Julie Garcia
October 24, 2017

Lab #6
Multiple testing

In this lab, we will be working with an Affymetrix data set that was run on the human HGU95A array. This experiment was designed to assess the gene expression events in the frontal cortex due to aging. A total of 18 male and 12 female postmortem brain samples were obtained to assess this.

The analysis that we are interested in conducting is a direct follow up to the previous lab of differential expression. We first want to identify those genes/probes that are differentially expressed in the frontal cortex between old and young subjects, then between males and females. Next, we would like to evaluate the differences between a couple of multiple testing adjustment methods. As explained in the lecture and the course website, multiple testing is a necessary step to reduce false positives when conducting more than a single statistical test. You will generate some p-value plots to get an idea of the how conservative some methods are compared to others.

I have identified 2 gene vectors for you to use below, so do not calculate the t-test or adjustments on the entire array of genes/probes.

For the second part of this lab, you will be working with RNA-sequencing data from The Cancer Genome Atlas (TCGA), specifically a breast invasive carcinoma dataset of 119 patient tumors. The data matrix and annotation files are on the course website. We will be trying to confirm an observation from a meta-analysis performed by Mehra et al, 2005 in Cancer Research. The authors identified the gene (using arrays) and protein (using immunohistochemistry) GATA3 as a prognostic factor in breast cancer, where patients with low expression of GATA3 experienced overall worse survival. The PubMed abstract is here: http://www.ncbi.nlm.nih.gov/pubmed/16357129.

1.) Download the GEO Brain Aging study from the class website. Also obtain the annotation file for this data frame.

2.) Load into R, using read.table() function and the header=T/row.names=1 arguments for each data file.

```
data = read.table("c:\\temp\\datasets\\agingStudy11FCortexAffy.txt",
header=T, row.names=1)
data[1:4,]

data.ann =
read.table("C:\\TEMP\\datasets\\agingStudy1FCortexAffyAnn.txt",
header=T)
data.ann[1:4,]
```

3.) Prepare 2 separate vectors for comparison. The first is a comparison between male and female patients. The current data frame can be left alone for this, since the males and females are all grouped together. The second vector is comparison between patients >= 50 years of age and those < 50 years of age.

To do this, you must use the annotation file and logical operators to isolate the correct arrays/samples.

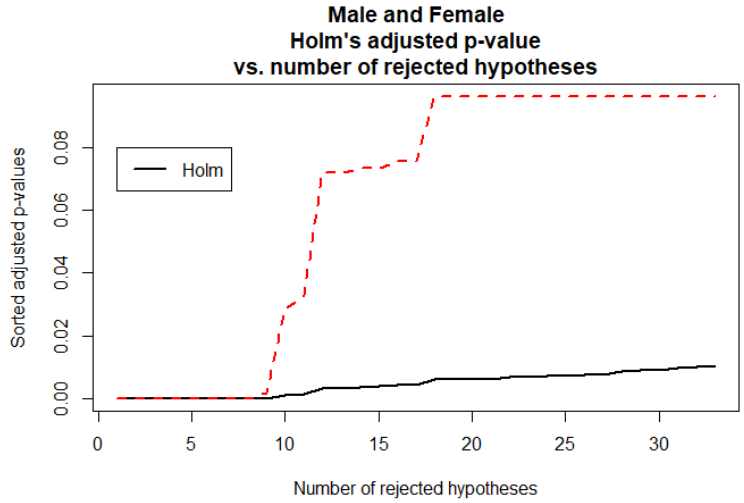
```
# get male and female vectors
g.g <- c(1394,  1474,  1917,  2099,  2367,  2428, 2625,  3168,  3181,
3641, 3832,  4526, 4731,  4863,  6062,  6356,  6684,  6787,  6900,
7223,  7244,  7299,  8086,  8652, 8959,  9073,  9145,  9389, 10219,
11238, 11669, 11674, 11793)
data.gender <- data[g.g,]
dim(data.gender)
data.gender.male <- data.gender[,1:18]
data.gender.female <- data.gender[,19:30]

#get age vectors
g.a <- c(25, 302,  1847,  2324,  246,  2757, 3222, 3675,  4429,  4430,
4912,  5640, 5835, 5856,  6803,  7229,  7833,  8133, 8579,  8822,
8994, 10101, 11433, 12039, 12353,
         12404, 12442, 67, 88, 100)
data.age <- data[g.a,]

data.ann <- data.ann[order(data.ann$Age),]
data.ann.50andup <- as.vector(data.ann[data.ann$Age >= 50, 1])
data.ann.under50 <- as.vector(data.ann[data.ann$Age < 50, 1])

colnames(data.age) <- lapply(colnames(data.age), function(x) {
strsplit(x, ".", fixed=TRUE)[[1]][1]})
data.age.50andup <- data.age[,data.ann.50andup]
data.age.under50 <- data.age[,data.ann.under50]

# check that we have the correct columns
colnames(data.age.50andup)
data.ann.50andup
colnames(data.age.under50)
data.ann.under50

data.age.sorted <- cbind(data.age.under50, data.age.50andup)
```

4.) Run the t.test function from the notes using the first gene vector below for the gender comparison. Then use the second gene vector below for the age comparison. Using these p-values, use either p.adjust in the base library or mt.rawp2adjp in the multtest library to adjust the values for multiple corrections with the Holm's method.

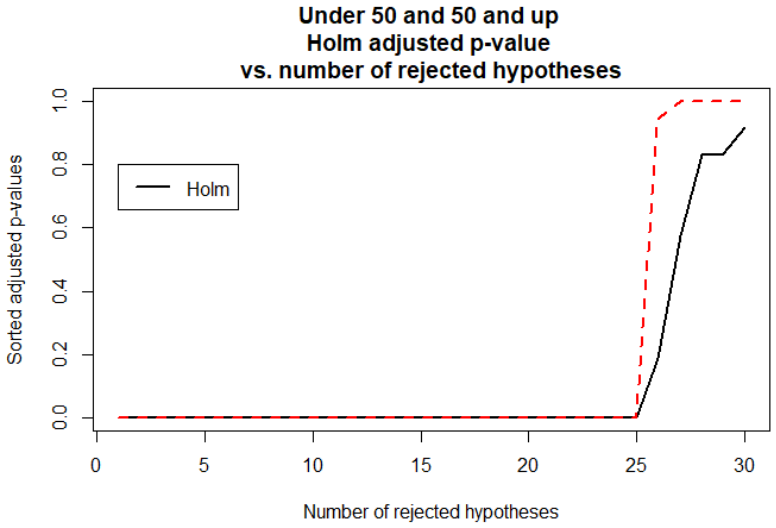
```
t.test.all.genes <- function(x,s1,s2) {
  x1 <- x[s1]
  x2 <- x[s2]
  x1 <- as.numeric(x1)
  x2 <- as.numeric(x2)
  t.out <- t.test(x1,x2, alternative='two.sided',var.equal=T)
  out <- as.numeric(t.out$p.value)
  return(out)
}

#data.gender and data.age already only have the genes from the
# vector below, I did this in #3
rawp.gender <- apply(data.gender,1, t.test.all.genes,s1=1:18,s2=19:30)
rawp.age <- apply(data.age.sorted, 1, t.test.all.genes, s1=1:12,
s2=13:30)

source("https://bioconductor.org/biocLite.R")
biocLite("multtest")
library(multtest)

help(mt.rawp2adjp)
adjustedp.gender <- mt.rawp2adjp(rawp.gender, proc="Holm")
adjustedp.age <- mt.rawp2adjp(rawp.age, proc="Holm")
```

5.) Sort the adjusted p-values and non-adjusted p-values and plot them vs. the x-axis of numbers for each comparison data set. Make sure that the two lines are different colors. Also make sure that the p-values are sorted before plotting.

```
#these already appear to be sorted when I got them back from rawp2adjp

mt.plot(adjustedp.gender$adjp,plottype="pvsr",proc=c("Holm"),
        lwd=2, leg=c(1,0.08))
title("Male and Female\n Holm's adjusted p-value \nvs. number of
rejected hypotheses")
```

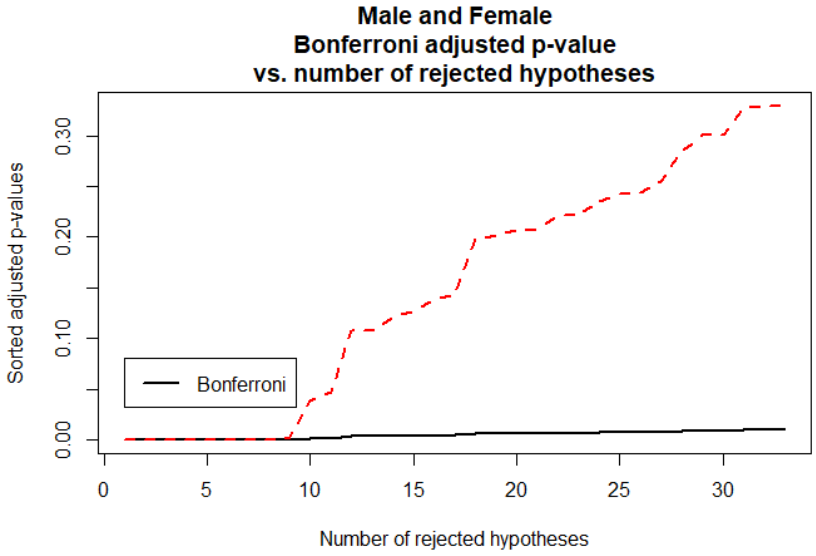


```
mt.plot(adjustedp.age$adjp,plottype="pvsr",proc=c("Holm"),
        lwd=2, leg=c(1,0.8))
title("Under 50 and 50 and up \nHolm adjusted p-value \nvs. number of
rejected hypotheses")
```

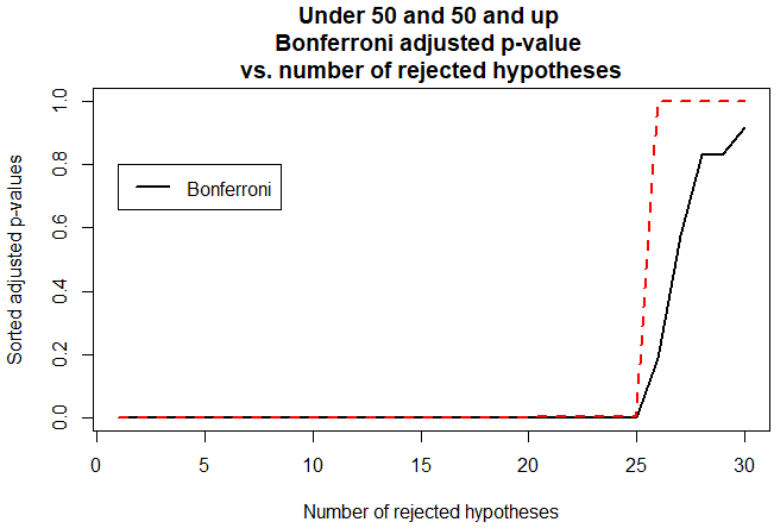



6.) Repeat #4 and #5 with the Bonferroni method.

```
mt.plot(bonf.gender$adjp,plottype="pvsr",proc=c("Bonferroni"),
        lwd=2, leg=c(1,0.08))
title("Male and Female\n Bonferroni adjusted p-value \nvs. number of
rejected hypotheses")
```



```
mt.plot(bonf.age$adjp,plottype="pvsr",proc=c("Bonferroni"),
        lwd=2, leg=c(1,0.8))
title("Under 50 and 50 and up \nBonferroni adjusted p-value \nvs.
number of rejected hypotheses")
```




7.) Read in the $\log_2$ normalized fragments per kb per million mapped reads (FPKM) data matrix and annotation files. This is RNA-sequencing data that has normalized read counts on a similar scale to microarray intensities.

```
data = read.table("c:\\temp\\datasets\\tcga_brca_fpkm.txt", header=T,
row.names=1)
data[1:4,1:4]

help(read.table)
data.ann = read.table("C:\\TEMP\\datasets\\tcga_brca_fpkm_sam.txt",
header=T, fill=T)
data.ann
```

8.) Use grep to subset the data matrix only by gene ‘GATA3’ and make sure to cast this vector to numeric.

```
help(grep)
grep("^*GATA3&?", rownames(data), ignore.case=FALSE)

[1] 6362

data.gata3 <- as.numeric(data[6362,])
```

9.) Create a binary (1/0) vector for the patients where the **upper** 25% expression of GATA3 is coded as 1 and all other patients are coded as 0. Call this new variable ‘group’.

```
length(data.gata3)
#get the number that each need to be, to be in the top 25%
twentyfive <- round(119/4)
topthreshold <- sort(data.gata3, decreasing=T)[twentyfive]
group <- lapply(data.gata3, function(x) { if(x > topthreshold) { return
(1);} else {return (0);} })
group <- as.numeric(group)
```

10.) Create a data matrix with the 'group' variable you created in #9 and the remaining variables in the annotation file.

```
data.ann.withgroup <- cbind(data.ann, group)
```

11.) Run a Kaplan-Meier (KM) analysis to determine if a difference in survival experience exists between the two GATA3 expression groups using the survdiff function. Extract the p-value from the chi squared test output.

```
set_status <- function(x) {
  if (is.na(x)) {return (NA)}

  if (x == "DECEASED") {return (1)}
  else {return (0)}
}

status <-
unlist(lapply(data.ann.withgroup[,"vital_status"],set_status))
time <- as.vector(data.ann.withgroup[,"months_to_event"])
x <- data.frame(status,time)
names(x) <- c("time","status")
library(survival)
survdiff(Surv(time, status) ~ group)

Call:
survdiff(formula = Surv(time, status) ~ group)

n=116, 3 observations deleted due to missingness.

          N Observed Expected (O-E)^2/E (O-E)^2/V
group=0 87       27     22.5     0.901      3.09
group=1 29        5      9.5     2.134      3.09

 Chisq= 3.1  on 1 degrees of freedom, p= 0.0788
```

12.) Now run a Cox proportion hazard (PH) regression model on just the grouping variable (i.e. no other covariates) and extract both the p-value and hazard ratio from the output.

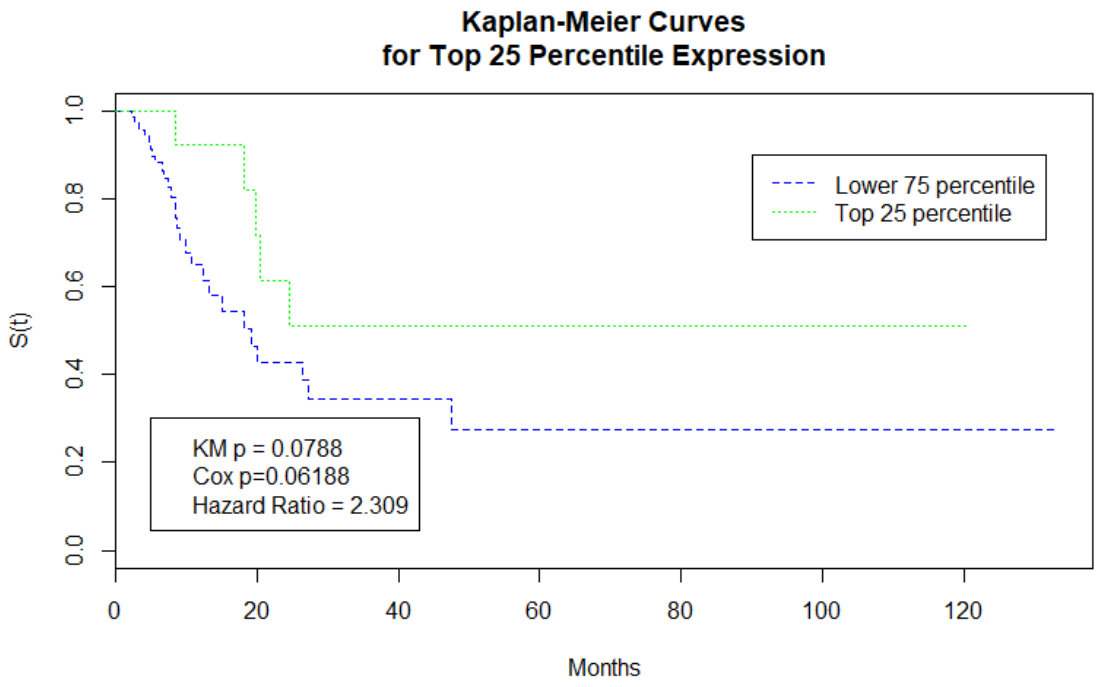
```
help(coxph)
fit <- coxph(Surv(time, status) ~ group)
summary(fit)
```

```
Call:
coxph(formula = Surv(time, status) ~ group)

  n= 116, number of events= 32
   (3 observations deleted due to missingness)

         coef exp(coef) se(coef)      z Pr(>|z|)
group -0.8369    0.4331   0.4896 -1.709   0.0874 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

      exp(coef) exp(-coef) lower .95 upper .95
group    0.4331      2.309    0.1659     1.131

Concordance= 0.593  (se = 0.045 )
Rsquare= 0.03   (max possible= 0.865 )
Likelihood ratio test= 3.49  on 1 df,   p=0.06188
Wald test            = 2.92  on 1 df,   p=0.08742
Score (logrank) test = 3.09  on 1 df,   p=0.07875
```

So the group in the lower 75 percentile of expression has a hazard ratio of 2.309, while the top 25 percentile of expression have a hazard ratio of 0.4331.

13.) Run the survfit() function only on the grouping variable (i.e. no other covariates) and plot the KM curves, being sure to label the two groups with a legend, two different colors for each line, and provide the KM p-value, Cox PH p-value, Cox PH hazard ratio, and sample sizes all in each of the two groups all on the plot.

```
f<-survfit(Surv(time, status) ~ group,type="kaplan-meier")
colors = c("blue", "green")
plot(f, lty = 2:3, xlab="Months", ylab="S(t)", col=colors)
legend(90, .9, c("Lower 75 percentile", "Top 25 percentile"), lty =
2:3, col=colors)
legend(5,0.3, c("KM p = 0.0788", "Cox p=0.06188", "Hazard Ratio =
2.309"), )
title("Kaplan-Meier Curves\nfor Top 25 Percentile Expression")
```




14.) Does this result agree with the Mehra et al, study result?

My Kaplan-Meier plot does resemble the plot in the study where survival is decreased over time with lower expression of GATA3.

Gene Vectors (indices for specific rows/genes)

# gender comparison gene vector

g.g <- c(1394, 1474, 1917, 2099, 2367, 2428, 2625, 3168, 3181, 3641, 3832, 4526, 4731, 4863, 6062, 6356, 6684, 6787, 6900, 7223, 7244, 7299, 8086, 8652, 8959, 9073, 9145, 9389, 10219, 11238, 11669, 11674, 11793)

# age comparison gene vector

g.a <- c(25, 302, 1847, 2324, 246, 2757, 3222, 3675, 4429, 4430, 4912, 5640, 5835, 5856, 6803, 7229, 7833, 8133, 8579, 8822, 8994, 10101, 11433, 12039, 12353, 12404, 12442, 67, 88, 100)