BIOS 7659 Homework 3

Tim Vigers

13 October 2020

1. T-statistics

Read in the data:

```
array <- read.table("./hw3data/hw3arraydata.txt")
gene_names <- read.table("./hw3data/hw3genenames.txt",
                         blank.lines.skip = FALSE)
```

a) Fold change

For each gene (row), find the mean $\log_2$ expression among controls and among the knock out group. Then calculate fold change using $\log_2(controls) - \log_2(knockouts)$:

```
fc <- apply(array,1,function(x){
  control = mean(as.numeric(x[1:8]))
  knockout = mean(as.numeric(x[9:16]))
  return(c(control,knockout,control-knockout))
})
fc <- t(fc)
fc_results <- as.data.frame(cbind(gene_names,fc))
colnames(fc_results) <-
  c("Gene","Control mean","Knockout mean","log2FC")
kable(head(fc_results[order(abs(fc_results$log2FC),
                            decreasing = T),],10),
      caption = "Top 10 genes with largest absolute value of fold change",
      row.names = F)
```

Table 1: Top 10 genes with largest absolute value of fold change

Gene	Control mean	Knockout mean	log2FC
ApoAI,lipid-Img	8.147709	3.398463	4.749247
EST,HighlysimilartoA	8.245822	3.672996	4.572826
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.772249
EST,WeaklysimilartoC	7.588145	6.047714	1.540431
ESTs,Highlysimilarto	7.835267	6.320549	1.514718
est	7.704867	6.238731	1.466135
similartoyeaststerol	7.356597	5.924143	1.432454
ApoCIII,lipid-Img	7.781127	6.382253	1.398874
psoriasis-associated	7.742241	6.485528	1.256714
Cy3RT	6.889612	8.082898	-1.193286

b) Standard t test

For each gene, calculate the two-sample independent t-statistic (not assuming equal variances) for the comparison between controls and knockouts:

```
# Tests
tp <- apply(array,1,function(x){
  control = as.numeric(x[1:8])
  knockout = as.numeric(x[9:16])
  t <- t.test(control,knockout)
  return(c(t$statistic,t$p.value))
})
# Format results
tp <- as.data.frame(t(tp))
colnames(tp) <- c("T","p value")
fc_results <- as.data.frame(cbind(fc_results,tp))
kable(head(fc_results[order(abs(fc_results$T),decreasing = T),],10),
      caption = "Top 10 genes with largest t-statistic",
      row.names = F)
```

Table 2: Top 10 genes with largest t-statistic

Gene	Control mean	Knockout mean	log2FC	T	p value
ApoAI,lipid-Img	8.147709	3.398463	4.7492467	23.104347	0.0000000
EST,WeaklysimilartoC	7.588145	6.047714	1.5404305	12.982368	0.0000000
EST,HighlysimilartoA	8.245822	3.672996	4.5728257	11.762486	0.0000019
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.7722489	11.759068	0.0000000
ApoCIII,lipid-Img	7.781127	6.382253	1.3988735	10.430072	0.0000020
est	7.704867	6.238731	1.4661354	9.087422	0.0000031
ESTs,Highlysimilarto	7.835267	6.320549	1.5147176	9.018613	0.0000061
similartoyeaststerol	7.356597	5.924143	1.4324539	7.208906	0.0000123
Caspase7,heart-Img	8.011684	7.558373	0.4533114	4.578842	0.0005343
EST,WeaklysimilartoF	7.945457	7.089572	0.8558850	4.434296	0.0007886

Out of the 6384 genes, 75 were significant at the $p < 0.01$ level.

c) Alternative t-statistics

i) Modified t-statistic (using the samr package)

```
y <- ifelse(grepl("c",colnames(array)),1,2)
x <- as.matrix(array)
data=list(x=x,y=y)
samr_obj <- samr(data)
```

```
samr_pvalues <- samr.pvalues.from.perms(samr_obj$tt,samr_obj$ttstar)
samr_results <- cbind(gene_names,samr_obj$tt,samr_pvalues)
colnames(samr_results) <- c("Gene","Modified t-statistic","p value")
# P values
kable(head(samr_results[order(abs(samr_results[,2]),
                              decreasing = T),],10),
      caption = "Top 10 genes with largest modified t-statistic",
      row.names = F)
```

Table 3: Top 10 genes with largest modified t-statistic

Gene	Modified t-statistic	p value
ApoAI,lipid-Img	-20.592874	0.0001566
EST,HighlysimilartoA	-11.049934	0.0001566
EST,WeaklysimilartoC	-10.717909	0.0001566
CATECHOLO-METHYLTRAN	-10.628833	0.0001566
ApoCIII,lipid-Img	-8.787524	0.0001566
est	-7.865276	0.0001566
ESTs,Highlysimilarto	-7.847305	0.0001566
similartoyeaststerol	-6.401300	0.0001598
EST,WeaklysimilartoF	-3.924562	0.0004464
Caspase7,heart-Img	-3.653656	0.0006986

Based on the modified t-statistic, there are 94 genes that are significantly different at the 0.01 level.

ii) Moderated t-statistic (using the `limma` package)

First, create the design matrix for limma:

```
design <- matrix(ncol = 2,nrow = ncol(array))
colnames(design) <- c("Control","Knockout")
rownames(design) <- colnames(array)
design[,1] <- rep(1,nrow(design))
design[,2] <- ifelse(grepl("k",rownames(design)),1,0)
```

Fit the model with limma:

```
fit <- lmFit(array, design)
eb <- eBayes(fit)
limma_res <- topTable(eb,coef = 2,number = 10)
rownames(limma_res) <- gene_names$V1[as.numeric(rownames(limma_res))]
kable(limma_res,
      caption = "Top 10 differentially expressed genes (based on the moderated t-statistic)")
```

Table 4: Top 10 differentially expressed genes (based on the moderated t-statistic)

	logFC	AveExpr	t	P.Value	adj.P.Val	B
ApoAI,lipid-Img	-4.749247	5.773086	-23.976817	0.0000000	0.0000000	14.9269328
EST,HighlysimilartoA	-4.572826	5.959409	-12.963071	0.0000000	0.0000005	10.8150265
CATECHOLO-METHYLTRAN	-2.772249	6.617134	-12.439908	0.0000000	0.0000006	10.4483231
EST,WeaklysimilartoC	-1.540431	6.817930	-11.749992	0.0000000	0.0000012	9.9246200
ApoCIII,lipid-Img	-1.398874	7.081690	-9.831229	0.0000000	0.0000157	8.1890866
ESTs,Highlysimilarto	-1.514718	7.077908	-9.012972	0.0000000	0.0000423	7.3031534
est	-1.466135	6.971799	-8.999811	0.0000000	0.0000423	7.2881051
similartoyeaststerol	-1.432454	6.640370	-7.440210	0.0000007	0.0005617	5.3097967
EST,WeaklysimilartoF	-0.855885	7.517514	-4.553948	0.0002495	0.1769590	0.5618636
	-0.549536	7.325818	-3.961031	0.0009254	0.5284860	-0.5563623

Based on the moderated t-statistic, there are 93 genes that are significantly different at the 0.01 level (without additional adjustment for multiple comparisons).

d) Method comparisons

Generally speaking, the four methods are pretty similar, at least in terms of ranking the top ten differentially expressed genes. The ranked order is not exactly the same for each method, but the same 10 genes are chosen regardless. However, the modified and moderated t-statistic approaches reject more null hypotheses than the standard t-statistic. One potential downside to the `samr` approach is that the number of significant genes changes slightly depending on the random seed and the number of permutations.

The standard t-statistic (assuming independent samples with unequal variances) for a gene g is calculated using the formula:

$$t_g = \frac{\hat{\mu}_{g1} - \hat{\mu}_{g2}}{\sqrt{\frac{s_{g1}^2}{n_1} + \frac{s_{g2}^2}{n_2}}}$$

where $\hat{\mu}$, s, and n represent the mean, sample variance, and number of samples, respectively, for groups 1 and 2. This generally works well for larger sample sizes, but the standard error (SE) estimates are unreliable with smaller samples as is often the case in gene expression studies. This can lead to artificially inflated t-statistics. The modified and moderated t-statistics try to address this issue in two different ways.

The modified t-statistic from the `samr` package adds a constant amount to the SE estimate (based on pooled variance) for each gene. This is often the α^{th} percentile of SE across all genes, but this can be altered. So, the formula is:

$$t_g = \frac{\hat{\mu}_{g1} - \hat{\mu}_{g2}}{s_g + s_0}$$

Where s_g represents the standard error estimate for gene g and s_0 is the added constant. This approach is not grounded in distributional theory, and is considered more of an ad-hoc approach.

The moderated t-statistic calculated by the `limma` package also aims to estimate a more stable SE, but uses a more complex Bayesian approach instead of simply adding a constant. The statistic is calculated as:

$$\tilde{t}_{gj} = \frac{\hat{\beta}_{gj}}{\tilde{s}_g \sqrt{\nu_{gj}}}$$

Where $\tilde{s}_g$ is a shrunken variance estimate that depends on hyperparameters chosen by sharing information across all genes. There is some complex theory behind this method, but the essential idea is that all of the information across genes contributes to the calculation of the t-statistic, which results in more stable SE estimates.

P Values and Multiple Testing

a) Permutation tests

First, find all the possible combinations of group labels (in this case control vs. knockout) using the `combinations()` functions. Then, for each gene calculate a t-statistic for each possible combination and count the number of permuted t-statistics that are larger than the "true" statistic. The proportion of permuted t-statistics greater than or equal to the "true" statistic is the permutation-based p value.

```
combos <- combinations(16,8,colnames(array))
cores <- detectCores()
cl <- makeCluster(cores,type = "FORK")
pvalues <- parApply(cl,array,1,function(g){
  maxt <- t.test(g[grep("c",names(array))],
                 g[grep("k",names(array))])
  perms <- apply(combos,1,function(c){
    control <- g[c]
    knockout <- g[setdiff(names(g),c)]
```

```
    t <- t.test(control,knockout)
    return(t$statistic)
  })
  return(sum(abs(perms)>=abs(maxt$statistic))/length(perms))
})
stopCluster(cl)
```

Using the permutation test approach, there are 117 genes significant at the 0.01 level.

*Note: I know that the homework sheet says not to use parallel computing methods, but I was curious about why this is and decided to test it. Using `parApply()` from the `parallel` package produced exactly the same results as using `apply()` and was approximately four times faster.

b) P value adjustment methods

i) Bonferroni

The Bonferroni correction rejects the null hypothesis for each p_i when $p_i \leq \frac{\alpha}{m}$, where m is the total number of null hypotheses. We have a total of 6384 tests, so will reject the null hypothesis when $p_i \leq \frac{0.01}{6384}$.

```
a_bonf <- 0.01 / nrow(array)
kable(fc_results[fc_results$`p value`<=a_bonf,],
      caption = "Significant genes after Bonferroni correction",
      row.names = F)
```

Table 5: Significant genes after Bonferroni correction

Gene	Control mean	Knockout mean	log2FC	T	p value
ApoAI,lipid-Img	8.147709	3.398463	4.749247	23.10435	0
EST,WeaklysimilartoC	7.588145	6.047714	1.540431	12.98237	0
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.772249	11.75907	0

After Bonferroni correction, we reject the null hypothesis for 3 genes.

ii) Sidak

The Sidak correction rejects the null hypothesis for each p_i when $p_i \leq 1-(1-\alpha)^{\frac{1}{m}}$, where again m is the total number of null hypotheses. We have a total of 6384 tests, so will reject the null hypothesis when $p_i \leq 1-(1-\alpha)^{\frac{1}{6384}}$.

```
a_sid <- 1-(1-0.01)^(1/nrow(array))
kable(fc_results[fc_results$`p value`<=a_sid,],
      caption = "Significant genes after Sidak correction",
      row.names = F)
```

Table 6: Significant genes after Sidak correction

Gene	Control mean	Knockout mean	log2FC	T	p value
ApoAI,lipid-Img	8.147709	3.398463	4.749247	23.10435	0
EST,WeaklysimilartoC	7.588145	6.047714	1.540431	12.98237	0
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.772249	11.75907	0

After Sidak correction, we reject the null hypothesis for 3 genes.

iii) Holm step-down

The Holm step-down procedure is as follows:

1. Rank p-values: $p_{(1)} \leq p_{(2)} ... \leq p_{(j)} \leq ... \leq p_{(m)}$.
2. Find the first j^* such that $p_{(j)} > \frac{\alpha}{m-j+1}$.
3. Reject all null hypotheses up to j^*.

```
ordered <- fc_results[order(fc_results$`p value`),]
m <- nrow(ordered)
j <- 1:m
jstar <- min(which((ordered$`p value` > 0.01/(m+1-j))==T))
kable(ordered[1:(jstar-1),],
      caption = "Significant genes after Holm correction",
      row.names = F)
```

Table 7: Significant genes after Holm correction

Gene	Control mean	Knockout mean	log2FC	T	p value
ApoAI,lipid-Img	8.147709	3.398463	4.749247	23.10435	0
EST,WeaklysimilartoC	7.588145	6.047714	1.540431	12.98237	0
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.772249	11.75907	0

After Holm step-down correction, we reject the null hypothesis for 3 genes.

iv) Benjamini-Hochberg

The Benjamini-Hochberg step-up procedure is as follows:

1. Rank p-values: $p_{(1)} \leq p_{(2)} ... \leq p_{(j)} \leq ... \leq p_{(m)}$.
2. Find the maximum j^* such that $p_{(j)} \leq \frac{j}{m}q$ where q is the desired false discovery rate.
3. Reject all null hypotheses through j^*.

```
q <- 0.01
jstar <- max(which((ordered$`p value` <= (j/m)*q)==T))
kable(ordered[1:jstar,],
      caption = "Significant genes after Benjamini-Hochberg correction",
      row.names = F)
```

Table 8: Significant genes after Benjamini-Hochberg correction

Gene	Control mean	Knockout mean	log2FC	T	p value
ApoAI,lipid-Img	8.147709	3.398463	4.749247	23.104347	0.00e+00
EST,WeaklysimilartoC	7.588145	6.047714	1.540431	12.982368	0.00e+00
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.772249	11.759068	0.00e+00
EST,HighlysimilartoA	8.245822	3.672996	4.572826	11.762486	1.90e-06
ApoCIII,lipid-Img	7.781127	6.382253	1.398874	10.430072	2.00e-06
est	7.704867	6.238731	1.466135	9.087422	3.10e-06
ESTs,Highlysimilarto	7.835267	6.320549	1.514718	9.018613	6.10e-06
similartoyeaststerol	7.356597	5.924143	1.432454	7.208906	1.23e-05

After Benjamini-Hochberg step-up correction, we reject the null hypothesis for 8 genes.

Comparison

The Bonferroni, Sidak, and Holm methods above are more conservative than the Benjamini-Hochberg approach, because they aim to control the family-wise error rate (FWER), or the probability of making at least 1 type 1 error. In other words, these methods ensure that $1 - (1 - p)^m \leq \alpha$. Of these methods, the Bonferroni approach is the most conservative, because it is a single step procedure and all tests are subject to the same stringent bound.

On the other hand, the Benjamini-Hochberg approach aims to control the false discovery rat (FDR), or the expected proportion of false positives among rejected hypotheses. Because FDR-based approaches focus on limiting type 1 error among "discoveries" (significant p values) as opposed to across all tests, they tend to be less conservative. This is why the Benjamini-Hochberg step-up correction rejects 8 null hypotheses compared to 3 for the more conservative methods.

c) Q-values

Calculate q-values using the `qvalue` package (without a pre-specified π_0 parameter):

```
q <- qvalue(fc_results$`p value`)
fc_results$qvalue <- q$qvalues
kable(fc_results[fc_results$qvalue<=0.01,],
      caption = "Significant genes based on q-value",
      row.names = F)
```

Table 9: Significant genes based on q-value

Gene	Control mean	Knockout mean	log2FC	T	p value	qvalue
EST,HighlysimilartoA	8.245822	3.672996	4.572826	11.762486	1.90e-06	0.0021906
est	7.704867	6.238731	1.466135	9.087422	3.10e-06	0.0027982
ApoCIII,lipid-Img	7.781127	6.382253	1.398874	10.430072	2.00e-06	0.0021906
ApoAI,lipid-Img	8.147709	3.398463	4.749247	23.104347	0.00e+00	0.0000020
ESTs,Highlysimilarto	7.835267	6.320549	1.514718	9.018613	6.10e-06	0.0047561
EST,WeaklysimilartoC	7.588145	6.047714	1.540431	12.982368	0.00e+00	0.0000197
similartoyeaststerol	7.356597	5.924143	1.432454	7.208906	1.23e-05	0.0084502
CATECHOLO-METHYLTRAN	8.003259	5.231010	2.772249	11.759068	0.00e+00	0.0000222

There are 8 genes with a q-value ≤ 0.01. π_0 represents the proportion of truly null hypotheses, and this package estimates it at approximately 0.859.