Workshop 2
- Introduction to CellRanger -

Valentin Voillet, Raphael Gottardo

vvoillet@fhcrc.org

FRED HUTCH

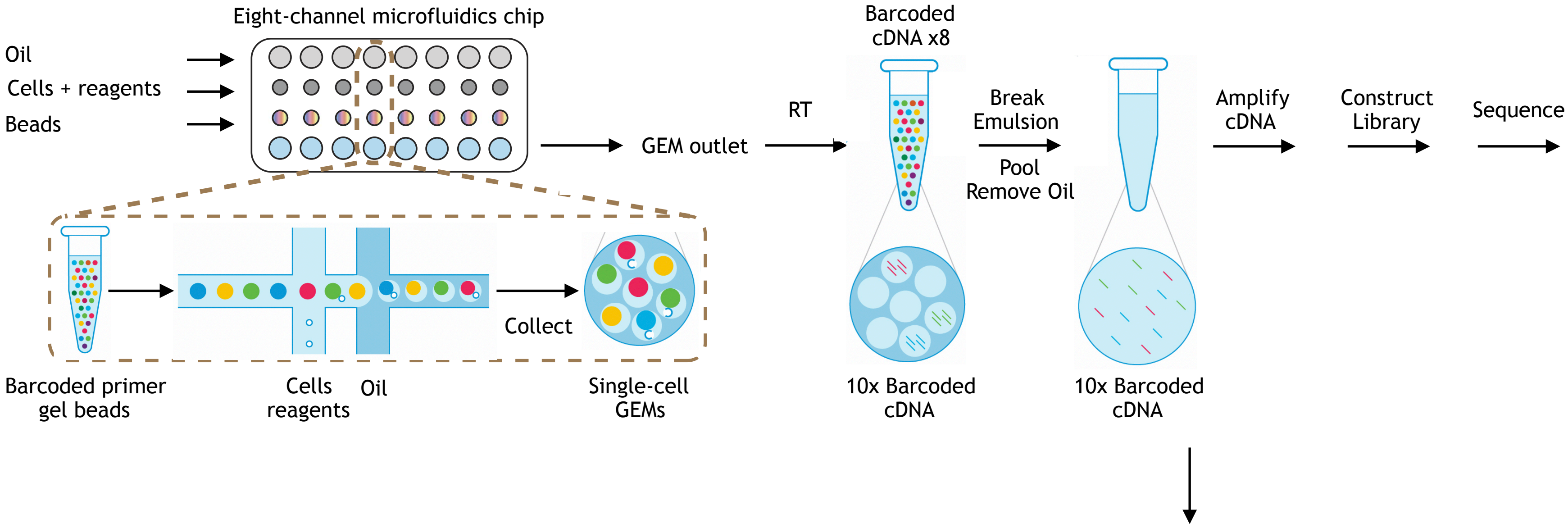
CURES START HERE®

Summary

- 10x Genomics Technology
- CellRanger Pipeline
- Loupe Cell Browser
- Loupe Cell Browser in live!!

10x Genomics technology

Single Cell Digital Gene Expression - 10x Genomics -



Eight-channel microfluidics chip

Oil

Cells + reagents

Beads

GEM outlet

RT

Barcoded cDNA x8

Break Emulsion

Pool Remove Oil

Amplify cDNA

Construct Library

Sequence

Barcoded primer gel beads

Cells reagents

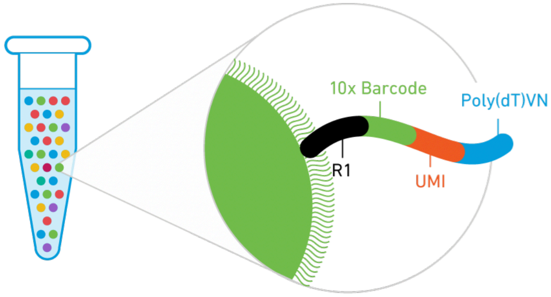
Oil

Collect

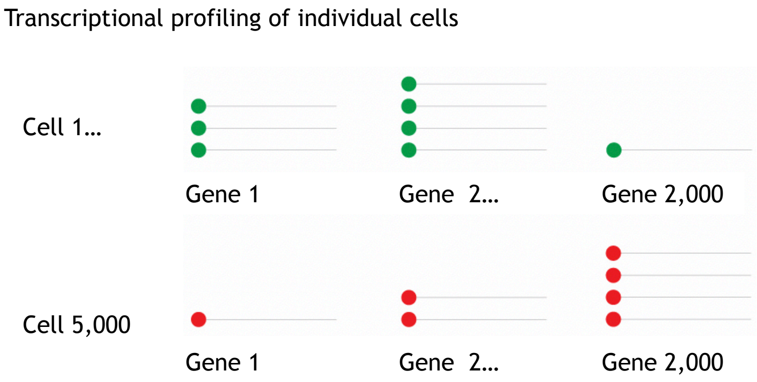
Single-cell GEMs

10x Barcoded cDNA

10x Barcoded cDNA



From 10x Genomics presentation - Workshop, Seattle (Feb, 2017)



Transcriptional profiling of individual cells

Cell 1…

Gene 1

Gene 2…

Gene 2,000

Cell 5,000

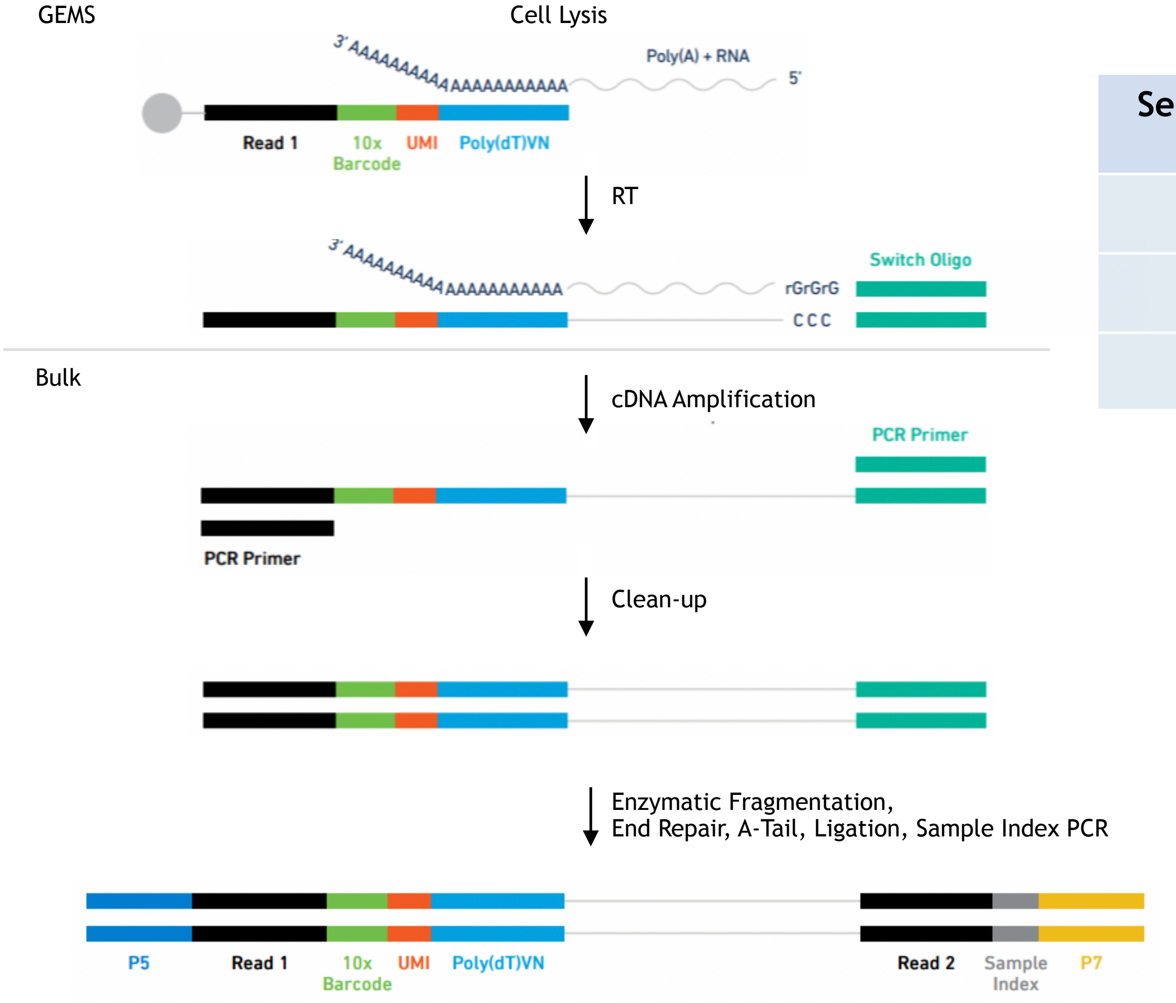
Gene 1

Gene 2…

Gene 2,000

Fig. inspired from Zheng et al., 2017

Single Cell Digital Gene Expression - 10x Genomics -



Sequencing Read	Description	Length
Read 1	Cell Barcode and UMI	26 nt
Read 2	mRNA	~98 nt (recommended)
i7 Index	Sample Index Read	8 nt

↓

Alignment & Quantification

↓

CellRanger™ Pipeline

From 10x Genomics presentation - Workshop, Seattle (Feb, 2017)

CellRanger™ Pipeline

CellRanger Pipeline - Overview -

- cellranger **mkfastq**
  - Barcode demultiplexing from .*bcl* (from Illumina) to .*fastq* files (files used for alignment)

CellRanger Pipeline - Overview -

- cellranger mkfastq
  - Barcode demultiplexing from *.bcl* (from Illumina) to *.fastq* files (files used for alignment)
- cellranger count
  - Read-level analysis of a single library
    - Alignment with STAR
    - Barcode processing
    - Gene counting
  - Produces gene/cell expression matrix
  - Produces some statistical analyses (dimension reduction, clustering and visualization)
  - Produces a *.cloupe* file

CellRanger Pipeline - Overview -

- cellranger **mkfastq**
    - Barcode demultiplexing from *.bcl* (from Illumina) to *.fastq* files (files used for alignment)
- cellranger **count**
    - Read-level analysis of a single library
        - Alignment with STAR
        - Barcode processing
        - Gene counting
    - Produces gene/cell expression matrix
    - Produces some statistical analyses (dimension reduction, clustering and visualization)
    - Produces a *.cloupe* file
- cellranger **aggr**
    - Aggregate *cellranger count* results
    - Combines data from multiple samples and normalizes for sequencing depth
    - Produces combined gene/cell expression matrix
    - Produces same outputs as *cellranger count* (statistical analyses and *.cloupe* file)

CellRanger Pipeline - Overview -

- cellranger **mkfastq**
  - Barcode demultiplexing from *.bcl* (from Illumina) to *.fastq* files (files used for alignment)
- cellranger **count**
  - Read-level analysis of a single library
    - Alignment with STAR
    - Barcode processing
    - Gene counting
  - Produces gene/cell expression matrix
  - Produces some statistical analyses (dimension reduction, clustering and visualization)
  - Produces a *.cloupe* file
- cellranger **aggr**
  - Aggregate *cellranger count* results
  - Combines data from multiple samples and normalizes for sequencing depth
  - Produces combined gene/cell expression matrix
  - Produces same outputs as *cellranger count* (statistical analyses and *.cloupe* file)
- cellranger **reanalyze**
  - Re-run expression/statistical analyses with custom parameters
  - Produces a *.cloupe* file

CellRanger Pipeline - Overview -

- Easy to run
- Give standard components and file formats
- Run on **Linux**

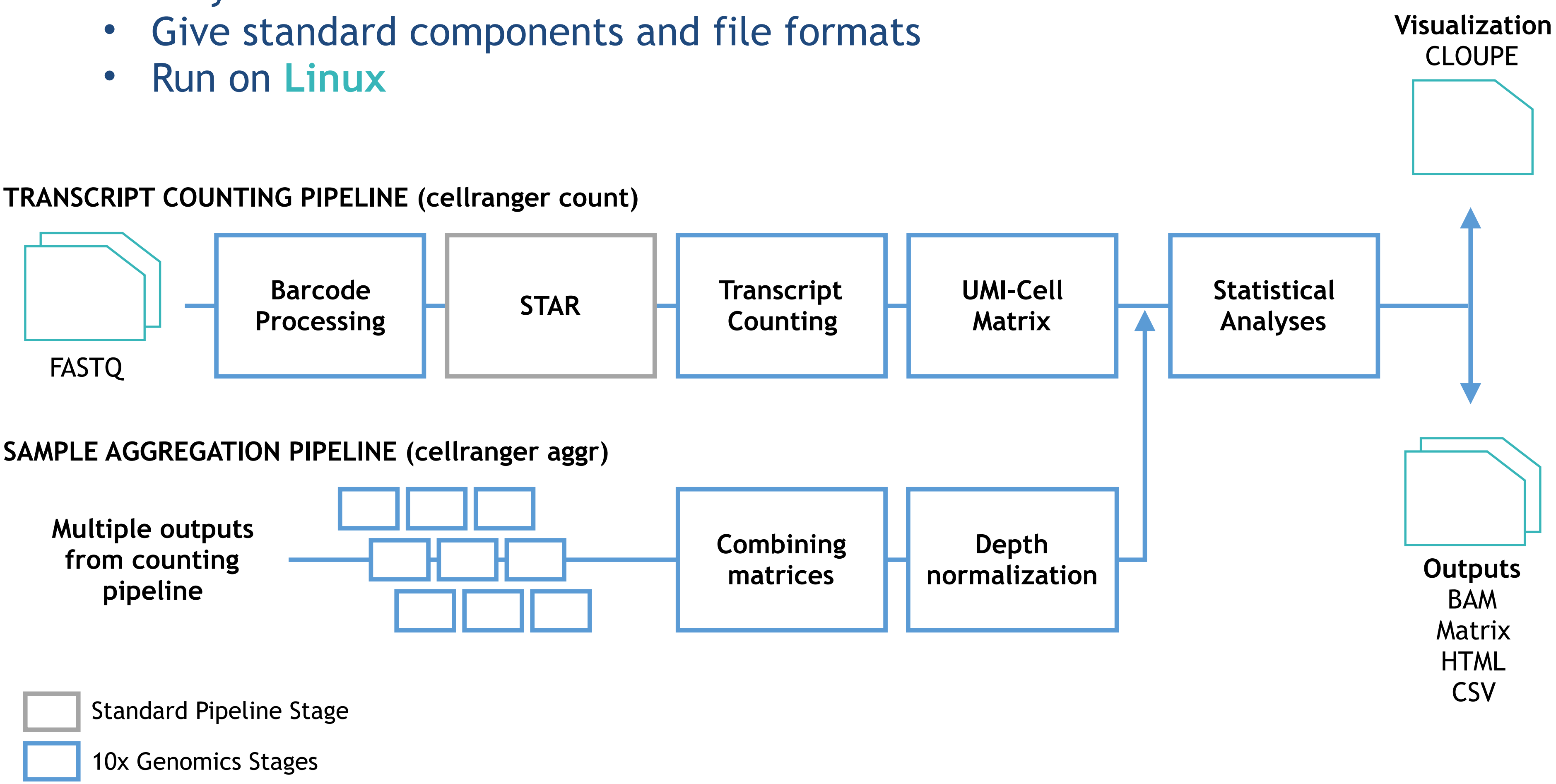


Fig. inspired from 10x Genomics presentation - Workshop, Seattle (Feb, 2017)

System Requirements - Download -

- Cell Ranger pipelines run on Linux systems that meet these minimum requirements:
  - 8-core Intel or AMD processor (16 recommended)
  - 64GB RAM (128GB recommended)
  - 1TB free disk space
  - 64-bit CentOS/RedHat 5.5 or Ubuntu 10.04
- Fred Hutch: Rhino Compute System + Gizmo cluster (contact: helpdesk@fredhutch.org) - see example below

```
vvoillet@rhino1:~$ grablargenode
```

```
and then select how much memory (GB) you would like to grab (here – 128)
```

FRED HUTCH
CURES START HERE®

System Requirements - Download -

- Cell Ranger pipelines run on Linux systems that meet these minimum requirements:
  - 8-core Intel or AMD processor (16 recommended)
  - 64GB RAM (128GB recommended)
  - 1TB free disk space
  - 64-bit CentOS/RedHat 5.5 or Ubuntu 10.04
- Fred Hutch: Rhino Compute System + Gizmo cluster (contact: helpdesk@fredhutch.org) - see example below

```
vvoillet@rhino1:~$ grablargenode
```

and then select how much memory (GB) you would like to grab (here – 128)

- Download CellRanger and 10x reference data from https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest

```
vvoillet@rhino1:~$ tar -xzvf cellranger-2.1.0.tar.gz
vvoillet@rhino1:~$ tar -xzvf refdata-cellranger-1.3.0.tar.gz
vvoillet@rhino1:~$ export PATH=/opt/cellranger-2.1.0:$PATH
```

→ Extraction - CellRanger Pipeline

→ Extraction - 10x Reference Genome

→ Make it executable (not mandatory)

cellranger mkfastq

- Run cellranger mkfastq pipeline to demultiplex an Illumina sequencing run folder into *.fastq* files

```
vvoillet@gizmog5:~$ cellranger mkfastq --run=/path_to_bcl_files/ --samplesheet=mySampleSheet.csv
```

Ex: sampleSheet.csv

Lane	Sample	Index
1	Sample_1	SI-GA-B2
2	Sample_1	SI-GA-B2
1	Sample_2	SI-GA-B3

```
vvoillet@gizmog5:~$ cellranger mkfastq --run=/shared/ngs/illumina/kpaulson/…HKMNCBCXY
                                       --samplesheet=mySampleSheet.csv
```

cellranger mkfastq

- Run cellranger mkfastq pipeline to demultiplex an Illumina sequencing run folder into .*fastq* files

```
vvoillet@gizmog5:~$ cellranger mkfastq --run=/path_to_bcl_files/ --samplesheet=mySampleSheet.csv
```

Ex: sampleSheet.csv

Lane	Sample	Index
1	Sample_1	SI-GA-B2
2	Sample_1	SI-GA-B2
1	Sample_2	SI-GA-B3

```
vvoillet@gizmog5:~$ cellranger mkfastq --run=/shared/ngs/illumina/kpaulson/…HKMNCBCXY
                                       --samplesheet=mySampleSheet.csv
```

- After mkfastq finished, the .*fastq* files are under <flowcell>/outs/fastq_path

```
vvoillet@gizmog5:~$ tree HKMNCBCXY_CellRanger_v2/outs/fastq_path/HKMNCBCXY/
HKMNCBCXY_CellRanger_v2/outs/fastq_path/HKMNCBCXY/
├── GS_PBMC_10_2016_GRN0760
│   ├── GS_PBMC_10_2016_GRN0760_S5_L002_I1_001.fastq.gz
│   ├── GS_PBMC_10_2016_GRN0760_S5_L002_R1_001.fastq.gz
│   └── GS_PBMC_10_2016_GRN0760_S5_L002_R2_001.fastq.gz
├── GS_PBMC_2_2015_GRN0304
│   ├── GS_PBMC_2_2015_GRN0304_S3_L002_I1_001.fastq.gz
│   ├── GS_PBMC_2_2015_GRN0304_S3_L002_R1_001.fastq.gz
│   └── GS_PBMC_2_2015_GRN0304_S3_L002_R2_001.fastq.gz
├── GS_PBMC_2_2016_GRN0535
…
```

→ Sample GS_PBMC_10_2016_GRN0760

→ Sample GS_PBMC_2_2015_GRN0304

cellranger count

- Run cellranger count to produce gene/cell expression matrix
- Using the .*fastq* files produced by cellranger mkfastq

```
vvoillet@gizmog5:~$ cellranger count --id=Sample_1
                                     --transcriptome=/refdata-cellranger/GRCh38
                                     --fastqs=/HKMNCBCXY/outs/fastq_path/Sample_1
                                     --expect-cells=1000
```

Path to your ref genome

Path to your .fastq files

FRED HUTCH
CURES START HERE®

cellranger count

- Run cellranger count to produce gene/cell expression matrix
- Using the *.fastq* files produced by cellranger mkfastq

```
vvoillet@gizmog5:~$ cellranger count --id=Sample_1
                                     --transcriptome=/refdata-cellranger/GRCh38
                                     --fastqs=/HKMNCBCXY/outs/fastq_path/Sample_1
                                     --expect-cells=1000
```

Path to your ref genome

Path to your .fastq files

- A successful run concludes with a message similar to this:

```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

cellranger aggr

- Combines and normalizes (for sequencing depth) libraries previously run with cellranger count

```
vvoillet@gizmo5:~$ cellranger count --id=Sample_1 …
. . . wait for pipeline to finish . . .

vvoillet@gizmo5:~$ cellranger count --id=Sample_2 …
. . . wait for pipeline to finish . . .
```

Ex: Two count pipelines
(Sample_1 and Sample_2) are run

cellranger aggr

- Combines and normalizes (for sequencing depth) libraries previously run with cellranger count

```
vvoillet@gizmo5:~$ cellranger count --id=Sample_1 …
. . . wait for pipeline to finish . . .

vvoillet@gizmo5:~$ cellranger count --id=Sample_2 …
. . . wait for pipeline to finish . . .
```

←

Ex: Two count pipelines
(Sample_1 and Sample_2) are run

- Construct an aggregation *.csv* file pointing to molecule_info.h5 output of cellarage count

Ex: aggregationFile.csv

library_id	molecule_h5
Sample_1	Sample_1/outs/molecule_info.h5
Sample_2	Sample_2/outs/molecule_info.h5

- Then, run cellranger aggr

```
vvoillet@gizmo5:~$ cellranger aggr --id=Aggr_Analysis
                                   --csv=aggregationFile.csv
```

→ ID for combined analysis

→ *.csv* file (ex see above)

- A successful run concludes with a message similar to the one shown before

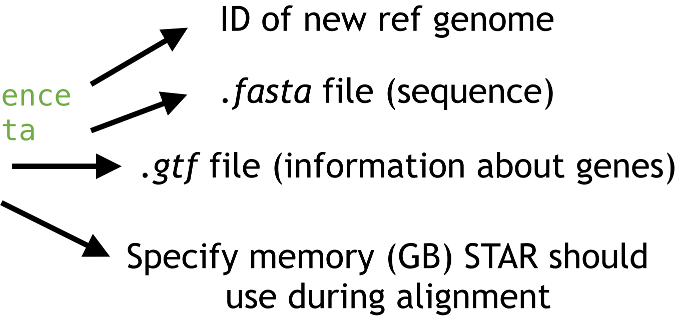
CellRanger Pipeline - References - mkreference

- 10x Website - pre-built references
  - Human (hg19 and GRCh38)
  - Mouse
  - Human and Mouse
  - ERCC

CellRanger Pipeline - References - mkreference

- 10x Website - pre-built references
  - Human (hg19 and GRCh38)
  - Mouse
  - Human and Mouse
  - ERCC
- If your own reference genome (ex: GRCh38 + CAR sequence) - cellranger **mkreference** utility generates a 10x reference package from any *.fasta* and *.gtf* gene file

```
vvoillet@gizmo5:~$ cellranger mkreference --genome=myNewReference
                                          --fasta=myFasta.fasta
                                          --genes=myGTF.gtf
                                          --nthreads 12
```



ID of new ref genome

.fasta file (sequence)

.gtf file (information about genes)

Specify memory (GB) STAR should use during alignment

FRED HUTCH
CURES START HERE®

CellRanger Pipeline - References - mkreference

- 10x Website - pre-built references
  - Human (hg19 and GRCh38)
  - Mouse
  - Human and Mouse
  - ERCC
- If your own reference genome (ex: GRCh38 + CAR sequence) - cellranger **mkreference** utility generates a 10x reference package from any *.fasta* and *.gtf* gene file

```
vvoillet@gizmo5:~$ cellranger mkreference --genome=myNewReference
                                          --fasta=myFasta.fasta
                                          --genes=myGTF.gtf
                                          --nthreads 12
```

ID of new ref genome

.fasta file (sequence)

.gtf file (information about genes)

Specify memory (GB) STAR should use during alignment

- A successful run concludes to with a message similar to this:

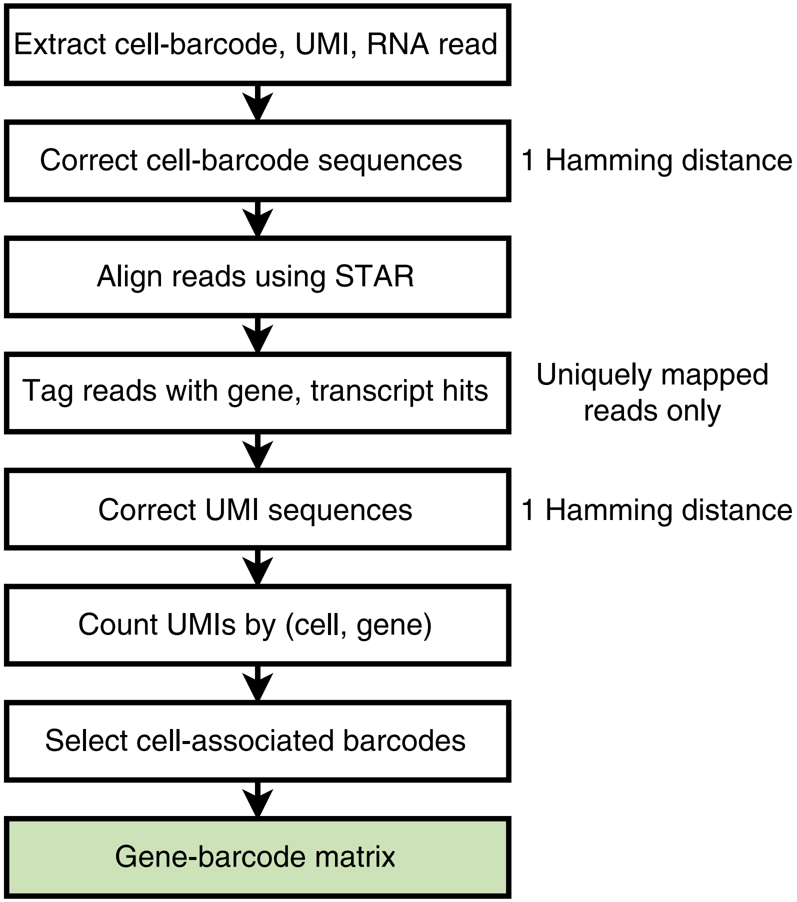
```
>>> Reference successfully created! <<<

You can now specify this reference on the command line:
cellranger –transcriptome=/path_to_your_new_ref_genome ...
```

FRED HUTCH
CURES START HERE®

CellRanger Pipeline - Steps -

- Overview of the pipeline - CellRanger



Extract cell-barcode, UMI, RNA read

Correct cell-barcode sequences — 1 Hamming distance

Align reads using STAR

Tag reads with gene, transcript hits — Uniquely mapped reads only

Correct UMI sequences — 1 Hamming distance

Count UMIs by (cell, gene)

Select cell-associated barcodes

Gene-barcode matrix

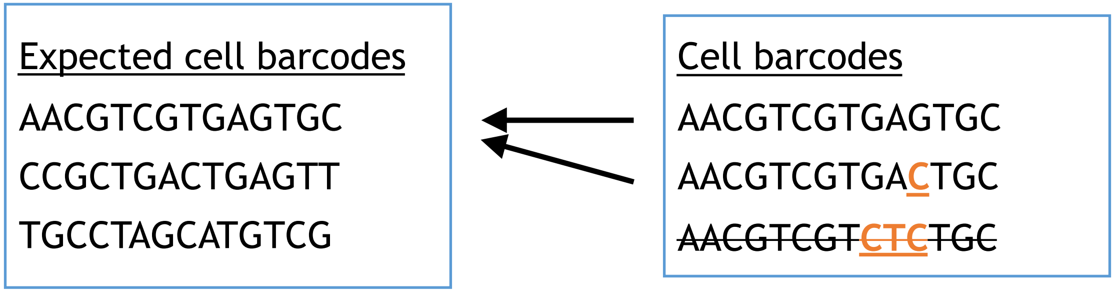
Fig. from Zheng et al., 2017

CellRanger Pipeline - Alignment -

- Alignment performed using STAR (Spliced Transcript Alignment to a Reference)
  - Well-know, open-source, robust
  - Aligns reads to the genome and transcriptome simultaneously
- STAR memory usage
  - cellranger **mkreference** pipeline builds STAR reference such that it uses max 16GB of memory
- Only use confidently mapped reads aligning to transcriptome

CellRanger Pipeline - Cell barcode and UMI Filtering -

- Cell barcodes
  - From a list of known cell barcode sequences
  - May be one mismatch away from the list (if the mismatch occurs at low-quality position (the barcode is then corrected))



Expected cell barcodes

AACGTCGTGAGTGC

CCGCTGACTGAGTT

TGCCTAGCATGTCG

Cell barcodes

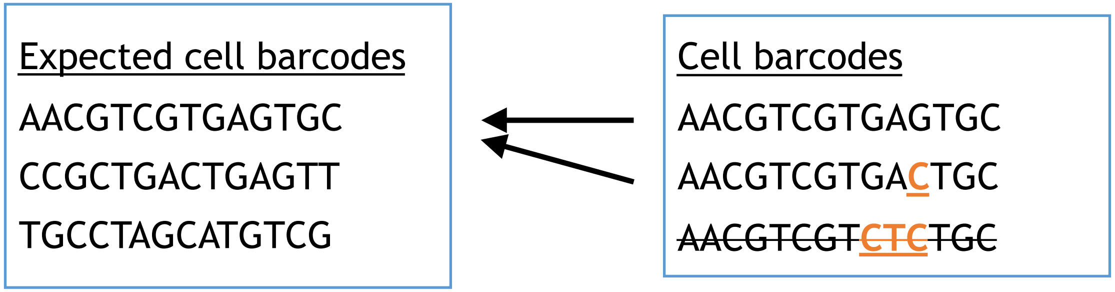
AACGTCGTGAGTGC

AACGTCGTGACTGC

~~AACGTCGTCTCTGC~~

CellRanger Pipeline - Cell barcode and UMI Filtering -

- Cell barcodes
  - From a list of known cell barcode sequences
  - May be one mismatch away from the list (if the mismatch occurs at low-quality position (the barcode is then corrected))



Expected cell barcodes

AACGTCGTGAGTGC

CCGCTGACTGAGTT

TGCCTAGCATGTCG

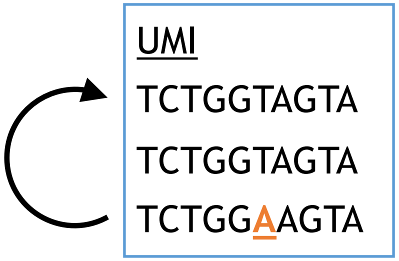
Cell barcodes

AACGTCGTGAGTGC

AACGTCGTGACTGC

~~AACGTCGTCTCTGC~~

- UMIs must
  - not be a homopolymer, e.g. AAAAAAAAAA
  - not contain N
  - not contain bases with base quality < 10
- UMIs with one mismatch away from a higher-count UMI are corrected to that UMI if they share a cell barcode and gene



UMI

TCTGGTAGTA

TCTGGTAGTA

TCTGGAAGTA

CellRanger Pipeline - Marking Duplicates -

- Using only confidently mapped reads with valid cell barcodes and UMIs
  - Correct the UMIs
  - Record which reads are duplicates of the same RNA molecule
  - Count only unique UMIs as unique RNA molecules
  - The UMI counts give us an **unfiltered gene/cell expression matrix**

CellRanger Pipeline - Filtering Cells -

- Select GEMs (Gel beads in EMulsion) that likely contain cells
  - Sum UMI counts for each barcode
  - Select barcodes with total UMI count >= 10% of the 99th percentile of the expected recovered cells
- Gives us a **filtered gene/cell expression matrix**

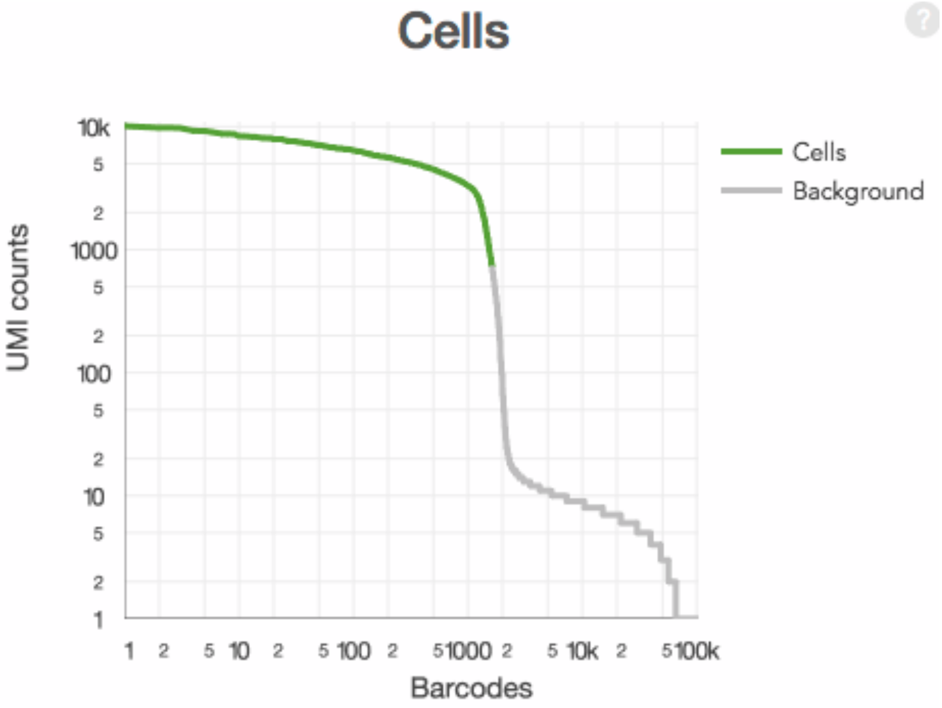


Fig. from 10x Genomics presentation - Workshop, Seattle (Feb, 2017)

CellRanger Pipeline - Outputs -

- A successful run concludes with a message similar to this:

```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

CellRanger Pipeline - Outputs -

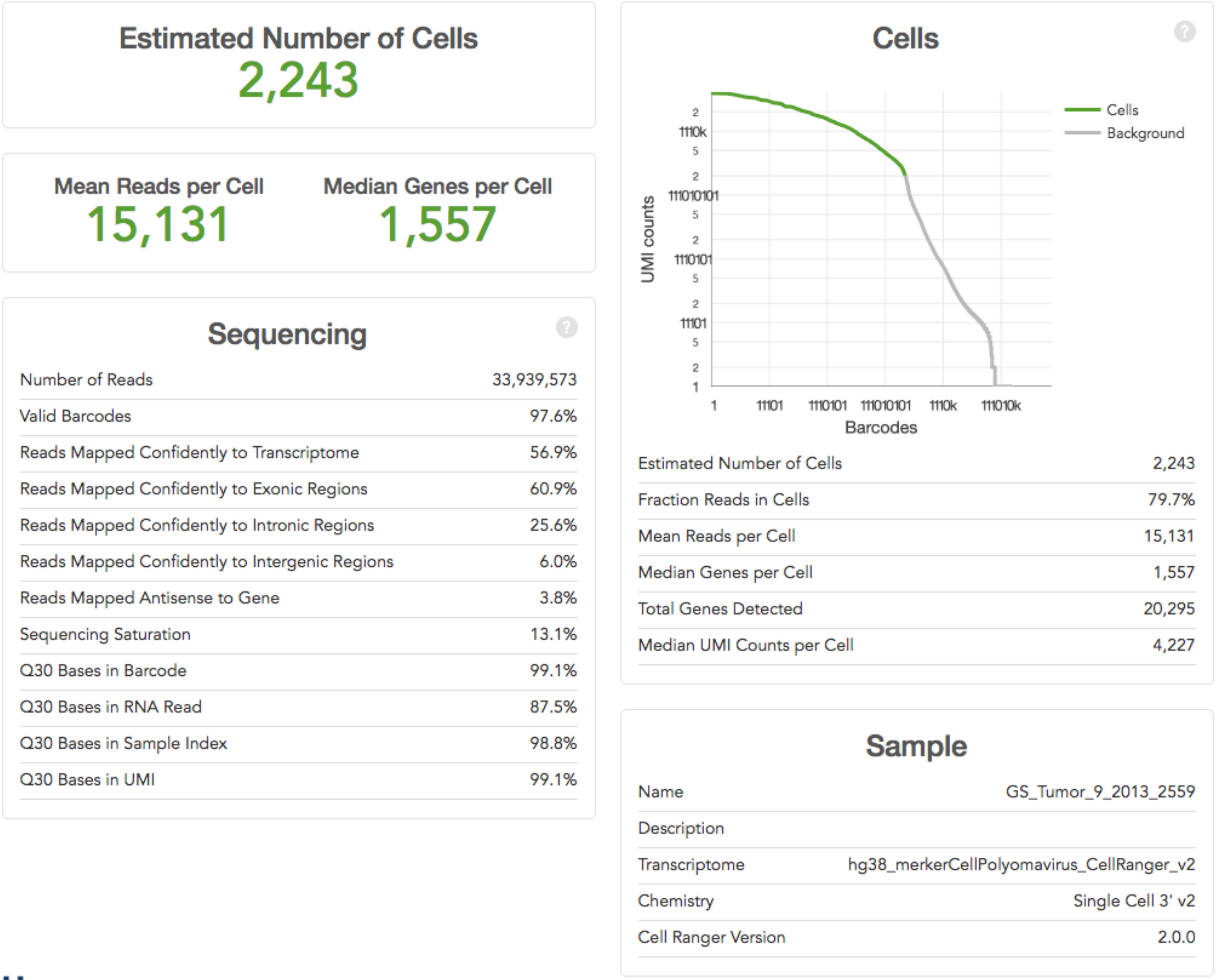
- A successful run concludes with a message similar to this:

```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

CellRanger Pipeline - Outputs -

MCC project
Tumor sample - before therapy



Estimated Number of Cells
2,243

Mean Reads per Cell
15,131

Median Genes per Cell
1,557

Sequencing

Number of Reads	33,939,573
Valid Barcodes	97.6%
Reads Mapped Confidently to Transcriptome	56.9%
Reads Mapped Confidently to Exonic Regions	60.9%
Reads Mapped Confidently to Intronic Regions	25.6%
Reads Mapped Confidently to Intergenic Regions	6.0%
Reads Mapped Antisense to Gene	3.8%
Sequencing Saturation	13.1%
Q30 Bases in Barcode	99.1%
Q30 Bases in RNA Read	87.5%
Q30 Bases in Sample Index	98.8%
Q30 Bases in UMI	99.1%

Cells

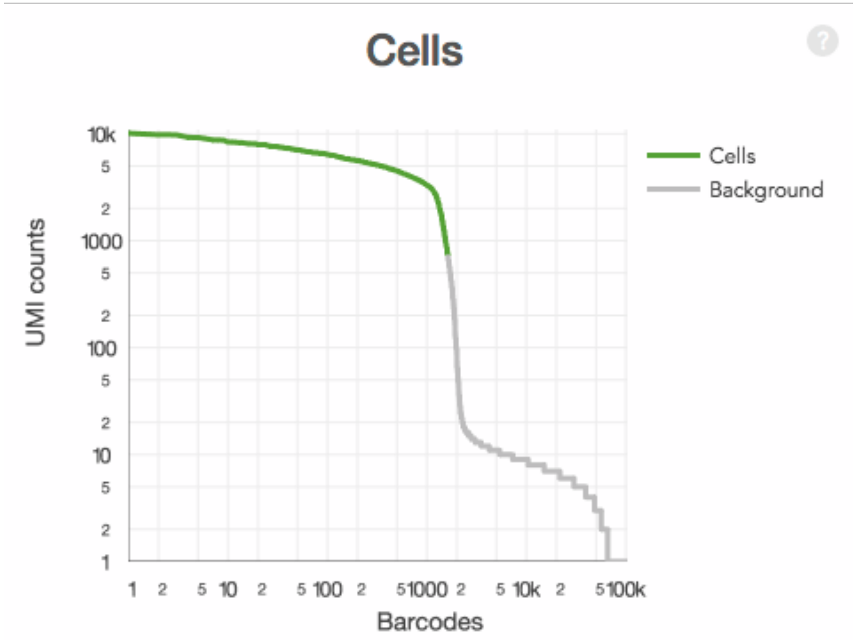
Estimated Number of Cells	2,243
Fraction Reads in Cells	79.7%
Mean Reads per Cell	15,131
Median Genes per Cell	1,557
Total Genes Detected	20,295
Median UMI Counts per Cell	4,227

Sample

Name	GS_Tumor_9_2013_2559
Description	
Transcriptome	hg38_merkerCellPolyomavirus_CellRanger_v2
Chemistry	Single Cell 3' v2
Cell Ranger Version	2.0.0

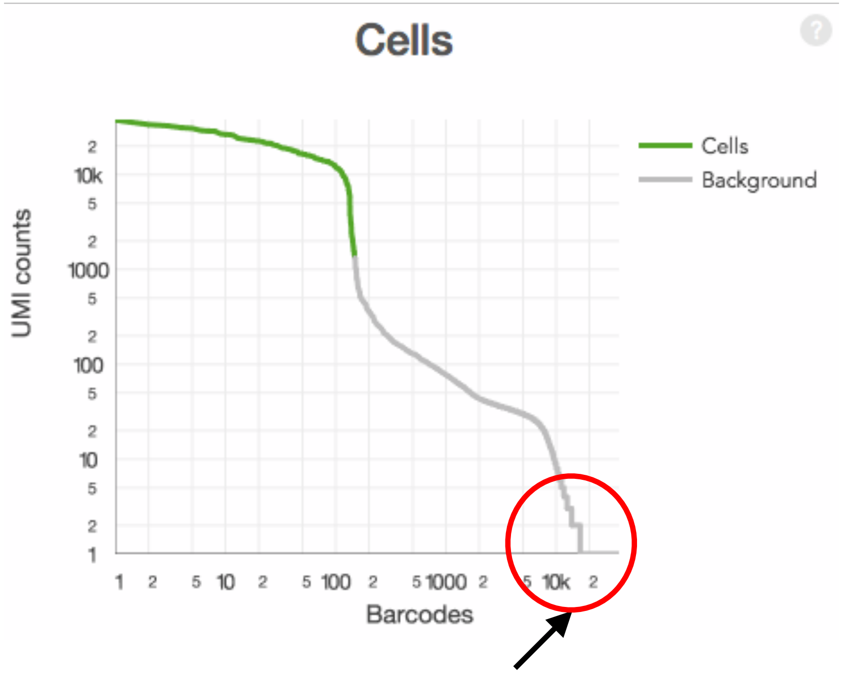
FRED HUTCH
CURES START HERE®

CellRanger Pipeline - Outputs -



Typical sample profile:

- >1,000 cell barcodes detected
- >10,000 UMI detected

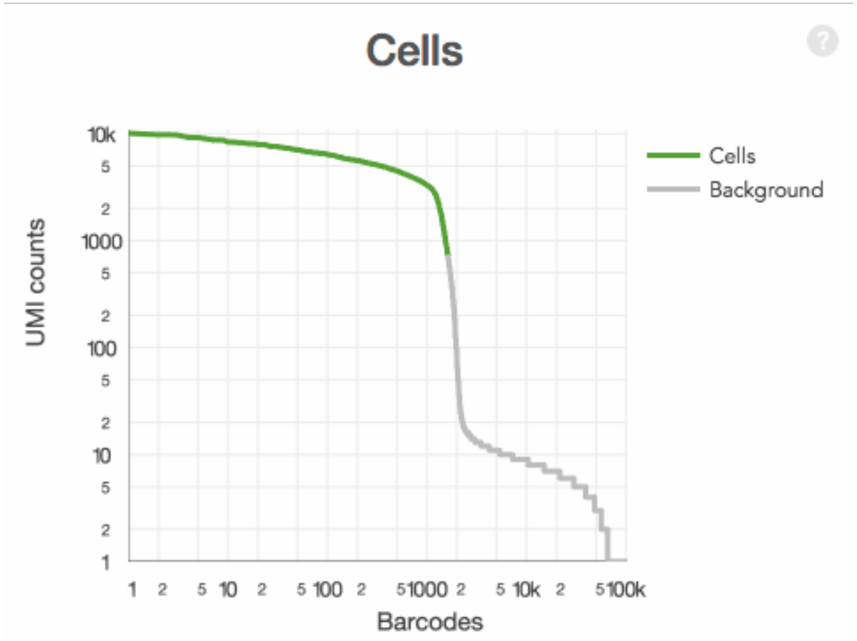


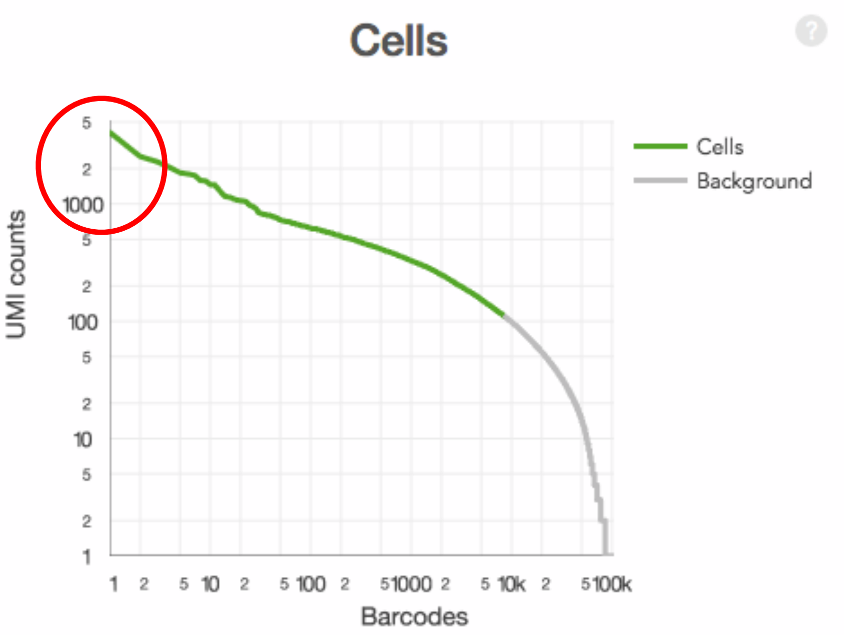
Low number of barcodes detected

Low barcode counts (low depth..)

Fig. from 10x Genomics presentation - Workshop, Seattle (Feb, 2017)

CellRanger Pipeline - Outputs -





Typical sample profile:

- >1,000 cell barcodes detected
- >10,000 UMI detected

Loss of single cell behavior (e.g. lysis failure or wetting failure)

- No cliff and knee
- High number of cell barcodes
- Algorithm has trouble discerning cells from the background

Fig. from 10x Genomics presentation - Workshop, Seattle (Feb, 2017)

CellRanger Pipeline - Outputs -

- Reads confidently mapped to transcriptome (<30%)
  - Reads mapped to wrong genome
  - Too short of read length
  - Custom reference may contain overlapping genes
- RNA Q30 metrics are low (<70%)
  - Sequencing issues

CellRanger Pipeline - Outputs -

- A successful run concludes with a message similar to this:

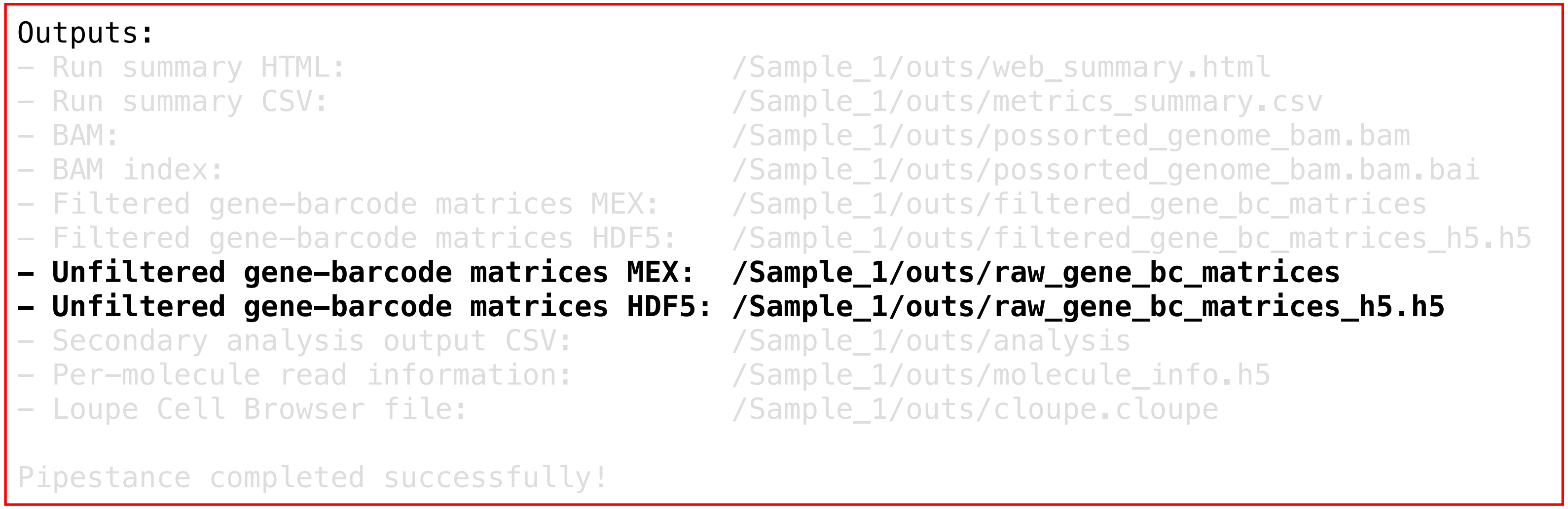
```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

Same information as in previous slide - .*csv* file

CellRanger Pipeline - Outputs -

- A successful run concludes with a message similar to this:



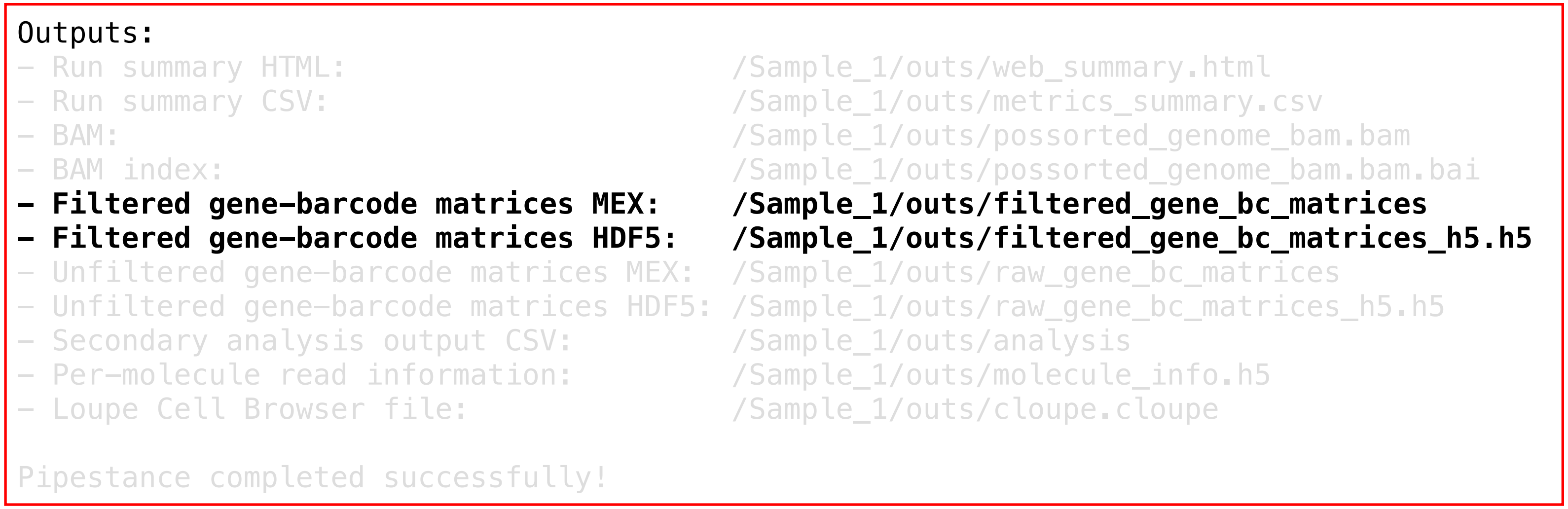
```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

Unfiltered gene-cell matrix - can be directly opened in R using ***cellrangerRkit*** R package

CellRanger Pipeline - Outputs -

- A successful run concludes with a message similar to this:



```
Outputs:
- Run summary HTML:                      /Sample_1/outs/web_summary.html
- Run summary CSV:                       /Sample_1/outs/metrics_summary.csv
- BAM:                                   /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                             /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:    /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:   /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:  /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5: /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:         /Sample_1/outs/analysis
- Per-molecule read information:         /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:               /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

Filtered gene-cell matrix - can be directly opened in R using ***cellrangerRkit*** R package

CellRanger Pipeline - Outputs -

- A successful run concludes with a message similar to this:

```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

Statistical analyses performed by cellranger

- Dimension reduction (PCA, tSNE)
- Clustering (k-means & graph-based clustering)
- Differential Expression Analysis (between clusters)

! Black box - parameters?

FRED HUTCH
CURES START HERE®

CellRanger Pipeline - Outputs -

- A successful run concludes with a message similar to this:

```
Outputs:
- Run summary HTML:                         /Sample_1/outs/web_summary.html
- Run summary CSV:                          /Sample_1/outs/metrics_summary.csv
- BAM:                                      /Sample_1/outs/possorted_genome_bam.bam
- BAM index:                                /Sample_1/outs/possorted_genome_bam.bam.bai
- Filtered gene-barcode matrices MEX:       /Sample_1/outs/filtered_gene_bc_matrices
- Filtered gene-barcode matrices HDF5:      /Sample_1/outs/filtered_gene_bc_matrices_h5.h5
- Unfiltered gene-barcode matrices MEX:     /Sample_1/outs/raw_gene_bc_matrices
- Unfiltered gene-barcode matrices HDF5:    /Sample_1/outs/raw_gene_bc_matrices_h5.h5
- Secondary analysis output CSV:            /Sample_1/outs/analysis
- Per-molecule read information:            /Sample_1/outs/molecule_info.h5
- Loupe Cell Browser file:                  /Sample_1/outs/cloupe.cloupe

Pipestance completed successfully!
```

.*cloupe* file - can be directly opened using Loupe Cell Browser

CellRanger Pipeline - Conclusion -

- CellRanger Pipeline - easy to run
- It is long to run (especially if many samples to analyze and aggregate)
- System requirements: **Linux**, min 64GB RAM - Run it on Gizmo
- Outputs easy to understand and analyze - .*cloupe* file

Loupe™ Cell Browser

Loupe Cell Browser - Overview -

- Loupe Cell Browser is a desktop application for Windows and MacOS that allows you to quickly and easily visualize and analyze 10x Chromium Single-Cell 5' and 3' gene expression data
- It opens .*cloupe* file produced by CellRanger and
  - Explore single-cell data interactively, no programming needed
  - Find significant genes from different clusters and conditions
  - Identify functional subgroups through gene expression tools
  - Create and save custom clusters, list of genes, etc.
  - Export screenshots and data for publication and further analysis
- Useful for exploratory analysis before going deeper in the analysis

⚠ Black box - What parameters it uses? etc..

System Requirements - Download -

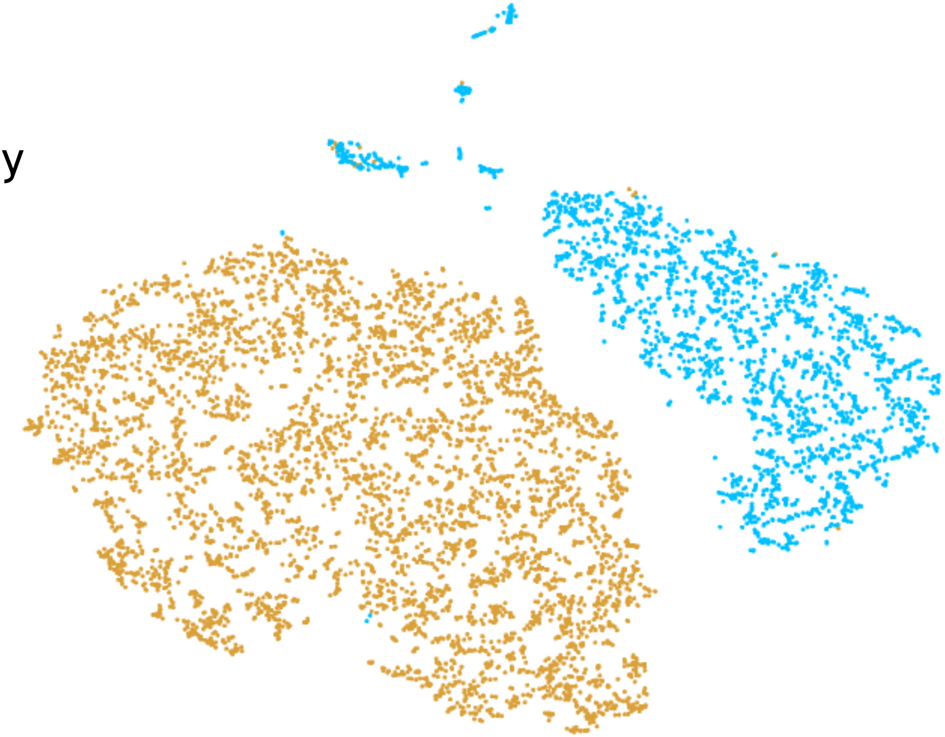
- Windows or Mac
  - 4GB RAM
  - SSD storage highly recommended
- Download directly from https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest
- Software comes with a tutorial!
- Public CellRanger 1.3 datasets include .*cloupe* files

Loupe Cell Browser - Visualization -

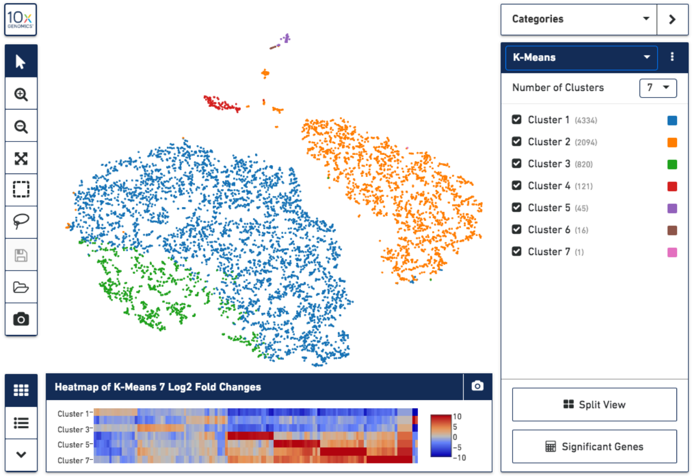
- Principal Component Analysis (PCA) is performed to reduce dimensionality of the data - from about 35,000 genes to 10 Principal Components (PCs)
- t-SNE (t-Distributed Stochastic Neighbor Embedding) is then used to visualize PCA results (# cells vs 10 PCs) in two dimensions
- Each dot is a cell

MCC project

Tumor samples -
before & after therapy



Loupe Cell Browser - Visualization -



Categories

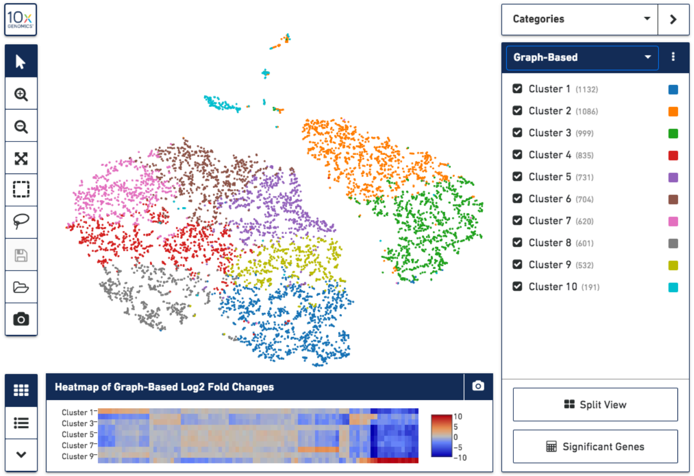
K-Means

Number of Clusters: 7

- ☑ Cluster 1 (4334) 🟦
- ☑ Cluster 2 (2094) 🟧
- ☑ Cluster 3 (820) 🟩
- ☑ Cluster 4 (121) 🟥
- ☑ Cluster 5 (45) 🟪
- ☑ Cluster 6 (16) 🟫
- ☑ Cluster 7 (1) 🟪

Heatmap of K-Means 7 Log2 Fold Changes

K-means



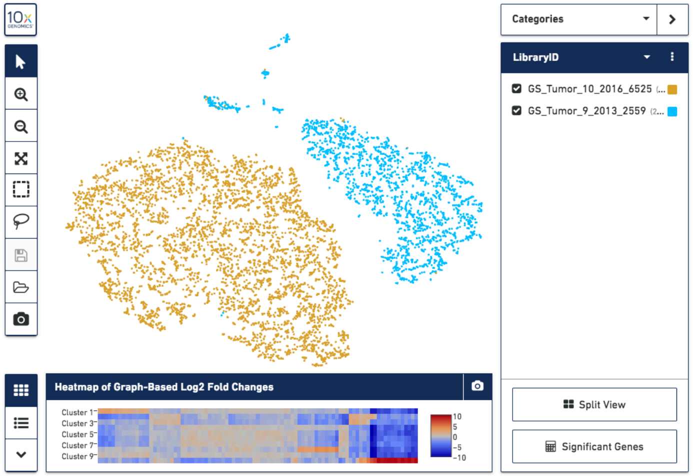
Categories

Graph-Based

- ☑ Cluster 1 (1132) 🟦
- ☑ Cluster 2 (1086) 🟧
- ☑ Cluster 3 (999) 🟩
- ☑ Cluster 4 (835) 🟥
- ☑ Cluster 5 (731) 🟪
- ☑ Cluster 6 (704) 🟫
- ☑ Cluster 7 (620) 🟪
- ☑ Cluster 8 (601) ⬛
- ☑ Cluster 9 (532) 🟨
- ☑ Cluster 10 (191) 🟦

Heatmap of Graph-Based Log2 Fold Changes

Graph-based clustering



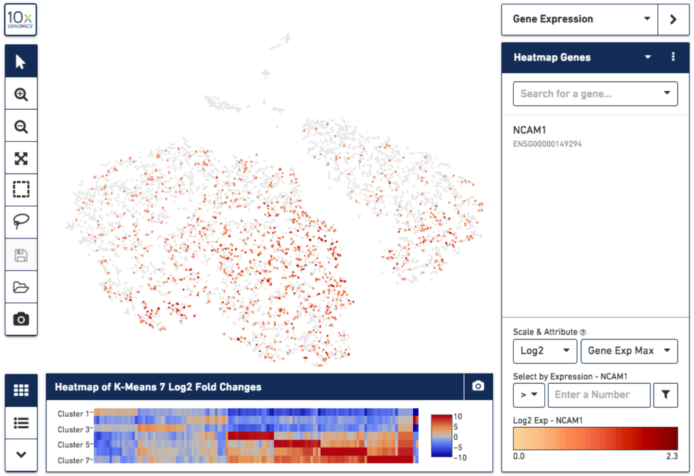
Categories

LibraryID

- ☑ GS_Tumor_10_2016_6525 (... 🟨
- ☑ GS_Tumor_9_2013_2559 (2... 🟦

Heatmap of Graph-Based Log2 Fold Changes

Experimental conditions



Gene Expression

Heatmap Genes

Search for a gene...

NCAM1
ENSG00000149294

Scale & Attribute: Log2; Gene Exp Max

Select by Expression - NCAM1: > ; Enter a Number

Log2 Exp - NCAM1: 0.0 – 2.3

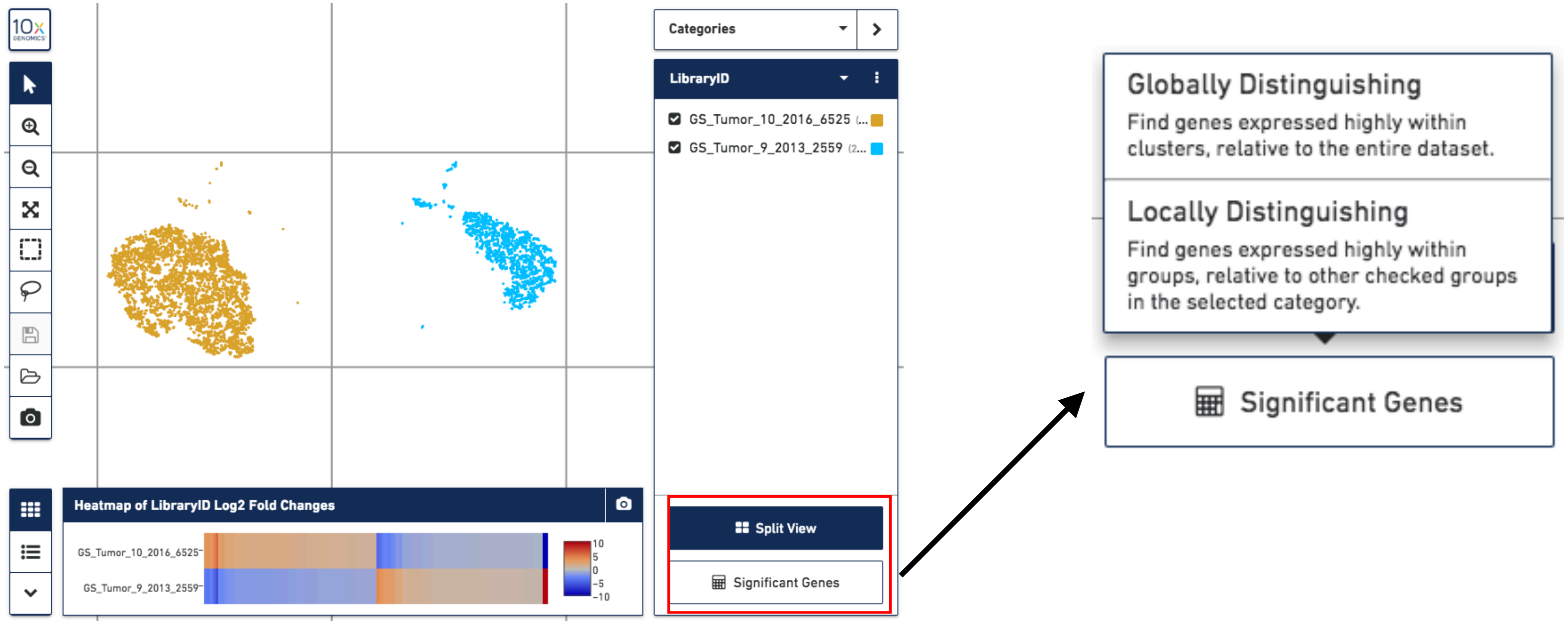
Heatmap of K-Means 7 Log2 Fold Changes

Gene expression level

FRED HUTCH
CURES START HERE®

Loupe Cell Browser - Finding Significant Genes -

- Determine genes that uniquely characterize at the press of a button
- **Significant Genes** allows us to find genes that distinguish cluster from entire population or other selected clusters



Globally Distinguishing

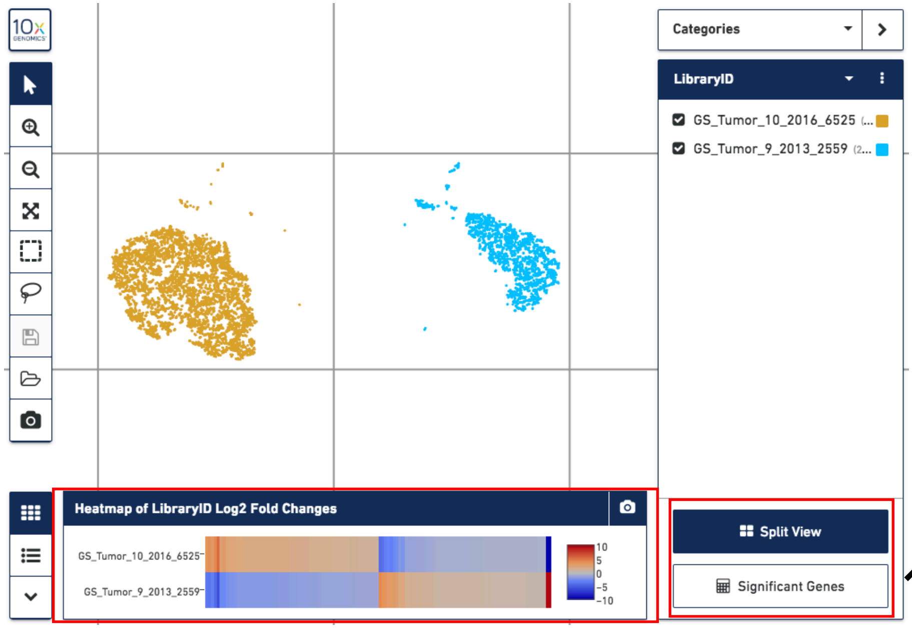
Find genes expressed highly within clusters, relative to the entire dataset.

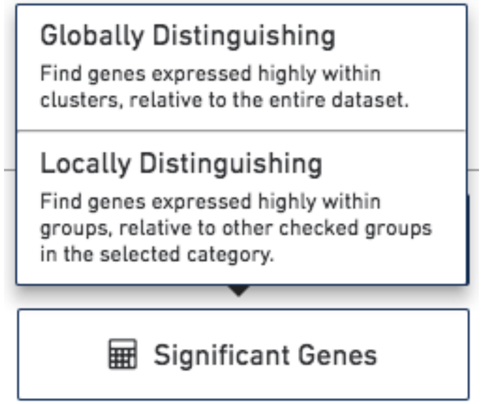
Locally Distinguishing

Find genes expressed highly within groups, relative to other checked groups in the selected category.

Loupe Cell Browser - Finding Significant Genes -

- Determine genes that uniquely characterize at the press of a button
- Significant Genes allows us to find genes that distinguish cluster from entire population or other selected clusters




Globally Distinguishing

Find genes expressed highly within clusters, relative to the entire dataset.

Locally Distinguishing

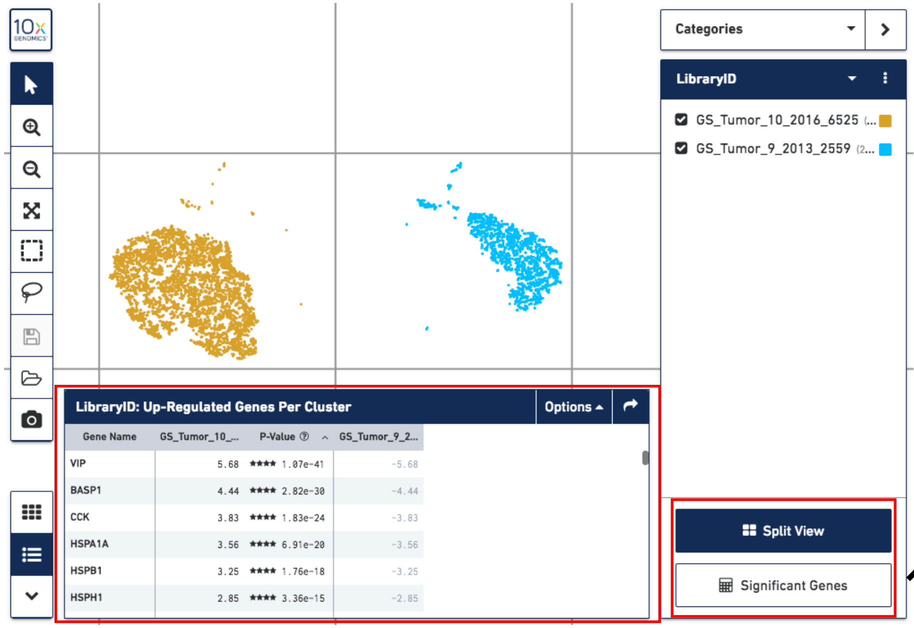
Find genes expressed highly within groups, relative to other checked groups in the selected category.

Heatmap - Top up-regulating genes per cluster

FRED HUTCH
CURES START HERE®

Loupe Cell Browser - Finding Significant Genes -

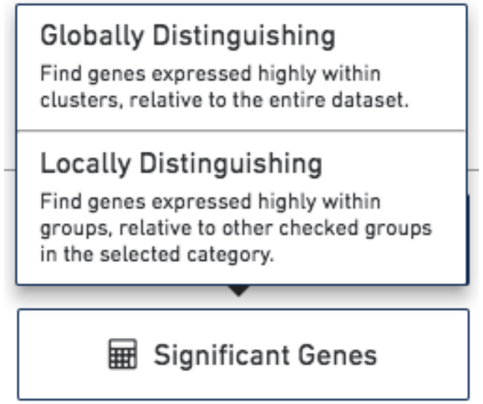
- Determine genes that uniquely characterize at the press of a button
- Significant Genes allows us to find genes that distinguish cluster from entire population or other selected clusters



☑ GS_Tumor_10_2016_6525 (...
☑ GS_Tumor_9_2013_2559 (2...

LibraryID: Up-Regulated Genes Per Cluster

Gene Name	GS_Tumor_10_...	P-Value ⓘ	GS_Tumor_9_2...
VIP	5.68	**** 1.07e-41	-5.68
BASP1	4.44	**** 2.82e-30	-4.44
CCK	3.83	**** 1.83e-24	-3.83
HSPA1A	3.56	**** 6.91e-20	-3.56
HSPB1	3.25	**** 1.76e-18	-3.25
HSPH1	2.85	**** 3.36e-15	-2.85



Globally Distinguishing

Find genes expressed highly within clusters, relative to the entire dataset.

Locally Distinguishing

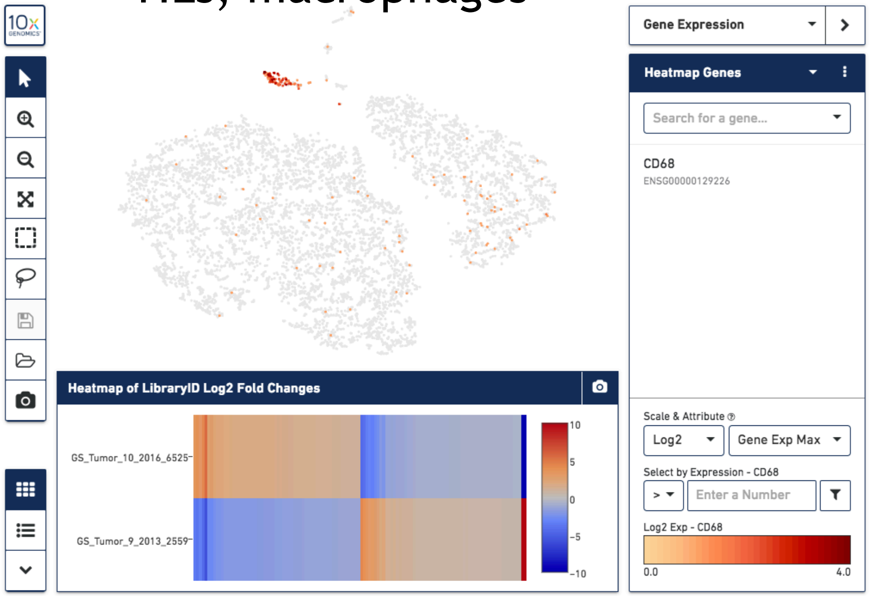
Find genes expressed highly within groups, relative to other checked groups in the selected category.

Significant gene information in a tabular view

Loupe Cell Browser - Identifying Cell Types -

- Look at well-know markers to identify cell types
- Refine functional subgroup by filtering by expression level

TILs, macrophages

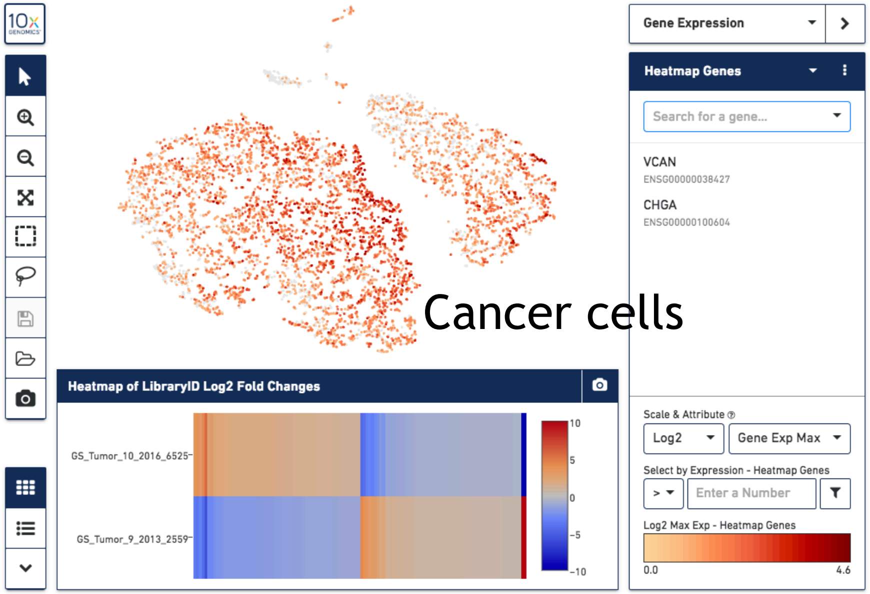


CD68
ENSG00000129226

Scale & Attribute
Log2
Gene Exp Max

Select by Expression - CD68
> Enter a Number

Log2 Exp - CD68
0.0 4.0



Cancer cells

VCAN
ENSG00000038427

CHGA
ENSG00000100604

Scale & Attribute
Log2
Gene Exp Max

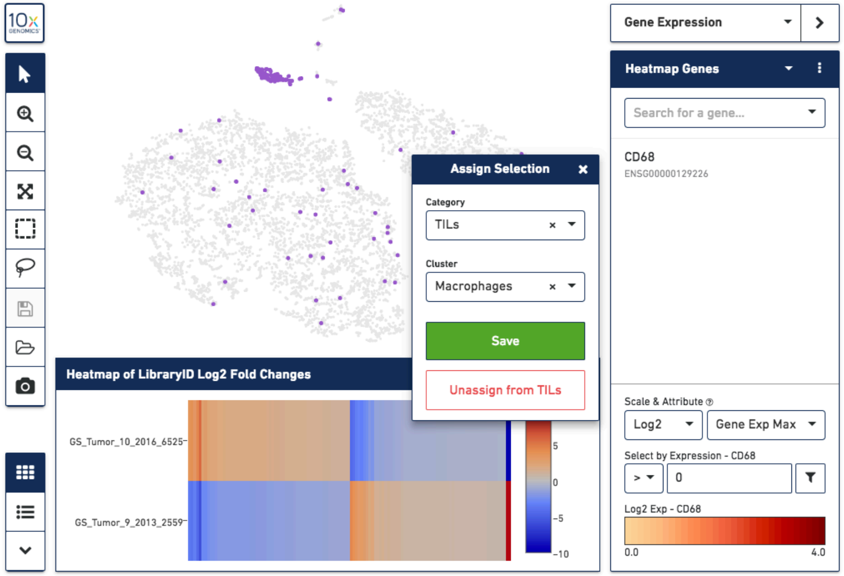
Select by Expression - Heatmap Genes
> Enter a Number

Log2 Max Exp - Heatmap Genes
0.0 4.6

FRED HUTCH
CURES START HERE®

Loupe Cell Browser - Exploring Substructure -

- Creating subgroups by expression level
- Modifying subgroups with manual selection



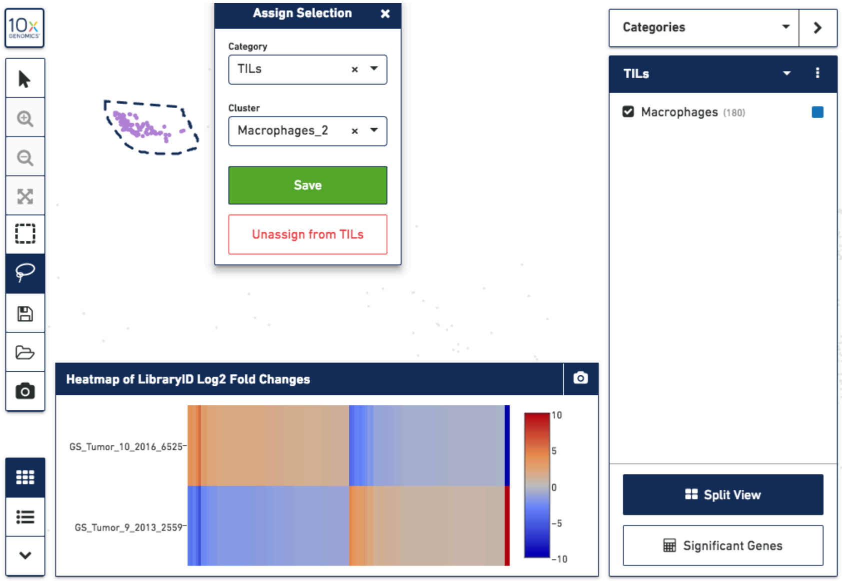
Assign Selection

Category: TILs

Cluster: Macrophages

Scale & Attribute: Log2, Gene Exp Max

Select by Expression - CD68: > 0



Assign Selection

Category: TILs

Cluster: Macrophages_2

Categories: TILs

☑ Macrophages (180)

Lasso/rectangular select points to create new categories and clusters/types

Loupe Cell Browser - Import/Export Functionality -

- Import: possible to import a *.csv* file with list of genes
  - Ensembl ID and/or gene name, list name
  - Ensembl/name must match value in transcriptome reference used in CellRanger Pipeline
- Export:
  - *.csv* gene list
  - Distinguishing genes data table
  - Distinguishing genes heatmap (*.png* - with gene names)
  - tSNE plot screenshot (*.png*)
- Save your work inside *.cloupe* file

Loupe™ Cell Browser *in live!!*

CURES START HERE

FRED HUTCH
CURES START HERE®

THANK YOU

FRED HUTCH
CURES START HERE®

fredhutch.org