HARVARD T.H. CHAN | SCHOOL OF PUBLIC HEALTH

# single-cell/DGE in bcbio

Rory Kirchner (roryk@alum.mit.edu)

3-13-2019

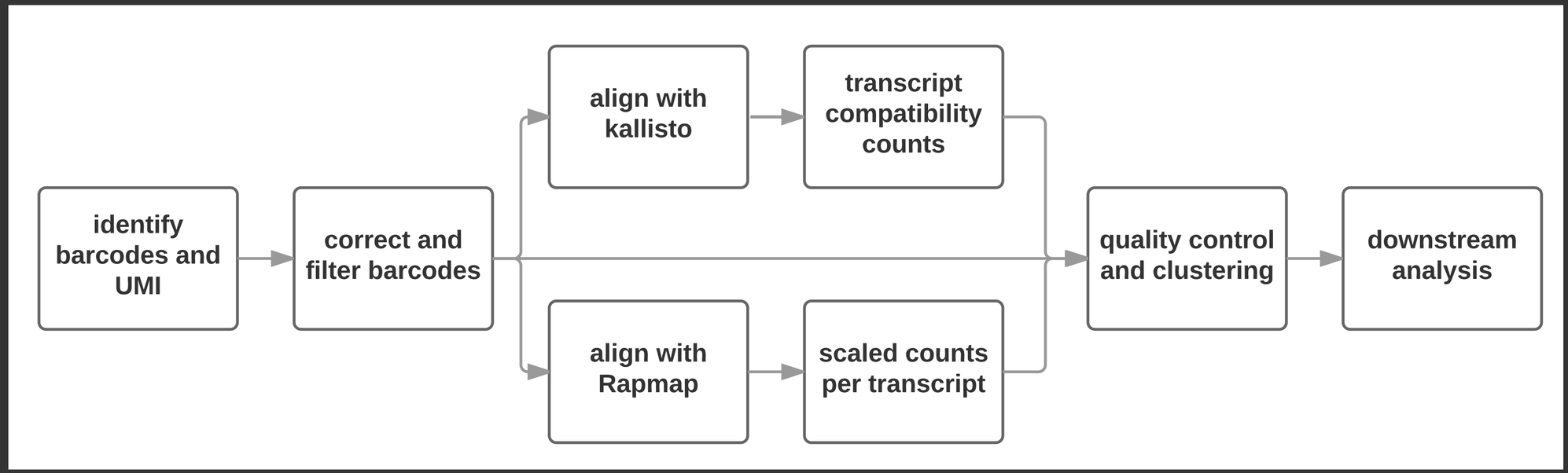
reasonably high signal to noise Twitter: @RoryKirchner

github: https://github.com/roryk/

# Overview

1. Preprocessing single-cell RNA-seq data
2. Implementation of single-cell RNA-seq quantification in bcbio
3. Quality control with particular focus on droplet based methods
4. Advice for performing differential expression
5. What is N in single-cell experiments?

# DGE/single cell pipeline



# umis

umis fastqtransform transform.json r1.fq r2.fq … rn.fq > transformed.fastq

umis cb_filter —bc1 known-barcodes.txt —nedit 1 transformed.fq > filtered.fastq

rapmap quasimap -1 filtered.fq -i index > alignment.bam

umis fasttagcount —genemap tx2gene.csv —bc1 known-barcodes.txt … —nedit 1 alignment.bam counts.csv

https://github.com/vals/umis

# Main complications of UMI

1. Across-technology variations in UMI/cellular/sample barcode specifications
2. Deduplicating UMIs during quantification
3. Absence of full-length transcript data is not a supported quantification scheme for many second generation expression callers

# HMS inDrop

biological read

cell barcode 1

sample barcode

cell barcode 2

UMI

```
==> klein-v3_R1.fq <==
@NS500233:572:H25VKBGX2:1:11101:16195:1041 1:N:0:1
GCTTTNCATGTTGTTTTGAAGGTTCCCACNGTNANCNTTCTTGTTNACNGNNNNNTTNNN
+
/AAAA#EEEEEEEEE<EEEEEEEEEEEEE#EE#E#/#EEEEEEEE#EA#/#####EE####
==> klein-v3_R2.fq <==
@NS500233:572:H25VKBGX2:1:11101:16195:1041 2:N:0:1
AGGGGGGG
+
/AA/A///
==> klein-v3_R3.fq <==
@NS500233:572:H25VKBGX2:1:11101:16195:1041 3:N:0:1
ATCGCCGG
+
AAAAAEA/
==> klein-v3_R4.fq <==
@NS500233:572:H25VKBGX2:1:11101:16195:1041 4:N:0:1
ATATNNNNNNNNNN
+
AAAA##########
```

# 10x (v2)

biological read

cell barcode 1

sample barcode

UMI

```
==> test_7_I1.fastq <==
@ST-K00126:314:HFYL2BBXX:7:1101:1631:1226 1:N:0:GTAATTGC
GTAATTGC
+
AAAFFJFJ
==> test_7_R1.fastq <==
@ST-K00126:314:HFYL2BBXX:7:1101:1631:1226 1:N:0:GTAATTGC
GGGCACTAGCTGATAAGGGCCCAACG
+
A-AFFJA-AAJ<FF-F<<F-7FJJJJ
==> test_7_R2.fastq <==
@ST-K00126:314:HFYL2BBXX:7:1101:1631:1226 2:N:0:GTAATTGC
GNTGTGGCAGAGCAGCGACCCGCGGCGGGGCGGCATCCCCAGCTGGTTCGGGCC
GGGACGGGGCGGCCAGCAGGGACGCGCCCCAGGGGGGCAGCTGT
+
A#-<<F7<AJF-FJ<JAAJJFJJ<AF-7AJF77<FJJJFFFJJ<JA-7-777<-F7<<F--7AA7AAAFF-
AF<A-AFFA7J7F--7)-)7--7A<J-
```

# Support all barcoding protocols

**10x (v2)**

```
"read1": "(?P<name>@.*) .*\\n(?P<CB>.{16})(?P<MB>.{10})(.*)\\n\\+(.*)\\n(.*)\\n",
"read2": "(@.*) .*\\n(?P<seq>.*)\\n\\+(.*)\\n(?P<qual>.*)\\n",
"read3": "(@.*)\\n(?P<SB>.*)\\n\\+(.*)\\n(.*)\\n"
}
```

**SureCell**

```
"read1": "(@.*)\\n(.*)(?P<CB1>.{6})TAGCCATCGCATTGC(?P<CB2>.
{6})TACCTCTGAGCTGAA(?P<CB3>.{6})ACG(?P<MB>.{8})GAC(.*)\\n\\+(.*)\\n(.*)\\n",
"read2": "(?P<name>@.*) .*\\n(?P<seq>.*)\\n\\+(.*)\\n(?P<qual>.*)\\n"
```

**CEL-Seq (v2)**

```
"read1": "(?P<name>@.*) .*UMI:(?P<MB>.{5,6}):.*\\n(?P<seq>.*)\\n\\+\\n(?
P<qual>.*)\\n"
```

**inDrop (v3)**

```
"read1": "(?P<name>[^\\s]+).*\\n(?P<seq>.*)\\n\\+(.*)\\n(?P<qual>.*)\\n",
"read2": "(.*)\\n(?P<CB1>.*)\\n(.*)\\n(.*)\\n",
"read3": "(.*)\\n(?P<SB>.*)\\n(.*)\\n(.*)\\n",
"read4": "(.*)\\n(?P<CB2>.{8})(?P<MB>.{6})(.*)\\n(.*)\\n(.*)\\n"
```

# whitelisted cellular barcodes

roryk / **singlecell-barcodes**

Watch 0 | Star 1 | Fork 2

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Insights | Settings

whitelisted singlecell barcodes and information regarding where molecular/sample/cellular barcodes are in each read, for various singlecell protocols

Edit

Manage topics

**13** commits | **1** branch | **0** releases | **1** contributor

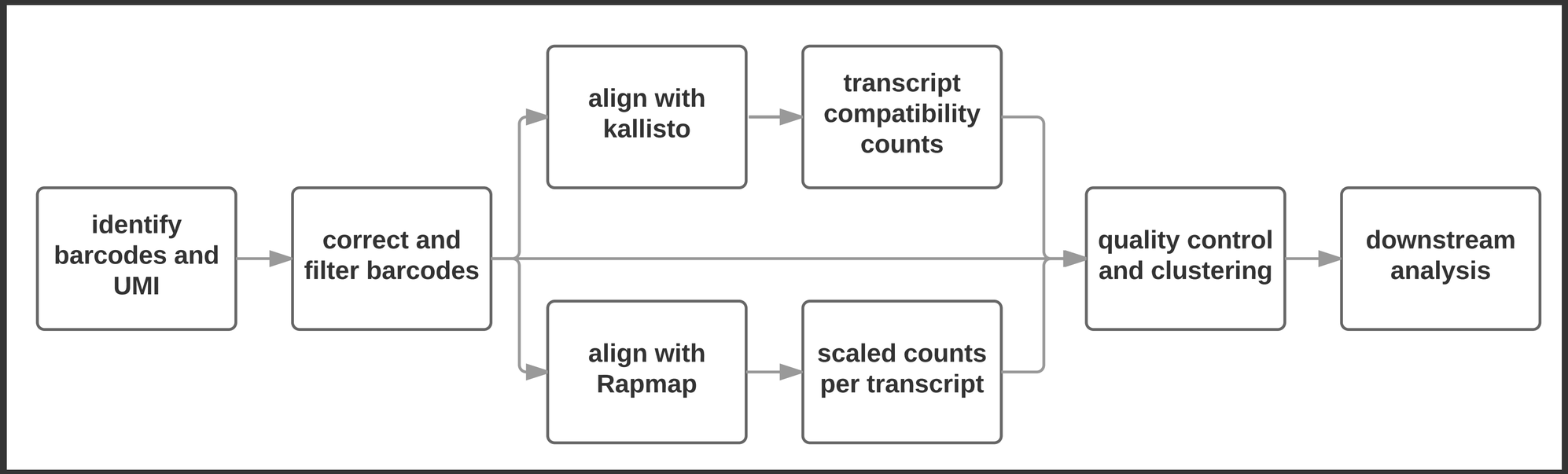
Branch: **master** | New pull request | Create new file | Upload files | Find File | Clone or download

**roryk** Add Lexogen DGE transformation. Latest commit `33eb3b0` 27 days ago

| | | |
|---|---|---|
| 10x | Change information regarding where 10x barcodes came from. | a year ago |
| 10x_v2 | Add 10x Chromium v2. | a year ago |
| harvard-indrop-v2 | Initial commit. | 2 years ago |
| harvard-indrop-v3 | Initial commit. | 2 years ago |
| harvard-scrb | Initial commit. | 2 years ago |
| lexogen-dge | Add Lexogen DGE transformation. | 27 days ago |
| missionbio | Add MissionBio transform. | a month ago |
| surecell | Initial commit. | 2 years ago |
| README.md | Fix grammar. | 2 years ago |
| VERSION | Bump version to 0.3. | a year ago |

**README.md**

# DGE/single cell pipeline



# Overview

1. Preprocessing single-cell RNA-seq data
2. Implementation of single-cell RNA-seq quantification in bcbio
3. Quality control with particular focus on droplet based methods
4. Advice for performing differential expression
5. What is N in single-cell experiments?

# umis

umis fastqtransform transform.json r1.fq r2.fq … rn.fq > transformed.fastq

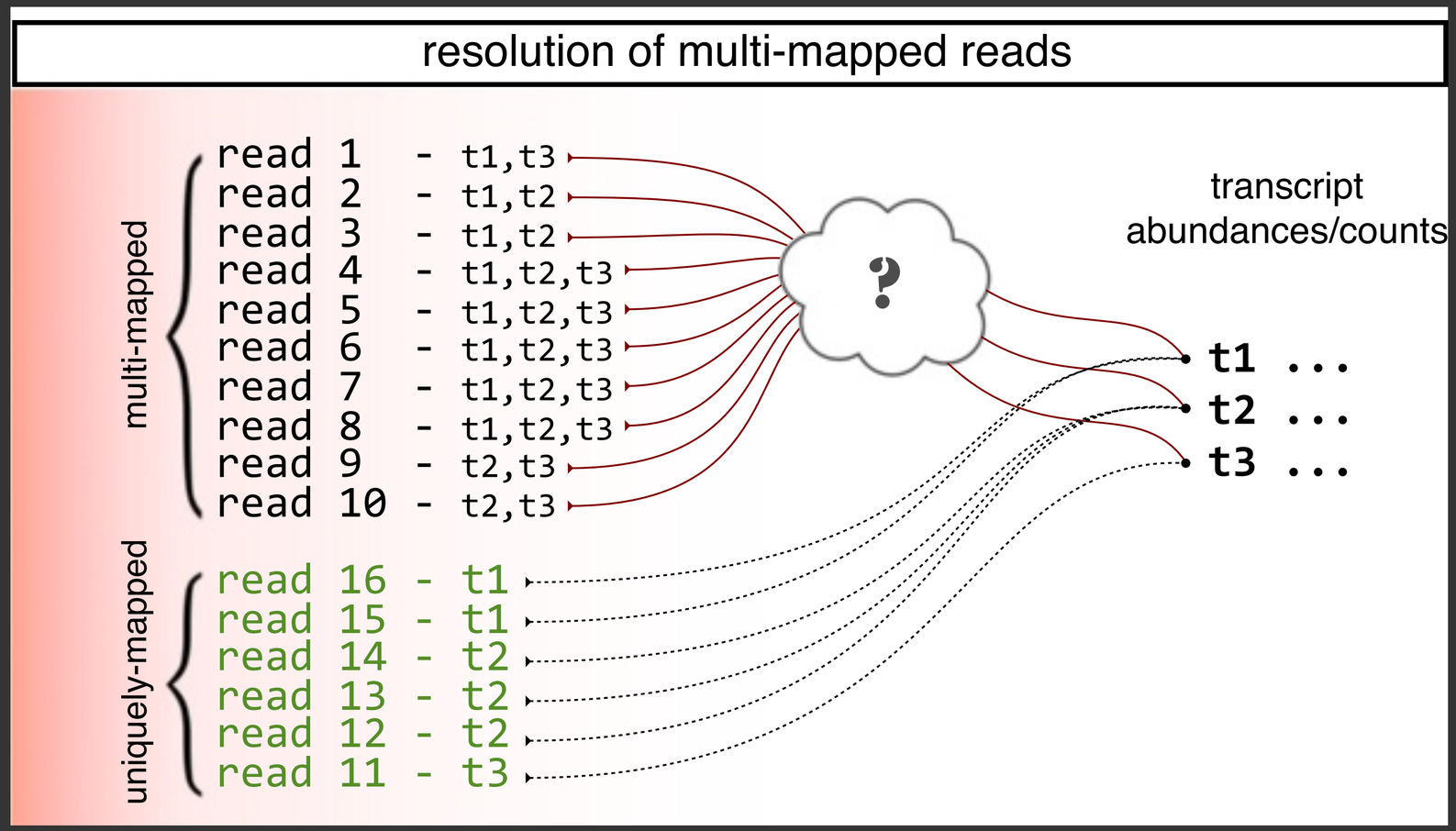
umis cb_filter —bc1 known-barcodes.txt —nedit 1 transformed.fq > filtered.fastq

rapmap quasimap -1 filtered.fq -i index > alignment.bam

umis fasttagcount —genemap tx2gene.csv —bc1 known-barcodes.txt … —nedit 1 alignment.bam counts.csv

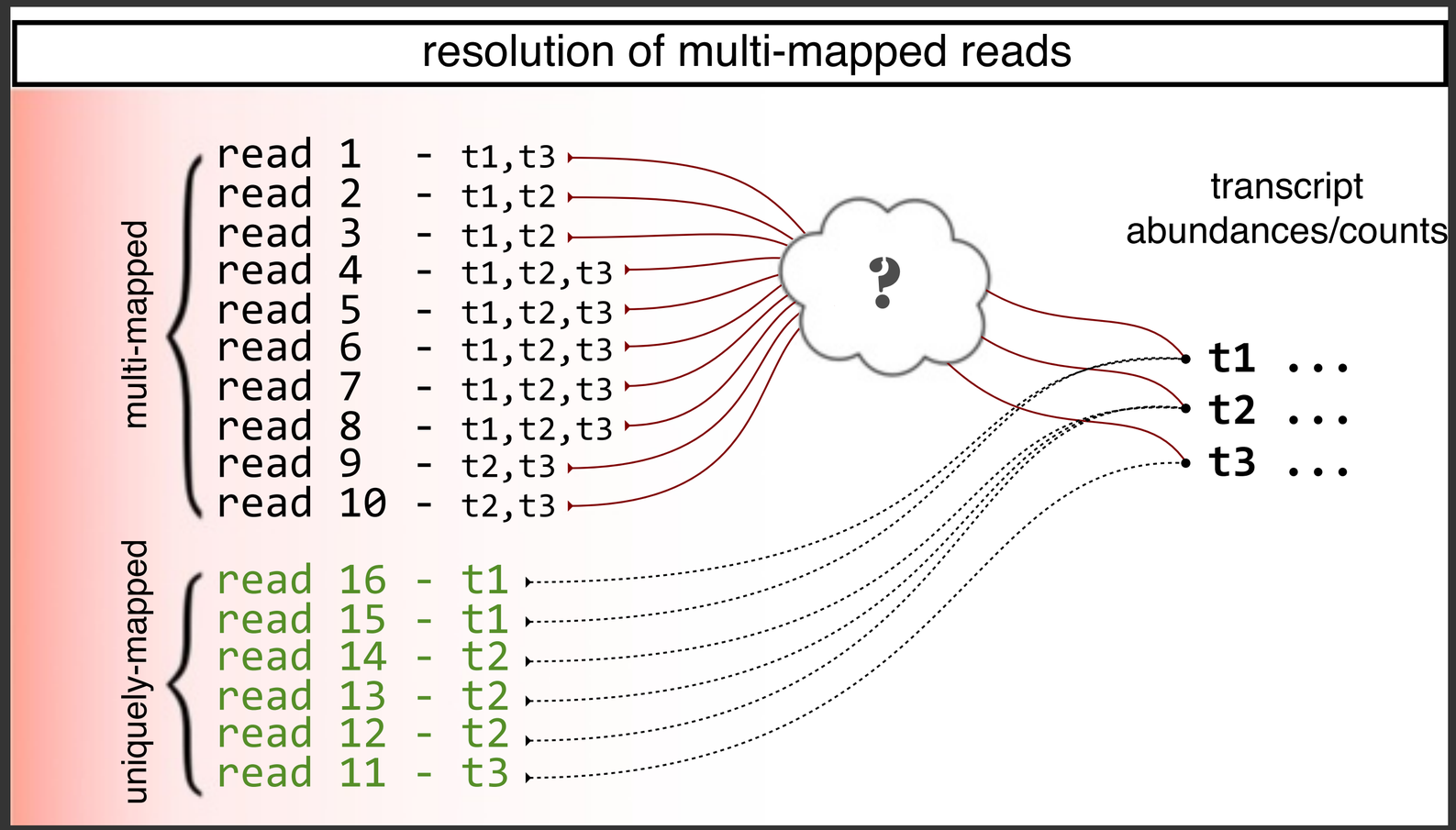
https://github.com/vals/umis

# quantification uncertainty




*Fast and accurate single-cell RNA-seq analysis by clustering of transcript-compatibility counts,* Genome Biology, 2016

# transcript compatibility counts



# Alevin

Alevin is a tool --- integrated with the salmon software --- that introduces a family of algorithms for quantification and analysis of 3' tagged-end single-cell sequencing data. Currently alevin supports the following two major droplet based single-cell protocols:

1. Drop-seq
2. 10x-Chromium v1/2/3

Alevin works under the same indexing scheme (as salmon) for the reference, and consumes the set of FASTA/Q files(s) containing the Cellular Barcode(CB) + Unique Molecule identifier (UMI) in one read file and the read sequence in the other. Given just the transcriptome and the raw read files, alevin generates a cell-by-gene count matrix (in a fraction of the time compared to other tools).

Alevin works in two phases. In the first phase it quickly parses the read file containing the CB and UMI information to generate the frequency distribution of all the observed CBs, and creates a lightweight data-structure for fast-look up and correction of the CB. In the second round, alevin utilizes the read-sequences contained in the files to map the reads to the transcriptome, identify potential PCR/sequencing errors in the UMIs, and performs hybrid de-duplication while accounting for UMI collisions. Finally, a post-abundance estimation CB whitelisting procedure is done and a cell-by-gene count matrix is generated.

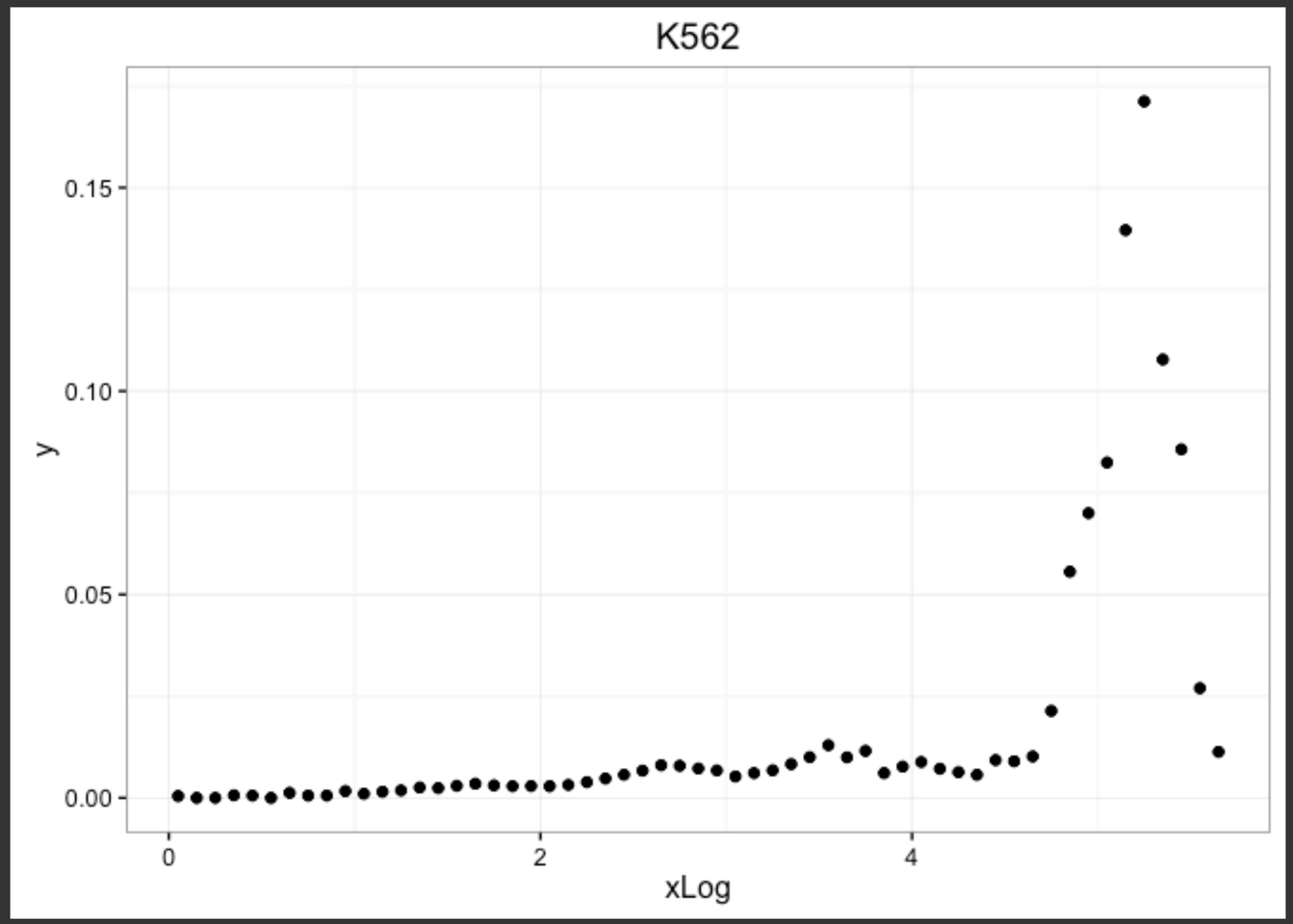
# Overview

1. Preprocessing single-cell RNA-seq data
2. Implementation of single-cell RNA-seq quantification in bcbio
3. Quality control with particular focus on droplet based methods
4. Advice for performing differential expression
5. What is N in single-cell experiments?

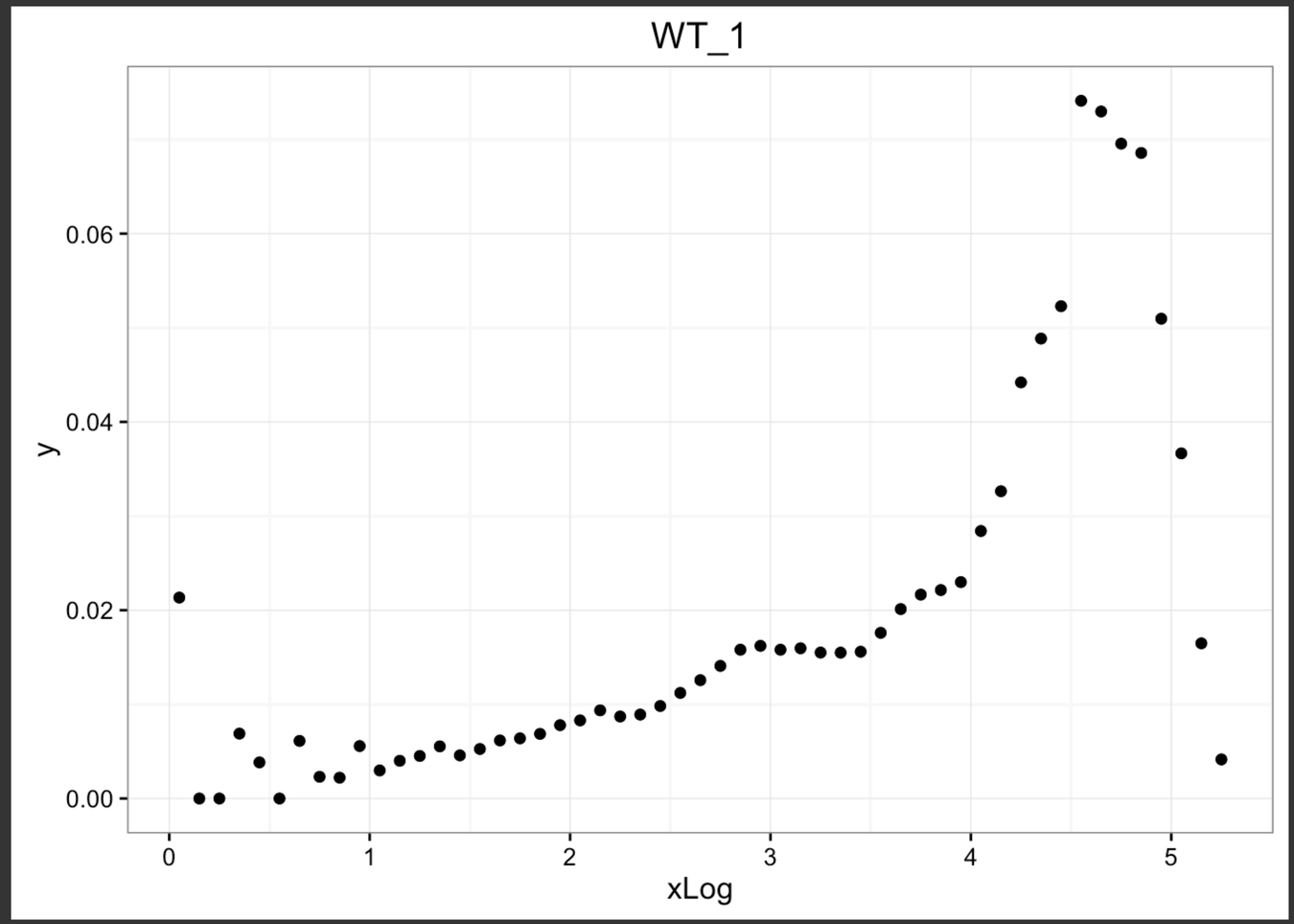
# quality control

- focus on getting high quality data, messy data is EXTREMELY hard to work with
- cells must be alive, before running the experiment shoot for viability > 95%
- when filtering, it is better to start out too strict than too lax
- essential to work closely with the biologists, many, many judgement calls need to be made based on expert knowledge
- before beginning analysis, have a good list of marker genes in hand for each cell type that could be present in the sample
- often markers used for FACS sorting are not good for single-cell RNA-seq

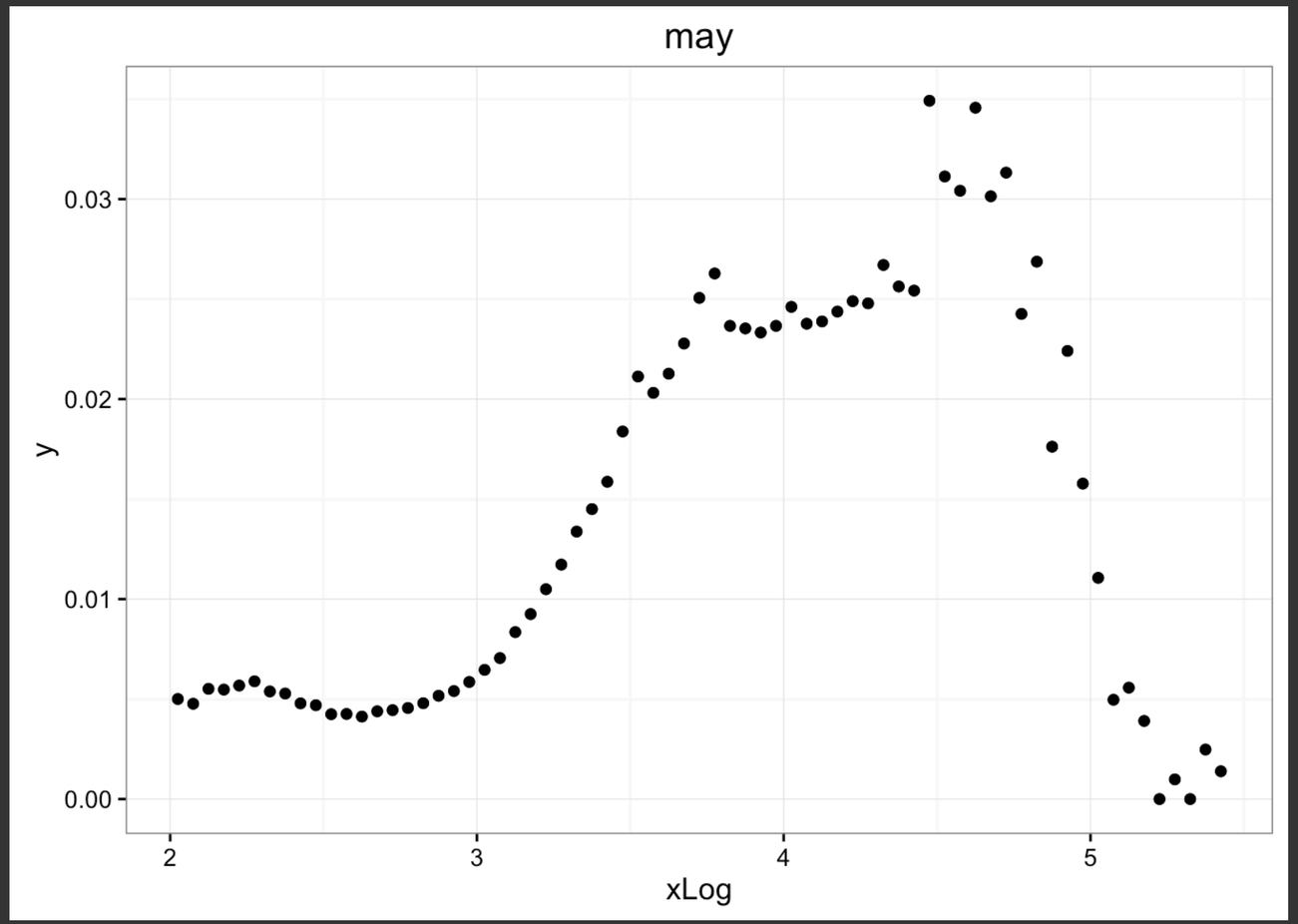
# cellular barcode histogram



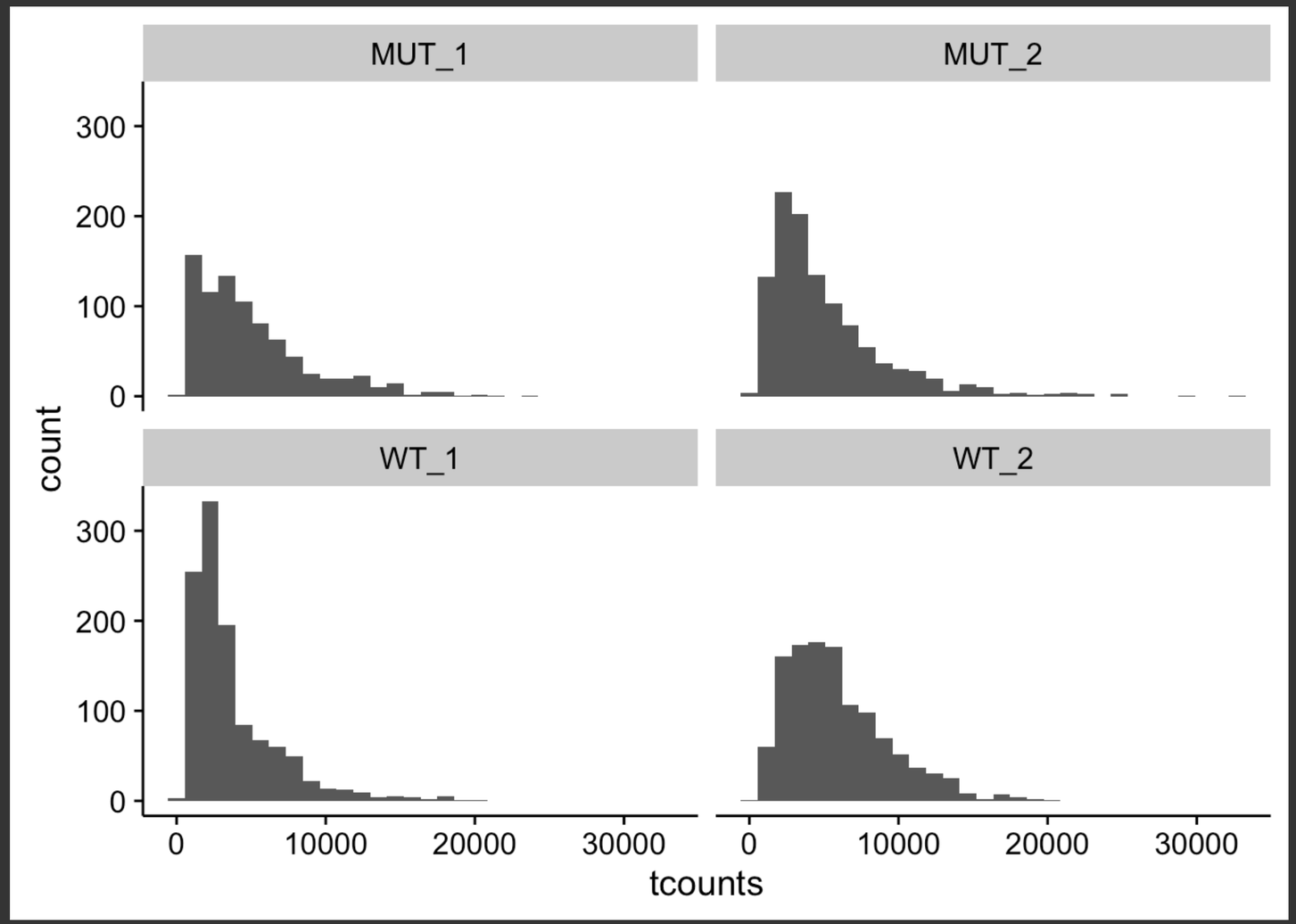
# zebrafish PMBC cells



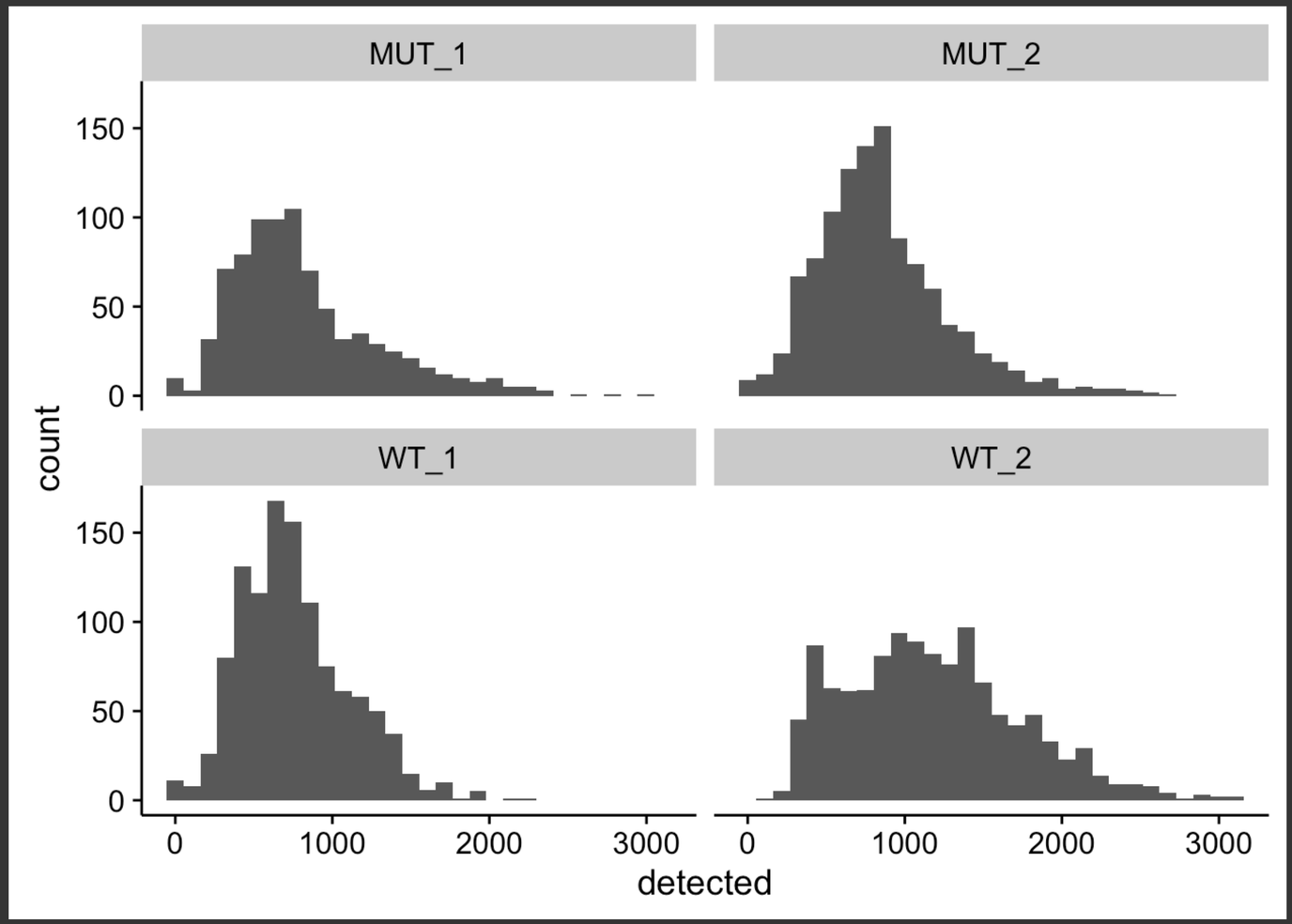
# free-floating RNA contamination



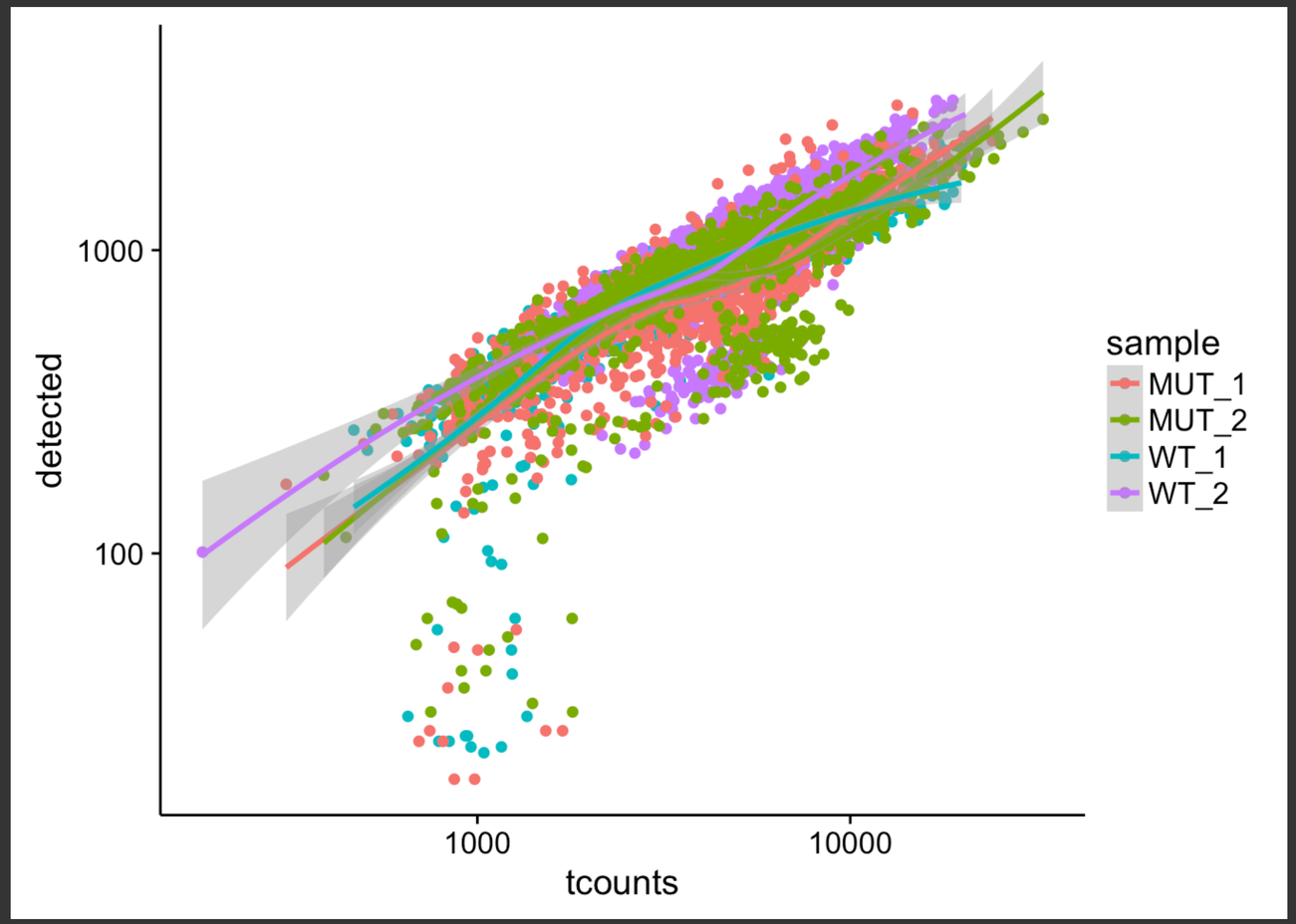
# total transcript counts per cell



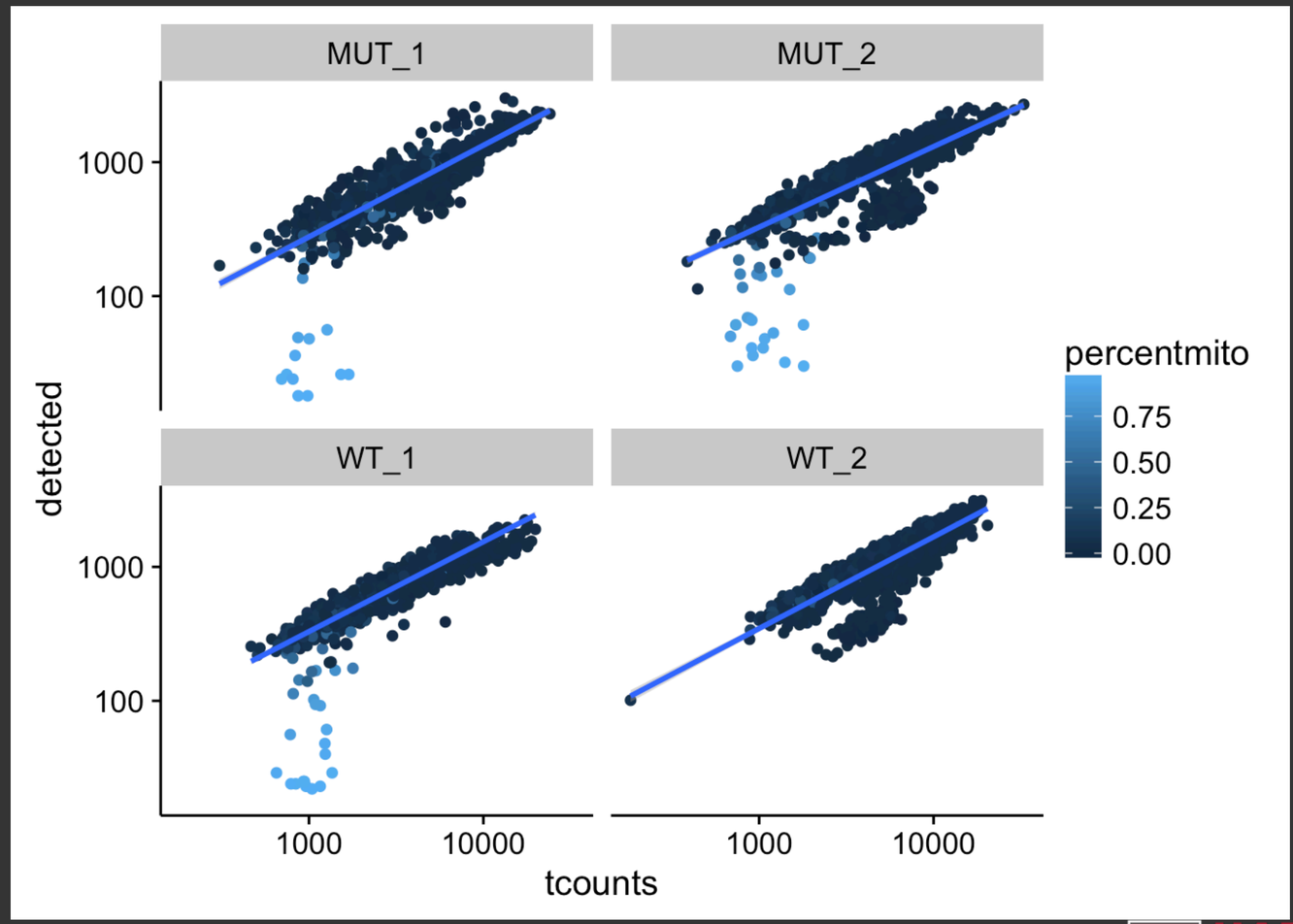
# genes detected per cell



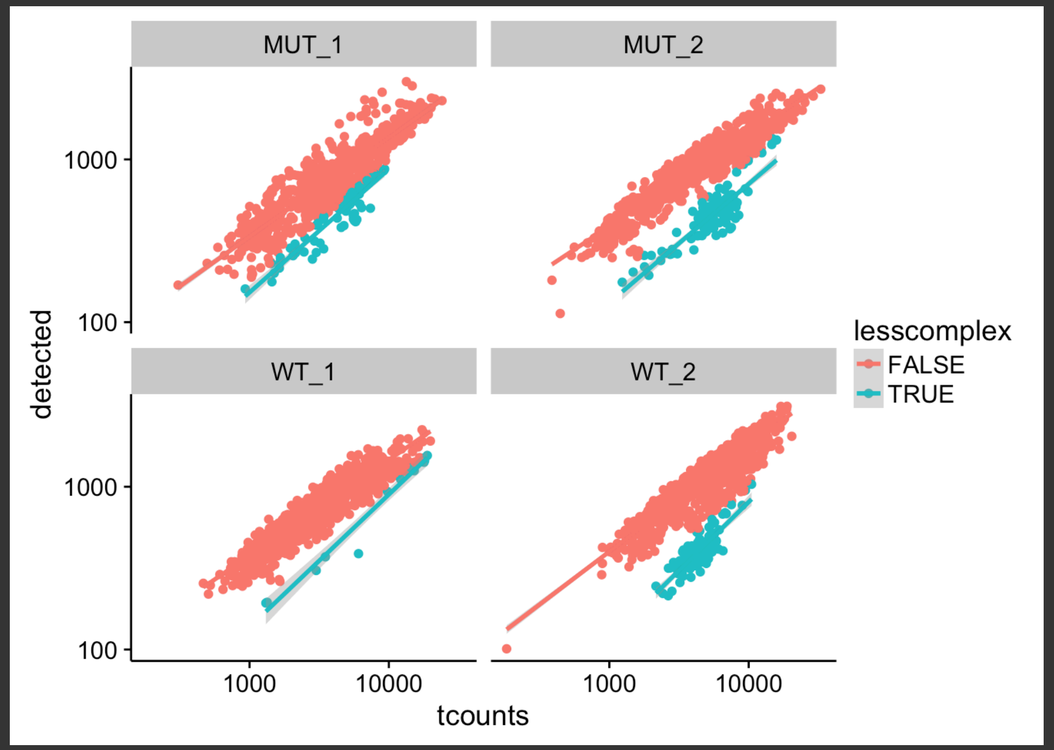
# total counts vs genes detected

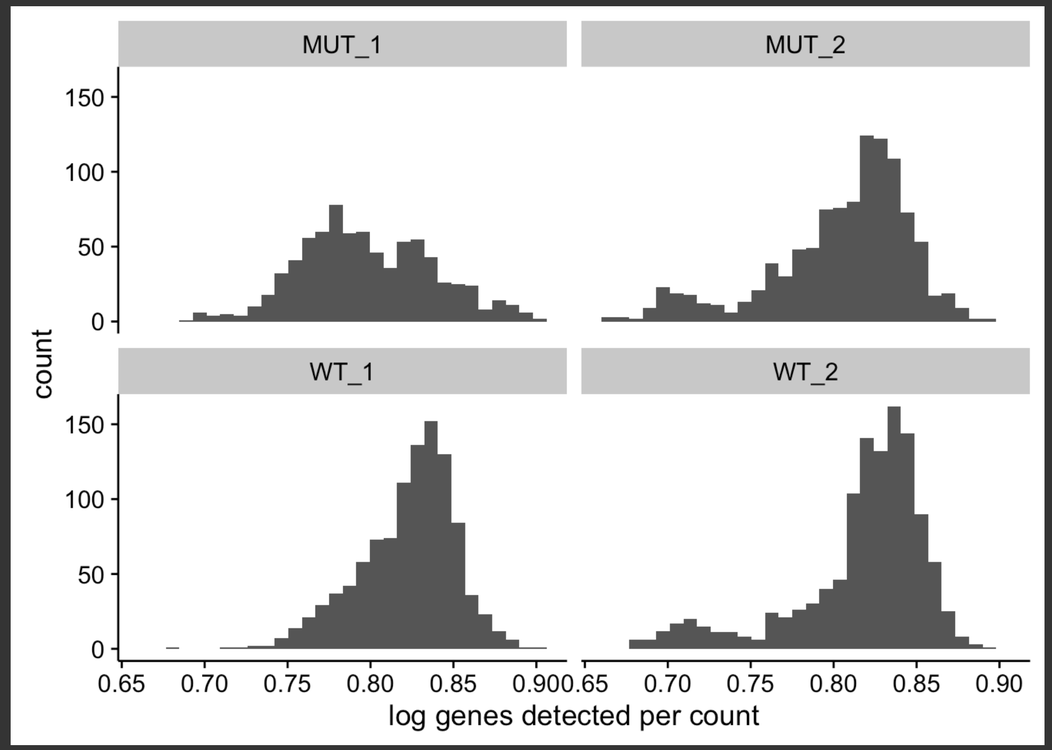


# high mitochondria indicates dying cells



# low complexity cells

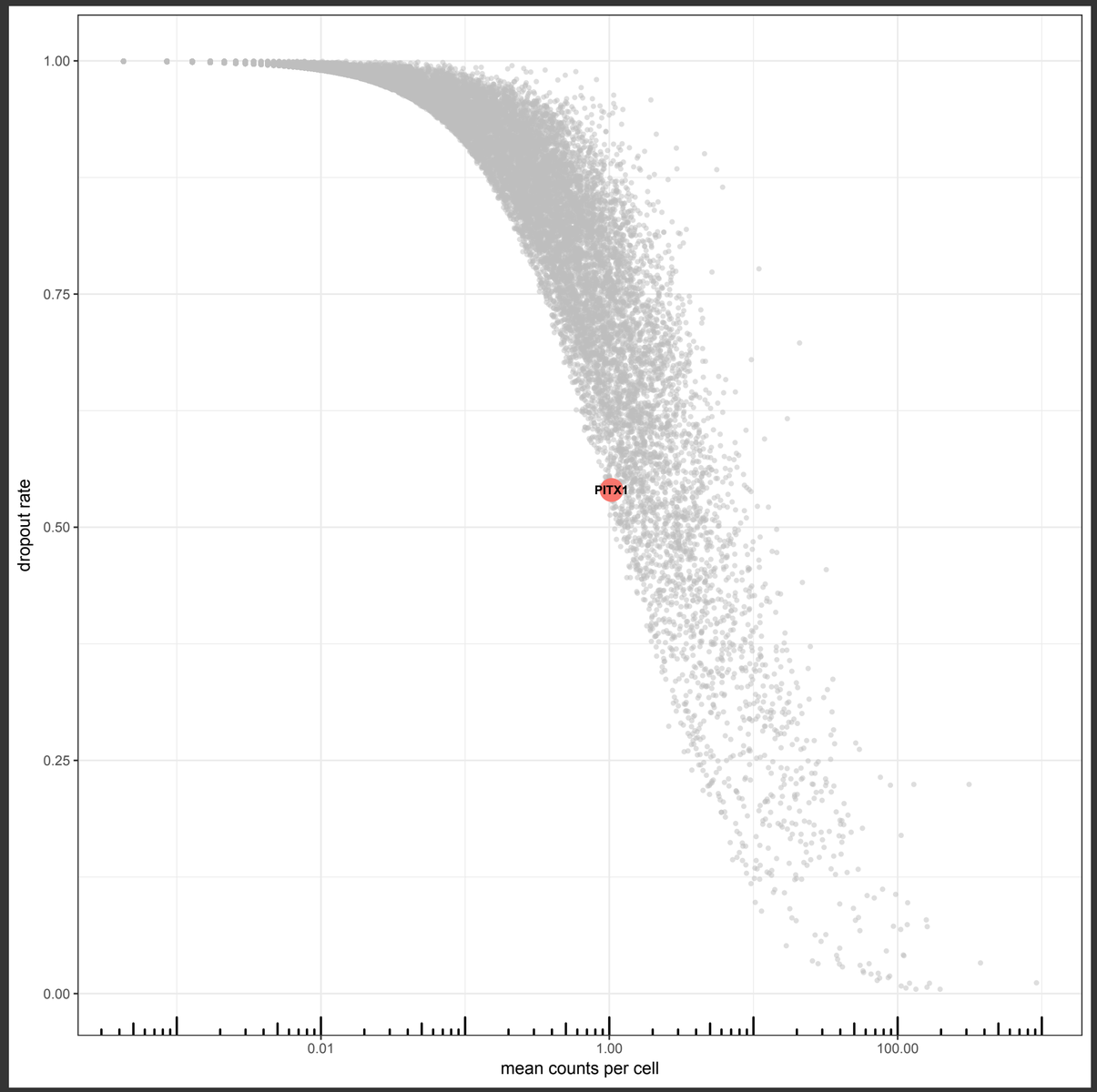


# filtering and correction

- remove cells with high mitochondrial RNA
- remove cells with abnormally low genes detected
- correct mitochondrial RNA percentage
- correct genes detected
- filter low complexity

# Differential expression: dropouts



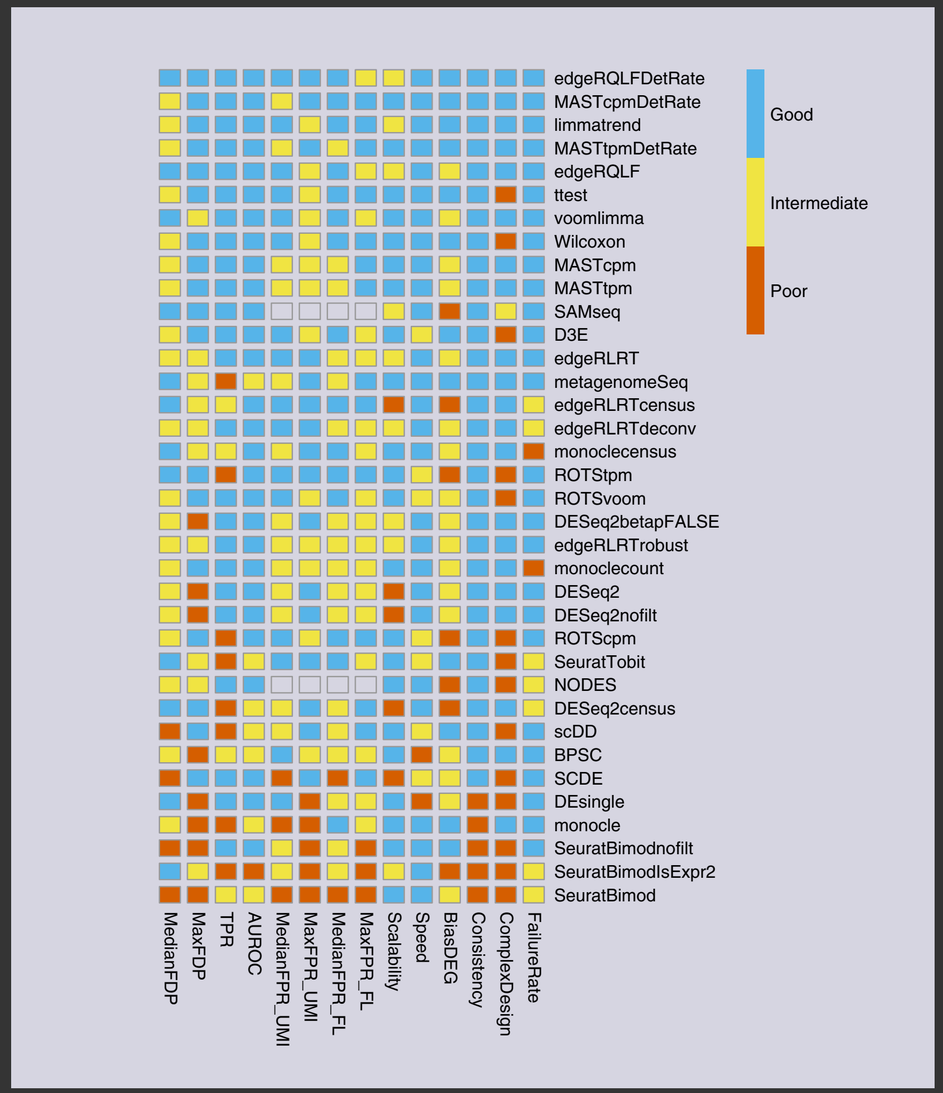
# Zero inflated models from ecology

- counting tigers and elephants
  - if an elephant exists in a quadrant, it will be seen and counted
  - if no elephants are seen in a quadrant there are no elephants in the quadrant
  - if a tiger exists in a quadrant, it may or may not be seen since they blend in
  - if no tigers are seen in a quadrant there may be tigers in the quadrant, but are missed
- account for that with a zero-inflated model

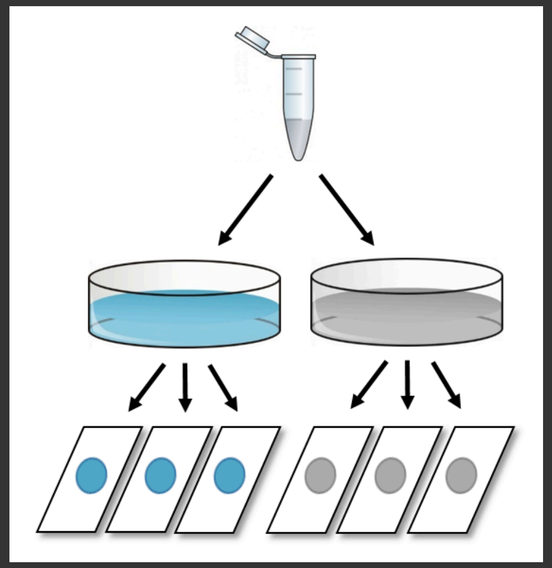
# single-cell RNA-seq zero inflated models

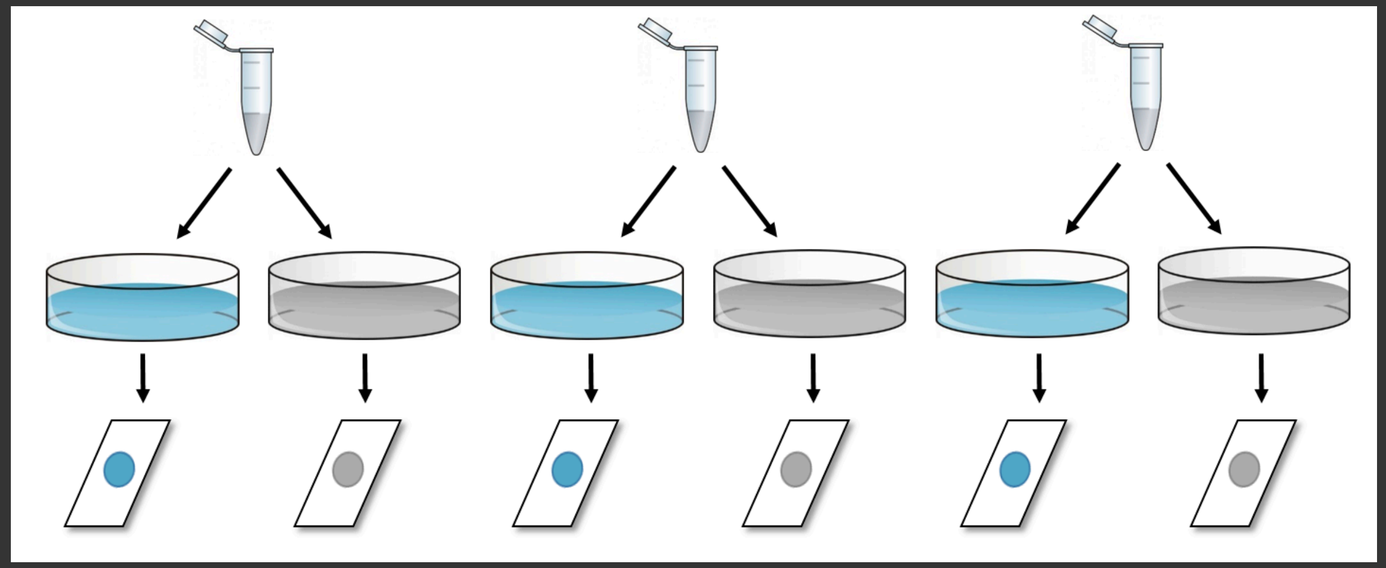
- many bulk RNA-seq differential expression callers model the expression of a gene by the negative binomial distirbution
- there is a zero inflrated version of the negative binomial
- R package zinbwave implements this for single-cell RNA-seq
- Another type of zero inflated model is a hurdle model
  - combine two models, a negative binomial model and a hurdle component of the model that accounts for the zero inflation. SCDE is an example of a package that uses this

# Differential expression: ignore all that



# What is N in cell-culture experiments?





Stanley Lazic: What is 'N' in cell culture and animal experiments? PLoS Biol. 2018 Apr; 16(4): e2005282.

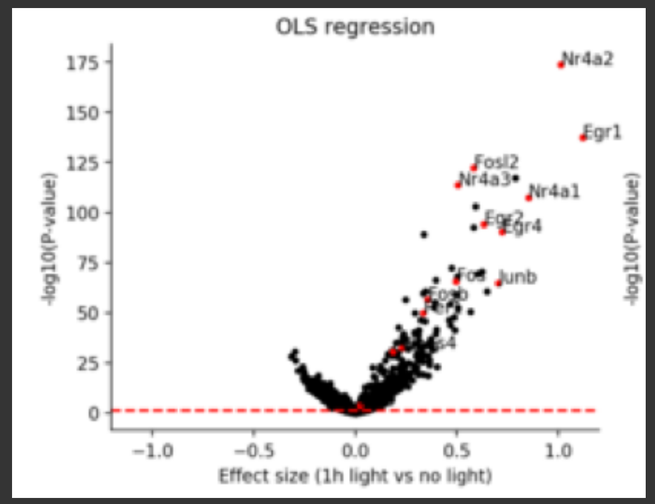
# What is N in single cell experiments?

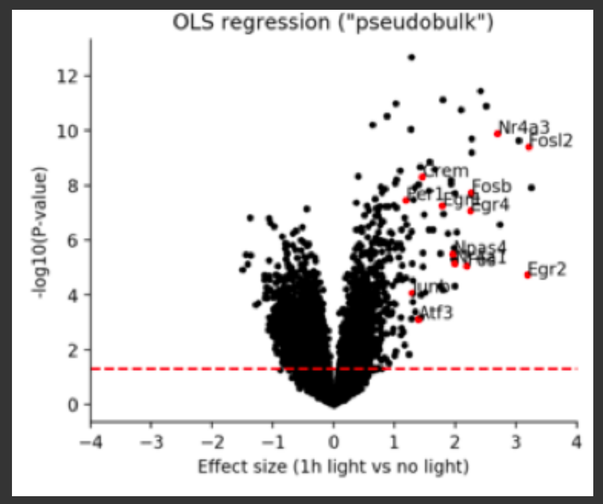
- Three treated patients, three control patients.
- Extract PBMC and want to look at the effect of the treatment on B cells.
- Indentified B-cell clusters in the treated and non-treated patients via marker genes and found 300 B cells in each patients for a total of 900 B cells in each treatment condition
- If I want to ask what the effect of the treatment is on B-cells, what is my N here? Is it 900 for each condition? No. But almost all single-cell papers to date (including mine!) treat it as if it is.
- N should be 3, not 900.

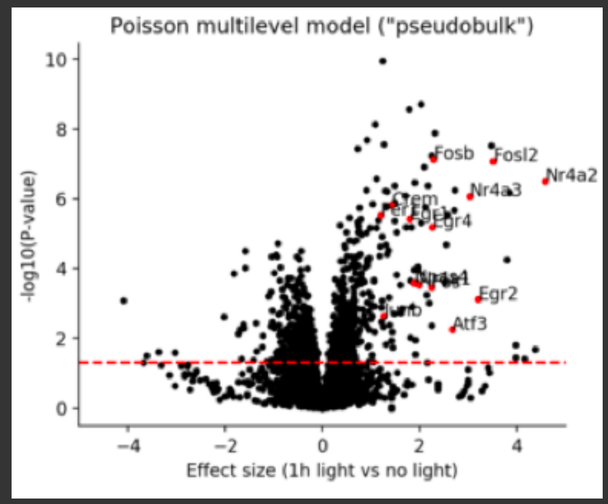
# How to get to N=3?

- pseudobulk: sum all B-cells for each sample, and treat it like an in-silico FACS sorted experiment
- multilevel model: model the patient level data in a multilevel model so you can account for the non-independence of measurements from B-cells of the same patient

# pseudobulk is simple and works well




http://www.nxn.se/valent/2019/2/15/handling-confounded-samples-for-differential-expression-in-scrna-seq-experiments (Valentine Svensson, via our twitter discussion)

# Things to work on together

- bcbio is community developed, and improvements from the community is how we get better
- if you are getting started with single cell, it would be awesome for someone to give Alevin a whirl on 10x data and report back if it works compared to Cellranger for example
    - I'm super interested in adding arbitrary support for single-cell protocols to Alevin, which we could work with Rob's group to do
- We've added support for single-cell DNA seq, would be good to have more folks working on improving variant calling for that
- explore multilevel modeling using negative binomial distribution instead of poisson