Single Cell protocols
From raw reads to cell clusters in a few easy steps

- Part 1 -

Roelli Patrick, Gardeux Vincent, Kanev Kristyian, Biocanin Marjan
27.06.18

Overview

11:00 Single-cell RNA seq and pre-processing

- Introduction to single cell RNA seq
- Introduction to Snakemake
- Hands-on session

12:15 Downstream analysis on ASAP

13:00 Poster Session and Lunch

13:30 Extended workshop

14:45 End

Why Single-cell RNA-seq ?

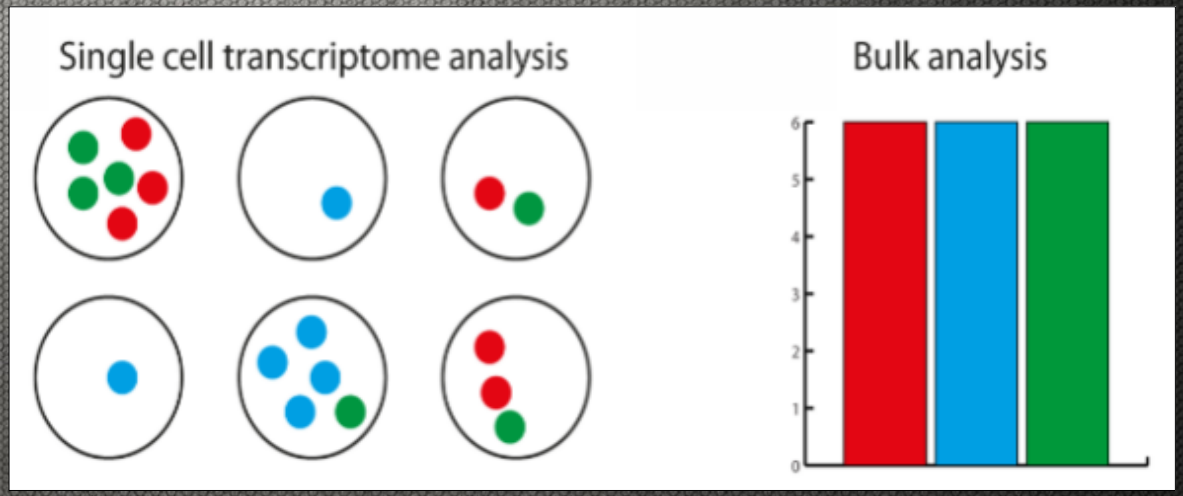
Single cell RNA-seq

Bulk RNA-seq

Why Single-cell RNA-seq ?

Single cell RNA-seq

Bulk RNA-seq



Macaulay IC, Voet T (2014) PLoS Genet

RNA-seq => average of thousands of cells.

But mRNA expression varies between cells

Single cell RNA

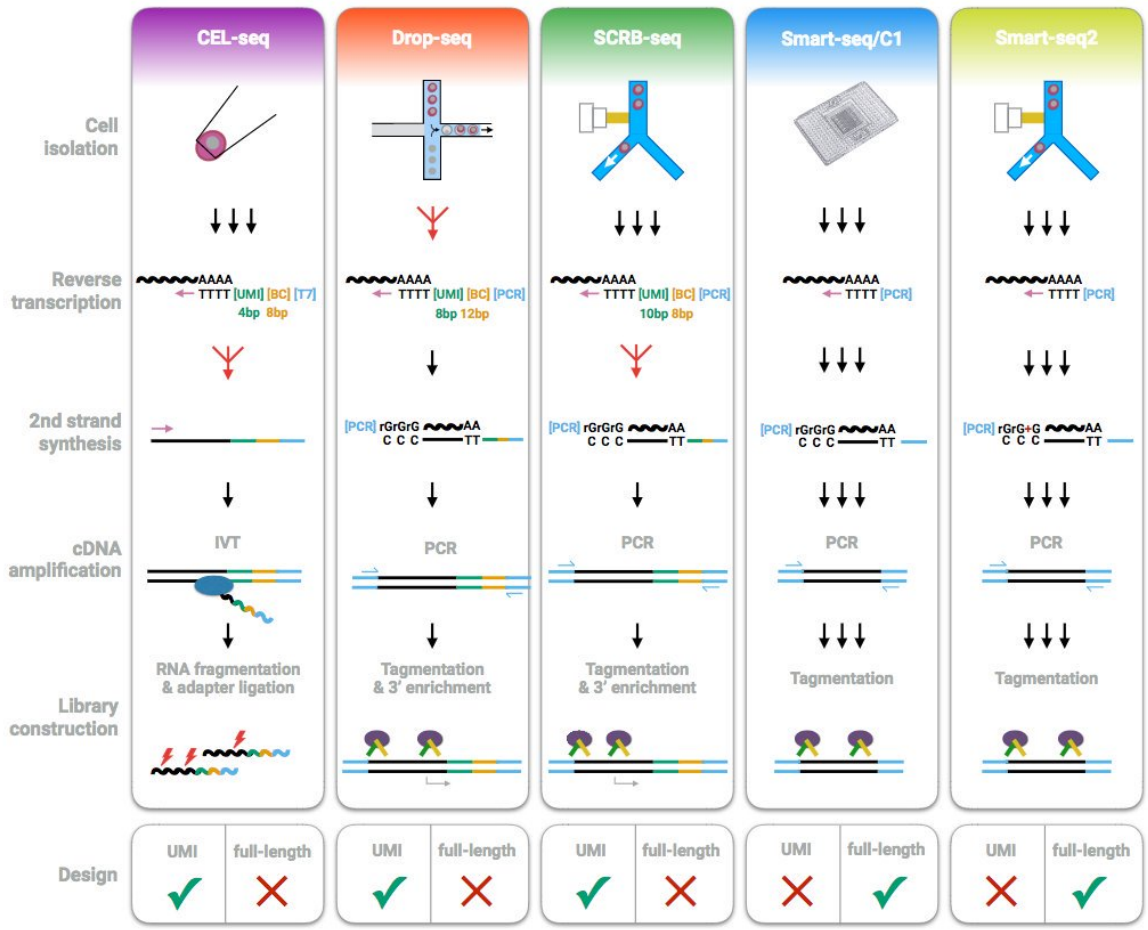


Figure 2 | Schematic overview of key library preparation steps in each method analyzed in this study.

Comparative analysis of single-cell RNA sequencing methods, Christoph Ziegenhain et al. 2016, biorxiv
https://www.biorxiv.org/content/early/2016/06/29/035758

Single cell RNA

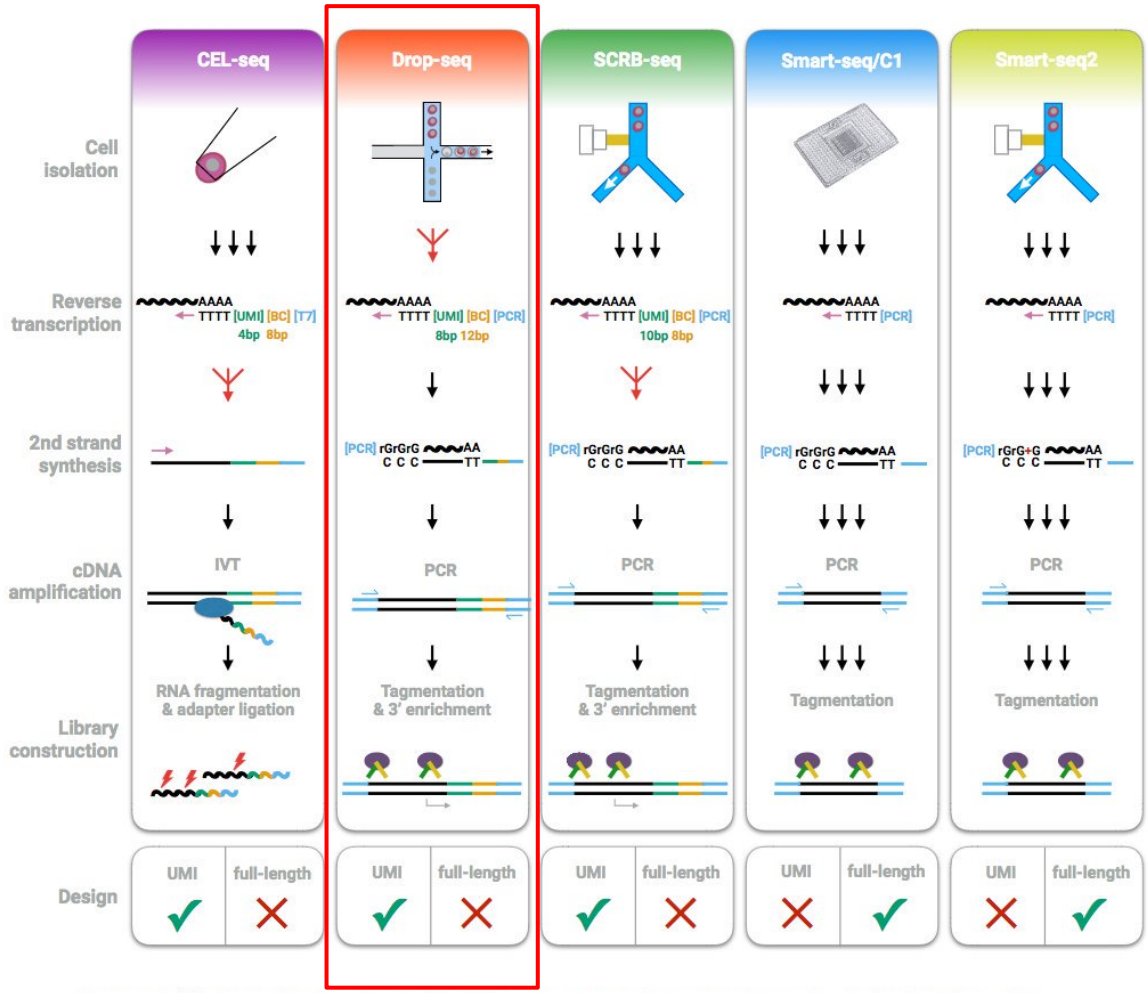
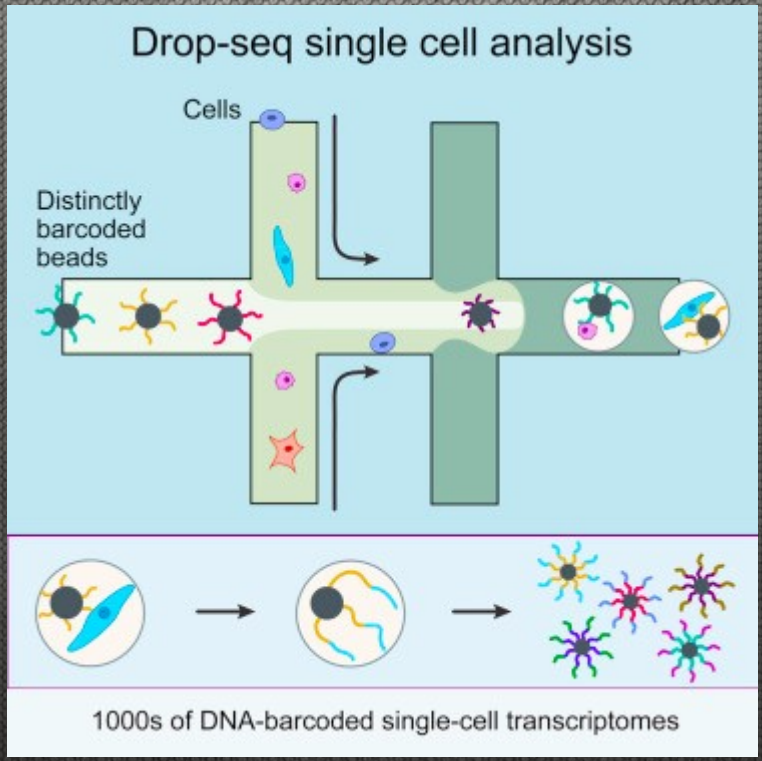


Figure 2 | Schematic overview of key library preparation steps in each method analyzed in this study.

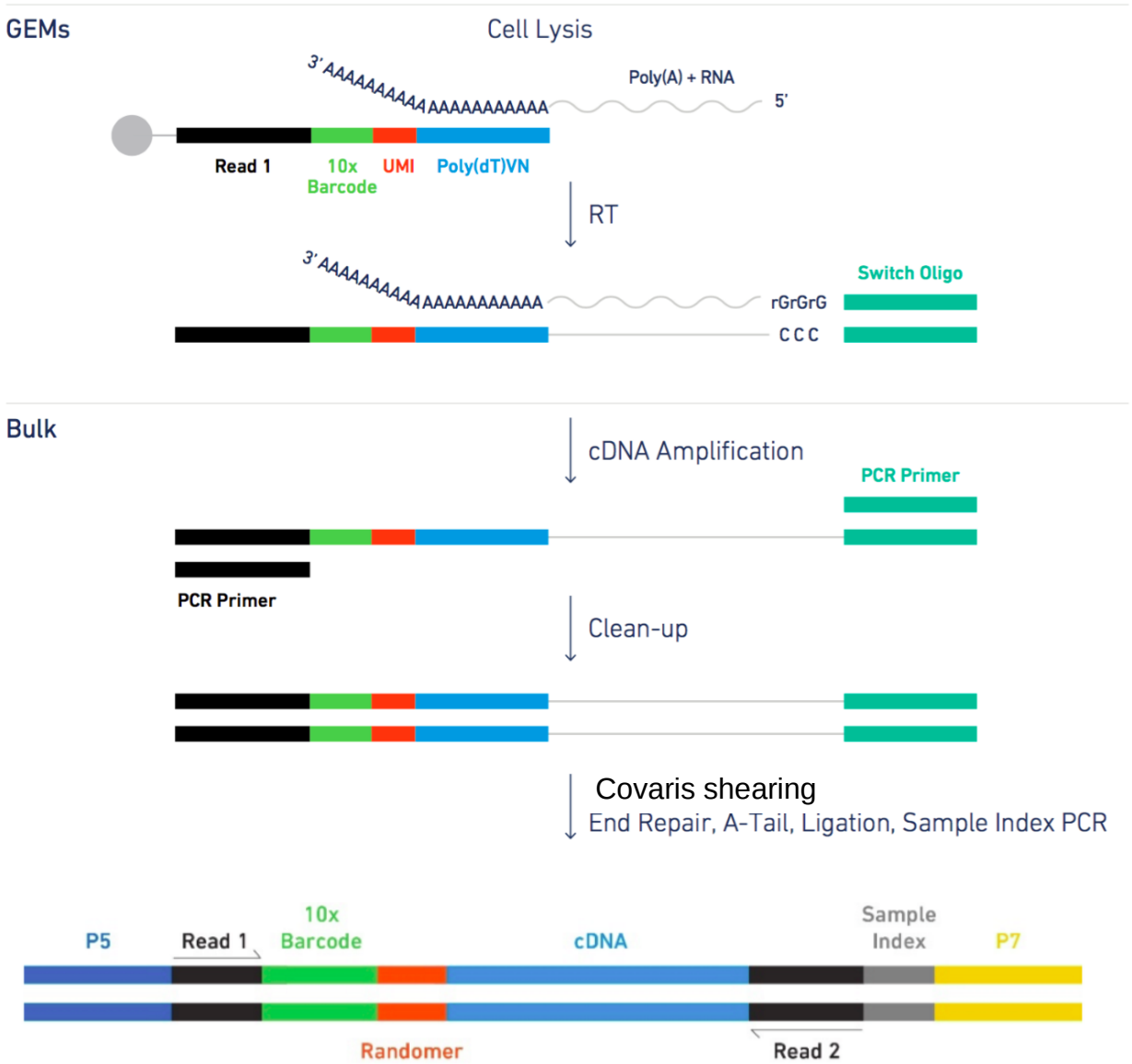
Comparative analysis of single-cell RNA sequencing methods, Christoph Ziegenhain et al. 2016, biorxiv
https://www.biorxiv.org/content/early/2016/06/29/035758

Droplet based protocols



Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets, Macosko, Evan Z. et al.Cell , Volume 161 , Issue 5 , 1202 - 1214

10x protocol

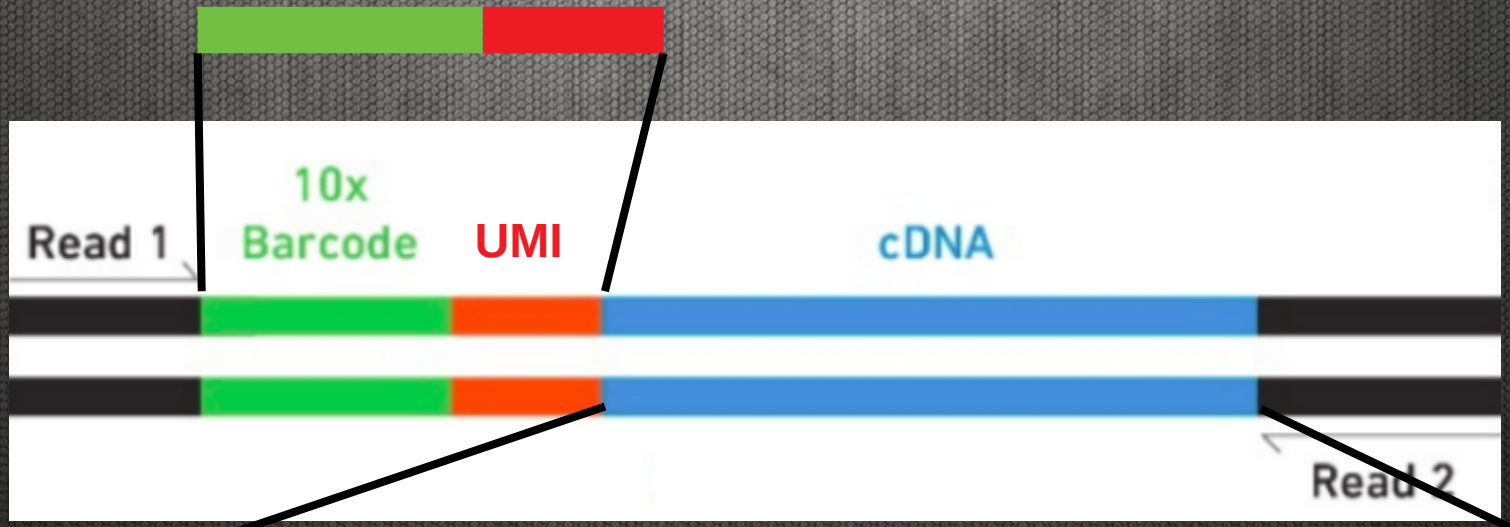


http://core-genomics.blogspot.com/2016/

3’ end capturing data structure

ReadID
Sequence
Base quality

```
@ST-K00126:307:HFM3NBBXX:1:1101:3772:1244 1:N:0:NTCGCCCT
NCATTTGAGTAACCCTGATGTCATAA
+
#AAFFJJJJJJJJJJJJJJJFJJJJJ
```

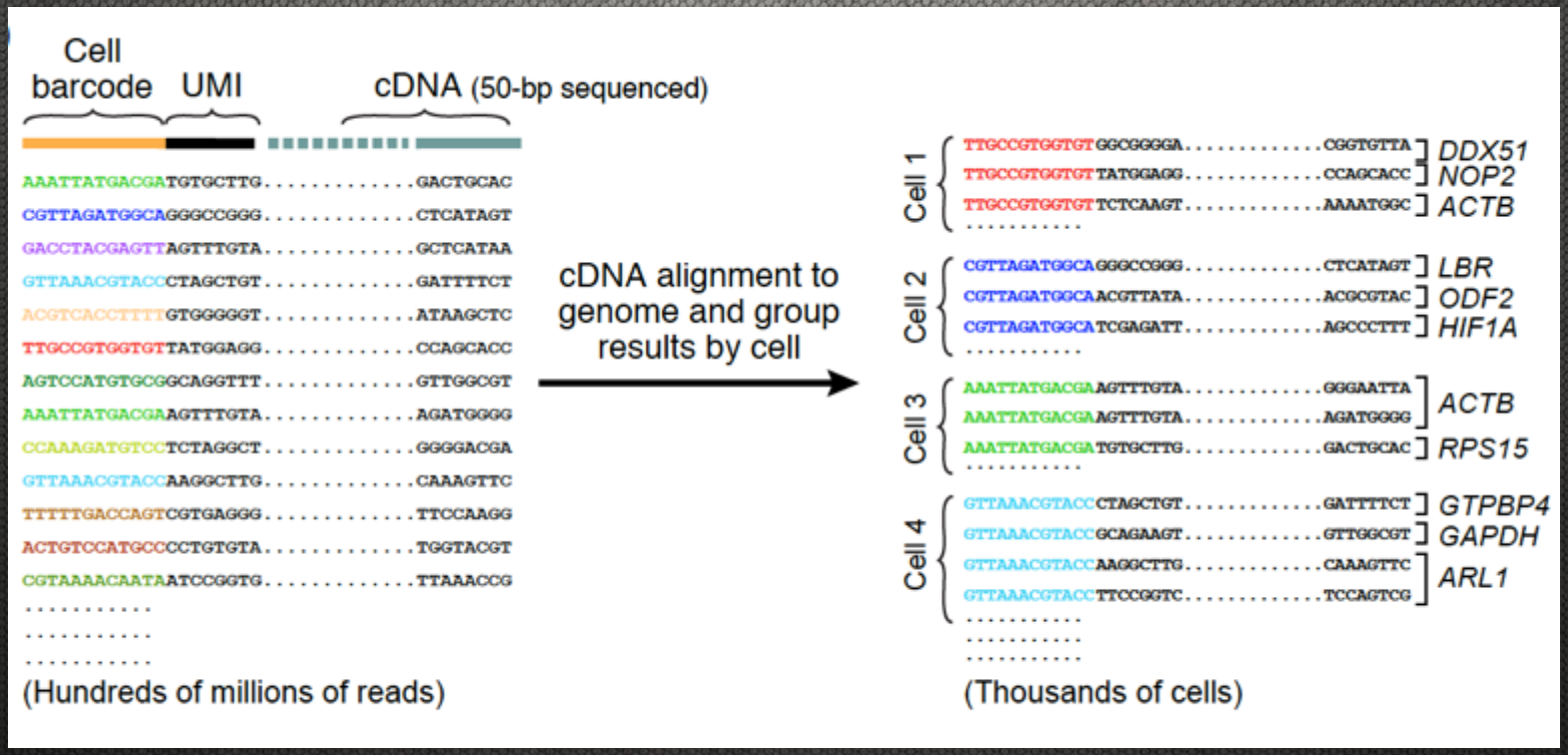


ReadID
Sequence
Base quality

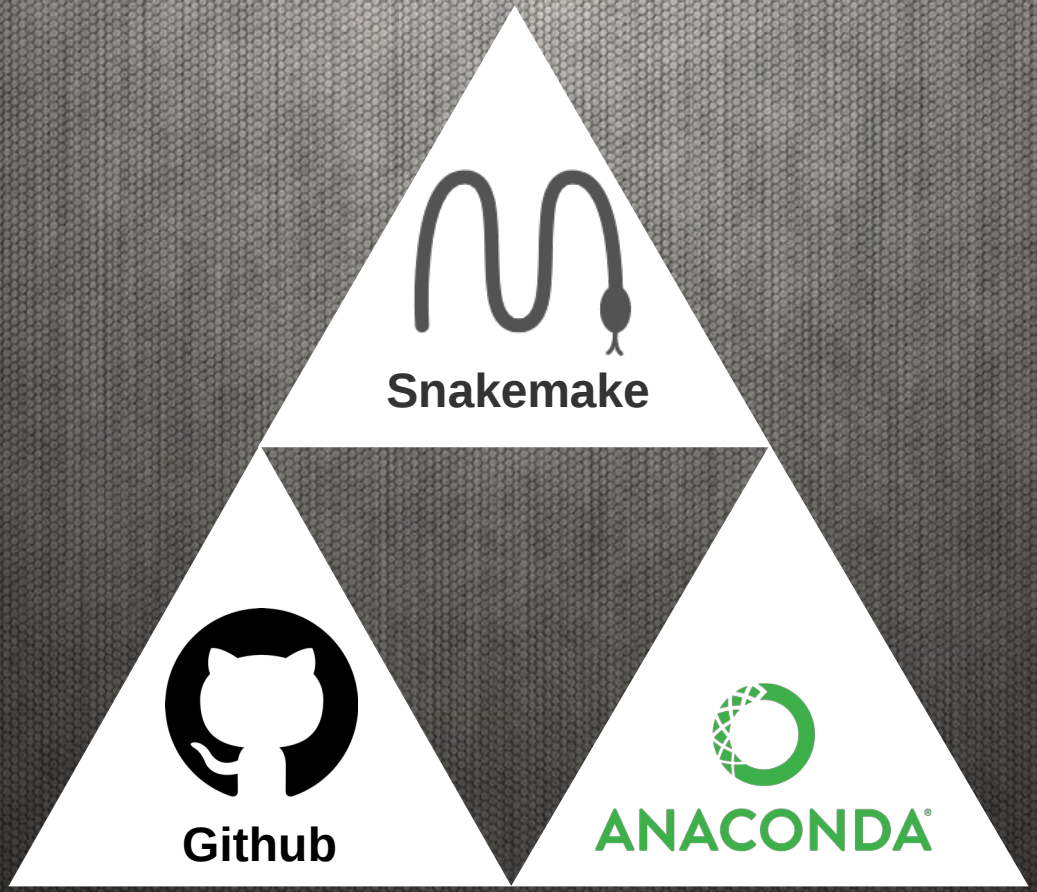
```
@ST-K00126:307:HFM3NBBXX:1:1101:3772:1244 2:N:0:NTCGCCCT
NAAGCCAGTTGTGAATCATGCACATCAGCTCCTTCTGAAATGTGTTTATGGCCTAGGACACAGGGACCCTGGAGACTATGGTGCTGCAGTGCATTATG
+
#<<A<FJJJFJFJJJJJJJJJFJFJJJJJJJJJJJJJJJFJJFFJJJJJAFJJFJF7JJJJFJAJJJ<J<7-A<FFFFJ-F<FJJJJJJJJJJ7FJJA
```

UMI: Unique molecular identifier

From raw data to count table



dropSeqPipe



https://github.com/Hoohm/dropSeqPipe

dropSeqPipe

Snakemake

Running parameters storage
Parallel processing
Cluster usability

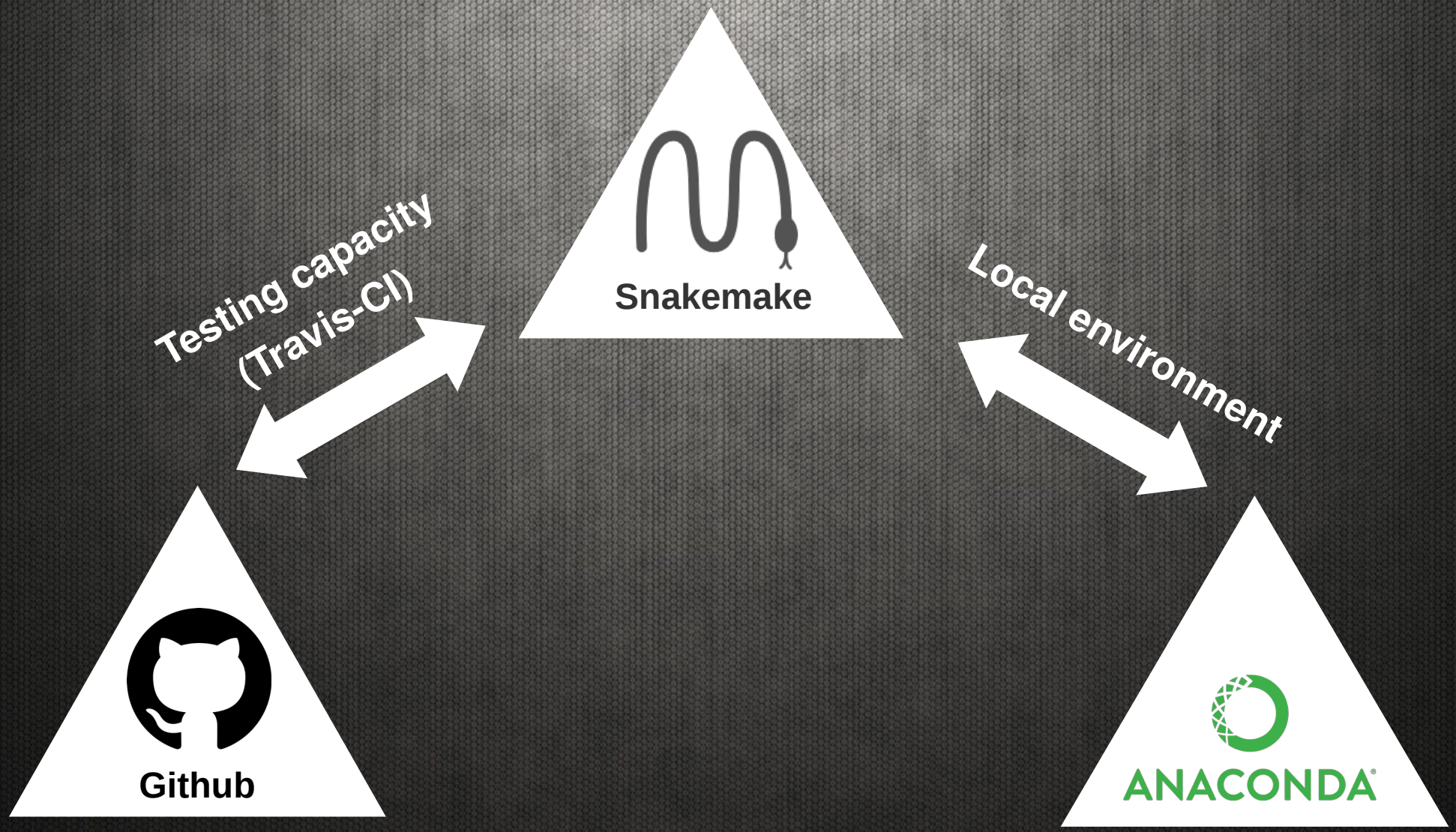
Github

Open source & accessible

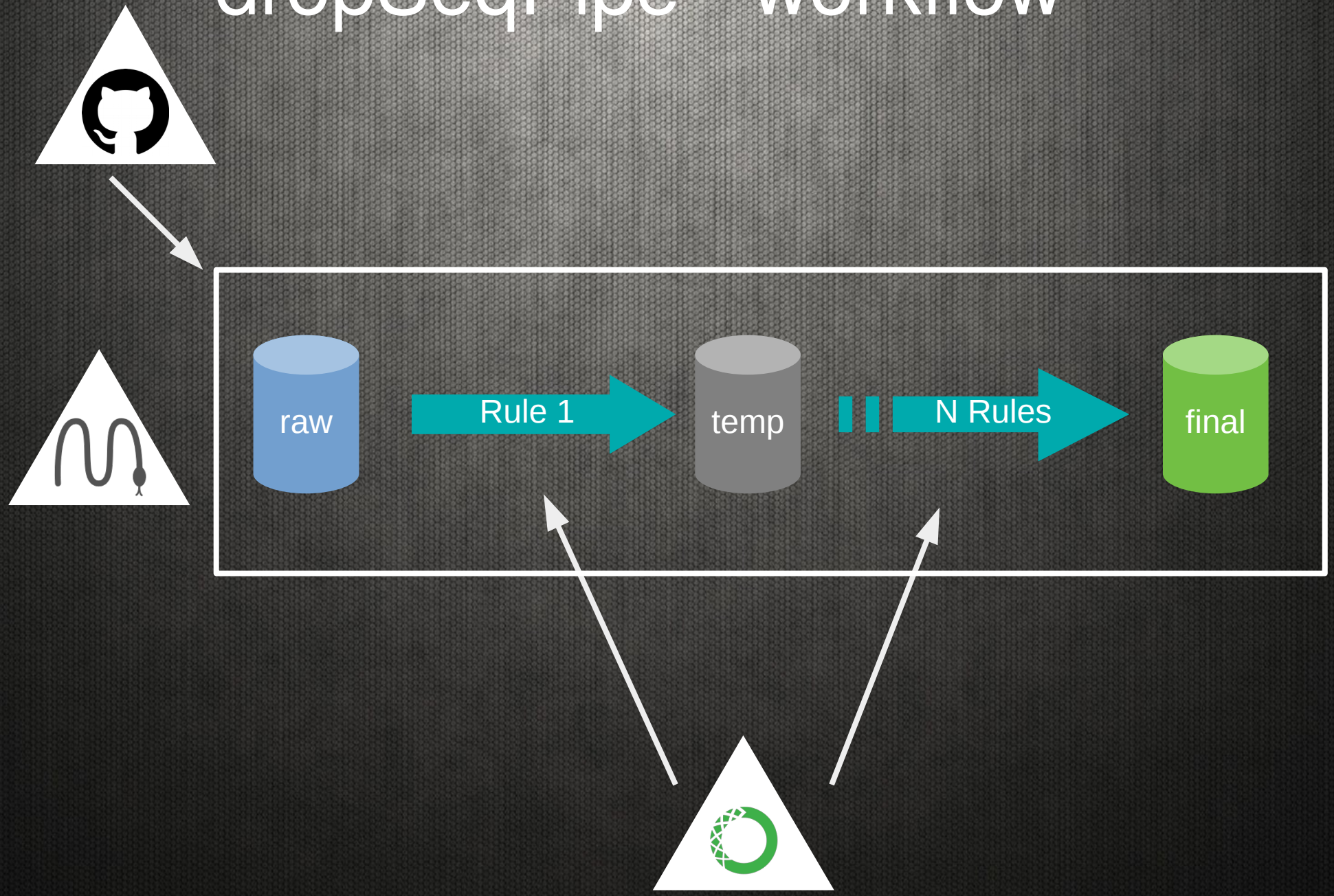
ANACONDA

Packages/softwares version control
Minimise dependencies
No sudo privilege required

dropSeqPipe



dropSeqPipe - workflow



A quick look at a snakemake rule

```
rule plot_yield:
    input:
        BC_tagged=expand('logs/{sample}_CELL_barcode.txt', sample=samples.index),
        UMI_tagged=expand('logs/{sample}_UMI_barcode.txt', sample=samples.index),
        reads_left=expand('logs/{sample}_reads_left.txt', sample=samples.index),
        STAR_output=expand('data/{sample}/Log.final.out', sample=samples.index),
        trimmomatic_filtered=expand('logs/{sample}_reads_left_trim.txt', sample=samples.index)
    params:
        BC_length=config['FILTER']['cell-barcode']['end'] - config['FILTER']['cell-barcode']['start']+1,
        UMI_length=config['FILTER']['UMI-barcode']['end'] - config['FILTER']['UMI-barcode']['start']+1,
        min_num_below_BC=config['FILTER']['cell-barcode']['num-below-quality'],
        min_num_below_UMI=config['FILTER']['UMI-barcode']['num-below-quality'],
        min_BC_quality=config['FILTER']['cell-barcode']['min-quality'],
        min_UMI_quality=config['FILTER']['UMI-barcode']['min-quality'],
        sample_names=lambda wildcards: samples.index,
        batches=lambda wildcards: samples.loc[samples.index, 'batch']
    conda: '../envs/plots.yaml'
    output:
        pdf='plots/yield.pdf'
    script:
        '../scripts/plot_yield.R'
```

Before the run

Download reference and annotation files

https://www.ensembl.org/info/data/ftp/index.html

★	Species	DNA (FASTA)	cDNA (FASTA)	CDS (FASTA)	ncRNA (FASTA)	Protein sequence (FASTA)	Annotated sequence (EMBL)	Annotated sequence (GenBank)	Gene sets		Whole databases	Variation (GVF)	Variation (VCF)	Variation (VEP)	Regulation (GFF)	Data files	BAM/BigWig
Y	Human *Homo sapiens*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF	GFF3	MySQL	GVF	VCF	VEP	Regulation (GFF)	Regulation data files	BAM/BigWig

File: Homo_sapiens.GRCh38.dna.primary_assembly.fa.gz	860562 KB	3/8/18	6:09:00 PM GMT+1
File: Homo_sapiens.GRCh38.92.gtf.gz	41916 KB	3/9/18	11:07:00 AM GMT+1

Run the installation part of the README

Hands-on session

Presented dataset : Pan T cells isolated from mononuclear cells of a healthy donor. T cells are primary cells with relatively small amounts of RNA (~1pg RNA/cell).

Sample dataset : 20000 reads from the Pan T cells

Filling the samples.csv and config.yaml files

dropSeqPipe/templates/config.yaml

```yaml
LOCAL:
    temp-directory:
    dropseq-wrapper:
    memory: 4g
META:
    species:
        - SPECIES_ONE
        - SPECIES_TWO
    ratio: 0.2
    reference-file:
    annotation-file:
    reference-directory:
FILTER:
    5-prime-smart-adapter:
    cell-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
    UMI-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
    trimmomatic:
        adapters-file:
        LEADING: 3
        TRAILING: 3
        SLIDINGWINDOW:
            windowSize: 4
            requiredQuality: 20
        MINLEN: 20
        ILLUMINACLIP:
            seedMismatches: 2
            palindromeClipThreshold: 30
            simpleClipThreshold: 10
MAPPING:
    STAR:
        outFilterMismatchNmax: 10
        outFilterMismatchNoverLmax: 0.3
        outFilterMismatchNoverReadLmax: 1
        outFilterMatchNmin: 0
        outFilterMatchNminOverLread: 0.66
        outFilterScoreMinOverLread: 0.66
EXTRACTION:
    UMI-edit-distance:
    minimum-counts-per-UMI:
```

dropSeqPipe/templates/samples.csv

```csv
samples,expected_cells,read_length,batch
sample1,100,75,Batch1
```

Filling the samples.csv and config.yaml files

dropSeqPipe/templates/config.yaml

```yaml
LOCAL:
    temp-directory:
    dropseq-wrapper:
    memory: 4g
META:
    species:
        - SPECIES_ONE
        - SPECIES_TWO
    ratio: 0.2
    reference-file:
    annotation-file:
    reference-directory:
FILTER:
    5-prime-smart-adapter:
    cell-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
    UMI-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
    trimmomatic:
        adapters-file:
        LEADING: 3
        TRAILING: 3
        SLIDINGWINDOW:
            windowSize: 4
            requiredQuality: 20
        MINLEN: 20
        ILLUMINACLIP:
            seedMismatches: 2
            palindromeClipThreshold: 30
            simpleClipThreshold: 10
MAPPING:
    STAR:
        outFilterMismatchNmax: 10
        outFilterMismatchNoverLmax: 0.3
        outFilterMismatchNoverReadLmax: 1
        outFilterMatchNmin: 0
        outFilterMatchNminOverLread: 0.66
        outFilterScoreMinOverLread: 0.66
EXTRACTION:
    UMI-edit-distance:
    minimum-counts-per-UMI:
```

dropSeqPipe/templates/samples.csv

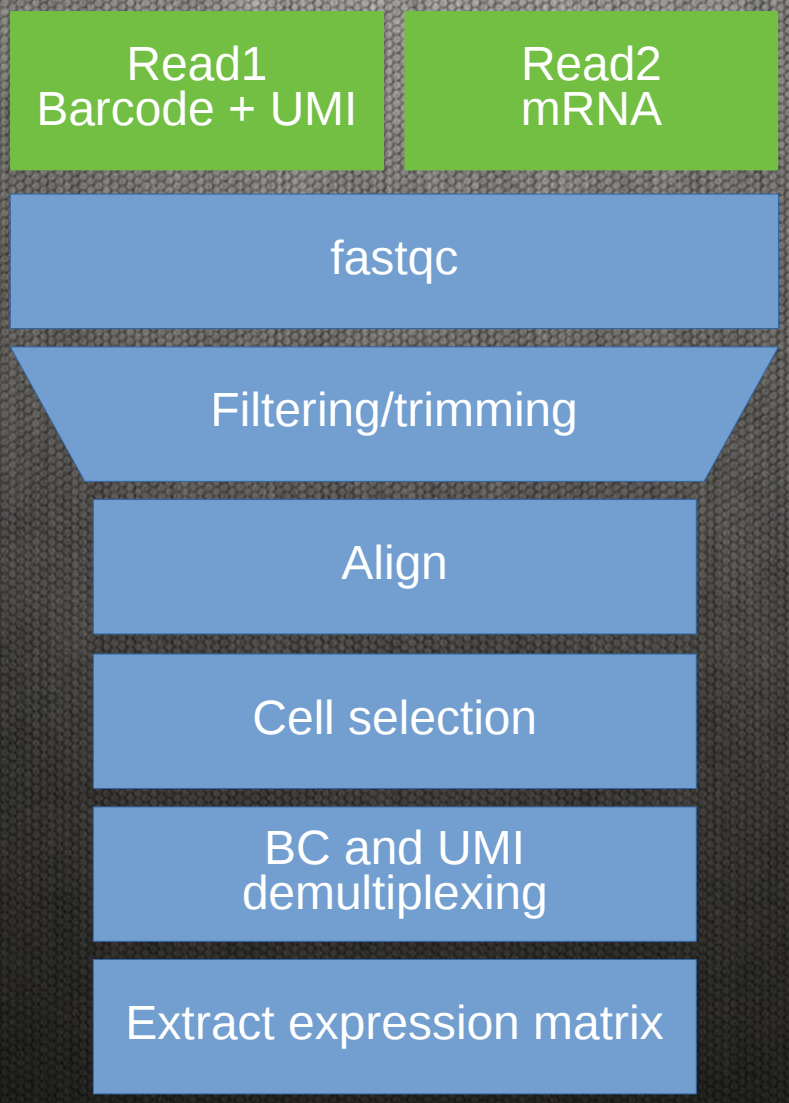
```csv
samples,expected_cells,read_length,batch
sample1,100,75,Batch1
```

```
input:
    R1='data/{sample}_R1.fastq.gz'
    R2='data/{sample}_R2.fastq.gz'
```

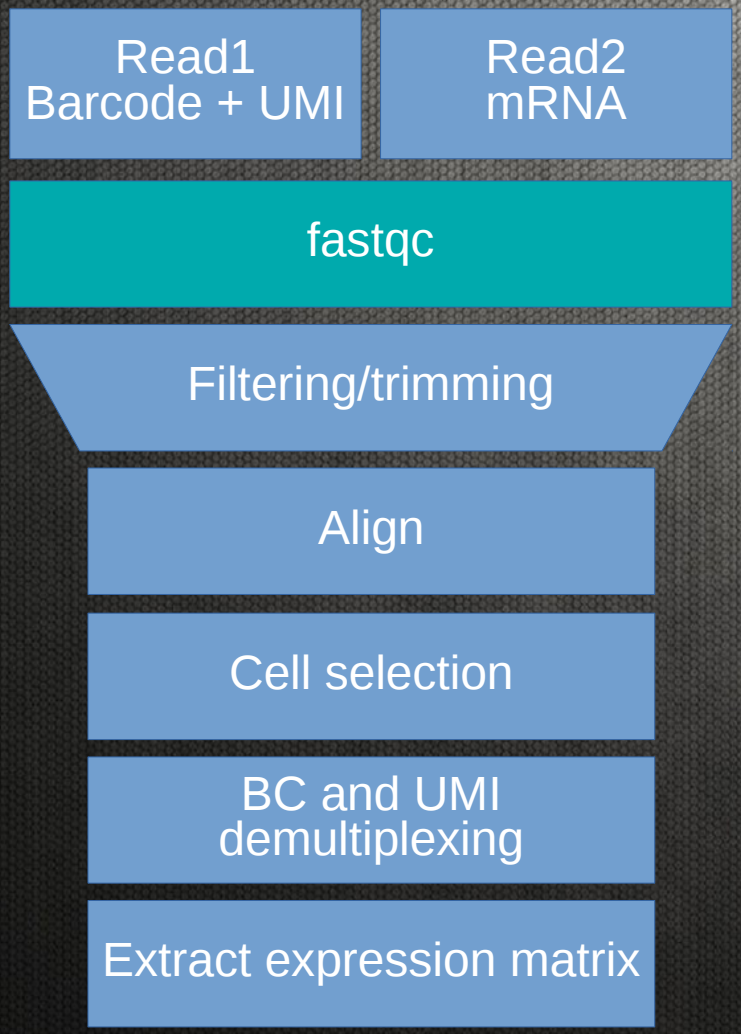
Local and Meta configuration

```yaml
LOCAL:
    temp-directory: /tmp
    dropseq-wrapper: data/Drop-seq_tools-1.13/drop-seq-tools-wrapper.sh
    memory: 4g
META:
    species:
        - SPECIES_ONE
        - SPECIES_TWO
    ratio: 0.2
    reference-file: genome.chr21.fa
    annotation-file: annotation.chr21.gtf
    reference-directory: data/ref
```

Overview of the pipeline



Sequencing quality



Step 1: Sequencing quality control

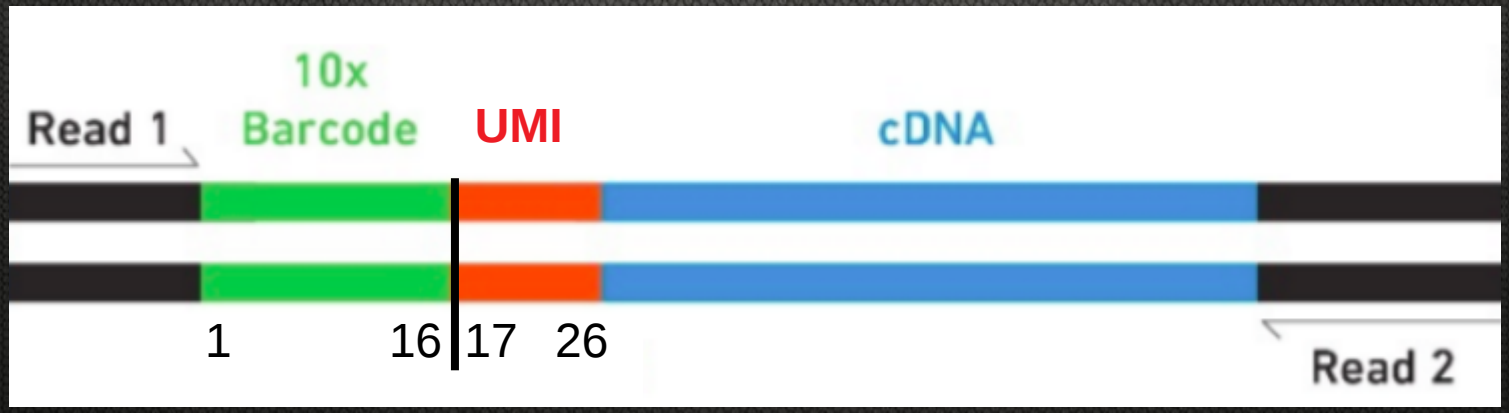
```
snakemake --use-conda --cores 2 --directory sib-days-single-cell qc
```

Barcodes

mRNA

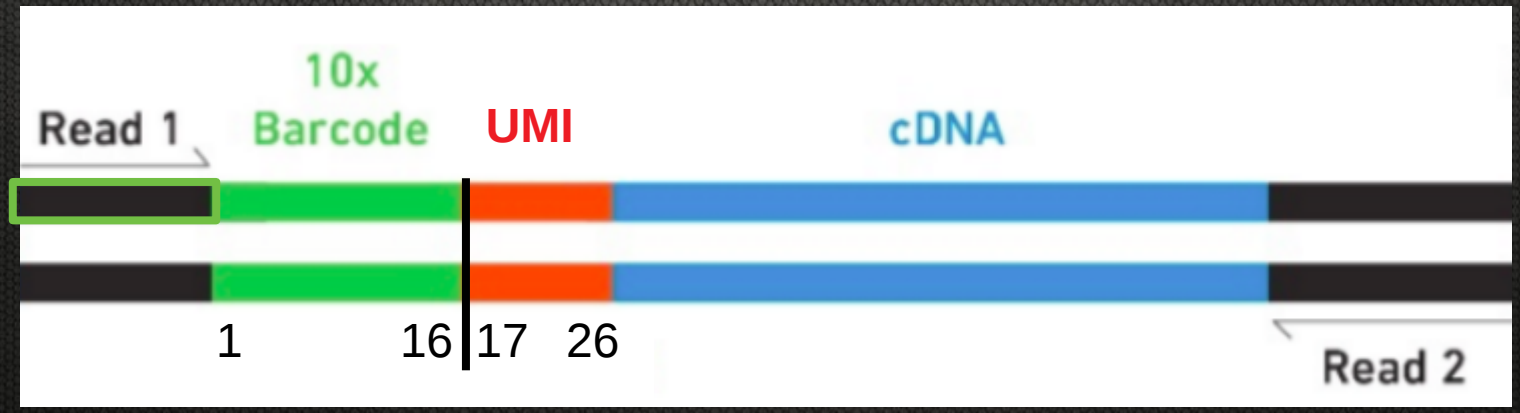
Filter configuration - Barcodes

```yaml
FILTER:
    5-prime-smart-adapter:
    cell-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
    UMI-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
```



Filter configuration - Barcodes

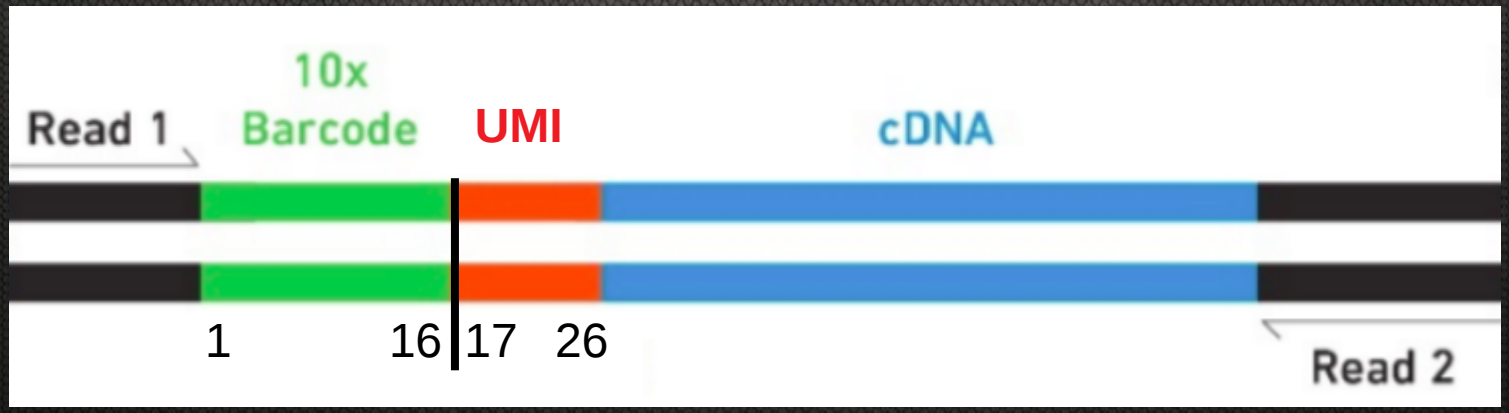
```
FILTER:
    5-prime-smart-adapter:
    cell-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
    UMI-barcode:
        start:
        end:
        min-quality:
        num-below-quality:
```



https://github.com/Teichlab/scg_lib_structs

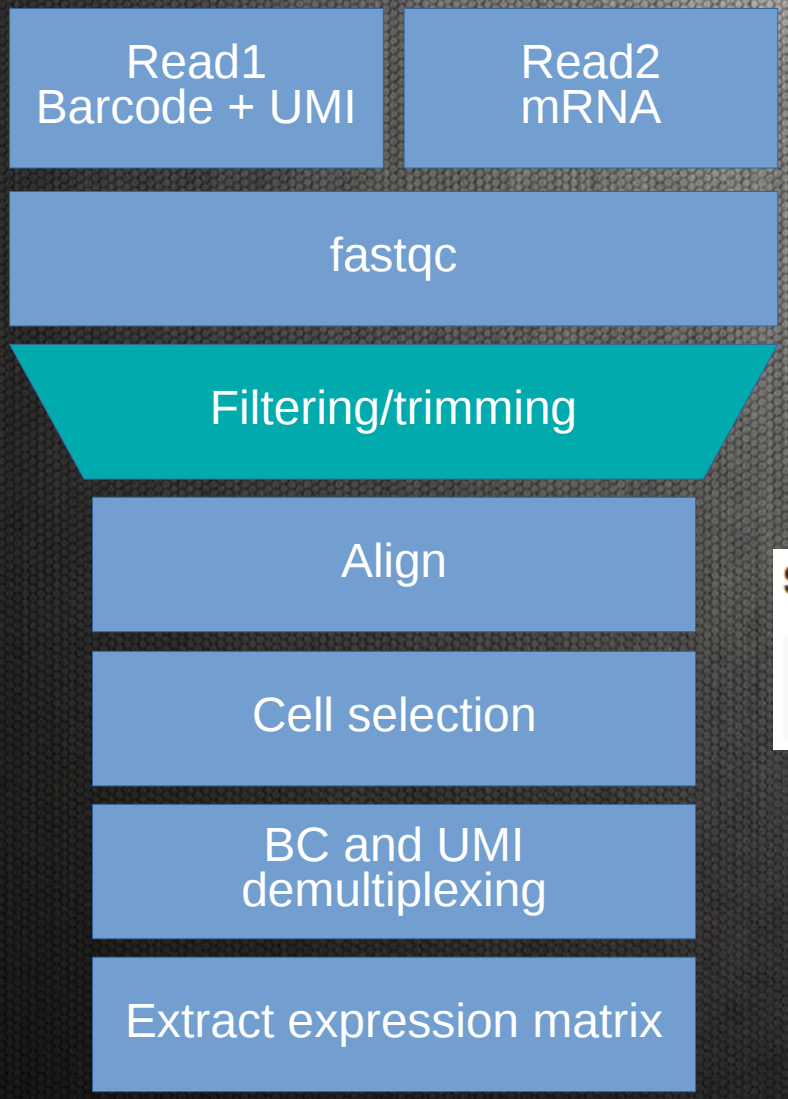
Filter configuration – Second read trimming

```yaml
trimmomatic:
    adapters-file: NexteraPE-PE.fa
    LEADING: 3
    TRAILING: 3
    SLIDINGWINDOW:
        windowSize: 4
        requiredQuality: 20
    MINLEN: 30
    ILLUMINACLIP:
        seedMismatches: 2
        palindromeClipThreshold: 30
        simpleClipThreshold: 10
```



https://github.com/Teichlab/scg_lib_structs

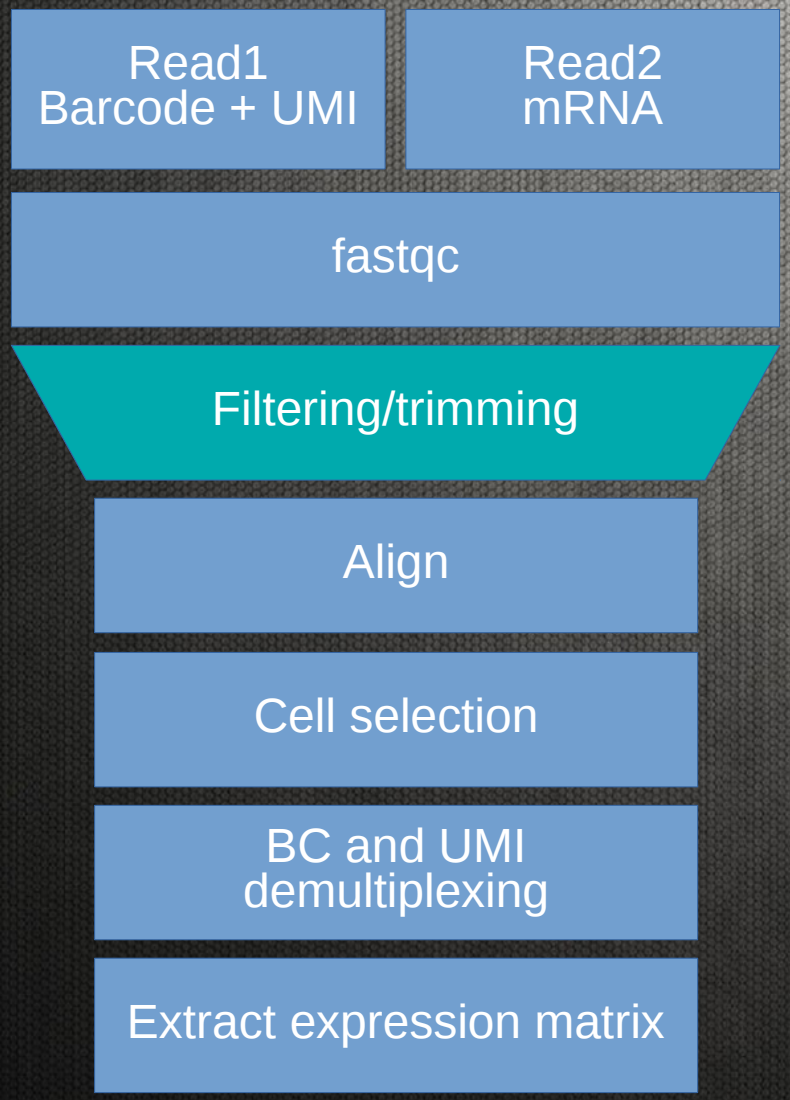
Filtering

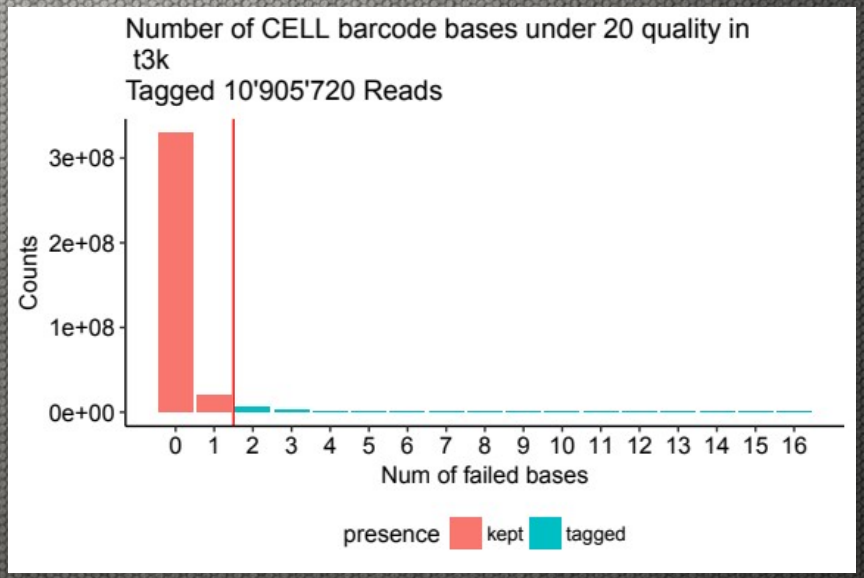


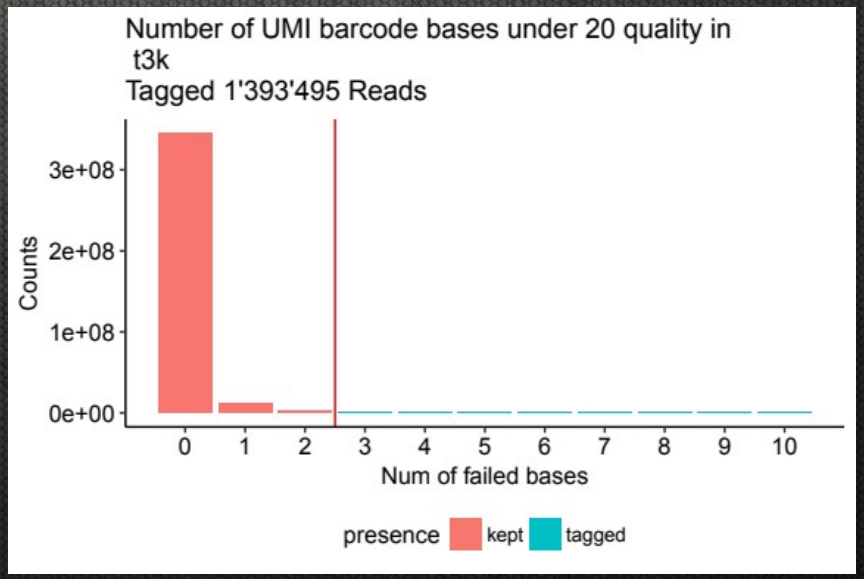
Step 2: Filtering

```
snakemake --use-conda --cores 2 --directory sib-days-single-cell filter
```

Filtering


Read1
Barcode + UMI
Read2
mRNA
fastqc
Filtering/trimming
Align
Cell selection
BC and UMI
demultiplexing
Extract expression matrix


Number of CELL barcode bases under 20 quality in
t3k
Tagged 10'905'720 Reads
Counts
3e+08
2e+08
1e+08
0e+00
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Num of failed bases
presence
kept
tagged


Number of UMI barcode bases under 20 quality in
t3k
Tagged 1'393'495 Reads
Counts
3e+08
2e+08
1e+08
0e+00
0 1 2 3 4 5 6 7 8 9 10
Num of failed bases
presence
kept
tagged

Filtering

Read1
Barcode + UMI

Read2
mRNA

fastqc

Filtering/trimming

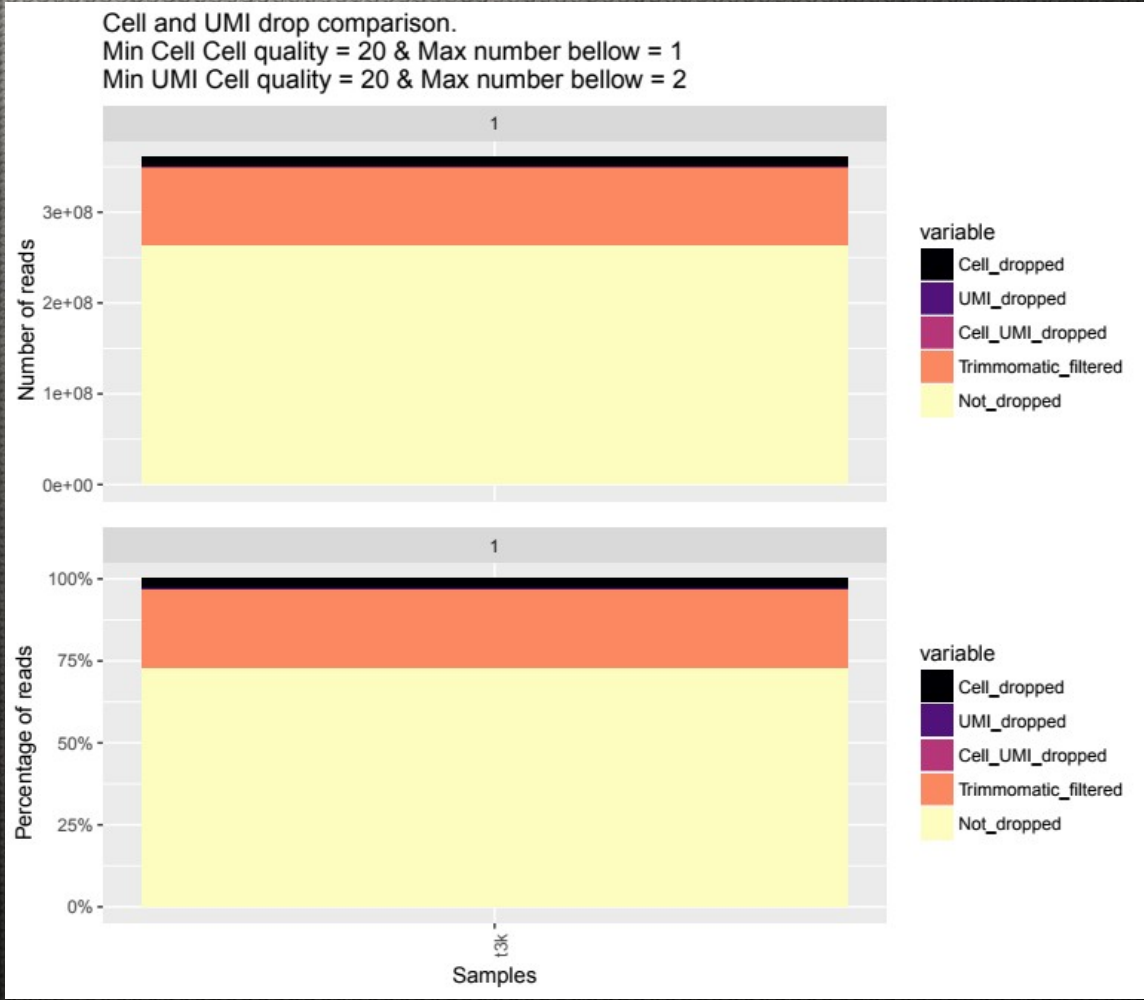
Align

Cell selection

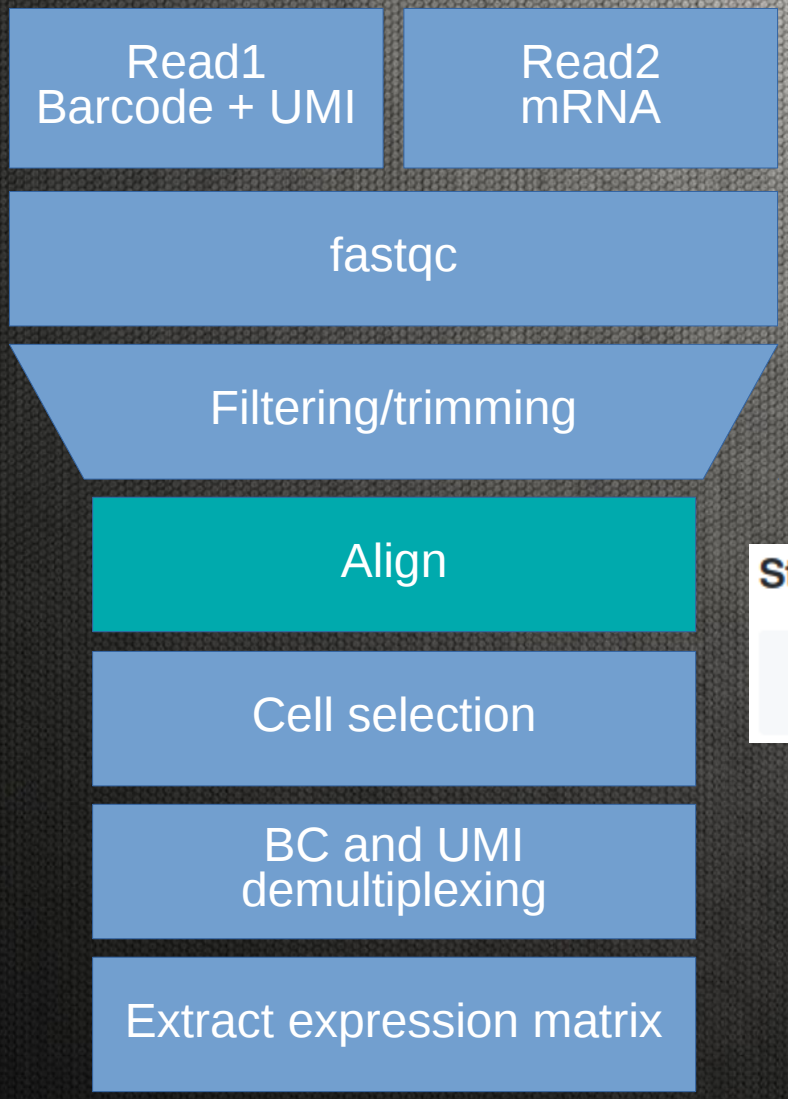
BC and UMI
demultiplexing

Extract expression matrix

trimmomatic



Mapping



Step 3: Mapping

```
snakemake --use-conda --cores 2 --directory sib-days-single-cell map
```

Mapping

Read1
Barcode + UMI

Read2
mRNA

fastqc

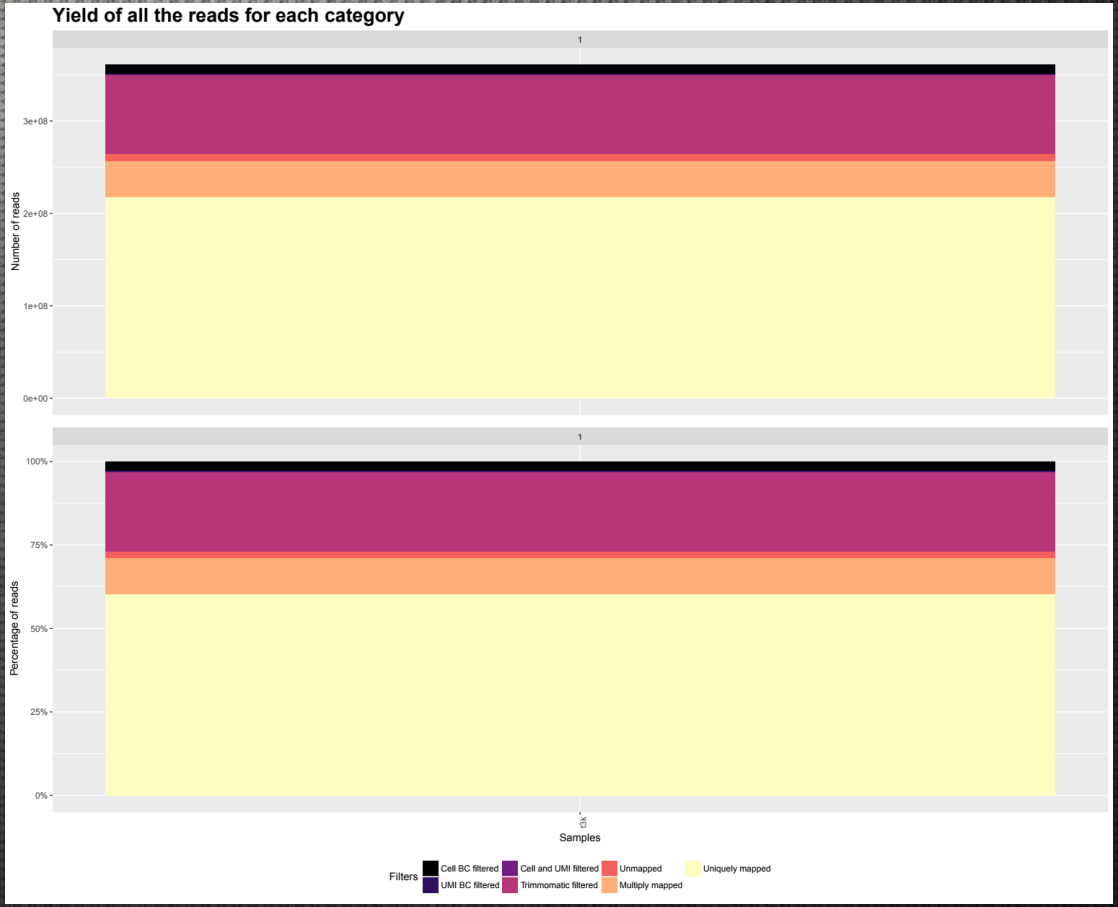
Filtering/trimming

Align

Cell selection

BC and UMI
demultiplexing

Extract expression matrix



STAR REPORT

Mapping

Read1
Barcode + UMI

Read2
mRNA

fastqc

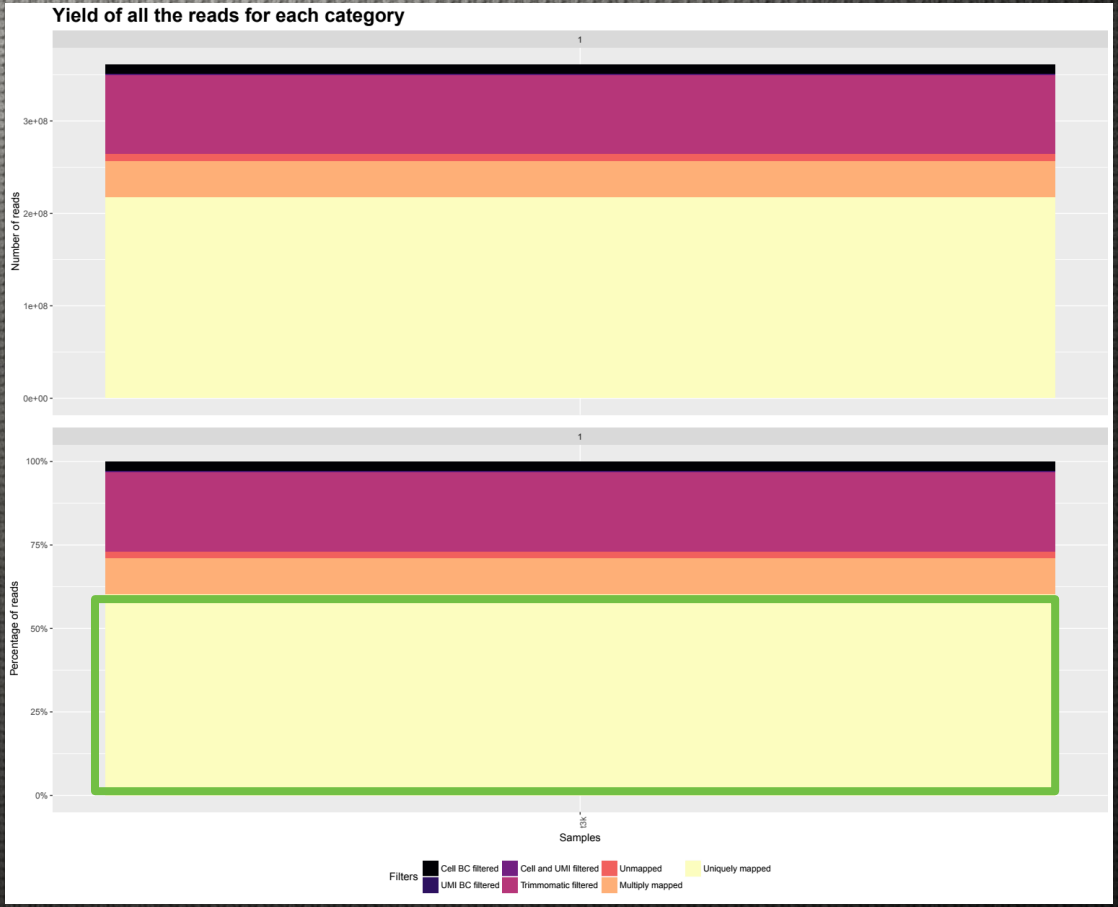
Filtering/trimming

Align

Cell selection

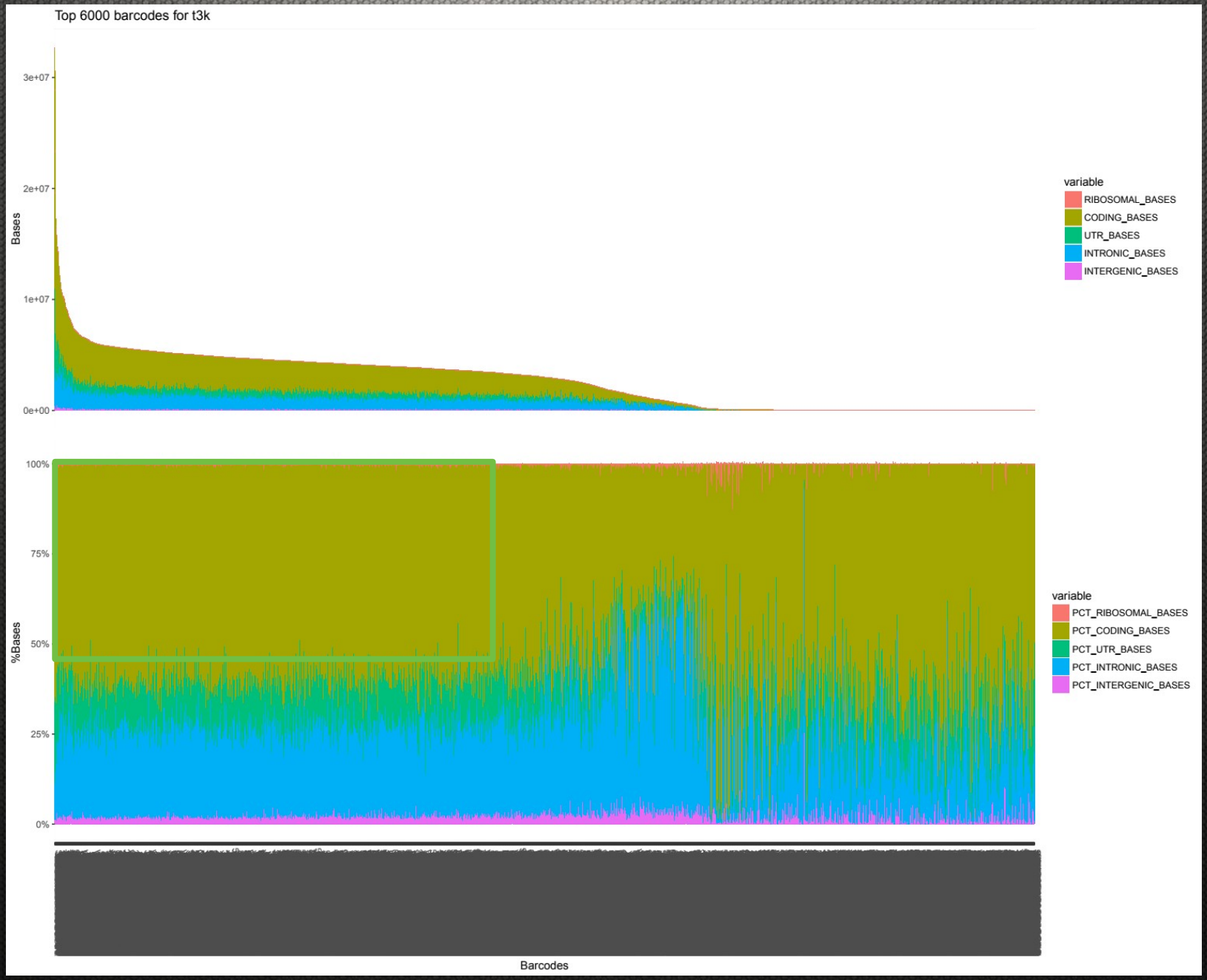
BC and UMI
demultiplexing

Extract expression matrix

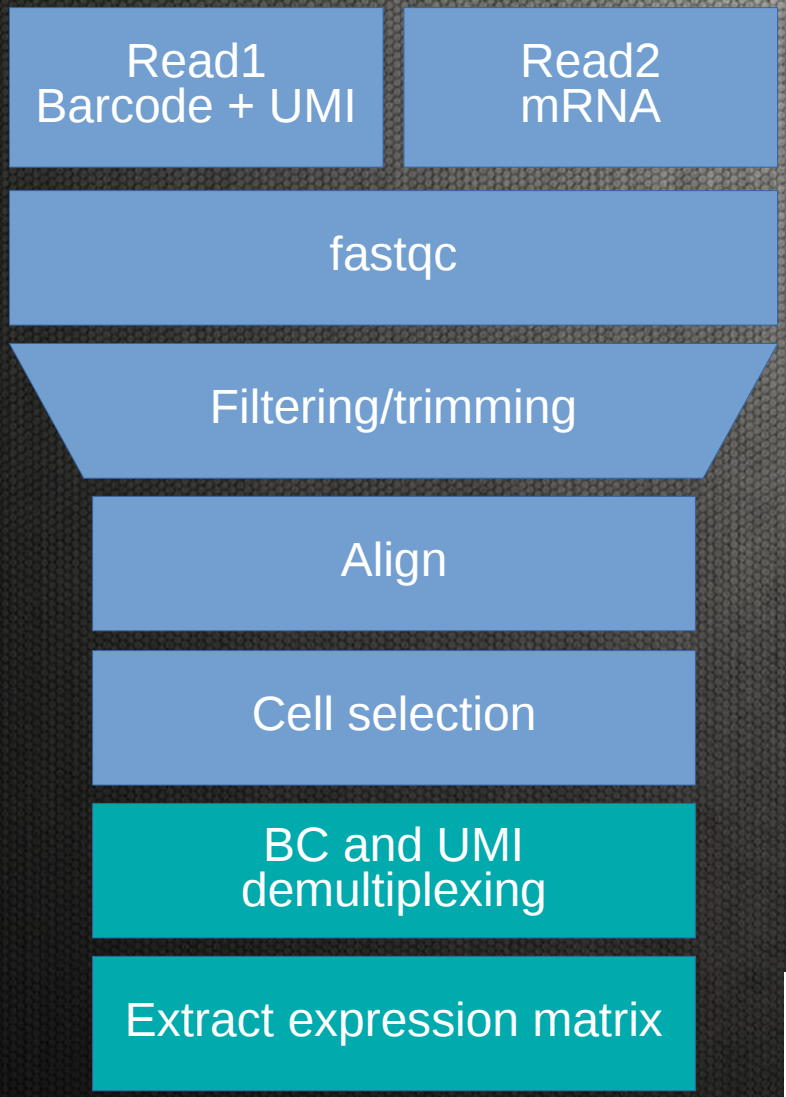


STAR REPORT

Mapping


Top 6000 barcodes for t3k
Bases
3e+07
2e+07
1e+07
0e+00
variable
RIBOSOMAL_BASES
CODING_BASES
UTR_BASES
INTRONIC_BASES
INTERGENIC_BASES
%Bases
100%
75%
50%
25%
0%
variable
PCT_RIBOSOMAL_BASES
PCT_CODING_BASES
PCT_UTR_BASES
PCT_INTRONIC_BASES
PCT_INTERGENIC_BASES
Barcodes

Demultiplexing

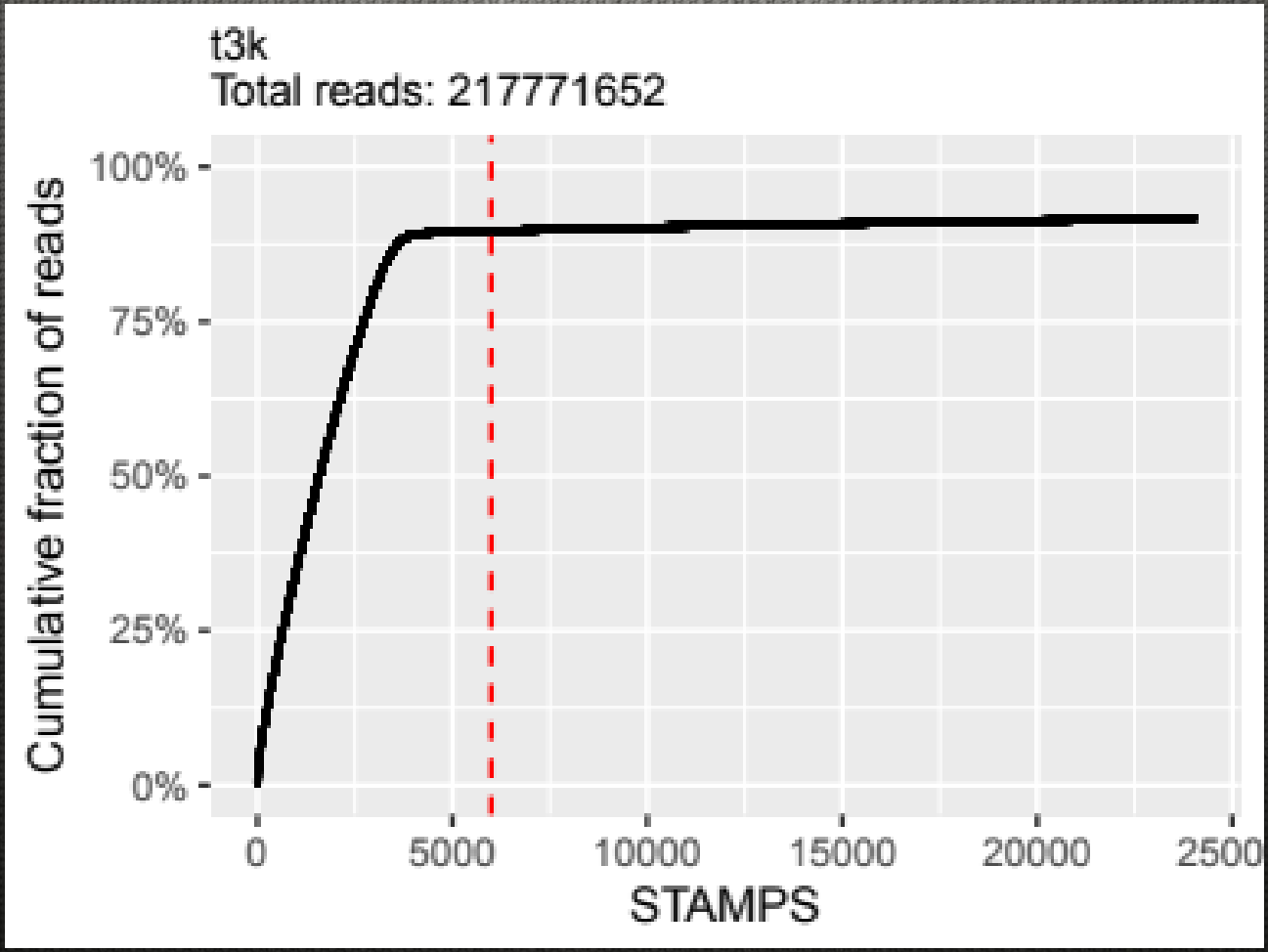



```
EXTRACTION:
    UMI-edit-distance: 1
    minimum-counts-per-UMI: 0
```

Step 4: Demultiplexing and extracting

```
snakemake --use-conda --cores 2 --directory sib-days-single-cell extract
```

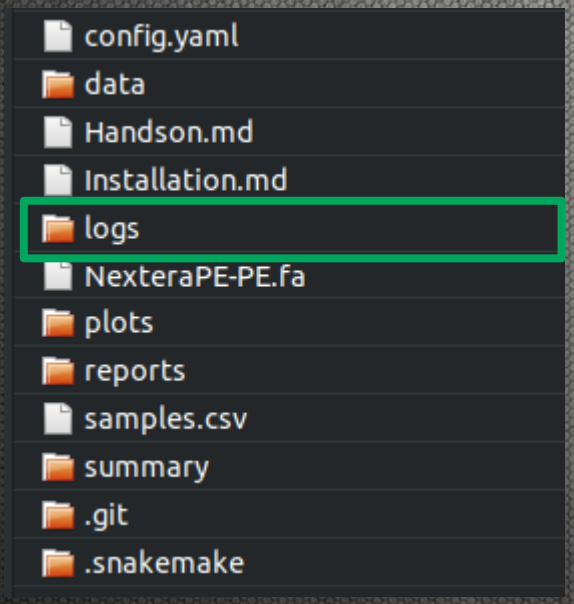
Demultiplexing



Results

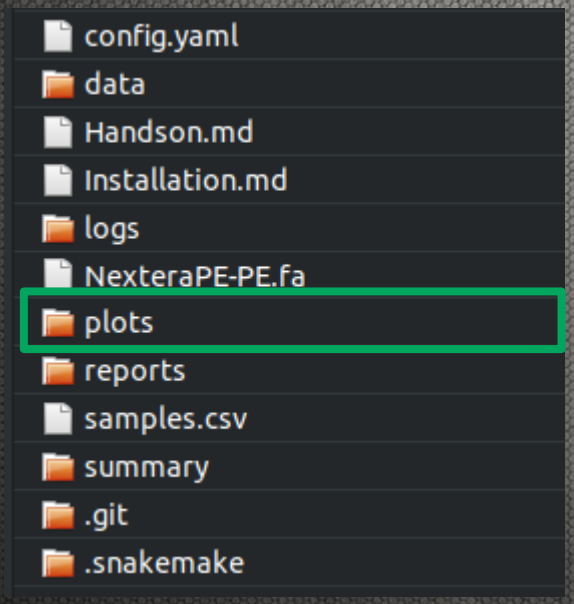
- config.yaml
- data
- Handson.md
- Installation.md
- logs
- NexteraPE-PE.fa
- plots
- reports
- samples.csv
- summary
- .git
- .snakemake

Results



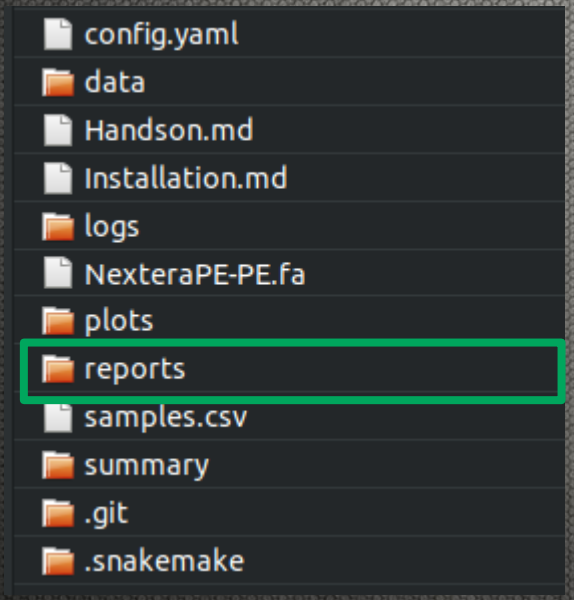
All the logfiles from the processing

Results



All the plots from the processing

Results



All the multiqc reports from the processing

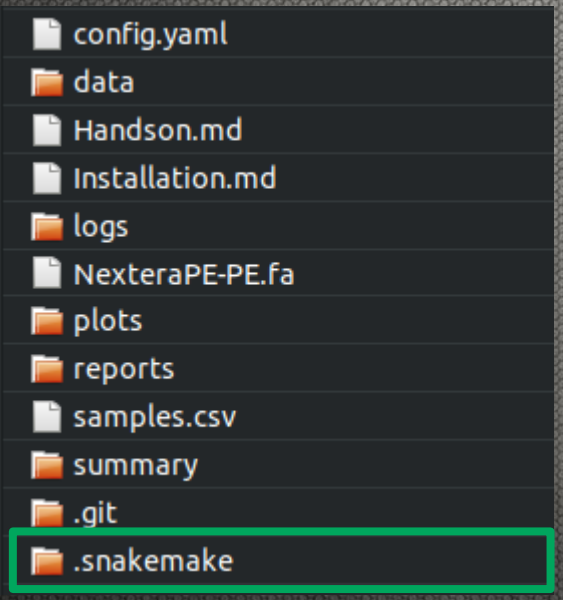
Results

- config.yaml
- data
- Handson.md
- Installation.md
- logs
- NexteraPE-PE.fa
- plots
- reports
- samples.csv
- summary
- .git
- .snakemake

All the summary files from the processing

- counts_expression_matrix.tsv
- t3k_sample_counts_expression_matrix.tsv
- t3k_sample_dge.summary.txt
- t3k_sample_umi_expression_matrix.tsv
- umi_expression_matrix.tsv

Results



All the environments, plus other things

Thank your for your attention