Pb and Parks

Marc Los Huertos

11/25/2019

## Introduction

As we have learned, R can be powerful, but it’s often hard to remember what to do if it’s been a while since you have used it. I have created this document to help you analyze the Pb data – why? Because I am more interested in having you understand what the analysis process is than the actual R code.

But some understanding of R is useful so you can see the steps needed for the analysis!

## Methods

## Results

### Read the Data

First, we read the data. I like to create a filename object that I can modify based on my own path and filename.

filename = "/home/CAMPUS/mwl04747/github/Environmental\_Hazards/Data/Pb\_and\_Parks/EA 30 Data Entry - Sheet1.csv"  
import <- read.csv(filename)  
str(import)

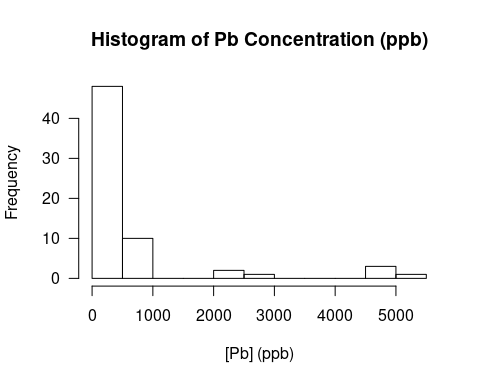
## 'data.frame': 67 obs. of 37 variables:  
## $ Type : Factor w/ 3 levels "CalBlk","CalStd",..: 1 2 2 2 3 3 3 3 3 3 ...  
## $ Level : int NA 1 2 3 NA NA NA NA NA NA ...  
## $ Sample.Name : Factor w/ 54 levels "1","10","11",..: 50 50 52 54 50 1 12 23 34 45 ...  
## $ Rack : Factor w/ 2 levels "Monday","Tuesday": NA NA NA NA NA 1 1 1 1 1 ...  
## $ Cap....marc.s.IDs.: int NA NA NA NA NA 1 2 3 4 5 ...  
## $ EA30\_IDs : Factor w/ 49 levels "#13 Sample 1 CPark",..: NA NA NA NA NA 8 11 12 13 17 ...  
## $ PARK : Factor w/ 10 levels "Blaisdell","CPark",..: NA NA NA NA NA 6 6 1 1 1 ...  
## $ Partner.Names : Factor w/ 6 levels "","Aurora, Ali, Yonatan",..: 1 1 1 1 1 1 1 4 6 4 ...  
## $ Replicate : int NA NA NA NA NA NA NA NA NA NA ...  
## $ DISTANCE10 : Factor w/ 9 levels "1,392.00","1,768.00",..: NA NA NA NA NA 2 2 9 9 9 ...  
## $ DISTANCE66 : num NA NA NA NA NA ...  
## $ Vial.Number : int 1101 1101 1301 1201 1101 3101 3102 3103 3104 3105 ...  
## $ Na23 : Factor w/ 64 levels "","<0.000","0",..: 2 3 60 48 6 58 27 59 9 10 ...  
## $ Mg24 : num 3.57e-04 0.00 6.65e+01 4.98e+02 1.62e-02 ...  
## $ Al27 : Factor w/ 64 levels "","<0.000","0",..: 4 3 10 55 5 42 43 23 33 26 ...  
## $ K39 : Factor w/ 64 levels "","<0.000","0",..: 2 3 63 54 4 42 48 35 52 51 ...  
## $ Ca44 : Factor w/ 64 levels "","<0.000","0",..: 2 3 26 19 4 52 51 50 55 35 ...  
## $ V51 : num 6.87e-03 0.00 7.16e+02 4.98e+03 1.68e-01 ...  
## $ Cr52 : Factor w/ 63 levels "","<0.000","0",..: 2 3 62 56 4 44 46 32 35 12 ...  
## $ Mn55 : num 3.94e-03 0.00 6.88e+02 4.98e+03 1.61e-01 ...  
## $ X56Fe : num 9.94e-05 0.00 6.74e+01 4.98e+02 2.03e-02 ...  
## $ Co59 : Factor w/ 64 levels "","<0.000","0",..: 2 3 39 24 4 8 11 61 7 43 ...  
## $ Cu63 : Factor w/ 63 levels "","<0.000","0",..: 2 3 61 51 4 36 38 23 25 17 ...  
## $ Zn66 : num 3.13e-02 0.00 6.95e+02 4.98e+03 2.49e-01 ...  
## $ As75 : num 6.11e-03 0.00 6.49e+02 4.99e+03 4.64e-01 ...  
## $ Se78 : Factor w/ 64 levels "","<0.000","0",..: 2 3 63 54 6 47 50 37 44 57 ...  
## $ As75.1 : num 2.40e-03 0.00 6.70e+02 4.98e+03 4.11e-01 ...  
## $ Se78.1 : Factor w/ 64 levels "","<0.000","0",..: 2 3 63 57 9 48 47 36 39 8 ...  
## $ Ag107 : Factor w/ 55 levels "","<0.000","0",..: 37 3 47 30 21 45 10 8 29 4 ...  
## $ Cd111 : num 1.69e-03 0.00 7.41e+02 4.98e+03 1.90e-01 ...  
## $ Cd114 : Factor w/ 64 levels "","<0.000","0",..: 2 3 63 50 5 42 39 18 24 19 ...  
## $ Sb121 : num 2.01e-03 0.00 6.64e+02 4.98e+03 3.68e-01 ...  
## $ Ba137 : Factor w/ 64 levels "","<0.000","0",..: 2 3 35 28 4 8 62 43 12 10 ...  
## $ Ba138 : Factor w/ 64 levels "","<0.000","0",..: 2 3 36 28 4 9 63 43 13 10 ...  
## $ Tl205 : num 3.58e-03 0.00 6.61e+02 4.98e+03 1.10 ...  
## $ Pb208 : num 0 0 664.466 4983.553 0.158 ...  
## $ U238 : num 1.73e-03 0.00 6.58e+02 4.98e+03 2.08e-01 ...

I fixed some of the values that were treated as factors, i.e. categorical data. For example, Pb has “<0.0001” which was treated as a character string instead of a numeric value – this the entire variable was being treated as a character string. We fixed that by changing “<0.0001” to 0. I haven’t done this for other variables yet.

## Exploratory Analysis

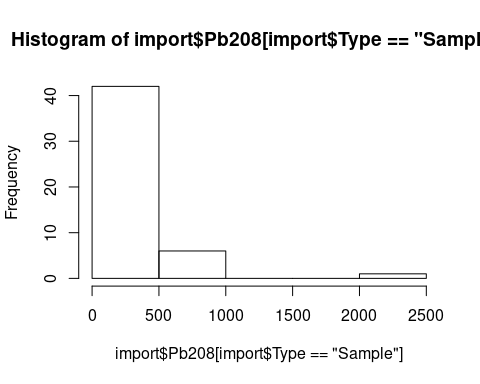
We can easily make a histogram as below…

hist(import$Pb208, main="Histogram of Pb Concentration (ppb)", xlab="[Pb] (ppb)", las=1)



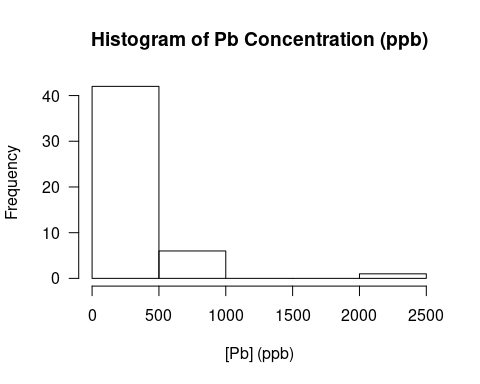
However, we only want to use the sample data, not the standards and blanks to calibrate the analysis run. See the Column A in the google sheet. So, let’s subset the data to use only the sample data.

hist(import$Pb208[import$Type=="Sample"])



Let’s clean this up a bit:

hist(import$Pb208[import$Type=="Sample"], main="Histogram of Pb Concentration (ppb)", xlab="[Pb] (ppb)", las=1)



Based on this, let’s create a dataset that only has the samples, so we don’t have to worry about this for the rest of the analysis. To do this we can use the subset() function.

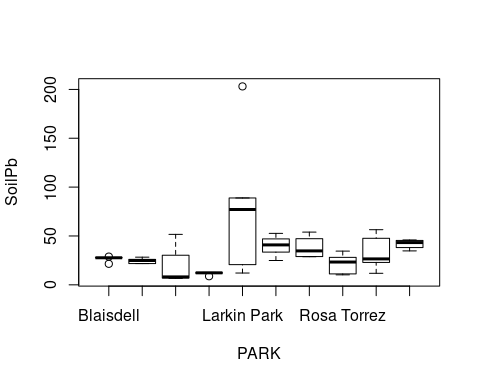
parkdata = subset(import, subset=import$Type=="Sample" & import$Sample.Name!="Blank")

## Converting the Values on a per gram of soil basis

Currently, we have a concentration based on 10 grams of soil that was digested and diluted upto 100 mL. So, we next need to change our concentration in the digestion relative to the soils (ug/g -> mg/kg)

parkdata$SoilPb = parkdata$Pb208/10

boxplot(SoilPb ~ PARK, data=parkdata)



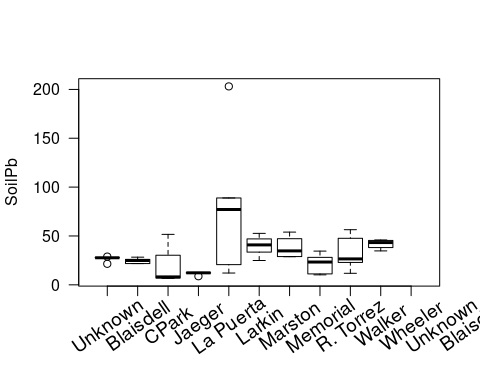
Let’s fix the names of the parks and then adjust the size to fit on the boxplot.

droplevels(parkdata$PARK)

## [1] Marston Marston Blaisdell   
## [4] Blaisdell Blaisdell Blaisdell   
## [7] Jaeger Marston Memorial Park   
## [10] Marston Jaeger Blaisdell   
## [13] Jaeger Jaeger Memorial Park   
## [16] Rosa Torrez Memorial Park Wheeler Park   
## [19] Marston Memorial Park Rosa Torrez   
## [22] Wheeler Park Wheeler Park CPark   
## [25] Jaeger Rosa Torrez Rosa Torrez   
## [28] Rosa Torrez Wheeler Park CPark   
## [31] <NA> La Puerta Sports Park La Puerta Sports Park  
## [34] La Puerta Sports Park La Puerta Sports Park La Puerta Sports Park  
## [37] CPark Larkin Park Walker Beach   
## [40] CPark CPark Larkin Park   
## [43] Walker Beach Walker Beach Larkin Park   
## [46] Larkin Park Walker Beach Larkin Park   
## [49] Walker Beach <NA>   
## 10 Levels: Blaisdell CPark Jaeger La Puerta Sports Park ... Wheeler Park

levels(parkdata$PARK) <- c("Unknown", "Blaisdell", "CPark", "Jaeger", "La Puerta", "Larkin", "Marston", "Memorial", "R. Torrez", "Walker", "Wheeler")

par(las=1)  
boxplot(SoilPb ~ PARK, data=parkdata, xlab="", xaxt="n")  
axis(1, 1:11, parkdata$PARK, labels=F)  
  
text(x = 1:length(parkdata$PARK),  
 y = par("usr")[3] - 40,  
 labels = unique(levels(parkdata$PARK)),  
 xpd = NA,  
 ## Rotate the labels by 35 degrees.  
 srt = 35,  
 cex = 1.2)



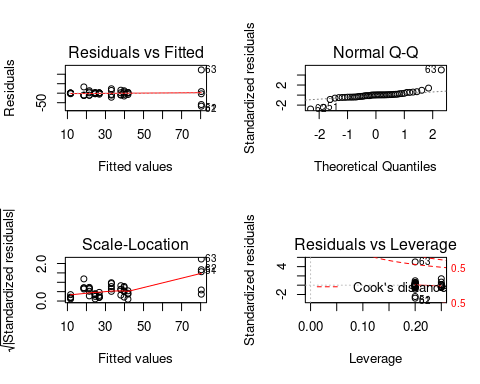
## Null Hypothesis Testing

No difference between Pb by parks.

pb.aov <- aov(SoilPb ~ PARK, data=parkdata)  
summary(pb.aov)

## Df Sum Sq Mean Sq F value Pr(>F)   
## PARK 9 16158 1795.3 2.399 0.0296 \*  
## Residuals 37 27684 748.2   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
## 3 observations deleted due to missingness

par(mfrow=c(2,2))  
plot(pb.aov)



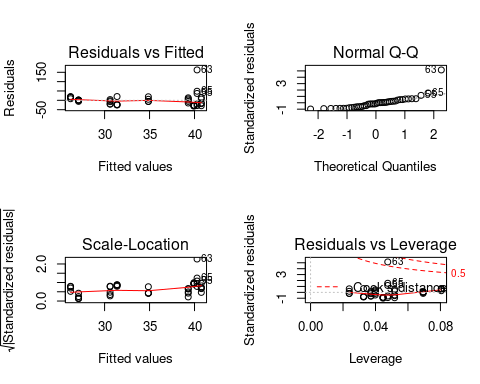
par(mfrow=c(1,1))

Testing distance

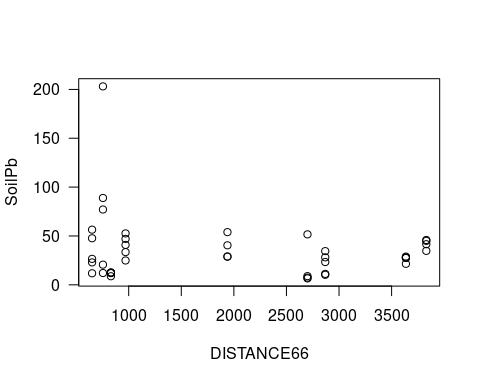
parkdata$DISTANCE10 <- as.numeric(parkdata$DISTANCE10)  
pb.lm <- lm(SoilPb ~ DISTANCE66, data=parkdata)  
summary(pb.lm)

##   
## Call:  
## lm(formula = SoilPb ~ DISTANCE66, data = parkdata)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -31.138 -20.161 -4.020 8.348 162.794   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 43.736892 9.537638 4.586 4.39e-05 \*\*\*  
## DISTANCE66 -0.004574 0.004137 -1.106 0.275   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 32.43 on 40 degrees of freedom  
## (8 observations deleted due to missingness)  
## Multiple R-squared: 0.02966, Adjusted R-squared: 0.005405   
## F-statistic: 1.223 on 1 and 40 DF, p-value: 0.2754

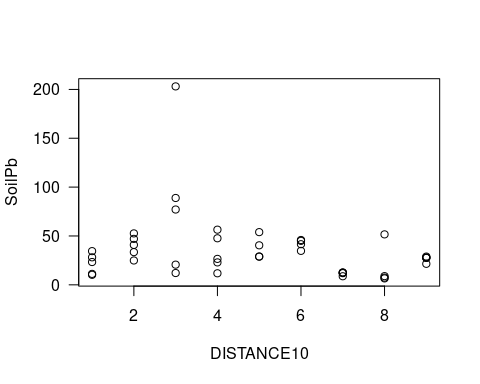
par(mfrow=c(2,2))  
plot(pb.lm)



par(mfrow=c(1,1), las=1)  
plot(SoilPb ~ DISTANCE66, data=parkdata)



plot(SoilPb ~ DISTANCE10, data=parkdata)



## Discussion

## Conclusion

## Literature Cited