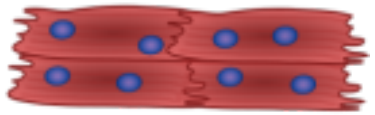
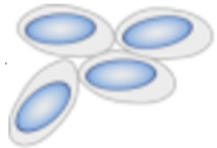
# scRNASeq
## Principles and analytic objectives (with examples)

Eric J. Kort MD MS FAAP

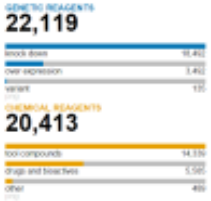
*Research Scientist,* DeVos Cardiovascular Research Program / Jovinge Lab
*Assistant Professor,* Department of Pediatrics and Human Development, Michigan State University
*Pediatric Hospitalist,* Helen DeVos Children's Hospital, Spectrum Health

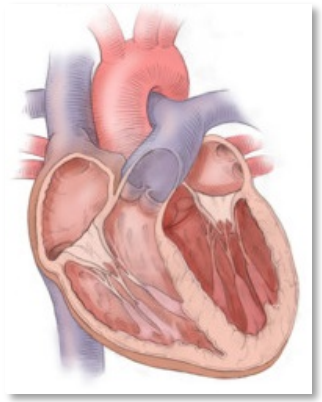


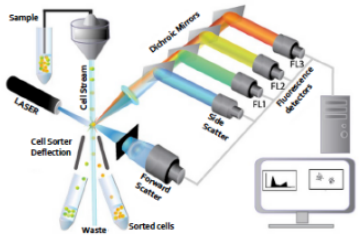
Cardiomyocytes

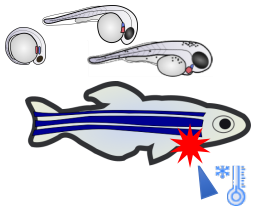


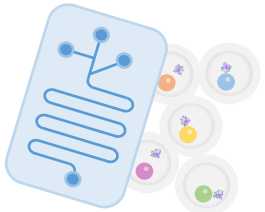
iPS cells

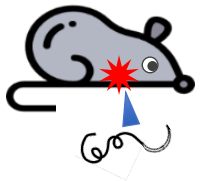








REDCap
Research Electronic Data Capture

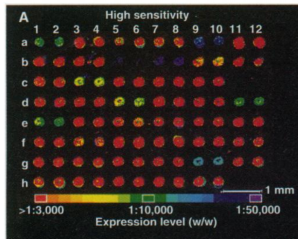
# Background and platforms

# A brief history of gene expression profiling

**Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray**

Mark Schena,* Dari Shalon,*† Ronald W. Davis, Patrick O. Brown‡

A high-capacity system was developed to monitor the expression of many genes in parallel. Microarrays prepared by high-speed robotic printing of complementary DNAs on glass were used for quantitative expression measurements of the corresponding genes. Because of the small format and high density of the arrays, hybridization volumes of 2 microliters could be used that enabled detection of rare transcripts in probe mixtures derived from 2 micrograms of total cellular messenger RNA. Differential expression measurements of 45 *Arabidopsis* genes were made by means of simultaneous, two-color fluorescence hybridization. SCIENCE • VOL. 270 • 20 OCTOBER 1995



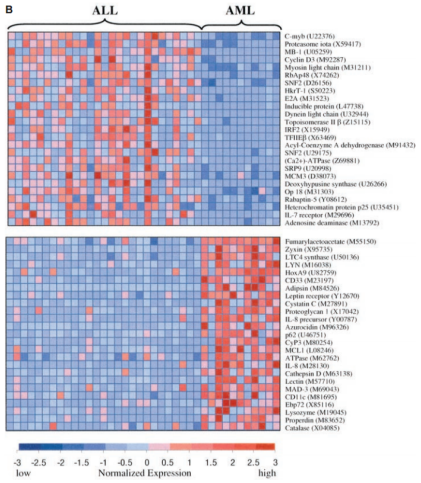


REPORTS

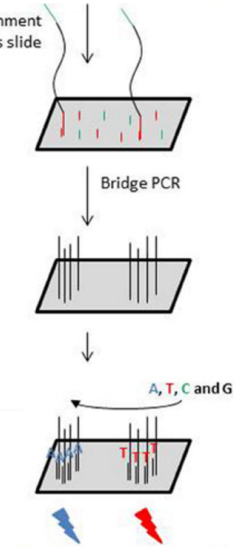
**Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring**

T. R. Golub,[1,2*†] D. K. Slonim,[1†] P. Tamayo,[1] C. Huard,[1] M. Gaasenbeek,[1] J. P. Mesirov,[1] H. Coller,[1] M. L. Loh,[2] J. R. Downing,[3] M. A. Caligiuri,[4] C. D. Bloomfield,[4] E. S. Lander[1,5*]

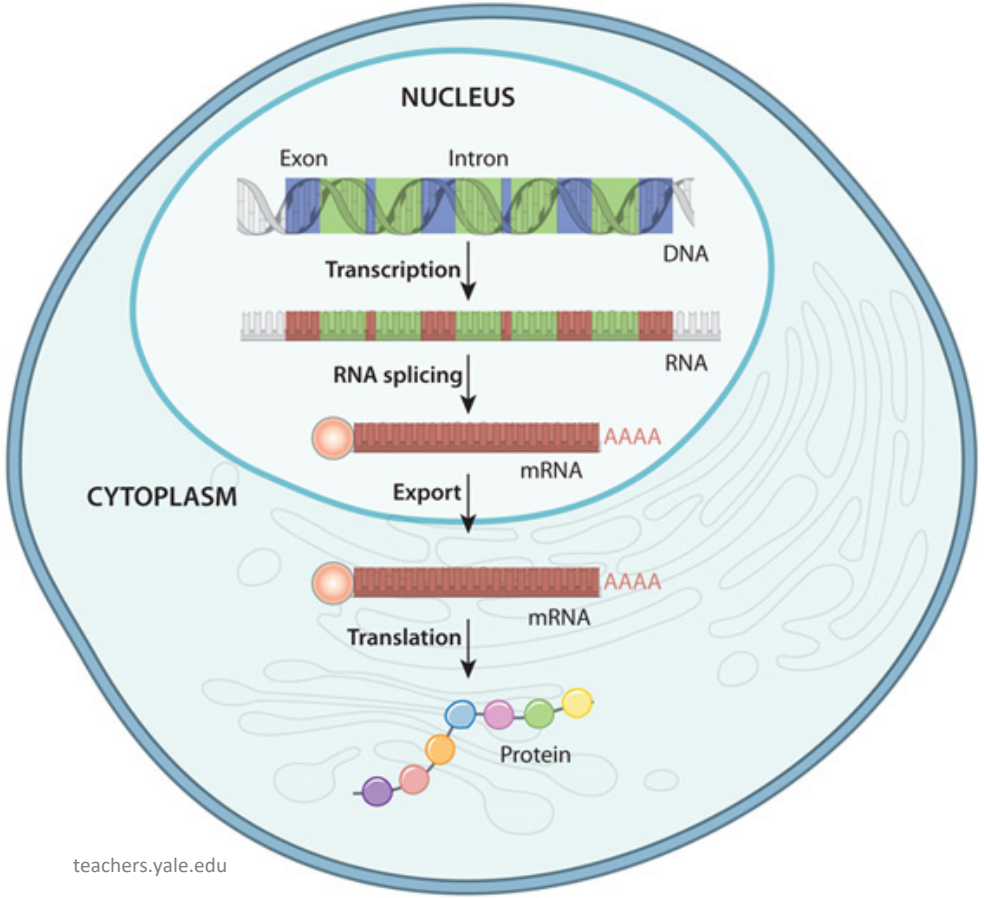
SCIENCE VOL 286 15 OCTOBER 1999







# Why bother?




teachers.yale.edu

# Single cells: the next frontier

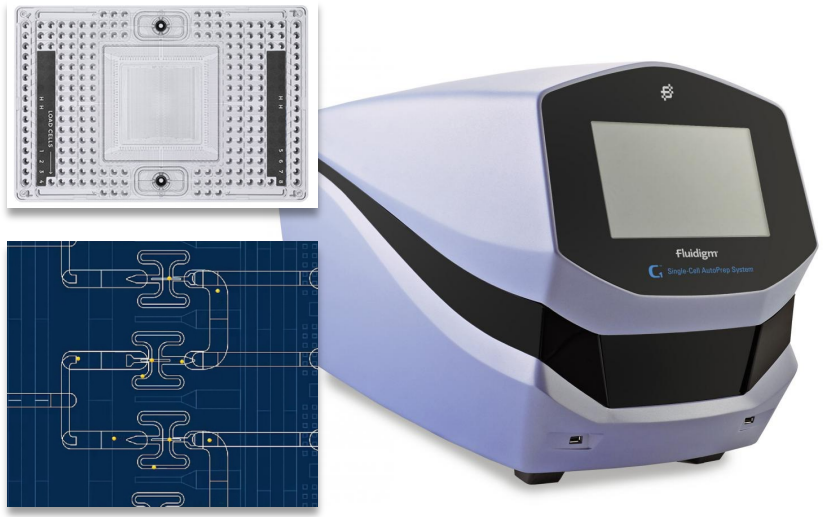
Bulk RNASeq

Single Cell RNASeq

Metaphor: Alex Shalek & Aviv Regev, Harvard University

# Single Cell Profiling Platforms

***96 Well Plate***
**Initial cost:** Very low
**Cost per cell:** Medium
**Throughput:** Low
**Transcript capture:** High
**Sequencing depth:** High




***Fluidigm C1***
**Initial cost:** Medium
**Cost per cell:** High
**Throughput:** Low
**Transcript capture:** High
**Sequencing depth:** High

***InDrop / DropSeq / 10X***
**Initial cost:** Medium
**Cost per cell:** Low
**Throughput:** High
**Transcript capture:** Low
**Sequencing Depth:** Low

# Fluidics Based scRNASeq

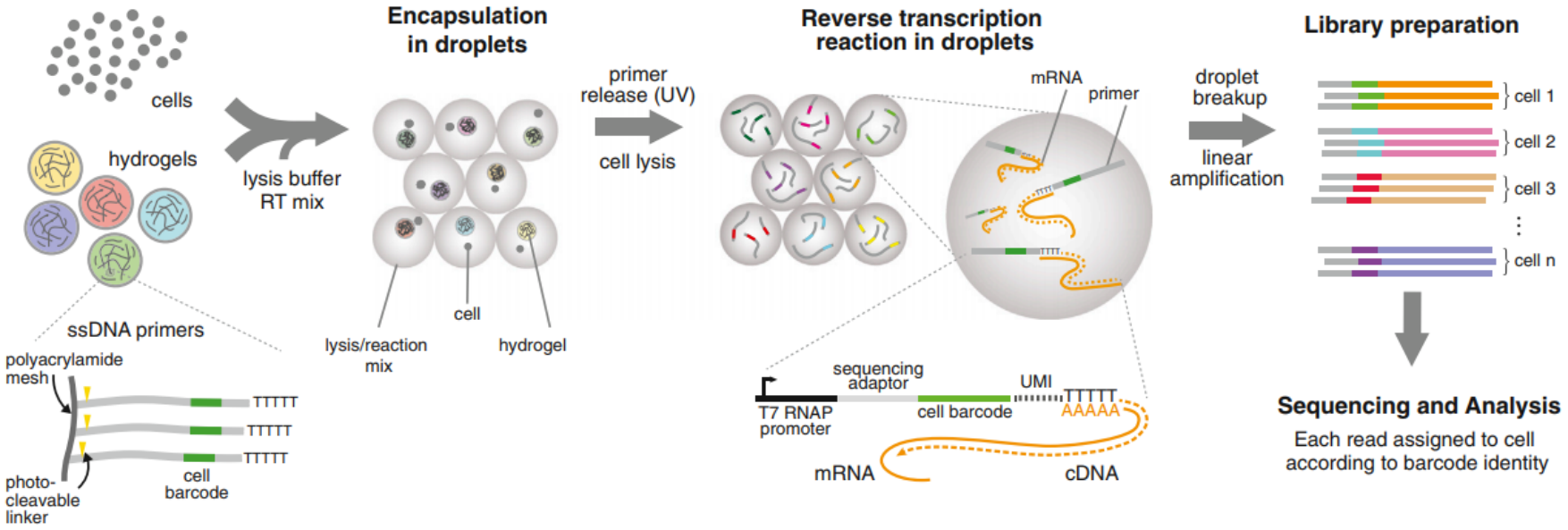



**Figure 1. A Platform for DNA Barcoding Thousands of Cells**

Cells are encapsulated into droplets with lysis buffer, reverse-transcription mix, and hydrogel microspheres carrying barcoded primers. After encapsulation primers are released. cDNA in each droplet is tagged with a barcode during reverse transcription. Droplets are then broken and material from all cells is linearly amplified before sequencing. UMI = unique molecular identifier.

# inDrop platform

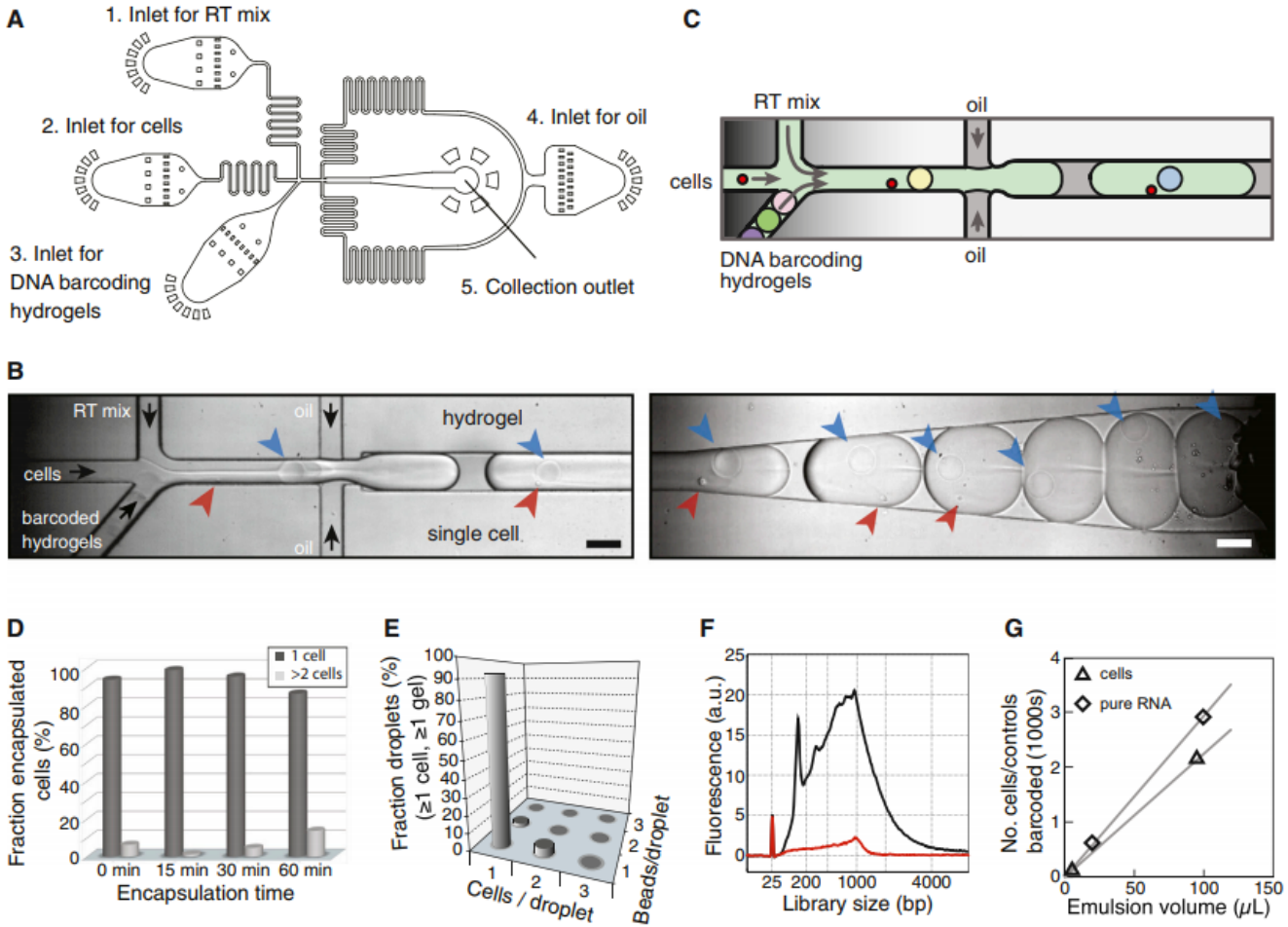



**Figure 3. A Droplet Barcoding Device**

(A) Microfluidic device design, see also Figure S2.

(B and C) Snapshots of encapsulation (left) and collection (right) modules, see also Movies S1 and S2. Arrows indicate cells (red), hydrogels (blue), and flow direction (black). Scale bars 100 μm.

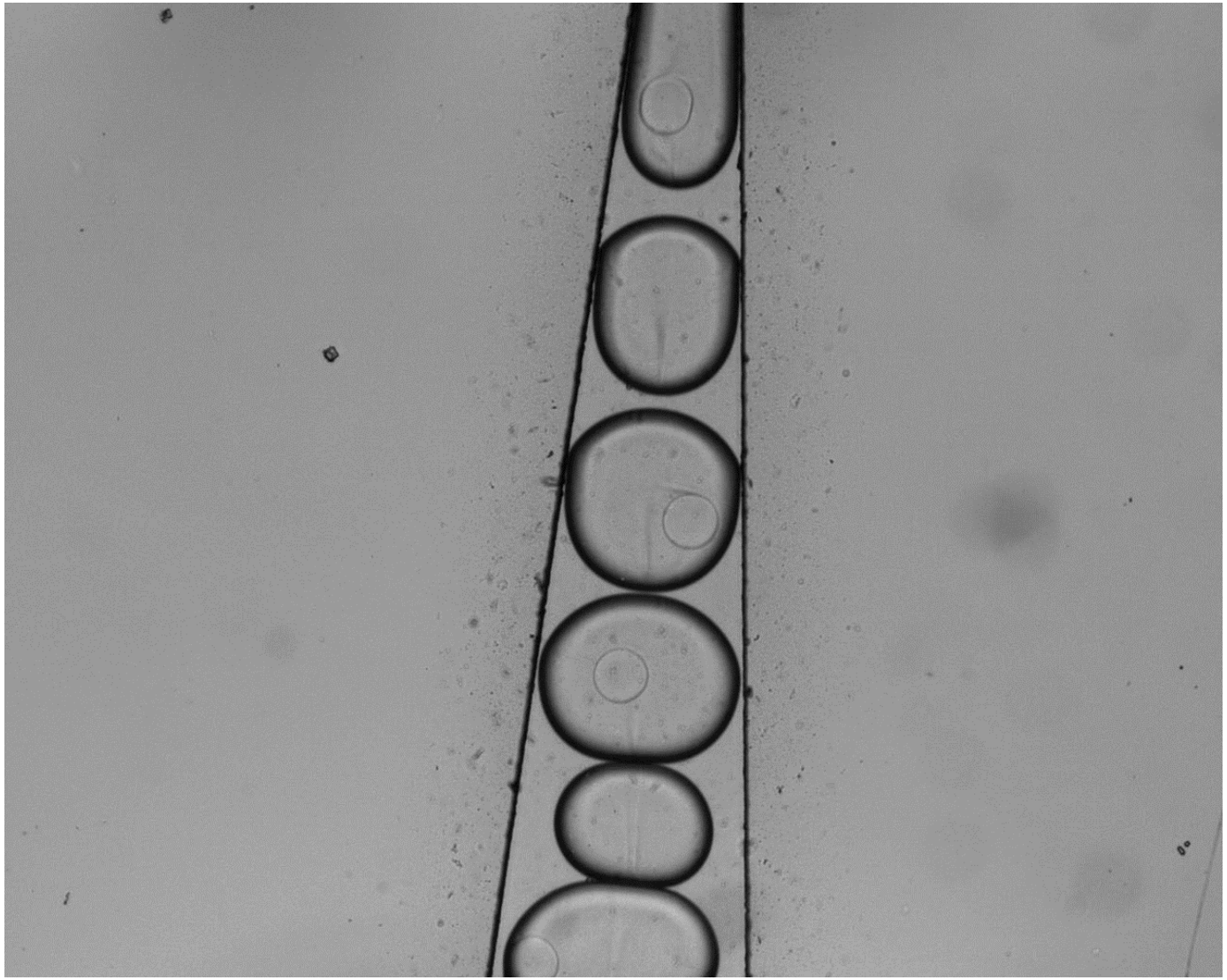
(D) Droplet occupancy over time.

(E) Cell and hydrogel co-encapsulation statistics showing a high 1:1 cell:hydrogel correspondence.

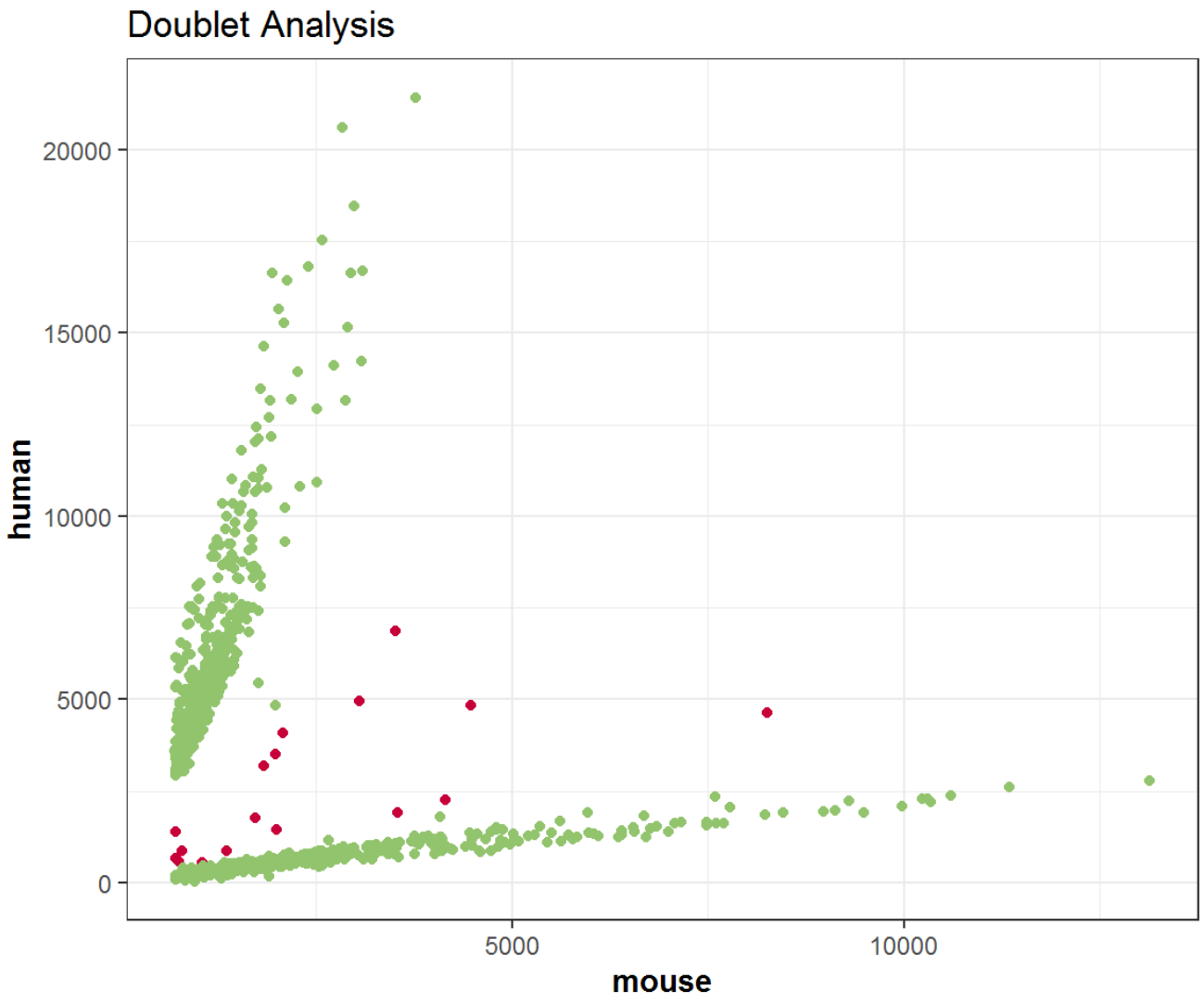
(F) BioAnalyzer traces showing dependence of library abundance on primer photo-release.

(G) Number of cells/controls as a function of collection volume.

# inDrop platform



# Species Mixing Experiment



# How do they do that?

- Sequencing adapter + Cell barcode + UMI + Other artefacts

- Challenge: everyone has their own scheme.

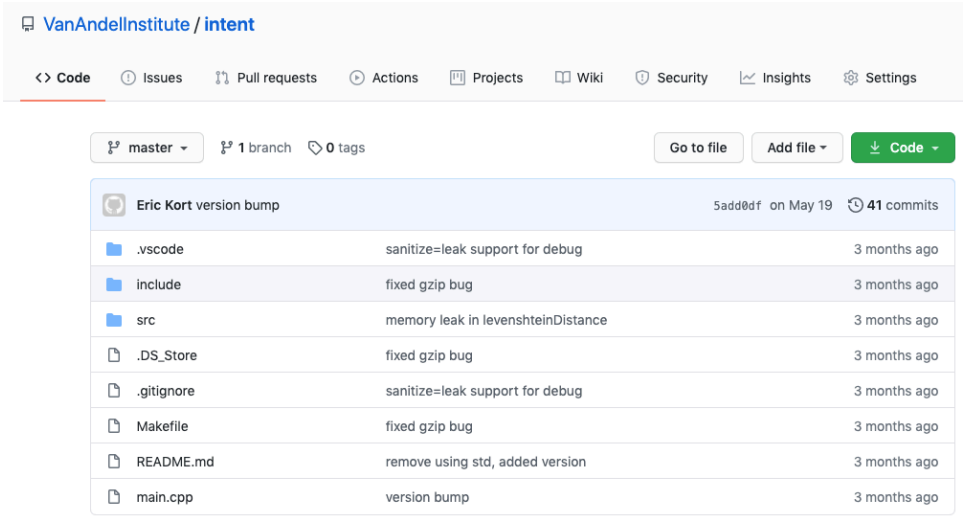
- Solution 1: Use the manufacturer's pipeline
- Solution 2: Rearrange the sequences yourself

10X Genomics (V3) Cell Barcode (C) and UMI (U)

```
CCCCCCCCCCCCCCCCUUUUUUUUUUUU
```

inDrop (V2) Cell Barcode (C) and UMI (U)

```
CCCCCCCCGAGTGATTGCTTGTGACGCCTTCCCCCCCCUUUUUU
```



# Analysis challenges and solutions

# scRNASeq: The Challenges

1. Deconvolution of the metadata encoded in the sequence
2. Discriminating non-cells, cells, and aggregates
3. Sparse matrix
   - Technical dropouts vs. biological dropouts
4. Normalization
   - Sequencing depth
5. Batch correction
6. Cell type identification

# Common tools: pre-requisites

- **Hardware**
  - ”Lots” of memory
  - More cores helps (to a point)
  - Maybe adequate: Your laptop (e.g., 4 cores, 16GB of RAM)
  - Very likely adequate: c5.9xlarge (36 cores, 72GB memory)
- **FASTQ → Counts**
  - Most pipelines require python
  - Or you can use STAR, Salmon, etc.
- Counts → Downstream analysis
  - R
  - But there’s a catch...package dependencies (curl, ssl, xml2, etc.)
  - https://vanandelinstitute.github.io/single_cell_analyses/setting_up_shop.html

# Common tools: sequences → counts

- **Goal:**

  *Take a set of FASTQ files and...*

  *convert sequences to transcript counts*

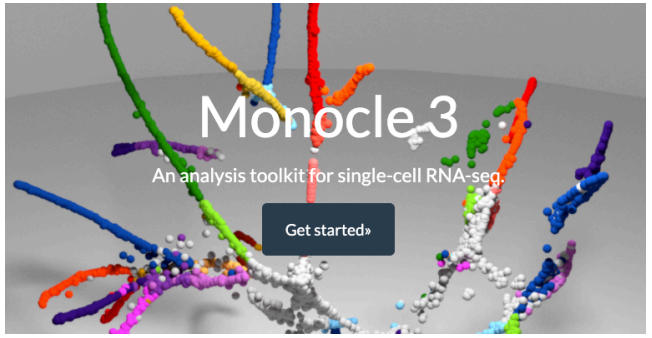
  *assign counts the appropriate cell barcode*

  *tossing out duplicated reads based on UMI*

- **Platform specific:**
  - InDrop Pipeline
  - CellRanger Pipeline (10X Chromium)
- **Platform agnostic:**
  - Salmon Alevin
  - STARSolo
  - Kallisto

# Common tools: counts → everything else

- **Monocle**
  - Mainly a set of tools developed by the Cole Trapnell lab
  - A fairly cohesive pipeline, well documented
- **Seurat**
  - A collection of tools developed by various people
  - Not so cohesive, but the latest and greatest tools are often created for Seurat

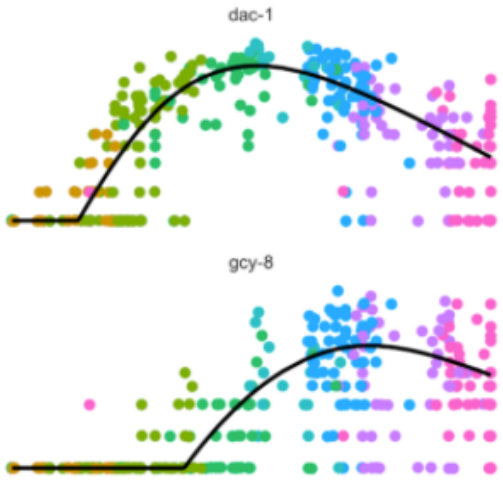




- Challenge: various data formats.

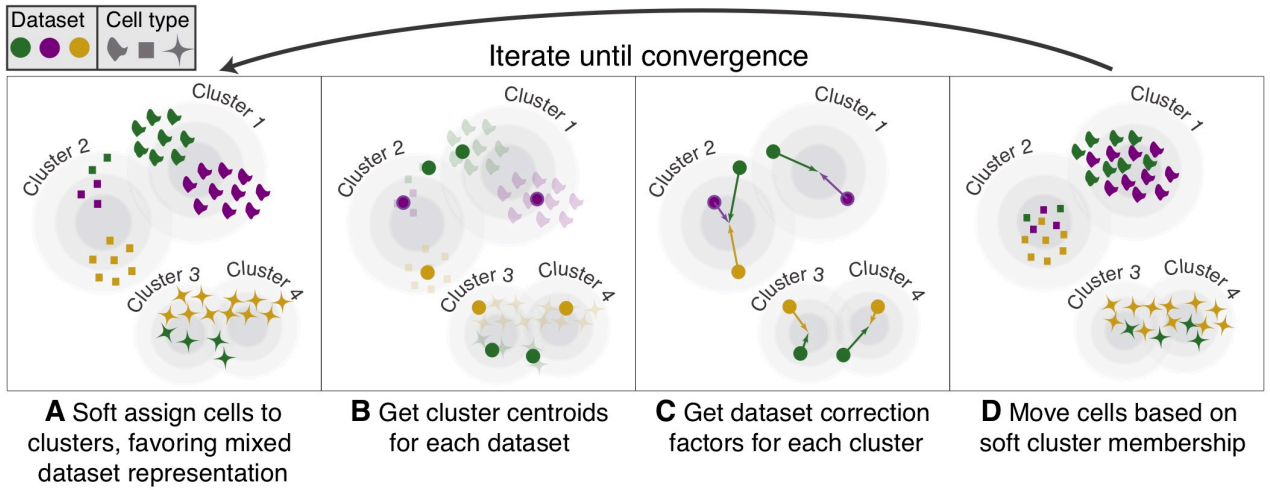
# The workflow

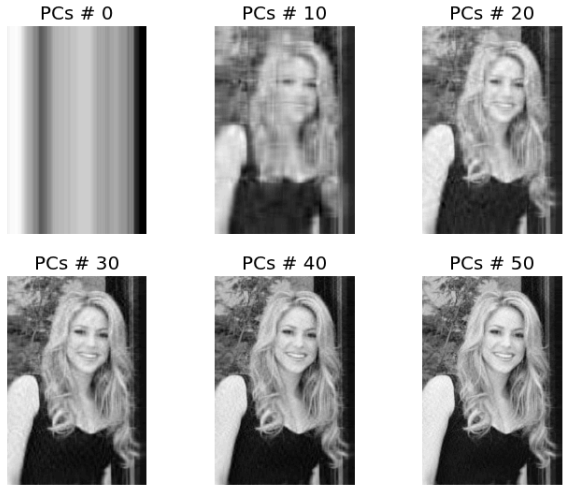
1. Process sequencing data to counts
2. Load into data object for your favorite toolset
3. Normalize
   - Depth normalize and log transform
     - (after you add 1 and multiply by 10,000)
   - Or something fancier like SCTransform
4. Remove batch effects
5. Assign cell types
6. Reduce dimensions =(
   - UMAP, TSNE
7. Do some interesting analysis
   - Pseudotime
   - Differential gene expression
   - Geneset enrichment

# Batch effect removal

- Low dimensional interpolation
  - Harmony, Batchelor



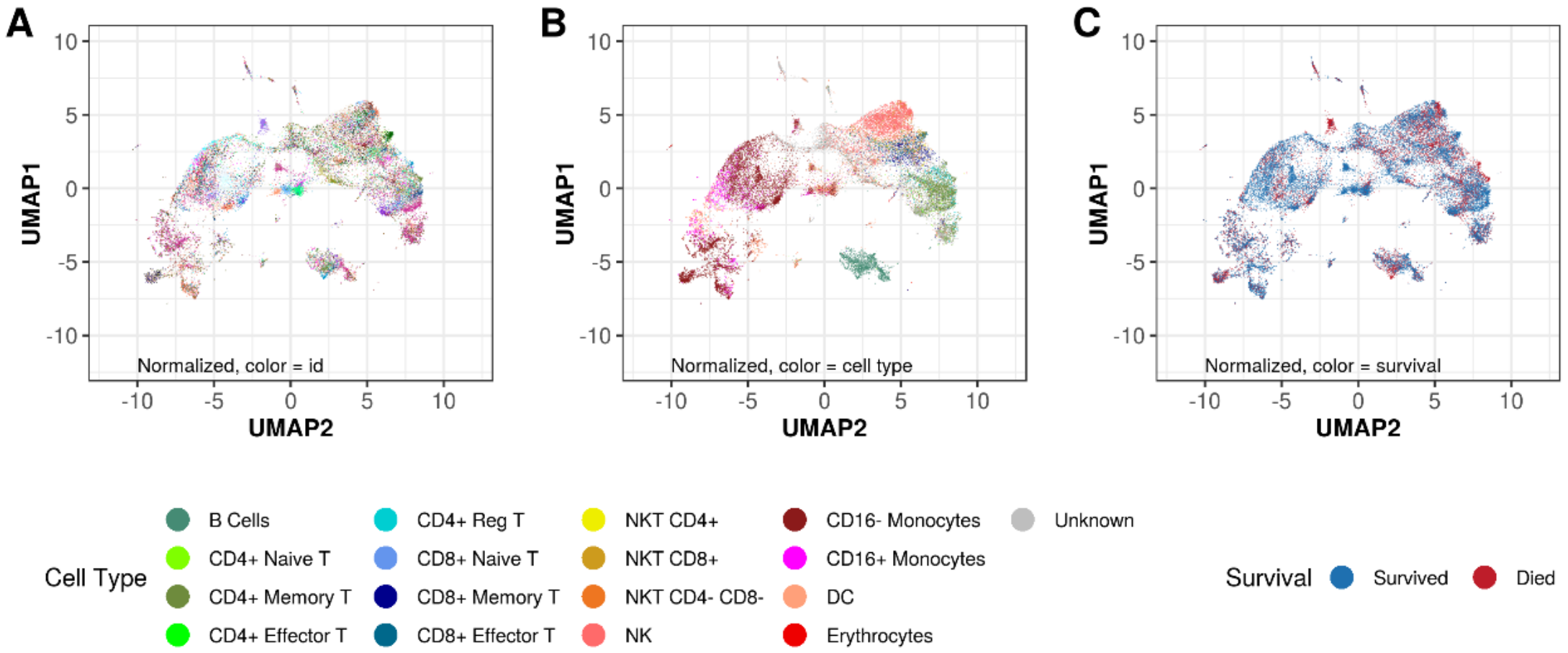




- Gene level modeling

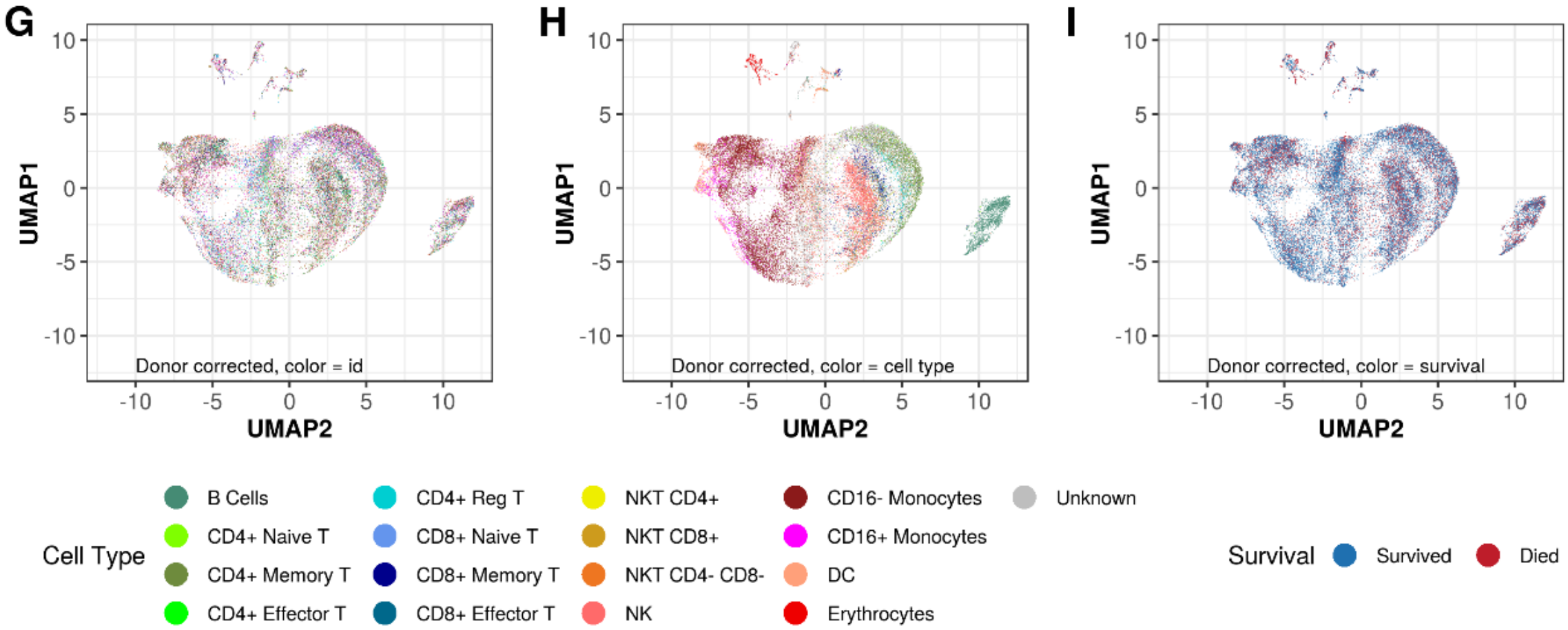
```
gene_fits <- fit_models(cds, model_formula_str = "~ CellTypen + Batch")
```

- Hybrid
  - Counts → PCA → Batch Removal → Counts (by reversing PCA)

# Batch Effect Example



# After correction



# Imputation of dropouts

New Results

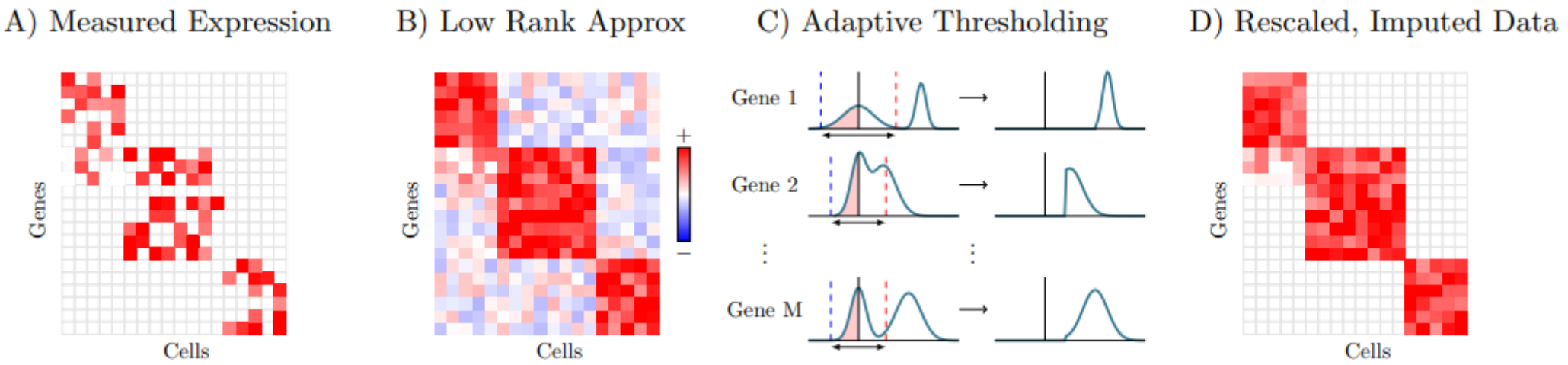
Comment on this paper

**Zero-preserving imputation of scRNA-seq data using low-rank approximation**

George C. Linderman, Jun Zhao, Yuval Kluger

**doi:** https://doi.org/10.1101/397588





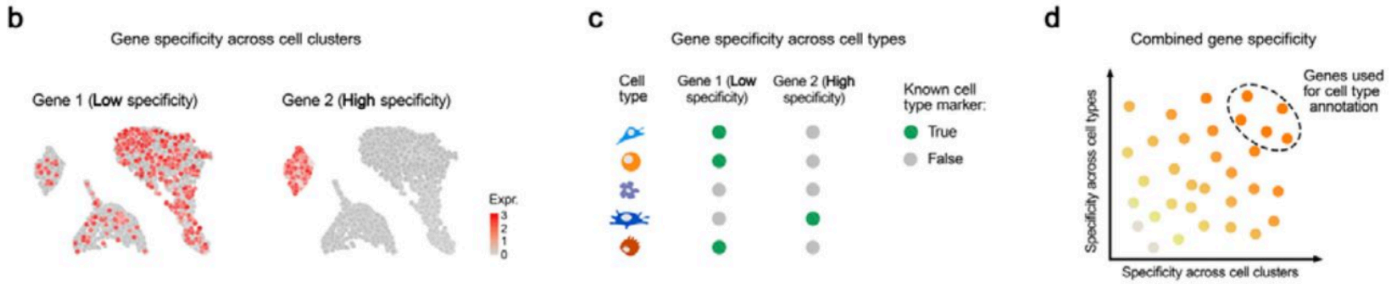
# So many tools...

## scType

**Fully-automated cell-type identification with specific markers extracted from single-cell transcriptomic data**

Aleksandr Ianevski, Anil K Giri, Tero Aittokallio

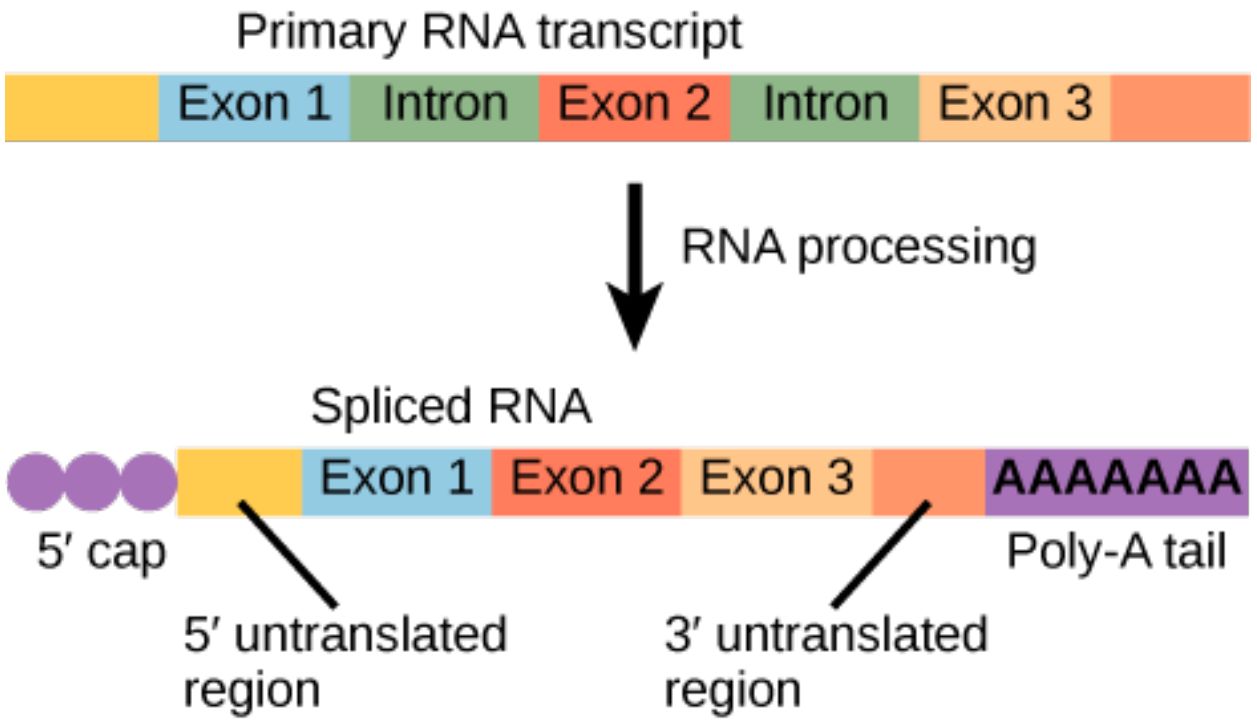
**doi:** https://doi.org/10.1101/812131




## Seurat v. 3.0

# One more thing

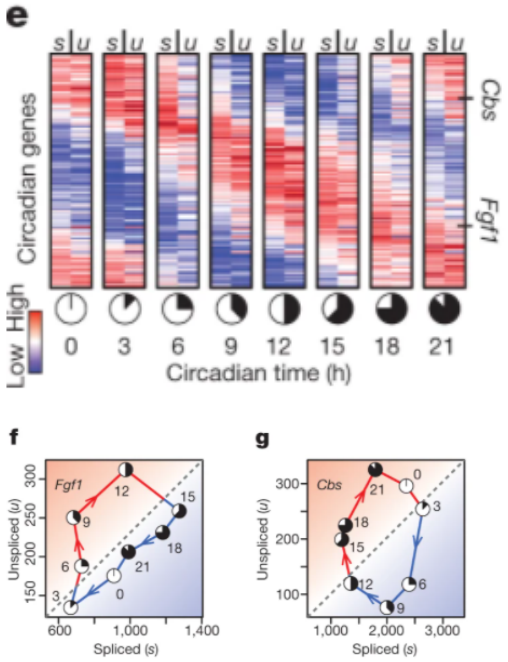
- Splicing???




https://courses.lumenlearning.com/

# One more thing

- Splicing → RNA “Velocity”



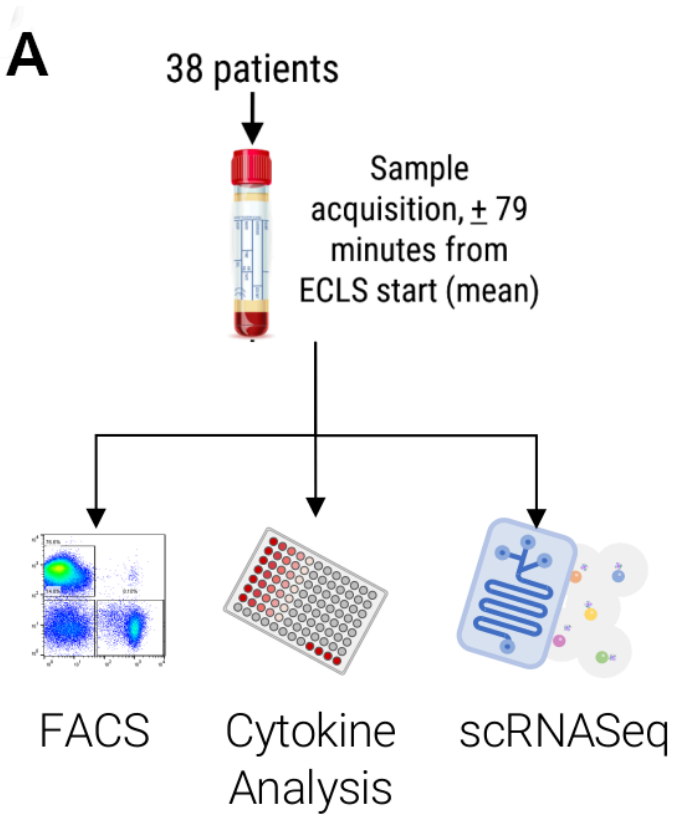


- For more details see:
  https://vanandelinstitute.github.io/single_cell_analyses/rnavel_alevin_scvelo.html

  (the **Rmarkdown/reticulate** equivalent to Salmon Alevin tutorial)

# An example from the wild

# An example


A
38 patients
Sample acquisition, ± 79 minutes from ECLS start (mean)
FACS
Cytokine Analysis
scRNASeq

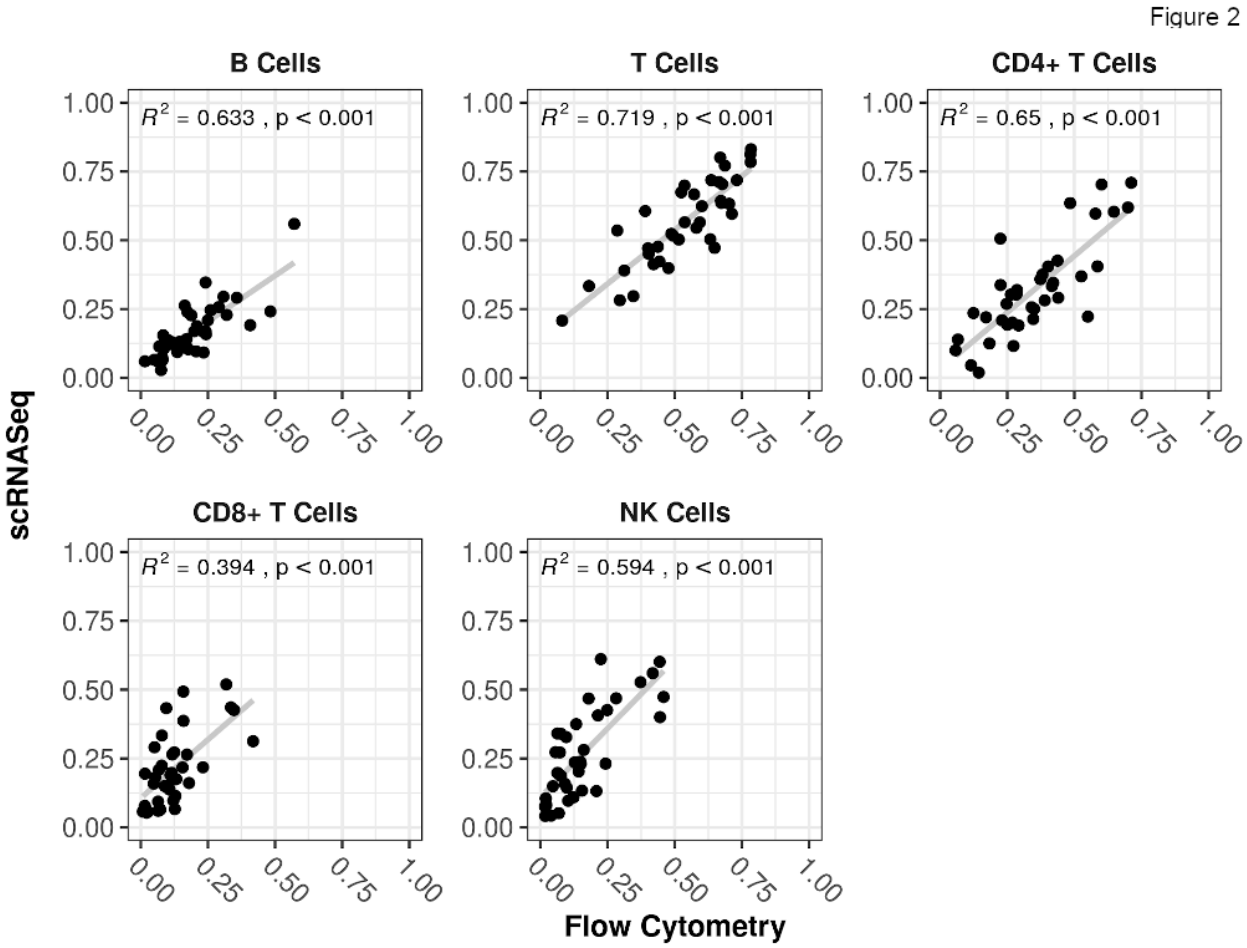
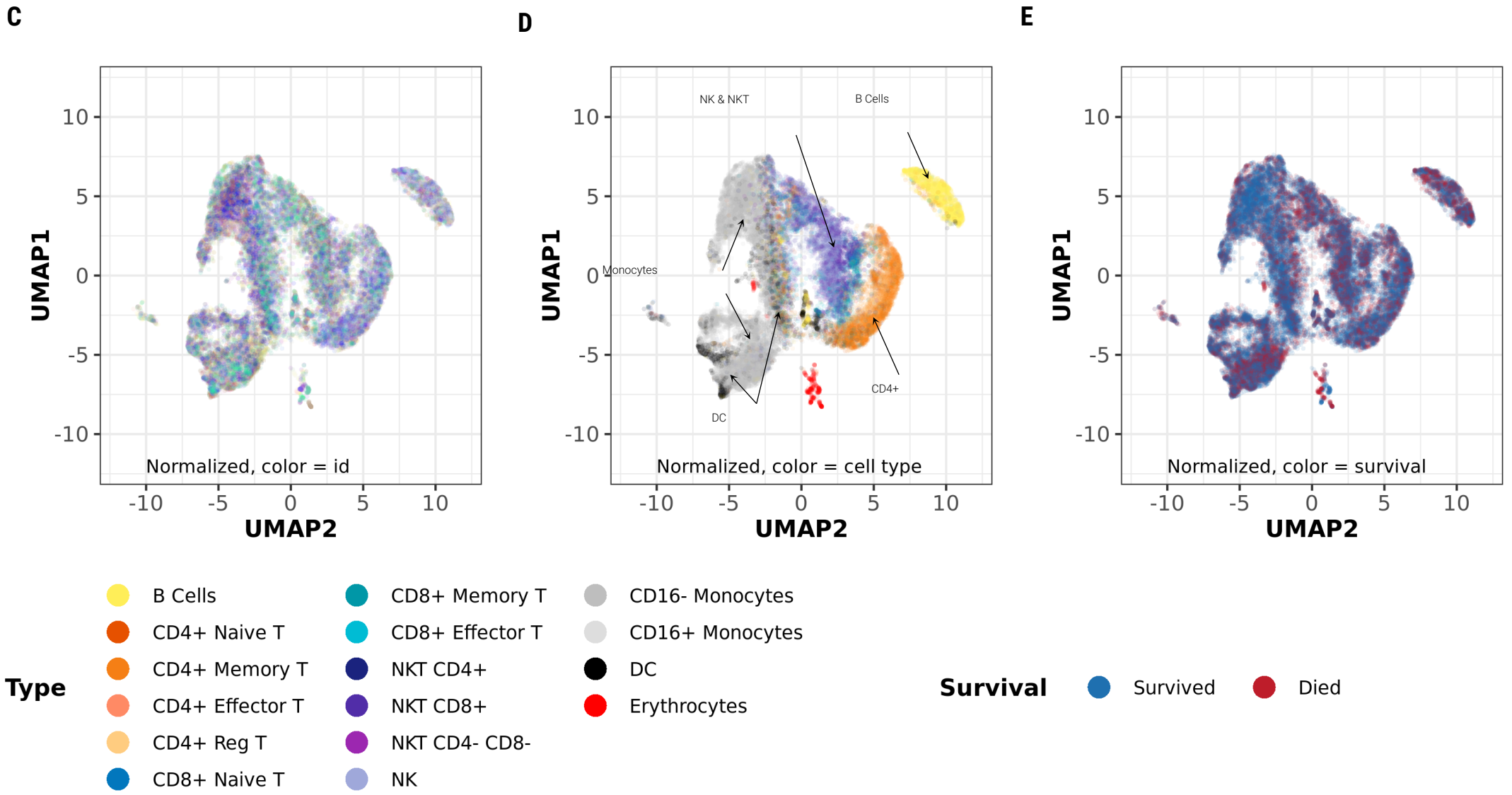


Figure 2
B
B Cells
T Cells
CD4+ T Cells
CD8+ T Cells
NK Cells
$R^2$ = 0.633 , p < 0.001
$R^2$ = 0.719 , p < 0.001
$R^2$ = 0.65 , p < 0.001
$R^2$ = 0.394 , p < 0.001
$R^2$ = 0.594 , p < 0.001
scRNASeq
Flow Cytometry
0.00
0.25
0.50
0.75
1.00

# An example


C
D
E
UMAP1
UMAP2
Normalized, color = id
NK & NKT
B Cells
Monocytes
CD4+
DC
Normalized, color = cell type
Normalized, color = survival
Type
B Cells
CD4+ Naive T
CD4+ Memory T
CD4+ Effector T
CD4+ Reg T
CD8+ Naive T
CD8+ Memory T
CD8+ Effector T
NKT CD4+
NKT CD8+
NKT CD4- CD8-
NK
CD16- Monocytes
CD16+ Monocytes
DC
Erythrocytes
Survival
Survived
Died

# An example

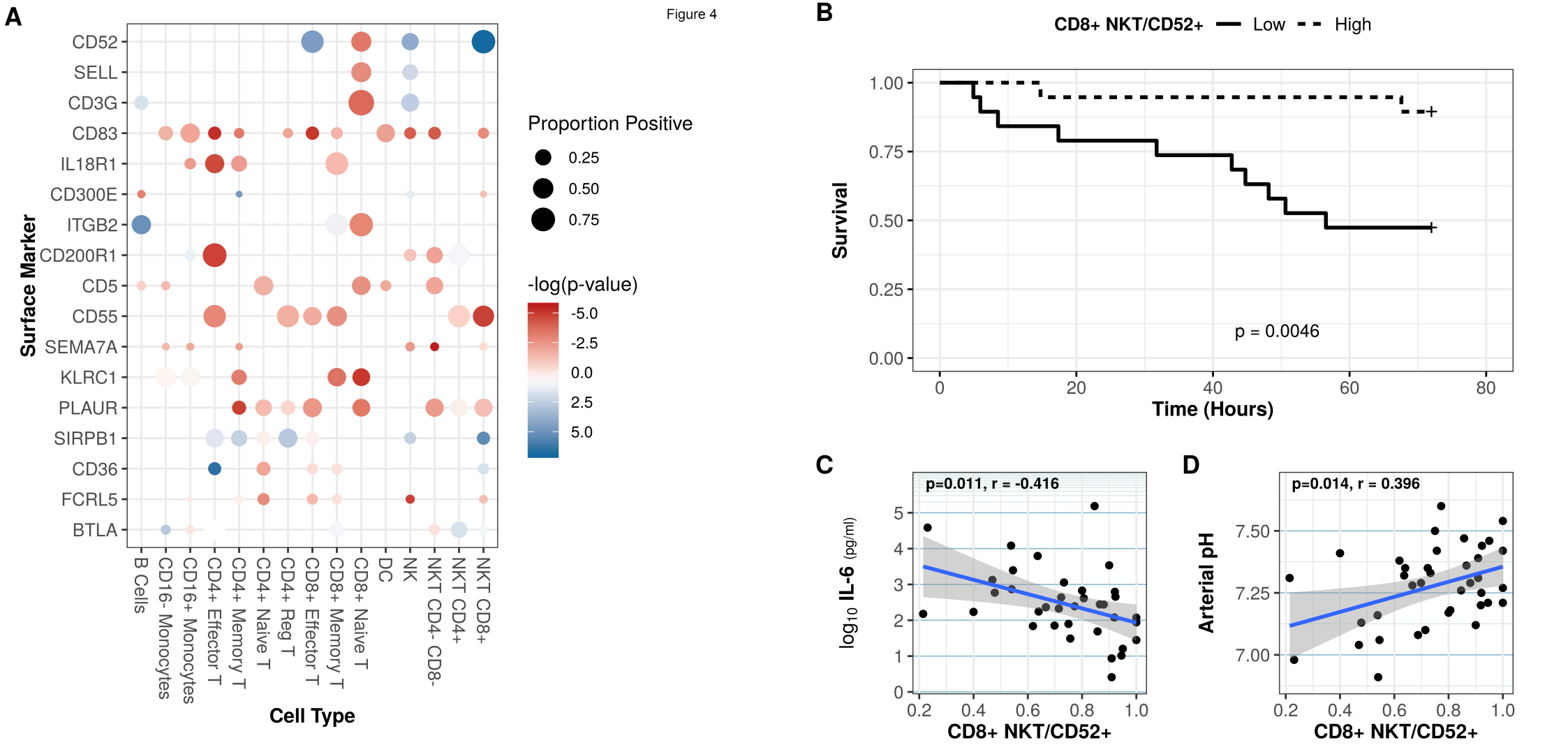
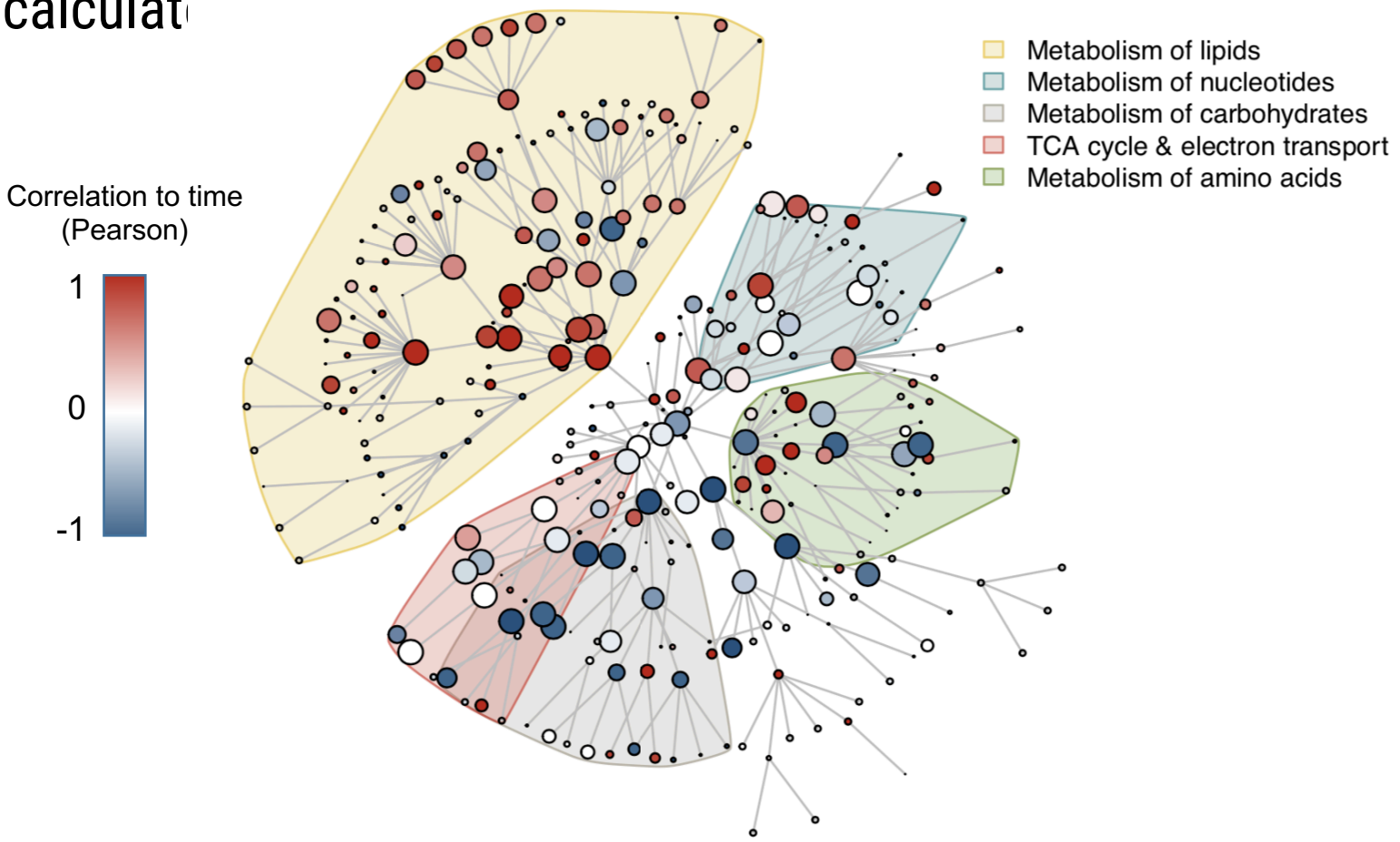


Figure 4
A
Surface Marker
CD52
SELL
CD3G
CD83
IL18R1
CD300E
ITGB2
CD200R1
CD5
CD55
SEMA7A
KLRC1
PLAUR
SIRPB1
CD36
FCRL5
BTLA
B Cells
CD16- Monocytes
CD16+ Monocytes
CD4+ Effector T
CD4+ Memory T
CD4+ Naive T
CD4+ Reg T
CD8+ Effector T
CD8+ Memory T
CD8+ Naive T
DC
NK
NKT CD4- CD8-
NKT CD4+
NKT CD8+
Cell Type
Proportion Positive
0.25
0.50
0.75
-log(p-value)
-5.0
-2.5
0.0
2.5
5.0
B
CD8+ NKT/CD52+ Low High
Survival
1.00
0.75
0.50
0.25
0.00
p = 0.0046
0
20
40
60
80
Time (Hours)
C
p=0.011, r = -0.416
log10 IL-6 (pg/ml)
CD8+ NKT/CD52+
D
p=0.014, r = 0.396
Arterial pH
7.50
7.25
7.00
CD8+ NKT/CD52+

# Gene enrichment example:

- For every reactome metabolism related term:
  - Genes identified
  - Enrichment for each cell calculat
    - Number non-zero cells
    - Mean of non-zero cells
  - Correlated with time




Click for interactive version

# A Note on Reproducible Research

“Six month ago you is your worst collaborator and doesn’t answer email”

# A Note on Reproducible Research



## Data Analysis Supplement

*April 9, 2019*

### Introduction

This document was prepared in support of our paper:

**Single cell transcriptomics identifies immunologic priming related to extra corporeal life support survival.**


Eric J. Kort MD, Matthew Weiland, Edgars Grins MD, Emily Eugster MS, Hsiao-yun Milliron PhD, Catherine Kelty MS, Nabin Manandhar Shrestha PhD, Tomasz Timek MD, Marzia Leacche MD, Stephen J Fitch MD, Theodore J Boeve MD, Greg Marco MD, Michael Dickinson MD, Penny Wilton MD, Stefan Jovinge MD PhD


The following sections document how the data for this paper were processed and how the figures were generated. Those who wish to do so can recreate the figures from the paper from data posted on GEO under accession GSE127221, and the code below. For efficient compiling of this document, the "eval=FALSE" option was set globally. Readers desiring to repeat the analysis can either run each code chunk below manually, or change the setting line at the top of the Rmd file from:

```
knitr::opts_chunk$set(echo = TRUE, eval=FALSE)
```

to

```
knitr::opts_chunk$set(echo = TRUE, eval=TRUE)
```

And then knit the entire document, a process which may take several hours, and may also fail if insufficient RAM is available.

The result of running this code is that all processed data and generated figures will be produced and saved to the Final_Data directory. The figures should match exactly what is in the publication, except that in figure 4 panel B the GO ontology term annotations were added manually (based on the GO ontology analysis that can be found below).

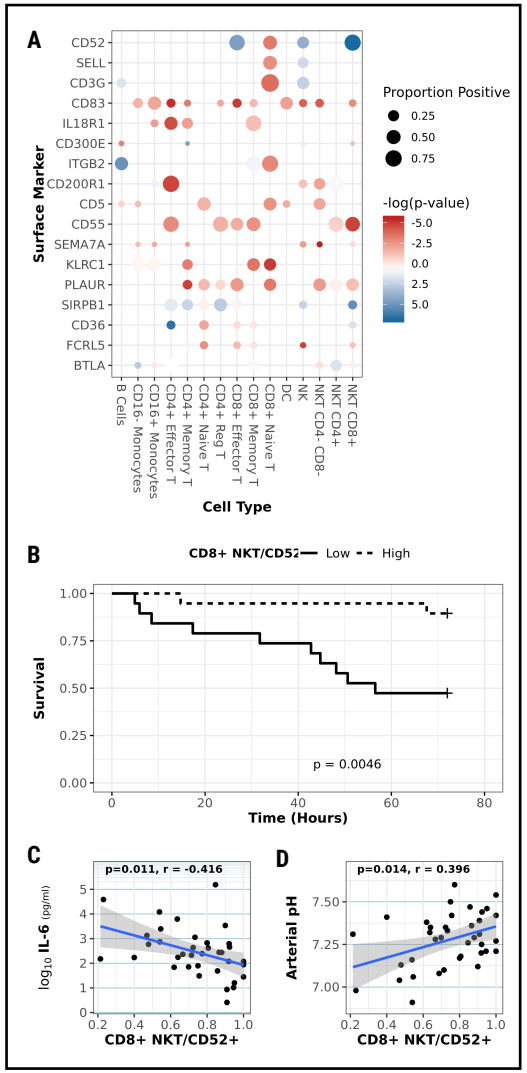
Since the processed data elements created below are save as RDS files in the chunks that create them, you can execute just some of the chunks and then pick up where you left off later.

### Pre-requisites

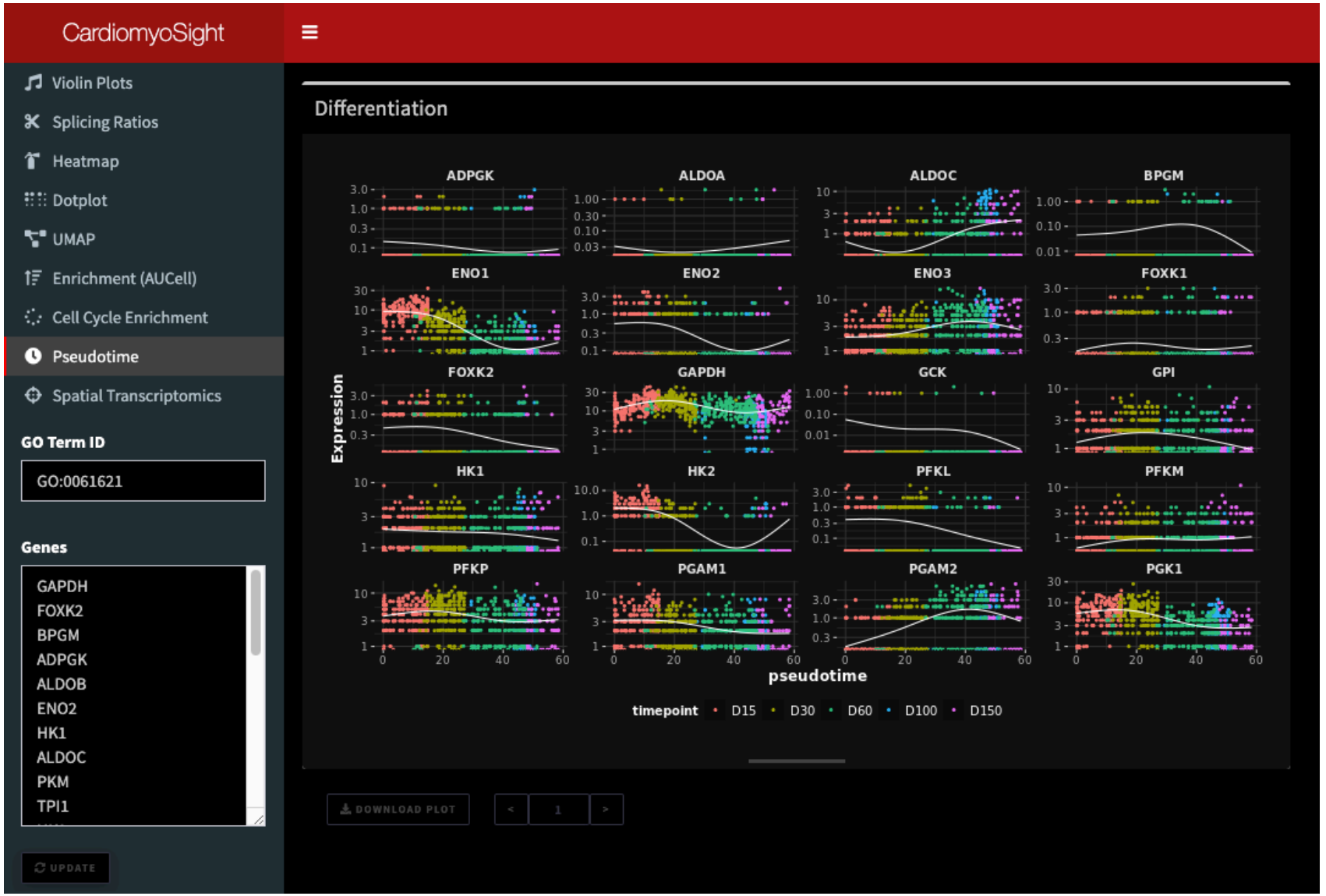
Running the following analysis requires a computer with R version >= 3.5.0 and 128GB of RAM, with `pandoc`, and development libraries for SSL, XML, and curl installed. This would be achieved on a debian flavored server as follows:

```
sudo -s
apt-get update
apt-get -y install pandoc
apt-get -y install libssl-dev
apt-get -y install libcurl4-openssl-dev
apt-get -y install libxml2-dev
```

To run the following analysis, the file `GSE127221_PBMC_merged_filtered_recoded.rds` must be obtained from the GEO series related to this study (GSE127221). The second panel of figure 4b requires the list of genes from Supplmental Table S5, which is available in the gitub repository as `Final_Data/table_s5.txt`. The following code assumes these files are within a subdirectory named



# Putting a stop to the “what about this gene” emails...



# A few parting thoughts

- Garbage In ≠ Garbage Out
  - Find the signal in the noise
- Transcriptomics is a first step on the journey of discovery
  - Perfect is the enemy of good
- Do not worship the UMAP

# Questions / Comments