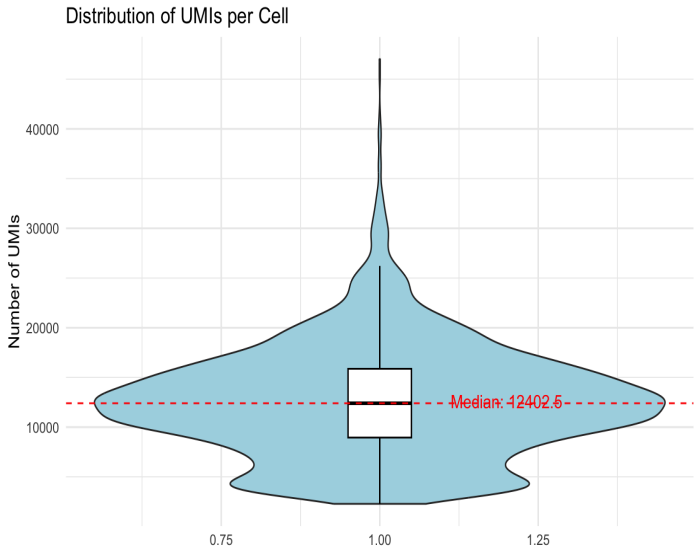
Q1. Dataset QC.

Q1-1. The expression matrix of your datasets contains the number of each gene transcript detected per cell.What are the dimensions of your dataset? Hint: In monocle3, you can access your expression matrix using the function exprs(). [5 pts]

```
The expression matrix has 8563 genes and 4192 cells.
```

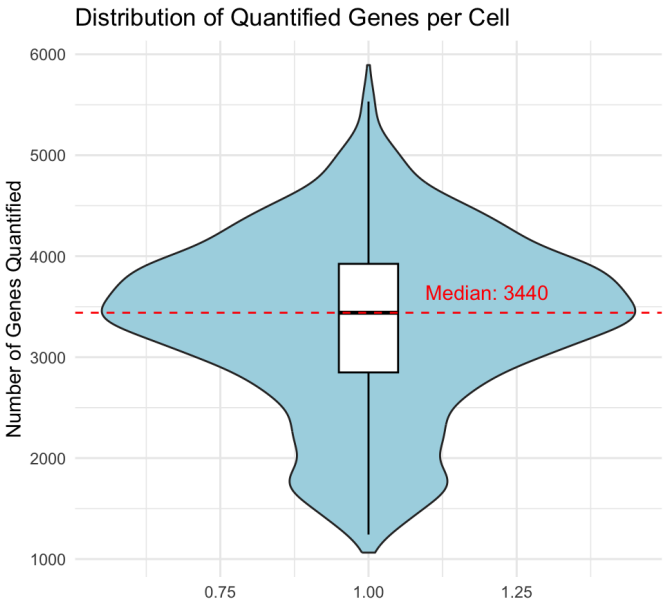
Q1-2. How many cells are in your dataset? What is the median number of unique transcripts (UMIs) per cell for your experiment? Provide the distribution of UMIs per cell in the form of a violin plot. The n.umi column in colData provides the total number of UMIs per cell. [5 pts] Bonus: Overlay (plot) the median number of UMIs per cell on top of your violins. Hint: Overlaying a boxplot could work. [5 pt]

```
[1] "Total number of cells: 4192"
[1] "Median number of UMIs per cell: 12402.5"
```



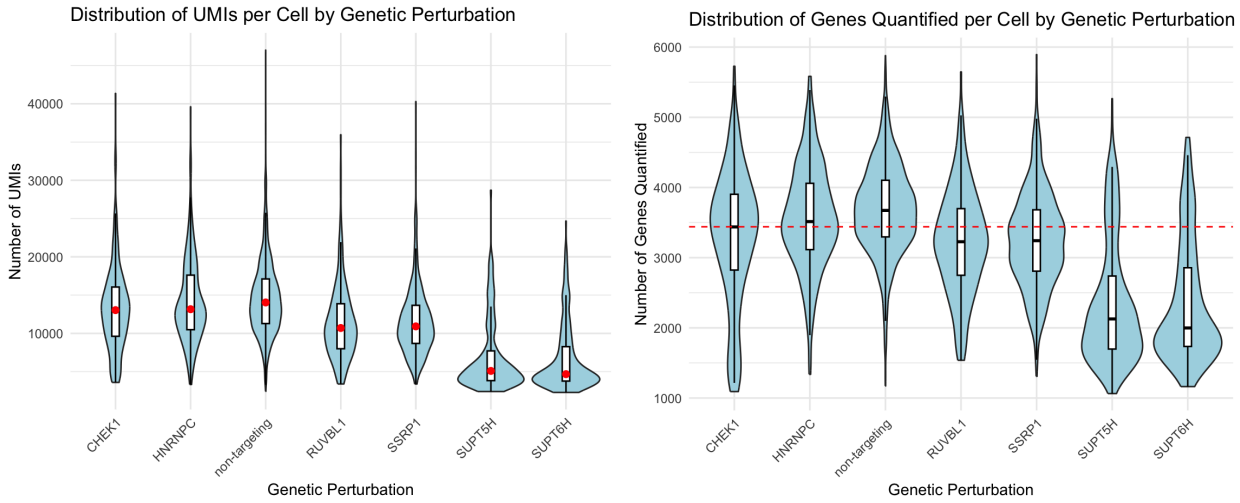
Q1-3. How many genes are in your dataset? What is the median number of genes quantified per cell? The function detect_genes annotates rowData to provide information on the number of cells in which each gene is expressed and annotates colData to provide information on the number of genes quantified in each cell. Provide the distribution of the number of capture genes per cell in the form of a violin plot. [5 pts] Bonus: Overlay (plot) the median number of genes per cell on top of your violins. Hint: Overlaying a boxplot could work. [5 pt]

```
[1] "Total number of genes: 8563"
[1] "Median number of genes per cell: 3440"
```



Q1-4. What unique genetic perturbations (dataset 1: crispr_target) or treatments (dataset 2: treatment) are present in your dataset? Generate a summary of the total number of cells for every treatment. Recreate your violin plots for the distribution of UMIs and genes per cell (**Q1-2**, **Q1-3**) with your plot containing a violin for every genetic perturbation or treatment. [5 pts]

	Genetic_Perturbation	Number_of_Cells
1	CHEK1	295
2	HNRNPC	334
3	non-targeting	2096
4	RUVBL1	211
5	SSRP1	562
6	SUPT5H	406
7	SUPT6H	288



Q1-5. Briefly (in 1-2 sentences) describe the biological pathways that have been previously associated with each genetic perturbation or the mechanism of action for each drug treatment in your respective dataset (i.e., what are their known roles). (Note: focus on the treatment or genotype that is not a non-targeting control or DMSO vehicle treatment). [5 pts]

- **CHECK1**: checkpoint kinase 1 plays a key role in damage response and cell cycle control, involved in ATR
- **HNRNPC**: involved in RNA splicing and processing, plays role in mRNA metabolism and influencing alternative splicing
- **RUVBL1**: part of the chromatin remodeling complex, plays a role in DNA repair, transcription regulation, and chromatin remodeling
- **SSRP1**: subunit of FACT complex, involved in chromatin reorganization during transcription, replication, and repair processes
- **SUPT5H**: transcription elongation factor that modulates rate of transcription elongation and it interacts with RNA polymerase II
- **SUPT6H**: transcription elongation factor that plays a role in chromatin organization by interacting with histones to maintain chromatin structure

Q2. Concepts from class.

Q2-1. The telomere-to-telomere (T2T) consortium aims to create a “gapless” sequence of the human genome. A major goal is to assemble large regions of highly repetitive sequences that could not be unambiguously mapped by the human genome project from Sanger sequencing. Which of the next-generation sequencing technologies discussed in class would you choose to increase the probability of obtaining a correct genome

assembly for these regions? Briefly describe your reasoning. [10 pts] Hint: The read lengths in Sanger sequencing rarely exceed 1 kb.

Nanopore and PacBio sequencing are good choices in order to generate a gapless human genome sequence. They are long-read sequencing technologies. They are capable of reading tens of kilobases in length, unlike Sanger sequencing which is limited to read lengths of less than 1kb. Nanopore sequencing passes DNA molecules through a nanopore and then it measures changes in current. PacBio sequencing generates long reads through single molecular real-time sequencing (SMRT), which provides high quality and long reads.

Q2-2. You performed a genome-wide association study across cases and controls to identify genomic regions associated with an increased risk of atherosclerosis. In total, you assayed 1 x 10^6 single nucleotide polymorphisms (SNPs), and analysis of association identified SNPs in linkage disequilibrium with the *IRF6* and *BHMT* as the two loci with the highest deviation in proportion across cases and controls from the expected (*IRF6*: *p-value* = 2.4 x10^{-7}, *BHMT*: *p-value* = 5.8 x 10^{-24}). After correction for multiple hypothesis testing by the Bonferroni method, which of these would you consider significant hits. Do your GWAS results alone confirm the involvement of that gene in the disease? What are other possibilities? [10 pts]

The Bonferroni-corrected p-value threshold ends up being 5x10^-8. The IRF6 p-value is greater, which makes it not significant. The BHMT p-value is significant, since it is smaller than the threshold. The GWAS results alone do not confirm the involvement of IRF6 or BHMT in atherosclerosis. It identifies regions associated with a trait but it does not establish a direct causal link. Linkage disequilibrium could explain the association, so could population stratification. Linkage disequilibrium is when nearby SNPs are inherited together. Population stratification is differences in ancestry between cases and controls.

Assignment 1 code

2024-09-22

#Q1-1 library(monocle3) cds <- readRDS("~/Documents/Columbia/Sem 3/Funct Genomics/Dataset1/dataset1_final.rds") expr_matrix <- exprs(cds)

#Dimensions dimensions <- dim(expr_matrix) cat("The expression matrix has", dimensions[1], "genes and", dimensions[2], "cells.")

#Q1-b library(ggplot2) library(readr)

#Number of cells cell_meta_df <- read_csv('~/Documents/Columbia/Sem 3/Funct Genomics/Dataset1/dataset1_final_cellmeta.csv') num_cells <- nrow(cell_meta_df)

#Median number of UMIs per cell median_umis <- median(cell_meta_df$n.umi) print(paste("Total number of cells:", num_cells)) print(paste("Median number of UMIs per cell:", median_umis))

#Violin plot with boxplot overlay and median line ggplot(cell_meta_df, aes(x = 1, y = n.umi)) + # Set x to 1 as a dummy variable geom_violin(fill = "lightblue") + geom_boxplot(width = 0.1, color = "black", outlier.shape = NA) + geom_hline(yintercept = median_umis, color = "red", linetype = "dashed") + labs(title = "Distribution of UMIs per Cell", y = "Number of UMIs", x = "") + theme_minimal() + annotate("text", x = 1.2, y = median_umis + 200, label = paste("Median:", round(median_umis, 1)), color = "red")

#Q1-c cds <- readRDS('~/Documents/Columbia/Sem 3/Funct Genomics/Dataset1/dataset1_final.rds') cds <- detect_genes(cds) num_genes <- nrow(rowData(cds))

#Number of genes genes_per_cell <- colData(cds)$num_genes_expressed

#Median number of genes quantified per cell median_genes_per_cell <- median(genes_per_cell) print(paste("Total number of genes:", num_genes)) print(paste("Median number of genes per cell:", median_genes_per_cell))

#Violin plot with boxplot overlay and median line for genes per cell ggplot(data = as.data.frame(genes_per_cell), aes(x = 1, y = genes_per_cell)) + geom_violin(fill = "lightblue") + geom_boxplot(width = 0.1, color = "black", outlier.shape = NA) + geom_hline(yintercept = median_genes_per_cell, color = "red", linetype = "dashed") + labs(title = "Distribution of Quantified Genes per Cell", y = "Number of Genes Quantified", x = "") + theme_minimal() + annotate("text", x = 1.2, y = median_genes_per_cell + 200, label = paste("Median:", round(median_genes_per_cell, 1)), color = "red")

#Q1-d #Column names colnames(colData(cds)) unique_perturbations <- unique(colData(cds)$crispr_target)

#Total number of cells for each genetic perturbation summary_table <- table(colData(cds)$crispr_target) summary_df <- as.data.frame(summary_table) colnames(summary_df) <- c("Genetic_Perturbation", "Number_of_Cells") print(summary_df)

#Q1-d-b #Genetic perturbations valid_perturbations <- c(“CHEK1”, “HNRNPC”, “non-targeting”, “RUVBL1”, “SSRP1”, “SUPT5H”, “SUPT6H”)

#Filter dataset to valid perturbations filtered_data <- colData(cds)[colData(cds)$crispr_target %in% valid_perturbations,]

#Violin plot for distribution of UMIs per cell by genetic perturbation ggplot(as.data.frame(filtered_data), aes(x = crispr_target, y = n.umi)) + geom_violin(fill = “lightblue”) + geom_boxplot(width = 0.1, color = “black”, outlier.shape = NA) + stat_summary(fun = median, geom = “point”, shape = 20, size = 3, color = “red”, show.legend = FALSE) + labs(title = “Distribution of UMIs per Cell by Genetic Perturbation”, y = “Number of UMIs”, x = “Genetic Perturbation”) + theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1))

#Q1-d-c #Number of genes (rows in rowData) and median genes per cell (colData) num_genes <- nrow(rowData(cds)) genes_per_cell <- colData(cds)$num_genes_expressed median_genes_per_cell <- median(genes_per_cell)

cat(“Total number of genes:”, num_genes, “”) cat(“Median number of genes per cell:”, median_genes_per_cell, “”)

#Violin plot for number of genes per cell by genetic perturbation library(ggplot2)

ggplot(as.data.frame(colData(cds)), aes(x = crispr_target, y = num_genes_expressed)) + geom_violin(fill = “lightblue”) + geom_boxplot(width = 0.1, color = “black”, outlier.shape = NA) + geom_hline(yintercept = median_genes_per_cell, color = “red”, linetype = “dashed”) + labs(title = “Distribution of Genes Quantified per Cell by Genetic Perturbation”, y = “Number of Genes Quantified”, x = “Genetic Perturbation”) + theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1))