**Diane scRNA-seq Analysis (original box note)**

* Fastq files were processed locally on 32 cores with CellRanger 6.1.2 with the most up-to-date human genome reference (GRCh-2020-A) available on the 10X Genomics site.
* After both CellRanger runs were completed, subsequent QC and analysis were completed in R (4.1.2) with the Seurat (4.0.5) and tidyverse (1.3.1) packages.
* Sequencing metric outputs from the CellRanger run were parsed into one table for both samples and that is available [here.](https://nevada.app.box.com/file/891500170950)
* Cell count plots for each biopsy sample that are recreations of the HTML CellRanger [outputs](https://nevada.app.box.com/folder/151490679941) were plotted and are [here](https://nevada.app.box.com/file/892065648874) and [here](https://nevada.app.box.com/file/892067237138).
* Low quality and/or dying cells often exhibit high mitochondrial gene contamination. The mitochondrial gene content of each sample was then calculated:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Min | 1st Qu. | Median | Mean | 3rd Qu. | Max |
| 0.000 | 2.955 | 4.267 | 4.394 | 5.705 | 9.959 |

* A violin plot of some other QC metrics (RNA features, RNA counts, and mitochondrial %) was then generated [here](https://nevada.app.box.com/file/892066451733) before other QC could be done.
* Ridgeplots of those same metrics were made [here](https://nevada.app.box.com/file/892069147595).

Filtering

* Based on plots of the above metrics (nCounts, nFeatures, percent mito.), the following cutoffs were used for filtering to eliminate the most noise while still retaining good cells with many features:
  + percent mitochondrial genes max of 10%.
  + Unique UMI counts under 500 or greater than 10000.
  + Contain at least 300 features within them.
* After filtering, went from:
  + biopsy20210817: 344 --> 271 cells
  + biopsy20210909: 2413 --> 1024 cells
    - These filtering losses seem to be in line with many online tutorials, with the 10X example data filtering ranging from ~65% to ~13% total cell loss
* A ridgeplot of the samples after filtering was plotted to make sure that not too many good cells were being lost post-filtering and those are avialable [here](https://nevada.app.box.com/file/892067328157).
* After filtering out cells from the dataset, the data was normalized. By default, the Log-normalizationhere method is used.
  + LogNormalize that normalizes the gene expression measurements for each cell by the total expression, multiplies this by a scale factor (10,000 here), and then log-transforms the data.
* Lastly variable genes within the new fitlered,normalized dataset were calculated and plotted. By default, the top 2000 variable genes were plotted and colored red. That plot is available [here.](https://nevada.app.box.com/file/892067627048)
* To visualize if any batch effect correction needs to take place, the samples were visualized with a PCA after all genes were scaled and centered. If there was a heacy batch effect, we would see two distinct clusters along the first principal component, but the points should overlap in all of the resulting principal components. In the [PCA](https://nevada.app.box.com/file/892067726799), we can see that although the cells were collected weeks apart, there is little to no batch effect.
* To confirm that the samples do not need a batch effect correction, a tSNE [plot](https://nevada.app.box.com/file/892069283283) was created and again we see very good cell overlap there as well.

Population Analysis

* Now that we have a quality, filtered, normalized dataset with no visible batch effect, we can continue our analysis by first doing a principal components analysis. By plotting the amount of variability explained by each PC as first an [elbow plot](https://nevada.app.box.com/file/892837970390) and then a [heatmap](https://nevada.app.box.com/file/892066895626) of the top 6 PCs we can see distinct variation in the first ~4 PCs, which is significantly less pronounced as more components are included.
* To decide exactly how many PCs to include for downstream analysis, a jackStraw [plot](https://nevada.app.box.com/file/892067083088) is created.
  + The JackStraw function randomly permutes a subset of data, and calculates projected PCA scores for these ‘random’ genes, then compares the PCA scores for the ‘random’ genes with the observed PCA scores to determine statistical significance. End result is a p-value for each gene’s association with each principal component. We identify significant PCs as those who have a strong enrichment of low p-value genes.
  + NOTE: the jackstraw plot is large and needs to be downloaded to view the full image with all 100 PC p-values.
* Based on the plot output, the first 60 principal components were chosen as they captured the most variance.
* Distance between the cells were then calculated with a k-nearest neigbor approach based on the 60 principal components previously chosen.
* tSNE dimensionality reduction [plots](https://nevada.app.box.com/file/892066744236) are then used to visualize clustering results at different resolutions: 0.25, 0.75, 1.25, 1.75. As input to the tSNE, the same 60 Principal components were used.
* uMAP dimensionality reduction was plotted as well to observe cell clusters, as well as clustering by cell cycle stage [here](https://nevada.app.box.com/file/892064233228). For the most part each cell type cluster has equal distribution of cell in each stage.
* uMAP plots were also created for each of the QC metrics to see how each cell type differs in [Features](https://nevada.app.box.com/file/892069201111), [counts](https://nevada.app.box.com/file/892064314577), and [percent mito](https://nevada.app.box.com/file/892064240428).
* A phylogenetic tree relating the ‘average’ cell from each group in the best resolution (0.25 here) was [plotted](https://nevada.app.box.com/file/892887057592). Tree is estimated based on a distance matrix constructed in either gene expression space or PCA space.
  + From the tree it looks like this is in fact the best resolution and each cell type is distinct
* Now that we can see 7 distinct cell types, 0-6, we can identify marker genes to figure out what kind of cell populations we see. After marker genes were identified, a violin [plot](https://nevada.app.box.com/file/892066852760) of the top genes for each cluster is created to see how distinct how some of the markers are.
* A table of all significant biomarkers for each cluster and the corresponding statistics was output and is available [here](https://nevada.app.box.com/file/892075452921).
  + Regarding dendritic cells, since that was a focus of this pilot study, this above table would be the place to look for DC-specific genes. For example, MZB1 is a marker for plasmacytoid DCs.
  + However, these DC groups are so rare, they are difficult to distinguish from background noise for a dataset of this size without prior knowledge. So there are likely specific DC subpopulations within some of the 7 clusters identified here.
* Violin plots of some of the top genes in cluster 0 were also made [here](https://nevada.app.box.com/file/892067587086) to further visualize some more of the signficant genes from tghe above table. This can be done for all of the clusters if desired.
* After marker genes were identified for each cell type/cluster, each cluster was looked at and those significantly ernriched genes were used to identify what type of cell that population is.
  + For example, B-cells express MS4A1 while the other cell types should not, and we see that in the plots.
* After identifying the types of cells in each cluster, the Seurat object was re-labeled and a a new uMAP [plot](https://nevada.app.box.com/file/892066780256) was made. A [heatmap](https://nevada.app.box.com/file/892896256875) of the top 10 biomarkers for  each cluster (based on fold-change) was also made after the cell types were labeled.