

qbs120_ps9_gibran_new

Gibran Erlangga

11/15/2021

Import Libraries

```
library(tidyverse)
library(resample)
```

Question 1

(Based on Rice 12.1) Repeat problem 12.1 but simulate 1000 separate data sets with each data set containing 7 batches of 10 values.

```
GetSimTabletData = function() {
  return(data.frame(
    lab=as.factor(c(1, 2, 3, 4, 5, 6, 7)),
    data=rnorm(70, 4, sd=sqrt(0.0037)))
}

# set iter
n_iter = 1000

# create list to store values
data <- list()

for (i in 1:n_iter) {
  data[[i]] <- GetSimTabletData()
}
```

- (a) For each of the simulated data sets, test the H_0 of no difference in the mean for the seven levels using an F test and compute the associated p-value. Hint: to get the p-value from an R `aov()` analysis, you can use: `summary(aov(...))[[1]][["Pr(>F)"]][1]`

```
pvalues <- c()

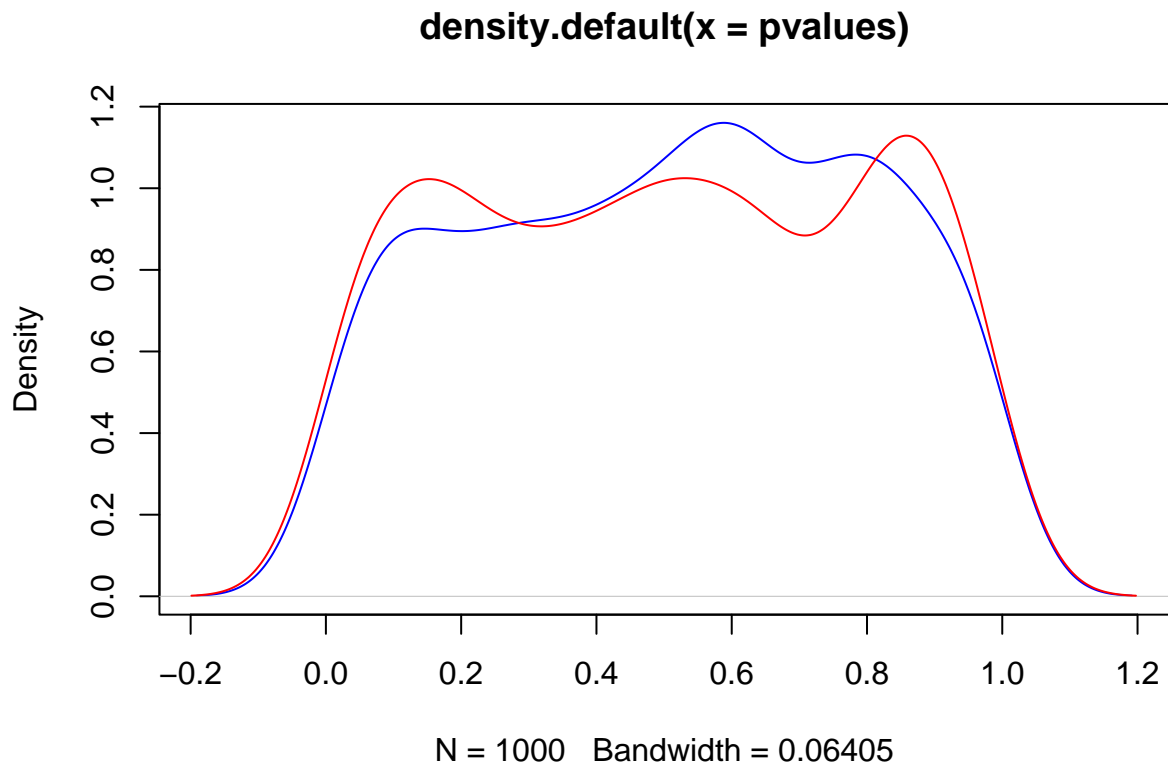
# one-way ANOVA test
for (i in 1:n_iter) {
  temp_result_aov <- aov(data ~ lab, data = data[[i]])
  pvalues <- append(pvalues, summary(temp_result_aov)[[1]][['Pr(>F)']][[1]])
}

head(pvalues)
```

```
## [1] 0.64413010 0.01971485 0.17978579 0.20791457 0.04135847 0.91926123
```

(b) Generate a probability plot for these p-values relative to $U(0,1)$.

```
plot(density(pvalues), col='blue')  
lines(density(runif(1000, 0, 1)), col='red')
```



(c) Does the distribution match your expectations? Explain.

This match with my expectation as the distribution is uniformly distributed.

(d) For how many simulations would the H_0 be rejected at $\alpha = 0.05$? Does this match your expectations? Explain.

```
cat(sum(pvalues < 0.05), "out of 1000 simulations rejected the null hypothesis.")
```

```
## 48 out of 1000 simulations rejected the null hypothesis.
```

This match with my expectation as we can observe from the graph in c), most values are higher than 0.05. In other words, this number represents the tails on both ends.

(e) After controlling for FWER using the Bonferroni method, how many simulations are significant? Does this match your expectations?

```
pval_bonferroni <- p.adjust(pvalues, method = "bonferroni", n = length(pvalues))
cat(sum(pval_bonferroni < 0.05), "out of 1000 simulations rejected the null hypothesis.")
```

```
## 0 out of 1000 simulations rejected the null hypothesis.
```

Bonferroni correction suggests that the p-value for each test must be equal to its alpha divided by the number of tests performed. The result match with my expectation as bonferroni method corrects is relatively conservative compared to other methods, hence most values are higher than 0.05.

- (f) After controlling for the FDR using the Benjamini & Hochberg method, how many simulations have FDR values less than 0.05. Does this match your expectations?

```
pval_bh <- p.adjust(pvalues, method = "BH", n = length(pvalues))
paste(sum(pval_bh < 0.05), "out of 1000 simulations rejected the null hypothesis.")
```

```
## [1] "0 out of 1000 simulations rejected the null hypothesis."
```

This match with my expectation as Benjamini & Hochberg method is a sequentially modified bonferroni correction, so it shares the same trait (relatively conservative compared to other methods), hence most values are higher than 0.05.

Question 2

Answer the following questions using the pancreatic cancer gene expression example from the last problem set. For these questions, assume that statistical tests are performed for all measured genes. Answer a) - c) for both Experiments A and B. Answer d) - h) for just Experiment A.

- (a) Which approach to MHC (FWER or FDR) do you think will provide the best balance of type I error control and power? Justify your answer.
For Experiment A, FWER will provide better balance for type I error control and power because the number of HA cases is small compared to overall hypotheses. Additionally, genes are independent and hence its associated p-values will also be independent.

For Experiment B, FDR will be a better measure because it can give better balance of type I error control and power - a sizeable fraction of the hypotheses correspond to HA (20%).

- (b) For FWER control, is Bonferroni, Holm, Hochberg, Hommel or Westfall & Young preferable? Justify your answer.
We know PRD applies given independence of the hypotheses so either Hochberg/ Hommel are justified (better than Bonferroni/ Holm). In this case Hommel is the best from technical standpoint for both experiments, but Bonferroni is easier to interpret for non-statisticians.
- (c) For FDR control, is Benjamini & Hochberg or Benjamini & Yekutieli preferable? Justify your answer.
Benjamini & Hochberg is preferred for both experiments as we can assume PRD.
- (d) Simulate data for Experiment A and test each gene for DE according to the H0 and HA specified above. Assume that $n = 50$, $\mu_{M0} = 0.5$ and $\mu_{M1} = 1.75$

```

n = 50
m0 = 0.5
pop_m0 = 1000
m1 = 1.75
true_a = 10

# generate data
exp_a_data = matrix(rnorm(n*pop_m0, m0, 1), nrow=n, ncol=pop_m0)
exp_a_data[(n/2+1):n, 1:true_a] = rnorm(true_a*n/2, m1, 1)

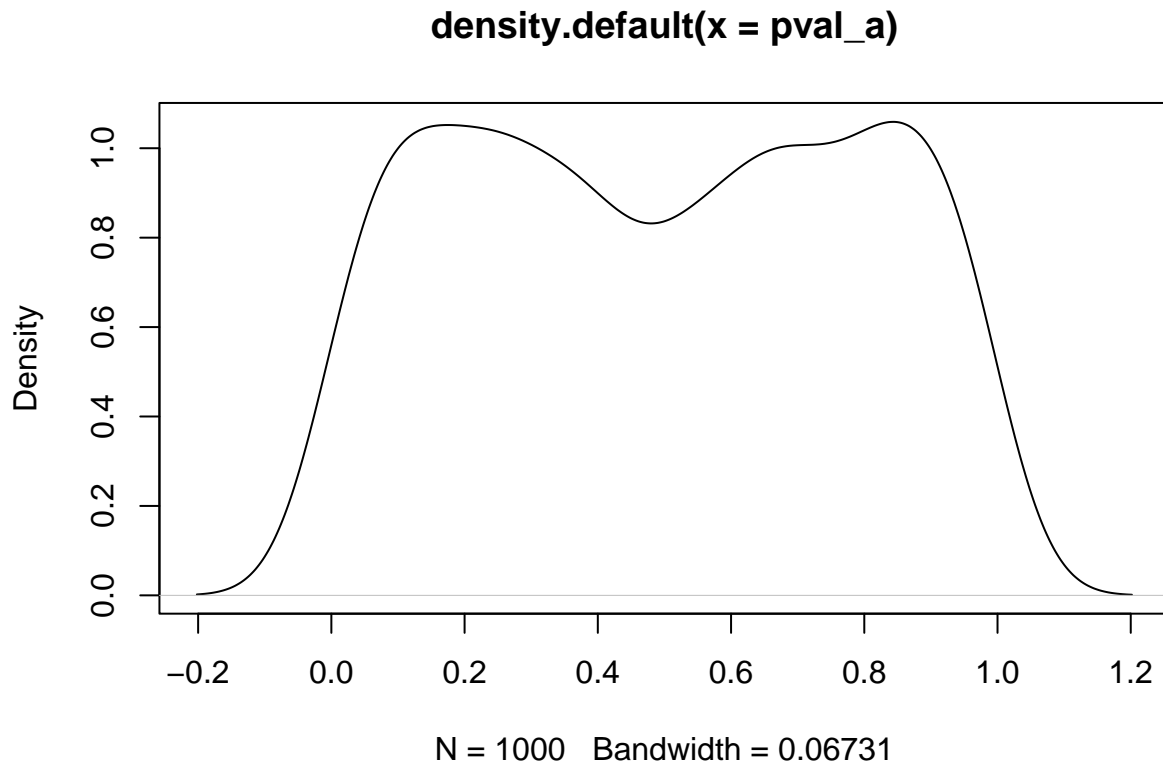
get_pval <- function(x) {
  pval = t.test(x[1:(n/2)], x[n/2+1:n], alternative="two.sided", var.equal=T)$p.value
  return(pval)
}

pval_a = apply(exp_a_data, 2, get_pval)

```

- (e) Plot the distribution of raw p-values. Is the distribution of p-values consistent with the global H_0 ? Explain.

```
plot(density(pval_a))
```



Given the small fraction of DEs, the results from one simulated data set will probably look consistent with the global null even though we know the global null is not true in this case.

- (f) How many significant DE genes are there at a FWER of 0.05 using Bonferroni, Holm and Hochberg? Do these results match your expectations? What is your empirical average power in this case?

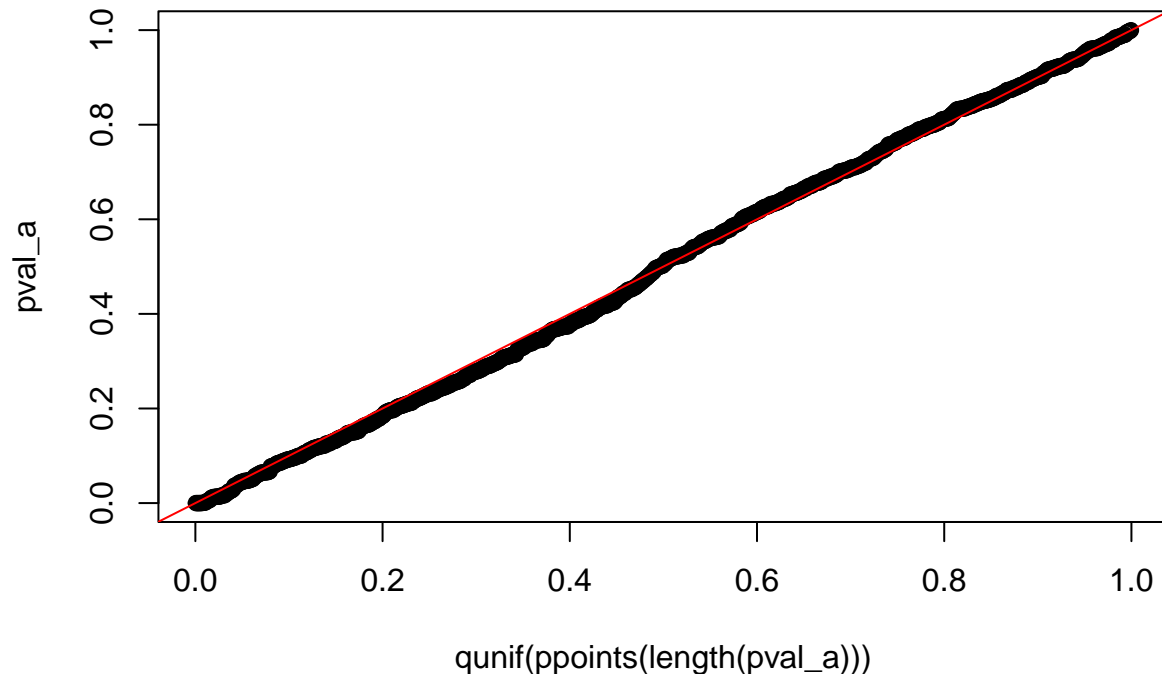
```
methods = c("bonferroni", "holm", "hochberg")

apply_corrections <- function(pval, method, num_true) {
  adjusted_pval = p.adjust(pval, method=method)
  significant_DEs = length(which(adjusted_pval[1:num_true] < 0.05))
  emp_power = significant_DEs/num_true
  cat("# of significant DEs in experiment A is", significant_DEs)
  cat("empirical power is", emp_power)
}

for (i in methods) {
  apply_corrections(pval_a, i, true_a)
}
```

```
## # of significant DEs in experiment A is 6empirical power is 0.6# of significant DEs in experiment A
```

```
qqplot(qunif(ppoints(length(pval_a))), pval_a)
abline(0, 1, col='red')
```



- (g) How many significant DE genes are there at an FDR of 0.05 using BH and BY methods? Do these results match your expectations? What is your empirical average power in this case?

```

methods = c("BH", "BY")

for (i in methods) {
  apply_corrections(pval_a, i, true_a)
}

```

of significant DEs in experiment A is 8empirical power is 0.8# of significant DEs in experiment A

- (h) Proposed a statistically valid p-value weight that could be used with a weighted FDR (wFDR) analysis to improve statistical power. Perform wFDR analysis using this weight. Was power improved relative to unweighted FDR?

```

get_variance <- function(data) {
  temp_var = apply(data, 2, var)
  return(temp_var/mean(temp_var))
}

weight_a = get_variance(exp_a_data)
pval_weight_a = pval_a/weight_a

for (i in methods) {
  apply_corrections(pval_weight_a, i, true_a)
}

```

of significant DEs in experiment A is 8empirical power is 0.8# of significant DEs in experiment A

Question 3

(Based on Rice 13.1) Adult-onset diabetes is known to be highly genetically determined. A study was done comparing frequencies of a particular allele in a sample of such diabetics and a sample of nondiabetics. The data are shown in the following table:

```

data <- data.frame(
  "Diabetic" = c(12, 39),
  "Normal" = c(4, 49),
  row.names = c("Bb or bb", "BB"),
  stringsAsFactors = FALSE)
data

```

```

##           Diabetic Normal
## Bb or bb      12      4
## BB           39     49

```

```

data <- matrix(c(12, 4, 39, 49), ncol=2, byrow = TRUE)
colnames(data) <- c("Diabetic", "Normal")
rownames(data) <- c("Bb or bb", "BB")
#data <- as.table(data)
data

```

```

##           Diabetic Normal
## Bb or bb      12      4
## BB           39     49

```

- (a) Are the frequencies of the alleles significantly different in the two groups? Test using both the Fisher Exact test and a chi-squared test (you can use R functions). Comment on the difference in p-values between the two methods.

Independence tests are used to determine if there is a significant relationship between two categorical variables.

The Chi-square test is used when the sample is large enough (in this case the p-value is an approximation that becomes exact when the sample becomes infinite, which is the case for many statistical tests). On the other hand, the Fisher's exact test is used when the sample is small (and in this case the p-value is exact and is not an approximation).

Some literatures indicate that the usual rule for deciding whether the χ^2 approximation is good enough is that the Chi-square test is not appropriate when the expected values in one of the cells of the contingency table is less than 5, and in this case the Fisher's exact test is preferred.

```
# check expected value to evaluate which independent tests to use
chisq.test(data)$expected
```

```
##           Diabetic    Normal
## Bb or bb  7.846154  8.153846
## BB       43.153846 44.846154
```

```
# Fisher Exact test
fisher.test(data)$p.value
```

```
## [1] 0.03034184
```

```
# Chi-squared test
chisq.test(data)$p.value
```

```
## [1] 0.04698425
```

We can see from the contingency tables of expected value generated above that none of the cells have value less than 5, so we can assume that p-value from chi-squared test is a good enough approximation for the case.

In Fisher's Exact test and chi-squared test above, the p-value is 0.03034 and 0.04698, respectively. Based on previous sentence, we can choose to use p-value from chi-square test in this case.

- (b) Explain in your own words how hypothesis testing is performed using each approach.

In Fisher's exact test, we try to calculate the exact p-value as opposed to do an estimation like what chi-square test does. Given the contingency tables:

```
sample_table <- data.frame(
  "Group A" = c("a", "c"),
  "Group B" = c("b", "d"),
  row.names = c("Exposure A", "Exposure B"),
  stringsAsFactors = FALSE)

sample_table
```

```
##           Group.A Group.B
## Exposure A      a      b
## Exposure B      c      d
```

The formula is:

$$\begin{aligned}
 p &= \frac{\binom{a+b}{a} \cdot \binom{c+d}{c}}{\binom{n}{a+c}} \\
 &= \frac{\binom{a+b}{b} \cdot \binom{c+d}{d}}{\binom{n}{b+d}} \\
 &= \frac{a+b! \cdot c+d! \cdot a+c! \cdot b+d!}{a! \cdot b! \cdot c! \cdot d!}
 \end{aligned}$$

While in Chi-squared test, the observed frequencies for two or more groups are compared with expected frequencies by chance. Formula is:

$$\chi^2 = \sum \frac{(O - E)^2}{E}$$

with O as observed frequencies and E as expected frequencies by chance.

Question 4

(Based on Rice 13.21) Do the following for problem 13.1: (a) Estimate the odds ratio (OR) using the unconditional MLE (what is described in Rice) without using special-purpose R packages.

```
odds_ratio = (data[1,1]*data[2,2])/(data[1,2]*data[2,1])
odds_ratio
```

```
## [1] 3.769231
```

- (b) Estimate the OR using the four different estimation methods implemented by the `oddsratio()` method in R `epitools` package. Confirm that your estimate from part a) matches the output for the "wald" method.

```
library(epitools)

or_estimate_fisher <- oddsratio.fisher(as.matrix(data))$measure[[2]]
or_estimate_midp <- oddsratio.midp(as.matrix(data))$measure[[2]]
or_estimate_small <- oddsratio.small(as.matrix(data))$measure[[2]]
```

```
## Warning in any(or, na.rm = TRUE): coercing argument of type 'double' to logical
```

```
or_estimate_wald <- oddsratio.wald(as.matrix(data))$measure[[2]]

cat("estimated OR with fisher method is", or_estimate_fisher)
```

```
## estimated OR with fisher method is 3.722766
```



```
cat("estimated OR with midp method is", or_estimate_midp)
```

```
## estimated OR with midp method is 3.639697
```

```
cat("estimated OR with small method is", or_estimate_small)
```

```
## estimated OR with small method is 2.94
```

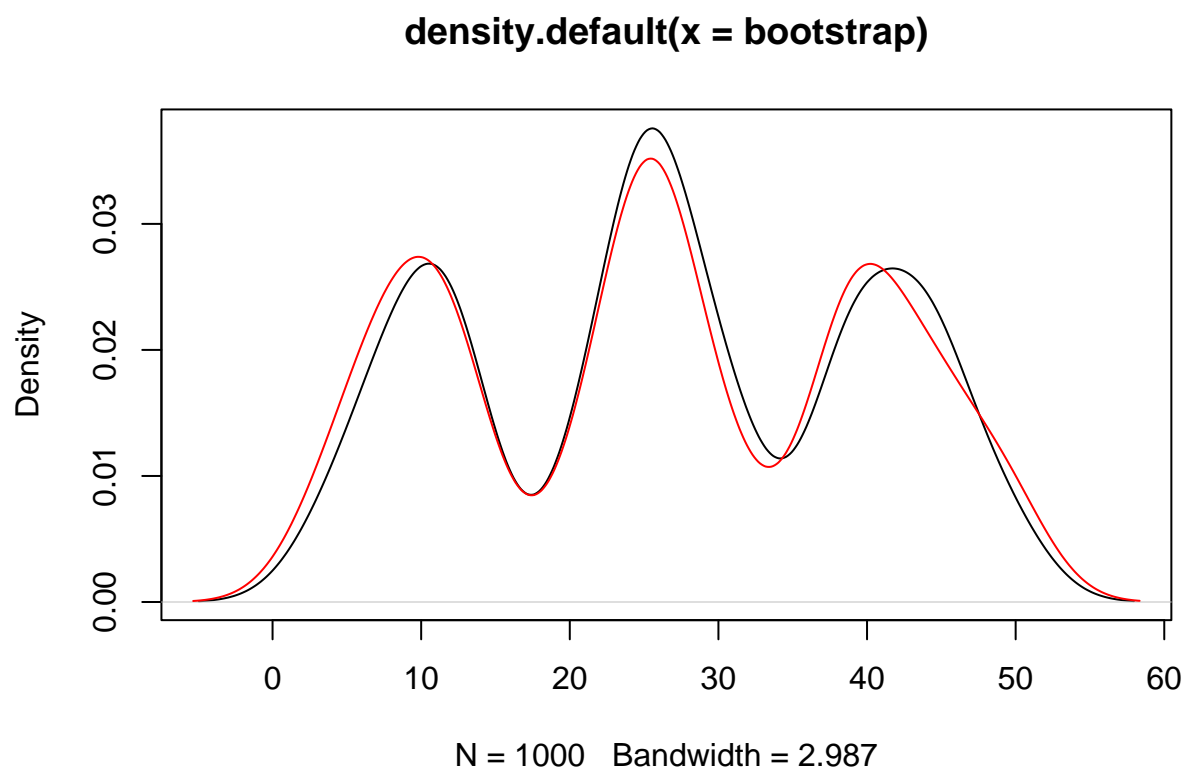
```
cat("estimated OR with wald method is", or_estimate_wald)
```

```
## estimated OR with wald method is 3.769231
```

- (c) Use the parametric bootstrap to estimate the sampling distribution of the OR following the approach outlined in Rice Section 13.6. Plot a kernel density estimate of this sampling distribution.

borrowing the function from PS7:

```
BootstrapDist = function(data, estimator, B=1000) {  
  n = length(data)  
  bootstrap.data = matrix(sample(data, B*n, replace=T), nrow=B, ncol=n)  
  bs.estimates = apply(bootstrap.data, 1, function(x) {  
    estimator(x)  
  })  
  return(bs.estimates)  
}  
  
bootstrap = BootstrapDist(data, function(x) {mean(x, trim=or_estimate_wald)})  
se_bootstrap = sd(bootstrap)  
bootstrap_median = BootstrapDist(data, function(x) {median(x)})  
se_bootstrap_median = sd(bootstrap_median)  
  
plot(density(bootstrap))  
lines(density(bootstrap_median), col='red')
```



- (d) Calculate the 95% CI using the percentile method. Compare your results to the OR and CI estimates from the R `fisher.test()` and `oddsratio()` methods. Do they differ? If so, why?

```
CI.percentile(bootstrap(as.matrix(data)))
```

```
##      2.5% 97.5%
## NULL1  12   39
## NULL2  12   39
## NULL3   4   49
## NULL4   4   49
```