Homework 5 - BIOS 7649

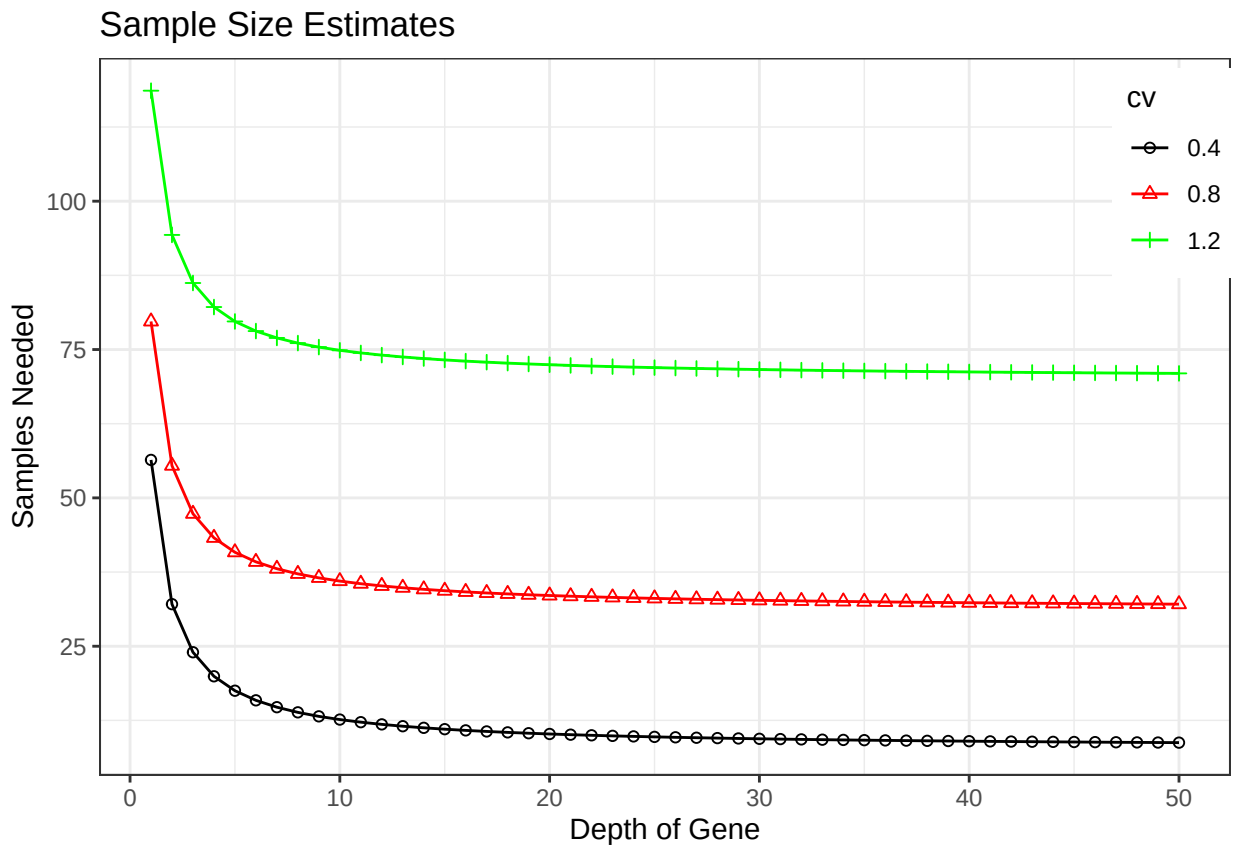
Dominic Adducci

Question 1 - Next Generation Sequencing: Sample Size Estimates

Part A

Using **rnapower()**, recreate Figure 3 from the journal club paper using: Hart et al., (2013) "Calculating sample size estimates for RNA sequencing data." What does this figure show?

```
## Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2
## 3.5.0.
## i Please use the `legend.position.inside` argument of `theme()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Part B

For the Montgomery data, create a row in Table 1 in the Hart et al. paper. Note that genes that have zero counts in all 10 samples were already excluded, so "% mapped", will be 100%. How does the Montgomery data compare to the other data sets in Table 1?

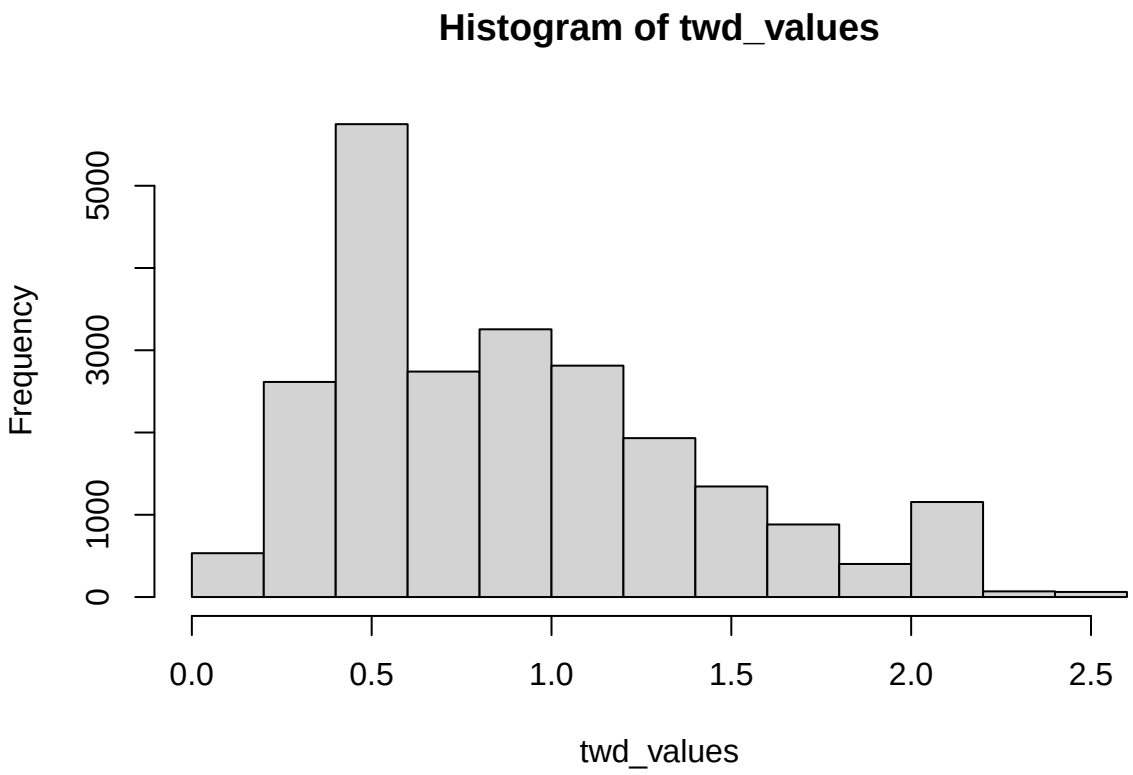
Part C

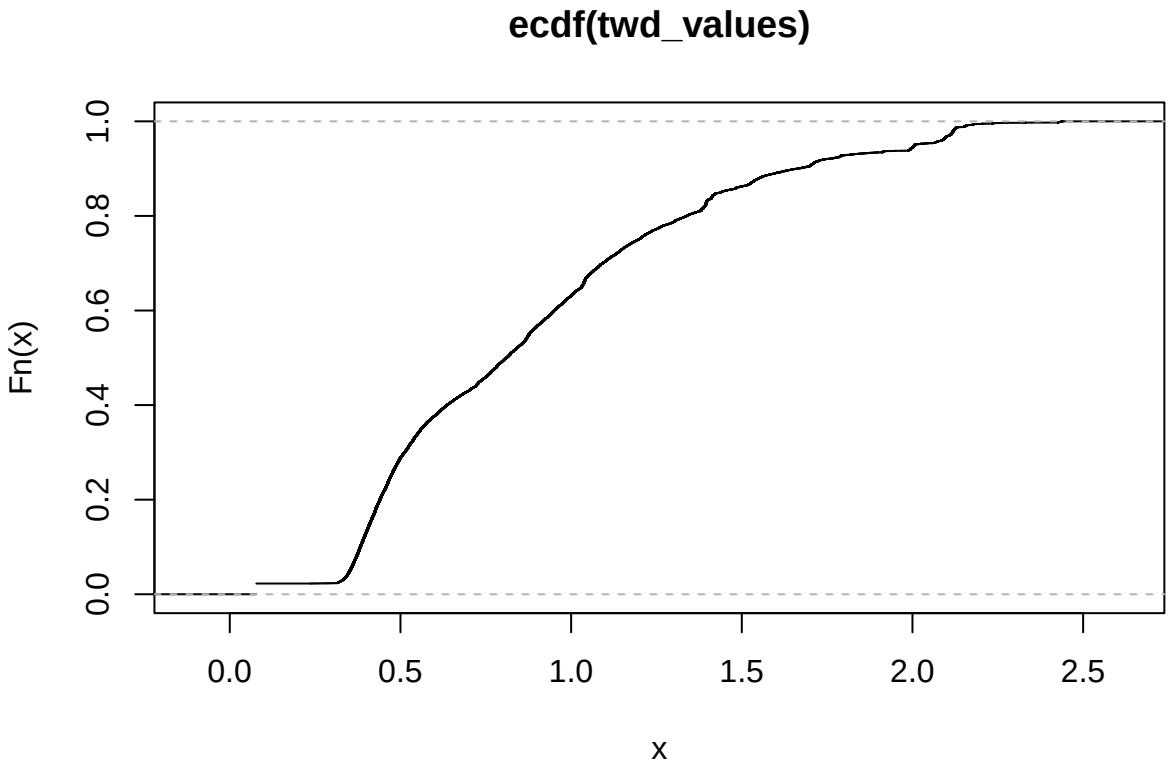
Calculate the biological coefficient of variations (CV) from the Montgomery human lymphoblastoid data (use the function **estimateTagwiseDisp()** in edgeR). Plot the histogram and empirical cdf (use **ecdf()** function) and report the median and 90% percentile. How do the CVs compare with the examples in the Hart et al., paper in Figure 2?

The square root of the negative binomial dispersion is known as the biological coefficient of variation (BCV).

$$BCV = \sqrt{\phi_g}\ ;\ where\ \phi_g = NB\ dispersion$$

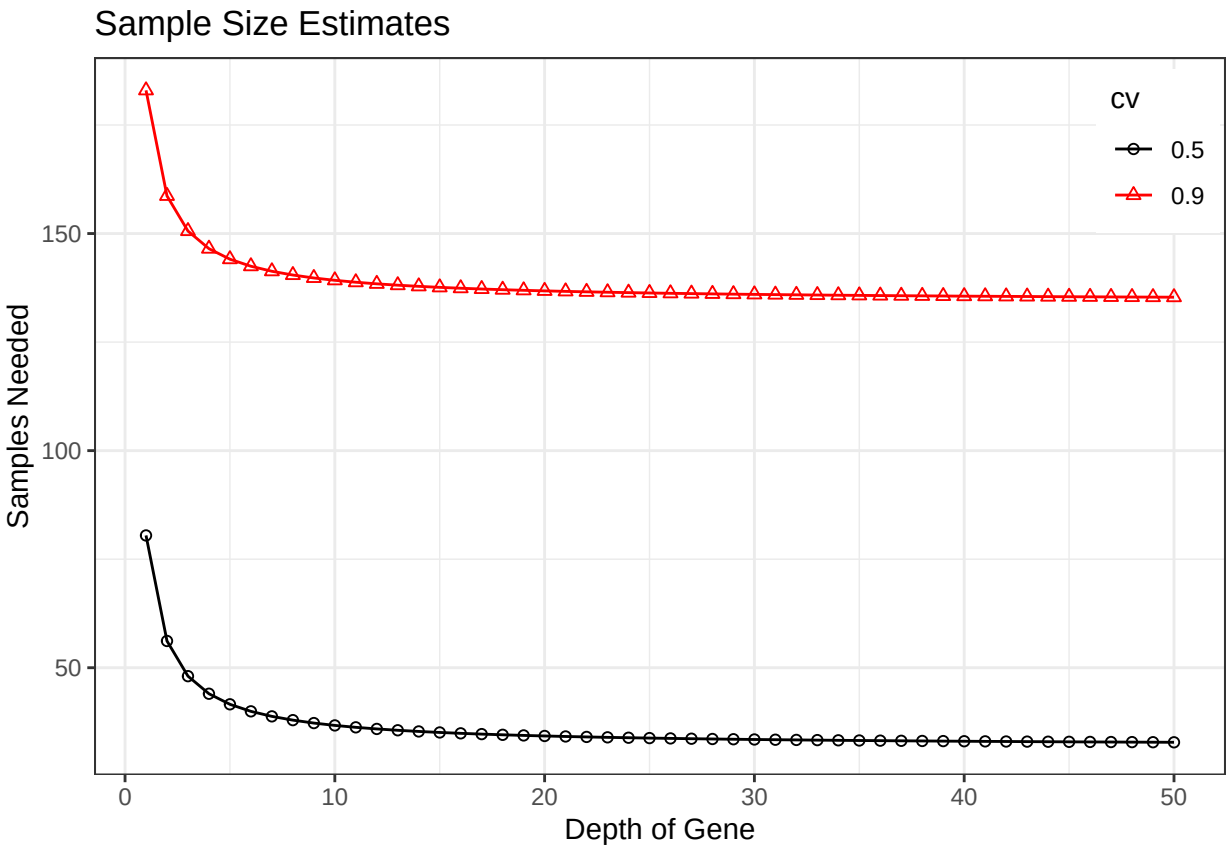
Calculating this first involved estimating the common dispersion and then the tagwise dispersion. After this the tagwise dispersions can be extracted and plotted using both a histogram and a cumulative distribution plot.





Part D

Using **rnapower()**, recreate Figure 3 from Hart et al. again but with two curves using the median and the 90% percentile CV across genes for the Montgomery data. What sample size do you recommend?



Part E

Using **rnapower()**, recreate the curve (not the histogram) in the top Figure 4 from the Hart et al., paper. If you cannot recreate the figure, please explain any differences.

CODE

```
library(tidyverse)
library(RNASeqPower)
library(edgeR)
library(cqn)
library(kableExtra)


### START QUESTION 1 CODE ###

# Loading data into R environment.
data("montgomery.subset")

data("uCovar")


## START QUESTION 1 PART A CODE ##

# Calculating sample size estimates using rnapower() function.
figure3_data <- data.frame(rnapower(depth = seq(1,50,by = 1),
                                    cv = c(.4,0.8,1.2),effect = 2,
                                    alpha = 0.01, power = 0.8))

# Formatting the results, This includes turning data into long format,
# changing values for cv, and then converting depth and cv from character
# to numeric.
colnames(figure3_data) <- c("cv0.4","cv0.8","cv1.2")

figure3_data <- rownames_to_column(figure3_data,"depth")

figure3_data_long <- pivot_longer(figure3_data,
                                  cols = starts_with("cv"),
                                  names_to = "cv",
                                  values_to = "sample_size")

figure3_data_long$cv <- replace(figure3_data_long$cv,
                                figure3_data_long$cv == "cv0.4",0.4)

figure3_data_long$cv <- replace(figure3_data_long$cv,
                                figure3_data_long$cv == "cv0.8",0.8)

figure3_data_long$cv <- replace(figure3_data_long$cv,
                                figure3_data_long$cv == "cv1.2",1.2)

figure3_data_long <- figure3_data_long %>%
  mutate(depth = as.numeric(depth),
         cv = factor(cv))

# Plotting samples size estimates.
figure3_plot <- ggplot(figure3_data_long, aes(x = depth,
                                              y = sample_size,
                                              color = cv, shape = cv)) +
```

```
  geom_point() +
  geom_line() +
  scale_color_manual(values = c("black","red","green")) +
  scale_shape_manual(values = c(1,2,3)) +
  labs(title = "Sample Size Estimates",
       x = "Depth of Gene", y = "Samples Needed",
       legend = "CV") +
  theme_bw() +
  theme(legend.position = c(0.95,0.84))

figure3_plot

## FINISH QUESTION 1 PART A CODE ##


## START QUESTION 1 PART B CODE ##

# Making a DGEList from the Montgomery data.
mgd_dgel <- DGEList(montgomery.subset)

# Calculating normal factors.
mgd_nf <- calcNormFactors(mgd_dgel)

# Calculating counts per million.
cpm_results <- cpm(mgd_nf)

# Initializing the ranges.
cpg_cpm_range <- c(0,0.01,0.1,1,10,100,1000,Inf)

# Storing genes counts in a vector.
gene_counts <- vector("numeric", 7)

# Using a for loop to calculate the distribution of counts for each range.
for(i in 1:7){
  gene_counts[i] <- sum(rowSums(cpm_results >= cpg_cpm_range[i] &
                                cpm_results < cpg_cpm_range[i + 1]) > 0)
}

# Calculating the average number of reads for each gene and then calculating
# the average counts in total. Lastly calculate counts per million.
# Total library size serve as denominator.
gene_average <- mean(rowMeans(mgd_dgel$counts))

library_size <- sum(mgd_dgel$samples$lib.size)

read_average <- (gene_average/library_size) * 1e6

table1_row <- data.frame(matrix(c(read_average,100,gene_counts),nrow = 1))

table1_row <- cbind("Montgomery Data",10,table1_row)

colnames(table1_row) <- c("Sample","n","Avg Reads","% Mapped","<0.01",
                          "0.01-0.1","0.1-1","1-10","10-100","100-10")
```

```r
table1_tbl <- table1_row %>%
  kbl(caption = "Montgomery Data: Counts per Gene per Millions Reads Mapped",
      booktabs = TRUE) %>%
  kable_styling(latex_options = "HOLD_position")

## FINISH QUESTION 1 PART B CODE ##


## START QUESTION 1 PART C CODE ##

# Calculating the biological coefficient of variation for the Montgomery data
# set. The estimateTagwiseDisp() function from edgeR is used.

# First calculating common dispersion.
cd_md <- estimateCommonDisp(mgd_dgel)

# Calculating the tagwise dispersion.
twd_md <- estimateTagwiseDisp(cd_md)

# Creating a histogram for the squareroot of the tagwise dispersion values.
twd_values <- sqrt(twd_md$tagwise.dispersion)

# Creating empirical CDF.
twd_ecdf <- ecdf(twd_values)

hist(twd_values)
plot(ecdf(twd_values))

## START QUESTION 1 PART D CODE ##

# Calculating median and 90th percentile cv across genes.
twd_median <- quantile(twd_ecdf,0.5)
twd_nintieth <- quantile(twd_ecdf,0.9)

# Making a plot to show sample size using rnapower() function.
twd_rnapower <- data.frame(rnapower(depth = seq(1,50,by = 1),
                                    cv = c(twd_median,twd_nintieth),effect = 2,
                                    alpha = 0.01, power = 0.8))

# Formatting the results. This includes turning data into long format,
# changing value for cv, and then converting depth and cv from character
# to numeric.
colnames(twd_rnapower) <- c("cv0.5","cv0.9")

twd_rnapower <- rownames_to_column(twd_rnapower,"depth")

twd_rnapower_long <- pivot_longer(twd_rnapower,
                                  cols = starts_with("cv"),
                                  names_to = "cv",
                                  values_to = "sample_size")

twd_rnapower_long$cv <- replace(twd_rnapower_long$cv,
                                twd_rnapower_long$cv == "cv0.5",0.5)
```

```r
twd_rnapower_long$cv <- replace(twd_rnapower_long$cv,
                                twd_rnapower_long$cv == "cv0.9",0.9)

twd_rnapower_long <- twd_rnapower_long %>%
  mutate(depth = as.numeric(depth),
         cv = factor(cv))

twd_ss_plot <- ggplot(twd_rnapower_long, aes(depth,
                                             y = sample_size,
                                             color = cv, shape = cv)) +
  geom_point() +
  geom_line() +
  scale_color_manual(values = c("black","red")) +
  scale_shape_manual(values = c(1,2)) +
  labs(title = "Sample Size Estimates",
       x = "Depth of Gene",y = "Samples Needed",
       legend = "CV") +
  theme_bw() +
  theme(legend.position = c(0.94,0.87))

twd_ss_plot
```