Introduction: For this project, single cell RNA Sequencing (scRNAseq) Quality Control Data was analyzed using a toolkit package for R Studio called Seurat. Regarding sample preparation, cells were collected from control and 5XFAD (Alzheimer's presenting) mice, fixed, and dissociated. Shortly after, samples were processed using different steps including probe hybridization, genetic barcoding, gene pre-amplification, gene library construction, and sequencing. As a result, final sequencing quality control data is analyzed using R Studio. In the following project, I demonstrate the process of using Seurat to analyze the Quality Control of my first time sequencing.

Data Set Description: For this analysis, one control sample was analyzed, so I could learn how to use the packages and ML system in isolation of multiple data sets. Here, the dataset comprises of 6 features that include seq_folder (sample identification), nUMI (number of Unique Molecular Identifiers), nGene (number of Genes), log10GenePerUMI (log 10 normalized Genes for every Unique Molecular Identifier), percent.mito (mitochondria concentration), and cells (defined by matching probe pair sequences). For every feature, there was 1,236,598 observations before preprocessing. After preprocessing,

there was 4,850.

Experimental Design: Before Quality Control and Data Analysis could be assessed, necessary libraries were loaded. These included Seurat, Matrix, dplyr, and ggplot2. As previously mentioned, Seurat is the data analysis toolkit used here for scRNAseq analysis. Matrix was used for sparse and dense matrix classes, dplyr was used for data manipulation, and ggplot2 was used for creating graphics.

Next, the dataset was downloaded using the Read10X function as part of the Seurat package. To begin with, data is stored as a count matrix, which must be converted to a readable object by Seurat. So, data is turned into a Seurat Object using the CreateSeuratObject function. Next, the content of the metadata was checked.

Before Quality Control metrics were assessed, some preprocessing was necessary to filter out "empty read" droplets. These droplets are considered background reads that are not specific to the cells being analyzed. Additionally, other cells were dropped from the data set that do not fit the mini-

mum requirements for successful downstream analysis. These metrics include nUMI, log10GenePerUMI, and percent Mitochondrial Gene expression. Here, observations were thrown out if the nUMI was less than or equal to 500, log10GenePerUMI was less than 0.8, and mitochondrial gene expression was greater than 0.10.

After these initial set-ups, quality metrics were assessed. These metrics include Cell Count, UMI counts per cell, Genes detected per cell, Complexity, and Mitochondrial counts ratio.

First, the number of cells were determined by running a line of code that identified unique cellular barcodes detected. Here, ggplot2 was used to plot the number of cells against the sample of interest(Figure 1). To ensure my code and analysis was correct, this value was compared to the estimated cell count generated by the sequencing machine as seen in Figure 2. Both the Seurat analysis and sequencing machine analysis gathered a similar cell count. Next, ggplot2 was used to show a visualization of the number of Unique Molecular Identifiers Figure 3) , genes detected per cell (Figure 4), complexity (Figure 5),

and mitochondrial gene expression (Figure 6).

Once these metrics were complete, a joint plot comparing the number of Unique Molecular Identifiers (UMIs) to the number of Genes was created (Figure 7). Mitochondrial gene presence is also seen in this joint plot. The darker the presence of the dot, the higher the presence of mitochondrial DNA. For this analysis, a lower percent of mitochondrial presence is ideal to ensure high quality cells ($< 10\%$).

Next, highly variable features were identified by feature selection. Here, this was done by creating a subset of features that only included genes. As mentioned previously, there were two measurements of gene expression: nGene (number of genes) and log10GenePerUMI. For the purpose of downstream analysis, log-normalized gene expression was stabilized using a variance-stabilizing function to make up for the mean-variance relationship that is lost without this correction. Next, to evaluate the mean-variance relationship, the mean and variance of individual genes was computed from unnormalized data and log transformed afterwards. Once this is complete, a curve is fit to predict vari-

ance in relation to the mean, and this is used to standardize the feature counts. This method avoids removing cases of high cell-cell gene variation. Taking into consideration outliers, standardized values were set to reach a maximum of $\sqrt{N}$ with N defining the total number of cells analyzed. This removed influence from technical outliers. From here, variance was calculated for each gene across all cells analyzed. The subset of features that contain high cell-cell variation was calculated and plotted (Figure 8). The variable count represents genes that show high expression in some cells, but low expression in others. The non-variable count represents genes that have little to no variation in expression from cell to cell.

Before more metrics were observed, data was log normalized. This was done to make expression counts comparable across genes via systematic variation adjustments. Gene reads were then proportional to other factors such as RNA reads. Data was also scaled. Next, a PCA algorithm was ran to analyze variability and similarity. This helps aid in visualization of similar cell types later. Then, the dimensionality of the data set was determined. This function in Seurat clusters cells based off of the principal component score (PCA). This

graph helps determine how many principal components to include in further downstream analysis. To graph this, an elbow plot was created that showed the ranks of principal components based on the percentage of overall variance (Figure 9). The number of PCs to include for further analysis is then chosen by where the elbow graph plateaus. This number was chosen to create the next visualization.

The last piece of quality control is creating a non-linear dimensional reduction to view the clustering of the sequencing samples. First, a Find Neighbors and Find Clusters algorithm was ran. For the Find Neighbors, rhe dims ratio was set based on the previous elbowplot plateau. Here, the PC chosen was 21. Next, a UMAP graph was made for cluster visualization (Figure 10). Here, the UMAP plots the structure of the dataset and groups cells with a similar profile together in dimensional space. Thus, cells that were grouped in the process above should plot near one another in the UMAP. This graph shows a qualitative look at how many clusters (different types of cells) were present in my sequencing sample, and sets up the stage for further cell identification after quality control.

Conclusions: After analyzing my first try at sequencing a few conclusion can be made. Once preprocessing was used to filter out cells that were not usable for downstream analysis, the number of observations went from 1,236,598 to 4,850. In order to improve the number of usable cells I can analyze, I plan on looking at my protocol and developing a plan to increase the number of total cells for analysis. An increase in total number of cells should increase the number of observations I can use for analysis. Secondly, of the cells that were usable, the gene count, mitochondrial gene expression percentage, complexity, and UMI count was good, signifying efficient and accurate completion of sample preparation. Thus, to improve my results in the future, finding a way to increase total observations would improve downstream analysis and results.

Results:

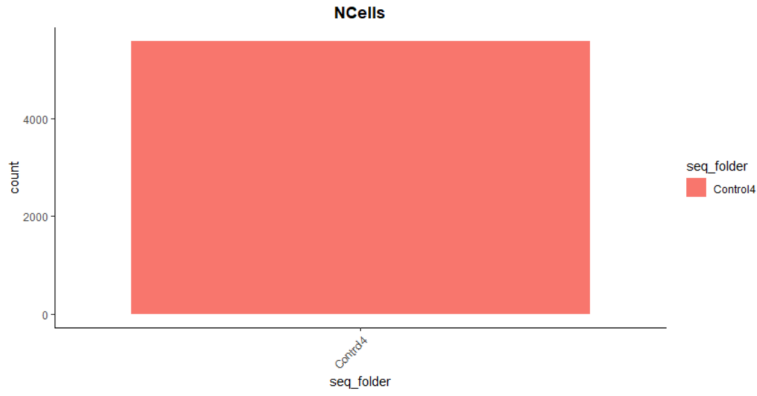

Figure 1: Seurat Calculated Sample Cell Count

Cells

Cells	Median reads per cell	Median genes per cell	Total genes detected	Median UMI counts per cell	Confidently mapped reads in cells
4,851	44,942	5,277	16,955	17,834	93.70%

Figure 2: Summary Generated Sample Cell Count

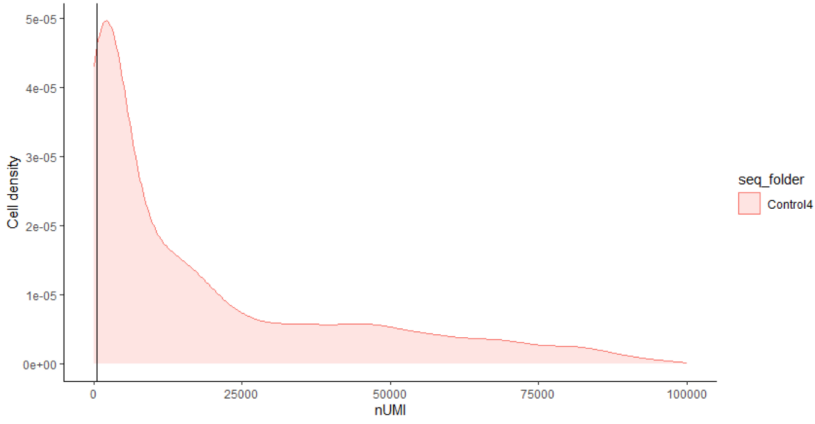

Figure 3: Number of Unique Molecular Identifiers (UMIs) per Cell

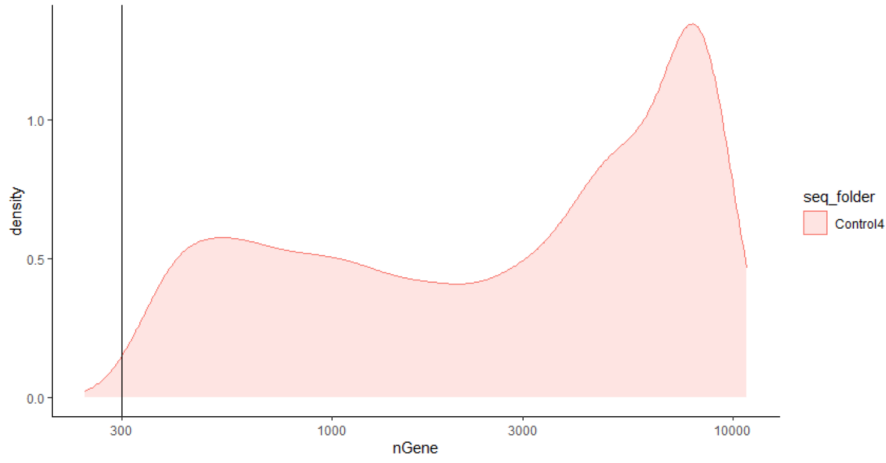

Figure 4: Number of Genes per Cell

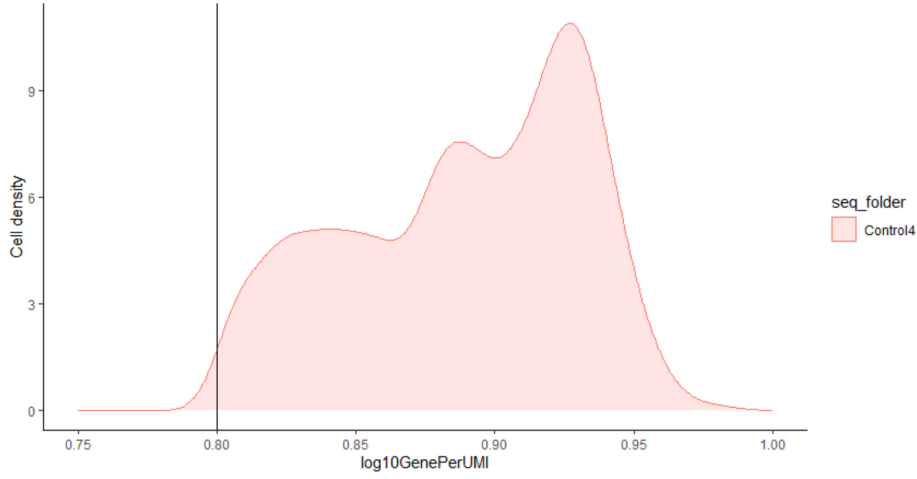

Figure 5: Number of Genes per UMI (log 10) per Cell

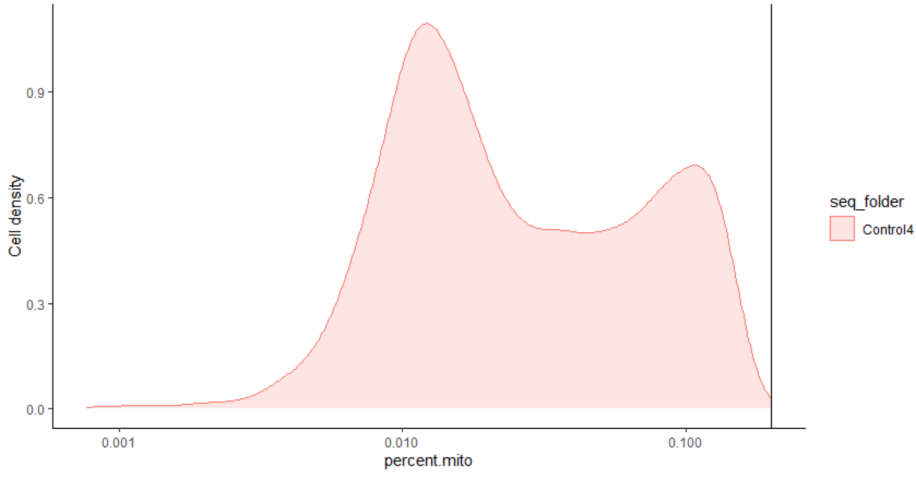

Figure 6: Percent Mitochondrial Genes per Cell

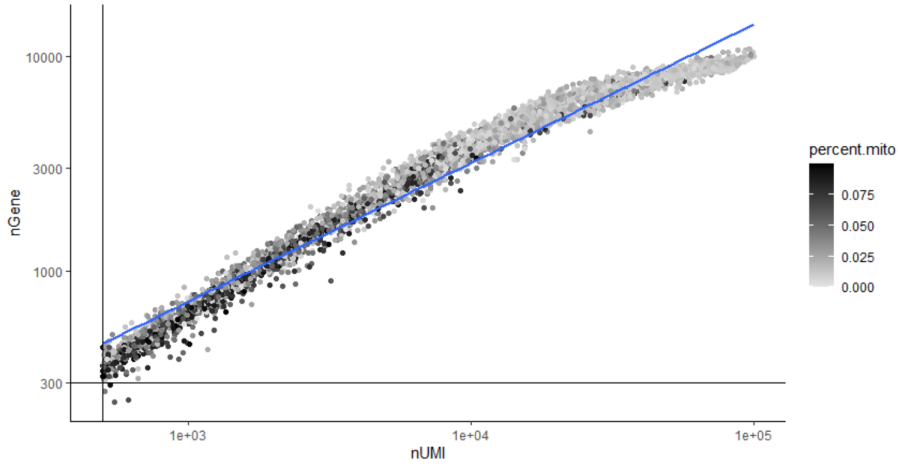

Figure 7: Number of Unique Molecular Identifiers (UMIs) Compared to Number of Genes. Representative Mitochondrial Gene Expression is Represented by the Gray Scale.

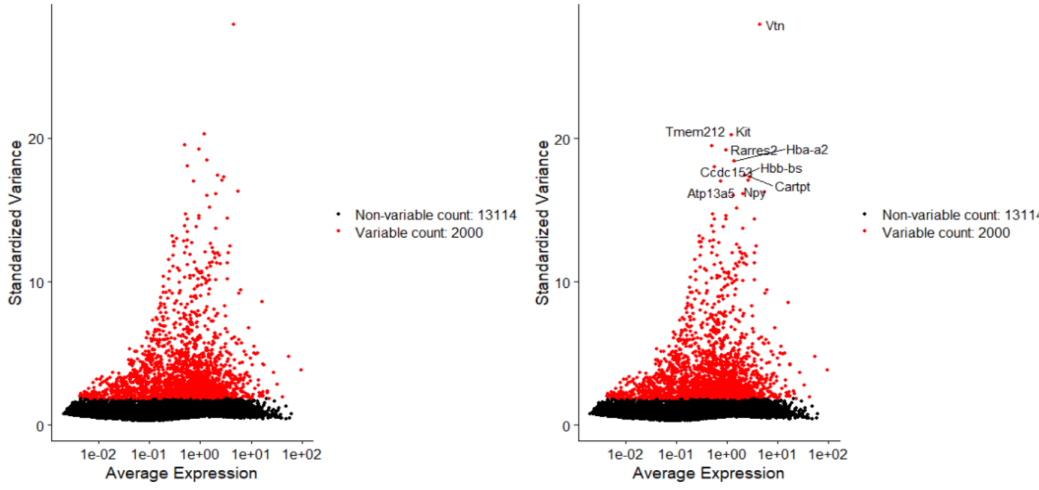

Figure 8: Cell-Cell Gene Variation

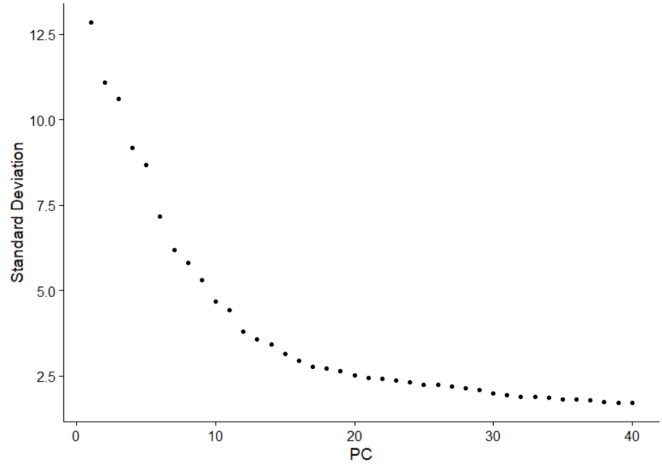

Figure 9: PCA Elbow Plot

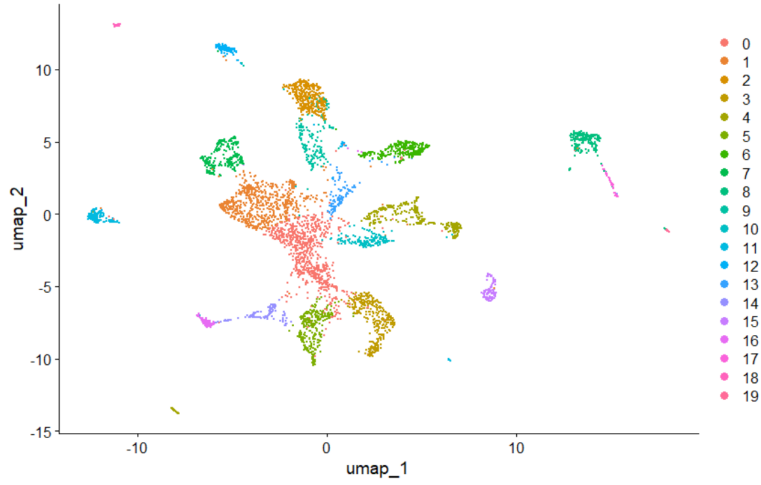

Figure 10: UMAP Representation of Cell Clusters

References:

1. "Getting Started with Seurat." Getting Started with Seurat • Seurat, satijalab.org/seurat /articles/get_started.html. Accessed 12 Apr. 2024.

2. "Single-Cell RNA-Seq Data Analysis Workshop." Introduction to Single-Cell RNA-Seq, hbctraining.github.io/scRNA-seq_online/schedule/links-to-lessons.html. Accessed 12 Apr. 2024.

I would also like to thank Derek Vern Walton for introducing some of the concepts.