Jaimee Beckett
Lab #6

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.

Done

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

```
dat <- read.table('agingStudy11FCortexAffy.txt', header=T, row.names=1)
ann <- read.table('agingStudy1FCortexAffyAnn.txt', header=T, row.names=1)
```

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.

```
male <- dat[,1:18]
female <- dat[19:30]
```

```
cl.over <- paste(rownames(ann[ann$Age>=50,]), ann[ann$Age>=50,2],
                 ann[ann$Age>=50,1], sep='.')
cl.under <- paste(rownames(ann[ann$Age<50,]), ann[ann$Age<50,2],
                  ann[ann$Age<50,1], sep='.')
```

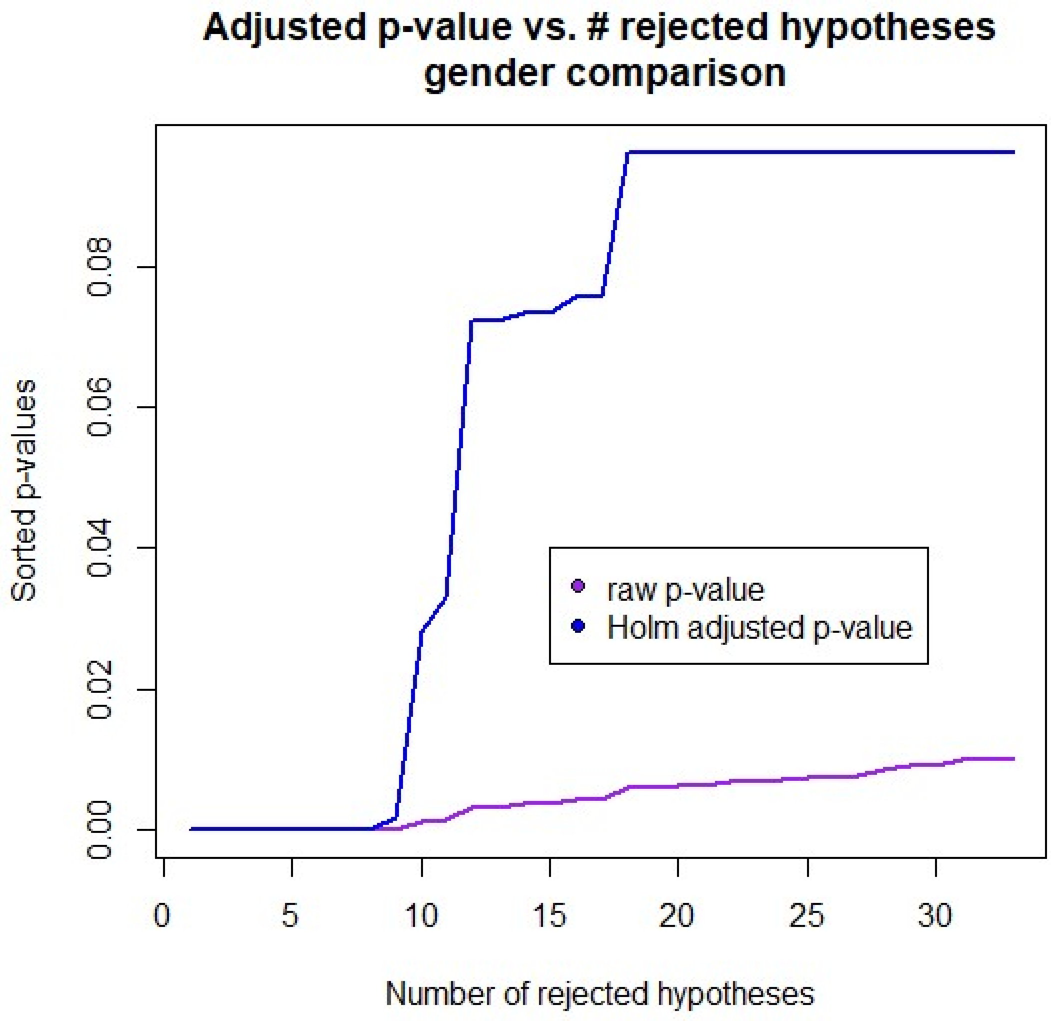
4.) Run the t.test function from the notes using the first gene vector below for the gender comparison. Then use the second gene vector below for the age comparison. Using these p-values, use either p.adjust in the base library or mt.rawp2adjp in the multtest library to adjust the values for multiple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

```
#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)

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)
}
g.rawp <- apply(dat[g.g,], 1, t.test.all.genes, s1=c(1:18), s2=c(19:30))
a.rawp <- apply(dat[g.a,], 1, t.test.all.genes, s1=cl.over, s2=cl.under)

#with p.adjust
library(base)
g.p.adj <- p.adjust(g.rawp, method="holm")
a.p.adj <- p.adjust(a.rawp, method="holm")
```

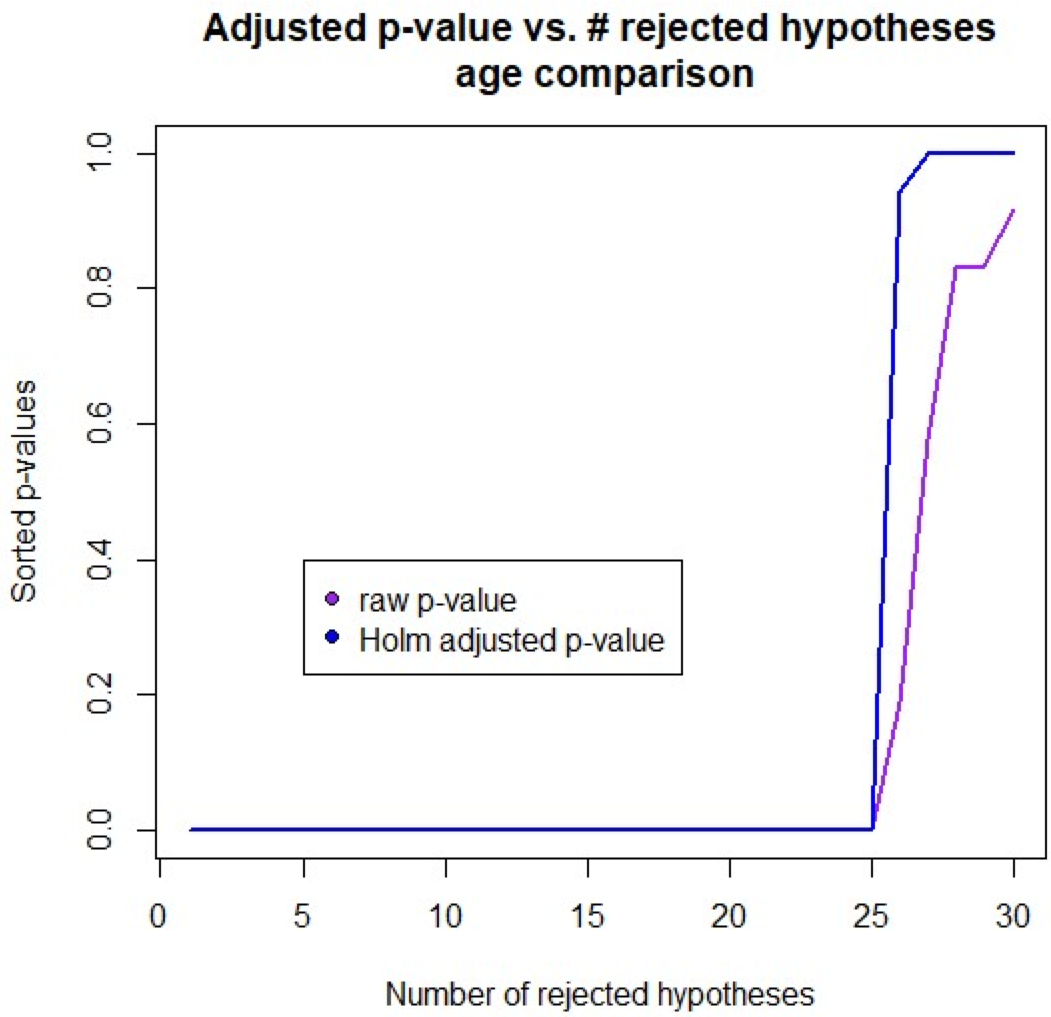
5.) Sort the adjusted p-values and non-adjusted p-values and plot them vs. the x-axis of numbers for each comparison data set. Make sure that the two lines are different colors. Also make sure that the p-values are sorted before plotting.

```
g.rawp.sorted <- sort(g.rawp)
a.rawp.sorted <- sort(a.rawp)
g.p.adj.sorted <- sort(g.p.adj)
a.p.adj.sorted <- sort(a.p.adj)

#Gender plot
plot(c(1,length(g.rawp)), range(as.numeric(g.p.adj)), type="n",
     main="Adjusted p-value vs. # rejected hypotheses\n gender comparison",
     ylab="Sorted p-values",xlab="Number of rejected hypotheses")
lines(c(1:length(g.rawp)), as.numeric(g.rawp.sorted), col="purple", lwd=2)
lines(c(1:length(g.p.adj)), as.numeric(g.p.adj.sorted), col="blue", lwd=2)
```

```
legend(20, 0.04, pch=21, col=1, pt.bg=c("purple", "blue"),
       c("raw p-value", "Holm adjusted p-value"))
```

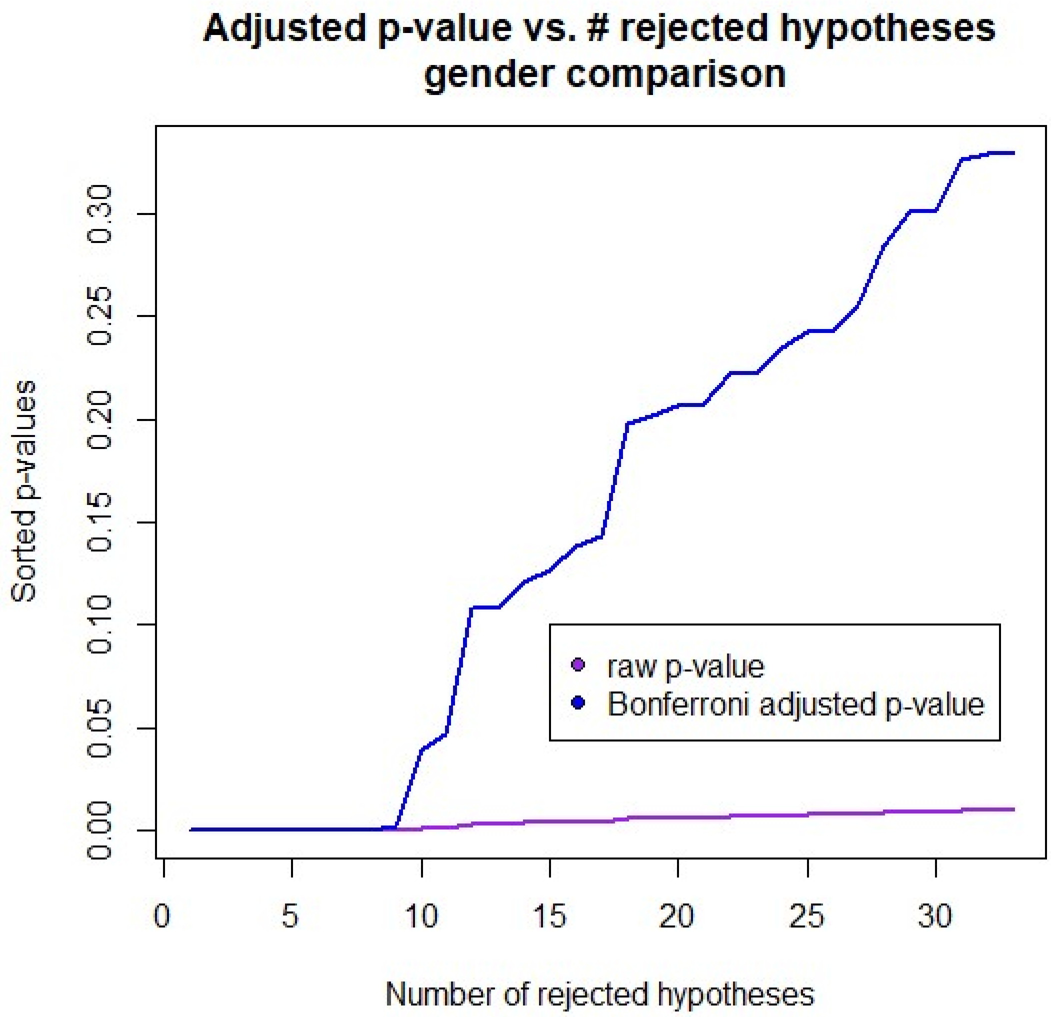


```
#Age plot
plot(c(1,length(a.rawp)), range(as.numeric(a.p.adj)), type="n",
     main="Adjusted p-value vs. # rejected hypotheses\n age comparison",
     ylab="Sorted p-values",xlab="Number of rejected hypotheses")
lines(c(1:length(a.rawp)), as.numeric(a.rawp.sorted), col="purple", lwd=2)
lines(c(1:length(a.p.adj)), as.numeric(a.p.adj.sorted), col="blue", lwd=2)
legend(5,0.4, pch=21, col=1, pt.bg=c("purple", "blue"),
       c("raw p-value", "Holm adjusted p-value"))
```

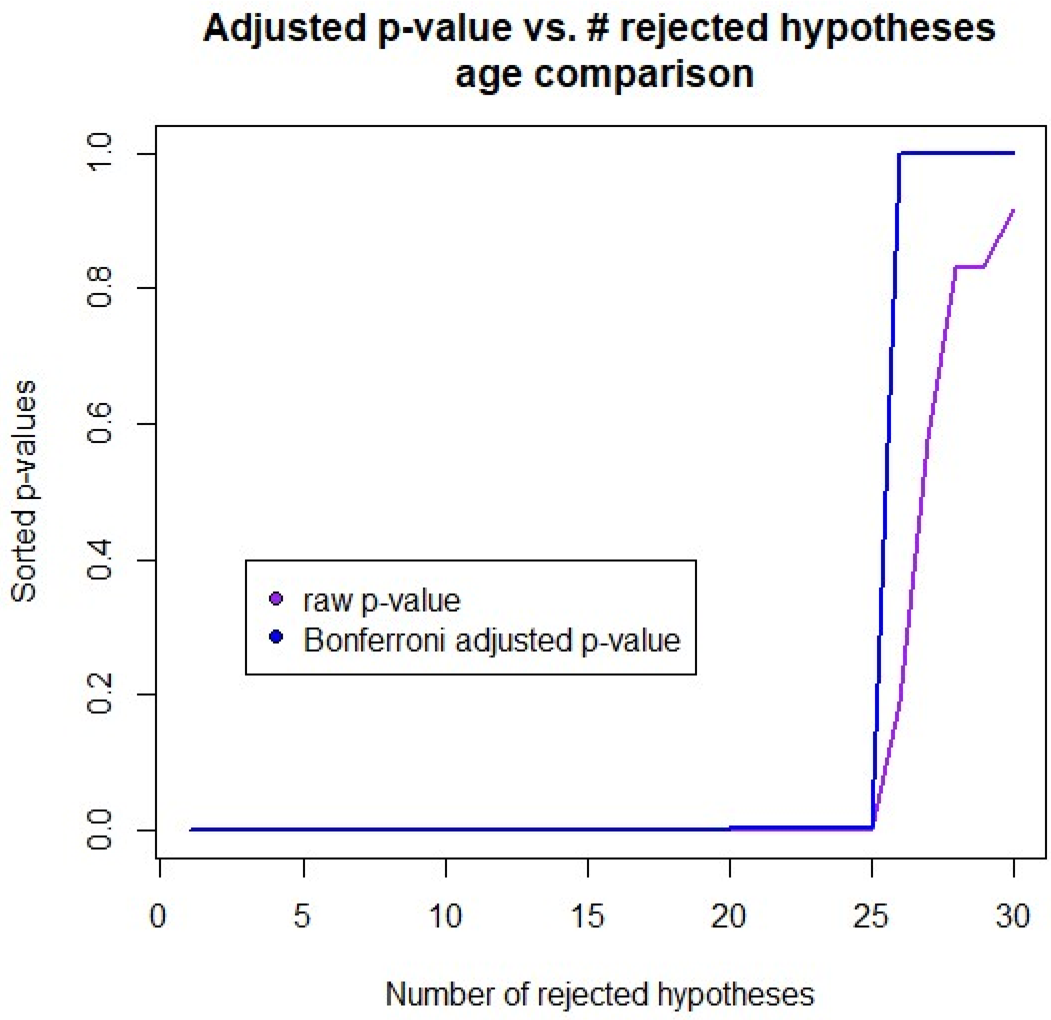


6.) Repeat #4 and #5 with the Bonferroni method.

```
g.p.adj.bonferroni <- p.adjust(g.rawp, method="bonferroni")
a.p.adj.bonferroni <- p.adjust(a.rawp, method="bonferroni")

g.p.adj.bonferroni.sorted <- sort(g.p.adj.bonferroni)
a.p.adj.bonferroni.sorted <- sort(a.p.adj.bonferroni)

#Gender plot
plot(c(1,length(g.rawp)), range(as.numeric(g.p.adj.bonferroni)), type="n",
     main="Adjusted p-value vs. # rejected hypotheses\n gender comparison",
     ylab="Sorted p-values",xlab="Number of rejected hypotheses")
lines(c(1:length(g.rawp)), as.numeric(g.rawp.sorted), col="purple", lwd=2)
lines(c(1:length(g.p.adj.bonferroni)), as.numeric(g.p.adj.bonferroni.sorted),
      col="blue", lwd=2)
legend(13, 0.10, pch=21, col=1, pt.bg=c("purple", "blue"),
       c("raw p-value", "Bonferroni adjusted p-value"))
```



```
#Age plot
plot(c(1,length(a.rawp)), range(as.numeric(a.p.adj.bonferroni)), type="n",
     main="Adjusted p-value vs. # rejected hypotheses\n age comparison",
     ylab="Sorted p-values",xlab="Number of rejected hypotheses")
lines(c(1:length(a.rawp)), as.numeric(a.rawp.sorted), col="purple", lwd=2)
lines(c(1:length(a.p.adj.bonferroni)), as.numeric(a.p.adj.bonferroni.sorted),
      col="blue", lwd=2)
legend(3,0.4, pch=21, col=1, pt.bg=c("purple", "blue"),
       c("raw p-value", "Bonferroni adjusted p-value"))
```



7.) Read in the log2 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.

```
dat.tcga <- read.table('tcga_brca_fpkm.txt', header=T, row.names = 1)
ann.tcga <- read.table("tcga_brca_fpkm_sam.txt", header=T, row.names=1, fill=T)
```

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

```
gata3 <- as.numeric(dat.tcga[grep("GATA3", rownames(dat.tcga)),])
```

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'.

```
percentile.75 <- quantile(gata3, probs = 0.75)
group <- gata3 >= percentile.75
group[group == TRUE] <- 1
```

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

```
gata3.mat <- cbind(as.matrix(ann.tcga),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.

```
library(survival)
time <- as.numeric(gata3.mat[,"months_to_event"])
status <- gata3.mat[,"vital_status"] == "LIVING"
status[status == TRUE] <- 1
survdiff(Surv(time,status)~group, data = as.data.frame(gata3.mat))

## Call:
## survdiff(formula = Surv(time, status) ~ group, data = as.data.frame(gata3.mat
))
##
## n=116, 3 observations deleted due to missingness.
##
##          N Observed Expected (O-E)^2/E (O-E)^2/V
## group=0 86       59     61.6     0.113     0.435
## group=1 30       25     22.4     0.310     0.435
##
##  Chisq= 0.4  on 1 degrees of freedom, p= 0.5
```

The p-value is 0.5

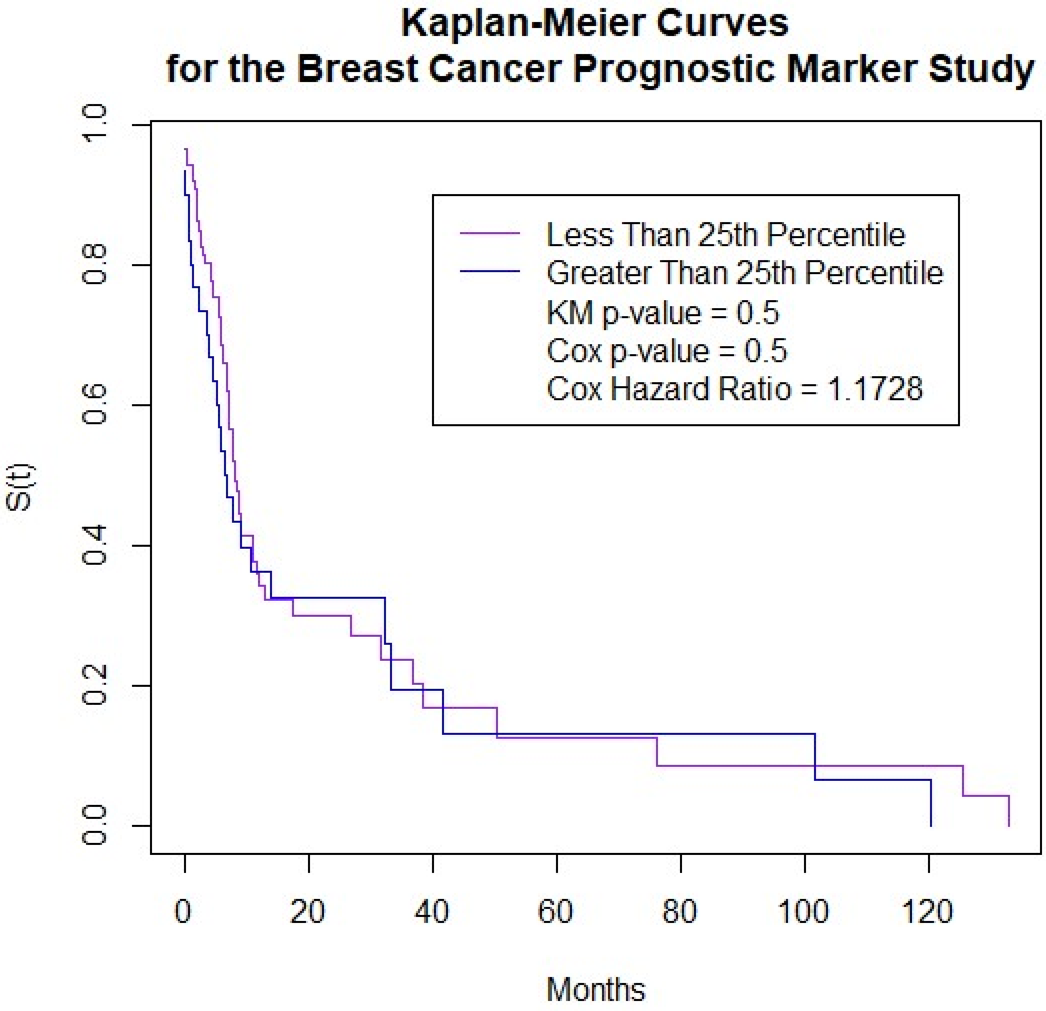
12.) Now run a Cox proportion hazard (PH) regression model on just the grouping variable (i.e. no other covariates) and extract both the p-value and hazard ratio from the output.

```
fit <- coxph( Surv(time,status)~group, data = as.data.frame(gata3.mat))
summary(fit)

## Call:
## coxph(formula = Surv(time, status) ~ group, data = as.data.frame(gata3.mat))
##
##   n= 116, number of events= 84
##    (3 observations deleted due to missingness)
##
##          coef exp(coef) se(coef)     z Pr(>|z|)
## group1 0.1594    1.1728   0.2418 0.659     0.51
##
##        exp(coef) exp(-coef) lower .95 upper .95
## group1     1.173     0.8527    0.7302     1.884
##
## Concordance= 0.531  (se = 0.03 )
## Likelihood ratio test= 0.43  on 1 df,   p=0.5
## Wald test            = 0.43  on 1 df,   p=0.5
## Score (logrank) test = 0.44  on 1 df,   p=0.5
```

The p-value is 0.5 The hazard ratio is 1.1728

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.

```r
f <- survfit(Surv(time,status)~group, type="kaplan-meier",
             data = as.data.frame(gata3.mat))
plot(f, col=c("purple", "blue"), xlab="Months", ylab="S(t)")
legend(40, 0.9, c("Less Than 25th Percentile", "Greater Than 25th Percentile",
                  "KM p-value = 0.5", "Cox p-value = 0.5",
                  "Cox Hazard Ratio = 1.1728"), lty=1, col=c("purple","blue",
                                                             NA, NA, NA))
title("Kaplan-Meier Curves\n for the Breast Cancer Prognostic Marker Study")
```



14.) Do the results agree with the Mehra et al, study result?

The results do not agree with the overall results of the Mehra et al, study, but there is a p-value of 0.5. The previous study took data from four previous studies and combined them, some of which also had high p-values.