Project 4:
A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure

Data Curator: Carol Muriithi
Programmer: Cory Williams
Analyst: Nicholas Mosca
Biologist: Zeyuan Cao
TA: Dakota

Introduction:

The pancreas is an essential organ that is responsible for helping the body maintain homeostasis. Functions of the pancreas include the secretion of multiple digestive enzymes and metabolic hormones.[2] Two major cell types, acinar and duct, make up close to 95% of the pancreas.[2] Islets make up the additional 5% of the pancreatic mass. Within the Islet mass lies exocrine tissue and channels that house endocrine cells secreting hormones responsible for glucose homeostasis.[3] Within the endocrine cells that are nested in the exocrine tissue, a vast population of other cell types is present. Some of the previously identified cell types that reside in exocrine tissue are alpha, beta, delta, gamma, and epsilon cells.[4] Dysfunction with the pancreas can lead to cancer, and Type 1 and 2 Diabetes.[4] It has been identified that this diverse population of cell types in exocrine is crucial for the overall pancreas function.[4] Therefore, comprehensive profiling of the molecular architecture of pancreatic cell types is necessary to gain a deeper understanding of their connection with disease.

Recently, single-cell RNA-sequencing has emerged as a prominent method that characterizes transcripts at a cellular level using unique barcode sequences, allowing for insights into the molecular heterogeneity of cell types within tissue. Using single-cell RNA-seq, Baron et al. classified the cell types within the human and mouse pancreas based on their transcriptome and showed clear separation of distinct cell types using a visualized projection plot. The aim of this study is to replicate the results of Baron et al. using different methods for library processing, clustering, and visualization on part of their data.

Data:

Klein et al implemented a droplet-based, single-cell RNA-seq method to determine the transcriptomes of over 12,000 individuals pancreatic cells from four human donors and two mouse strains. In our study, we only used human pancreatic islets (sample 2) associated with the 51 year old female donor for our analysis. Cells were barcoded using the inDrop platform which makes use of the CEL-Seq protocol for library construction (Hashimshony et al). The human single cell RNA-Seq dataset samples in this study were downloaded from NCBI functional genomics data repository (GEO). The data for the three runs done on samples associated with the 51 year old female donor were found in SRR files SRR3879604, SRR3879605 and SRR3879606. The accession number used was GSE84133.

Link(https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM2230758).

Methods/Results:

Count the number of reads per barcodes and Whitelist informative barcodes:

Processed raw read 1 barcode files containing the barcode for each cell and UMIs for each transcript. The reads in these files all had a format that had 19 barcoded bases(bc) with corresponding 6 Unique molecular identifiers (UMIs). The UMIs are normally added to the transcripts during reverse-transcription. Their purpose in the files is to enable sequencing reads to be assigned to individual transcript molecules and the removal of amplification noise and biases from scRNAseq data. Respective distinct barcodes in these files represented individual cells while respective distinct UMIs were representative of molecules. Analysis to find the number of reads per distinct barcodes was done using bash and python scripts. This was done by counting the frequency of each barcode in each file. Upon reading our files, most of the reads were in the format shown below.

	bc	**counts**
0	TAACTACTACTGATTTGGGA	30,5097
1	GATTGAGGGTAATCAATCG	7223
2	TGAAACACACGTGCTTCAT	3701
3	ACGGACAACACCCGACTTT	363354
4	TAGTCTCTACTTGTTATCA	224380

Table 1: The table shows the format of some of the barcodes from our data and the associated counts.

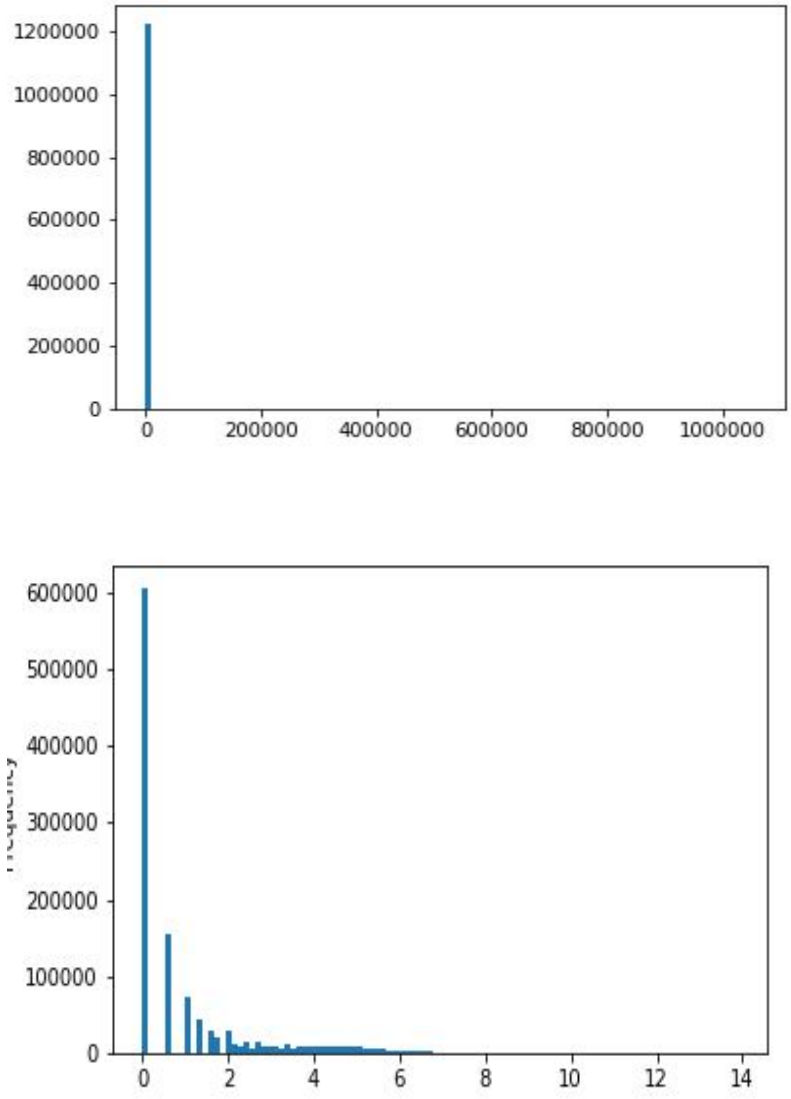


Figure 1: Histograms showing the frequency plotted against the log distribution of the counts put in a 100 bins. The figure shows that most of the data in the files came from single counts reads. Part 1 of this figure shows the frequency of the reads per barcode. The majority of the raw data had single counts. Part 2 shows the distribution of reads of each barcode plotted against the frequency and also shows that a majority of our data was from single counts. On average, 98,715 reads were associated with each analyzed cell

The figure shows that a majority of the counts associated with each barcode were approximately ~98, 715. A majority of our raw data had single counts. The higher counts appeared less as seen by the distribution in the bottom plot of figure 1. After counting the number of reads per, we encountered an average ~98,715 counts per associated barcode with each analyzed cell. This number was close to the average ~100,000 reads associated with each analyzed cell reported by Baron et al.

Whitelist informative barcodes:

To generate white list informative barcodes, reads with infrequent barcodes were eliminated from consideration. A read threshold to filter out codes that were too infrequent to be informative was then chosen. We then wrote the remaining barcodes to one barcode per line in a new file as whitelist barcodes. The whitelist barcodes were then provided to salmon.

The threshold used to generate whitelist informative barcodes:

Data was read from the count output files and stored in a python data frame. The columns were renamed for easy indexing. The index was reset to make it linear and values sorted by counts. The values for counts are shown in table 1. A sum of these counts was then done and values stored in an output file labeled as 'total counts'. A cumulative sum of the sorted counts was also performed using a python script. To find the percentile counts, the cumulative sum of the counts was divided by the total counts. A filter was performed by picking percentile counts with values greater than 0.05. This would make sure that we were only filtering for the top 95% of barcodes (in terms of cumulative count percentile). Filtering for the top 95% of the data was essential so as to keep a majority of the data while filtering out the noise generated by PCR amplification where some transcripts become over represented in the final library compared to their true abundance. A second filter was done using z score on the output from the first filtering step. This gave an idea of how far from the mean a data point was and also gave us a measure of how many standard deviations below or above the population mean a the raw score was. Subsequently, a log transformation of the z score was done on the output from this second filtering step. We then picked for counts between -3 to 3 standard deviations in the third filtering step and histograms(figures 2,3 and 4) from these steps generated.

Filter 1

1) (filter_1 = bc_df[bc_df.counts_pctl > 0.05).

Filter 2

2) filter_1['log_z'] = z_score(np.log(filter_1.counts))

Filter 3: whitelist generating step

3) whitelist_df = filter_1[filter_1['log_z'].between(-3,3)]

4) Finding the z score: z_score(s): return (s - s.mean())/s.std()

Determining the means of the counts(takes the counts and find the means): bc_df.counts.mean()

The means of the counts was determined to be 258.3270898796563

The standard deviation of the counts was determined to be 6106.199073249066

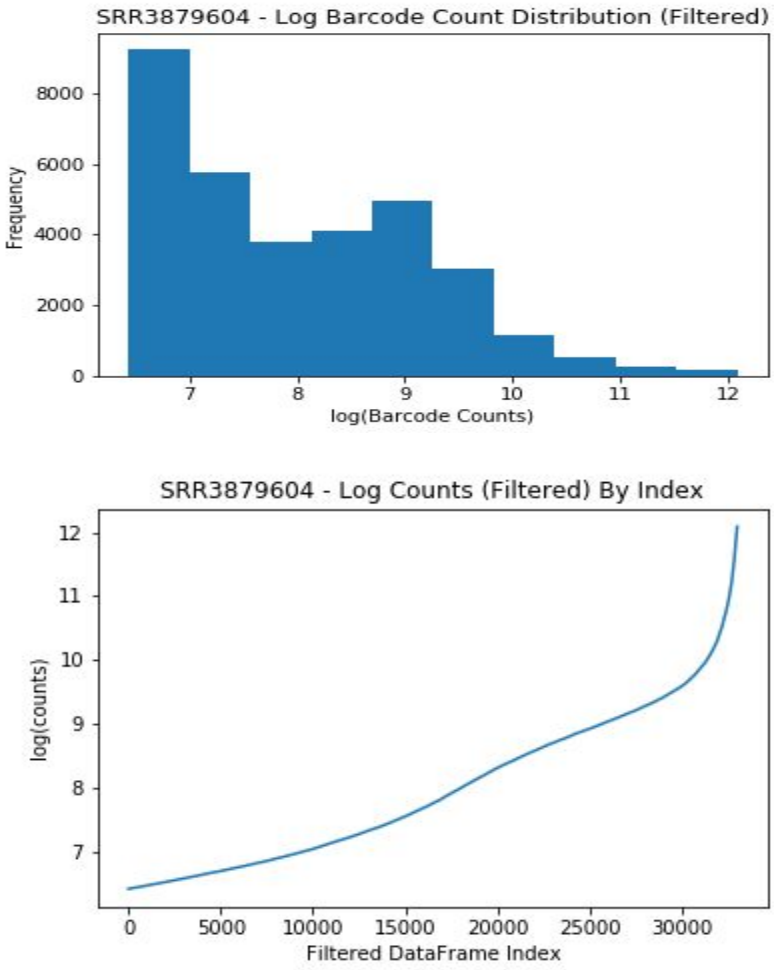
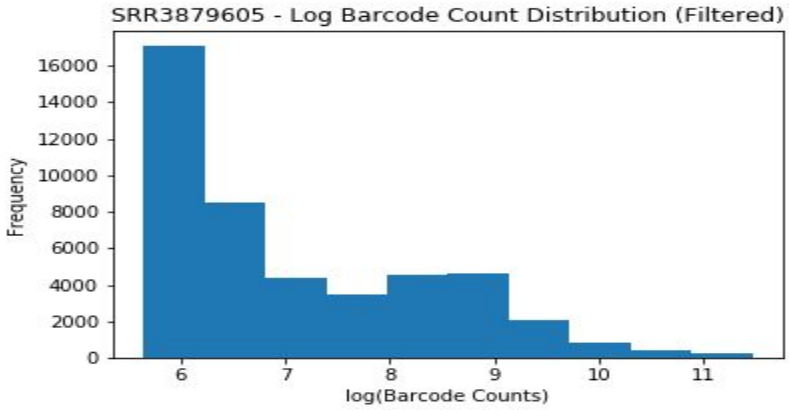


Figure 2: A histogram of Log Barcode Count Distribution (Filtered). The histogram shows the frequency of each barcode in file SRR3879604 plotted against the filtered counts from the whitelist informative barcodes. After filtering and counting the number of reads per distinct barcodes, we encountered an average ~4000 uniquely detected distinct barcodes. The second plot shows the Log Counts (Filtered) By Index.



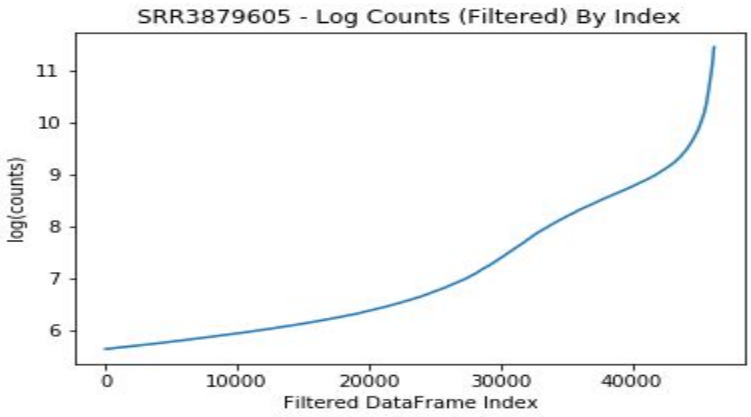


Figure 3: A histogram of Log Barcode Count Distribution (Filtered). The histogram showsThe frequency of each barcode in file SRR3879605 plotted against the filtered counts from the whitelist informative barcodes. After filtering and counting the number of reads per distinct barcodes, we encountered an average ~5943 uniquely detected distinct barcodes. The second plot shows the Log Counts (Filtered) By Index.

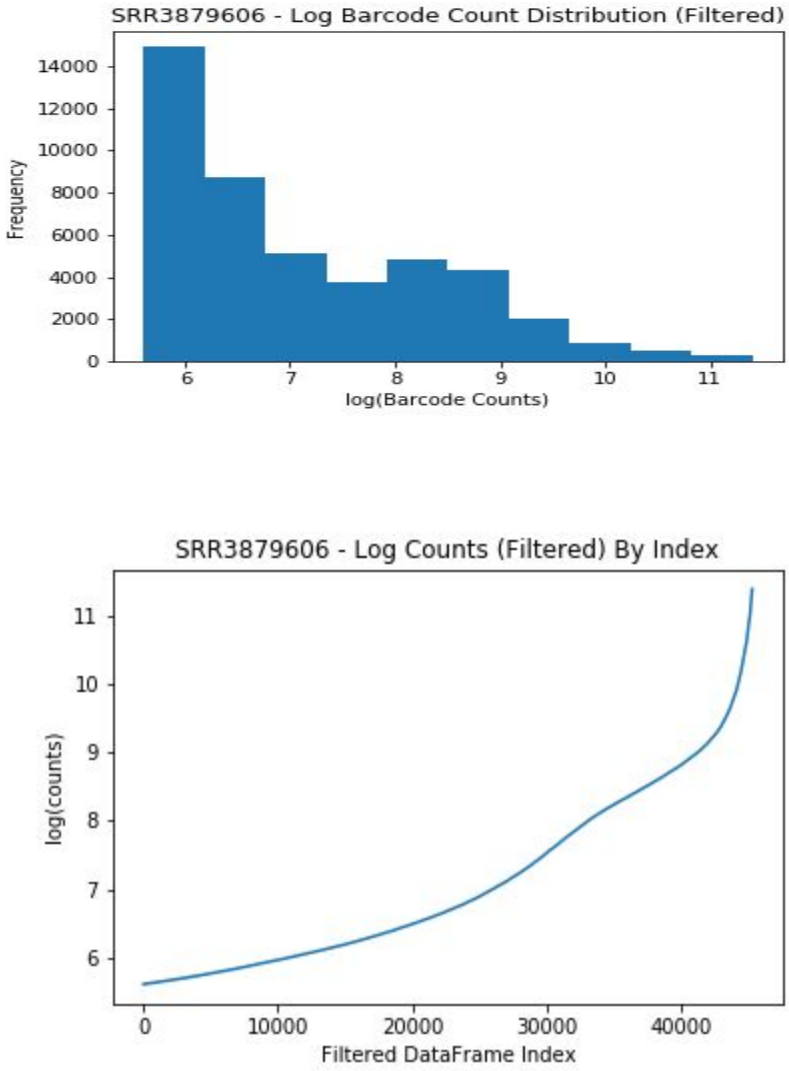


Figure 4: A histogram of Log Barcode Count Distribution (Filtered). The histogram shows the frequency of each barcode in file SRR3879606 plotted against the filtered counts from the whitelist informative barcodes. After counting the number of reads per distinct barcodes, we encountered an average counts ~5531uniquely detected distinct barcodes. The second plot shows the Log Counts (Filtered) By Index.

The histograms in figures 2, 3 and 4 show histograms showing the frequency of each barcode in files SRR3879604, SRR3879605 and SRR3879606 plotted against the filtered counts from the whitelist informative barcodes. After counting the number of reads per distinct barcodes, we encountered an average ~5000- ~5900 distinctly detected barcodes. This number was close to the average ~6000 uniquely detected transcripts reported by Baron et al.

Generating a UMI counts matrix:

A UMI counts matrix was generated using salmon alevin with the provided fastq files and the whitelisted barcodes. The human reference transcriptome was obtained from the Gencode website. A transcript ID ((ENSTXXX) was created to (ENSGXXX) and the file mapped as instructed in the salmon alevin documentation. This was done so as to allow salmon to collapse from the transcript to the gene level.

<u>Record Mapping statistics</u>

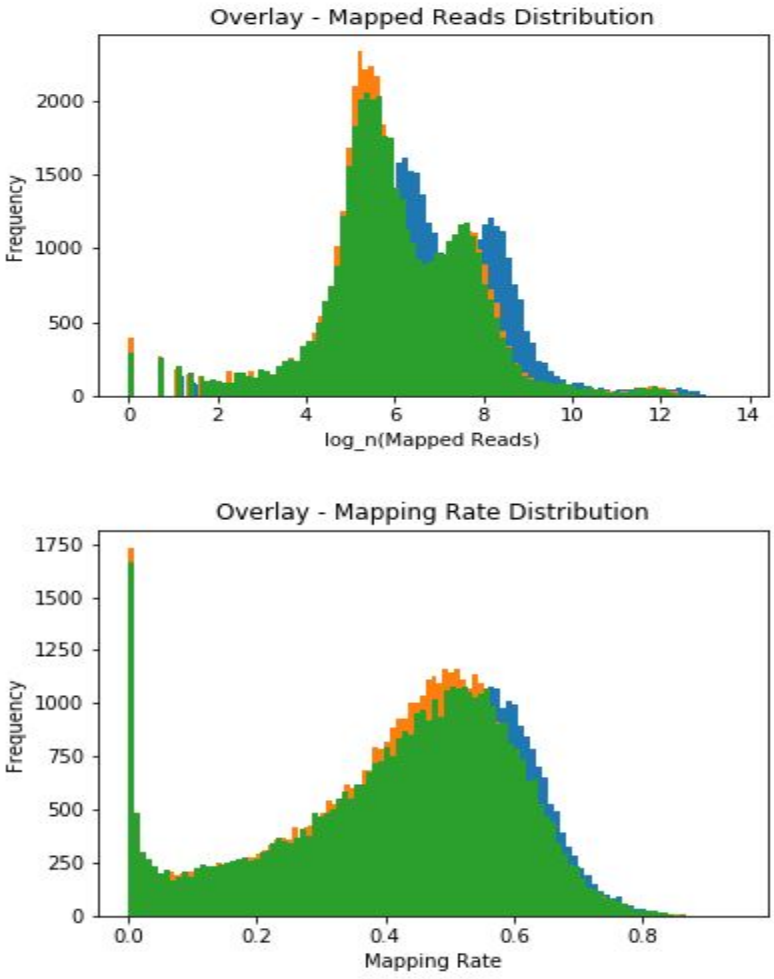


Figure 5: An overlay of the mapped reads distribution and mapping rate distribution. The mapping reads distribution and mapping rates from the salmon alevin output all three SRR(SRR3879604, SRR3879605 and SRR3879606) files were almost similar.

The overlay of mapping reads distribution and mapping rate distribution from the SRR3879604, SRR3879605 and SRR3879606 showed an overlap from all three files.

This would be as expected because all the samples were obtained from the same 51 year old female donor.

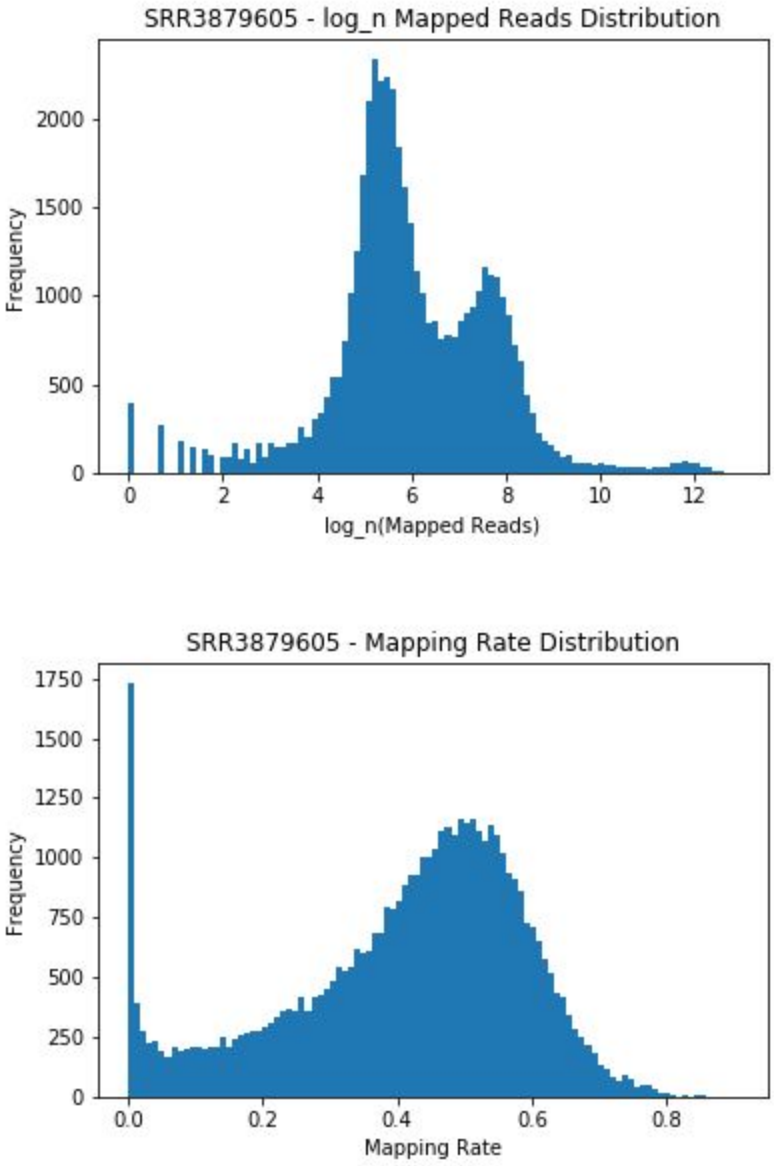


Figure 6: **The mapped reads distribution and mapping rate distribution.** The frequency was plotted against mapping rate and the log of the mapped reads so as to visualize the results for SRR3879604. The mapping rate distribution was shown to have a frequency of ~1250.

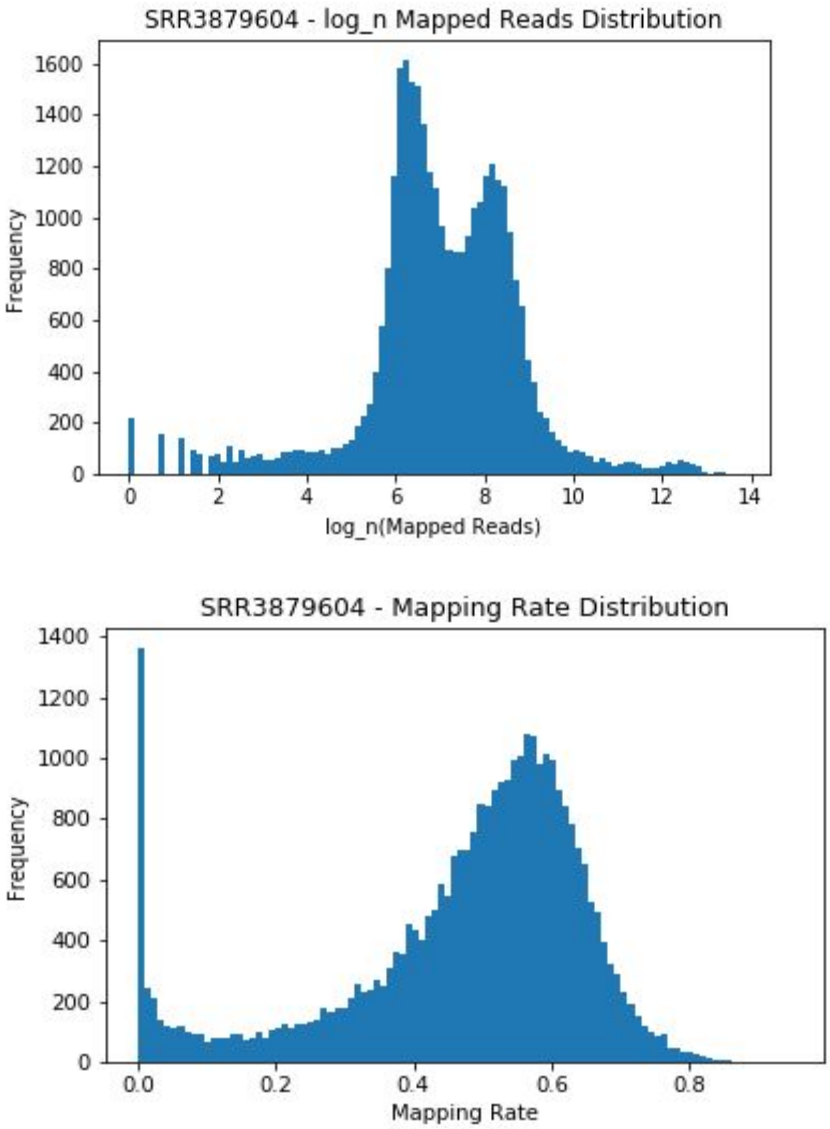


Figure 7: The mapped reads distribution and mapping rate distribution. The frequency was plotted against mapping rate and the log of the mapped reads so as to visualize the results for SRR3879605. The mapping rate distribution for this file had a frequency of ~1100

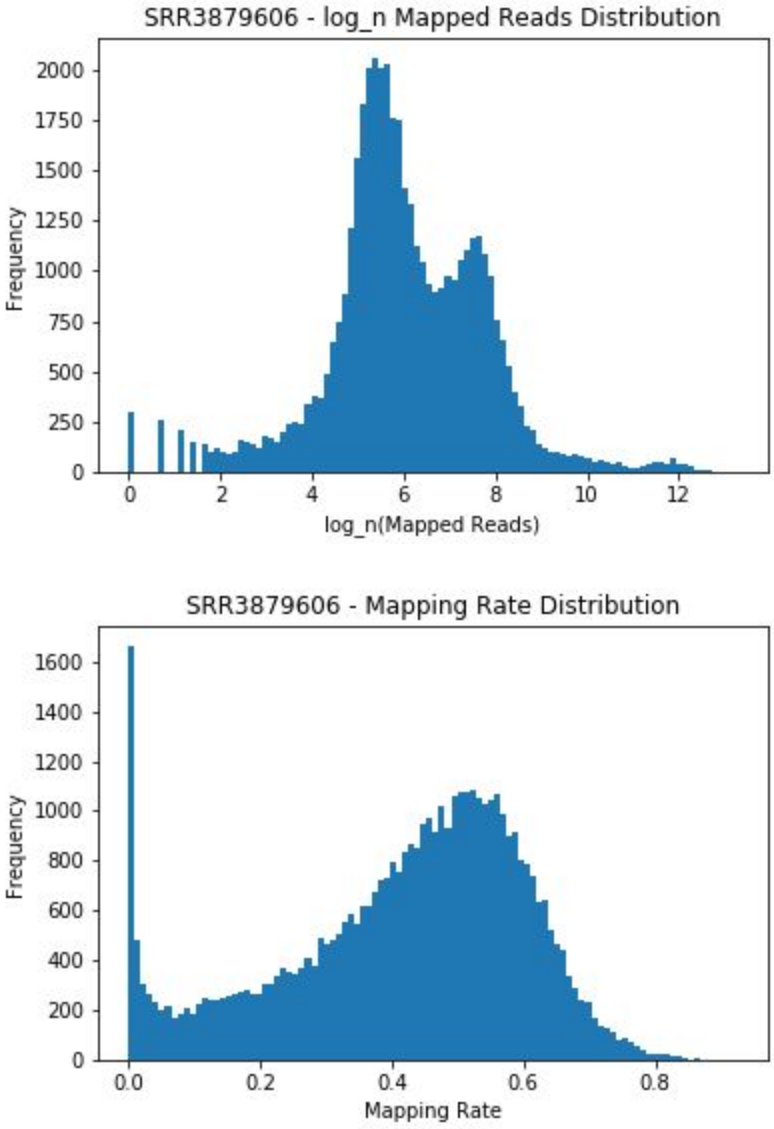


Figure 8: The mapped reads distribution and mapping rate distribution. The frequency was plotted against mapping rate and the log of the mapped reads so as to visualize the results for SRR3879606. The mapping rate distribution for this file had a frequency of ~1100

Figures 6, 7 and 8 show plots of the mapped reads distribution and mapping rate. The frequency was plotted against mapping rate and the log of the mapped reads so as to visualize the results for individual files. The individual plots(figures 6, 7 and 8) of SRR3879604, SRR3879605 and SRR3879606 confirmed the overalp from the results from the overlay in figure 5. This is because all the samples were from the same 51 year old female donor.

Precomputed Alevin data was obtained from the BF528 project folder. To import and convert the Alevin output to be pipelined into R, the package tximport version

1.12.7 was utilized. The Alevin output file used contained gene expression levels, gene identification code, and cell sample barcodes for around 30,000 individual cells. For filtering and further downstream analysis, the package Seurat version 2.7 was used. Tximport converts Alevin output files into a matrix that the Seurat package can turn into an object to apply multiple analysis tools easily.

The raw alevin file contained varying gene counts per cell with the highest gene count for a single cell being in the upper 7000's (Fig. 9) A gene count that high is problematic and usually points into the compilations of doublets. Per the project guidelines provided cross-referenced by information in the supplementary article, the range of genes per cell should be 4000 or lower. With those complications and parameters in mind, furthering filtering had to be conducted before normalization.

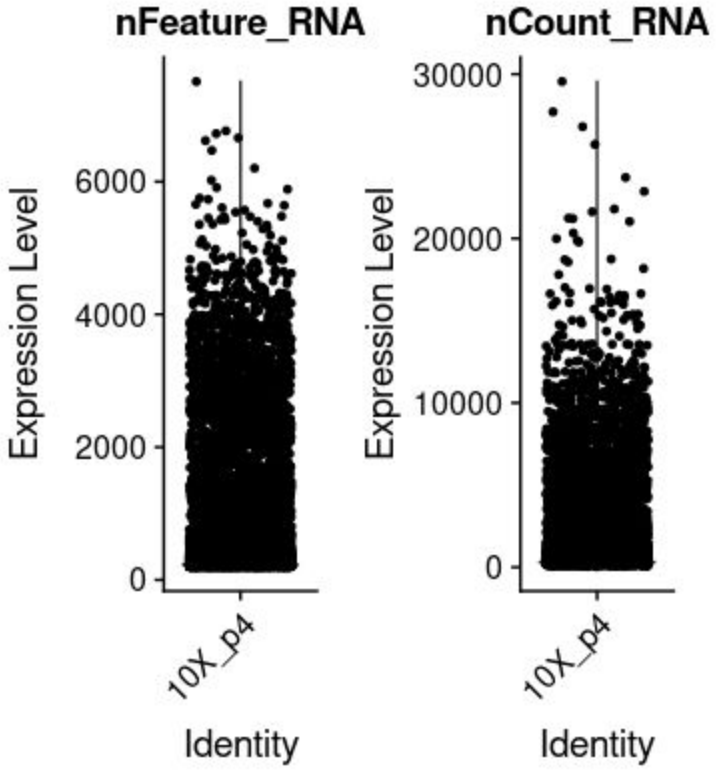


Figure 9. Plot of Alevin output cell and gene count before filtering. On the right (nCount_RNA) represents the total cell count hitting 30,000 displayed in an expression level format. Upon first glance nCount_RNA implies that cells have different "expression levels" however just for nCount_RNA plot this actually represents the total number of cells recorded on the y axis with the highest point at 30,000 representing that is the max. On the left plot (nFeature_RNA is gene expression) represents a true expression level plot with the y axis scale representing gene expression per cell with only a few cells having over 6000 genes recorded and a majority of the cells having 4000 genes or less recorded.

Filtering:

The first filtering that was conducted before normalization was removing cells that have a gene count (feature count) of more than 2700. This filtering step was done with a command from the Seurat package with parameters set for keeping all cells with gene counts above 200 and below 2700 (Fig. 10).

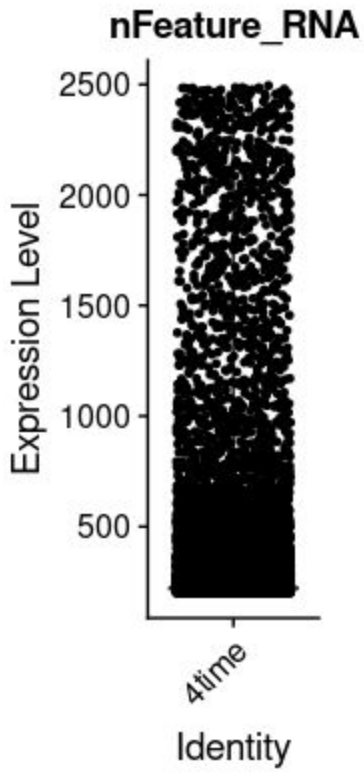


Figure 10. Post filtering gene expression plot. This plot represents cells retained after feature count filtering of less than 2700 but more than 200. This is pre normalization so the cells that appear to have expressions levels of less than 200 should be adjusted for automatically post normalization.

Normalization:

After this filtering step, normalization was done using the normalize function of the Seurat package with the specific normalization method being “LogNormalize with a scale factor of 10,000.

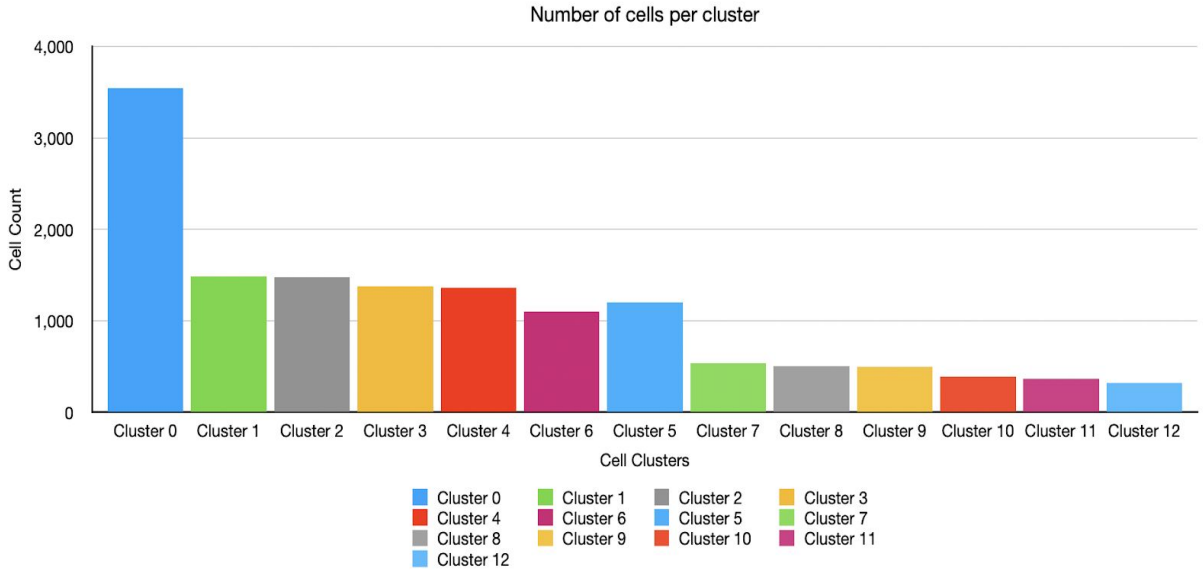
Post Normalization Filtering:

Post normalization filtering was done by identifying genes with high cell variability (variability filtering). This filtering step looks across all normalized cells and their genes and removes the peripheral cells that have abnormally low and or high expression of specific genes. This ensures that the cells kept for further downstream analysis do not have a large group of cells that are uncommonly expressing too high or too low levels of a particular gene. Variability filtering was followed by linear transformation. Linear

transformation was used to shift the expression of each gene so that the mean expression across all cells is 0 in addition to scaling the expression of each gene so that the variance across cells is 1. This step gives equal weight in downstream analyses so that highly expressed genes do not dominate.

Initial filtering, normalization, and post normalization filtering dramatically dropped the cell count from around 30,000 to around 14,000 cells of higher quality. Clustering was done using the Seurat clustering function and produced a total of 13 clusters. Clusters had an average of 1000 cells per cluster (Fig. 11).

A.



B.

Number of cells per cluster

Cluster legend	Cell count
Cluster 0	3,542
Cluster 1	1,484
Cluster 2	1,480
Cluster 3	1,374
Cluster 4	1,362
Cluster 5	1,204
Cluster 6	1,105
Cluster 7	537
Cluster 8	502
Cluster 9	497
Cluster 10	392
Cluster 11	371
Cluster 12	320

Figure 11. Post filtering and normalization clustering. (A) Color assigned bar chart of the 13 produced clustering after filter and normalization. Cluster 0 appears to have the most cells while cluster 12 having an alarming low count. **(B)** Table containing the raw cell count number records per cluster, can also be used as a legend for above chart.

Differential expression:

Differential expression was conducted using the Wilcoxon Rank Sum test on the clustered Seurat dataset. Each cluster produced a set of associated marker genes based on the top genes up and downregulated. Top up and down regulated genes or marker genes were used for classification. Log2 fold change was the parameter used to separate marker genes per cluster.

Cell Type Classification:

Top differentially expressed genes were filtered using Log2 fold change for each individual cluster. Marker genes were linked to cell types using the Panglao Cell type identification Database. Panglao matches single cell expression data with associated cell types displayed in figure 12. The top three genes for each cluster were passed through Panglao and classified with the closest associated cell type. Baron et al was also used for additional classification. Marker genes associated with cell type are shown in figure 13. Additional marker genes were detected and shown in table 2**.**

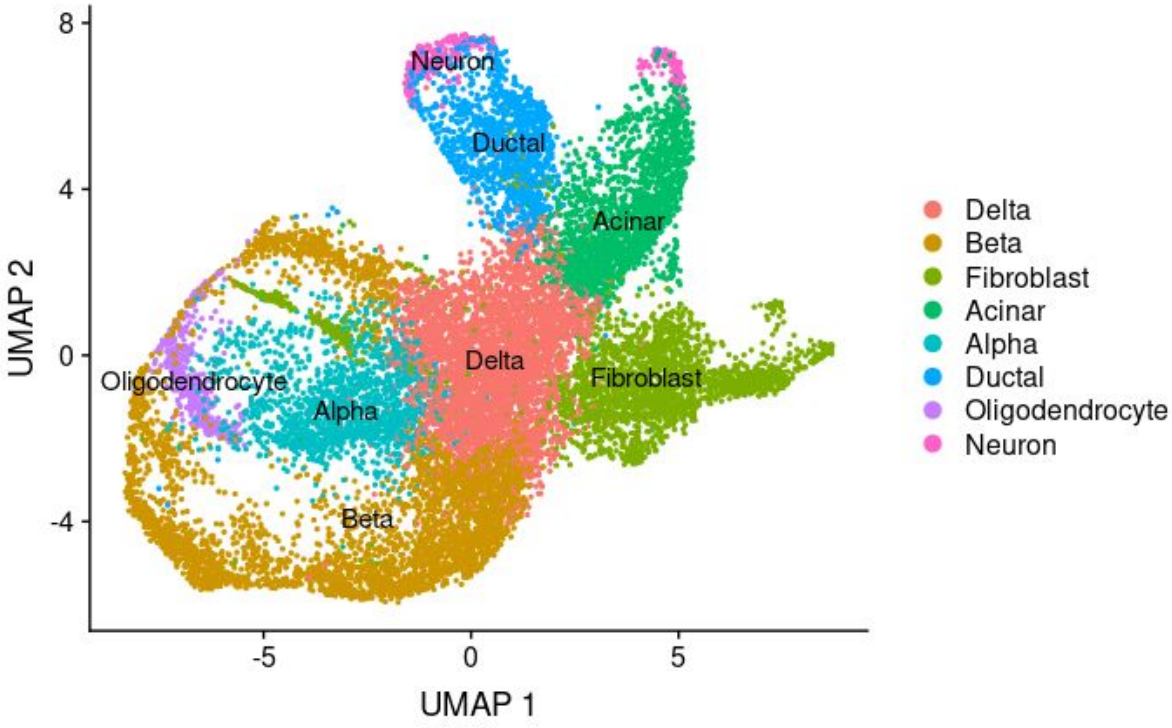


Figure 12 : Identified Cell types based on Marker genes
Ten different cell types were identified based on marker genes. Top genes were attached to cell types using the Panglao Cell type identification Database. Each cell type is indicated by a different color.

Marker genes were distinguished based on highest absolute value for Log2 fold change. Three marker genes for each cluster were taken into consideration when determining cell type.

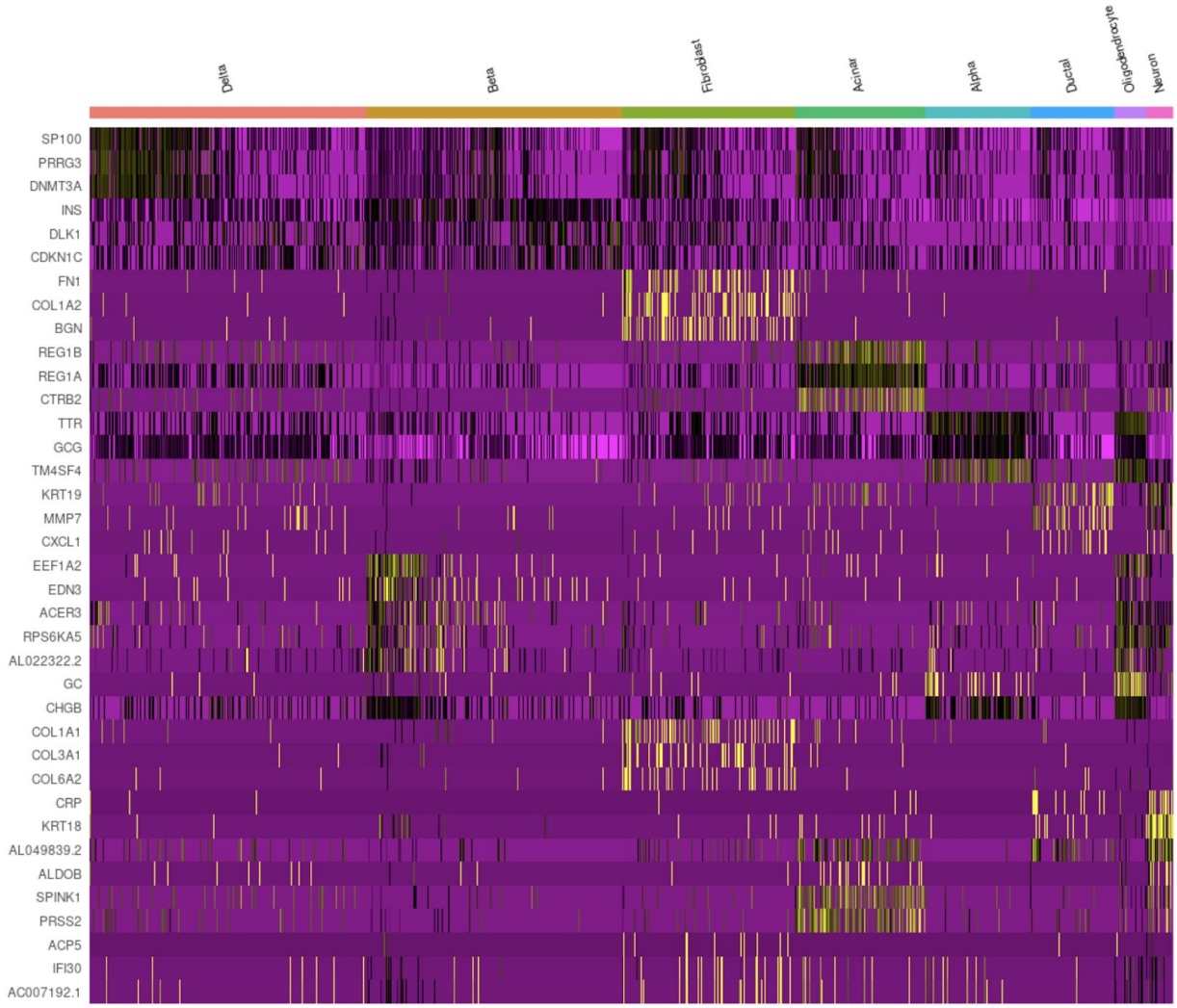


Figure 13 : Top Marker Genes identified for each cell cluster
Figure X shows the expression of each marker gene relative to the identified cell type clusters. Each clustered cell type is based on log normalized UMI counts of associated genes .

Novel Marker genes	avg_logFC
SP100	0.96223863
INS	1.7656454
COL1A1	2.52259049
REG1B	2.47236796
TTR	1.64652921
CXCL1	2.11527736
ACER3	1.80392646
TTR	1.99688735
AL049839.2	2.23971288
ACP5	3.91226455

Table 3: Novel marker genes
Ten highly differentially expressed marker genes that could be used to further classify custers. Avg_logFC is linked to the biological significance or expression of these marker genes.

Gene Set Enrichment Analysis:

Gene set enrichment analysis(GSEA) was performed on the results of differential expression using metascape. The differential expression gene list was filtered with a criteria of adjusted p-value less than 0.01. The genes belonging to each cluster after filtering were extracted and sent to metascape for enrichment analysis. Enrichment analysis was performed on each of the following databases: KEGG Pathway, GO Biological Processes, Reactome Gene Sets, Canonical Pathways and the Comprehensive Resource of Mammalian Protein Complexes(CORUM). The results of

GSEA are shown in table 3. We found regulation of hormone level and regulated exocytosis were enriched in several cell types. There are significantly large numbers of genes in both neuron and oligodendrocyte cell types, which suggests some genes in the differential expression gene list may be mislabeled. Also, oligodendrocytes should not exist in pancreas as they are only found in the central nervous system. This could be the result of mislabelling schwann cells as oligodendrocytes.

Cluster Label	Number of Genes	Enriched Terms
Acinar	328	Eukaryotic Translation Elongation*
		regulation of hormone level
		regulated exocytosis
Alpha	209	regulated exocytosis
		Post-translational protein phosphorylation*
		response to wounding
Beta	459	regulation of hormone levels
		regulated exocytosis
		Nervous system development*
Delta	268	Ribosome, cytoplasmic#
		TRBP containing complex#
		ribosomal large subunit biogenesis
Ductal	356	regulation of hormone levels
		response to wounding
		myeloid leukocyte activation
Fibroblast	191	extracellular matrix organization
		Regulation of Insulin-like Growth Factor (IGF) transport
		regulation of hormone levels
Neuron	1586	Eukaryotic Translation Elongation*
		myeloid leukocyte activation
		apoptotic signaling pathway
Oligodendrocyte?	1473	SRP-dependent cotranslational protein targeting to membrane
		purine ribonucleoside monophosphate metabolic process
		mitochondrion organization

Table 4: results of GSEA
Top 3 GSEA results for each cluster label are displayed, ranked in decreasing order of significance. All unmarked terms are from GO Biological Process. *: term in Reactome Gene Sets. #: term in CORUM. ?: possible mislabel of schwann cells.

Discussion:

After reading our raw data, we were able to identify that the majority of the raw data had single counts and the higher counts appeared less as seen by the distribution in the bottom plot of figure 1. After counting the number of reads per, we encountered an average ~98,715 counts per associated barcode with each analyzed cell. This number was close to the average ~100,000 reads associated with each analyzed cell reported by Baron et al. After counting the number of reads per distinct barcodes, we encountered an average ~5000- ~5900 distinctly detected barcodes. This number was close to the average ~6000 uniquely detected transcripts reported by Baron et al. The salmon alevin output from both the individual and combined sample files (SRR3879604, SRR3879605 and SRR3879606) showed an average mapping rate of 0.972939(97%). The average number of deduplicated reads was estimated to be ~0.45. The dedup rate ranged from 0.004 to 0.01.

Identified cell types do align with the results of Baron et al although marker genes may have differed due to the use of a different precomputed dataset. The initial clustered results produced 13 different clusters. After classification only ten different cell types were confirmed displayed in figure 12. Many marker genes overlapped in clusters even after further filtering . Upon further review these cell types are all closely related and reside in the same exocrine tissue in the pancreas. Cell type classification is a critical part of the analysis conducted by Baron et al due to the vast unknown populations inside the pancreas. Cell types classified in figure 12 are targets for chronic diseases such as cancer and Type 1 and 2 Diabetes.[1]

The cell type classification was informative but may be not as accurate because of the overlap in identified marker genes. The projection plot shown in figure 12 does look similar to the results of Figure1D in Baron et al. It does appear that Baron et al had a larger degree of separation between clusters which may be a result of filtering differences.

Log normalized expression counts for the top genes in each cluster is shown below in figure 4. These results do differ from the results in figure 1B shown in Baron et al. The top differentially expressed genes do vary based on cell type as shown in figure 12 but much more overlap is apparent. Novel marker genes in table 2 show high expression levels but were not classified as cell types in exocrine tissue.

There were two main filtering methods of choice used for pre analyses/quality control of the alevin output data. The first two main filtering methods accounted for removing cell doublets and lowering gene expression. The threshold selected for doublet removal was chosen with the knowledge that cells expressing lower than 200 genes are usually projected to be insignificant to analysis along with cells expressing

over 2700 to be abnormally high given the cell type described in the article. The lowering gene expression variation filtering was broken into two sub filtering methods where the first removed outlier cells with abnormally low or abnormally high gene expression relative to a majority of the cells. The second sub filtering, linear transformation, shifts the expression of each gene so that the mean expression across cells is 0 and to scale the expression of each gene so that the variance across cells is 1. The LogNormalization is a standard normalization method that is applied to many other single-cell output applications. When applied to this dataset, it ran relatively quickly and as expected. This compound filtering combined with normalization resulted in producing 13 cluster groups that not only made sense but seemed to be reasonably similar to the number of clustered groups that were present in the precomputed data for the analyst role.

Overall, the results of GSEA do correspond well to each cell type as classified. For example, ribosome-related terms are enriched in delta cells, corroborating with their function of somatostatin secretion. Insulin secretion pathway is enriched in beta cells, although not shown in table 3. Due to large overlap between genes in each cluster, the enrichment results also showed overlap of terms, increasing the difficulty of distinguishing between different cell types. Both regulation of hormone level and regulated exocytosis are enriched in multiple cell types, given that several cell types in pancreas secrete hormones such as insulin, glucagon and somatostatin. Both neuron and oligodendrocyte cell types have exceptionally large numbers of genes assigned to them, and these genes are enriched in terms that do not suggest neuronal function, which indicates that some genes may be misassigned into these cell types. We failed to detect gamma cells and two types of stellate cells, possibly due to their small amount or misassignment.

Conclusion:

On average, we had an average of 98, 715 reads associated with each analyzed cell. After counting the number of reads per distinct barcodes, we encountered an average ~5000- ~5900 distinctly detected barcodes. The mapping rate from the salmon output was 97% and we were also able to obtain a low DedupRate that ranged from 0.004 to 0.01. For the most part, we were able to replicate this part of the study using a subset of the dataset from the 51 years old female donor.

Some of the challenges encountered included not having the data curator role being clearly defined. I made a lot of mistakes but I also was able to learn a lot from them. Working on the SCC cluster had a lot of time lags that can be attributed to the high number of users at this time of the year. There were also no clear directions on what files to use for some steps given that each SRR file had 3 different files in it.

Package versions were the most challenging aspect of this experiment, with the provided version being severely out of date, leading to multiple installations and updating of the required packages. Computing power also played a significant factor when importing the Alevin output file into R. This took over an hour when ran on the Boston University clustering computing network with four cores selected; however, when the cores were increased to 16 that time was cut in half. Future directions would lead to making sure a sufficient amount of cores is selected from the start to dramatically reduce load times.

It has been shown that classification of thousands of transcriptomes from pancreatic cells can be classified into known cell type identities. Additional novel marker genes were identified as well. Working with the sample data for this project did make classification of cell types a little difficult. I came across some overlap in my marker genes . Some clusters have the same marker genes so i classified based on the top three genes for each cluster. I would have liked to see the difference if the filtered gene set was provided. I would guess that this redundancy in genes would have been different. Overall we did identify cell types that made sense with Baron et al, and a pattern of expressed genes is clear in figure 13. The flow of this report may also also suffer due to the communication disconnect between roles.

We believe that we have for the most part successfully replicated the result of Baron et al., although using pre-computed data made drawing conclusions a little bit difficult. Some difficulties in the GSEA analysis include the overlap of enrichment terms between cell types, and also some enrichment terms are not correlated with functions of the cell type they are assigned to. These problems are solved after consulting a number of articles doing the same analysis to ascertain the cell types. In particular, an article published by Muraro et al. proved to be very helpful, since they also performed single-cell RNA-seq on pancreas.[7] Therefore, I was able to draw conclusions after comparing both Baron’s and Muraro’s with mine.

References:

1. Baron M, Veres A, Wolock SL, et al. A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure. *Cell Syst*. 2016;3(4):346–360.e4. doi:10.1016/j.cels.2016.08.011
2. Kimmel RA, Meyer D. Molecular regulation of pancreas development in zebrafish. Methods Cell Biol. 2010; 100:261–280. [PubMed: 21111221]
3. Drucker DJ. The role of gut hormones in glucose homeostasis. J Clin Invest. 2007; 117:24–32. [PubMed: 17200703]
4. Mastracci TL, Sussel L. The endocrine pancreas: insights into development, differentiation, and diabetes. Wiley Interdiscip Rev Dev Biol. 2012; 1:609–628. [PubMed: 23799564]
5. Klein AM, Mazutis L, Akartuna I, Tallapragada N, Veres A, Li V, Peshkin L, Weitz DA, Kirschner MW. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. Cell. 2015 May 21; 161(5):1187-1201.
6. Hashimshony T, Wagner F, Sher N, Yanai I. CEL-Seq: single-cell RNA-Seq by multiplexed linear amplification. Cell Rep. 2012 Sep 27; 2(3):666-73
7. Muraro MJ, Dharmadhikari G, Grün D, Groen N, Dielen T, Jansen E, van Gurp L, Engelse MA, Carlotti F, de Koning EJ, van Oudenaarden A. A Single-Cell Transcriptome Atlas of the Human Pancreas. Cell Syst. 2016 Oct 26;3(4):385-394.