Q1. Performing differential expression (DE) analysis.

Q1-1. Let's begin by identifying a set of features genes for differential expression analysis to speed up our analysis by excluding genes for which we are unlikely to have adequate power to detect differential expression. Identify how many genes are expressed in 5% or more of the cells found in your dataset. Hint: detect_genes annotates gene information in both colData (num_genes_expressed) and rowData (num_cells_expressed). [5 pts]

```
[1] "Number of genes expressed in 5% or more of the cells: 1439"
```

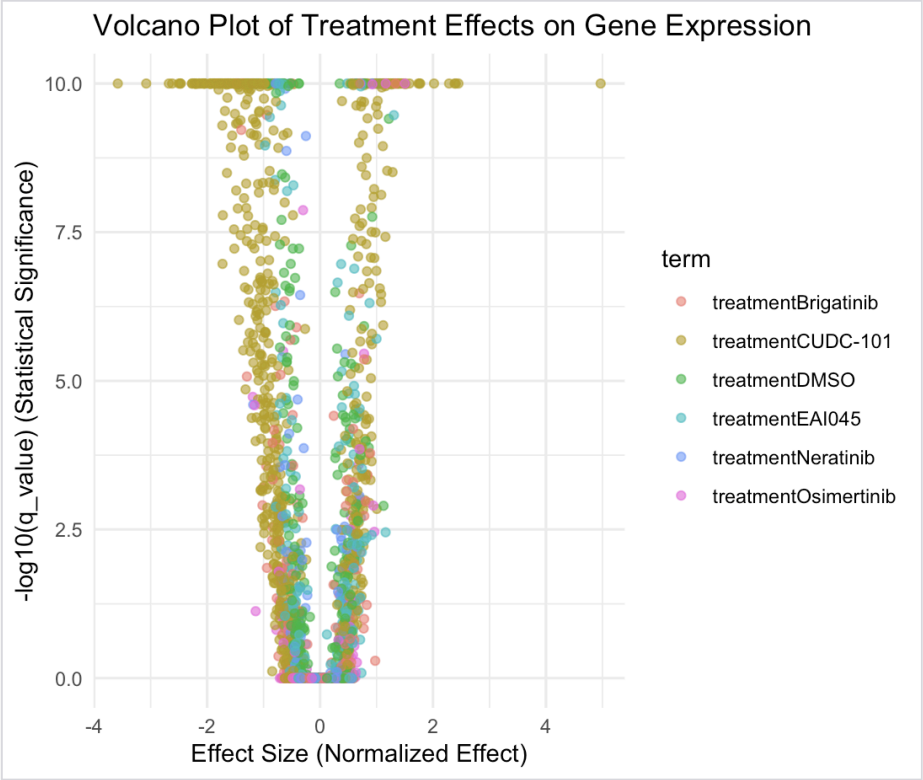
Q1-2. Generalized linear models (glm) can be used to identify genes whose expression is significantly different as a function of one or more covariates. Using the monocle3 function *fit_models* and the list of genes identified in **Q1-1**, perform a differential gene expression test to identify expressed genes that vary as a function of **chemical or genetic perturbation** as necessary for your dataset. Make sure you specify the proper column of colData in your model. The monocle3 function *coefficient_table*, tests whether the coefficients (model terms) in your model significantly deviate from zero. Examine the table returned by *coefficient_table* including the names of the columns. Provide the dimensions of the output below. After removing unnecessary data (see below), save this table for future use (using write.table for example) and submit as a separate tab-delimited text file in Courseworks. [5 pts].

Coefficient: `[1] 10073    15`

Filtered: `[1] 10073     8`

Q1-3. Using the results from your coefficient table (**Q1-2**) and focusing on the coefficients (model term) that describe the effect of each compound on gene expression (i.e., excluding the Intercept terms), plot the relationship between statistical significance (as the -log10 of the q_value) and effect size (normalized_effect) as a volcano plot. Briefly describe the relationship between these. [5 pts]

This plot implies that the treatments affect a broad range of genes significantly. The clustering near zero with high significance means there are small regulatory effects. The treatments may be targeting similar pathways/genes (as seen by overlap). This is typical in experiments where treatments modulate gene expression without large disruptions.



Q2. Examining DE results and performing gene set analysis.

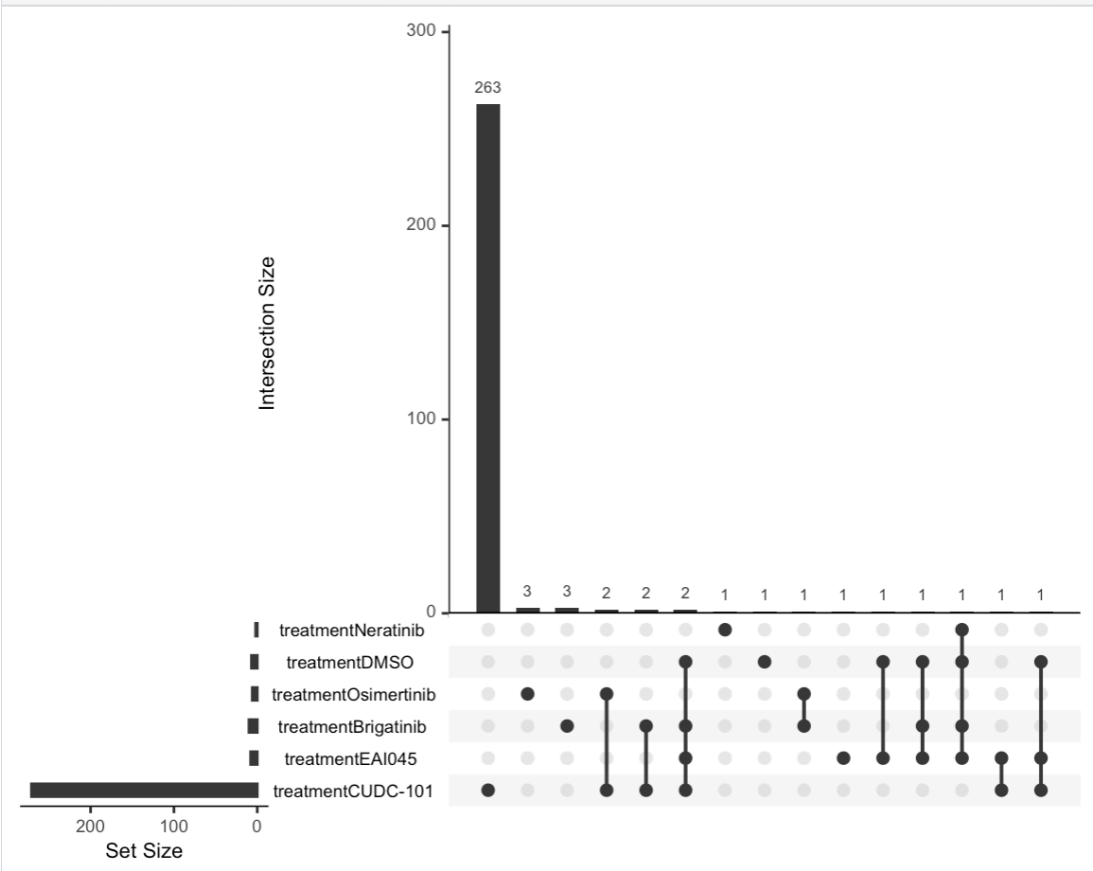
Q2-1. Identify the total number of genes differentially regulated by each compound or genetic perturbation at a multiple hypothesis testing adjusted p-value (i.e., q_value) cutoff of less than 0.05 and an absolute normalized effect size larger than 1. What type of changes are we focusing on by prioritizing genes that pass these cutoffs? [5 pts] Note: Exclude the Intercept term as in Q1-3 and group differentially expressed genes by term, which contains the name for each compound tested, to arrive at per compound differentially expressed genes.

	term	num_de_genes
	<chr>	<int>
1	treatmentBrigatinib	10
2	treatmentCUDC-101	271
3	treatmentDMSO	7
4	treatmentEAI045	8
5	treatmentNeratinib	2
6	treatmentOsimertinib	6

Prioritizing genes that pass the specified cutoff focuses on the biologically meaningful changes in gene expression. It is statistically significant and large enough to

have potential biological relevance. Higher counts (treatmentCUDC-101) suggests that this treatment has a more substantial change in gene expression across a larger number of genes. Lower counts (treatmentNeratinib) implies that this has fewer substantial impacts.

Q2-2. Let's examine the overlap between the differentially expressed genes (DEGs) per compound. An upset plot (example code at end of assignment) is a useful way to compare sets when the total number of comparisons is larger than 3 (i.e., where Venn diagrams get complicated). To create an upset plot, generate a binary matrix where columns are compounds or genetic perturbations, and rows are DEGs with entries set as 1 if the gene is a DEG for that compound and 0 if not. Order the intersects by frequency (*order.by = "freq"*).



The horizontal bars on the left show the total number of DEGs for each individual treatment. The vertical bars above the plot show the number of DEGs that overlap between specific combinations of treatments. This is ordered by frequency. At the bottom, each row represents a treatment and each column represents a specific intersection. In the columns where two or more dots are connected, the bar height represents the number of DEGs shared between those treatments.

Examine the set size, how does the total number of DEGs vary as a function of the compound used or gene perturbed? Examine the intersects, and describe the top 3 intersects across 2 or more compounds (overlap of DEGs) for your dataset. Describe any similarities or differences across compound or genetic perturbation-induced expression changes [10 pts]

The largest intersection (263 DEGs for treatmentCUDC-101) indicates most DEGs for this treatment are unique. The shared DEGs are relatively few, suggesting limited overlap in genes affected by different treatments. treatmentBrigatinib, treatmentDMSO, and treatmentEAI045 have moderate numbers of DEGs which indicates a smaller impact on gene expression. treatmentNeratinib and treatmentOsimertinib have the fewest DEGs, suggesting a more specific effect on gene expression.

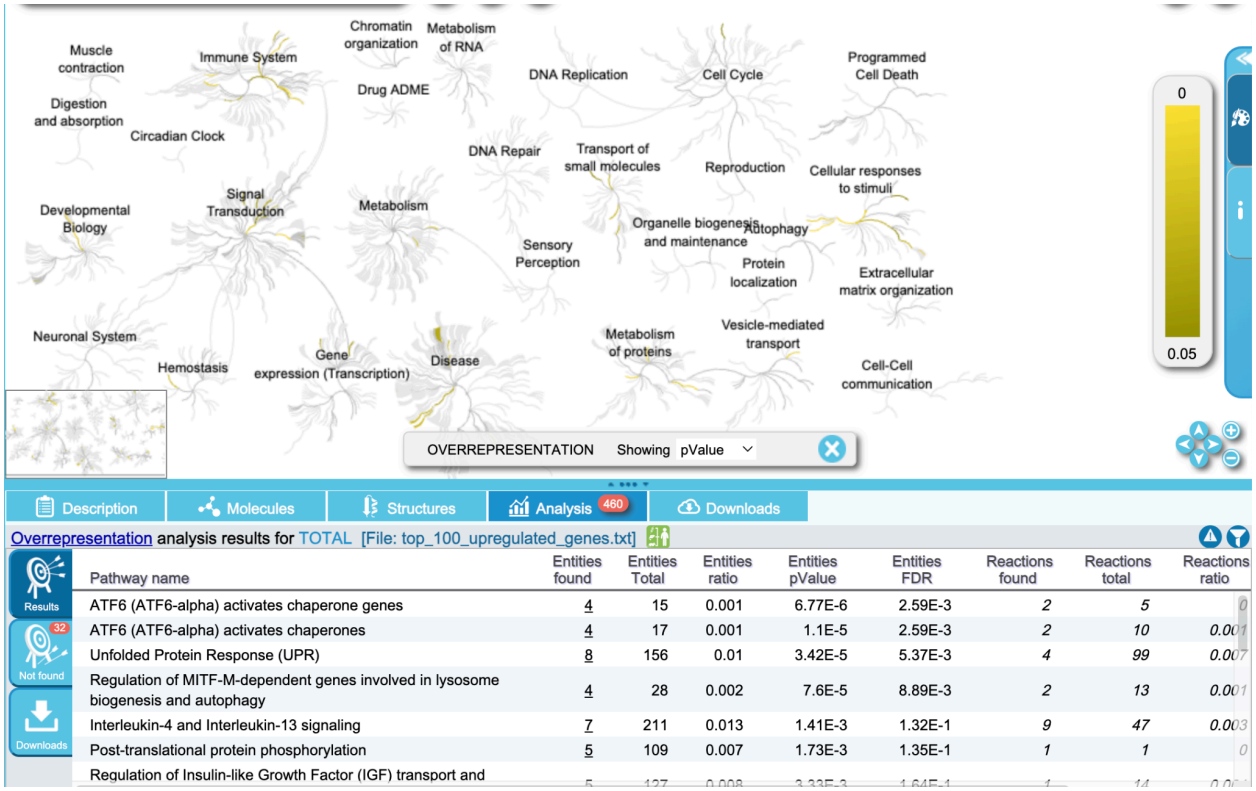
The largest intersection is between treatmentBrigatinib, treatmentCUDC-101, and treatmentDMSO. There can possibly be a common response pathway activated by all three. The second largest intersection is between treatmentCUDC-101, treatmentDMSO, and treatmentOsimertinib. Possibly, treatmentOsimertinib shares some impact with treatmentCUDC-101 and treatmentDMSO, maybe they modulate similar downstream pathways. The third largest intersection is between treatmentEAI045 and treatmentCUDC-101. They may share just a few common genes.

treatmentCUDC-101 stands out with the largest unique set of DEGs, indicating lots of gene expression changes, potentially impacting multiple pathways not affected by other compounds. treatmentNeratinib and treatmentOsimertinib have minimal overlap with other treatments, indicating selective gene expression changes that might be specific to particular pathways or cellular responses.

Q2-3. Let's examine whether the top differentially expressed genes contain information regarding changes in the activity of biological pathways upon inhibition of molecular targets by the compounds or genetic perturbations in your dataset. To simplify our search, let's focus on the effects of just one of the compounds or genetic perturbations in your dataset.

Identify the top 100 upregulated and top 100 downregulated genes for your compound or genetic perturbation and save their HUGO gene symbols (i.e., their gene_short_name). Using the Reactome Pathway Database Analysis tool

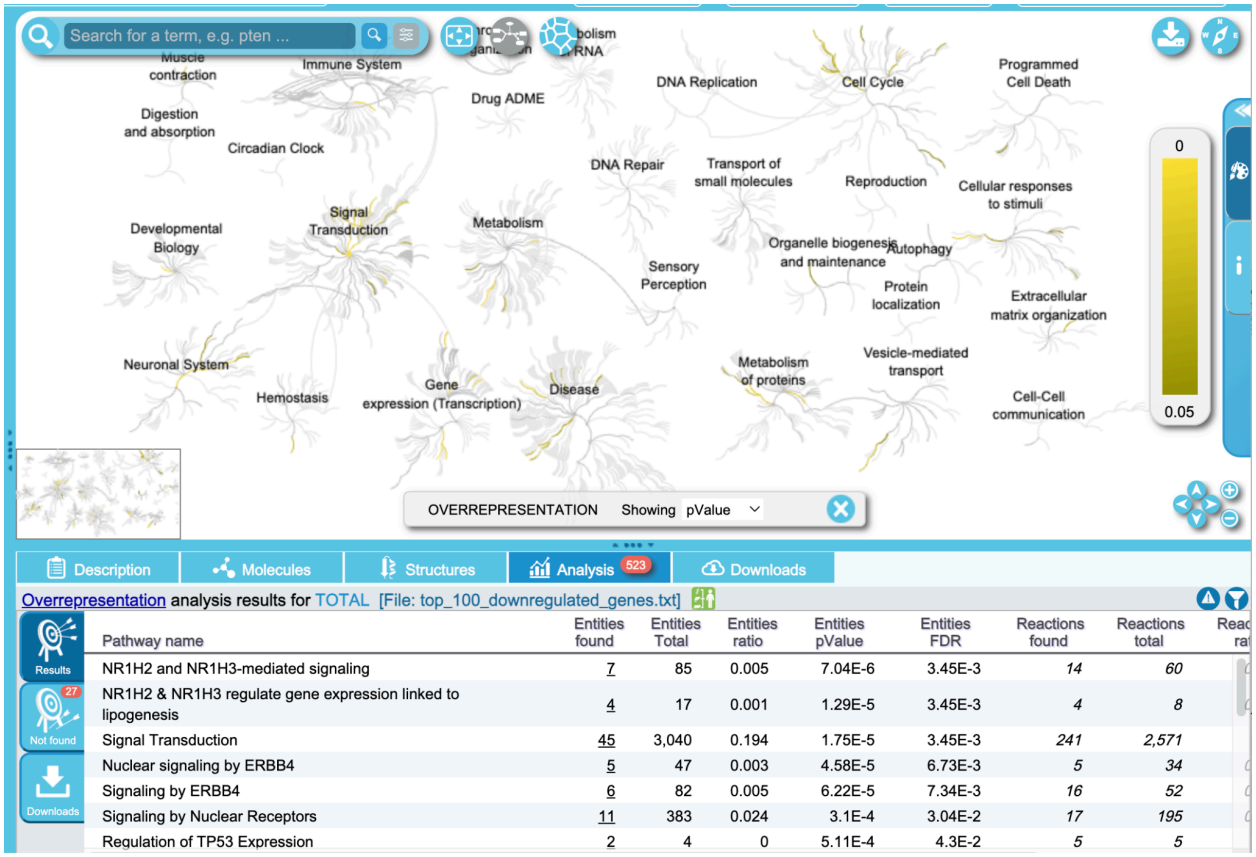
(https://reactome.org/PathwayBrowser/#TOOL=AT), identify gene terms associated with each set of genes. Examine and provide the top gene sets (e.g., the first 5 associated gene sets). What can you conclude of the effects of treatment or genetic perturbation on cells given the biological pathways that gene set analysis identifies as associated with your differentially expressed genes? [10 pts]



Pathway name	Entities found	Entities Total	Entities ratio	Entities pValue	Entities FDR	Reactions found	Reactions total	Reactions ratio
ATF6 (ATF6-alpha) activates chaperone genes	4	15	0.001	6.77E-6	2.59E-3	2	5	
ATF6 (ATF6-alpha) activates chaperones	4	17	0.001	1.1E-5	2.59E-3	2	10	0.00
Unfolded Protein Response (UPR)	8	156	0.01	3.42E-5	5.37E-3	4	99	0.00
Regulation of MITF-M-dependent genes involved in lysosome biogenesis and autophagy	4	28	0.002	7.6E-5	8.89E-3	2	13	0.00
Interleukin-4 and Interleukin-13 signaling	7	211	0.013	1.41E-3	1.32E-1	9	47	0.00
Post-translational protein phosphorylation	5	109	0.007	1.73E-3	1.35E-1	1	1	
Regulation of Insulin-like Growth Factor (IGF) transport and								

Upregulation:

- *ATF6 (ATF6-alpha) Activates Chaperone (Genes)* which assist in proper protein folding, particularly in times of endoplasmic reticulum stress. This can induce ER stress which in turn activates chaperones as a cellular response.
- *Unfolded Protein Response (UPR)* is related to ER stress when there is an accumulation of misfolded or unfolded proteins. This places stress on the ER due to an increase in protein synthesis demands or disruption of normal protein folding.
- *Regulation of MITF-M-dependent Genes Involved in Lysosome Biogenesis and Autophagy*. Upregulation means stimulation of autophagy and lysosome formation.
- *Interleukin-4 and Interleukin-13 Signaling* is related to immune signaling and inflammatory response. This can trigger immune/inflammatory pathways potentially as a response to stress damage done by the treatment.
- *Post-translational Protein Phosphorylation* is adding phosphate groups to protein after synthesis. This can stimulate signaling pathways that rely on phosphorylation to activate or deactivate proteins.



Pathway name	Entities found	Entities Total	Entities ratio	Entities pValue	Entities FDR	Reactions found	Reactions total
NR1H2 and NR1H3-mediated signaling	7	85	0.005	7.04E-6	3.45E-3	*14*	*60*
NR1H2 & NR1H3 regulate gene expression linked to lipogenesis	4	17	0.001	1.29E-5	3.45E-3	*4*	*8*
Signal Transduction	45	3,040	0.194	1.75E-5	3.45E-3	*241*	*2,571*
Nuclear signaling by ERBB4	5	47	0.003	4.58E-5	6.73E-3	*5*	*34*
Signaling by ERBB4	6	82	0.005	6.22E-5	7.34E-3	*16*	*52*
Signaling by Nuclear Receptors	11	383	0.024	3.1E-4	3.04E-2	*17*	*195*
Regulation of TP53 Expression	2	4	0	5.11E-4	4.3E-2	*5*	*5*

Downregulation:

- *NR1H2 and NR1H3-mediated signaling pathways* regulate cholesterol, lipid homeostasis, and inflammation. This treatment inhibits lipid metabolism and inflammation-related responses.
- *NR1H2 & NR1H3 regulate gene expression linked to lipogenesis*. This probably decreases the cell's ability to produce lipids, which impacts energy storage and membrane synthesis.
- *Signal transduction* pathways transmit signals from external stimuli to create intracellular responses. This can dampen the cell's responsiveness to external cues.
- *Nuclear signaling by ERBB4* (a receptor tyrosine kinase). This can inhibit pathways linked to cell proliferation/survival, which has treatment potential for tumor cells.
- *Signaling by ERBB4* in a broader context than just nuclear. This can reduce certain key survival and proliferative pathways.

Q3-1. You are interested in identifying genes responsible for a rare genetic disorder that you suspect is associated with germline mutations of a disease-causing gene. You have access to 5 sets of parent-child triads where only the offspring is affected by the disorder. To identify the disease-causing variant, you perform exome sequencing of the 15

individuals. How would you compare differences in coding regions within and between triads to hone in on the most likely causal gene? Do you expect the same mutation in all affected individuals? [10 pts]

First I would identify de novo mutations within each parent-child triad by sequencing the parents and the affected child in each triad. Then, I would compare across the five triads to look for any recurring mutations or affected genes among the different children. I will focus on genes that have mutations in more than one child. I will also search for rare inherited variants present in the children but absent in the parents. The exact mutation may not be the same in all affected children by the same gene/pathway can be affected in multiple cases. That is why it is important to focus on identifying common genes affected by different mutations across triads.

Q3-2. You are interested in defining the regulation of the *TCF7L2* locus from Assignment 1. Specifically, you would like to identify non-coding loci that interact with the *TCF7L2* locus and may vary across diabetes patients. Which general class of methods would you choose to answer this question? Which in particular do you believe is best suited for un-biasedly identifying these elements? After identification, which subsequent method could you use to validate interactions for your most promising hits? [10 pts]

A general class of methods I would choose is a 3D chromatin interaction method like Hi-C or Capture-C to map regulatory regions that spatially interact with the TCF7L2 locus. This is to identify distal non-coding elements by bringing them into close proximity within the nucleus. In particular, I would choose Hi-C since it provides a comprehensive genome-wide approach that captures all possible chromatin interactions while avoiding bias towards a specific loci. After identification, I would use CRISPR-based validation techniques like CRISPR interference or CRISPR activation. This is to assess the functional impact of the regions on TCF7L2 expression.

---
title: "Assignment 2"
output: html_document
date: "2024-10-09"

```
#Load libraries
library(monocle3)
library(Matrix)
library(dplyr)
library(tidyr)
library(ggplot2)
library(UpSetR)

#Define file paths
cell_metadata_path <- "~/Documents/Columbia/Sem 3/Funct
Genomics/Dataset2/dataset2_final_cellmeta.csv"
gene_metadata_path <- "~/Documents/Columbia/Sem 3/Funct
Genomics/Dataset2/dataset2_final_genemeta.csv"
expression_matrix_path <- "~/Documents/Columbia/Sem 3/Funct
Genomics/Dataset2/dataset2_final.mtx"

#Load data
cell_metadata <- read.csv(cell_metadata_path, row.names = 1)
gene_metadata <- read.csv(gene_metadata_path, row.names = 1)
expression_matrix <- readMM(expression_matrix_path)

#Data set and genes
cds <- new_cell_data_set(expression_data = expression_matrix,
                         cell_metadata = cell_metadata,
                         gene_metadata = gene_metadata)
cds <- detect_genes(cds)

# Q1-1
num_cells <- ncol(cds)
gene_counts <- rowData(cds)$num_cells_expressed
filtered_genes <- rowData(cds)[gene_counts >= 0.05 * num_cells, ]
num_filtered_genes <- nrow(filtered_genes)
print(paste("Number of genes expressed in 5% or more of the cells:", num_filtered_genes))
```

```
# Q1-2
gene_ids <- rownames(filtered_genes)
model_formula_str <- "~treatment"
fit_results <- fit_models(cds[gene_ids, ], model_formula_str = model_formula_str)
fit_results_df <- coefficient_table(fit_results)

# Save and load filtered fit results
filtered_fit_results <- fit_results_df[, c("id", "gene_short_name", "term", "estimate", "std_err",
"p_value", "q_value", "normalized_effect")]
write.table(filtered_fit_results, file = "filtered_fit_results.txt", sep = "\t", row.names =
FALSE, quote = FALSE)
filtered_fit_results <- read.table("filtered_fit_results.txt", sep = "\t", header = TRUE)


# Q1-3: Volcano plot of treatment effects
treatment_effects <- fit_results_df[fit_results_df$term != "(Intercept)", ]
treatment_effects$log10_q_value <- -log10(treatment_effects$q_value + 1e-10)
ggplot(treatment_effects, aes(x = normalized_effect, y = log10_q_value, color = term)) +
  geom_point(alpha = 0.6) +
  labs(title = "Volcano Plot of Treatment Effects on Gene Expression", x = "Effect Size
(Normalized Effect)", y = "-log10(q_value)") +
  theme_minimal()


# Q2-1: Count DEGs by treatment
de_gene_counts <- fit_results_df %>%
  filter(term != "(Intercept)" & q_value < 0.05 & abs(normalized_effect) > 1) %>%
  group_by(term) %>%
  summarize(num_de_genes = n())
print(de_gene_counts)


# Q2-2: UpSet plot of DEG overlap
binary_matrix <- differentially_expressed_genes %>%
  select(gene_short_name, term) %>%
  mutate(value = 1) %>%
  pivot_wider(names_from = term, values_from = value, values_fill = list(value = 0))
binary_matrix <- as.data.frame(binary_matrix)
```

```
rownames(binary_matrix) <- binary_matrix$gene_short_name
binary_matrix$gene_short_name <- NULL
upset(binary_matrix, order.by = "freq", sets = colnames(binary_matrix), keep.order =
TRUE)


# Q2-3: Save top 100 upregulated/downregulated genes for selected compound
selected_compound <- "treatmentCUDC-101"
selected_degs <- fit_results_df %>%
  filter(term == selected_compound & q_value < 0.05 & abs(normalized_effect) > 1) %>%
  arrange(desc(normalized_effect))
top_100_upregulated <- selected_degs %>%
  top_n(100, normalized_effect) %>%
  pull(gene_short_name)
top_100_downregulated <- selected_degs %>%
  top_n(-100, normalized_effect) %>%
  pull(gene_short_name)
write.table(top_100_upregulated, file = "top_100_upregulated_genes.txt", row.names =
FALSE, col.names = FALSE, quote = FALSE)
write.table(top_100_downregulated, file = "top_100_downregulated_genes.txt",
row.names = FALSE, col.names = FALSE, quote = FALSE)
```