ENHANCING SINGLE-CELL RNA-SEQ ANALYSIS THROUGH THE INTEGRATION OF UNACCOUNTED INTERGENIC REGIONS

Master Thesis

Systems biology master program

Vilnius university

STUDENT NAME:	Juozapas Ivanauskas
STUDENT NUMBER:	2316457
SUPERVISOR:	dr. Simonas Juzėnas
CONSULTANT:	dokt. Justina Žvirblytė

SUPERVISOR DECISION:	.....
FINAL GRADE	.....
DATE OF SUBMISSION:	DD MMMM 20YY

Contents

1. LIST OF ABBREVIATIONS

CPM	counts per million
GRN	gene regulatory network
NGS	next generation sequencing
RT	reverse transcription
scRNAseq	single cell RNA sequencing
ORF	open reading frame
UMI	unique molecular identifier

2. INTRODUCTION

The single cell RNA sequencing becomes more and more popular tool for analysis of cellular systems. This technology enables to sequence thousands of cells and provides huge amount of data. The typicall workflow of the anaysis of such data is to map reads to the known transcriptome and and construct cell-gene matrices, which are used in the downstream analysis. However, there are several problems regarding mapping to the known transcriptome:

1. The transcriptomes used are not fully comprehensive. Even though there are given great efforts to annotate all genes, it is very likelly that not all genes are annotated, and many remains to be found. Such yet undefined genes are missed in the typical scRNAseq analysis.

2. The human (and many other species) transcriptome is very complex, with many overlapping features. This prevents mapping algorithms to assign short reads to a single feature, usually resulting in discarding such reads from analysis.

Addressing such problems could reveal some biologically significant information, that is not used in the most current scRNAseq data analysis.

3. AIM AND TASKS

Aim The aim of this project is to improve scRNA-seq analysis by investigating unassigned reads to enhance transcriptomic reference and identify potential new gene candidates.

Tasks

1. Identify genomic regions containing unassigned reads, and classiffy them as either intersecting with genes or intergenic.
2. Identify reasons why there are unassigned reads in the gene regions, and if possible, resolve the transcriptomic reference such that those reads would be assigned to genes.
3. Identify possible genomic regions containing unannotated genes, based on the unassigned intergenic reads, and check if there are any other evidence supporting existence of those genes.

4. LITERATURE REVIEW

In this chapter I will provide general review of single cell transcriptomics and related challenges.

4.1 Introduction to single cell transcriptomics

Cells are the fundamental units of life, forming the basis of all living organisms. One of the major goals of biology is to understand cellular systems and the processes occurring within cells. Since the discovery of the DNA structure in 1953 and the development of the conceptual framework for genetic information transfer, scientists have made significant efforts to sequence the genomes of various organisms. This led to the development of the first sequencing methods, such as Sanger sequencing in 1975, which laid the foundation for next-generation sequencing (NGS) technologies in use today, including the widely used Illumina platform (Heather and Chain 2016). Current sequencing methods allow us to obtain the complete genetic sequence of any organism. However, the genome alone cannot explain the full diversity of cells in multicellular organisms, as all cells share the same genome but exhibit significant variation in shape, size, and function.

RNA sequencing (RNAseq), on the other hand, enables the measurement of gene expression within cells, providing valuable insights into cellular processes. RNAseq methods largely follow DNA sequencing protocols, with the addition of a step where complementary DNA (cDNA) is synthesized from RNA (Heumos et al. 2023). The first RNAseq methods were developed for bulk sequencing, where RNA from entire cell populations is sequenced, providing an average gene expression profile across the population. Although bulk RNAseq has provided valuable insights into the dynamics of cellular processes (such as changes in disease states in response to therapeutics, detection of gene isoforms, gene fusions, and various other properties of target cells (Heumos et al. 2023)), this approach masks non-dominant processes and cell-to-cell variability through averaging. This limitation was addressed by the introduction of single-cell RNA sequencing (scRNAseq) methods, which allow for the generation of transcriptomic profiles from individual cells, providing high-resolution insights into cellular systems.

Current scRNAseq methods enable the generation of transcriptomic profiles from thousands of cells at unprecedented resolution in a single experiment. These data can be used for constructing cellular atlases (Rozenblatt-Rosen et al. 2017), understanding disease mechanisms (Z. Zhang, Chen, and X. Peng 2024), exploring cell differentiation and developmental processes (Skinner, Asad, and Haque 2024), among many other applications.

4.2 scRNAseq data generation and analysis

Something here

4.2.1 Overview of scRNAseq protocols

The generation of scRNA-seq data is a complex, multi-step process that varies across different protocols. These protocols can be grouped based on several criteria, such as the type of RNA capture (e.g., 3' end, 5' end, or full-length) or the method of cell isolation (e.g., droplet-based methods such as inDrops (Klein et al. 2015), Drop-seq (Macosko et al. 2015), and Chromium by 10X Genomics (Zheng et al. 2017), or plate-based methods such as CEL-Seq2 (Hashimshony et al. 2016) and Smart-seq2 (Picelli et al. 2013)).

In this project, datasets generated using droplet-based 3' end sequencing methods were analyzed. Therefore, these methods will be the focus of the following literature review.

3' End Sequencing and Polyadenylation 3' end sequencing captures RNA molecules using primers complementary to poly(A) tails. Polyadenylation at the 3' end is a post-transcriptional modification in which non-templated adenosines are added to the 3' end of mRNA molecules. Although the poly(A) tail is present in almost all mRNAs, its length varies and can influence mRNA fate, stability, and translation efficiency (Brouze et al. 2022).

3' end sequencing protocols exploit this feature to selectively capture RNA molecules, enabling high-throughput and cost-effective sequencing. However, some RNA molecules lack poly(A) tails and are therefore not captured by these methods, such as replication-dependent histone mRNAs (Brouze et al. 2022).

Common Steps in scRNA-seq Protocols Despite differences between scRNA-seq protocols, they share key steps: single-cell isolation, library preparation, and sequencing (Andrews and Hemberg 2018). In droplet-based methods, single cells are encapsulated in individual droplets containing hydrogel primers and a lysis mix (an example of a droplet generation device is shown in Figure 4.3). Primers used in these protocols typically share a common structure, including:

- **Cell barcodes:** Unique sequences that identify the cell from which a particular read originates (since all droplets are pooled and sequenced together).
- **Unique molecular identifiers (UMIs):** Short sequences used to quantify the original number of RNA molecules, helping to eliminate amplification bias.
- **PCR handles:** Sequences that facilitate amplification.
- **Poly-T sequences**: In 3' end sequencing methods are used to selectively capture polyadenylated RNAs (X. Zhang et al. 2019).

An example of primer design is shown in Figure 4.1. Once cells are encapsulated in droplets, lysis occurs, releasing RNA, which is then captured by the primers. A schematic overview of library preparation is provided in Figure 4.2.

Depending on the method, reverse transcription may occur within the droplets (as in inDrops and 10X Genomics) or after demulsification (as in Drop-seq). Subsequent steps typically include RNA fragmentation, PCR amplification, and next-generation sequencing (NGS).

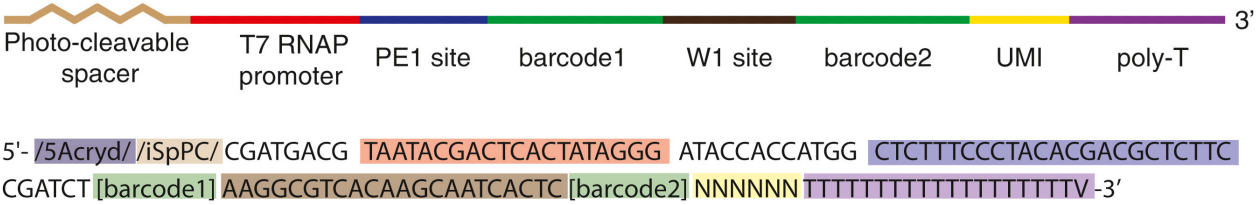


Figure 4.1: Example of primer design (inDrops). The image above shows schematic view, while below is given example with sequences. The geometry of primers varies between the protocols, the main parts (UMI site, barcodes, poly-T's) are found in most of them. Here also present are promoter region for RNA polymerase (red), sequencing primer (blue), synthesis adaptor (dark brown). Figure taken from Klein et al. (2015).

4.2.2 scRNAseq data analysis

Raw data processing The output from NGS is typically FASTQ files, containing recorded sequences, as well as (depending on method) barcode and UMI sequences, and quality scores. The subsequent processing steps include quality control of FASTQ file (based on quality scores), filtering dublicate reads (using UMIs), mapping reads to the genome sequence, assigning the reads to the genes, and finally, counting gene expression per cell (barcode) (Heumos et al. 2023) (see figure 4.4). Usually, all these steps are performed with a single piece of dedicated software, such as STARsolo (Kaminow, Yunusov, and Dobin 2021), CellRanger (Zheng et al. 2017) or other. It should be noted, that there are variations in the pipeline described above, depending on many experiment-related (e.g., whether the genome sequence or transcriptone of the study organism is known), or method-related (e.g., whether UMIs are used in the protocol) factors. The typical result of such processing is cell-gene matrix (i.e., a matrix where rows represent cells, columns represent genes, and each entry indicates the number of captured RNAs for a given gene in a specific cell).

Cell-gene matrix processing The next steps in the analysis of the scRNAseq data involves following steps:

- **Quality control** The quality of indidual cells (barcodes) can be evaluated based on several factors, such as mitochondrial gene content (apoptotic cells tend to have a higher proportion of mitochondrial genes (Heumos et al. 2023)) or total number of captured genes (very low numbers can be produced by empty droplets). In some cases, two cells can end up in one droplet, resulting in count matrix row corresponding to genes from both cells. Such matrix entries (doublets) can be filtered by using specialized software such Scrublet (Wolock, Lopez, and Klein 2019) or scDblFinder (Germain et al. 2022). Another source of noise in scRNAseq data is ambient RNA, which consists of RNA that escapes individual droplets and spreads into the medium or other droplets, leading to background noise. Even though the amount of such RNA is not high (in good quality datasets it can be around 2% (Young and Behjati 2020)), removing these RNAs from the count matrix can improve data quality. This can be achieved by identifying the background noise profile from empty droplets and adjusting the count matrix accordingly. There are dedicated softwares, such as SoupX (Young and Behjati 2020), decontX (Yang et al. 2020), CellBender (Fleming et al. 2023) and others.

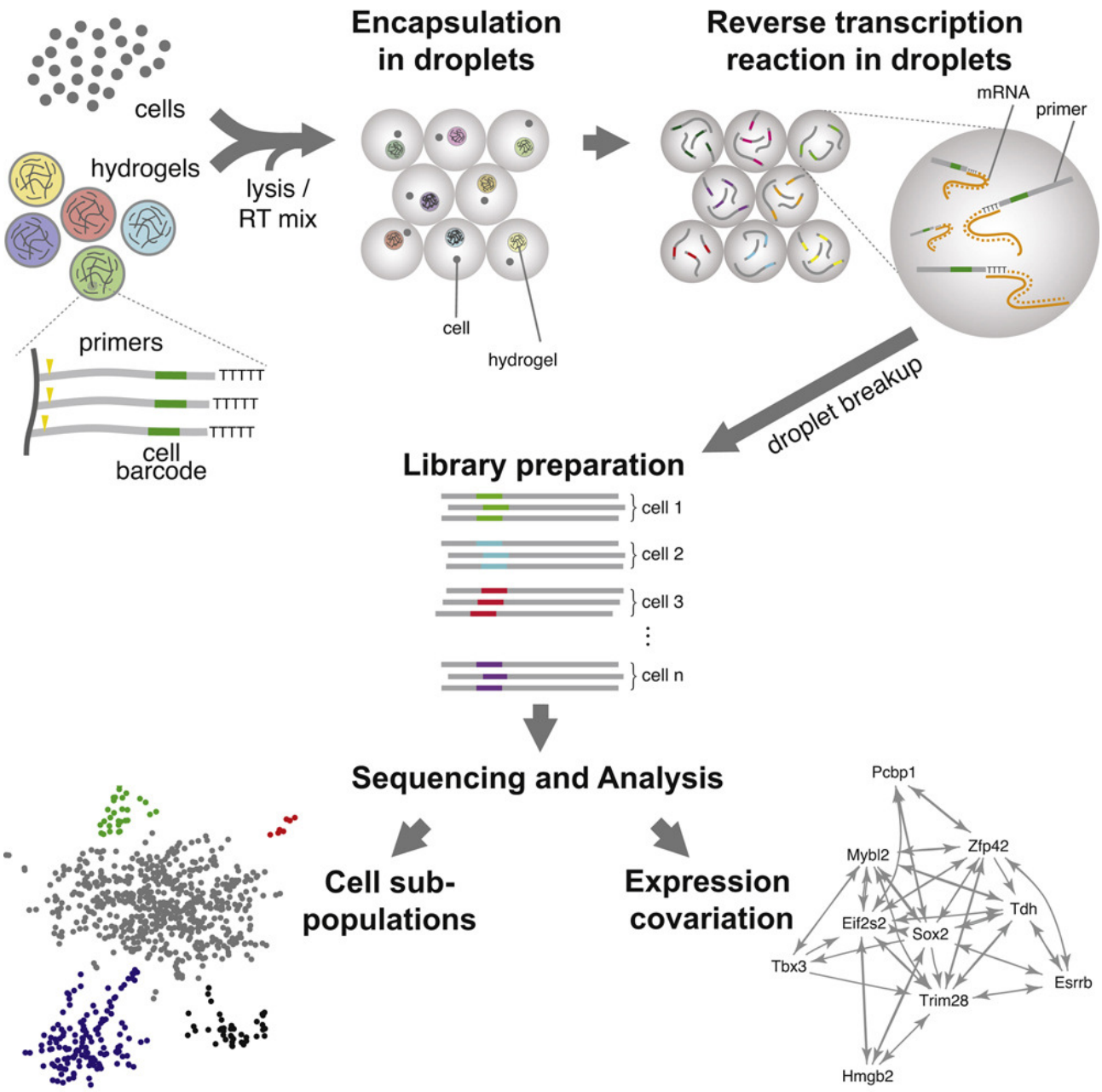


Figure 4.2: The schematic presentation of droplet-based scRNAseq (particularly inDrops, however main steps are shared between most of the droplet based protocols. Figure taken from Klein et al. (2015).

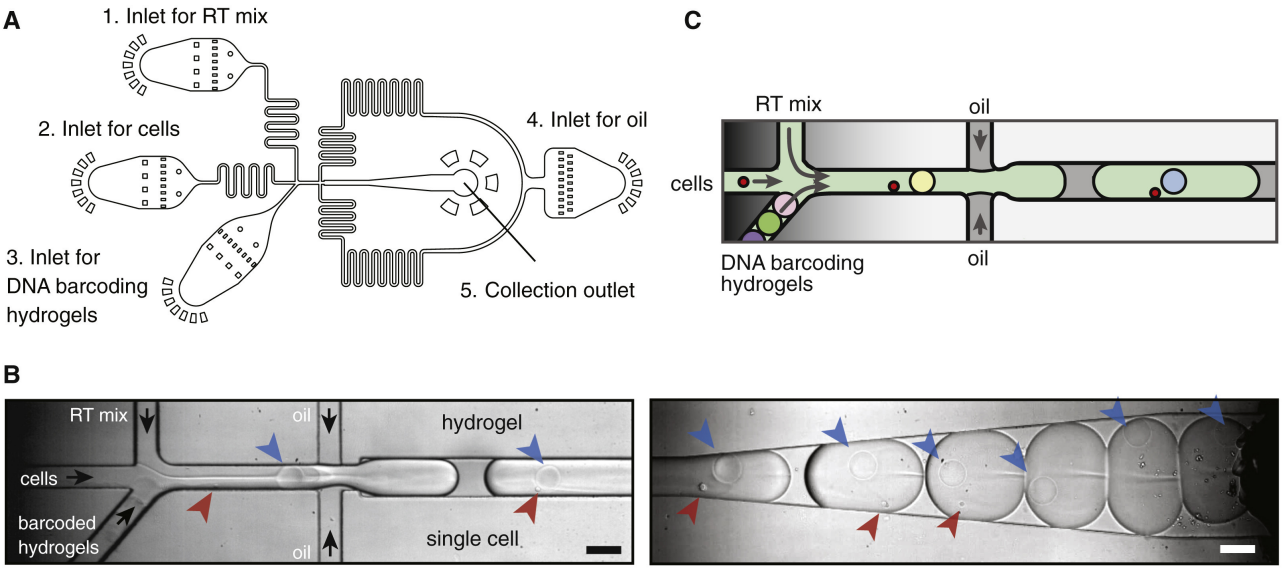


Figure 4.3: An example of microfluidics device for droplet generation (inDrops). A) Schematic view of the device. B) Snapshots of droplet generation and colection. C) Scheme of droplet generation. Figure taken from Klein et al. (2015).

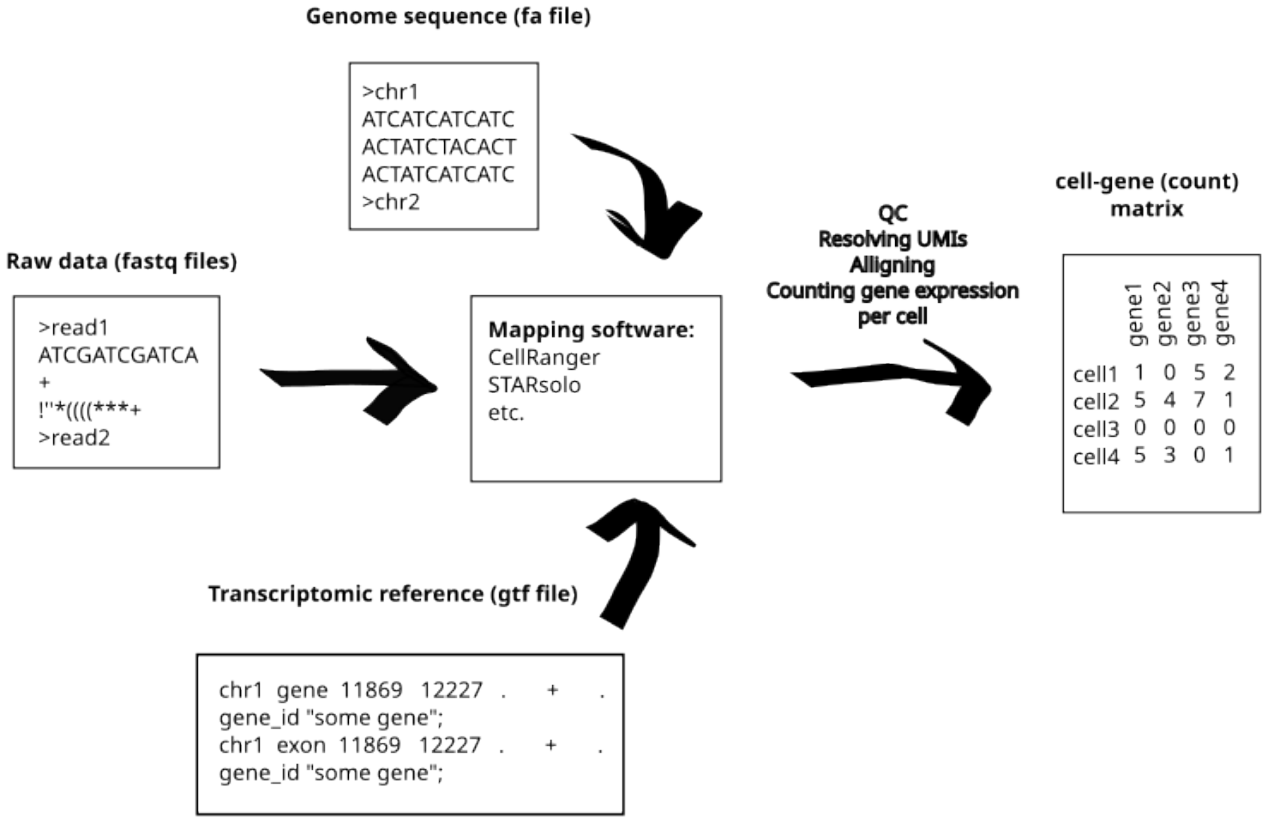


Figure 4.4: Pipeline of processing raw data.

- **Normalization** The next step in preprocessing pipeline is normalization. The goal of normalization is to transform the data so that the variation in gene expression levels is comparable, making subsequent analysis more efficient (Ahlmann-Eltze and Huber 2023). Normalization can also help eliminate biases, such as differences in sequencing depth when combining data from multiple samples (Lingen, Suarez-Diez, and Saccenti 2024). There are numerous normalization methods, based on different approaches (e.g., delta-method-based, residual-based, latent gene expression-based, count-based (Ahlmann-Eltze and Huber 2023)). Thus, selecting a normalization method should be done carefully, depending on the experimental design. General recommendations for normalization suggest comparing several methods, and if the results are similar, opting for the simpler method (Lingen, Suarez-Diez, and Saccenti 2024). Sophisticated methods do not necessarily show better results, and a recent benchmarking study by Ahlmann-Eltze and Huber (2023) has shown that simpler method (particularly the logarithm normalization, where each element y of count matrix is transformed by formula $y_{trnasformed} = log(y + 1)$) performs as well or better than more advanced methods.

- **Filtering genes** Once the data is normalized and cleaned, one can filter out non-informative genes. Initially, count matrices contain all the genes that are present in the transcriptome. However, not all of them are expressed in the sequenced data, or are expressed in negligable numbers (Heumos et al. 2023). Therefore, it is common practice to filter such genes (e.g., genes that are expressed in less than three cells). Moreover, some genes might be expressed in all the cells more or less evenly (housekeeping genes), which do not provide useful information that could be usefull in, for instance, grouping cells or determining cell types. Therefore, in many applications, it is beneficial to leave only those genes, that are highly variable between cells. In such way, the dimensionality of the count matrix is greatly reduced without loosing significant information. Additionally, genes that are outside the scope of the specific study can also be filtered out.

- **Dimensionality reduction** Even after filtering and selecting only highly variable genes, several thousand genes usually remain. It is not feasible to visualize (and hard to interpret in general) data of such high dimentionality, therefore, dimensionality reduction is essential step of subsequent analysis. The idea of dimentionality reduction is simple: to reduce the dimentions of the data loosing as little information as possible. There are number dimensionality reduction methods based on different mathematical concepts, but the most widely used today include t-SNE (Hinton and Roweis 2002), UMAP (McInnes, Healy, and Melville 2018) and principal component analysis (PCA). Although the use of these algorithms are supported by some benchmarking studies (in the study of Xiang et al. (2021), t-SNE was showed best performance, while UMAP showed the highest stability), other benchmarking studies report different findings. The study of Koch et al. (2021) suggested that such overlooked methods as latent Dirichlet alloacation (LDA) and PHATE show best performance. Meanwhile Sun et al. (2019) provided guidelines for choosing dimensionality reduction method depending on downstream analysis tasks, and in their results UMAP and tSNE were not on the top choices. Thus, while UMAP and t-SNE remain the most popular methods in the field, it is worth considering alternative methods as well.

- **Clustering and other analyses** One of the most common tasks of scRNAseq data analysis

is to identify and classify cell populations (Andrews and Hemberg 2018). This task requires to assign cells to different groups (clusters), such that cells in the same clusters are similar and distinct from cells in other clusters. There is a great variety of clustering algorithms available, including k-means, hierarchical and consensus clustering (L. Peng et al. 2020). Benchmarking studies suggest that "no individual scRNA-seq clustering algorithm can capture true clusters and achieve optimal performance in all situations" (L. Peng et al. 2020).

Clustering is usually followed by cell typing (i.e., assigning cell type to the identified clusters), which is done by finding cell type specific markers or using automatic (machine learning) tools such as CellTypist (Domínguez Conde et al. 2022). The subsequent steps in the analysis depend on the focus of the particular study and can include analysis of the dynamics of cellular systems (RNA velocity, pseudotime), inferring gene regulatory networks (GRNs), and more.

The described analysis pipeline enables insights into multicellular systems. It is evident that all downstream analyses are directly influenced by the initial steps of raw data processing, with mapping being particularly crucial in the context of this project. While mapping algorithms and tools differ, they all rely on the genome and its annotation (also refered as 'transcriptomic reference'), which directly impact their results. In the next section, I will further expand on this topic.

4.3 Genome annotation

The genome annotation process typically refers to the identification and mapping of genes within a given genome sequence (Guigó 2023). While the definition itself is straightforward, the process is highly complex. This is evident from the fact that, even more than 20 years after the first human genome assembly, human genome annotations are continuously updated with new transcripts and are expected to evolve further (Mudge et al. 2024). Figure 4.5 illustrates the changes in GENCODE annotation over time.

It is important to note that many medical and scientific research efforts rely on an accurate human gene list. Examples include genome-wide association studies (GWAS), which attempt to link genomic variants to nearby genes; RNA-seq analysis; and exome sequencing projects that use capture kits targeting most known exons (Pertea et al. 2018).

Currently, the two most widely used genome annotations are GENCODE (Mudge et al. 2024) and RefSeq (O'Leary et al. 2015) (maintained by NCBI). Despite their status as mature genome references, they report different numbers of genes. For instance, RefSeq includes 20,078 protein-coding genes (NCBI 2025), whereas GENCODE contains 19,868 (Ensembl 2025). This discrepancy highlights the ongoing challenge of accurately annotating genomes. To better understand these difficulties, the genome annotation process is first reviewed.

Genome annotation proccess RNA sequencing (RNA-seq) is the primary tool used in genome annotation. Full-length RNA sequencing allows for the capture of RNA molecules, which, when aligned to a reference genome, help construct the genome annotation of a given species (Salzberg 2019). However, RNA-seq has limitations, the most significant being its inability to capture all RNA molecules. This poses particular challenges for detecting rare transcripts, which may either be treated as noise or not captured at all (Salzberg 2019). Therefore, bioinformatics tools are necessary to

complement RNA-seq data (Guigó 2023), and most modern genome annotation pipelines integrate computational methods alongside sequencing data.

Computational genome annotation methods can be broadly categorized into two types: comparative annotation, which leverages the fact that protein-coding sequences tend to be more evolutionarily conserved, and *ab initio* annotation, which uses known sequence biases to predict genes (Guigó 2023). Despite significant advancements, genome annotation remains imperfect.

While the number of protein-coding genes is reaching a consensus — major databases report around 19,000 to 20,000 protein-coding genes — the number of non-coding genes is expected to increase in the future (Amaral et al. 2023). This is largely due to the structured nature of protein-coding genes (e.g., open reading frames, codon biases that skew nucleotide distributions), which makes them easier to detect using computational tools (Guigó 2023). Additionally, protein-coding genes have historically received greater attention because of their direct links to phenotype, leading to more experimental validation.

Current challenges in human genome annotation Despite extensive efforts, a fully comprehensive human genome annotation has not yet been achieved. Several factors contribute to this challenge.

First, the complexity of the human genome itself presents significant obstacles. Compared to the total genome length, the number of genes is relatively low, and their sequences are interrupted by introns (Salzberg 2019). Additionally, bulk RNA sequencing often has relatively low sequencing depth, making it difficult to detect rare transcripts or those expressed in a cell-type-specific manner (Guigó 2023).

In principle, full-length single-cell RNA sequencing (scRNA-seq) could help overcome some of these limitations. However, it still has inherent biases introduced during library preparation, including those related to RNA processing status, post-transcriptional modifications, transcript length, cellular localization, and structural features (Guigó 2023). Short-read scRNA-seq typically provides higher throughput (Heumos et al. 2023), enabling the detection of rare transcripts. However, it does not capture full transcript structures, limiting its utility to supporting evidence, such as validating computationally predicted genes or indicating transcriptional activity at specific genomic locations.

Beyond technical limitations, there are ontological challenges, such as defining what constitutes a gene. While genes are often regarded as well-defined, discrete entities, the reality is more complex. Coding and non-coding transcripts frequently overlap in intricate arrangements with unclear boundaries, suggesting that transcripts may not be discrete, countable units but instead form a transcriptional continuum (Salzberg 2019). Additionally, the classification systems used in genome annotations may be partially artificial. For example, some pseudogenes, generally considered non-functional copies of functional genes, are transcribed and may have biological functions (Pei et al. 2012). Similarly, the distinction between protein-coding and non-coding genes is debated, as many protein-coding loci generate both coding and non-coding transcripts, and numerous long non-coding RNAs (lncRNAs) contain potentially coding ORFs (Salzberg 2019).

Future perspectives of annotating genomes The size of eukaryotic genomes necessitates the use of automated methods for genome annotation. The accuracy of such methods depends directly on RNA capture technologies. If highly accurate and sensitive methods are developed, high-quality genome annotations can be expected (Salzberg 2019). However, at present, manual curation of human

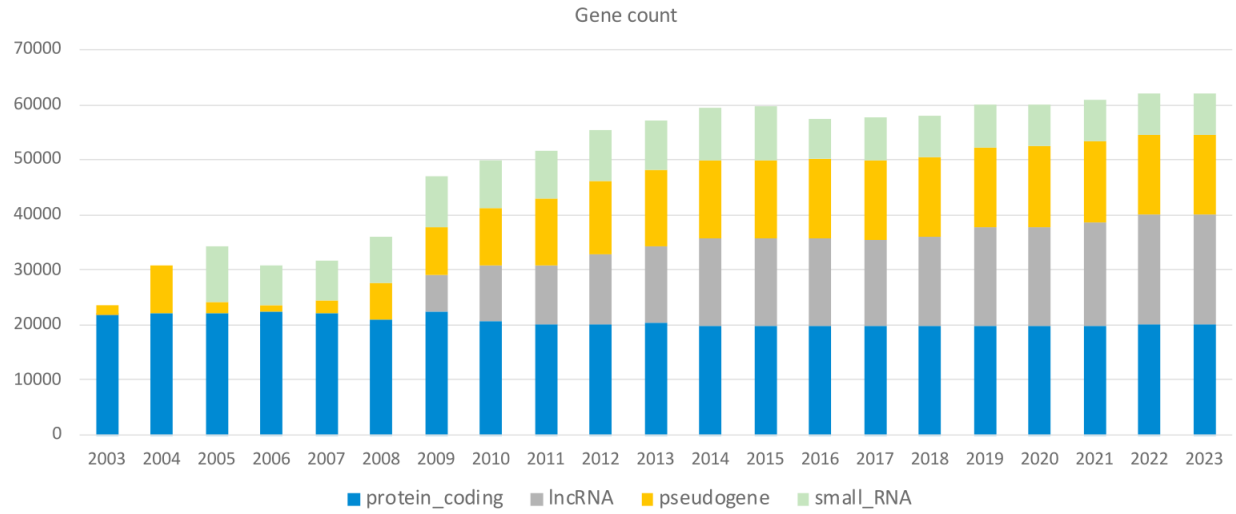


Figure 4.5: Number of different genes in the gencode annotation during time. Figure taken from paper by Guigó (2023)

genome annotations remains essential due to the imperfections of both sequencing technologies and computational tools. Given the critical role of accurate gene annotation in both medical and scientific applications, continued improvements in annotation methods remain a priority.

4.4 Transcriptomic references for scRNAseq

Having a well-annotated genome does not solve all the challenges encountered during the mapping step. In simple terms, mapping algorithms determine the genomic location to which a sequence aligns, and if an annotated gene is present at that location, the read is assigned to that gene. However, there are cases where assigning a read to a gene is not straightforward, such as when a read maps to multiple locations or to a unique location where overlapping genes are annotated.

Multimappers Reads with ambiguous origins (commonly referred to as multimappers) are typically excluded from analysis (Almeida da Paz, Warger, and Taher 2024). However, as demonstrated by (Almeida da Paz, Warger, and Taher (2024)), this approach can introduce biases that affect downstream analyses. Currently, various methods exist to handle such cases (Deschamps-Francoeur, Simoneau, and Scott 2020), yet no entirely satisfactory solution has been established (Almeida da Paz, Warger, and Taher 2024).

The aforementioned methods primarily rely on computational strategies to resolve multimappers, but the issue can also be approached from a transcriptomic reference perspective. For example, in the case of CellRanger (the mapping software from 10x Genomics) the GENCODE reference is filtered based on gene types (e.g., retaining protein-coding genes while filtering out pseudogenes) (10x Genomics 2025). While this filtering improves the number of reads included in downstream analyses, there is still room for improvement.

Pool et al. (2023) proposed three key steps to enhance transcriptomic references:

1. Including reads mapped to intronic sequences in the analysis.

2. Extending the 3' ends of certain genes.

3. Resolving overlaps between specific genes.

The first suggestion is not new in scRNA-seq research. Concepts such as RNA velocity, which rely on the ratio of spliced to unspliced RNA (La Manno et al. 2018), demonstrate that including intronic reads can provide valuable information. Moreover, most mapping tools (e.g., STARsolo, CellRanger) offer options to align reads either to exonic regions only or to entire genes. The second suggestion is based on the observation that scRNA-seq data often exhibits peaks of reads just beyond the 3' ends of genes. While the exact biological reasons remain unclear — possibly due to imprecise annotations — it is reasonable to associate these reads with the nearest genes. The third suggestion addresses gene overlaps. Reads originating from overlapping regions are often unassigned to any gene, yet in some cases, they are more likely to come from one gene rather than another. Overlapping gene resolution seeks to correct this by modifying the transcriptomic reference, either by shortening or removing certain genes.

Although Pool et al. (2023) proposed a tool to implement these strategies, it has limitations. Some of its aspects remain debatable (e.g., threshold choices), others seem unnecessary (such as handling exon-intron distinctions when most alignment tools already provide this option), and the process still requires significant manual effort. Thus, there is still a need for a more comprehensive tool for enhancing transcriptomic references — a need that will be addressed in this thesis.

Reads mapped to the intergenic regions Some reads map to the genome but not to any annotated gene. While these could simply be noise in scRNA-seq data, there is also the possibility that they contain biologically relevant information. As will be shown in the results section, scRNA-seq datasets indeed contain such reads, and they are not entirely noise.

5. METHODS

5.1 Data Acquisition

scRNAseq datasets We analyzed datasets from four different tissues: brain, blood, lung, and eye. All samples, except for two Peripheral Blood Mononuclear Cell (PBMC) datasets, were generated using the 10x Genomics v3.1 protocol. The two exceptions were prepared using the inDrops2 and inDrops protocols. All protocols capture short reads from the 3' ends of RNA molecules. The datasets are publicly available, with sources and additional details provided in Table 5.1.

sample name	tissue	cell count	donors	protocol	source
PBMC_10x	blood (PBMC)	5000	1	10x v3.1	10x genomics
PBMC_10x_2	blood (PBMC)	10000	1	10x v3.1	10x genomics
PBMC_10x_3	blood (PBMC)	10000	1	10x v3.1	10x genomics
PBMC_indrops	blood (PBMC)	2000	1	indrops2	-
PBMC_indrops_2	blood (PBMC)	9000	1	indrops(?)	-
brain	brain	6000	1	10x v3.1	Siletti et al. 2023
brain_2	brain	7000	1	10x v3.1	Siletti et al. 2023
eye	retina	10000	6	10x v3.1	Menon et al. 2019
eye_2	peripheral retina	2500	1	10x v3.1	Voigt et al. 2019
eye_3	peripheral retina	2500	1	10x v3.1	Voigt et al. 2019
lung_2	lung	5000	1	10x v3.1	Mould et al. 2021
lung_5	lung	5000	1	10x v3.1	Mould et al. 2021
lung_7	lung	4500	1	10x v3.1	Mould et al. 2021
lung_8	lung	5000	1	10x v3.1	Mould et al. 2021

Table 5.1: Datasets summary.

Genome and Transcriptomic references The human genome GRCh38 was used in this project, downloaded from the Ensembl website.

Four transcriptomic references were analyzed (see Table 5.2 for details). To ensure compatibility with the genome FASTA file, chromosome name prefixes ('chr') were removed from references that included them, as the genome file does not use these prefixes. This modification was performed using basic Linux command-line tools. Additionally, for the NCBI reference, chromosome names were converted to Ensembl-style notation using a custom Python script (e.g., 1, 2, 3 instead of NC_000001.11, NC_000002.12, NC_000003.12, etc.).

For references that do not have 'gene' entries, such entries were added (entry that spans all the components of a particular gene).

reference name	source	version
10x	10x website	2024-A
GENCODE	Gencode website	47
RefSec (NCBI)	NCBI website	GCF_000001405.40
lnc	LNCipedia website	5.2

Table 5.2: References used in this project.

Gene Prediction Tracks and Conservation Scores Gene predictions (tracks 'AUGUSTUS', 'Geneid genes', 'Gescan genes', 'SGP genes', 'SIB genes') and genome conservation scores ('phastCons100way' track) were downloaded from UCSC genome browser.

5.2 Computational Tools and Environment

All analyses were performed on a high-performance computing (HPC) cluster running a Linux environment. Software packages and tools were managed using Conda. The main tools and their versions are listed in Table ??? (is it needed) The exact Conda environment specifications (YAML file) can be found on [GitHub](GIVE LINK). Besides tools managed by conda, also basic command line tools were used (e.g. *awk, wc, grep, sed, uniq, sort* and similar). The specific functions and parameters used are described in the following sections, where the general analysis pipeline will be explained.

5.3 Enhancing transcriptomic reference

The full scripts are available on the GITHUBLINK, here only general description of the workflow and used tools provided. Here is provided general description of the pipeline:

1. Map reads with initial transcriptomic reference.
2. Take unassigned (and uniquely mapped) reads.
3. Split into intersecting and intergenic reads.
   (a) For intersecting:
      i. Resolve overlapping genes that have unassigned reads.
      ii. From the second reference and further: add genes to the original GTF that contain unassigned reads and do not overlap with entries from the original one.
   (b) For intergenic:
      i. Cluster.
      ii. Filter-out relatively small clusters (custom threshold).
      iii. For the first reference only: filter-out AT-rich reads.
      iv. For reads that are left, repeat from the beginning with the next reference.
      v. For the last reference only: clusters that start just after 3' ends are assigned to genes (i.e., extend genes).

vi. For the last reference only: add largest intergenic unexplained regions to GTF (INTERGENIC entries).

4. Create final GTF and map initial sequences to it.

5. Create list of large (>5CPM) intergenic clusters.

Mapping and filtering bam files Reads were mapped using STARsolo (Kaminow, Yunusov, and Dobin 2021), both in the case of mapping from fastq and bam input files. Parameters were used the same for all samples (see GITHUBLINK), except the ones regarding barcode geometries. Filtering unassigned reads was done using *awk*, taking those that have valid barcode (i.e. filtering out those reads that have barcode length not equal to the defined length) and were not assigned to any gene (i.e. had 'GN:Z:-' tag). Also, only uniquelly mapped reads were taken (tag 'NH:i:1').

Classifying reads as 'intergenic' or 'intersecting' and reads clustering Intersections of unassigned reads with references were checked using *bedtools intersect* command. Those that intersect with genes were classified as 'intersecting', and those that do not – as 'intergenic'. Intersections were check in strand-specific manner (bedtools '-s' flag). The intergenic reads were clustered using *bedtools merge*, again in a strand-specific manner.

Manipulating transcriptomic references The overlapping gene resolving and construction of enhanced transcriptomic reference was done using custom R script, particularly *rtracklayer* library. The criterions for the resolving of overlapping genes are following:

1. **Gene type**: prefer protein coding genes over other types, lncRNA over remaining (e.g. pseudogenes).

2. **Level**: some annotations have 'level' field, indicating if the the is verified (score 1), manually annotated (score 2) or automatically annotated (score 3). Lower 'level' score was preffered.

3. **Intersection types**: if 5' end gene is overlapping with 3' end of gene, 5' end of gene was shortened, as data we were using is generated using 3' end method, suggesting that reads in the 5' end regions of genes were not expected.

This is rough description of the usage of various tools in the pipeline given above, the pipeline itself was implemented using GNU MAKE.

5.4 Intergenic regions

Extracting intergenic regions The intergenic reads were extracted as described in the previous section (i.e. taked unassigned reads that do not overlap with any reference used) and merged into clusters using *bedtools merge* command. Clusters were filtered based on cluster size, to include at least 5 reads per million (CPM) (computed from the totol number of primary reads in the datasets). The filtering was done using *awk*.

Cluster locations were adjusted using deeptools (to compute coverage) and custom python script, to avoid reads containing long introns making the intergenic clusters very wide. To accomplish this, for

each intergenic region the maximum coverage location was found and extended to include neighbouring regions that had at least $max\_value/2$ coverage. In such way, intergenic regions were reduced to include 'peaks' of reads.

The lists acquired from all samples were then merged using *bedtools merge* function. This combined filter then was filtered to contain only regions detected in sufficient number of samples (i.e. in all 'eye', 'lung', 'brain', 'PBMC_10x' or 'PBMC_indrops' samples). Additionally, for each entry it was determined whether it overlaps with predicted genes from UCSC gene prediction archive (using *bedtools merge*) and distances to the closest GENCODE or RefSec genes were found (*bedtools closest*).

This filtered combined list were then converted into GTF format and unassigned reads from each sample were mapped using this combined intergenic annotation.

Analysis of cell-gene matrices The matrices produced by STAR were filtered, allowing cells that have sufficient number of reads (thresholds selected manually), and only genes that were expressed in at least 3 cells. Also cells were filtered based on mitochondrial gene count (allowing up to 10% of mitochondrial gene in a cell). Doublets were filtered using *scrublet*. Afterwards, matrices were normalized (using *normalize_total* function from *scanpy* package) and log-transformaed (i.e. for each entry $x = log(x + 1)$, *log1p* function from *scanpy*).

Then only highly-variable genes (min_mean=0.0125, max_mean=3, min_disp=0.5) were selected. For the clustering, principal components (PCs) were computed and optimal number of them were selected based on elbow rule (manually). For blood samples, cells were annotated automatically using CellTypist (for other samples, annotation was skipped, as it was not in the main focus of this project). Then visualizations were made using UMAP embeddings (functions from the same *scanpy* package).

All the scripts with descriptions can be found in the GITHUBLINK.

6. RESULTS

6.1 Intergenic regions

Fourteen datasets from various tissues (blood, brain, eye, and lung) were analyzed. These datasets contained between 11% and 29% unassigned uniquely mapped reads (as shown in Table 6.1) when using the 10x reference. Additionally, 6% to 18% of total reads were intergenic, meaning they did not overlap with any reference used. Given the size of the datasets, such a proportion of reads could potentially contain biologically relevant information.

??

Also, one can observe that the proportions of unassigned and intergenic reads are higher when samples are first demultiplexed by UMIs. This suggests that unassigned transcripts are present in lower amounts than assigned ones. If all unassigned transcripts originated from unannotated genes, one would expect amplification to occur similarly for both assigned and unassigned transcripts. However, this is not the case, which suggests that additional factors contribute to the discrepancy. One possible explanation is polymerase error, particularly template switching.

Kebschull and Zador (2015) demonstrated that template switching occurs during PCR amplification, albeit rarely. Notably, these events are more common in the later cycles of PCR (Kebschull and Zador 2015), meaning that the resulting chimeric transcripts undergo fewer amplification cycles and thus appear at lower abundances in the data. While this could partially account for the reduced amplification of unassigned transcripts, it is unlikely to be the sole explanation.

??????????????????????????????

Analysing integenic regions For each sample, intergenic regions containing a sufficient number of unassigned reads were identified, as described in the methods section (with a threshold set at 5 CPM). The term "intergenic" is used here in a strand-specific manner, meaning an "intergenic" region may be located on the opposite strand of a known gene.

To determine whether these intergenic regions contain biologically meaningful information, cells in each sample were clustered based solely on reads from these regions. As shown in selected examples in Figure 6.1, clustering was observed and roughly corresponded to clustering based on the standard ("10x") annotation. This indicates that biologically meaningful information is indeed present in these intergenic regions. Consequently, further analyses were performed on these regions.

The lists of intergenic regions from all samples were combined and filtered based on the number of samples in which they were detected, resulting in a final set of 2,590 intergenic regions. Of these, 147 did not overlap known genes on the opposite strand and will be referred to as "isolated". Those located on the opposite strand of known genes will be referred to as "antisense".

Sample	Total Reads	Unassigned	Intergenic	Demultiplexed	Unassigned	Intergenic
PBMC_10x	182330834	15.95%	11.37%	70305137	18.51%	12.86%
PBMC_10x_2	496387931	18.52%	14.26%	195948926	21.29%	16.32%
PBMC_10x_3	368640939	18.59%	14.34%	167804193	20.99%	16.17%
PBMC_indrops	112932507	11.12%	6.5%	8176578	22.86%	10.25%
PBMC_indrops_2	471705924	14.87%	8.36%	98063027	25.52%	13.73%
brain	206360627	16.87%	10.67%	143916177	19.02%	12.07%
brain_2	122556503	22.32%	15.98%	95031864	23.77%	17.00%
eye	375397270	28.02%	18.97%	161882445	31.87%	21.60%
eye_2	140981808	29.35%	14.85%	69146995	31.43%	15.99%
eye_3	161261977	29.46%	18.86%	68157696	32.11%	20.58%
lung_2	511080104	15.99%	12.07%	269118747	17.98%	13.33%
lung_5	452105505	14.49%	10.74%	256353198	16.19%	11.83%
lung_7	524095146	18.41%	14.34%	236157103	20.64%	15.79%
lung_8	342092138	14.53%	10.66%	217801055	16.07%	11.67%

Table 6.1: Statistics of unassigned and intergenic reads per sample. Unassigned reads were counted after mapping with 10x reference. Intergenic reads here are those that do not intersect with any reference used. 'Demultiplexed' column shows number of total reads after demultiplexing using UMIs, and following 'unassigned' and 'intergenic' percentages were computed compared to this number.

6.1.1 Isolated intergenic regions

The intergenic regions that are distant from known genes and contain reads may indicate novel genes. To determine whether these are genuine signals rather than artifacts, several aspects were considered:

1. Differential expression: Are these regions differentially expressed in any samples, i.e., are they specific to certain cell types? If so, this strongly suggests a biological origin rather than an artifact.

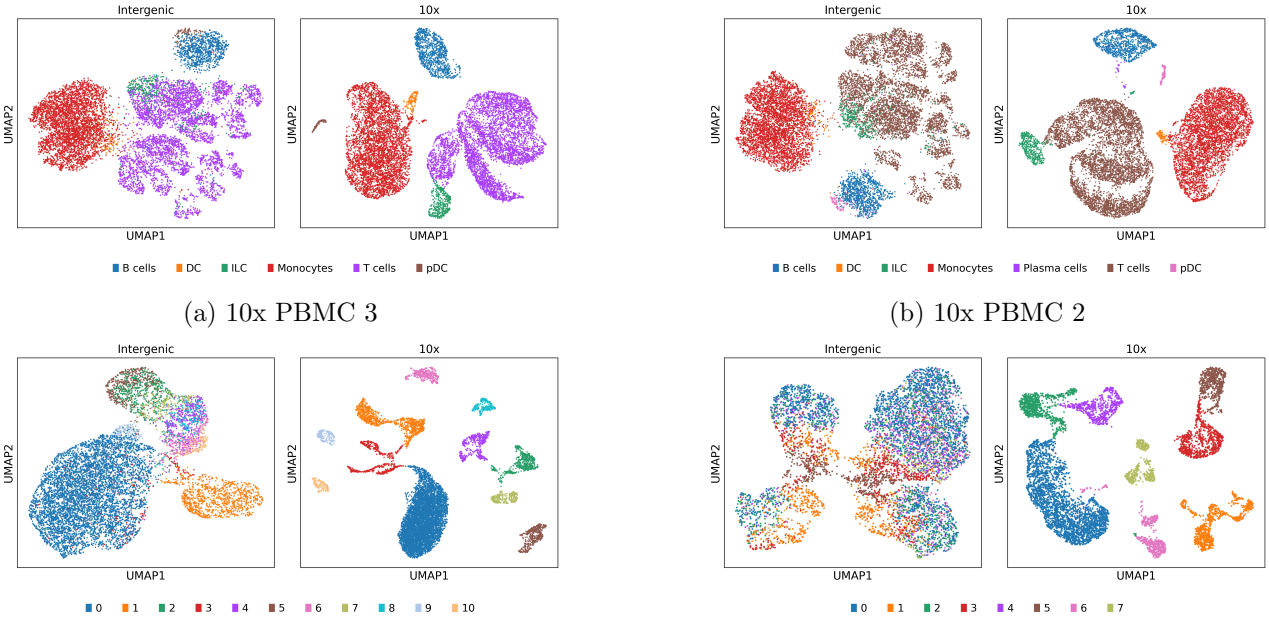
2. AT-rich sequences: Do nearby sequences contain 10 consecutive A/Ts? This could explain polymerase binding but does not necessarily confirm whether the region represents a valid gene or an error.

3. Open chromatin: Are open chromatin regions present upstream? Such regions often indicate active transcription.

4. Conservation score: Coding regions tend to be more evolutionarily conserved than non-coding regions.

5. Gene predictions: Do these regions overlap with predicted genes identified by computational tools?

Differential expression analysis To assess differential expression, all data samples were first clustered. For *PBMC_10x* samples, clustering and cell annotation were performed using CellTypist. For other samples, the Leiden algorithm was used, with manually chosen parameters to ensure a comparable number of clusters across similar datasets. For instance, all *lung* samples were clustered into five groups, aligning approximately with visible UMAP structures. The UMAP visualizations, colored by clusters, can be found in Appendix 14.1.



(a) 10x PBMC 3

(b) 10x PBMC 2

(c) eye

(d) 10x Brain 2

Figure 6.1: Comparison of clustering using standard annotation ('10x' reference) and using only defined intergenic regions. As can be seen, for some samples intergenic regions are sufficient for rough clustering, for others (e.g. 'brain_2' sample), noise gives some clustering artefacts. Nevertheless, this rough clustering implies that these unassigned read contain biological information.

After filtering those regions that were differentially expressed (tresholds were used 0.01 for adjusted p-value and 0.5 for 'logfoldchange'), the 29 differentially expressed isolated intergenic regions were found. Some examples can be seen in Figure 6.2.

AT-rich regions AT-rich regions (defined as 10 consecutive A/Ts, allowing one mismatch) were identified upstream of the 3' ends of isolated intergenic regions, with the first occurrence recorded.

The distances from isolated intergenic regions to the nearest AT-rich sequences varied, with a mean of 938.537 and a median of 508. The distribution is shown in Figure 6.3. To determine whether these distances support the hypothesis that these locations are not random noise, comparisons were made against 10,000 random genomic locations and 10,000 randomly selected gene 3' ends. The results, summarized in Table 6.2, indicate that distances are larger for both randomly selected genes and random genomic locations. Notably, distances from random locations to AT-rich sequences are shorter on average than those from randomly selected gene 3' ends. This may be due to the fact that gene bodies, which typically lack AT-rich sequences, contribute to these distances. Additionally, these results suggest that if small genes are indeed present between these isolated intergenic regions, they should be relatively short.

Open chromatin Open chromatin regions may provide further evidence of transcriptional activity when located upstream. However, distances to the nearest known genes must also be considered, as upstream open chromatin sites may not be associated with the intergenic regions but rather with other genes.

For each intergenic region, distances to the nearest open chromatin sites and upstream genes

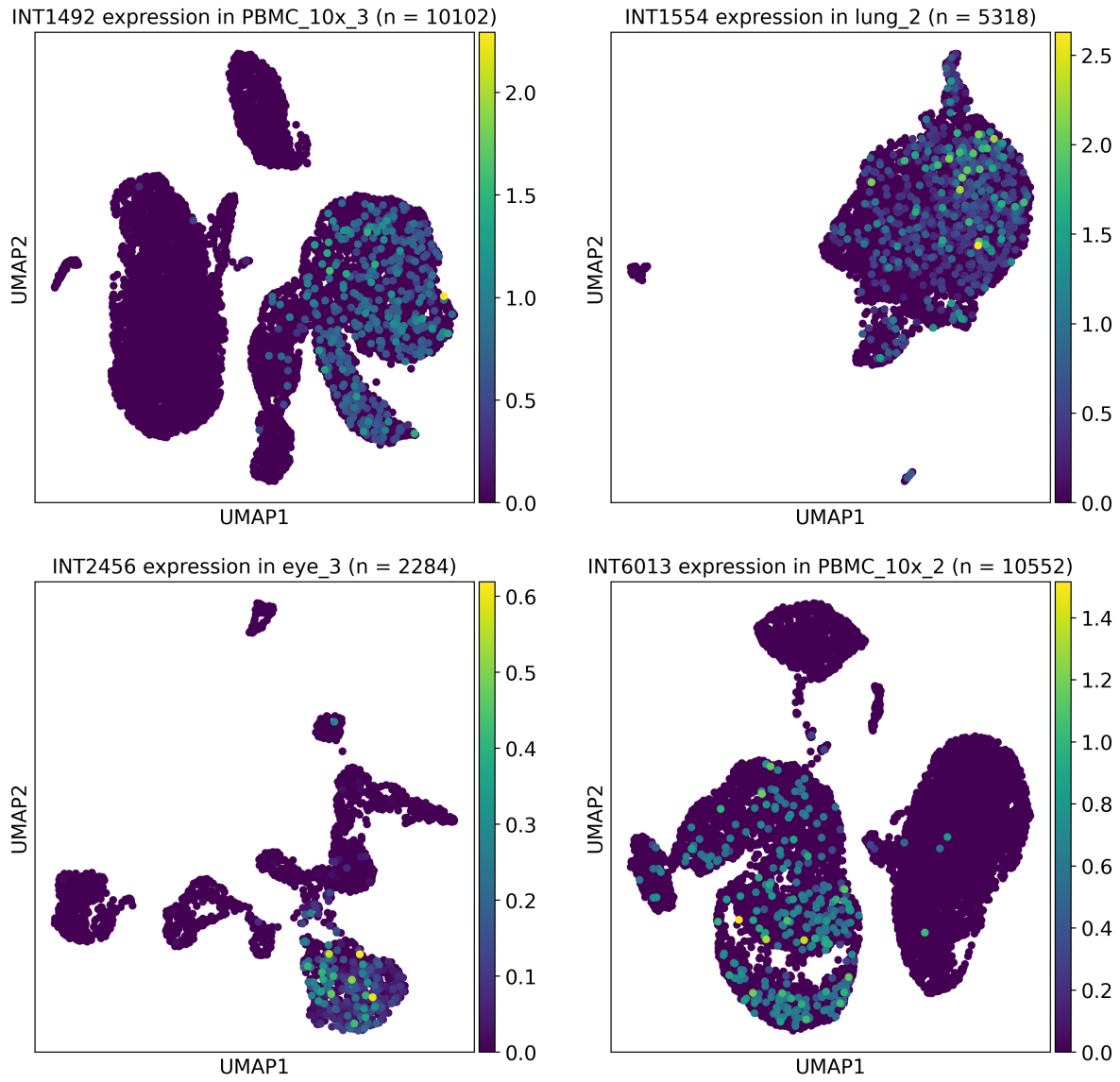


Figure 6.2: Examples of differentially expressed isolated intergenic regions.

	mean	median
isolated intergenic 3' ends	938.537	508
3' ends of random genes	1391.019	849
random locations	1123.194	682

Table 6.2: The distances to the closest AT-rich regions.

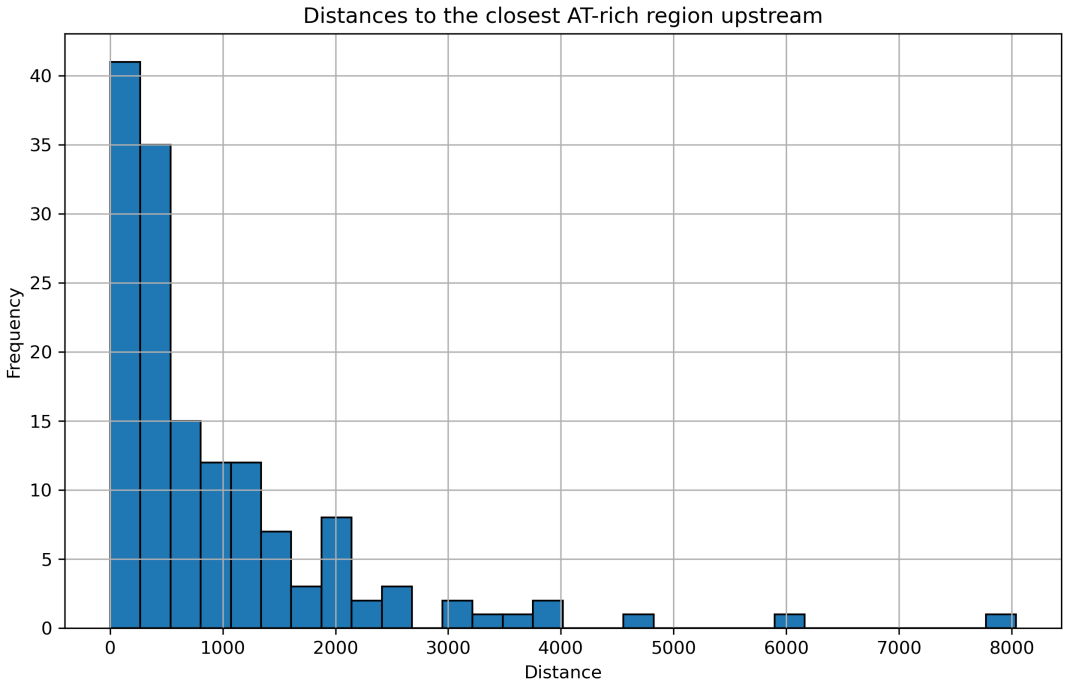


Figure 6.3: Distances from 3' ends of isolated intergenic regions to closest AT-rich region upstream.

(strand-unspecific) were calculated. Of the 147 isolated intergenic regions, 66 had open chromatin regions closer than any known upstream genes. To ensure that these sites were not merely associated with distant genes, an additional filter was applied, requiring that the closest open chromatin region be at least 2,000 bp from the nearest gene. This yielded 40 intergenic regions.

Furthermore, if these open chromatin regions function as promoters for the isolated intergenic regions, they should also contain AT-rich sequences. This was confirmed: 38 of the 40 intergenic region-associated open chromatin sites contained AT-rich segments.

Conservation scores A high conservation score in an intergenic region suggests potential functionality, providing additional evidence for biological relevance. To identify such regions, those with conservation scores above 0.6 that did not overlap with known genes from references used were selected. The resulting regions are presented in Table 6.3.

Name	Genomic coordinates	Conservation score	Found in samples
INT1216	1:72282700-72282900	0.7534172617	brain, eye
INT1829	1:77772950-77773050	0.6863111111	lung
INT2199	6:148122550-148122750	0.6364459652	brain, eye
INT2208	5:18162950-18163100	0.6138651685	PBMC (indrops)
INT3636	X:129412750-129413050	0.7218984354	brain

Table 6.3: Conserved intergenic regions not overlapping with known genes from ncbi and gencode annotations.

Gene predictions To assess whether isolated intergenic regions overlap with predicted genes, UCSC gene prediction archives were examined, focusing on overlaps with 3' ends of predicted genes.

Of the identified regions, 12 overlapped with predicted genes. However, three of these did not overlap with the 3' ends of predicted genes, which would be expected given that the data was generated using 3' end sequencing methods. Five predicted genes were extensions of known genes, while the remaining four corresponded to genes absent from RefSeq and NCBI references but supported by our data. Table 6.4 summarizes these findings. Only regions overlapping the 3' ends of predicted genes were considered as supported by gene prediction tools.

Name	Genomic coordinates	Prediction tool	Overlaps with 3' region of predicted gene
INT196	6:89086200-89086500	SIB	Yes (PNRC1)
INT827	9:94111950-94112200	SIB	Yes
INT1216	1:72282700-72282900	SIB	No (5' end)
INT1387	11:63570800-63571050	SIB	No
INT1525	8:11842200-11842350	SIB	Yes (CTSB)
INT1801	5:151272550-151272700	SIB	Yes (GM2A)
INT2044	19:4041150-4041400	SIB	Yes (ZBTB7A)
INT4070	4:47430500-47430700	SIB	Yes
INT4147	1:34861500-34861700	SIB	Yes (DLGAP3)
INT4577	5:702050-702300	SIB	Yes
INT4948	5:703250-703450	SIB	Yes
INT5710	4:47428500-47428700	SIB	No

Table 6.4: Isolated intergenic regions overlapping with predicted genes from UCSC gene prediction archive. The gene in the brackets shows if the predicted gene is extended version of already annotated genes.

Combining supporting features In total, five features were examined as supportive evidence for the biological significance of these intergenic regions: conservation, AT-rich/open chromatin presence, differential expression, and gene predictions. Based on these, four sublists of isolated intergenic regions were generated:

- Conserved regions
- Regions associated with AT-rich sequences and open chromatin
- Differentially expressed regions
- Regions overlapping predicted genes

Notably, the list of conserved regions did not intersect with the other three. The predicted gene list had two regions in common with the differential expression list (INT196, INT1525) and one in common with the open chromatin list (INT4147). All three of these predicted genes were extended versions of already annotated genes. The differential expression and open chromatin lists had 11 regions in common (INT1312, INT1426, INT1492, INT1554, INT1609, INT1993, INT2209, INT349, INT368, INT4216, INT6013). No regions appeared in more than two lists.

6.1.2 Antisense intergenic regions

Interpretation of antisense intergenic regions is more complicated. The features such as open chromatin, AT-richness or conservativness are not strand specific, meaning that if they would be present, they could not be attributed with certainty for our intergenic regions.

The differential expression can be checked, however, even if the antisense intergenic region is differentially expressed, it still can be an artifact, originating, for example, from template switch (Kebschull and Zador 2015).

Hence the only real supporting evidence would be computational gene prediction.

Gene predictions As in the case of isolated intergenic regions, antisense intergenic regions were checked for overlaps with predicted genes from UCSC prediction archive. There were 49 such regions found, out of them 35 were in the 3' ends of predicted genes. 9 of those 35 predicted genes were longer version of already annotated genes in RefSeq or GENCODE references. The all list can be seen in the Table 6.5.

Name	Genomic coordinates	Prediction tool	Overlaps with 3' region of predicted gene
INT7	6:24718850-24719100	SIB	Yes
INT33	5:172767250-172767500	SIB	Yes (DUSP1)
INT134	15:41280550-41280800	SIB	Yes (extends a bit after)
INT218	5:61409150-61409400	SPG	No (5' end)
INT285	10:110898200-110898550	SIB	Yes (BBIP1)
INT327	1:169132100-169132350	SIB	Yes (NME7)
INT347	14:75277700-75278000	Gescan	No
INT382	12:11892250-11892500	Gescan	Yes
INT386	6:26215200-26215450	SIB	Yes (H2BCB?)
INT438	4:39713300-39713600	SIB	Yes (almost all (short) gene spanned)
INT533	8:130110900-130111150	SIB	Yes
INT617	15:85733950-85734200	SIB	Yes
INT621	9:75148850-75149400	SIB	Yes (OSTF1)
INT651	1:28579100-28579350	SIB	Yes (TRNAU1AP)
INT654	6:34382850-34383050	SIB	No (but short gene)
INT828	8:133035800-133036050	SIB	Yes (SLA)
INT859	2:71373500-71373750	SIB	No (slight overlap on 5' end)
INT898	17:51190400-51190700	SIB	Yes (extends after)
INT903	8:22613400-22613650	SIB	Yes
INT949	9:111693050-111693400	SIB	Yes
INT972	15:64954800-64955350	SIB	Yes
INT984	18:63291900-63292200	Geneid	Yes
INT1028	5:50414200-50414500	SIB	Yes
INT1155	1:36296050-36296300	SIB	No (5' end)
INT1231	10:19890700-19891050	Gescan	Yes
INT1402	5:142786700-142786950	SIB	No (5' end)
INT1440	2:235746450-235746700	SIB	Yes
INT1445	9:40885650-40885900	SIB	Yes
INT1548	7:50342700-50342950	Gescan	No (exon in the middle)
INT1616	17:30895450-30895700	SIB	No (5' end)
INT1749	3:172333550-172333750	SIB	No (5' end)
INT1931	17:82522100-82522350	SIB	Yes
INT2065	10:103606150-103606400	SIB	Yes
INT2158	14:49636100-49636350	SIB	No (5' end, DNAAF2)
INT2194	7:130521250-130521500	SIB	No (but gene is short)
INT2371	1:14632650-14632900	Geneid	No (middle exon)
INT2380	22:33853900-33854150	Geneid	Yes
INT2825	13:45300050-45300300	Geneid	No (middle exon)
INT2979	4:98929350-98929600	SIB	No (5' end, EIF4E)
INT3328	2:184604250-184604500	SIB	Yes
INT3429	17:48172750-48173150	Augustus	Yes
INT3504	15:25332950-25333150	SIB	Yes (UBE3A)
INT3849	3:121432200-121432400	SIB	Yes
INT3868	17:31448500-31448700	SIB	Yes
INT4458	5:44820700-44820900	SIB	Yes (MRP530)
INT4743	20:3188900-3189050	SIB	Yes (DDRGK1)
INT5002	11:92232750-92232950	SIB	Yes
INT5315	20:3183750-3184050	SIB	Yes
INT5329	13:95603000-95603200	SIB	Yes (extends after)

Table 6.5: Antisense intergenic regions overlapping with predicted genes from UCSC gene prediction archive. The gene in the brackets shows if the predicted gene is extended version of already annotated genes.

7. DISCUSSION

8. CONCLUSIONS

9. RECOMMENDATION

10. ACKNOWLEDGEMENTS

11. REFERENCES

(1) 10x Genomics (2025). *10x Transcriptomic References*. URL: https://www.10xgenomics.com/support/software/cell-ranger/downloads/cr-ref-build-steps (visited on 02/19/2025).

(2) Ahlmann-Eltze, Constantin and Wolfgang Huber (Apr. 2023). "Comparison of transformations for single-cell RNA-seq data". In: *Nature Methods* 20.5, pp. 665–672. ISSN: 1548-7105. DOI: 10.1038/s41592-023-01814-1.

(3) Almeida da Paz, Michelle, Sarah Warger, and Leila Taher (May 2024). "Disregarding multimappers leads to biases in the functional assessment of NGS data". In: *BMC Genomics* 25.1. ISSN: 1471-2164. DOI: 10.1186/s12864-024-10344-9.

(4) Amaral, Paulo et al. (Oct. 2023). "The status of the human gene catalogue". In: *Nature* 622.7981, pp. 41–47. ISSN: 1476-4687. DOI: 10.1038/s41586-023-06490-x.

(5) Andrews, Tallulah S. and Martin Hemberg (Feb. 2018). "Identifying cell populations with scRNASeq". In: *Molecular Aspects of Medicine* 59, pp. 114–122. ISSN: 0098-2997. DOI: 10.1016/j.mam.2017.07.002.

(6) Brouze, Aleksandra et al. (May 2022). "Measuring the tail: Methods for poly(A) tail profiling". In: *WIREs RNA* 14.1. ISSN: 1757-7012. DOI: 10.1002/wrna.1737.

(7) Deschamps-Francoeur, Gabrielle, Joël Simoneau, and Michelle S. Scott (2020). "Handling multi-mapped reads in RNA-seq". In: *Computational and Structural Biotechnology Journal* 18, pp. 1569–1576. ISSN: 2001-0370. DOI: 10.1016/j.csbj.2020.06.014.

(8) Domínguez Conde, C. et al. (May 2022). "Cross-tissue immune cell analysis reveals tissue-specific features in humans". In: *Science* 376.6594. ISSN: 1095-9203. DOI: 10.1126/science.abl5197.

(9) Ensembl (2025). *Ensembl website*. URL: https://www.ensembl.org/Homo_sapiens/Info/Annotation (visited on 02/13/2025).

(10) Fleming, Stephen J. et al. (Aug. 2023). "Unsupervised removal of systematic background noise from droplet-based single-cell experiments using CellBender". In: *Nature Methods* 20.9, pp. 1323–1335. ISSN: 1548-7105. DOI: 10.1038/s41592-023-01943-7.

(11) Germain, Pierre-Luc et al. (May 2022). "Doublet identification in single-cell sequencing data using scDblFinder". In: *F1000Research* 10, p. 979. ISSN: 2046-1402. DOI: 10.12688/f1000research.73600.2.

(12) Guigó, Roderic (Aug. 2023). "Genome annotation: From human genetics to biodiversity genomics". In: *Cell Genomics* 3.8, p. 100375. ISSN: 2666-979X. DOI: 10.1016/j.xgen.2023.100375.

(13) Hashimshony, Tamar et al. (Apr. 2016). "CEL-Seq2: sensitive highly-multiplexed single-cell RNA-Seq". In: *Genome Biology* 17.1. ISSN: 1474-760X. DOI: 10.1186/s13059-016-0938-8.

(14) Heather, James M. and Benjamin Chain (Jan. 2016). "The sequence of sequencers: The history of sequencing DNA". In: *Genomics* 107.1, pp. 1–8. ISSN: 0888-7543. DOI: 10.1016/j.ygeno.2015.11.003.

(15) Heumos, Lukas et al. (Mar. 2023). "Best practices for single-cell analysis across modalities". In: *Nature Reviews Genetics* 24.8, pp. 550–572. ISSN: 1471-0064. DOI: 10.1038/s41576-023-00586-w.

(16) Hinton, Geoffrey E and Sam Roweis (2002). "Stochastic Neighbor Embedding". In: *Advances in Neural Information Processing Systems*. Ed. by S. Becker, S. Thrun, and K. Obermayer. Vol. 15. MIT Press. URL: https://proceedings.neurips.cc/paper_files/paper/2002/file/6150ccc6069bea6b5716254057a194ef-Paper.pdf.

(17) Kaminow, Benjamin, Dinar Yunusov, and Alexander Dobin (May 2021). "STARsolo: accurate, fast and versatile mapping/quantification of single-cell and single-nucleus RNA-seq data". In: DOI: 10.1101/2021.05.05.442755.

(18) Kebschull, Justus M. and Anthony M. Zador (July 2015). "Sources of PCR-induced distortions in high-throughput sequencing data sets". In: *Nucleic Acids Research*, gkv717. ISSN: 1362-4962. DOI: 10.1093/nar/gkv717.

(19) Klein, Allon M. et al. (May 2015). "Droplet Barcoding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells". In: *Cell* 161.5, pp. 1187–1201. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.04.044.

(20) Koch, Forrest C et al. (Aug. 2021). "Supervised application of internal validation measures to benchmark dimensionality reduction methods in scRNA-seq data". In: *Briefings in Bioinformatics* 22.6. ISSN: 1477-4054. DOI: 10.1093/bib/bbab304.

(21) La Manno, Gioele et al. (Aug. 2018). "RNA velocity of single cells". In: *Nature* 560.7719, pp. 494–498. ISSN: 1476-4687. DOI: 10.1038/s41586-018-0414-6.

(22) Lingen, Henk J. van, Maria Suarez-Diez, and Edoardo Saccenti (Dec. 2024). "Normalization of gene counts affects principal components-based exploratory analysis of RNA-sequencing data". In: *Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms* 1867.4, p. 195058. ISSN: 1874-9399. DOI: 10.1016/j.bbagrm.2024.195058.

(23) Macosko, Evan Z. et al. (May 2015). "Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets". In: *Cell* 161.5, pp. 1202–1214. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.05.002.

(24) McInnes, Leland, John Healy, and James Melville (2018). *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. DOI: 10.48550/ARXIV.1802.03426.

(25) Menon, Madhvi et al. (Oct. 2019). "Single-cell transcriptomic atlas of the human retina identifies cell types associated with age-related macular degeneration". In: *Nature Communications* 10.1. ISSN: 2041-1723. DOI: 10.1038/s41467-019-12780-8.

(26) Mould, Kara J. et al. (Apr. 2021). "Airspace Macrophages and Monocytes Exist in Transcriptionally Distinct Subsets in Healthy Adults". In: *American Journal of Respiratory and Critical Care Medicine* 203.8, pp. 946–956. ISSN: 1535-4970. DOI: 10.1164/rccm.202005-1989oc.

(27) Mudge, Jonathan M et al. (Nov. 2024). "GENCODE 2025: reference gene annotation for human and mouse". In: *Nucleic Acids Research* 53.D1, pp. D966–D975. ISSN: 1362-4962. DOI: 10.1093/nar/gkae1078.

(28) NCBI (2025). *NCBI website*. URL: https://www.ncbi.nlm.nih.gov/datasets/gene/GCF_000001405.40/ (visited on 02/13/2025).

(29) O'Leary, Nuala A. et al. (Nov. 2015). "Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation". In: *Nucleic Acids Research* 44.D1, pp. D733–D745. ISSN: 1362-4962. DOI: 10.1093/nar/gkv1189.

(30) Pei, Baikang et al. (Sept. 2012). "The GENCODE pseudogene resource". In: *Genome Biology* 13.9. ISSN: 1474-760X. DOI: 10.1186/gb-2012-13-9-r51.

(31) Peng, Lihong et al. (Mar. 2020). "Single-cell RNA-seq clustering: datasets, models, and algorithms". In: *RNA Biology* 17.6, pp. 765–783. ISSN: 1555-8584. DOI: 10.1080/15476286.2020.1728961.

(32) Pertea, Mihaela et al. (Nov. 2018). "CHESS: a new human gene catalog curated from thousands of large-scale RNA sequencing experiments reveals extensive transcriptional noise". In: *Genome Biology* 19.1. ISSN: 1474-760X. DOI: 10.1186/s13059-018-1590-2.

(33) Picelli, Simone et al. (Sept. 2013). "Smart-seq2 for sensitive full-length transcriptome profiling in single cells". In: *Nature Methods* 10.11, pp. 1096–1098. ISSN: 1548-7105. DOI: 10.1038/nmeth.2639.

(34) Pool, Allan-Hermann et al. (Sept. 2023). "Recovery of missing single-cell RNA-sequencing data with optimized transcriptomic references". In: *Nature Methods* 20.10, pp. 1506–1515. ISSN: 1548-7105. DOI: 10.1038/s41592-023-02003-w.

(35) Rozenblatt-Rosen, Orit et al. (Oct. 2017). "The Human Cell Atlas: from vision to reality". In: *Nature* 550.7677, pp. 451–453. ISSN: 1476-4687. DOI: 10.1038/550451a.

(36) Salzberg, Steven L. (May 2019). "Next-generation genome annotation: we still struggle to get it right". In: *Genome Biology* 20.1. ISSN: 1474-760X. DOI: 10.1186/s13059-019-1715-2.

(37) Siletti, Kimberly et al. (Oct. 2023). "Transcriptomic diversity of cell types across the adult human brain". In: *Science* 382.6667. ISSN: 1095-9203. DOI: 10.1126/science.add7046.

(38) Skinner, Oliver P, Saba Asad, and Ashraful Haque (June 2024). "Advances and challenges in investigating B-cells via single-cell transcriptomics". In: *Current Opinion in Immunology* 88, p. 102443. ISSN: 0952-7915. DOI: 10.1016/j.coi.2024.102443.

(39) Sun, Shiquan et al. (Dec. 2019). "Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis". In: *Genome Biology* 20.1. ISSN: 1474-760X. DOI: 10.1186/s13059-019-1898-6.

(40) Voigt, A.P. et al. (July 2019). "Molecular characterization of foveal versus peripheral human retina by single-cell RNA sequencing". In: *Experimental Eye Research* 184, pp. 234–242. ISSN: 0014-4835. DOI: 10.1016/j.exer.2019.05.001.

(41) Wolock, Samuel L., Romain Lopez, and Allon M. Klein (Apr. 2019). "Scrublet: Computational Identification of Cell Doublets in Single-Cell Transcriptomic Data". In: *Cell Systems* 8.4, 281–291.e9. ISSN: 2405-4712. DOI: 10.1016/j.cels.2018.11.005.

(42) Xiang, Ruizhi et al. (Mar. 2021). "A Comparison for Dimensionality Reduction Methods of Single-Cell RNA-seq Data". In: *Frontiers in Genetics* 12. ISSN: 1664-8021. DOI: 10.3389/fgene.2021.646936.

(43) Yang, Shiyi et al. (Mar. 2020). "Decontamination of ambient RNA in single-cell RNA-seq with DecontX". In: *Genome Biology* 21.1. ISSN: 1474-760X. DOI: 10.1186/s13059-020-1950-6.

(44) Young, Matthew D and Sam Behjati (Dec. 2020). "SoupX removes ambient RNA contamination from droplet-based single-cell RNA sequencing data". In: *GigaScience* 9.12. ISSN: 2047-217X. DOI: 10.1093/gigascience/giaa151.

(45) Zhang, Xiannian et al. (Jan. 2019). "Comparative Analysis of Droplet-Based Ultra-High-Throughput Single-Cell RNA-Seq Systems". In: *Molecular Cell* 73.1, 130–142.e5. ISSN: 1097-2765. DOI: 10.1016/j.molcel.2018.10.020.

(46) Zhang, ZhenWei, MianMian Chen, and XiaoLian Peng (July 2024). "Integrated analysis of single-cell and bulk RNA-sequencing identifies a signature based on drug response genes to predict prognosis and therapeutic response in ovarian cancer". In: *Heliyon* 10.13, e33367. ISSN: 2405-8440. DOI: 10.1016/j.heliyon.2024.e33367.

(47) Zheng, Grace X. Y. et al. (Jan. 2017). "Massively parallel digital transcriptional profiling of single cells". In: *Nature Communications* 8.1. ISSN: 2041-1723. DOI: 10.1038/ncomms14049.

12. SUMMARY

13. SUMMARY IN LITHUANIAN

14. APPENDICES

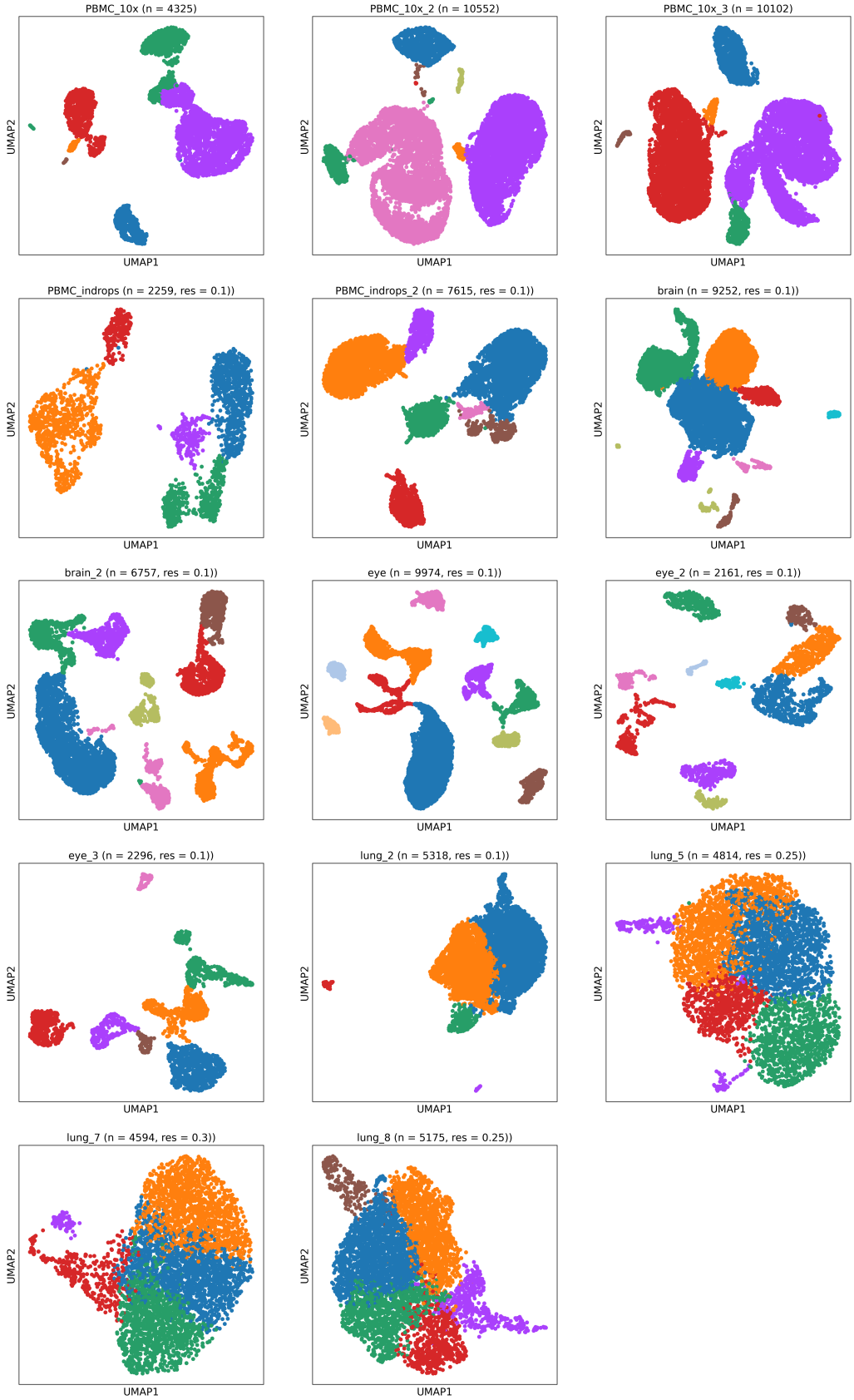


Figure 14.1: Clusterings of datasets. 'n' stands for number of cells in the plot, 'res' for the resolution parameter of leiden algorithm. *PBMC_10x* samples are colored by CellTypist annotations.