Initial data exploration

J. Swall

2016-07-14

I saved the first worksheet ("All guano Data") from your Excel workbook (Hg_Guano Only.xlsx) into a separate file. Here, we read in this data.

```
library("openxlsx")
fileWpath = "C://Users//jenise//Google Drive//amy_bat_hg//explore//all_guano_w_species.xlsx"
rawDF <- read.xlsx(xlsxFile = fileWpath, colNames=TRUE, rowNames=FALSE)
## Replace all strings "N/A" with "NA" (which R understands).
rawDF[rawDF=="N/A"] <- NA
```

Some of these observations are taken in 1 inch segments from the same core. The next piece of code identifies these observations based on the entries used in the Sample ID column and gives these more "human-readable" names. The code also introduces a column which contains the order in which the samples were taken from each core. **I assume here that a notation such as CLM C1 means that this is the top one inch from the core (i.e., this is the 1 inch segment that was most recently deposited), and that CLM C2 is the next 1 inch down.**

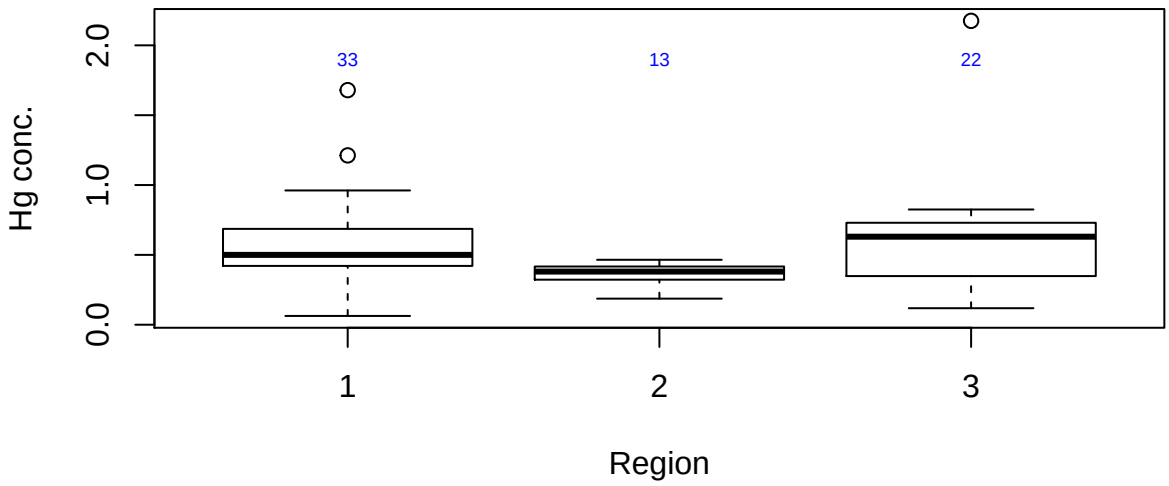
```
## Some of these data come from core samples, with measurements taken every inch throughout the core.  W
idPrefix <- c("CLM C", "COT C1 ", "FCCI C1 ", "FCCI C2 ", "JUD C1 ", "JUD C2 ",
              "UFBH C1-", "UFBH C2-")
names(idPrefix) <- c("GSS 36 core", "FCS 872 core", "FCS 555 core 1", "FCS 555 core 2",
                     "FCS 556 core 1", "FCS 556 core 2", "UFBH core 1", "UFBH core 2")
rawDF[,"coreName"] <- NA
rawDF[,"coreOrder"] <- NA
for (i in 1:nrow(rawDF)){
  for (j in 1:length(idPrefix)){
    which.match <- which(paste0(idPrefix[j], 1:20) == rawDF[i, "SampleID"])
    if (length(which.match) == 1){ #Match was found
      rawDF[i, "coreName"] <- names(idPrefix[j])
      rawDF[i, "coreOrder"] <- which.match
    }
  }
}
rm(i, j, which.match)
```

There is reason to assume that the concentrations collected from a single core are **not** independent. This is important because many statistical procedures assume that we either have independent observations, or they include complicated techniques to account for the inherent dependence. I would like to try to compare mercury concentrations that are likely to have been deposited most recently, and I'm thinking that these are the ones that were not collected as part of a core **or** were collected in the top 1 inch of the core. Also, it looks like region 4 only has 2 observations, which is not sufficient to compare them with the other regions. I'll exclude those.

```
toplayerDF <- subset( rawDF, ( is.na(coreName) | coreOrder==1 ) & (Region!=4) )
```

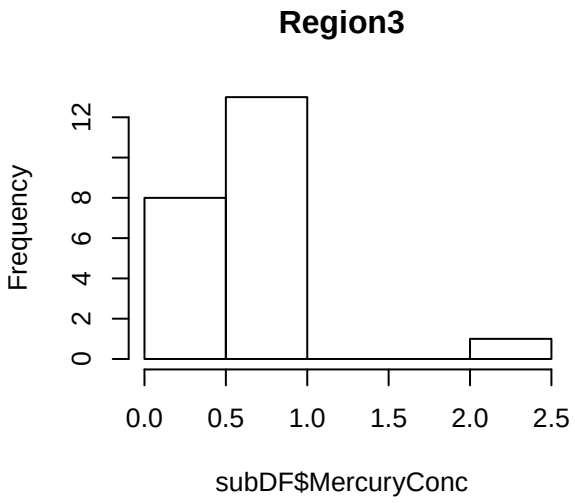
Boxplots of mercury concentrations vs. region number, with sample sizes above each boxplot:

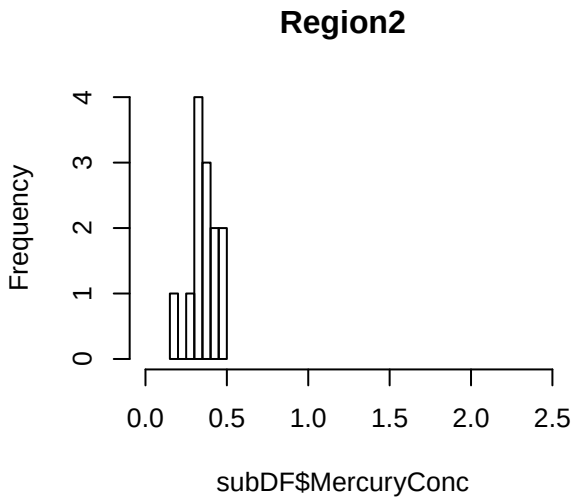
```
ctsByRegion <- table(toplayerDF$Region)
boxplot(MercuryConc ~ Region, data=toplayerDF, xlab="Region", ylab="Hg conc.")
for (iNm in names(ctsByRegion)){
  text(as.numeric(iNm), 1.9, ctsByRegion[iNm], col="blue", cex=0.6)
}
```

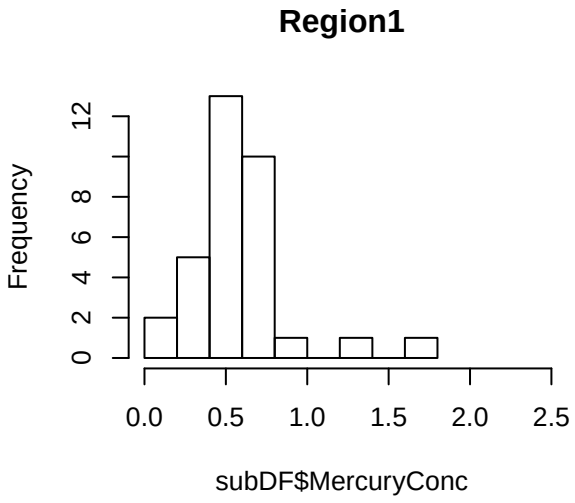


These boxplots indicate a wide variability in the top layer mercury concentrations among the various regions. The histograms of mercury concentrations also show that the shapes of the distributions aren't normal. In fact, the distributions seem to be quite different from one another. I tried the log and square root transformations (not shown), but they didn't adequately address these issues.

```
par(mfrow=c(2,2))
my.xlim <- c(0, max(toplayerDF$MercuryConc) + 0.25)
for (i in unique(toplayerDF$Region)){
  subDF <- subset(toplayerDF, Region==i)
  hist(subDF$MercuryConc, xlim=my.xlim, main=paste("Region", i, sep=""))
}
rm(subDF, my.xlim, i)

par(mfrow=c(1,1))
```



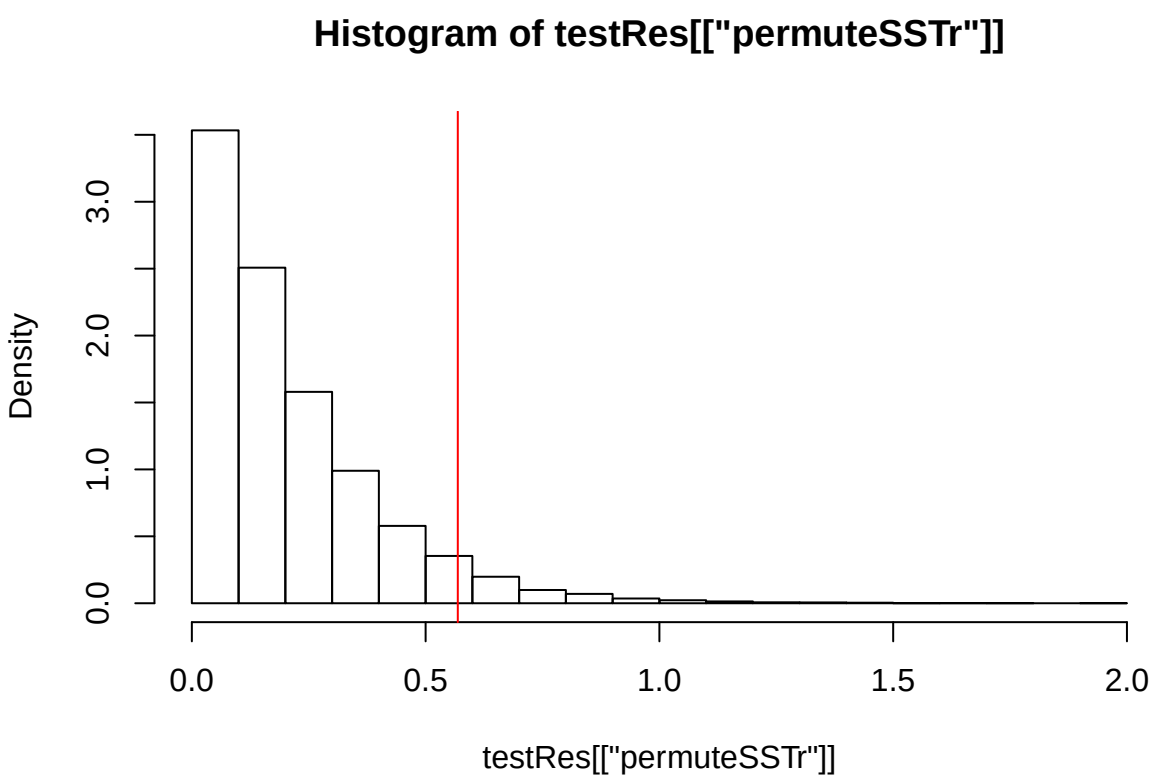




It seems like we might want to test the hypothesis that the average mercury concentrations are equal for regions 1-3. To do this, we'll have to choose a nonparameteric approach. Here, I've written some code to do a permutation test that is based on the sum of squares for the treatment effect; this is similar to the strategy used in a typical ANOVA approach, but without the distributional assumptions.

```
permute1WayAnova <- function(x, grp, numPermutations = 1000){

    ## Calculate overall mean, which is the same, regardless of
    ## what groups the observations are in.
    overallMean <- mean(x)
    ## The number of observations per group also stays the same.
    grpN <- table(grp)

    ## Calculate the test stat for the original grouping.
    origSSTr <- calcSSTrt(x, grp, overallMean, grpN)

    permuteSSTr <- NULL
```

```
    ## Permute the order of the data and recalculate the test stat.
    for (i in 1:numPermutations){
        permuteX <- sample(x, size=length(x), replace=FALSE)
        permuteSSTr <- c(permuteSSTr,
                         calcSSTrt(permuteX, grp, overallMean, grpN))
    }

    ## Approx. p-value by calculating what percentage of statistics from
    ## the permutations exceed this test statistic calculated from the
    ## original data.
    approxPval <- sum(permuteSSTr >= origSSTr)/numPermutations
    return(list(approxPval=approxPval, origSSTr=origSSTr, permuteSSTr=permuteSSTr))
}

calcSSTrt <- function(x, grp, overallMean, grpN){

    ## Calculate group means.
    grpMeans <- tapply(x, grp, mean)
    ## Calculate sum of squares associated with the treatment groups.
    sstrt <- sum( grpN * ( (grpMeans - overallMean)^2 ) )

    return(sstrt)
}

## Run this permutation test on our data.
testRes <- permute1WayAnova(toplayerDF$MercuryConc, toplayerDF$Region, numPermutations=20000)
hist(testRes[["permuteSSTr"]], prob=T)
abline(v=testRes[["origSSTr"]], col="red")
```




```
print(testRes[["approxPval"]])
```

```
## [1] 0.05465
```

This gives p-value just about the 5% level. I also got a similar result from a canned routine in the R package "coin". The code is below.

```
library("coin")
```

```
## Loading required package: survival
```

```
independence_test(MercuryConc ~ as.factor(Region), data=toplayerDF, teststat="quadratic", distribution=
```

```
##
##  Approximative General Independence Test
##
## data:  MercuryConc by as.factor(Region) (1, 2, 3)
## chi-squared = 5.4979, p-value = 0.0521
```

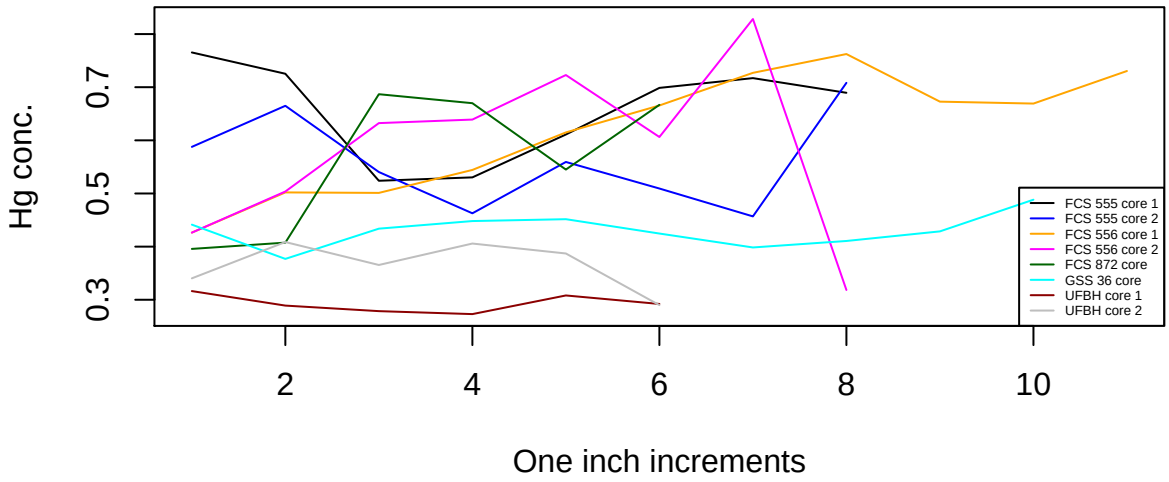
```
independence_test(MercuryConc ~ as.factor(Region), data=toplayerDF, teststat="quadratic", distribution=
```

```
##
##  Asymptotic General Independence Test
```

```
##
## data:  MercuryConc by as.factor(Region) (1, 2, 3)
## chi-squared = 5.4979, df = 2, p-value = 0.06399
```

In the following plot, I tried to look at the measurements that were taken at intervals along the same core. On the x-axis, I have the order the measurements were taken from the core (assuming again that 1 is the top 1 inch, 2 is the concentration from the next inch down, etc.). I don't see a clear relationship. **Do you think I'm intrepreting this correctly?**

```
## Find all rows with "intervals" in the Notes section.
subDF <- rawDF[!is.na(rawDF$coreName),]
## Divide into groups according to which core the measurements came from.
splitDF <- split(subDF, subDF$coreName)
my.col <- c("black", "blue", "orange", "magenta", "darkgreen", "cyan", "darkred", "gray")
y.rng <- range(subDF[,"MercuryConc"])
plot(c(1, max(subDF$coreOrder)), y.rng, type="n", xlab="One inch increments", ylab="Hg conc.")
legLabels <- NULL
for (i in 1:length(splitDF)){
  with(splitDF[[i]], lines(coreOrder, MercuryConc, col=my.col[i]))
  legLabels <- c(legLabels, names(splitDF)[i])
}
legend("bottomright", legend=legLabels, lty=1, col=my.col, cex=0.4)
```



I also make a plot of the mercury concentrations vs. the organic matter. I thought these might be related, but I don't have a good understanding of what kind of measurement is represented by the organic matter column. Should there be a relationship?

```
par(mfrow=c(1,1))
plot(MercuryConc ~ OrganicMatter, data=rawDF, xlab="Org. matter", ylab="Hg conc.", type="n")
my.col = c("black", "blue", "orange", "magenta")
my.pch = c(1, 16, 15, 17)
for (i in 1:4){
```

```
  subDF <- subset(rawDF, Region==i)
  points(MercuryConc ~ OrganicMatter, data=subDF, col=my.col[i], pch=my.pch[i])
}
legend("topleft", legend=1:4, cex=0.7, pch=my.pch, col=my.col)
```