Computational Health Laboratory Report

Ninniri Matteo (student ID: 543873), Piras Andrea (student ID: 619640)

May 2023

1 Introduction

The project consisted in performing a series of single-cell RNA sequencing analyses on several datasets containing the data of a multitude of patients affected, to different degrees, by Chron's Disease, as well as sane controls. In particular, we were asked to:

- Clusterize the cells in the dataset and identify the cells clusterized in each cluster.
- Perform gene enrichment analysis on the clusters obtained, using a list of genes given to us by the professors.

As an extra, we have clusterized the enriched genes and found which ones did constantly behave like specific cell markers, as well as which of such genes did behave this way only under certain disease statuses. We have performed pathway enrichment analysis of such genes as well.

We have also performed experiments on patients who donated, for the same dataset, inflamed and non-inflamed portions of their intestines, to determine which genes did behave constantly like certain cell markers, regardless of the specific sample's condition.

The project had to be developed using the "development" branch of the COTAN library [1] (in particular, we used version 2.1.1). All the code and the results are publicly available at `https://github.com/aprs3/CHL_Project.git`. We have also suggested some performance improvements to the authors of the library and some of them have already been implemented.

Some of the images and tables we intended to show in this document were too large to allow for the entire content of this report to fit in 10 pages as requested by the professors. Since they were not crucial for the comprehension of this document, we have prepared a second document called report_supplementary.pdf. All the plots and tables not featured in either have been inserted in the GitHub repository.

1.1 Code

This is the list of R scripts. The usage is found commented at the top of every script:

- main.R: Performs the data cleaning, clustering, and gene set enrichment analysis using COTAN. Outputs multiple cluster cuts from a given range as .csv files. It can be run by either setting the arguments by command line or by setting the *args* array manually such that the first and second element of the list are respectively the dataset folder name and the patient's ID. The script will create all the folders automatically. If one of the args is set to -1, the script will procedurally ask the user to insert variable values in an interactive way. Also, this script will calculate the clusterization for the targeted enrichment genes for all the number of clusters ranging between two variables, *start* and *end*. If one of the *args* is set to -1, the script will procedurally ask the user to insert the variable values interactively.
- venn.R: Given a list of patients, it calculates the optimal number of enriched genes clusters for each patient (within a range set by the variables *start* and *end*) such that the clusters where a cell type's markers are placed variates as least as possible. Finally, it saves into a .txt file the optimal number of clusters for each patient and the .csv file with the intersections found with the method described in section 5.1.1. To execute it, set the variable *dataset* with the selected dataset name (for example TI_IMM), the list *to_load* with the patients' IDs, and the variables *start* and *end* that specify the range where to search the optimal number of clusters.
- find_state_markers.R: This script loads the .csv files containing, for each patient group, the intersection for each cell type of the optimal clusters (as calculated by venn.R) containing the genes which behaved like that specific cell's markers, and removes from each patient group the union of the genes which appeared in the other

two intersections. The resulting files contain the genes which behave, for each cell type, like that specific cell's markers only when the patient presents a specific condition (inflamed, not inflamed, or healthy).

- separate_state_markers.R: This script loads the .csv files generated by *find_state_markers.R* and removes (or better, puts into a separate column) the genes which were already present in their respective *known_cells_genes.csv* rows, meaning that what is left are genes which do behave just like the cells' markers only under a specific patient condition (healthy, inflamed or not inflamed) and that were not already known.
- utils.R: Various utilities for reading files, as well as displaying the various plots.

The files specific to the pathway enrichment analysis experiments were:

- analyze_single_patients.R: given the path to a folder containing a set of .csv files, each one listing the clusters of enriched genes calculated for a specific patient as obtained by venn.R, it proceeds to extract, for each cell listed in the known_cells_genes.csv file, the pathways enriched by the genes which do behave like the markers of such cell.
- analyze_intersections.R: like analyze_single_patients.R, but it calculates the pathway enrichment analysis on files that do already list the genes behaving like a certain cluster, for each cell type.
- venn.R: a customized version of venn.R used which skips the search of the optimal clusterization for each patient (thus assuming that it had already been found).

List of helper files:

- known_cells_genes.csv: a copy of this file is present in each dataset. They contain the set of cells found in the datasets by Kong et al. [2], as well as their gene markers.
- enrichment_list.txt: the list of genes we were requested to perform gene enrichment analysis on.

2 Datasets

The data that we had available was from biopsies of the intestine of 71 donors (25 healthy controls and 46 donors with the disease) obtained by multiple experiments.

There were a total of six datasets to work with. Three datasets consisted of samples extracted from the Terminal Ileum (TI from here onward) section of the intestine, while the remaining three consisted of samples obtained from the Colon (CO from here onward). For each group, the three datasets contained, respectively, data from Stromal Cells (STR), Immune Cells (IMM), and Epithelial Cells (EPI).

The samples in each dataset were divided into three categories: healthy controls, inflamed and not-inflamed. It is important to note that we have non-inflamed samples because a person with Chron's Disease can also present portions of the intestine that are not inflamed. The patient associated with each sample was identifiable through the UMI's barcode initials. If the sample came from a healthy donor, the UMI would start with an "H" letter. "I" and "N" were respectively for the inflamed and non-inflamed ones.

3 Procedure

3.1 Preliminaries

After a preliminary discussion with the professors, it has been determined that the best approach to the project was to analyze samples for each possible condition separately to make comparisons. For each dataset, using the script main.R, we performed data cleaning, clustering, and gene enrichment analysis of every single patient considered. First, we selected samples from patients who all came from the same experiment. The reason is that different experiments with different single-cell RNA sequencing may have different statistics between the patients and could have complicated the analysis. Another problem was that some patients were significantly underrepresented. For example, table 1 is a partial list of the cell counts for the donors present in the dataset of the Stromal Cells in the Terminal Ileum (TI_STR).

As we can see, some donors counted less than 100 cells, while two patients represented over 60 percent of the 75695 samples that composed the dataset.

After further discussions, it was determined that the best approach would be to limit ourselves, for each dataset, to analyze up to four donors for each condition (healthy, inflamed, and not inflamed) among the ones which counted

Table 1: The five least represented and the five most represented patients in the TI STR dataset

H158160	H157844	N115208	I127643	H110216	N109389	H158108	N166301	I130064	N130064
3	11	42	75	78	3414	3906	6561	20434	24914

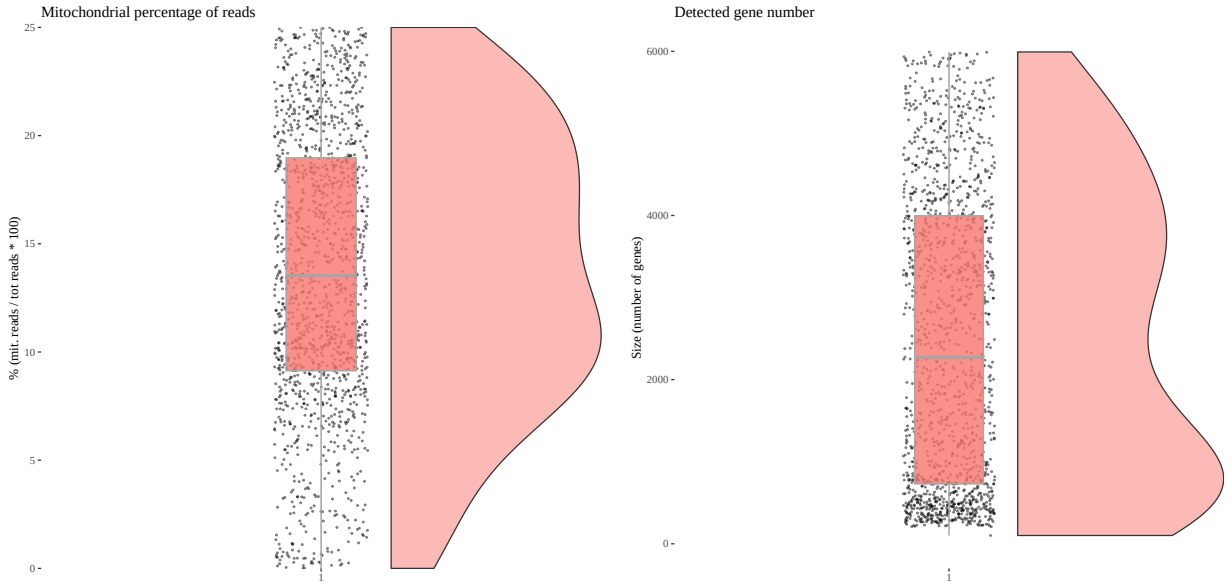

Figure 1: Percentages of mitochondrial genes and gene count numbers for the cells belonging to patient I191305 (TI EPI dataset).

at least 1000 cells. In the end, we reached the target of four patients only on the TI IMM dataset, as the other datasets did not have enough good samples.

3.1.1 Issues with the Epithelial datasets

During our preliminary analysis of the two datasets which contained the epithelial data, we quickly noticed some issues which made us question whether it made sense to analyze them.

First of all, epithelial cells are known to die quickly by anoikis during tissue dissociation. This meant that most of the cells present in the datasets were dying cells which presented very high percentages of mitochondrial genes, as shown in figure 1. As we can see in the right picture, the gene count in many patients was upper bounded by 6000, which is a strong indication of the fact that someone had already removed the cells with counts higher than that threshold. The plots also revealed how they followed a multinomial distribution, which is something that we did not see anywhere else. As a result of these warning signs, we were given permission to discard the two datasets.

3.2 Quality control and data cleaning

Once the preliminary discussions were completed, we began to analyze each chosen donor individually. The following section will focus on patient I104689, extracted from the dataset "TI IMM" (immune cells of the Terminal Ileum), but the procedure is identical for each patient. The plots for each patient are available in the Github repository. Each folder associated to a patient in a particular dataset has, inside of it, a subfolder named "plot" which contains all the figures produced.

Before performing any kind of clustering, we needed to perform some sort of data cleaning with COTAN, to remove possible outliers, doublets, and dying cells. In particular, we have performed the following steps:

Removal of cells with excessively large library sizes Each analysis started by removing the entries with excessively high library sizes. The most common threshold was 10000 genes, but there have been some exceptions.

Removal of cells with an excessively large gene count We have also removed all the cells with excessively high gene counts. The threshold was more variable than the one associated with the cell count, but it was usually

in the range of 4000 counts. To calculate this value for each entry in the dataset, COTAN originally employed a technique that involved the conversion of the sparse matrix containing our dataset into an actual instance of data.frame, which was very expensive to produce both in terms of time and space. We have managed to exploit a property of the dgCMatrix class to skip this conversion which allowed us to save minutes and gigabytes worth of computation. This exploit is now implemented in COTAN.

Removal of cells with excessively high percentages of mitochondrial genes As already discussed, the cells with an excessively high percentage of mitochondrial genes are usually dying cells, so we proceeded to remove them. Under ideal circumstances, the usual threshold should fall under ten percent. However, almost no patient allowed us to cut below such a threshold without removing less than 70 percent of the cells. As a result, the threshold usually fell between 15 and 7.5 percent.

Removal of mitochondrial genes altogether After removing the cells with excessively high percentages of mitochondrial genes, we proceed to remove the mitochondrial genes altogether from all the remaining cells. To not be confused with the previous step, which does not remove mitochondrial genes for all cells, but rather the cells with too many mitochondrial genes.

B cells removal We then proceeded to call the COTAN method clean(), after which we called the method cleanPlots to check if our ν values (refer to COTAN's paper [1] for further explanations of this value) did correlate with the resulting PCA plot, which did not happen in any of our experiments. Afterward, we determined whether the "B" cluster generated after calling the clean() could be treated as an outlier and be removed safely.

Removal of low ν genes Finally, we proceeded to remove the data points with a low ν coefficient. The threshold used in most experiments was equal to 0.35.

3.3 Pre-clustering statistics computation

Before we could proceed with the actual clustering, we needed to calculate some preliminary statistics:

- The dispertion bijection
- The gene pair coexpressions

At the same time, we have also calculated the GDI coefficients, although it was not mandatory.

4 Clustering

COTAN's clustering procedure, although computationally expensive, is relatively straightforward. For each experiment, we have:

- Called the method cellsUniformClustering()
- Added the cluster's data to our COTAN object via addClusterization()
- Calculated the cluster's coexpression coefficients through DEAOnClusters()
- Added such coefficients via addClusterizationCoex()
- Attempted to merge the mergeable clusters via mergeUniformCellsClusters()
- Added the merged clusters' data into our COTAN object.

All the plots produced by COTAN are saved in the "clustering" folder associated to each patient.

On rare occasions, we had to discard some patients because of how the cellsUniformClustering() method looped indefinitely due to all the clusters found at one iteration being non-uniform. Luckily, this only happened in large datasets featuring a lot of "spare" patients ready to replace them.

4.1 Making sense of the clusters

After we performed the clusterization, we still needed to identify what kind of cells they did represent. A way to do it would have been to analyze, for each cluster, which genes were the most expressive for that particular cluster, and then search in the literature whether there was any kind of cells known to be related to such genes. We started with this method. However, Kong et al. [2] provided us with a comprehensive list of cells that they found in our datasets, alongside their associated gene markers (see Table 1 and 2 in the supplementary material for the full list). As a result, we tried to see in which clusters those markers were particularly enriched. If similar cells' markers were to be enriched in a restricted group of clusters, it would have been a strong indication that the clusters were likely to represent such cell types. The results, as we are going to see in section 7, were so close to the ones obtained by the original paper that we have determined that it wasn't necessary to explore a different solution. This section of the experiment was carried out with the aid of the method clustersMarkersHeatmapPlot(), which allowed us to determine the cell type of each cluster through a heatmap. The plot also revealed some interesting information on the proportion of cell types present in each patient, and we found out that they closely resembled the ones shown by Kong et al., although we have not been able to explore such results for the reasons explained in section 7.1. Finally note how, in each heatmap generated by the method, the last column on the right ("Enrichment genes") contains the whole list of genes we were asked to analyze the enrichment of. This feature can help in giving a good idea of which clusters do have the highest "activity".

5 Gene enrichment analysis

The gene enrichment analysis was even more straightforward. After we performed the clustering procedure described in the previous sessions, we passed the resulting clusters, as well as our list of target genes, to a customized clustersMarkersHeatmapPlot method we have called clustersMarkersHeatmapPlotB, which in turn proceeds to call COTAN's function "geneSetEnrichment", which is the library's method designed for this kind of analysis. Our method returns a heatmap to allow for the quick identification of interesting genes. It also allows the user to calculate a dendrogram for the list of genes passed as a parameter, and it can cluster them into an arbitrary number of clusters. The cells (or better, the sets of markers associated with such cells) were inserted in the heatmap as well, so that they could be clustered alongside the genes as well. In this way, it is very easy to identify which genes do behave like the cells' markers. An example of such plots is shown in Figure 3 of the supplementary material. As per the professor's request, the plot features on the left the same heatmap which, in this patient's case, is the same as seen in Figure 1 of the supplementary material, in order to have an even better visual of in which clusters each gene is enriched in.

5.1 Identifying new cell markers

It is of particular interest to identify which genes, among the ones requested, behaved similarly to the markers of our cell types, in terms of enrichment. Even better, it would be interesting to see which genes are enriched similarly to the markers of a cell type only under a certain condition, which in our case could be whether the sample comes from a healthy, inflamed, or non-inflamed part of the intestine.

An easy way to do so would be to get the heatmaps of all the patients with a certain condition and then look, for each cell type, which are the genes which do get constantly clusterized alongside their respective cell markers by intersecting such sets. However, this procedure requires a more rigorous definition.

5.1.1 Finding the optimal number of clusters for each patient

The first thing to notice is that different patients will unavoidably cluster the genes better with different numbers of clusters between each other. So it comes naturally to ask ourselves what is the optimal number of clusters for each patient, such that the content of the clusters associated with certain cells changes as least as possible between different patients.

The most logical step would be to check, for each combination of assignments of numbers of clusters for the various patients with the same condition, some sort of metric that could tell us how stable the various clusters are.

The most common metric to estimate such "stability" is the Jaccard score: given a set of N sets $\{X_i\}_{i=0}^{N}$, the Jaccard metric is equal to:

$$\frac{\bigcap_{i=1}^{N} X_i}{\bigcup_{i=1}^{N} X_i} \tag{1}$$

In detail, we evaluated a certain assignment of cluster counts between multiple patients with the same condition by averaging the Jaccard metric calculated on the clusters where a cell type is present.

A possible future development might be using less strict metrics than the Jaccard score, such as ones that also allow genes that are present in most of the patients, but not all of them, to contribute to the final score with reduced weight. Another development would be to find a different way to select the best cluster cuts. The current one is a brute force approach with an exponential complexity that searches all the possible combinations and, for this reason, can be used only with a small number of patients.

In all our experiments, we have searched for the optimal number of clusters in the range between 12 and 20 clusters. Higher numbers of clusters were never chosen as the ideal amount for a patient, while fewer clusters resulted in large, not homogeneous groups.

5.2 Finding status-related markers

The procedure described in section 5.1 does not tell us whether the genes found do behave like a cell's markers because they are markers for the cell itself, or if they do because they are markers for the specific status of the population analyzed (healthy, inflamed or not inflamed).

The solution is simple: we can calculate the enriched genes for a particular cell type *separately for each condition*, meaning that for each cell type, we produce three sets: healthy, inflamed, and non-inflamed.

As a result, if we want to find, for example, the genes which do behave like a particular cell type's markers *exclusively* when the donor has an inflamed intestine, all we need to do is subtract the union of the genes appearing in healthy and non-inflamed donors from the set of genes present in the inflamed intestines. Using a more mathematical language, we could say that if H, I, and N are the sets of genes associated with a certain cell under the healthy, inflamed, and non-inflamed conditions, then the markers are equal to $I \backslash (H \bigcap N)$. The operation is symmetrical for the other two sets.

The only drawback of this procedure is that, if two or more cell types are clustered together, they will inherently share similar sets of markers, meaning that further studies are required to determine to which cell a gene belonged exactly. On a positive note, our results have shown that cell types are clustered together, in most cases, when they are members of the same family (for example, different types of Macrophages).

6 Pathway enrichment analysis

As another extra, we attempted to perform pathway enrichment analysis on the clusters of enriched genes obtained by our experiments, to see whether we could discover any new pathway associated with the disease.

The procedure was the following. After we clusterized the enrichment target genes for each patient as described in section 5.1.1, we proceeded to extract the pathways associated with the genes which behaved like the markers of a particular cell (meaning the ones that got clusterized alongside the markers themselves). We have used the clusterProfiler package to query the KEGG database. The code to query Wiki Pathways, DAVID, and Reactome is left commented for future studies.

Using 0.05 as the p-value threshold for all the experiments we obtained, for each cell type, a large number of pathways. This, as well as the fact that we had up to 348 sets of pathways for each experiment (29 cell types for 12 patients in the TI IMM dataset), meant that manual analysis of the resulting data was unfeasible. To extract the relevant pathways, as well as reduce the number of pathways we had to focus on, we intersected, for each cell type, the sets obtained by the various patients with a specific condition, just like we did with the genes as explained in section 5.2, only that this time we did not remove, from each intersection, the elements which appeared in the union of the other twos. This way, we obtained a set of pathways that appeared to be constantly enriched by the genes which featured an enrichment similar to the one displayed by the markers of a specific cell type.

We have also attempted to do the same analysis on the genes obtained in section 5.1, but before section 5.2, but most of them either involved only one gene or were deemed irrelevant since they were associated to viruses such as HIV or COVID-19. We will not discuss them since there is not much to discuss, but they are still available in the Github repository (they are in the folders named [dataset name]_intersect, inside the CHL_enrichment folder).

7 Results

Because of our 10-page constraint, the images accompanying the explanations will be available in either the supplementary material section or on the Github repository.

7.1 A note on the cluster sizes

If two or more clusters get merged by the method mergeUniformCellsClusters, there is a bug in COTAN that can sometimes turn the information regarding the number of cells assigned to each cluster into "NA"s (or, even worse, mix some of the results). We were forced to use a customized version of such a method (clustersMarkersHeatmapPlotB, found in utils.R) to avoid, at least, code exceptions. Because of this, we have been unable to compare our results regarding how abundant each cell type was in each patient, which would have allowed us to compare our numbers against the original paper's, although it was obvious that there were some matches (see the following sections for more details). More detailed examinations are left for future studies, once the bug is fixed.

7.2 TI IMM

The patients analyzed are listed in table 2.

Table 2: Patient data for the TI IMM dataset (on parentheses, how many cells each patient featured in the dataset).

	Patient 1	Patient 2	Patient 3	Patient 4
Healthy	H101694 (2416)	H152638 (2835)	H158108 (3671)	H180844 (4464)
Inflamed	I104689 (8027)	I130064 (11381)	I139892 (3668)	I182231 (3583)
Not inflamed	N109389 (7172)	N119540 (9316)	N130064 (23118)	N158891 (6764)

In general, the clusterization procedure tended to cluster the cells this way:

- Several, isolated clusters containing almost exclusively Plasma Cells
- Fewer, isolated clusters containing almost exclusively Mast Cells
- B cells
- All the sub-types of T-cells were clustered alongside the Tregs and the NK-Like cells
- All subtypes of Macrophages, DC1, DC2s, Monocytes and Neutrophils cells were clustered together.

The last item on the list made us wonder whether it was correct to cluster so many cell types together. We will use patient I104689 to show how our results matched the ones obtained by Kong et al. In Figure 1 of the supplementary material, we can see the heatmap representing the enrichment of each cell marker, along the clusters stated in the list, while in Figure 2 we can see a PCA from Kong et al. of the same patient (clusters annotations added manually). As we can see, the clusterization closely matches ours, validating our results.

7.2.1 Inflamed samples

The Plasma Cells and the T-cells were the most abundant type of cells present in the patients, with the first one seeming more abundant, on average, than with other cell types. The T-cells and the Macrophages clusters were the clusters with the highest enrichment of the genes we had to calculate the enrichment of.

7.2.2 Non-inflamed samples

One thing worthy of notice is that COTAN attempted to cluster the Plasma cells and the B cells in many more clusters than any other condition. The Monocytes seemed non-existent, which was something observed by Kong et al. as well, where they found reasonable quantities only inside inflamed samples. Finally, this was the only group where the "standard" Macrophages seemed to appear in high percentages (patient 119540 being the sole exception), matching once again the results observed by the authors of the original paper. Like with the inflamed samples, the T-cells and the Macrophages clusters were once again the clusters with the highest enrichment of the genes we had to calculate the enrichment of.

7.2.3 Healthy samples

The clusters which contained the Macrophages displayed much higher enrichment activity than any other cluster, including the ones containing the T-cells.

7.2.4 Enrichment results

No pathways involving more than two enriched genes per cell were found to be *persistently* present in the cells of a patient. If we consider pathways involving only one gene as well, the most interesting thing found is the fact that the "TNF signaling pathway" (hsa04668) appears to be enriched in both healthy and non-inflamed samples regularly (it appears in inflamed samples as well, but they do not appear in every patient consistently and have been discarded by the intersection as a consequence). TNFs (Tumor necrosis factor) are known to be related to Chron's disease. Other than that, most pathways are mostly related to viruses such as HIV which we can safely assume to be not be the cause of Chron's Disease (or, at least, the sole factor).

7.3 CO IMM

The patients analyzed are listed in table 3.

Table 3: Patient data for the CO IMM dataset (on parentheses, how many cells each patient featured in the dataset).

	Patient 1	Patient 2	Patient 3	Patient 4
Healthy	H139073 (3121)	H197396 (7853)		
Inflamed	I114902 (3675)	I121881 (4404)	I130084 (8073)	I175041 (1961)
Not inflamed	N104689 (14300)	N124246 (6822)	N128400 (7156)	N154787 (7888)

Unsurprisingly, the clusterization followed the same patterns as its Terminal Ileum counterpart, meaning that the Plasma cells were the most abundant cell type in the dataset. Another interesting statistic is that the "standard" macrophages did not appear to be as abundant as in the original paper.

On the inflamed samples, the Macrophages cluster was the one with the highest enrichment activity. The clusters containing the T-cells also displayed high activity, but not as high as the Macrophages did. The same phenomenon occurred with the non-inflamed and healthy samples, although the difference seemed less obvious.

7.3.1 Enrichment results

The most interesting result found regards the fact that the non-inflamed patients were the only ones to feature the "Cytosolic DNA-sensing pathway", which is involved in generating an immune response when an invading microbe's DNA is detected. However, it should be noted that it was detected inside Tregs and Macrophages LYVE1+ cells, which were clusterized in what we called the "junkyard cluster", a cluster present in each patient where the lowly-enriched genes (and weakly represented cell types) are inserted. This cluster tends to be very large as a result and, consequently, a lot of pathways are usually found here which might not necessarily be present due to the nature of the cluster.

7.4 TI STR

The patients analyzed are listed in table 4.

Table 4: Patient data for the TI STR dataset (on parentheses, how many cells each patient featured in the dataset).

	Patient 1	Patient 2
Healthy	H158108 (3906)	H180844 (1091)
Inflamed	I104689 (2001)	I130064 (20434)
Not inflamed	N130064 (24914)	N166301 (6561)

There were several macro clusterizations:

- Glial Cells
- Endothelial Cells
- Fibroblasts
- Myofibroblasts
- Pericytes

The number of patients and disparity of sizes between patients with the same condition made it difficult to extract recurrent patterns. Many patients found small numbers of clusters. The healthy samples showed a high percentage of Activated fibroblasts, Myofibroblasts, and Pericytes. The inflamed samples showed a higher presence of Endothelial than the non-inflamed ones. The non-inflamed patients had Myofibroblasts and Pericytes that in one case (N166301) were clustered together, while in the other (N130064) were in different clusters. The Endothelial clusters were the clusters with the highest enrichment of the genes, the only exception is one patient (I104689) that showed an enrichment also on Glial cells.

7.4.1 Enrichment results

Endothelial cells CD36+ displayed and enrichment of the pathway related to Endocytosis for both healthy and non-inflamed patients.

7.5 CO STR

The patients analyzed are listed in table 5.

Table 5: Patient data for the CO STR dataset (on parentheses, how many cells each patient featured in the dataset).

	Patient 1	Patient 2	Patient 3	Patient 4
Healthy	H197396 (6548)			
Inflamed	I130084 (1356)			
Not inflamed	N107306 (2181)	N124246 (2379)	N104152 (3866)	N104689 (7189)

The Fibroblast cells seem to be the most abundant in the dataset (especially the ones associated with the gene ADAMDEC1, the others being present in smaller percentages), with the Endothelial cells in second place. The Pericytes seemed to be the least represented cell type, alongside the Myofibroblasts.

Regardless of the condition, the Endothelial Cells appeared to be the ones with the highest enrichment of the targeted genes, with the Fibroblasts also displaying some activity, although reduced to the one shown by the Endothelial Cells.

7.5.1 Enrichment results

Since we only had one donor each for the inflamed and healthy samples, we found many pathways which were likely to be irrelevant (such as a pathway related to the HIV). The ones found in the non-inflamed samples were all present in the other sets as well and we have therefore considered them to be irrelevant for the condition.

8 Single Patient Analysis

Two patients (100064 and 100084) donated both inflamed and non-inflamed portions of their intestines in all the examined datasets but CO STR. We were asked, as another extra, to calculate for each patient in each dataset the intersection of the enrichment clusters obtained respectively from the inflamed and not inflamed portions of their intestines, to obtain what we called "condition independent genes", meaning genes which do appear to behave like the markers of a particular cell *regardless of the condition of the sample*, which might be of independent interest. We have used the same number of clusters for the enriched genes as calculated in section 5.1.1. The files with the results can be found on the Github repository, inside the folder "SINGLE_PATIENT_ANALYSIS". (they are the files named "[Dataset name]_[patient IDs]_OptimalIntersection.csv").

The pathways enriched in such intersections are also available in our repository, although we have not found any pathway of particular interest and, just like with the main experiments, most of them either involved only one gene, were "junkyard clusters' pathways", or were related to diseases such as COVID-19 which came into existence only in the last three years, while Chron's Disease had been around for much longer.

8.1 Conclusions

We have performed a lot of experiments with the data we had and, as a result, we have not been able to be as detailed as we wished because of our 10-page limit. We hope that the supplementary material available in the GitHub repository will be able to compensate.

Most of our experiments involved a limited amount of patients and, consequently, some of our results might not be of any statistical significance. However, many of our results, especially the ones discussed in the clusterization section, matched the ones obtained by a scientific publication that used our same datasets, suggesting that we were on the right track.

9 Contributions

The algorithms and the code were designed by both students, and they gradually got modified as the project and its needs evolved. For the analysis of the various patients, we worked in parallel, splitting the work among the two students equally. The pathway enrichment analysis was done by Ninniri, mostly because it did not require too many resources and could be completed in less than an evening.

References

[1] Silvia Giulia Galfrè, Francesco Morandin, Marco Pietrosanto, Federico Cremisi, and Manuela Helmer-Citterich. COTAN: scRNA-seq data analysis based on gene co-expression. *NAR Genomics and Bioinformatics*, 3(3), 08 2021. lqab072.

[2] Lingjia Kong, Vladislav Pokatayev, Ariel Lefkovith, Grace T. Carter, Elizabeth A. Creasey, Chirag Krishna, Sathish Subramanian, Bharati Kochar, Orr Ashenberg, Helena Lau, Ashwin N. Ananthakrishnan, Daniel B. Graham, Jacques Deguine, and Ramnik J. Xavier. The landscape of immune dysregulation in crohn's disease revealed through single-cell transcriptomic profiling in the ileum and colon. *Immunity*, 56(2):444–458.e5, 2023.