Northeastern University
College of Science

Problem 1 (30 points)

For the Golub et al. (1999) data set, use appropriate Wilcoxon two-sample tests to find the genes whose mean expression values are higher in the ALL group than in the AML group. Use FDR adjustments at the 0.05 level. How many genes are expressed higher in the ALL group? Find the gene names for the top three genes with smallest p-values.

Answer)

```
#(a)
data(golub, package = "multtest")
gol.fac <- factor(golub.cl,levels=0:1, labels= c("ALL","AML"))
wilcox_test <- apply(golub, 1, function(x) wilcox.test(x~gol.fac, alternative
= "greater", exact=F)$p.value)
p.fdr <-p.adjust(p=wilcox_test, method="fdr")
sum(p.fdr<0.05)

#(b)
wilcox_test <- apply(golub, 1, function(x) wilcox.test(x~gol.fac, alternative
= "greater", exact=F)$p.value)
o <- order(wilcox_test,decreasing=FALSE)
golub.gnames[o[1:3],2]
```

Output)

a)

```
> sum(p.fdr<0.05)

[1] 388
```

b)

```
> golub.gnames[o[1:3],2]
[1] "TCF3 Transcription factor 3 (E2A immunoglobulin enhancer binding factors E 12/E47)"
[2] "Macmarcks"
[3] "VIL2 Villin 2 (ezrin)"
```

Problem 2 (15 points)

For the Golub et al. (1999) data set, apply the Shapiro-Wilk test of normality to every gene's expression values in the AML group. How many genes do not pass the test at 0.05 level with FDR adjustment? Please submit your R script with the answer.

Answer)

```
gol.fac <- factor(golub.cl, levels=0:1, labels = c("ALL","AML"))

AML<-(golub[,gol.fac=="AML"])

shapiro <- apply(AML,1,shapiro.test)

p.values <- sapply(shapiro, function(x) x$p.value)

p.fdr <- p.adjust(p=p.values, method="fdr")

p_fdr <- sum(p.fdr <0.05)

p_fdr
```

output)

```
> p_fdr

[1] 225
```

Problem 3 (15 points)

Gene "HOXA9 Homeo box A9" can cause leukemia (Golub et al., 1999). Use appropriate Wilcoxon two-sample tests to test if, for the ALL patients, the gene "HOXA9 Homeo box A9" expresses at the same level as the “CD33” gene. Please submit your R script with the answer.

Answer)

```
gol.fac <- factor(golub.cl, levels=0:1, labels = c("ALL","AML"))

wilcox.test(golub[1391,gol.fac=="ALL"],golub[808,gol.fac=="ALL"], paired = T,
alternative="two.sided")
```

Output)

```
Wilcoxon signed rank test with continuity
  correction

data:  golub[1391, gol.fac == "ALL"] and golub[808, gol.fac == "ALL"]
V = 62, p-value = 0.01242
alternative hypothesis: true location shift is not equal to 0
```

Problem 4 (20 points)

The data set “UCBAdmissions” in R contains admission decisions by gender at six departments of UC Berkeley. For this data set, carry out appropriate test for independence between the admission decision and gender for each of the departments.

What are your conclusions? Please submit your R script with the answer.

Answer)

```
data("UCBAdmissions")

for (i in 1:6) {

 dept_ad <- UCBAdmissions[,,i]

 value_t <- fisher.test(dept_ad)

 dept_name <- colnames(UCBAdmissions)[i+2]

 cat("Fisher test result for department", i, value_t$p.value,"\n")
```

}

Northeastern University
College of Science

Output)

Fisher test result for department 1 1.669189e-05
Fisher test result for department 2 0.6770899
Fisher test result for department 3 0.3866166
Fisher test result for department 4 0.5994965
Fisher test result for department 5 0.3603964
Fisher test result for department 6 0.5458408

Conclusion:

For department 1 we reject null hypothesis because the p-value is 1.669189e-05.
For department 2,3,4,5,6, we do not reject null hypothesis.

Problem 5 (20 points)

There are two random samples $X_1 \ldots X_n$ and $Y_1 \ldots Y_m$ with population means μ_X and μ_Y and population variances σ^2_X and σ^2_Y.

For testing H_0: $\sigma^2_X = \sigma^2_Y$ versus H_A: $\sigma^2_X < \sigma^2_Y$, we can use a permutation test for the statistic

$$S = \frac{s_X^2}{s_Y^2}.$$

Please program this permutation test in R. Use this nonparametric test on the "CD33" gene of the Golub et al. (1999) data set. Test whether the variance in the ALL group is smaller than the variance in the AML group. Please submit your R code with the answer.

Answer)

```
gol.fac <- factor(golub.cl,levels=0:1, labels= c("ALL","AML"))

dataALL <- golub[808,gol.fac=="ALL"]

dataAML <- golub[808,gol.fac=="AML"]
```

```
T.obs<- var(dataALL)/var(dataAML)

n.perm<-2000

T.perm<-numeric(n.perm)

for(i in 1:n.perm) {

  data.perm=sample(c(dataALL, dataAML), length(c(dataALL, dataAML)), replace
= FALSE)

  s1 <- var(data.perm[1:length(dataALL)])

  s2 <- var(data.perm[(length(dataALL)+1):(length(dataALL)+length(dataAML))])

  T.perm[i]<-s1/s2

}

mean(T.perm<=T.obs)
```

Output)

```
> mean(T.perm<=T.obs)

[1] 0.0415
```