# Computer Exercise 4: Metagenomic data analysis

Introduction to Bioinformatics (MVE510), Autumn 2024

**Group Member:** Houshi He

**Swedish Social Security Number:** 20011114-4838

January 18, 2025

# Contents

# 1 Questions and Answers

## 1.1 Question 1

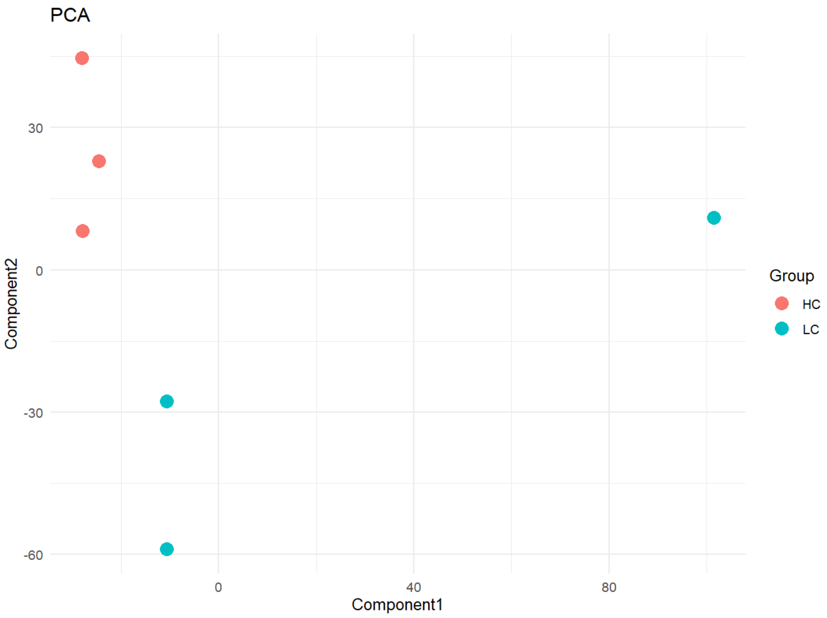
**Problem:**
1. How many counts in total do the different samples have?
2. How is the annotation file structured? Why do you think the annotation for some OTUs is incomplete?

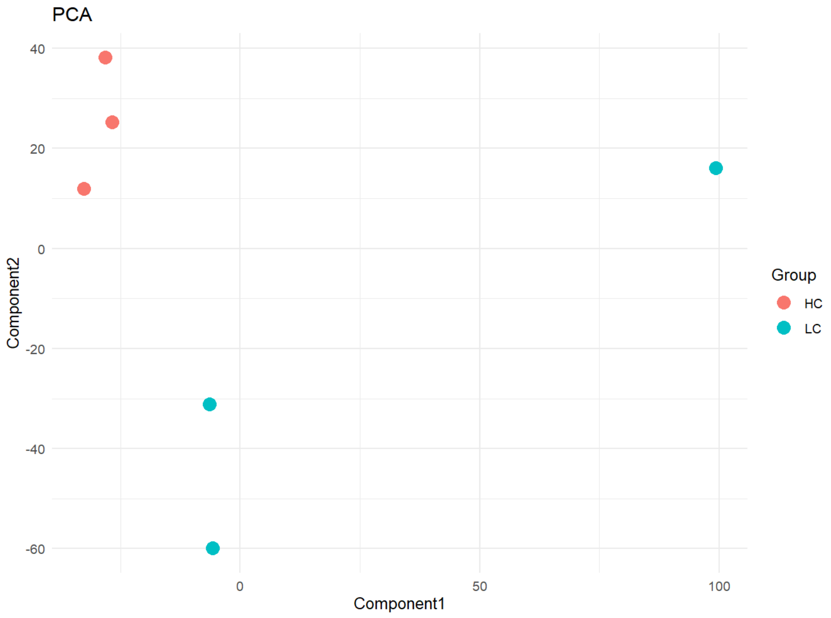
**Answer:**
1. The total read counts for each sample are: HC1: 51732, HC2: 41426, HC3: 43220, LC1: 35622, LC2: 34242, LC3: 30593.
2. The annotation file has 8 columns: OTU.ID, Kingdom, Phylum, Class, Order, Family, Genus, Species. I think it's because 1. The database is incomplete and some OTUs are not identified or classified. 2. The sequence similarity is low, resulting in an inability to match known taxa. 3. OTUs may belong to unknown or unstudied species.

## 1.2 Question 2

**Problem:**
1. Is there a separation between samples from high and low oil exposure?
2. Can you see any difference? Did the samples within the groups become more or less homogenous?

**Answer:**
1. Yes, from the plots we can see a separation. Only one sample from the LC group doesn't seem to obey the separation rule.
2. The top plot shows the result using raw data. The bottom plot shows the result after transformation. After transformation, HC and LC samples show a clearer separation, and samples within each group are closer together.





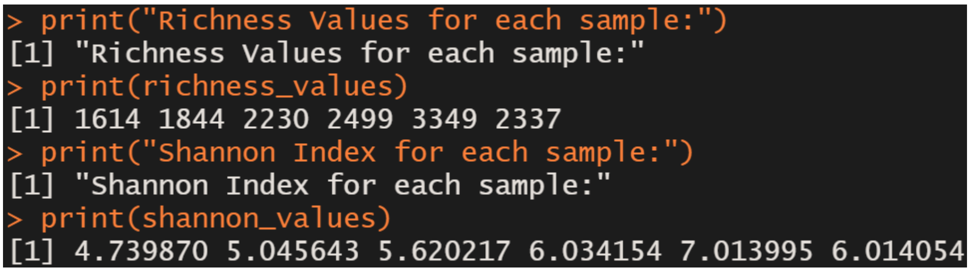
## 1.3 Question 3

**Answer:**
1. The code can be found in the Appendix.

## 1.4 Question 4

**Problem:**
1. Do you see any difference in diversity between samples from high and low oil exposure?

**Answer:**
1. The richness of the low oil exposure group was generally higher, indicating that the samples of the low oil exposure group contained more unique species. The richness of the high oil exposure group was relatively low, which may indicate that the diversity of the microbial community under high oil exposure is lower and the species are fewer. The Shannon index of the low oil exposure group was generally higher, indicating that the species distribution in these samples was more even, with less difference in abundance between species. The Shannon index of the high oil exposure group was lower, which may indicate that the abundance of certain species may dominate under high oil exposure, resulting in a more uneven distribution of species.



```
> print("Richness Values for each sample:")
[1] "Richness Values for each sample:"
> print(richness_values)
[1] 1614 1844 2230 2499 3349 2337
> print("Shannon Index for each sample:")
[1] "Shannon Index for each sample:"
> print(shannon_values)
[1] 4.739870 5.045643 5.620217 6.034154 7.013995 6.014054
```

## 1.5 Question 5

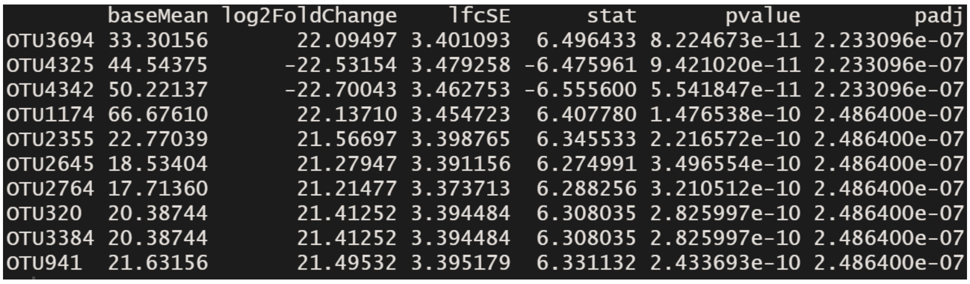
**Problem:**
1. Can you find any arguments why it may be especially important to work directly with the count data in this exercise?
2. How do you interpret the adjusted p-value? Set a reasonable significant cut-off and describe how many OTUs are significant.

**Answer:**
1. Transforming or normalizing the data might violate the distribution assumption, which can reduce the accuracy of the results. Additionally, using the count data can directly reflect the biological information measured in the experiment without introducing potential biases from data transformation or normalization.
2. The adjusted p-value accounts for multiple tests and controls the false positives. I use 0.05 as the cut-off for the adjusted p-value, and 19 OTUs are significant.

## 1.6 Question 6

**Problem:**
1. Do these bacteria increase or decrease in the oil-contaminated samples?
2. Are bacteria from these families present in your result? Do they increase or decrease in the exposed sediments?

**Answer:**
1. The log2FoldChange of OTU4325 and OTU4342 was less than 0, indicating that they were significantly reduced in the contaminated samples. Except for these two, the other eight bacteria have increased significantly.
2. OTU4325 and OTU1174 are in my result, and they belong to Alteromonadaceae. OTU4325 decreases and OTU1174 increases.



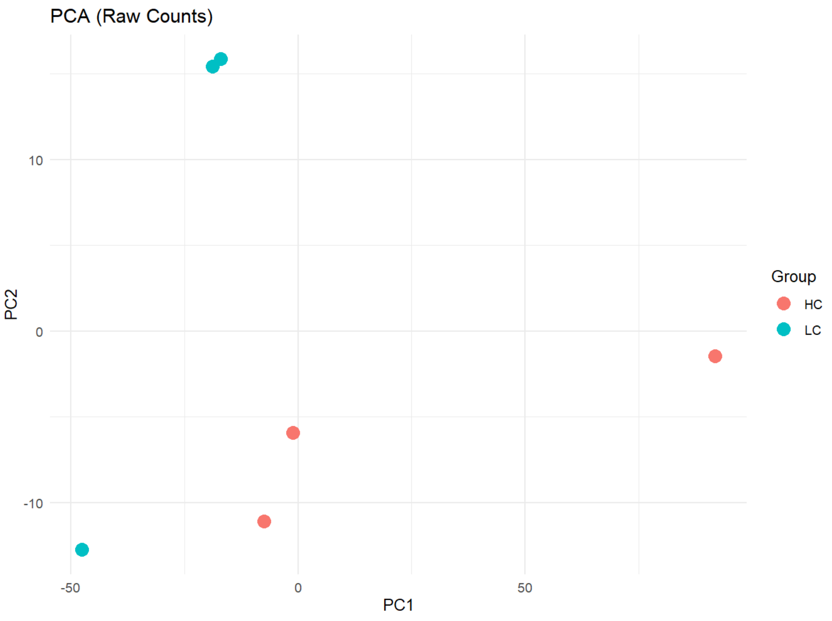
```
        baseMean log2FoldChange    lfcSE      stat       pvalue         padj
OTU3694 33.30156       22.09497 3.401093  6.496433 8.224673e-11 2.233096e-07
OTU4325 44.54375      -22.53154 3.479258 -6.475961 9.421020e-11 2.233096e-07
OTU4342 50.22137      -22.70043 3.462753 -6.555600 5.541847e-11 2.233096e-07
OTU1174 66.67610       22.13710 3.454723  6.407780 1.476538e-10 2.486400e-07
OTU2355 22.77039       21.56697 3.398765  6.345533 2.216572e-10 2.486400e-07
OTU2645 18.53404       21.27947 3.391156  6.274991 3.496554e-10 2.486400e-07
OTU2764 17.71360       21.21477 3.373713  6.288256 3.210512e-10 2.486400e-07
OTU320  20.38744       21.41252 3.394484  6.308035 2.825997e-10 2.486400e-07
OTU3384 20.38744       21.41252 3.394484  6.308035 2.825997e-10 2.486400e-07
OTU941  21.63156       21.49532 3.395179  6.331132 2.433693e-10 2.486400e-07
```

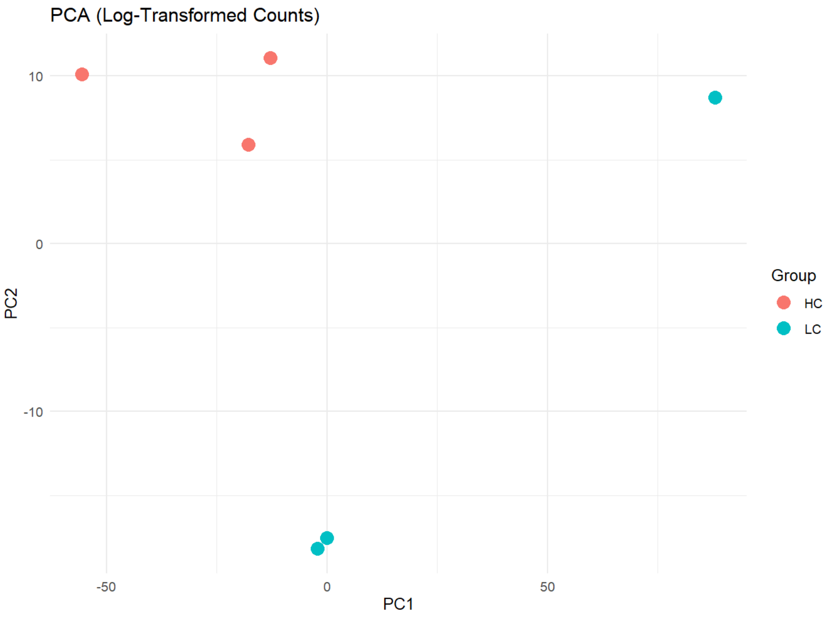
## 1.7 Question 7

**Problem:**
1. How many reads do you have for each sample?
2. Do the samples separate according to the level of exposure? If not, discuss why this may be the case.

**Answer:**
1. The total read counts for each sample are: HC1: 2682675, HC2: 7404610, HC3: 2879691, LC1: 2205249, LC2: 531900, LC3: 1936041.

2. I think we can see a separation, especially after transformation. Before the transformation, the separation is unclear, maybe it is because the raw count data has a lot of variation that makes it difficult to visualize clear patterns.



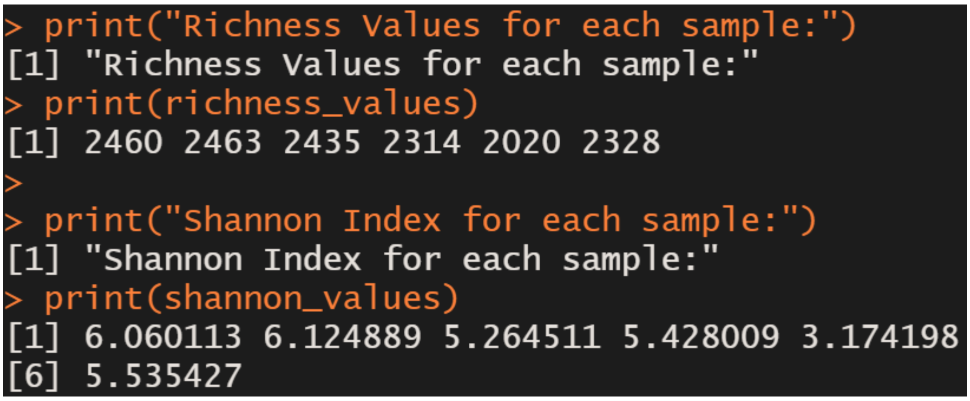


## 1.8 Question 8

**Problem:**
1. What do richness and evenness mean when it comes to gene count data?
2. Do you see any differences between the samples?

**Answer:**
1. Richness is the unique number of OTUs, reflecting the diversity of genes in the sample. Evenness reflects whether the distribution of genes in the sample is uniform.

2. The richness of the samples varied slightly among the six samples, ranging from 2314 to 2463, indicating that the number of gene types in these samples was relatively similar. The Shannon index (evenness) is quite different. From the results, we can see that the Shannon index of the HC group is significantly higher than that of the LC group. This indicates that the gene distribution of the HC group is more uniform, while the gene distribution of the LC group is more uneven, and the abundance of some genes may be dominant.



```
> print("Richness Values for each sample:")
[1] "Richness Values for each sample:"
> print(richness_values)
[1] 2460 2463 2435 2314 2020 2328
>
> print("Shannon Index for each sample:")
[1] "Shannon Index for each sample:"
> print(shannon_values)
[1] 6.060113 6.124889 5.264511 5.428009 3.174198
[6] 5.535427
```

## 1.9 Question 9

**Problem:**
1. How many genes are significant?
2. Are the relative abundance of the most significant genes increasing or decreasing?
3. Find the gene pdxA in the gene list. Does it increase or decrease in the contaminated samples?

**Answer:**
1. 369 genes are significant.
2. Among them, the log2FoldChange of 195 genes is less than 0, which means that the relative abundance of these genes decreases, and the log2FoldChange of 174 genes is greater than 0, which means that the relative abundance of these genes increases.
3. The TIGRFAM for pdxA is TIGR00557. The log2FoldChange of gene TIGR00557 is 1.017108, indicating that it increases in the contaminated samples.

## 1.10 Question 10

**Problem:**
1. In what way does the exposure seem to affect the bacterial communities?

**Answer:**
1. The bacterial communities between the oil-exposed group (HC) and the low-oil-exposed group (LC) showed significant differences in diversity. The Shannon index of the high-oil exposure group was generally lower, which means that in the high-oil pollution environment, the species distribution of the bacterial community was more uneven and some species might dominate the structure of the community. The Shannon index was higher in the low oil exposure group, indicating that the species distribution in the community was more uniform and there were more species. Therefore, oil contamination appears

to reduce bacterial communities' uniformity and diversity. The oil-polluted environment may selectively promote the expression of certain genes that can degrade oil pollutants while inhibiting the expression of other genes. The pdxA gene (4-hydroxythreonine-4-phosphate dehydrogenase) is a gene related to the degradation of oil pollutants by bacteria. According to the results, the expression of the pdxA gene increased in samples with high oil pollution, indicating that this gene was enhanced under the selective pressure of the oil pollution environment.

# 2 Appendix

```r
# Q1
counts_file <- "16s_counts.txt"
annotation_file <- "16s_annotation.txt"

counts <- read.table(counts_file, header = TRUE, sep = "\t", quote = "", comment.char = "")
annotations <- read.table(annotation_file, header = TRUE, sep = "\t", quote = "", comment.char = "")

print("Counts Data:")
print(head(counts))
print("Annotations Data:")
print(head(annotations))

total_counts <- colSums(counts)
print("Total counts per sample:")
print(total_counts)

print("Annotations structure:")
print(str(annotations))

incomplete_annotations <- annotations[apply(annotations, 1, function(row) any(is.na(row))), ]
print("Incomplete annotations:")
print(incomplete_annotations)
print(paste("Number of incomplete annotations:", nrow(incomplete_annotations)))

counts <- counts[rowSums(counts) >= 5, ]
annotations <- annotations[rownames(counts), ]
print(paste("Remaining OTUs after filtering:", nrow(counts)))

# Q2
library(ggplot2)

counts_file <- "16s_counts.txt"
counts <- read.table(counts_file, header = TRUE, sep = "\t", quote = "", comment.char = "")

counts_t <- t(counts)
```

```r

pca_raw <- prcomp(counts_t, scale. = TRUE)

pca_data_raw <- data.frame(pca_raw$x, Group = c("HC", "HC", "HC", "LC", "LC", "LC"))

ggplot(pca_data_raw, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 4) +
  labs(title = "PCA", x = "Component1", y = "Component2") +
  theme_minimal()

counts_vst <- log(counts + 1)

counts_vst_t <- t(counts_vst)

pca_vst <- prcomp(counts_vst_t, scale. = TRUE)

pca_data_vst <- data.frame(pca_vst$x, Group = c("HC", "HC", "HC", "LC", "LC", "LC"))

ggplot(pca_data_vst, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 4) +
  labs(title = "PCA", x = "Component1", y = "Component2") +
  theme_minimal()

# Q3
rarefy_sample <- function(OTUs, counts, depth) {
  reads <- rep(OTUs, times = counts)
  reads_sample <- sample(reads, size = depth, replace = FALSE)
  counts_sample <- as.data.frame(table(reads_sample))
  colnames(counts_sample) <- c("OTU", "Count")
  return(counts_sample)
}

depth <- min(colSums(counts))

rarefied_data_list <- list()
for (i in 1:ncol(counts)) {
  OTUs <- rownames(counts)
  counts_for_OTUs <- counts[, i]
  rarefied_data_list[[i]] <- rarefy_sample(OTUs, counts_for_OTUs, depth)
}

print("Rarefied Data for each sample:")
for (i in 1:length(rarefied_data_list)) {
  print(paste("Sample", i, ":"))
  print(rarefied_data_list[[i]])
}

# Q4
```

```r
richness <- function(counts_sample) {
  richness_value <- sum(counts_sample$Count > 0)
  return(richness_value)
}

shannon_index <- function(counts_sample) {
  total_count <- sum(counts_sample$Count)
  p_i <- counts_sample$Count / total_count
  H_prime <- -sum(p_i * log(p_i))
  return(H_prime)
}

richness_values <- numeric(length(rarefied_data_list))
shannon_values <- numeric(length(rarefied_data_list))

for (i in 1:length(rarefied_data_list)) {
  counts_sample <- rarefied_data_list[[i]]
  richness_values[i] <- richness(counts_sample)
  shannon_values[i] <- shannon_index(counts_sample)
}

print("Richness Values for each sample:")
print(richness_values)
print("Shannon Index for each sample:")
print(shannon_values)

# Q5
library(DESeq2)

design.matrix <- data.frame(exposure = c(1, 1, 1, 0, 0, 0))
counts.ds <- DESeqDataSetFromMatrix(countData = counts, colData = design.matrix, design = ~exposure)
res.ds <- DESeq(counts.ds)
results_ds <- results(res.ds, independentFiltering = FALSE, cooksCutoff = FALSE)
result_df <- as.data.frame(results_ds)
result_df <- result_df[order(result_df$padj), ]

print("Differentially Abundant OTUs:")
print(result_df)

significant_OTUs <- result_df[result_df$padj < 0.05, ]
print("Significant OTUs:")
print(significant_OTUs)

num_significant_OTUs <- nrow(significant_OTUs)
print(paste("Number of Significant OTUs:", num_significant_OTUs))

# Q6
top10_OTUs <- head(result_df, 10)
print("Top 10 most significant OTUs:")
print(top10_OTUs)

# Q7
counts_file <- "gene_counts.txt"
annotation_file <- "gene_annotation.txt"

gene_counts <- read.table(counts_file, header = TRUE, sep = "\t", quote = "", comment.char = "")
gene_annotations <- read.table(annotation_file, header = TRUE, sep = "\t", quote = "", comment.char = "")

print("Gene Counts (head):")
print(head(gene_counts))

print("Gene Annotations (head):")
print(head(gene_annotations))

total_counts <- colSums(gene_counts)
print("Total reads per sample:")
print(total_counts)

print("Annotations structure:")
print(str(gene_annotations))

filtered_counts <- gene_counts[rowSums(gene_counts) >= 5, ]
filtered_annotations <- gene_annotations[rownames(filtered_counts), ]
print(paste("Remaining genes after filtering:", nrow(filtered_counts)))

library(ggplot2)

counts_t <- t(filtered_counts)
pca_raw <- prcomp(counts_t, scale. = TRUE)
pca_data_raw <- data.frame(pca_raw$x, Group = c("HC", "HC", "HC", "LC", "LC", "LC"))

ggplot(pca_data_raw, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 4) +
  labs(title = "PCA (Raw Counts)", x = "PC1", y = "PC2") +
  theme_minimal()

counts_vst <- log(filtered_counts + 1)
counts_vst_t <- t(counts_vst)
pca_vst <- prcomp(counts_vst_t, scale. = TRUE)

pca_data_vst <- data.frame(pca_vst$x, Group = c("HC", "HC", "HC", "LC", "LC", "LC"))

ggplot(pca_data_vst, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 4) +
```

```r
  labs(title = "PCA (Log-Transformed Counts)", x = "PC1", y = "PC2") +
  theme_minimal()

# Q8
richness <- function(counts_sample) {
  richness_value <- sum(counts_sample$Count > 0)
  return(richness_value)
}

shannon_index <- function(counts_sample) {
  total_count <- sum(counts_sample$Count)
  p_i <- counts_sample$Count / total_count
  p_i <- p_i[p_i > 0]
  H_prime <- -sum(p_i * log(p_i))
  return(H_prime)
}

rarefied_gene_data_list <- list()
for (i in 1:ncol(filtered_counts)) {
  genes <- rownames(filtered_counts)
  counts_for_genes <- filtered_counts[, i]
  depth <- min(colSums(filtered_counts))
  rarefied_gene_data_list[[i]] <- rarefy_sample(genes, counts_for_genes, depth)
}

richness_values <- numeric(length(rarefied_gene_data_list))
shannon_values <- numeric(length(rarefied_gene_data_list))

for (i in 1:length(rarefied_gene_data_list)) {
  counts_sample <- rarefied_gene_data_list[[i]]
  richness_values[i] <- richness(counts_sample)
  shannon_values[i] <- shannon_index(counts_sample)
}

print("Richness Values for each sample:")
print(richness_values)

print("Shannon Index for each sample:")
print(shannon_values)

# Q9
library(DESeq2)

design.matrix <- data.frame(exposure = c(1, 1, 1, 0, 0, 0))
counts.ds <- DESeqDataSetFromMatrix(countData = filtered_counts, colData = design.matrix, design = ~exposure)
res.ds <- DESeq(counts.ds)
results_ds <- results(res.ds, independentFiltering = FALSE, cooksCutoff = FALSE)
```

```r

result_df <- as.data.frame(results_ds)
result_df <- result_df[order(result_df$padj), ]

print("Differentially Abundant Genes:")
print(result_df)

significant_genes <- result_df[result_df$padj < 0.05, ]
print("Significant Genes:")
print(significant_genes)

num_significant_genes <- nrow(significant_genes)
print(paste("Number of Significant Genes:", num_significant_genes))

genes_less_than_zero <- sum(significant_genes$log2FoldChange < 0)
genes_greater_than_zero <- sum(significant_genes$log2FoldChange > 0)
print(paste("Number of genes with log2FoldChange < 0:", genes_less_than_zero))
print(paste("Number of genes with log2FoldChange > 0:", genes_greater_than_zero))

gene_pdxA <- grep("TIGR00557", rownames(significant_genes), ignore.case = TRUE, value = TRUE)
pdxA_result <- result_df[rownames(result_df) %in% gene_pdxA, ]
print("pdxA Gene Result:")
print(pdxA_result)
```