Homework1.R

Frank

Mon Sep 12 09:01:24 2016

```r
# *****>>>>> Dingjue Ji dj333 <<<<<******
#
# I worked with * list other students here if applicable *
#
# STAT 530 Homework 1
# Due Monday, September 12, 9 AM
#
################################################################################
################################################################################

############################################# 80 characters ####################
# PRELIMINARY EXPLORATION
#

x <- read.csv("Zinc.csv", as.is=TRUE)  # Run this line without an error
                                        # before continuing.

# Two groups of rats were given a dietary supplement of calcium (group A) or
# a placebo (group B).  The groups were formed at random. Later, zinc
# concentrations were measured from blood samples, because researchers
# hypothesized that blood zinc levels might be a side effect of a calcium
# dietary supplement.
#
# How large is the data set?  Take a look at the first few cases (rows) and
# the last few cases to see if it looks reasonable.  Sometimes, it doesn't.
# If you get errors, below, then you likely ignored an error, above.
#
# In the area immediately below, provide any code you use for this basic
# exploration.  It won't be graded, but it should be included here for
# completeness and for your own use:

x.dim<-dim(x)
head(x)
```

```
##   ZINC GROUP
## 1 1.31     A
## 2 1.45     A
## 3 1.12     A
## 4 1.16     A
## 5 1.30     A
## 6 1.50     A
```

```r
tail(x)
```

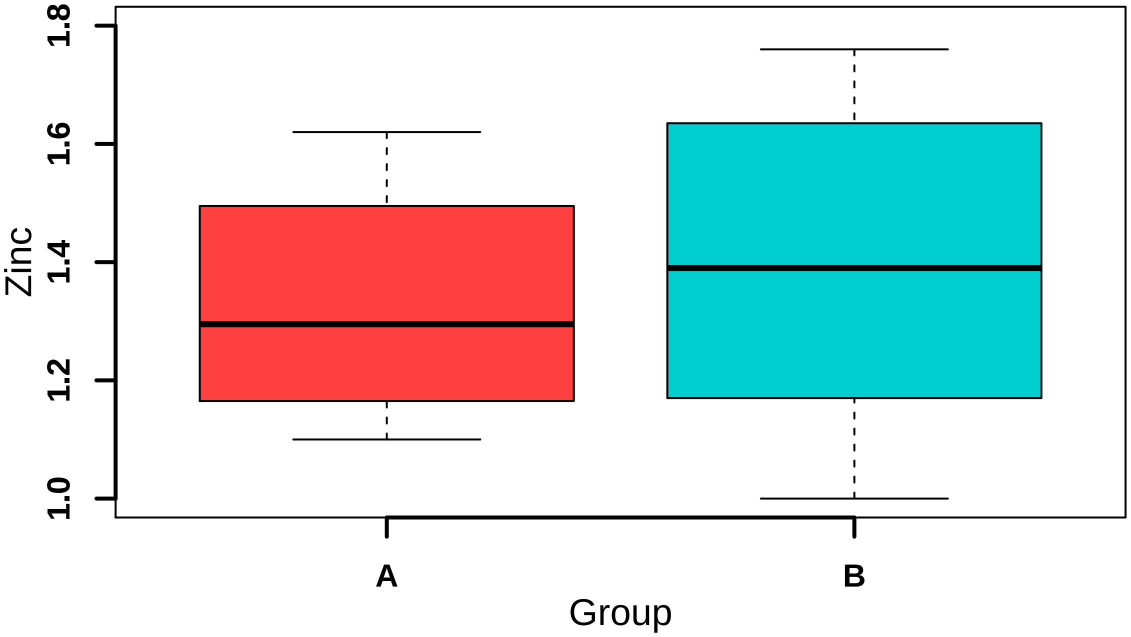
```
##    ZINC GROUP
## 34 1.74     B
## 35 1.19     B
## 36 1.15     B
## 37 1.20     B
## 38 1.59     B
## 39 1.47     B
```

```
# END of your informal exploratory data analysis code.

################################################################################
# Problem A: produce a side-by-side boxplot that illustrates the distribution of
# ZINC measurements broken down by GROUP.  It's a single command (which might be
# broken onto multiple lines because you should make nice axis labels, etc...).
# When you "compile notebook" you'll get the plot for free in the compiled
# report!

# Your code to produce the plot for Problem A:

par(mgp = c(2, 1, 0), font.axis=2)
boxplot(ZINC ~ GROUP, data = x, col = c('brown1', 'cyan3'),
        xlab = 'Group', ylab = 'Zinc', ylim = c(1,1.8),
        cex.lab=1.2, cex.axis=1)
axis(1, lwd=2, at = c(1, 2), labels=FALSE)
axis(2, lwd=2)
```



```
# End of Problem A
################################################################################
# Problem B: The following command might be expected
# to produce a scatterplot.  But when run, it produces an error:
#
## > plot(x$ZINC[x$GROUP=="A"], x$ZINC[x$GROUP=="B"])
## Error in xy.coords(x, y, xlabel, ylabel, log) :
##  'x' and 'y' lengths differ
#
# Special note: if you try to run this plot(...) command in your
# console and got a different error about object 'x' not found,
# then you forgot to interactively read in the data set for exploration
# via the read.csv(...) above.  The "compile notebook" process is run
# separately from your interactive R session in the Console.  Use the
# keyboard shortcuts or copy-paste to do the read.csv(...) in the console.
#
# We're going to ask you to answer two problems on ClassesV2 (as an
# experiment to see if it could save us all time).  But I'll pose
# the main question here.  First, with another student, discuss the error
# shown above with respect to R objects.  Second, for the homework,
# explain the error in the context of this problem
# (rat zinc measurements). This might require playing around just a
# little bit in R.
#
try(plot(x$ZINC[x$GROUP=="A"], x$ZINC[x$GROUP=="B"]))
# Sample number for A group
length(which(x$GROUP=='A'))
```

```
## [1] 20
```

```
# Sample number for B group
length(which(x$GROUP=='B'))
```

```
## [1] 19
```

```
# t test of two groups
t.test(ZINC~GROUP,data = x)
```

```
##
##  Welch Two Sample t-test
##
## data:  ZINC by GROUP
## t = -1.0952, df = 31.532, p-value = 0.2817
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.22187364  0.06676838
## sample estimates:
## mean in group A mean in group B
##        1.323500        1.401053
```

```
# End of Problem B
##################################################################################
# Problem C: Go find a data set that is (preferably) a CSV file or some other
# nicely-formatted text file.  At this point, it should have variable names
# in the first row with the actual data cases starting in the second row.
# Otherwise, you may have trouble. If it's Excel, save it as CSV.  Put a copy of
# it into your Homework1 folder.  This should be a data set that you at least
# find mildly interesting.  Read the data set into R.  Look at the beginning
# and end (the first and last few rows) to make sure it seems sensible.
# Explore two or three of the variables with things like tables, summary
# statistics, and basic graphics (if all variables are categorical, you should
# investigate the function mosaicplot() in R).  Experiment with R and
# show us that you can manage the basics.

# The expression profile of gene 'scap' in different brain regions and time
# periods
SCAP<-read.csv(file = 'SCAP_exp.csv', as.is = TRUE)
# Basic data check
```

```r
dim(SCAP)
```

```
## [1] 13 16
```

```r
names(SCAP)
```

```
##  [1] "MFC" "OFC" "VFC" "DFC" "STC" "ITC" "A1C" "IPC" "S1C" "M1C" "V1C"
## [12] "AMY" "HIP" "STR" "MD"  "CBC"
```

```r
head(SCAP)
```

```
##        MFC      OFC      VFC      DFC      STC      ITC      A1C      IPC
## 1 8.744387 8.663495 8.596488 8.619858 8.688224 8.754185 8.775285 8.647463
## 2 8.644623 8.629600 8.636587 8.529962 8.827285 8.712013 8.701597 8.672037
## 3 8.503067 8.441852 8.536901 8.565611 8.575513 8.853400 8.507492 8.578437
## 4 8.430021 8.430411 8.404712 8.362493 8.471079 8.453048 8.524388 8.475302
## 5 8.959435 8.815523 8.859580 9.021460 8.874622 8.843715 9.081646 9.053245
## 6 9.216678 9.249978 9.088164 8.986520 9.040408 9.214680 9.090944 8.994538
##        S1C      M1C      V1C      AMY      HIP      STR       MD      CBC
## 1 8.674208 8.588565 8.574953 8.550908 8.611942 8.700203 8.723485 8.720055
## 2 8.653538 8.616485 8.728142 8.559422 8.649772 8.675378 8.481930 8.678543
## 3 8.824065 8.851560 8.571700 8.662872 8.688441 8.621385 8.625292 8.403870
## 4 8.425553 8.576310 8.505037 8.513193 8.434508 8.462282 8.616298 8.570858
## 5 8.947650 9.049425 8.922935 8.881595 8.866197 9.056257 8.758453 8.794325
## 6 9.089284 9.054550 9.052242 9.105542 9.237150 9.254738 9.343258 9.142226
```

```r
tail(SCAP)
```

```
##         MFC      OFC      VFC      DFC      STC      ITC      A1C      IPC
## 8  9.098743 8.757187 9.132324 9.091225 9.209668 8.857398 9.184486 9.010752
## 9  8.705680 8.725150 8.583513 8.643987 8.485073 8.622987 8.830763 8.613153
## 10 8.815167 8.901825 8.760612 8.790693 8.834462 8.878132 8.690672 8.855627
## 11 8.592212 8.534123 8.458836 8.515926 8.637872 8.545644 8.578296 8.554803
## 12 8.608860 8.574125 8.567475 8.522835 8.515325 8.563047 8.595400 8.673030
## 13 8.542843 8.700028 8.435668 8.701892 8.600365 8.682900 8.584435 8.598925
##         S1C      M1C      V1C      AMY      HIP      STR       MD      CBC
## 8  9.111516 9.094212 9.003980 8.930955 9.161960 9.193960 9.281093 9.149914
## 9  8.575145 8.795195 8.598980 8.794747 8.813143 9.260710 8.975760 8.703860
## 10 8.808102 8.853403 8.958214 8.796505 9.019002 8.963962 8.950770 8.962154
## 11 8.575971 8.563802 8.642674 8.692508 8.770513 8.801501 8.794058 8.887430
## 12 8.718144 8.646766 8.796765 8.742406 8.820164 8.926180 8.874913 8.660288
## 13 8.647820 8.655707 8.693668 8.744413 8.751648 8.856757 8.885203 8.848202
```

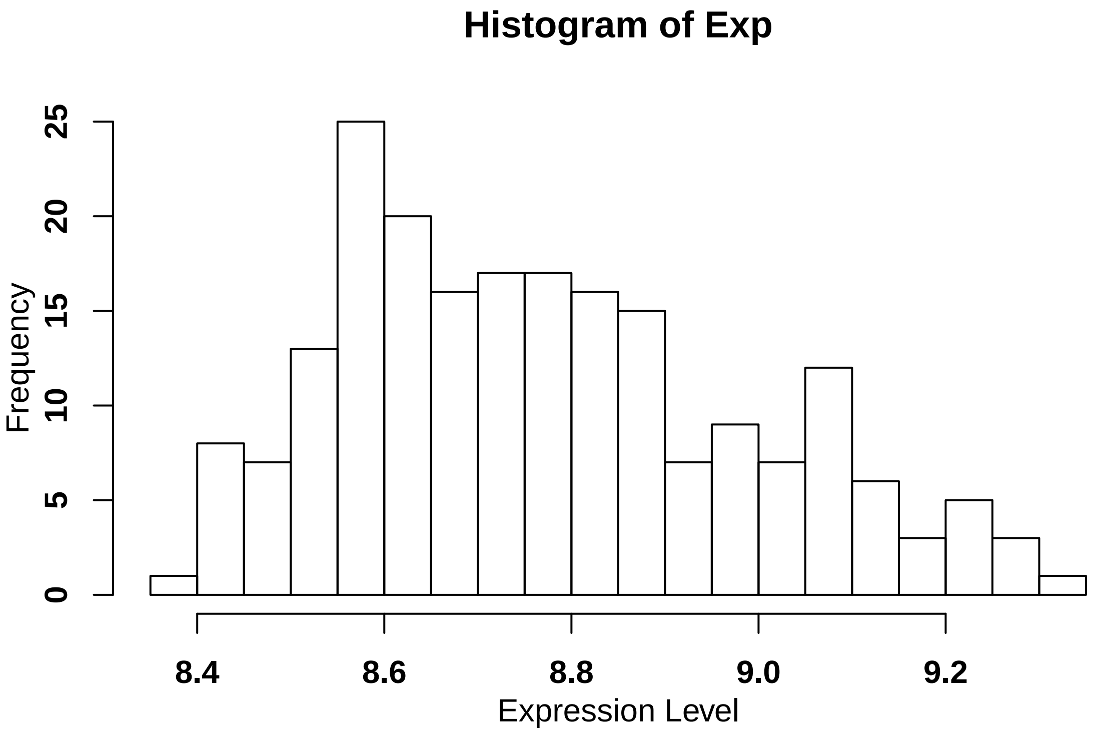
```r
# Data distribution check
sum_SCAP<-summary(SCAP)
mean_SCAP<-as.numeric(gsub('[^0-9\\.]', '', sum_SCAP[4,]))
mean_SCAP
```

```
##  [1] 8.747 8.721 8.685 8.708 8.729 8.756 8.774 8.743 8.753 8.777 8.758
## [12] 8.750 8.834 8.910 8.879 8.793
```

```r
var_SCAP<-apply(SCAP,2,var)
var_SCAP
```

```
##        MFC        OFC        VFC        DFC        STC        ITC 
## 0.05520539 0.05051739 0.05598373 0.05033115 0.04931896 0.03820314 
##        A1C        IPC        S1C        M1C        V1C        AMY 
## 0.05313673 0.03916159 0.04052556 0.03690113 0.03278933 0.02703565 
##        HIP        STR         MD        CBC 
## 0.05125559 0.06393665 0.06549831 0.04455770
```

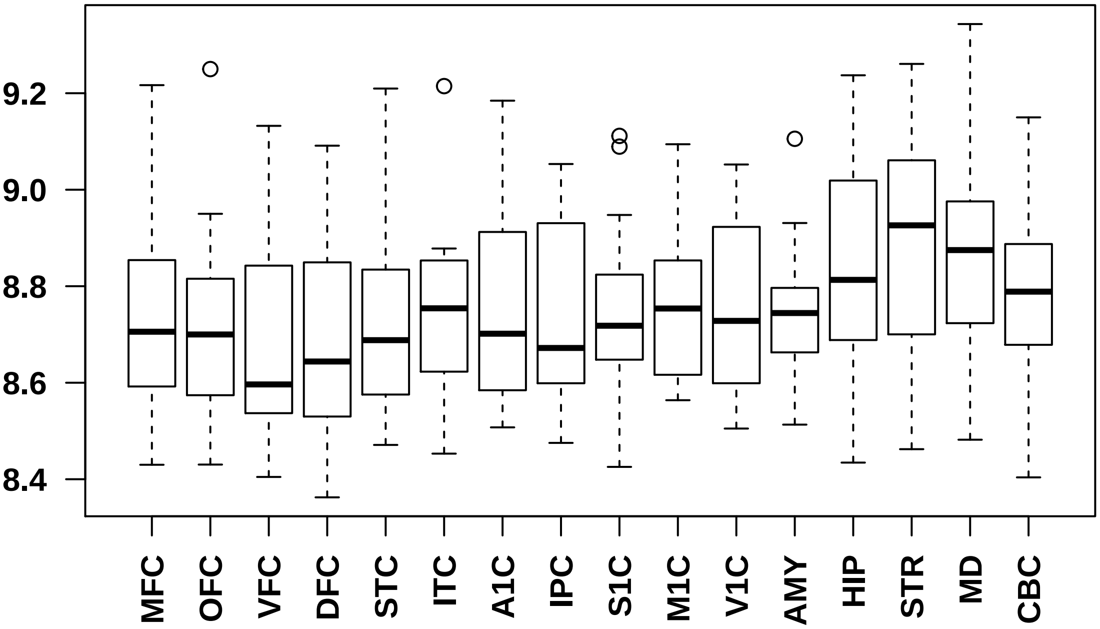
```
hist(as.vector(as.matrix(SCAP)), xlab = 'Expression Level',
     breaks=30, main = 'Histogram of Exp')
```



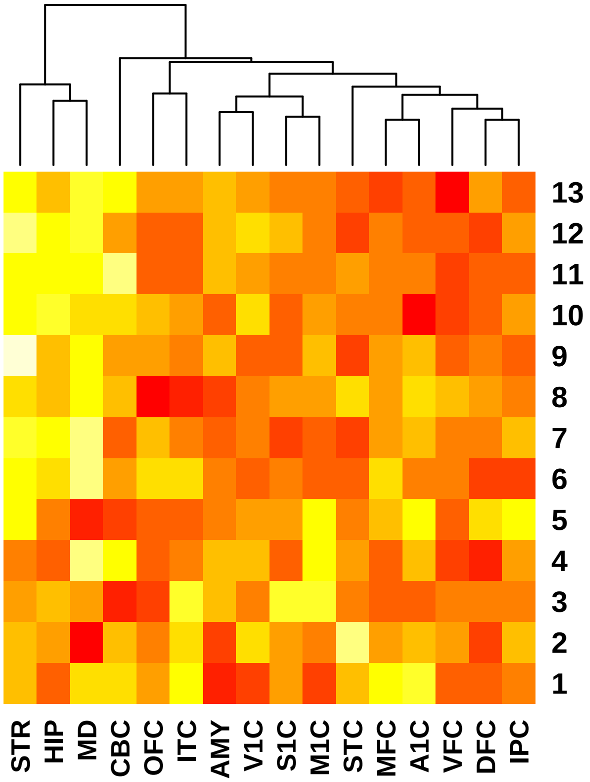
```
par(las = 2)
#Function to flatten tables with col & row names
trtab<-function(x, col.name){
  rows<-rownames(x)
  cols<-colnames(x)
  col1<-rep(rows, length(cols))
  col2<-rep(cols, each=length(rows))
  col3<-as.vector(as.matrix(x))
  X<-data.frame(cbind(col1, col2, col3), stringsAsFactors = FALSE)
  colnames(X)<-col.name
  return(X)
}
scap<-trtab(SCAP, c('Time', 'Region', 'Exp'))
# ANOVA for randomized block design
summary(aov(Exp~Time+Region ,data=scap))
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)    
## Time         12  7.099  0.5916  54.531  < 2e-16 ***
## Region       15  0.699  0.0466   4.294 8.28e-07 ***
## Residuals   180  1.953  0.0108                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

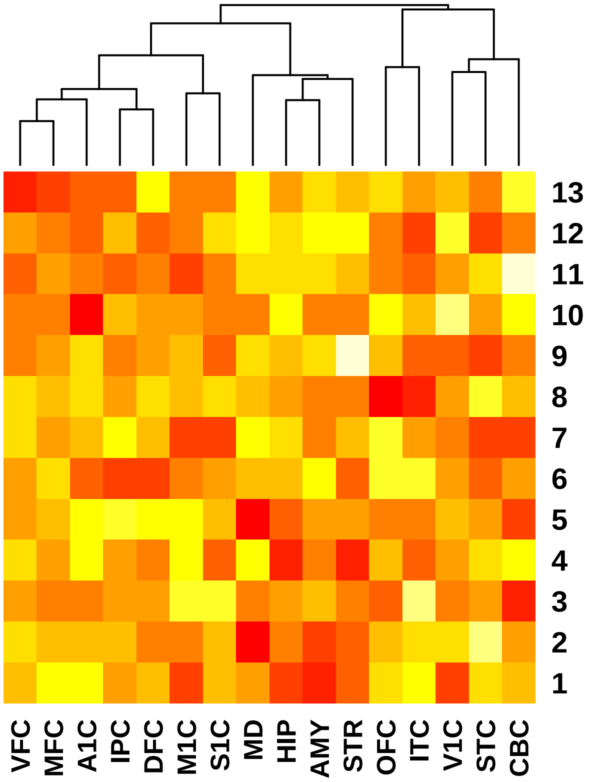
```r
boxplot(SCAP, use.cols=TRUE)
```



```r
# Regions are not clustered by their physical location
heatmap(as.matrix(SCAP), Rowv  = NA)
```



```r
#Normalize the data by columns
norm_SCAP<-apply(SCAP, 2, function(x) ((x-mean(x))/sd(x)))
# Regions are more likely to be juxtaposed based on location
heatmap(norm_SCAP, Rowv  = NA)
```



```
# End of code/discussion for Problem C
##################################################################################
#
# Done?  Press the "compile notebook" button in R Studio and produce either
# an HTML file or (if you get LaTeX installed) a PDF file.  Check it, make
# sure your name is on it, and print.  Printing this .R script without doing
# "compile notebook" is not sufficient.
```