STAT 579 Homework 5

Yifan Zhu

October 13, 2016

Problem 1

(a) read data

```
diurnaldata <- read.csv(file = "http://maitra.public.iastate.edu/stat579/datasets/diurnaldata.csv",
    header = T)
```

(b) mean at each time point

```
# create array of dimension 22810*11*2
diurnal.array <- as.matrix(diurnaldata[, -1])
dim(diurnal.array) <- c(22810, 11, 2)
diurnal.mean <- apply(X = diurnal.array, MARGIN = c(1, 2), FUN = mean)
```

(c) standardization

i.

```
# calculate the mean of the mean abundance level over all time-points for
# each gene
diurnal.mean.mean <- apply(X = diurnal.mean, MARGIN = 1, FUN = mean)
```

ii.

```
# replicate 11 times
diurnal.mean.mean.rep <- rep(diurnal.mean.mean, 11)
dim(diurnal.mean.mean.rep) <- c(22810, 11)

# eliminate the mean
diurnal.mean.mean.minusmean <- diurnal.mean - diurnal.mean.mean.rep
```

iii.

```
# calculate the standard deviation of each row of the matrix
diurnal.mean.sd <- apply(X = diurnal.mean, MARGIN = 1, FUN = sd)

# replicate the standard deviation
diurnal.mean.sd.rep <- rep(diurnal.mean.sd, 11)
dim(diurnal.mean.sd.rep) <- c(22810, 11)

# standardize
diurnal.mean.standardized <- diurnal.mean.mean.minusmean/diurnal.mean.sd.rep
```

(d)

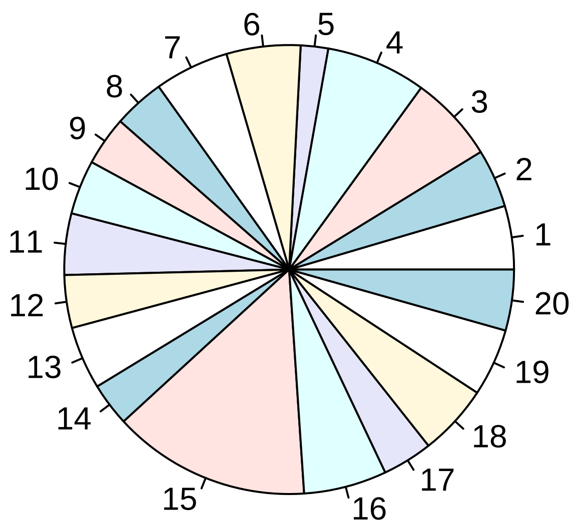
```r
micromean <- matrix(scan(file = "http://maitra.public.iastate.edu/stat579/datasets/micromeans.dat"),
    ncol = 11, byrow = T)

# calculate mean for each row and repliacate
micromean.mean <- apply(X = micromean, MARGIN = 1, FUN = mean)
micromean.mean.rep <- rep(micromean.mean, 11)
dim(micromean.mean.rep) <- c(20, 11)

# calculate standard deviation for each row and replicate
micromean.sd <- apply(X = micromean, MARGIN = 1, FUN = sd)
micromean.sd.rep <- rep(micromean.sd, 11)
dim(micromean.sd.rep) <- c(20, 11)

# standardization
micromean.standardized <- (micromean - micromean.mean.rep)/micromean.sd.rep
```

(e)

```r
diurnal.mean.rep <- rep(diurnal.mean.standardized, 20)
dim(diurnal.mean.rep) <- c(22810, 11, 20)

micromean.rep <- rep(t(micromean.standardized), each = 22810)
dim(micromean.rep) <- c(22810, 11, 20)

# calculate the distance
distance <- sqrt(apply((diurnal.mean.rep - micromean.rep)^2, MARGIN = c(1, 3),
    FUN = sum))

min.id <- apply(X = distance, MARGIN = 1, FUN = which.min)

# tabulate the frequency
table(min.id)
```

```
## min.id
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
## 1043  959 1424 1641  449 1213 1221  838  807  891 1008  868 1033  709 3246
##   16   17   18   19   20
## 1361  823 1168 1109  999
```

```r
# piechart
pie(table(min.id))
```



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

Problem 2

(a)

```r
tapply(X = state.x77[, "Income"], INDEX = state.region, FUN = mean)
```

```
##     Northeast         South North Central          West 
##      4570.222      4011.938      4611.083      4702.615
```

```r
aggregate(x = state.x77[, "Income"], by = list(state.region), FUN = mean)
```

```
##         Group.1        x
## 1     Northeast 4570.222
## 2         South 4011.938
## 3 North Central 4611.083
## 4          West 4702.615
```

(b)

```r
tapply(X = state.x77[, "Illiteracy"], INDEX = state.division, FUN = max)
```

```
##        New England    Middle Atlantic     South Atlantic 
##                1.3                1.4                2.3 
## East South Central West South Central East North Central 
##                2.4                2.8                0.9 
## West North Central           Mountain            Pacific 
##                0.8                2.2                1.9
```

```r
aggregate(x = state.x77[, "Illiteracy"], by = list(state.division), FUN = max)
```

```
##              Group.1   x
## 1        New England 1.3
## 2    Middle Atlantic 1.4
## 3     South Atlantic 2.3
## 4 East South Central 2.4
## 5 West South Central 2.8
## 6 East North Central 0.9
## 7 West North Central 0.8
## 8           Mountain 2.2
## 9            Pacific 1.9
```

(c)

```r
count <- rep(1, nrow(state.x77))

tapply(X = count, INDEX = state.region, FUN = sum)
```

```
##     Northeast         South North Central          West 
##             9            16            12            13
```

```r
aggregate(x = count, by = list(state.region), FUN = sum)
```

```
##         Group.1  x
## 1     Northeast  9
## 2         South 16
## 3 North Central 12
## 4          West 13
```

(d)

```r
names <- rownames(state.x77)

tapply(X = names, INDEX = state.division, FUN = "[")
```

```
## $`New England`
## [1] "Connecticut"   "Maine"         "Massachusetts" "New Hampshire"
## [5] "Rhode Island"  "Vermont"
##
## $`Middle Atlantic`
## [1] "New Jersey"   "New York"     "Pennsylvania"
##
## $`South Atlantic`
## [1] "Delaware"       "Florida"        "Georgia"        "Maryland"
## [5] "North Carolina" "South Carolina" "Virginia"       "West Virginia"
##
## $`East South Central`
## [1] "Alabama"     "Kentucky"    "Mississippi" "Tennessee"
##
## $`West South Central`
## [1] "Arkansas"  "Louisiana" "Oklahoma"  "Texas"
##
## $`East North Central`
## [1] "Illinois"  "Indiana"   "Michigan"  "Ohio"      "Wisconsin"
##
## $`West North Central`
## [1] "Iowa"         "Kansas"       "Minnesota"    "Missouri"
## [5] "Nebraska"     "North Dakota" "South Dakota"
##
## $Mountain
## [1] "Arizona"    "Colorado"   "Idaho"      "Montana"    "Nevada"
## [6] "New Mexico" "Utah"       "Wyoming"
##
## $Pacific
## [1] "Alaska"     "California" "Hawaii"     "Oregon"     "Washington"
```

```r
aggregate(names ~ state.division, data = state.x77, FUN = "[")
```

```
##        state.division
## 1         New England
## 2     Middle Atlantic
## 3      South Atlantic
```

```
## 4 East South Central
## 5 West South Central
## 6 East North Central
## 7 West North Central
## 8           Mountain
## 9            Pacific
##                                                                              names
## 1                       Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island, Vermont
## 2                                                      New Jersey, New York, Pennsylvania
## 3 Delaware, Florida, Georgia, Maryland, North Carolina, South Carolina, Virginia, West Virginia
## 4                                               Alabama, Kentucky, Mississippi, Tennessee
## 5                                                    Arkansas, Louisiana, Oklahoma, Texas
## 6                                             Illinois, Indiana, Michigan, Ohio, Wisconsin
## 7                       Iowa, Kansas, Minnesota, Missouri, Nebraska, North Dakota, South Dakota
## 8                          Arizona, Colorado, Idaho, Montana, Nevada, New Mexico, Utah, Wyoming
## 9                                             Alaska, California, Hawaii, Oregon, Washington
```

(e)

```r
state.size <- cut(x = state.x77[, "Population"], breaks = c(0, 2000, 10000,
    Inf), labels = c("Small", "Medium", "Large"))

tapply(X = state.x77[, "HS Grad"], INDEX = list(state.region, state.size), FUN = median)
```

```
##                Small Medium Large
## Northeast       55.9  56.00 51.45
## South           48.1  41.30 47.40
## North Central   53.3  54.50 52.90
## West            62.4  61.75 62.60
```

```r
aggregate(state.x77[, "HS Grad"] ~ state.region + state.size, data = state.x77,
    FUN = median)
```

```
##     state.region state.size state.x77[, "HS Grad"]
## 1      Northeast      Small                  55.90
## 2          South      Small                  48.10
## 3  North Central      Small                  53.30
## 4           West      Small                  62.40
## 5      Northeast     Medium                  56.00
## 6          South     Medium                  41.30
## 7  North Central     Medium                  54.50
## 8           West     Medium                  61.75
## 9      Northeast      Large                  51.45
## 10         South      Large                  47.40
## 11 North Central      Large                  52.90
## 12          West      Large                  62.60
```

Problem 3

(a)

```r
apply(X = mtcars, MARGIN = 2, FUN = mad)
```

```
##         mpg         cyl        disp          hp        drat          wt 
##   5.4114900   2.9652000 140.4763500  77.0952000   0.7042350   0.7672455 
##        qsec          vs          am        gear        carb 
##   1.4158830   0.0000000   0.0000000   1.4826000   1.4826000
```

(b)

```r
mtcar.median <- apply(mtcars, MARGIN = 2, FUN = median)
1.4826 * apply(abs(sweep(mtcars, MARGIN = 2, mtcar.median)), MARGIN = 2, FUN = median)
```

```
##         mpg         cyl        disp          hp        drat          wt 
##   5.4114900   2.9652000 140.4763500  77.0952000   0.7042350   0.7672455 
##        qsec          vs          am        gear        carb 
##   1.4158830   0.0000000   0.0000000   1.4826000   1.4826000
```