Cell Hashing

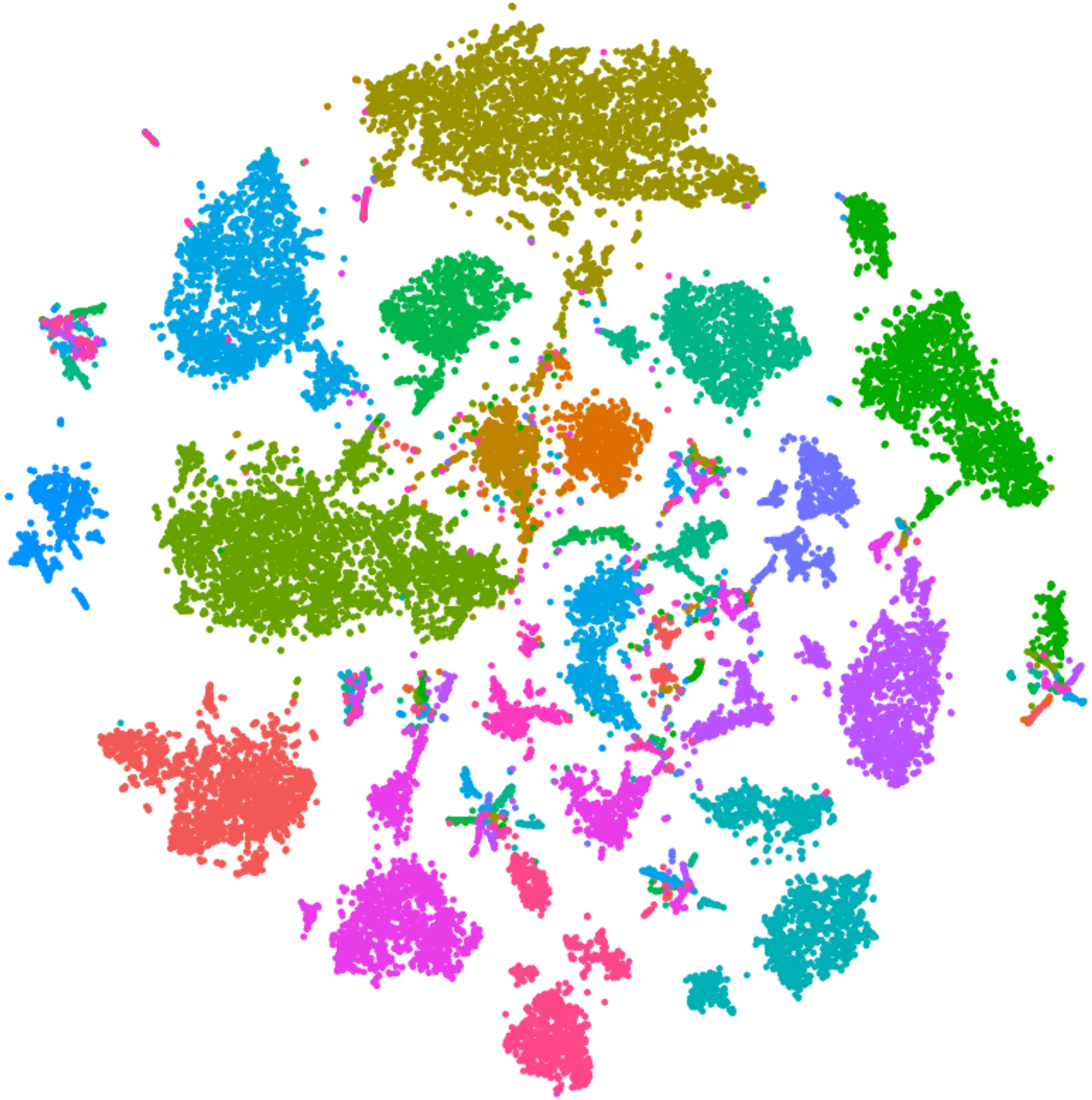
Cihan Oguz

Bioinformatics Analyst

NIAID Collaborative Bioinformatics Resource (NCBR)

Leidos Biomedical Research, Inc.

October 10, 2019



The term “cell hashing” originates from computer science

- A hash is a function that converts one value to another.

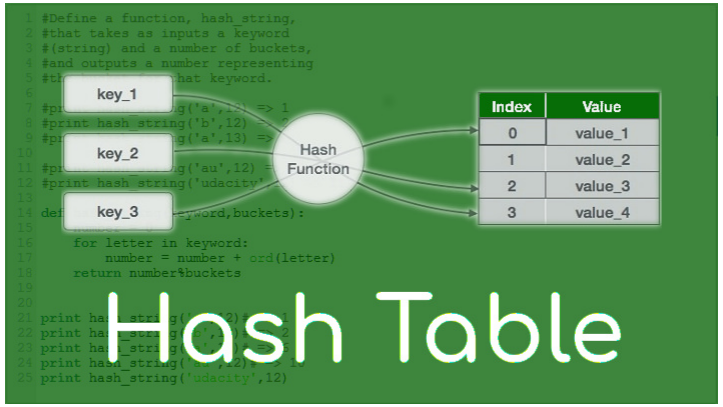
- Hashing data is common practice in computer science.

- “Cell Hashing” is based on the concept of using hash functions to index datasets with specific features.

> Cell Hashing uses oligo-tagged antibodies against highly expressed surface proteins to place a “sample barcode” on each single cell, enabling different samples to be multiplexed together and run in a single experiment.

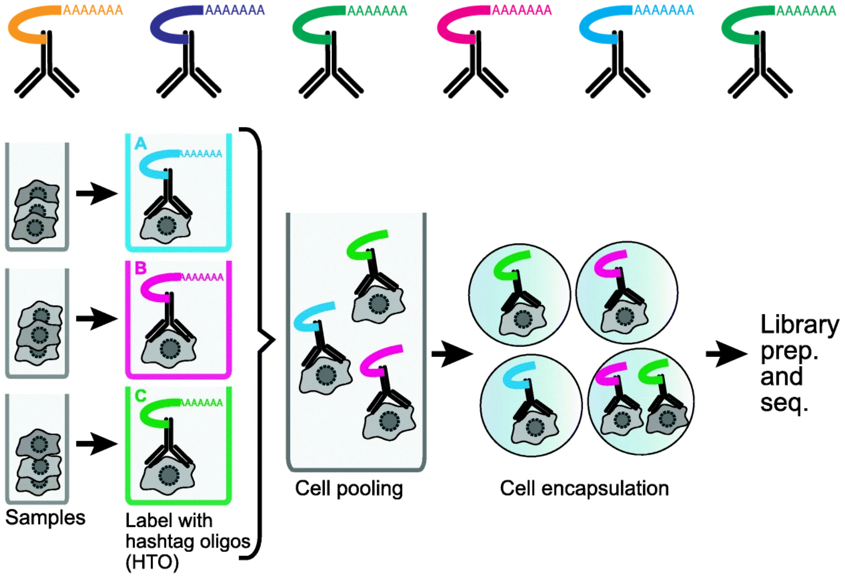
- Hashtags define a “lookup table” that assign each multiplexed cell to its original sample (e.g, KO vs WT or STIM vs CONT) by converting the detection of cell surface proteins into a sequenceable readout.

- Cell hashing enables “super-loading” commercial droplet-based systems (significantly higher cell concentration than usual).



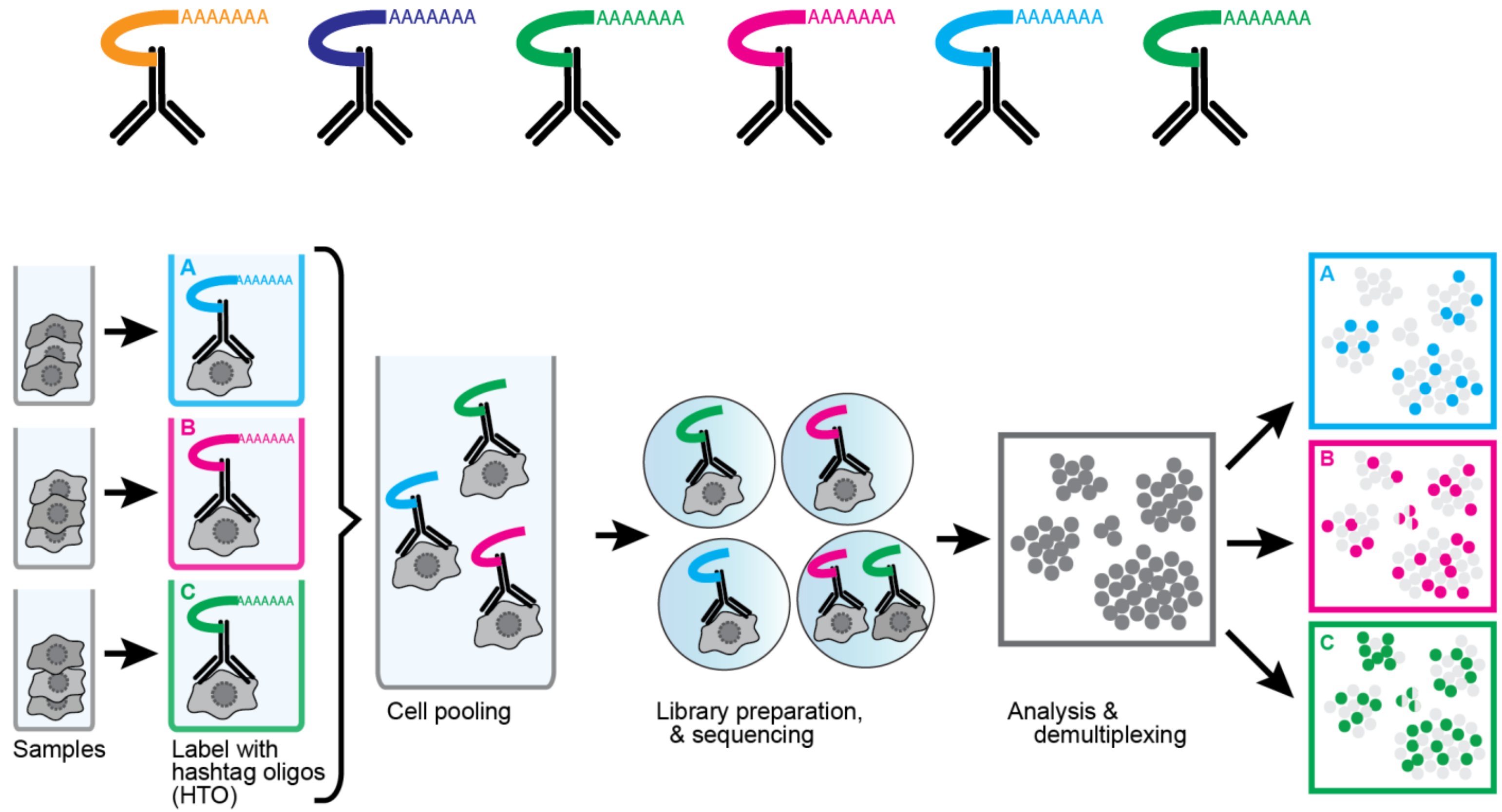
```
#Define a function, hash_string,
#that takes as inputs a keyword
#(string) and a number of buckets,
#and outputs a number representing
#th                    hat keyword.

#pr                g('a',12) => 1
#print hash_string('b',12) =>
#pr                g('a',13) =>

#pr                g('au',12) =
#print hash_string('udacity',

def                    eyword,buckets):

    for letter in keyword:
        number = number + ord(letter)
    return number%buckets


print has  stri g(   ,12)#   1
print has  stri g( b',1 )#=> 2
print has  stri g( a', )# => 5
print hash_string( au ,12)# => 10
print hash_string('udacity',12)
```

Index	Value
0	value_1
1	value_2
2	value_3
3	value_4



Stoeckius et al. Cell hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome biology. 2018 Dec;19(1):224.

Cell Hashing with barcoded antibodies


AAAAAAA
AAAAAAA
AAAAAAA
AAAAAAA
AAAAAAA
AAAAAAA
A
B
C
Samples
Label with hashtag oligos (HTO)
Cell pooling
Library preparation, & sequencing
Analysis & demultiplexing
A
B
C

- Multiplexing mitigates batch effects that can mask the biological signal in the integrated analysis of multiple scRNA-seq experiments.

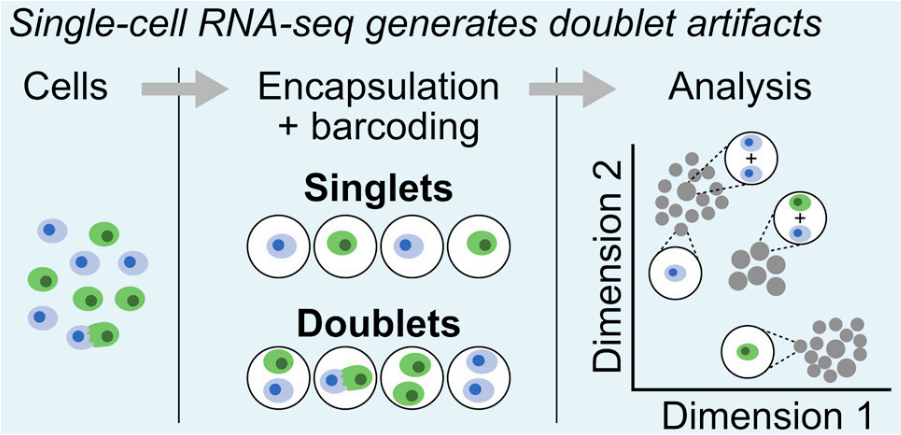
- Multiplexing achieves increased experimental throughput while reliably identifying multiplets (expression profiles corresponding to more than one cell).

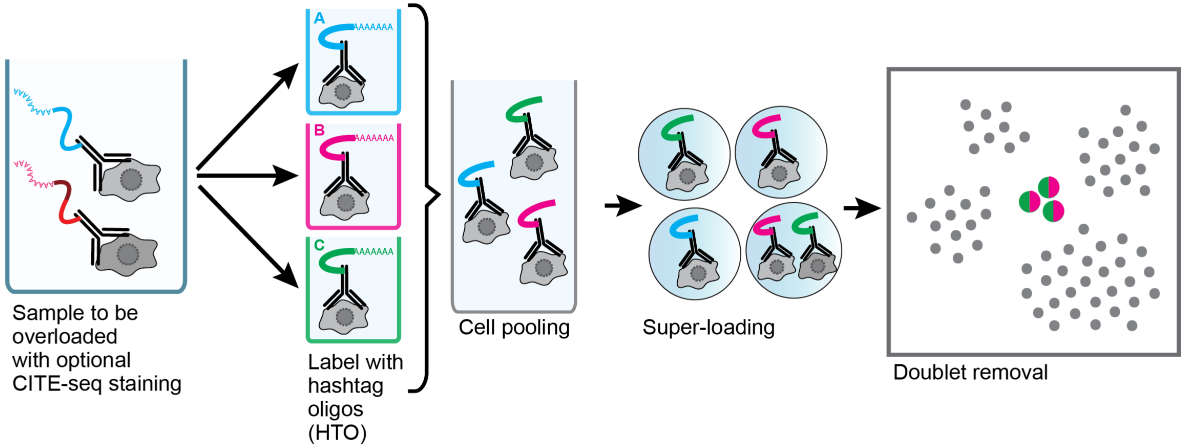
- Multiplets are expected to generate higher complexity libraries (more UMIs detected) compared to singlets.

- The strength of this signal is not always sufficient for unambiguous multiplet identification.

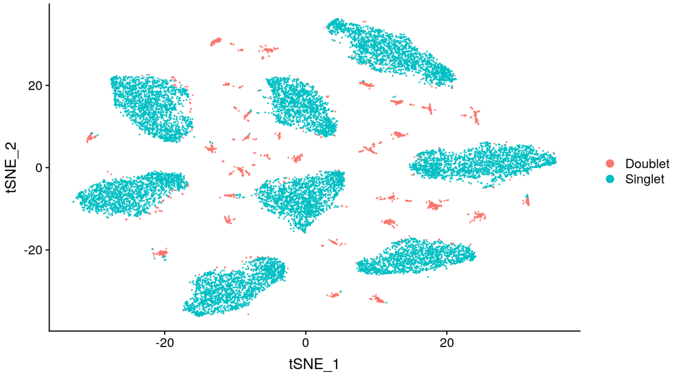
Cell hashtags allow for robust sample multiplexing, confident multiplet identification, and discrimination of low-quality cells from ambient RNA.

Doublets/multiplets are present in single cell data


Single-cell RNA-seq generates doublet artifacts
Cells
Encapsulation + barcoding
Analysis
Singlets
Doublets
Dimension 2
Dimension 1


Sample to be overloaded with optional CITE-seq staining
A
B
C
AAAAAAA
Label with hashtag oligos (HTO)
Cell pooling
Super-loading
Doublet removal

- scRNA-seq technologies co-encapsulate cells and barcoded primers in a small reaction volume (droplets or wells).
- mRNA molecules in each cell have unique DNA barcodes.
- Multiplets arise when two or more cells are captured within the same reaction, generating a hybrid transcriptome (per barcode).
- Multiplets can impact downstream analysis of scRNA-seq data (detecting intermediate cell states not actually present in the samples before sequencing).

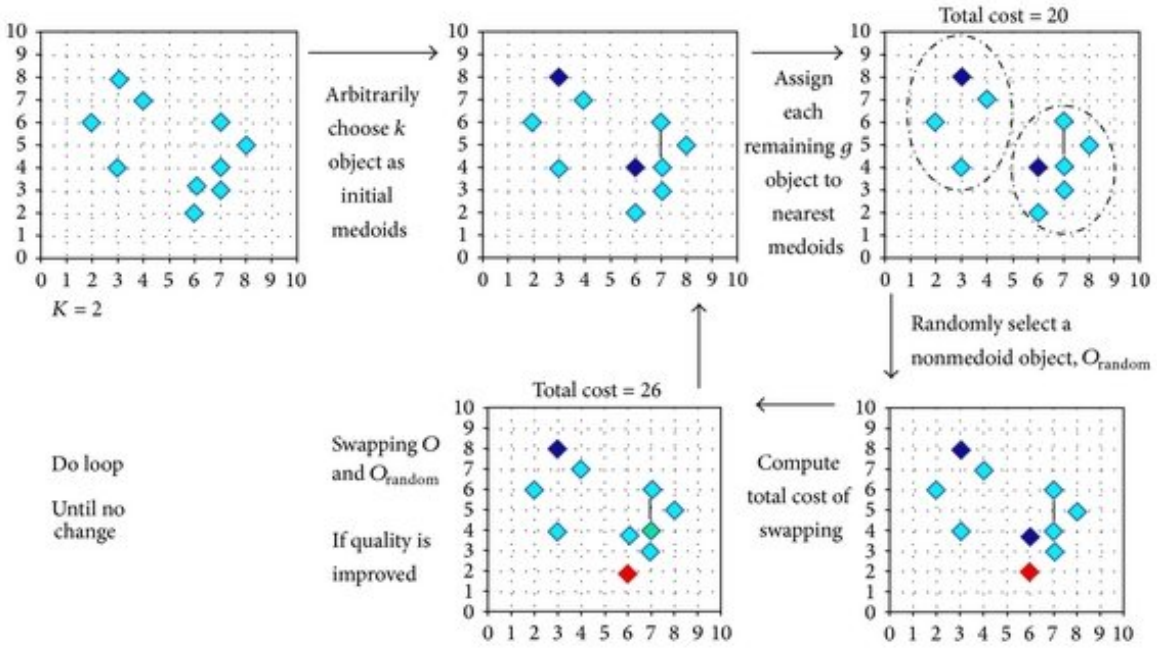


Cell Hashing enables robust identification of doublets originating from multiple samples.

Wolock et al. Scrublet: computational identification of cell doublets in single-cell transcriptomic data. Cell systems. 2019 Apr 24;8(4):281-91.
Stoeckius et al. Cell hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome biology. 2018 Dec;19(1):224.

Demultiplexing and doublet detection using k-medoids clustering

Frederick National Laboratory for Cancer Research



K = 2
Arbitrarily choose k object as initial medoids
Total cost = 20
Assign each remaining g object to nearest medoids
Randomly select a nonmedoid object, O_random
Compute total cost of swapping
Total cost = 26
Swapping O and O_random
If quality is improved
Do loop
Until no change
0 1 2 3 4 5 6 7 8 9 10

K-medoids clustering finds k data points (medoids) such that the total cost (distance) between each data point and the closest medoid is minimal.

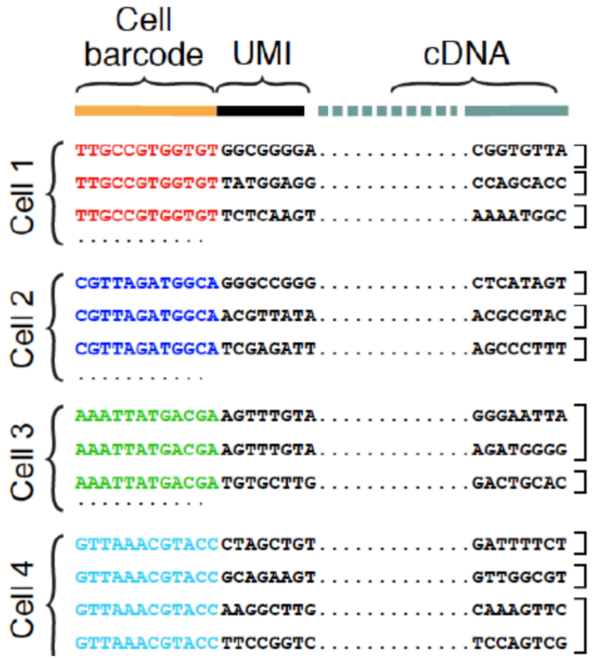
Barcodes positive for only one HTO are classified as singlets.

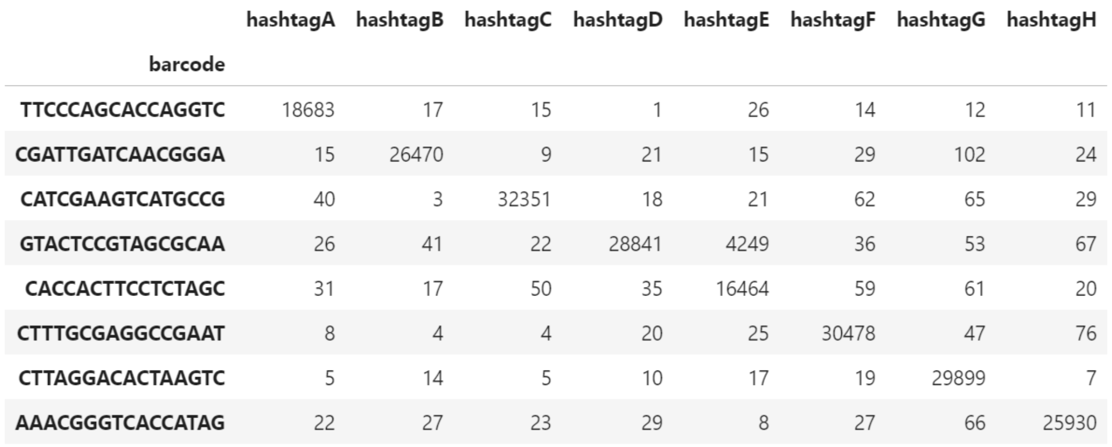
Barcodes negative for all eight HTOs classified as "negative."

Barcodes positive for two or more HTOs classified as multiplets (assigned sample IDs based on the top expressed HTOs).

- Cells with background expression for each HTO are called "negative cells", remaining cells have positive signal for at least one HTO.
- Use k-medoid clustering on the normalized HTO values to separate cells into k (# of samples)+1 clusters.
- Each cluster enriched for normalized expression of one HTO.
- Remaining cluster highly enriched for cells with low expression of all HTOs (negative cells).
- After initial clustering, the following is repeated independently for each HTO.
  - Identify the k-medoids cluster with the highest average HTO expression (excluded these cells from following steps).
  - Fit a negative binomial distribution to remaining HTO values (after removing the highest 0.5% values as potential outliers).
  - Calculate the q = 0.99 (or lower) quantile of the fitted distribution and threshold each cell in the dataset based on this HTO-specific value.

Seurat workflow for demultiplexing and doublet detection


Cell barcode
UMI
cDNA
Cell 1
TTGCCGTGGTGT GGCGGGGA.............CGGTGTTA
TTGCCGTGGTGT TATGGAGG.............CCAGCACC
TTGCCGTGGTGT TCTCAAGT.............AAAATGGC
Cell 2
CGTTAGATGGCA GGGCCGGG.............CTCATAGT
CGTTAGATGGCA ACGTTATA.............ACGCGTAC
CGTTAGATGGCA TCGAGATT.............AGCCCTTT
Cell 3
AAATTATGACGA AGTTTGTA.............GGGAATTA
AAATTATGACGA AGTTTGTA.............AGATGGGG
AAATTATGACGA TGTGCTTG.............GACTGCAC
Cell 4
GTTAAACGTACC CTAGCTGT.............GATTTTCT
GTTAAACGTACC GCAGAAGT.............GTTGGCGT
GTTAAACGTACC AAGGCTTG.............CAAAGTTC
GTTAAACGTACC TTCCGGTC.............TCCAGTCG



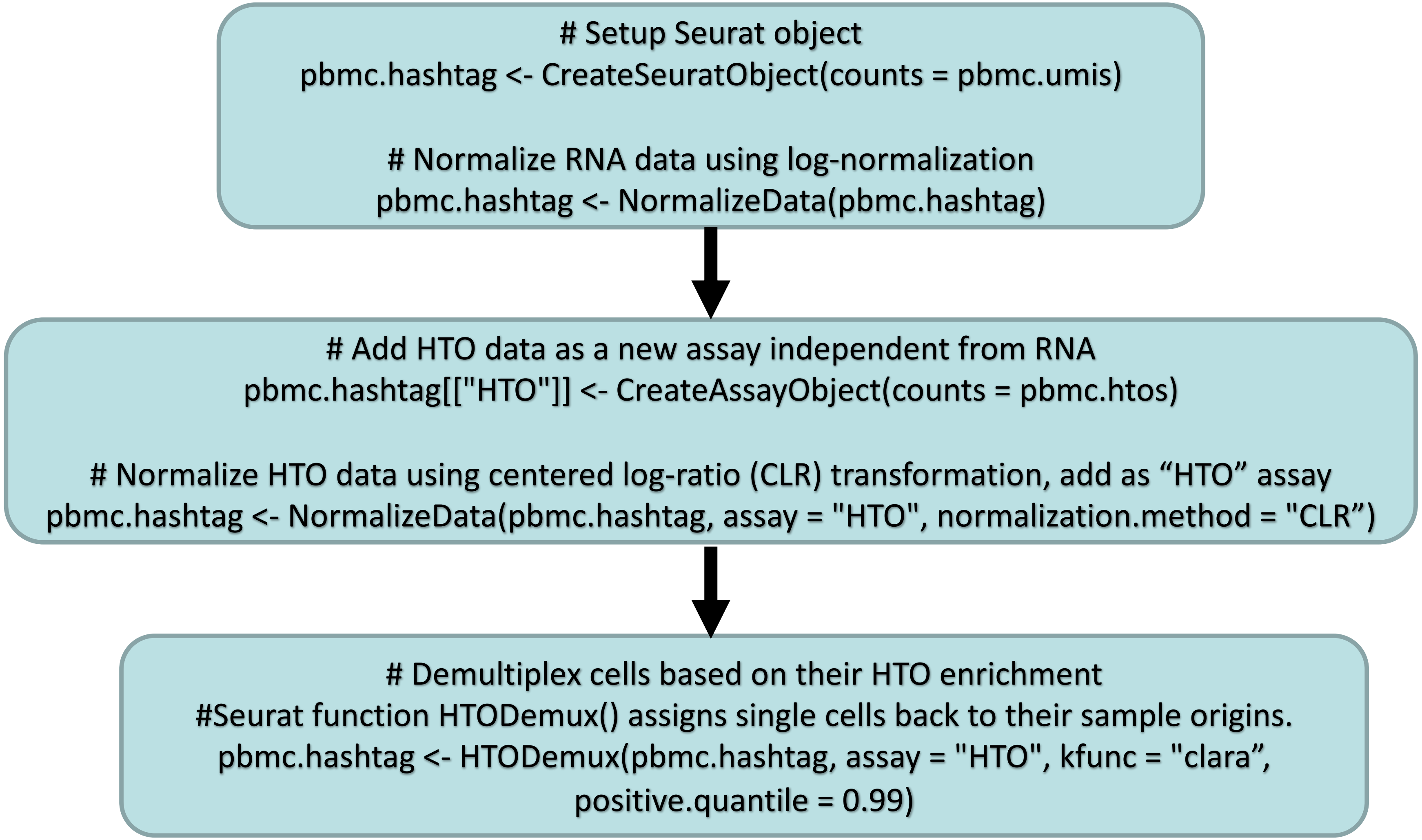
barcode	hashtagA	hashtagB	hashtagC	hashtagD	hashtagE	hashtagF	hashtagG	hashtagH
TTCCCAGCACCAGGTC	18683	17	15	1	26	14	12	11
CGATTGATCAACGGGA	15	26470	9	21	15	29	102	24
CATCGAAGTCATGCCG	40	3	32351	18	21	62	65	29
GTACTCCGTAGCGCAA	26	41	22	28841	4249	36	53	67
CACCACTTCCTCTAGC	31	17	50	35	16464	59	61	20
CTTTGCGAGGCCGAAT	8	4	4	20	25	30478	47	76
CTTAGGACACTAAGTC	5	14	5	10	17	19	29899	7
AAACGGGTCACCATAG	22	27	23	29	8	27	66	25930

(HTO) count matrix generated with CITE-seq-Count that processes the fastq files
https://github.com/Hoohm/CITE-seq-Count

```
# Load in the UMI/feature matrix from Seurat
pbmc.umis <- readRDS("../data/pbmc_umi_mtx.rds")

#Load in the hashtag (HTO) count matrix
pbmc.htos <- readRDS("../data/pbmc_hto_mtx.rds")
```

```
# Select cell barcodes detected by both RNA and HTO
joint.bcs <- intersect(colnames(pbmc.umis), colnames(pbmc.htos))

# Subset RNA and HTO counts by joint cell barcodes
pbmc.umis <- pbmc.umis[, joint.bcs]
pbmc.htos <- as.matrix(pbmc.htos[, joint.bcs])
```

Seurat workflow for multiplexing and doublet detection



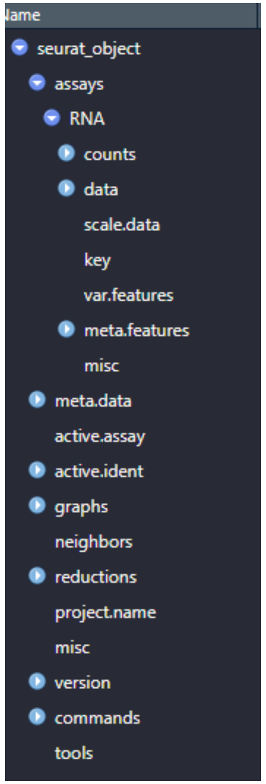
```
# Setup Seurat object
pbmc.hashtag <- CreateSeuratObject(counts = pbmc.umis)

# Normalize RNA data using log-normalization
pbmc.hashtag <- NormalizeData(pbmc.hashtag)
```

```
# Add HTO data as a new assay independent from RNA
pbmc.hashtag[["HTO"]] <- CreateAssayObject(counts = pbmc.htos)

# Normalize HTO data using centered log-ratio (CLR) transformation, add as “HTO” assay
pbmc.hashtag <- NormalizeData(pbmc.hashtag, assay = "HTO", normalization.method = "CLR”)
```

```
# Demultiplex cells based on their HTO enrichment
#Seurat function HTODemux() assigns single cells back to their sample origins.
pbmc.hashtag <- HTODemux(pbmc.hashtag, assay = "HTO", kfunc = "clara”,
positive.quantile = 0.99)
```

Clara uses a k-medoid clustering function for large sets (k-means used for smaller sets)

The HTODemux threshold for classification of cells can be adjusted:. https://rdrr.io/github/satijalab/seurat/man/HTODemux.html



```
Name
seurat_object
  assays
    RNA
      counts
      data
      scale.data
      key
      var.features
      meta.features
      misc
  meta.data
  active.assay
  active.ident
  graphs
  neighbors
  reductions
  project.name
  misc
  version
  commands
  tools
```

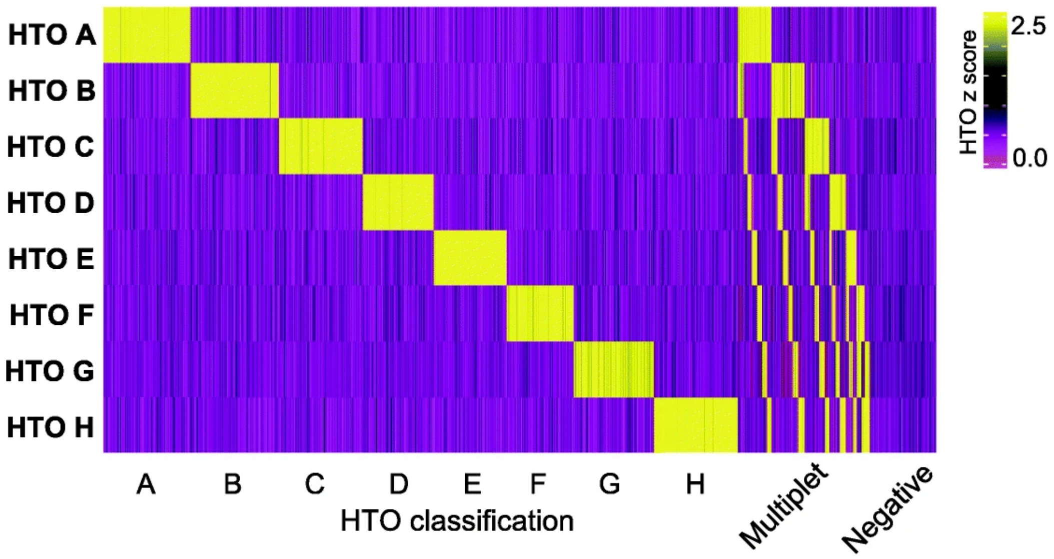
$$x_i' = \log \frac{x_i}{\left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}}}$$

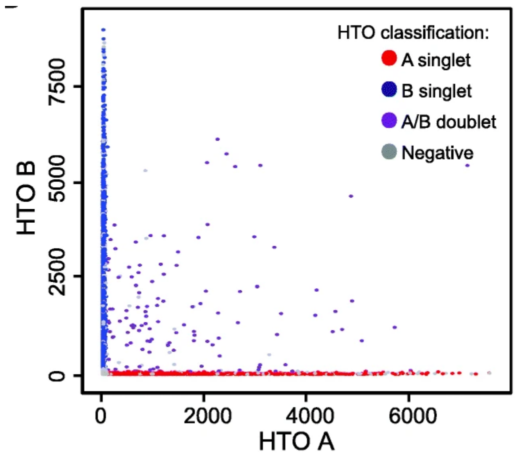
CLR transformation: Counts divided by the geometric mean.

xi= count of an HTO in cell i

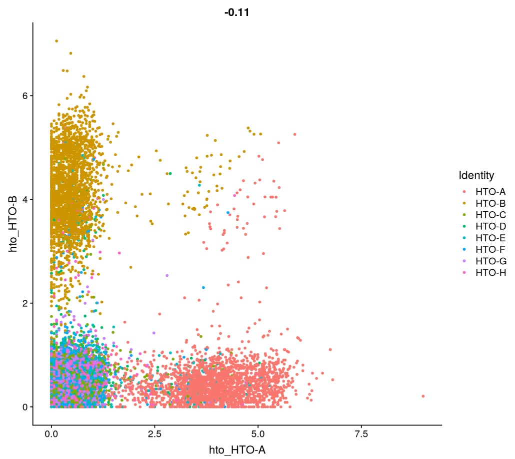
n=total # cells

Results of demultiplexing & doublet detection



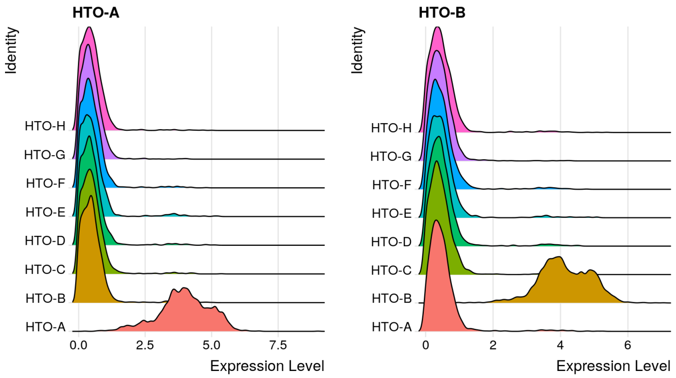


HTO-A and HTO-B signals are mutually exclusive between A and B singlets.



Remaining singlets are at the bottom left of the HTO-A/B expression space.

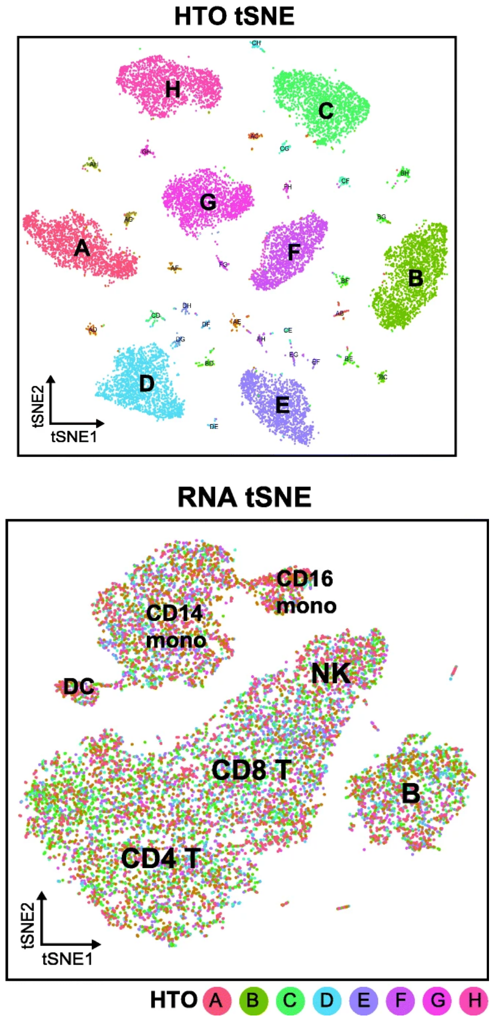
- HTOHeatmap in Seurat draws a heatmap of hashtag oligo signals across singlets/doublets/negative cells.
- HTOHeatmap(pbmc.hashtag, assay = "HTO", ncells = 5000)
- Subsampling cells to generate heatmaps quickly with ncells.

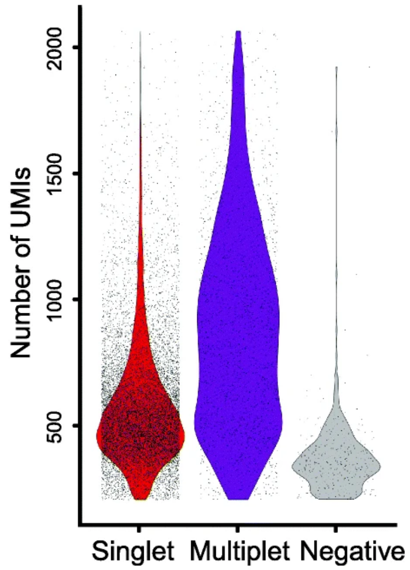


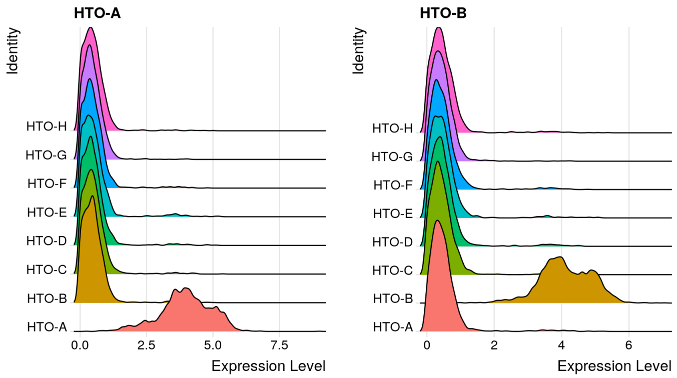
Ridgeline plots

Stoeckius et al. Cell hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome biology. 2018 Dec;19(1):224.

Visualization of demultiplexing & doublet detection results





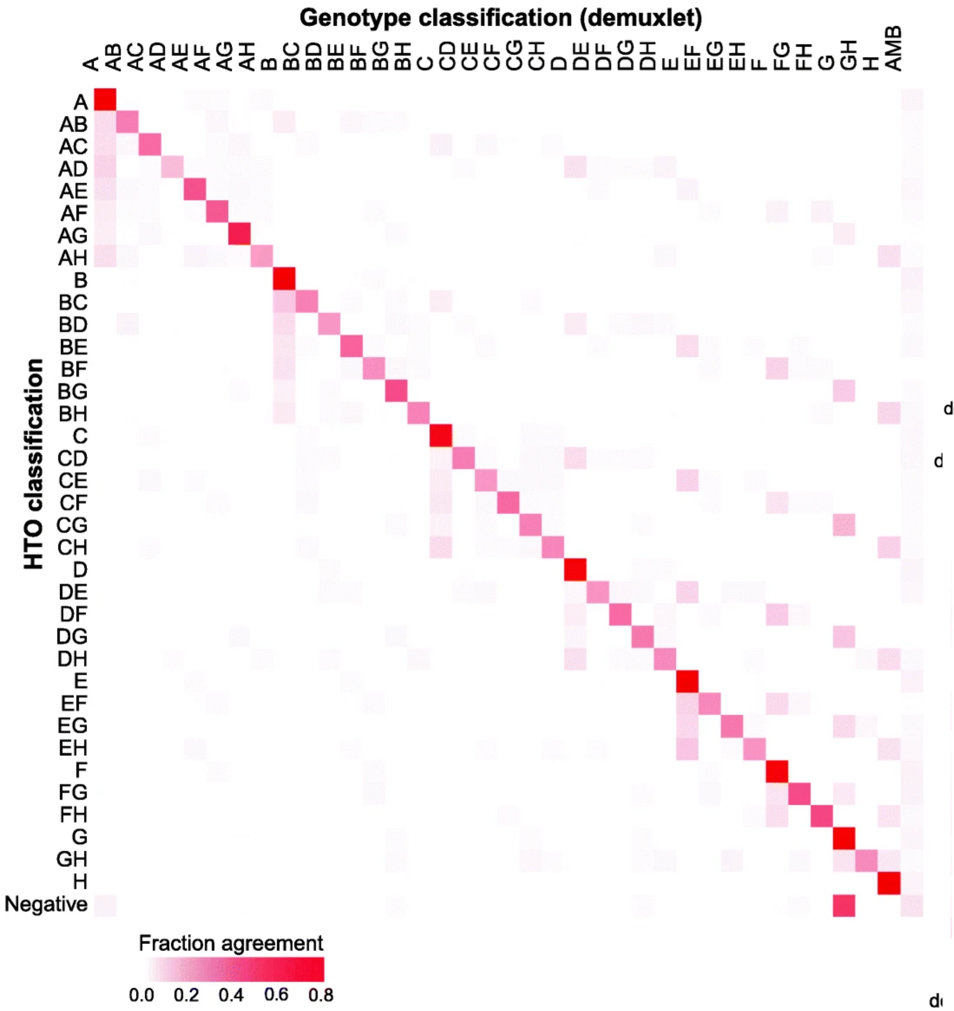


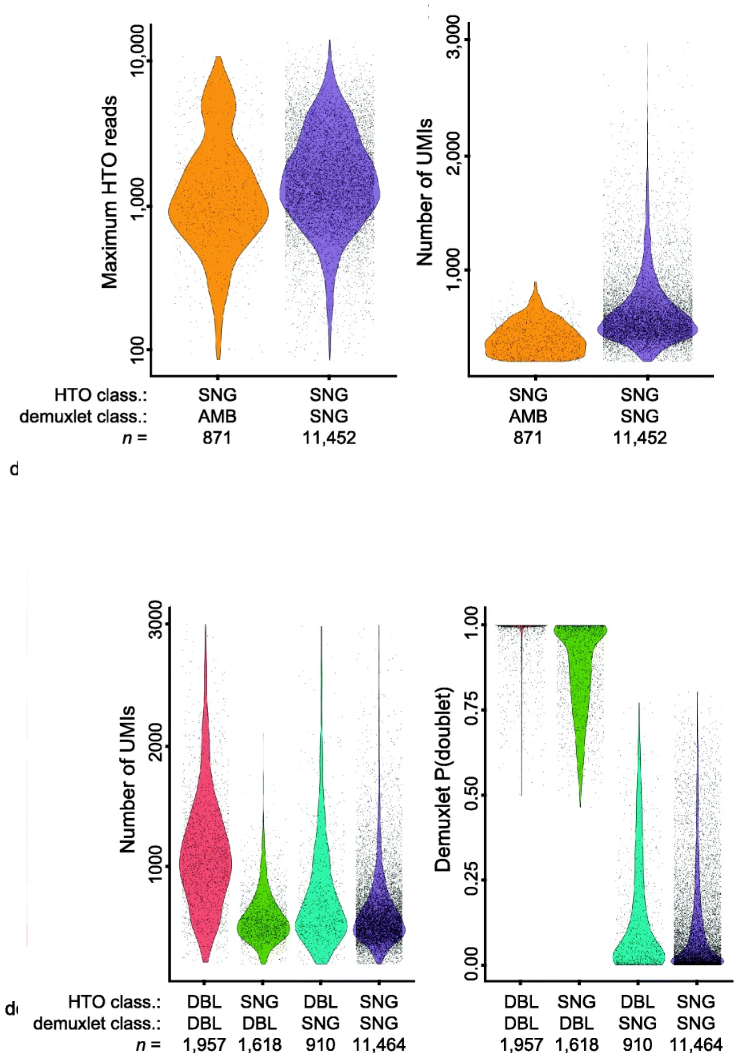
- Cells mapped to HTOs A-H form distinct clusters on the t-SNE based on their overall expression profiles.
- Remaining clusters of doublets are clearly separated from clusters formed by singlets.
- Distribution of number of UMIs shift up in multiplets and down in the negative group.
- Wide UMI range in multiplets shows the difficulty of identifying/predicting multiplets using only a UMI cut-off (conventional QC filtering).
- Clustering of singlets show seven distinct hematopoietic subpopulations interspersed across all 8 donors (HTO-A through HTO-H)

Stoeckius et al. Cell hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome biology. 2018 Dec;19(1):224.

Validation of demultiplexing & doublet using demuxlet (genotype driven sample fingerprinting)

Frederick National Laboratory for Cancer Research



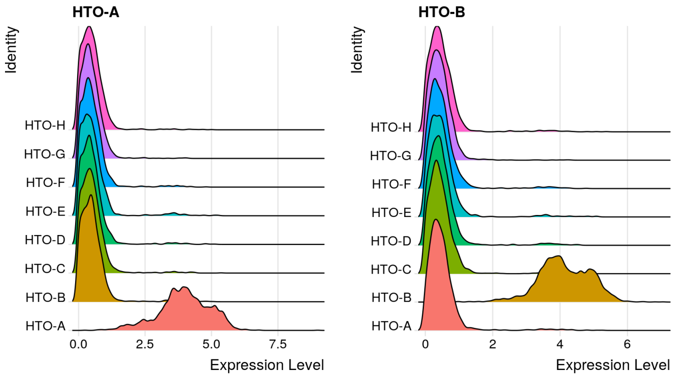


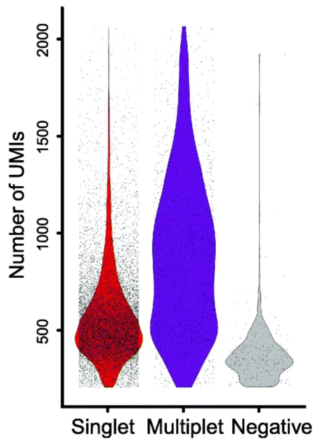
- Strong concordance between HTO-based classifications of HTODemux and genotype-based classifications (demuxlet).
- Comparison made between fraction of cell barcodes in agreement between the two classifications.
- Number of reads supporting the highest expressed HTO distributed the same way in discordant & concordant cells
- Discordant cells have lower UMI counts (below minumum depth for demuxlet for genotype based classification).
- Barcodes classified as doublets by both techniques have positive shift in their UMI distribution (increased library complexity).
- Demuxlet has lower doublet confidence for discordant doublet/singlet calls.

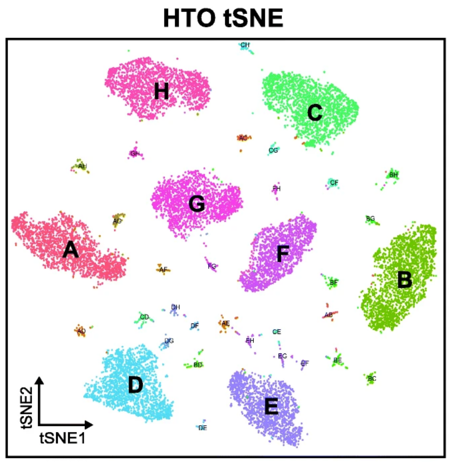
Stoeckius et al. Cell hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome biology. 2018 Dec;19(1):224.

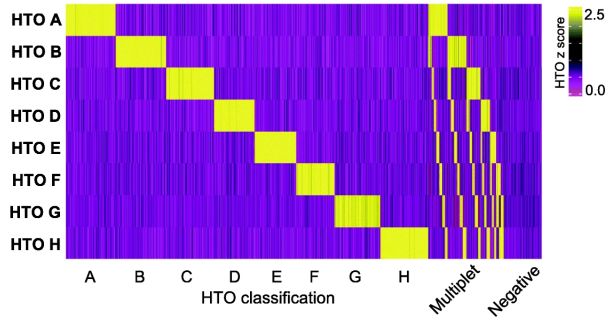
Wrapping-up

- Cell hashing with barcoded antibodies
- Demultiplexing and doublet detection
- Seurat workflow for integrating RNA and HTO assays, demultiplexing and doublet detection
- Visualization of results with heatmaps, scatter, violin and ridgeline plots
- Validation of cell hashing results using demuxlet









Acknowledgements:

NIAID Collaborative Bioinformatics Resource (NCBR)

Justin Lack (Lead), Arun Boddapati, Susan Huse, Vasu Kuram, Tovah Markowitz, Paul Schaughency