Module 2 Homework

Chengbo Gu

Problem 1. (30 points)

Computations on gene means of the Golub data set.

(a) Compute the mean expression values for every gene among "ALL" patients.

```
library(multtest)
data(golub)
gol.fac <- factor(golub.cl, levels=0:1, labels = c("ALL", "AML"))
meanALL <- apply(golub[,gol.fac=="ALL"], 1, mean)
```

(b) Compute the mean expression values for every gene among "AML" patients.

```
meanAML <- apply(golub[,gol.fac=="AML"], 1, mean)
```

(c) Give the biological names of the three genes with the largest mean expression value among "ALL" patients.

```
orderALL = order(meanALL, decreasing = TRUE)
print(golub.gnames[orderALL[1:3],2])
```

```
[1] "GB DEF = Chromosome 1q subtelomeric sequence D1S553"
[2] "37 kD laminin receptor precursor/p40 ribosome associated protein gene"
[3] "RPS14 gene (ribosomal protein S14) extracted from Human ribosomal protein S14 gene"
```

(d) Give the biological names of the three genes with the largest mean expression value among "AML" patients.

```
orderAML = order(meanAML, decreasing = TRUE)
print(golub.gnames[orderAML[1:3],2])
```

```
[1] "GB DEF = mRNA fragment for elongation factor TU (N-terminus)"
[2] "GB DEF = HLA-B null allele mRNA"
[3] "Globin, Beta"
```

Problem 2. (30 points)

More work on the Golub data set.

(a) Save the expression values of the first five genes (in the first five rows) for the AML patients in a csv file "AML5.csv".

```
AML5 <- golub[1:5, gol.fac == "AML"]
write.csv(AML5, file="AML5.csv")
```

(b) Save the expression values of the first five genes for the ALL patients in a plain text file "ALL5.txt".

```
ALL5 <- golub[1:5, gol.fac == "ALL"]
write.table(ALL5, file="ALL5.txt")
```

(c) Compute the standard deviation of the expression values on the first patient, of the 100th to 200th genes (total 101 genes).

```
exp100To200 <- golub[100:200,1]
print(sd(exp100To200))
```

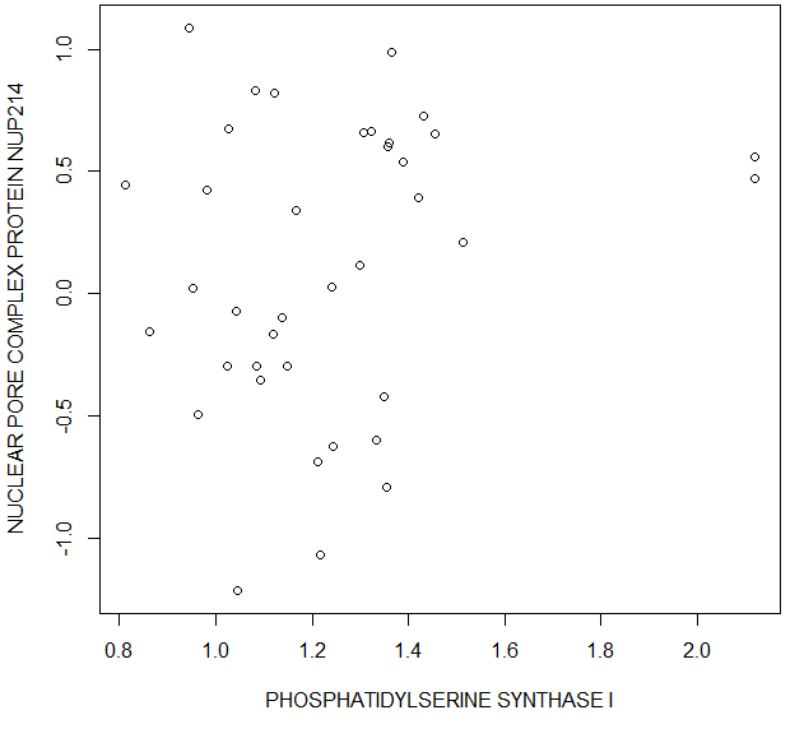
```
[1] 0.9174976
```

(d) Compute the standard deviation of the expression values of every gene, across all patients. Find the number of genes with standard deviation greater than 1.

```
print(sum(apply(golub, 1, sd)>1))
[1] 123
```

(e) Do a scatter plot of the 101th gene expressions against the 102th gene expressions, label the x-axis and the y-axis with the genes' biological names using xlab= and ylab= control options.

```
exp101th = golub[101,]
exp102th = golub[102,]
windows()
plot(exp102th, exp101th, xlab=golub.gnames[102,2], ylab=golub.gnames[101,2])
```

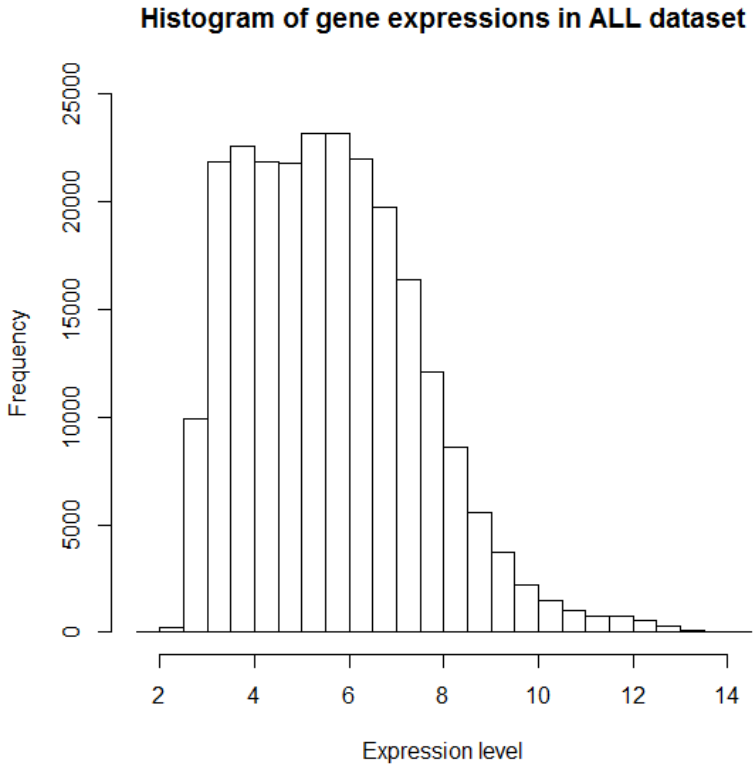


Problem 3. (20 points)

Work with the ALL data set. Load the ALL data from the ALL library and use str and openVignette() for a further orientation.

(a) Use exprs(ALL[,ALL$BT=="B1"] to extract the gene expressions from the patients in disease stage B1. Produce one histogram of these gene expressions in the this matrix.

```
rm(list=ls())
library(ALL)
data(ALL)
windows()
hist(exprs(ALL[,ALL$BT=="B1"]),ylim=c(0,25000), xlab="Expression level", main="Histogram of gene expressions in ALL dataset")
```



(b) Compute the mean gene expressions for every gene over these B1 patients.

```
meanExp <- apply(exprs(ALL[,ALL$BT=="B1"]), 1, mean)
```

(c) Give the gene identifiers of the three genes with the largest mean.

```
orderB1 <- order(meanExp, decreasing = TRUE)
print(meanExp[orderB1[1:3]])
```

```
AFFX-hum_alu_at         31962_at       31957_r_at
       13.41648         13.16671         13.15995
```

Problem 4. (20 points)

We work with the "trees" data set that comes with R.

(a) Find the type of the trees data object.

```
rm(list=ls())
data(trees)
print(class(trees))
```

```
[1] "data.frame"
```

(b) Produce a figure with two overlaid scatterplots: Height versus Girth, Volume versus Girth(The Girth is on the x-axis). Do the Height plot with blue "+" symbols, and do the Volume plot with red "o" symbols. You need to learn to set the ylim= control option so that all points from the two plots can all show up on the merged figure.

```
windows()
plot(trees$Height~trees$Girth, xlab="Girth", ylab="Height and Volume", col="blue", pch="+",
ylim = c(0,90))
```

```
points(trees$Volume~trees$Girth, col = "red" , pch="o")
legend(18,20, c("Height", "Volume"), pch=c("+","o"), col = c("blue","red"))
```