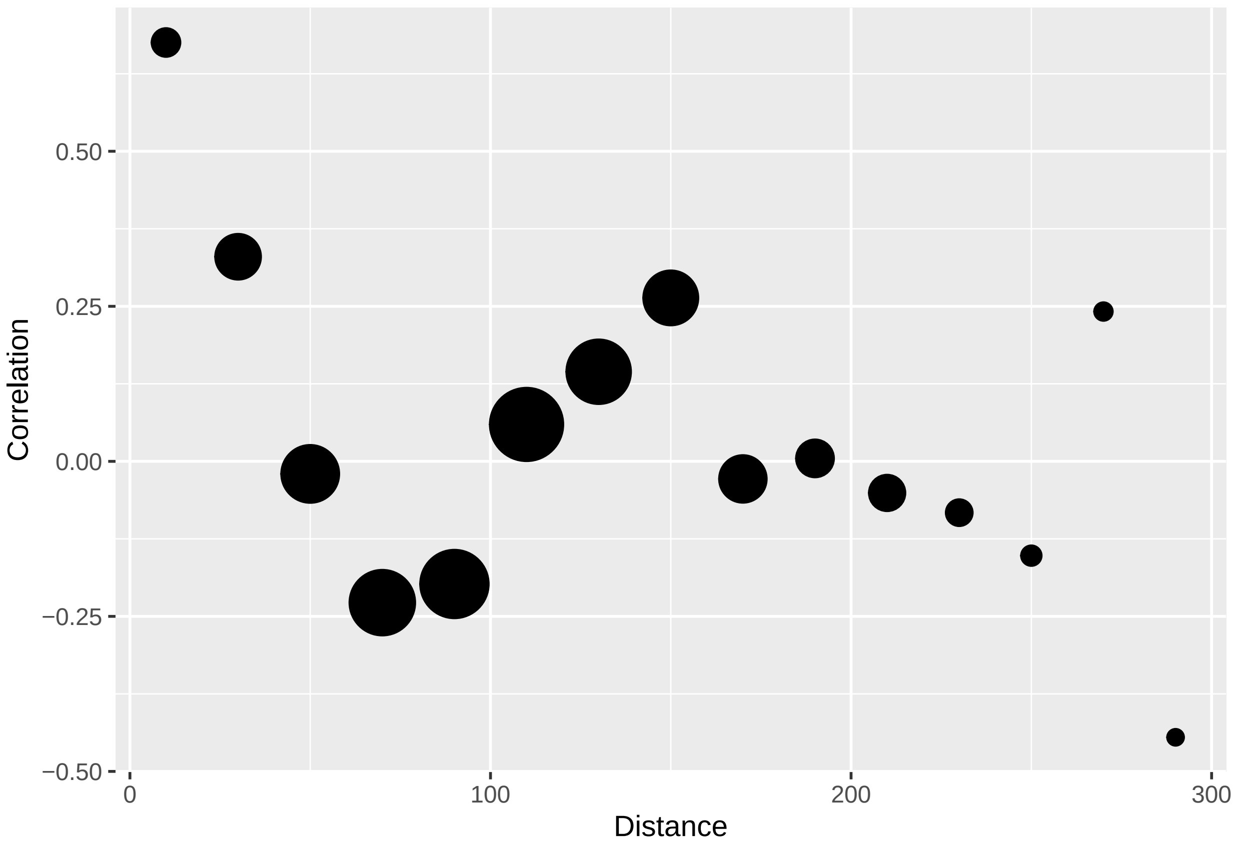
```
residuals<-resid(lR_RA)

bins_ranges<-c(20,40,60,80,100,120,140,160,180,200,220,240,260,280,300)






df1 <- data.frame(x_1=c(rain$x[1]),
                  x_2=c(rain$x[2]),distance=c( ( (rain$x[1]-rain$x[2])^2 +(rain$y[1]-rain$y[2])^2 )^(1/2


for (i in 1:N)
  {
  for(j in (i+1):N)
    {
    if (i!=j && (i!=1 && j!=2)  )
      {
      count<-count+1
      distances[count]<- ( (rain$x[i]-rain$x[j])^2 +(rain$y[i]-rain$y[j])^2 )^(1/2)


          df2 <- data.frame(x_1=c(rain$x[i]),
                            x_2=c(rain$x[j]),distance=c( ( (rain$x[i]-rain$x[j])^2 +(rain$y[i]-rain$y[j]

          df1<-bind_rows(df1, df2)

      }

  }


  }


correlation<- rep(0,length(bins_ranges))
counters_dist<- rep(0,length(bins_ranges))

count<-1

for (bin in bins_ranges)
{
  df_to_use<-filter(df1, distance<bin & distance>=bin-20 & x_1 !=x_2)
  counters_dist[count]<-length(df_to_use$x_1)
  correlation[count]<-cor(df_to_use$res1,df_to_use$res2)
  count<-count+1
}


df_to_use
```

```
##            x_1      x_2 distance     res1      res2
## 2...1 33.77939 312.0674 280.2653 3.129911 -6.188751
## 2...2 33.77939 315.2924 284.3167 3.129911 -7.978380
## 2...3 33.77939 320.9114 291.4023 3.129911 -5.552570
## 4     48.71439 320.9114 281.4636 1.013164 -5.552570
```

```
centers<-c(10,30,50,70,90,110,130,150,170,190,210,230,250,270,290)



df2<-data.frame(centerBins=centers,corr=correlation,num_dist=counters_dist)


counters_dist
```

```
##  [1] 143 341 482 572 605 662 560 448 362 249 232 121  47  24   4
```

```
as_geo_corr <- as.geodata(
  cbind(counters_dist,
        centers,
        correlation),
  data.col = 1,
  coords.col = 2:3)

#points.geodata(as_geo_corr,xlab = "center bins", ylab = "correlation")


ggplot() + geom_point(data = df2, aes(x =centerBins , y = corr),cex=10*(counters_dist+min(counters_dist)
```



Part e

In Switzerland, a small country it shouldn't be very likely to have pair of locations extremely far one from the other. That's why it is much more likely a distance between 80 and 100 than distances greater than 280.