BB512/BB612 - Homework II

Apr 7, 2022

Use the `montpick` eset to perform the required analyses:

```
library(Biobase)
```

```
## Loading required package: BiocGenerics
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which.max, which.min
```

```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
con <- url("http://bowtie-bio.sourceforge.net/recount/ExpressionSets/montpick_eset.RData")
load(file = con)
close(con)

pdata <- pData(montpick.eset)
edata <- exprs(montpick.eset)
fdata <- fData(montpick.eset)
```

1. [20 pt] Transform, remove lowly expressed genes, and normalize the expression data

```
edata <- log2(as.matrix(edata) + 1)
edata <- edata[rowMeans(edata) > 10, ]

norm_edata <- preprocessCore::normalize.quantiles(edata)
colnames(norm_edata) <- colnames(edata)
rownames(norm_edata) <- rownames(edata)
```

2. [30 pt] Perform t-tests to compare the expressions of all genes between the two populations ("CEU" and "YRI"). Obtain the p-values. Adjust the p-values. How many genes have FDR < 0.1?

```r
idxA <- which(colnames(norm_edata) %in% pdata$sample.id[pdata$population == "CEU"])
idxB <- which(colnames(norm_edata) %in% pdata$sample.id[pdata$population == "YRI"])
p_values_t <- apply(norm_edata, 1, function(x) t.test(x[idxA], x[idxB])$p.value)

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

sum(adj_p_t < 0.1)
```

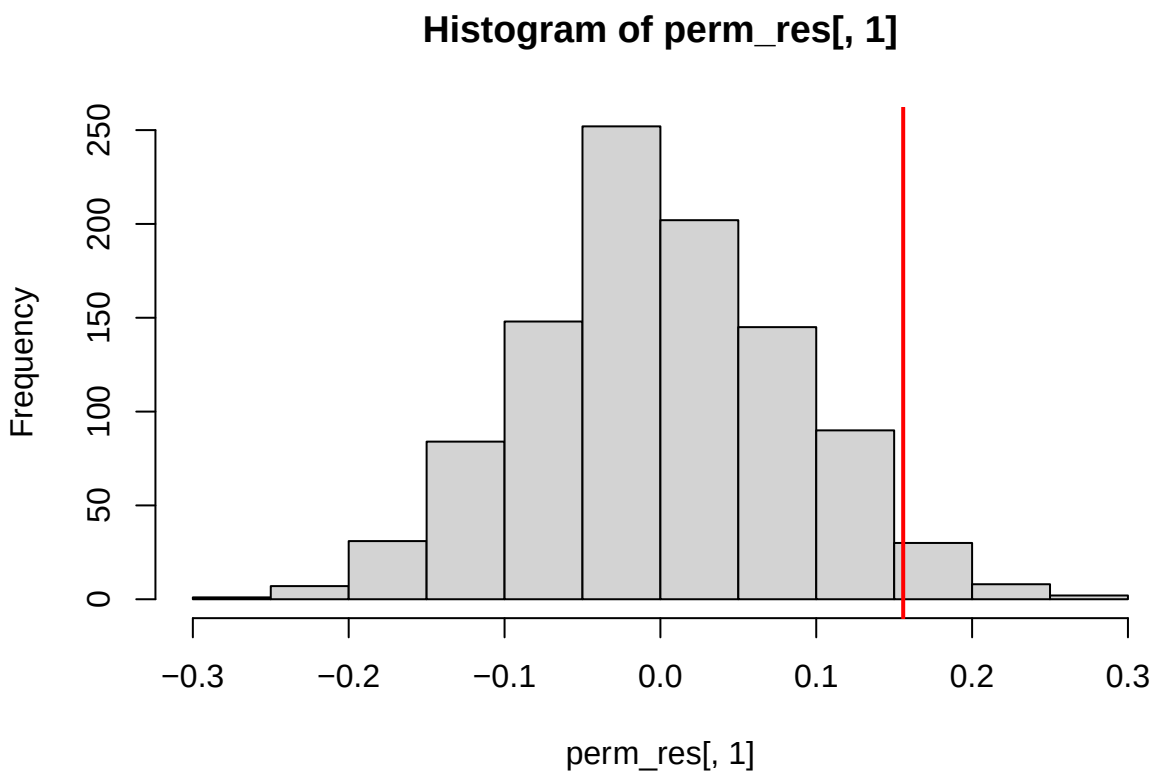
```
## [1] 218
```

3. [50 pt] Perform permutation tests to compare the expressions of all genes between the two populations ("CEU" and "YRI"). Obtain the p-values. Adjust the p-values. How many genes have FDR < 0.1?

```r
idxA <- which(colnames(norm_edata) %in% pdata$sample.id[pdata$population == "CEU"])
idxB <- which(colnames(norm_edata) %in% pdata$sample.id[pdata$population == "YRI"])
actual_diffs <- apply(norm_edata, 1, function(x) mean(x[idxA]) - mean(x[idxB]))

B <- 1000
perm_res <- c()
set.seed(123)
for (i in seq_len(B)) {
    cur_idxA <- sample(seq_len(ncol(norm_edata)), 60)
    cur_idxB <- setdiff(seq_len(ncol(norm_edata)), cur_idxA)
    cur_diffs <- apply(norm_edata, 1, function(x) mean(x[cur_idxA]) - mean(x[cur_idxB]))
    perm_res <- rbind(perm_res, cur_diffs)
}

hist(perm_res[, 1])
abline(v = actual_diffs[1], col = "red", lwd = 2)
```




```
empirical_p <- c()
for (j in 1:ncol(perm_res)) {
    if (actual_diffs[j] < 0) {
        empirical_p <- c(empirical_p, 2 * sum(perm_res[, j] <= actual_diffs[j]) / B)
    } else {
        empirical_p <- c(empirical_p, 2 * sum(perm_res[, j] >= actual_diffs[j]) / B)
    }
}

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

sum(adj_p_per < 0.1)
```

```
## [1] 222
```