SINGLE CELL RNA-seq WORKSHOP

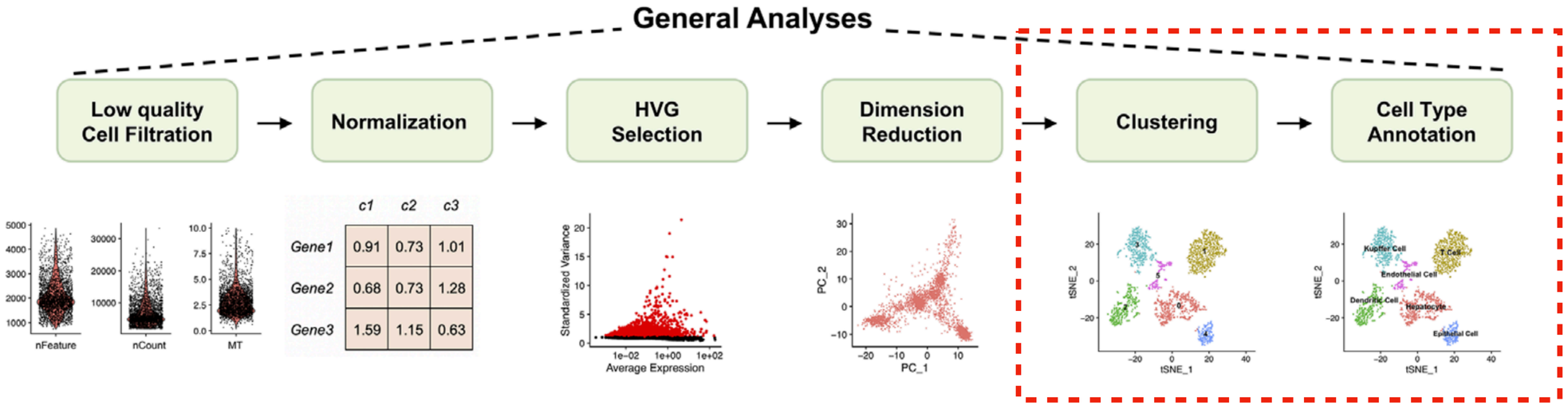
TREX x BioHPC

Week 3

Faraz Ahmed - 02/26/24

Analysis Overview:

Scoring Methods +
Reference Based Alignment

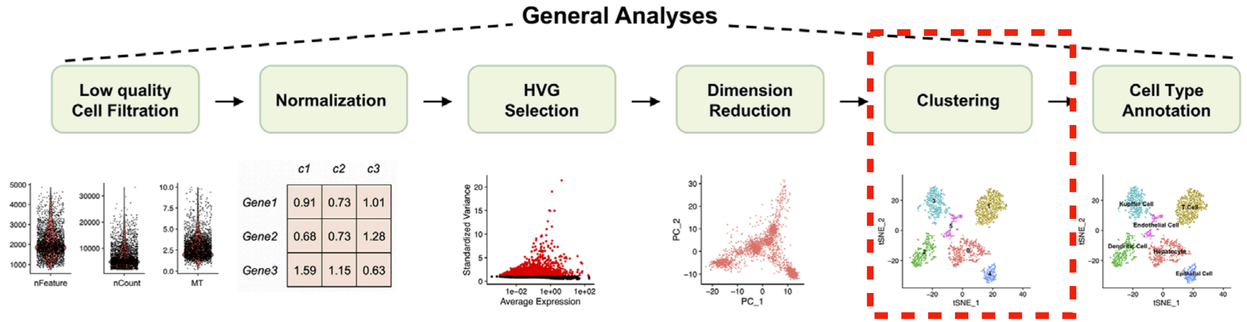


Doublet Filtering
Ambient RNA Removal
CellCycle Regression
Integration

Comprehensive Quality Control Analysis

1. Removing Unwanted Technical or Biological Variation

Clustering



Ideal Cluster's:

Cell Type Specific Clusters;

Roughly Good representation of all samples in each cluster;

Free of Technical Artifacts/Unwanted Variation;

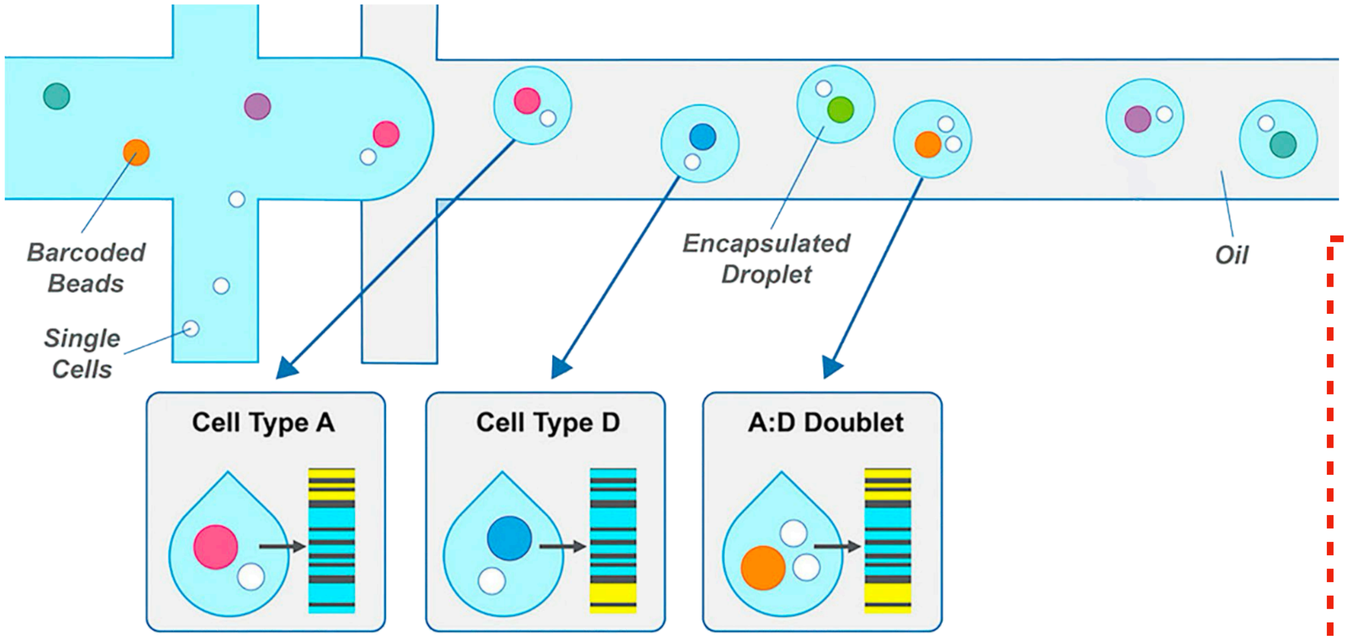
Unwanted Technical Variation:

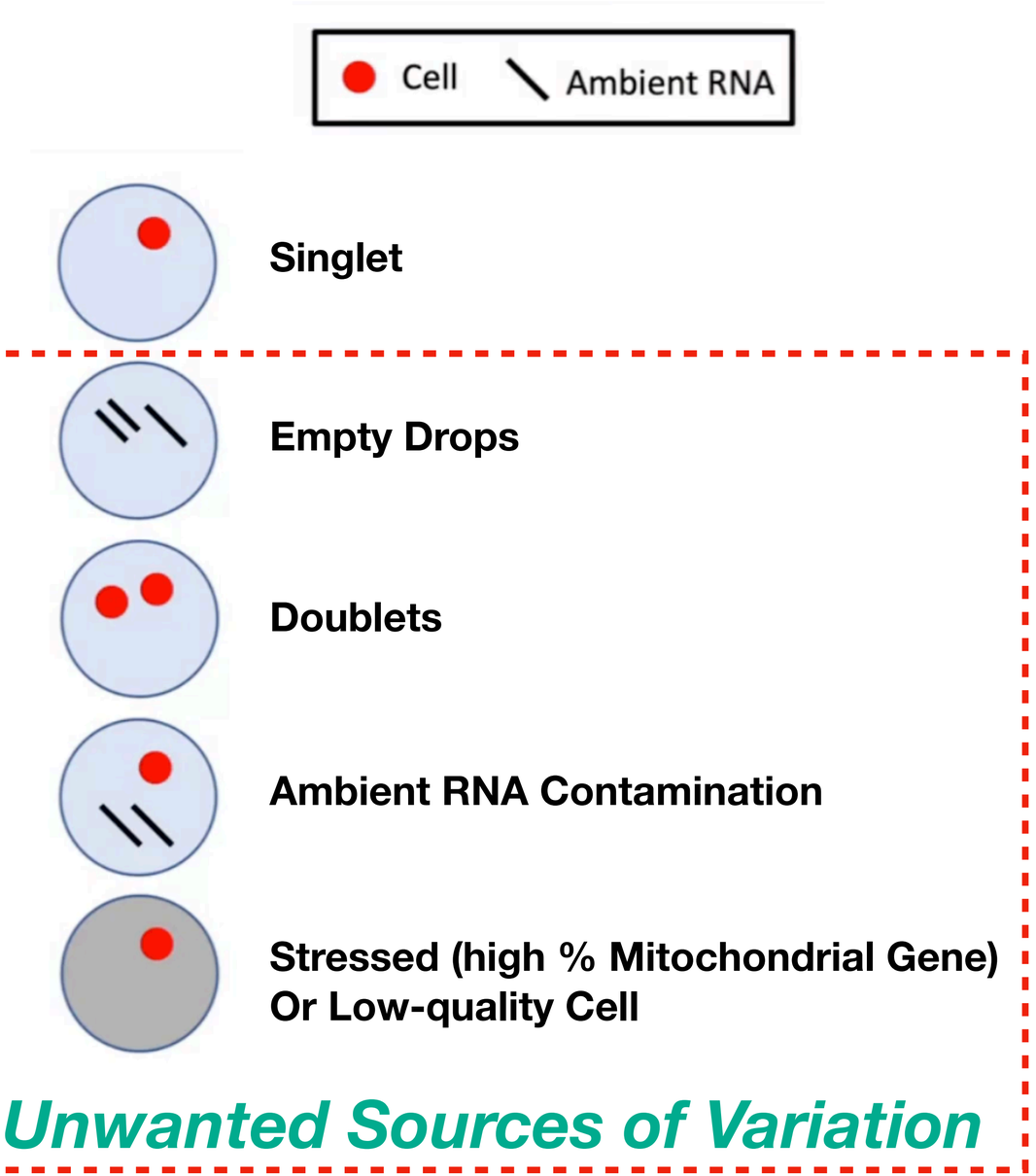
- **Batch**
- **Ambient RNA**
- **Doublets**
- **Library Protocol**

Unwanted Biological Variation:

- **CellCycle**
- **Proportion of MT reads**
- **Sample/Patient Sex or Age etc.**

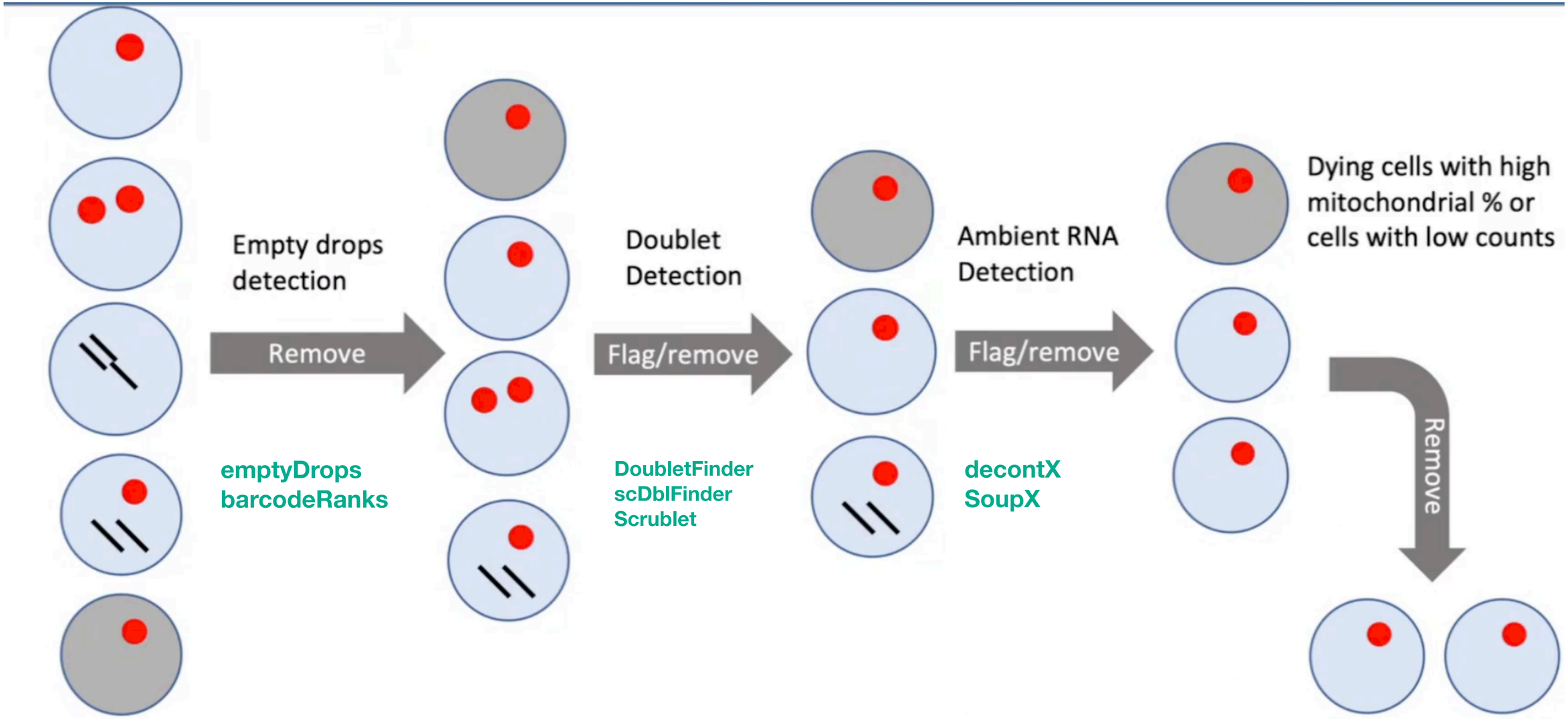
10x Chromium Overview:





https://scienceblog.cincinnatichildrens.org/new-tool-improves-outcomes-of-single-cell-research/

Strategy For Technical Contamination Filtering



Strategy For Technical Contamination Filtering

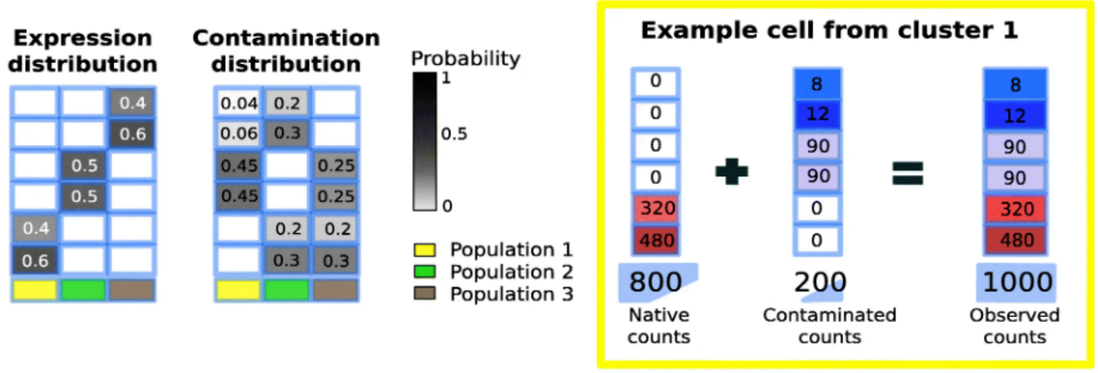
Yang *et al. Genome Biology* (2020) 21:57
https://doi.org/10.1186/s13059-020-1950-6

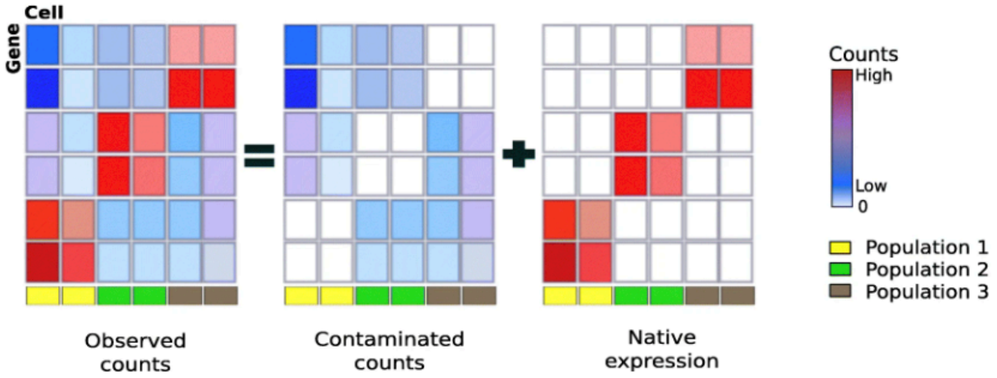
Genome Biology

METHOD Open Access

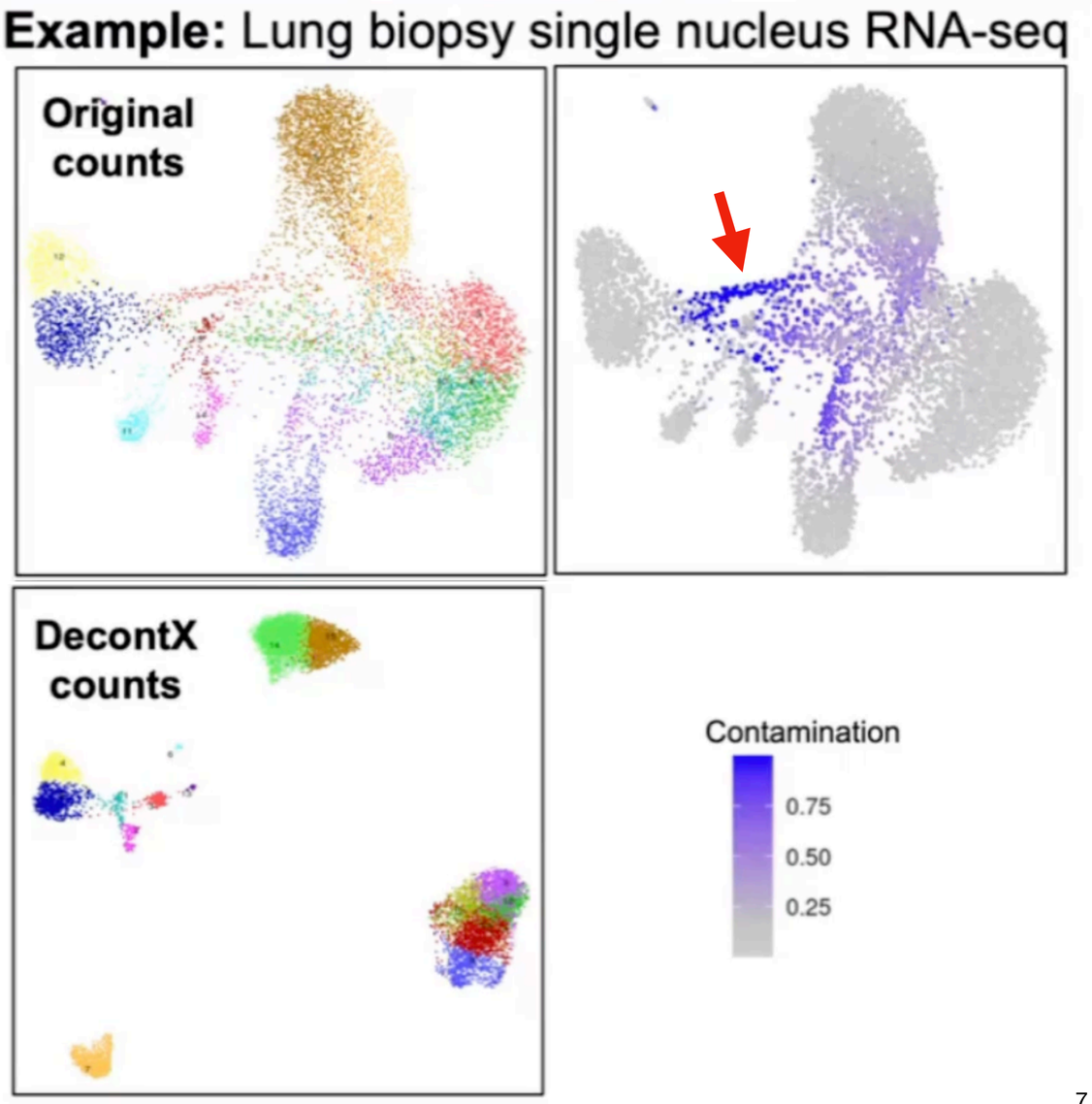
Decontamination of ambient RNA in single-cell RNA-seq with DecontX

Shiyi Yang[1], Sean E. Corbett[1], Yusuke Koga[1], Zhe Wang[1], W Evan Johnson[1], Masanao Yajima[2] and Joshua D. Campbell[1*]

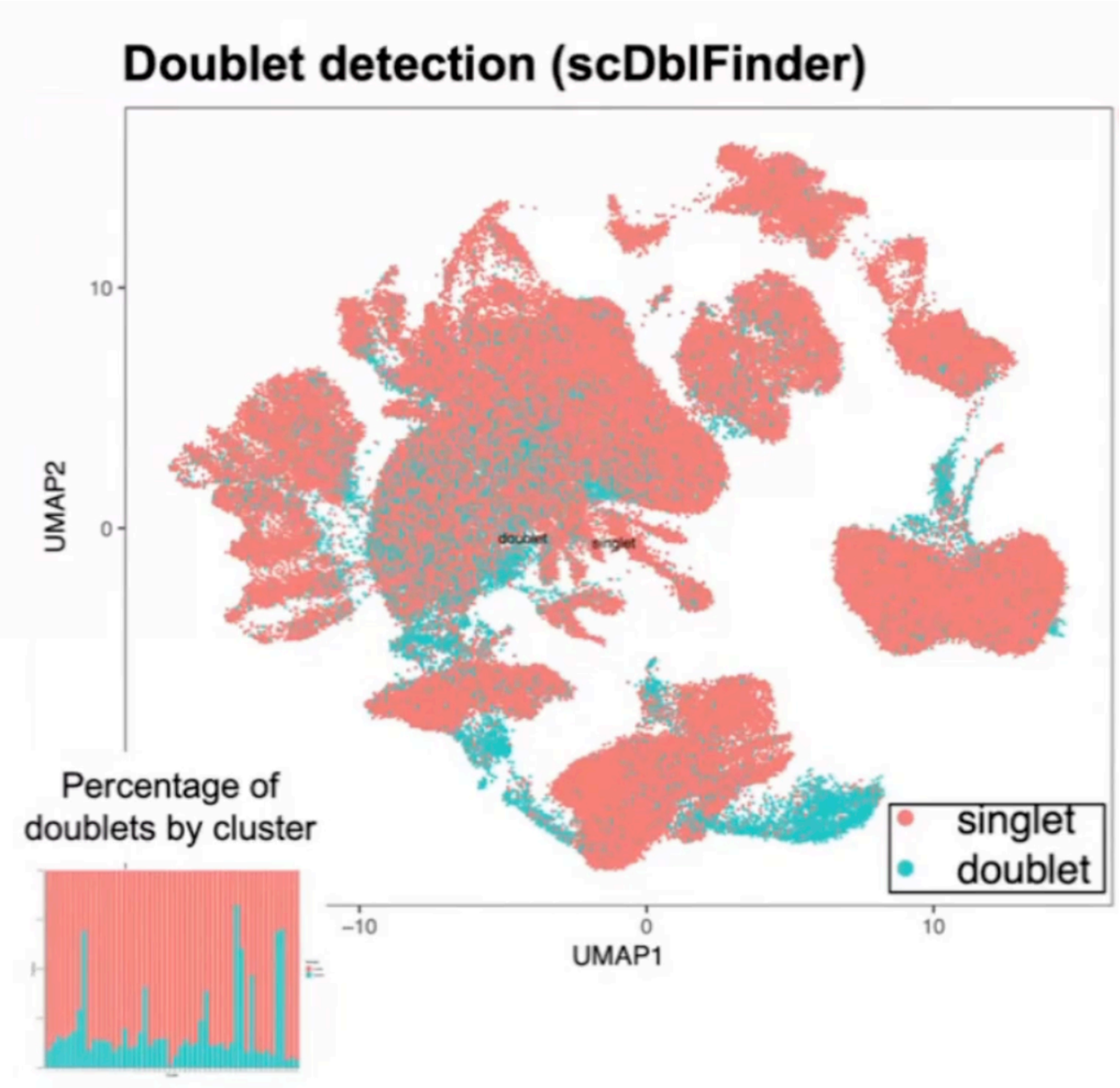




Example: Lung biopsy single nucleus RNA-seq



Strategy For Technical Contamination Filtering



Doublet often appear as *fingers* coming off of clusters

Key is to remove clusters enriched for doublets (set resolution high)

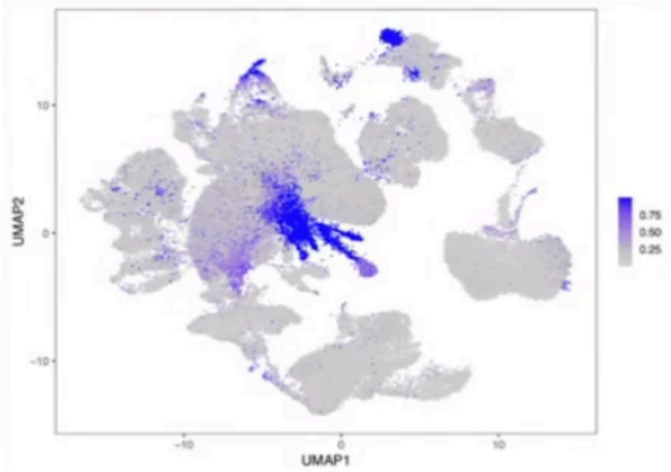
Doublets can become more prominent after sub-clustering of specific cell types

Best strategy is to verify doublet calls with *double positive* expression of known markers

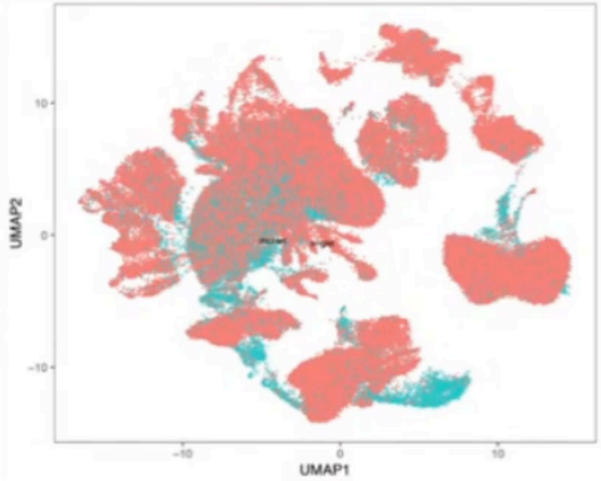
Caveat: Watch out for transitory cell types
- False Positives

Example — Large number of captured cells per sample

Ambient RNA (decontX)

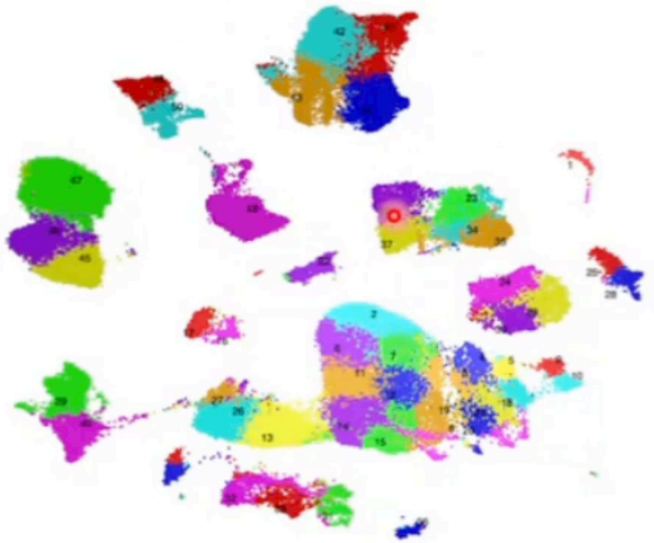


Doublet detection (scDblFinder)



Original Counts

1) Removed doublet enriched clusters
2) Removed highly contaminated cells
3) Used decontaminated counts



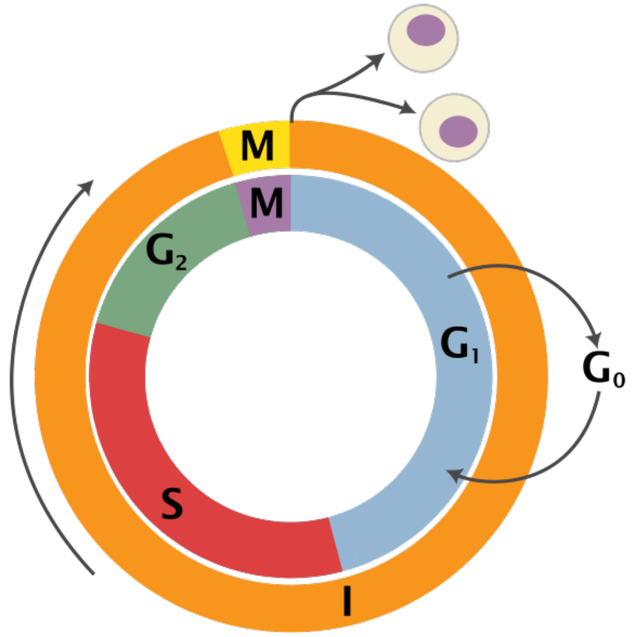
Unwanted Biological Variation

Cell Cycle Scoring:

The most common biological data correction (or source of "uninteresting" variation) in single cell RNA-seq is the effects of the cell cycle on the transcriptome.

After normalizing the seurat object, assign each cell a score based on its expression of G2/M and S phase canonical markers.

The score is calculated using the `CellCycleScoring()` function in Seurat.

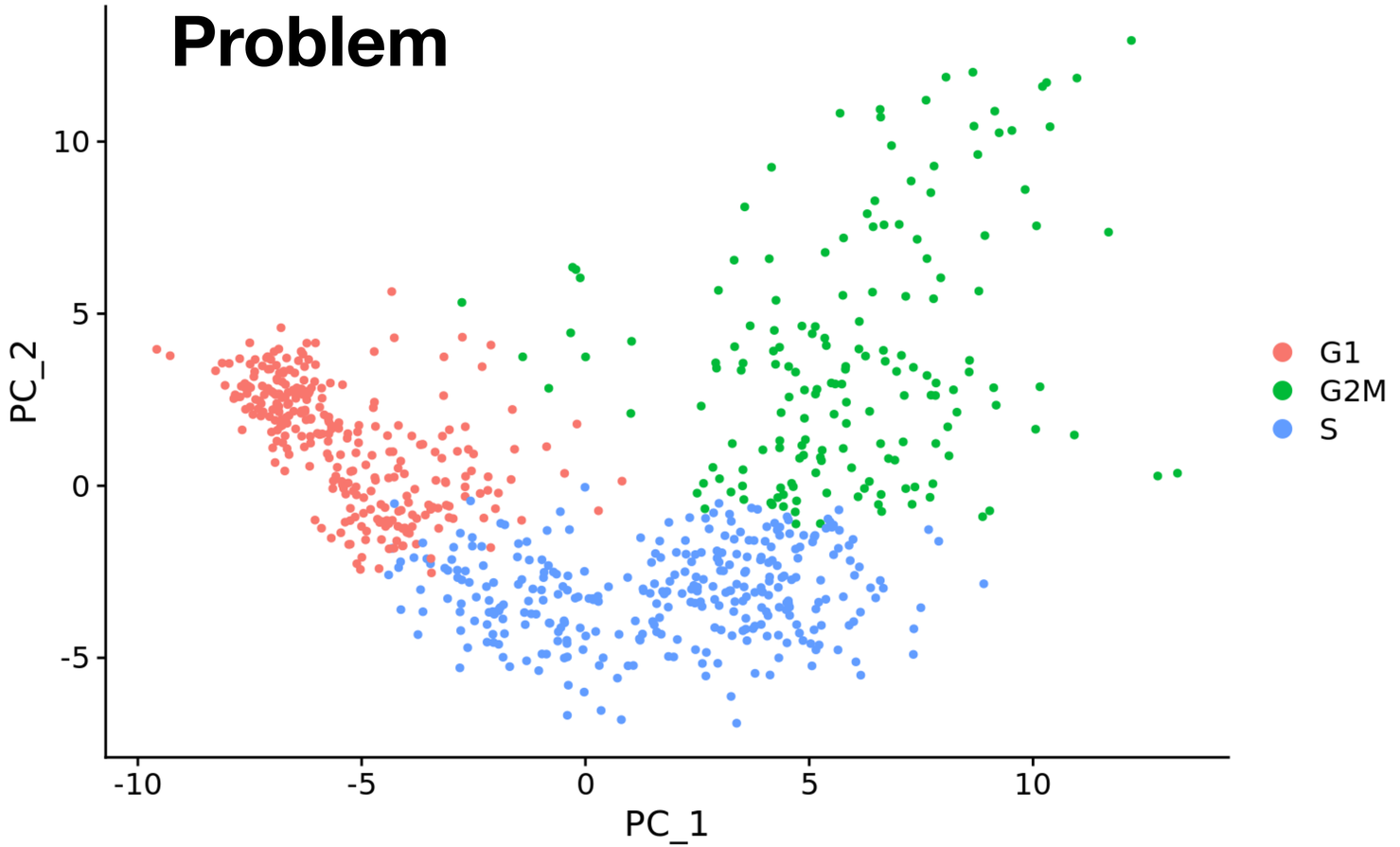


State	Phase	Abbreviation	Description
Resting	Gap 0	G_0	A phase where the cell has left the cycle and has stopped dividing.
Interphase	Gap 1	G_1	Cell growth. The *G_1 checkpoint* ensures that everything is ready for DNA synthesis.
	Synthesis	S	DNA replication.
	Gap 2	G_2	Growth and preparation for mitosis. The *G_2 checkpoint* ensures that everything is ready to enter the M (mitosis) phase and divide.
Cell division	Mitosis	M	Cell division occurs. The *Metaphase Checkpoint* ensures that the cell is ready to complete cell division.

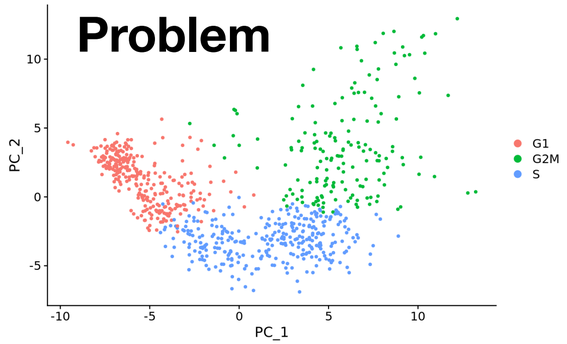
Removing Unwanted Biological Variation (CellCycle)

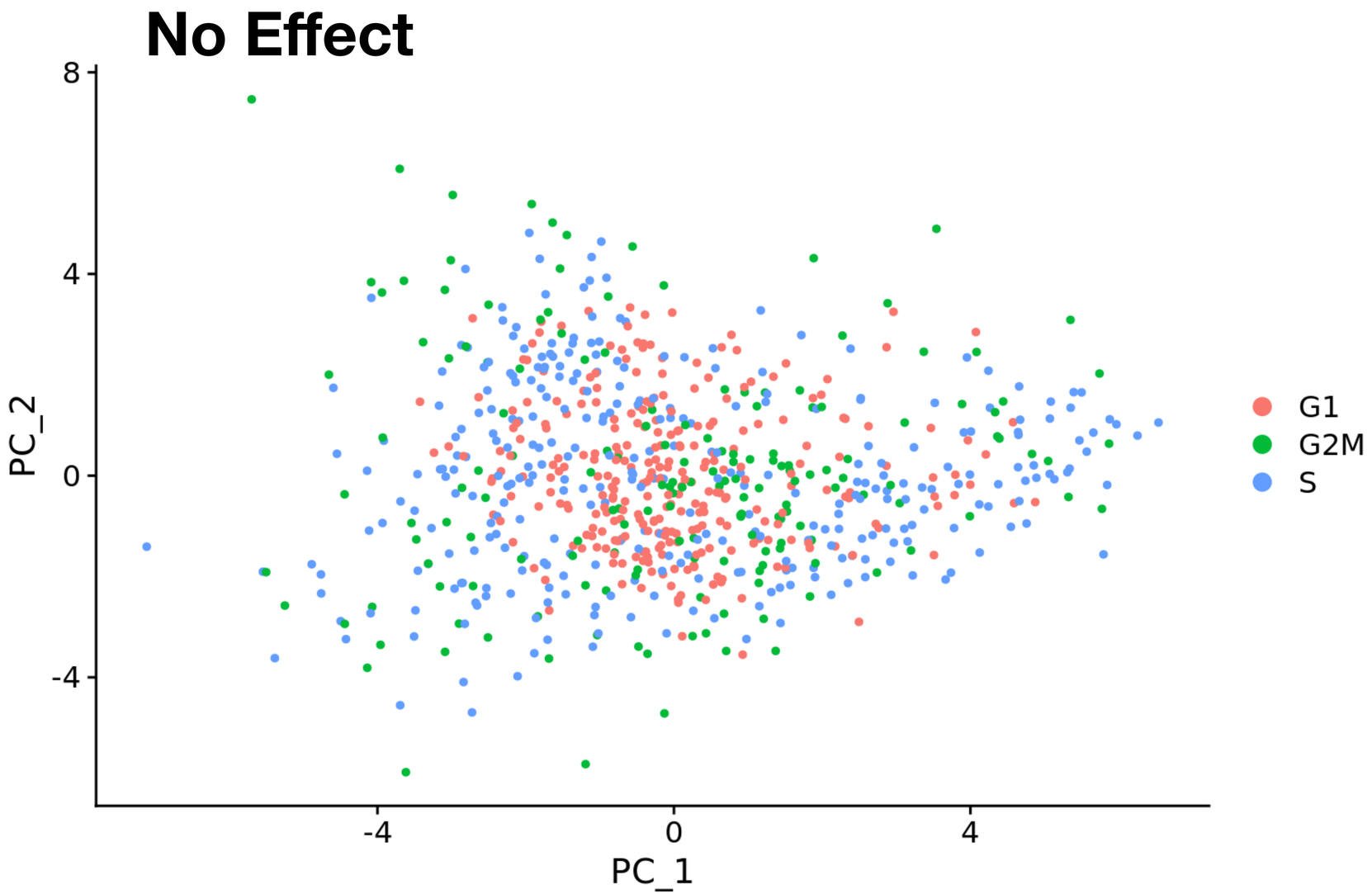
```
seurat_object <- CellCycleScoring(seurat_object, g2m.features = g2m_genes,
                                  s.features = s_genes)

View(seurat_object@meta.data) # View Scores in meta.data slot of seurat object

seurat_object <- seurat_object %>% FindVariableFeatures() %>% ScaleData() %>% RunPCA()
```

Removing Unwanted Biological Variation (CellCycle)



Removing Unwanted Biological Variation (CellCycle)





Removing Unwanted Biological Variation (%MT)

Mitochondrial Proportions:

```
# Check quartile values
summary(seurat_object@meta.data$percent.mt)

# Turn percent.mt into categorical factor vector based on quartile values
seurat_object@meta.data$mitoFr <- cut(seurat_object@meta.data$mitoRatio,
                   breaks=c(-Inf, $q1, $q2/median, $q3, Inf),
                   labels=c("Low","Medium","Medium high", "High"))


seurat_object <- seurat_object %>% FindVariableFeatures() %>% ScaleData() %>% RunPCA()
```

LogNormalize v. SCTransform

```
ScaleData(
  object,
  features = NULL,
  vars.to.regress = NULL,
  latent.data = NULL,
  split.by = NULL,
  model.use = "linear",
  use.umi = FALSE,
  do.scale = TRUE,
  do.center = TRUE,
  scale.max = 10,
  block.size = 1000,
  min.cells.to.block = 3000,
  verbose = TRUE,
  ...
)
```

```
SCTransform(
  object,
  cell.attr,
  reference.SCT.model = NULL,
  do.correct.umi = TRUE,
  ncells = 5000,
  residual.features = NULL,
  variable.features.n = 3000,
  variable.features.rv.th = 1.3,
  vars.to.regress = NULL,
  do.scale = FALSE,
  do.center = TRUE,
  clip.range = c(-sqrt(x = ncol(x = umi)/30), sqrt(x = ncol(x = umi)/30)),
  vst.flavor = "v2",
  conserve.memory = FALSE,
  return.only.var.genes = TRUE,
  seed.use = 1448145,
  verbose = TRUE,
  ...
)
```

2. Integration

Batch Effects

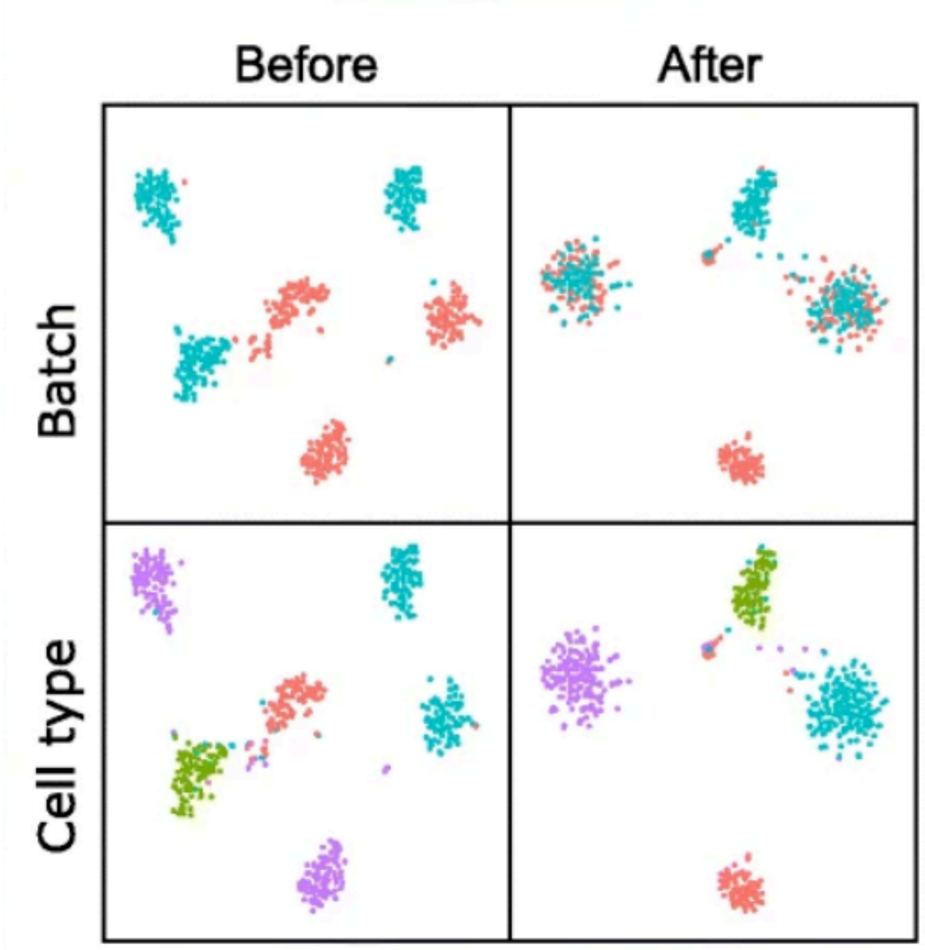
In Context of scRNA seq, *batch effect* refers to *non-biological variations* that arise when *samples* are processed in separate runs or under slightly different conditions.

Variations can occur due to differences in:

- **Sample Handling**
- **Reagent Lots**
- **Equipment , or**
- **Time of Processing**

To remove these effects, one may find it necessary to run integration/alignment

Integration



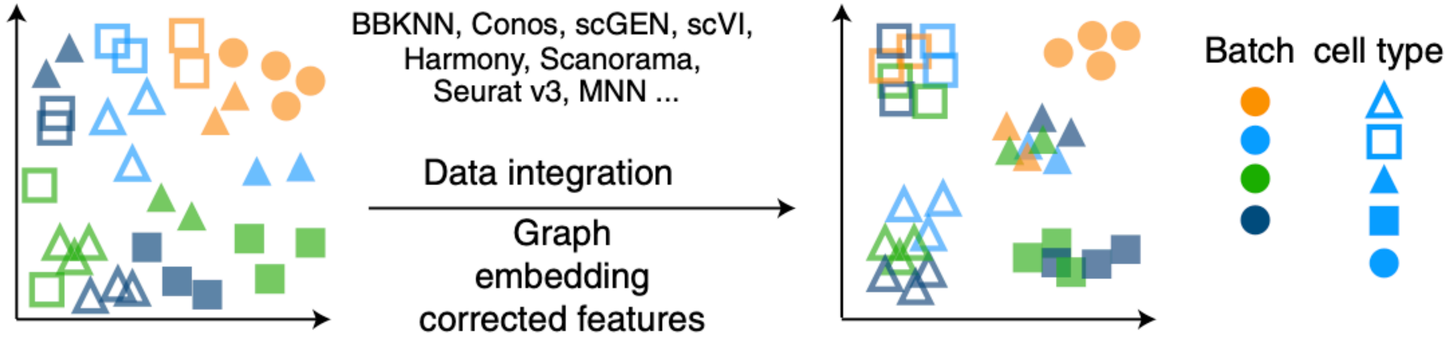
Tran et al. 2020 *Genome Biology*

nature methods
ANALYSIS
https://doi.org/10.1038/s41592-021-01336-8
Check for updates

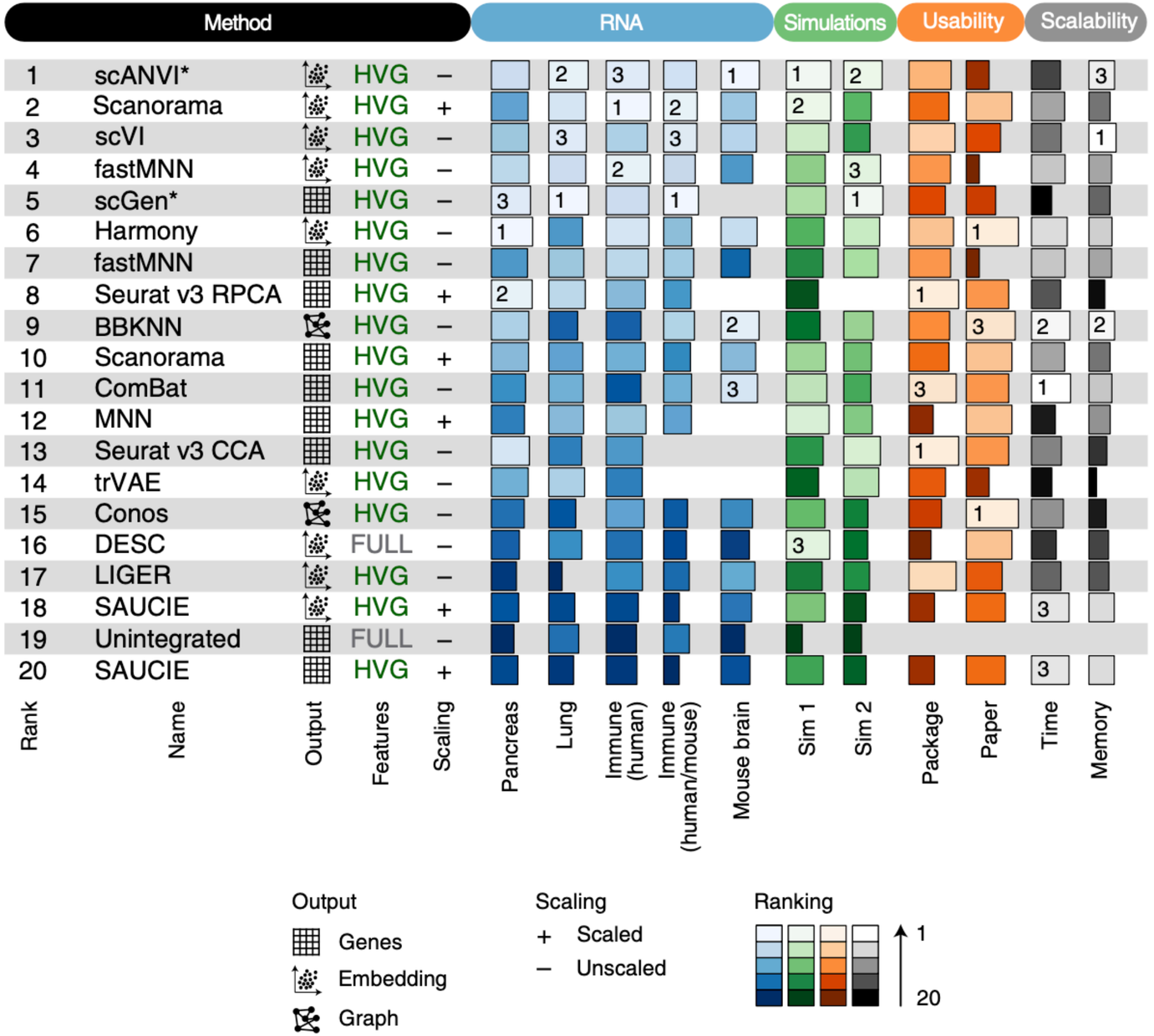
OPEN

Benchmarking atlas-level data integration in single-cell genomics

Malte D. Luecken[1], M. Büttner[1], K. Chaichoompu[1], A. Danese[1], M. Interlandi[2], M. F. Mueller[1], D. C. Strobl[1], L. Zappia[1,3], M. Dugas[4], M. Colomé-Tatché[1,5,6] ✉ and Fabian J. Theis[1,3,5] ✉



Integration



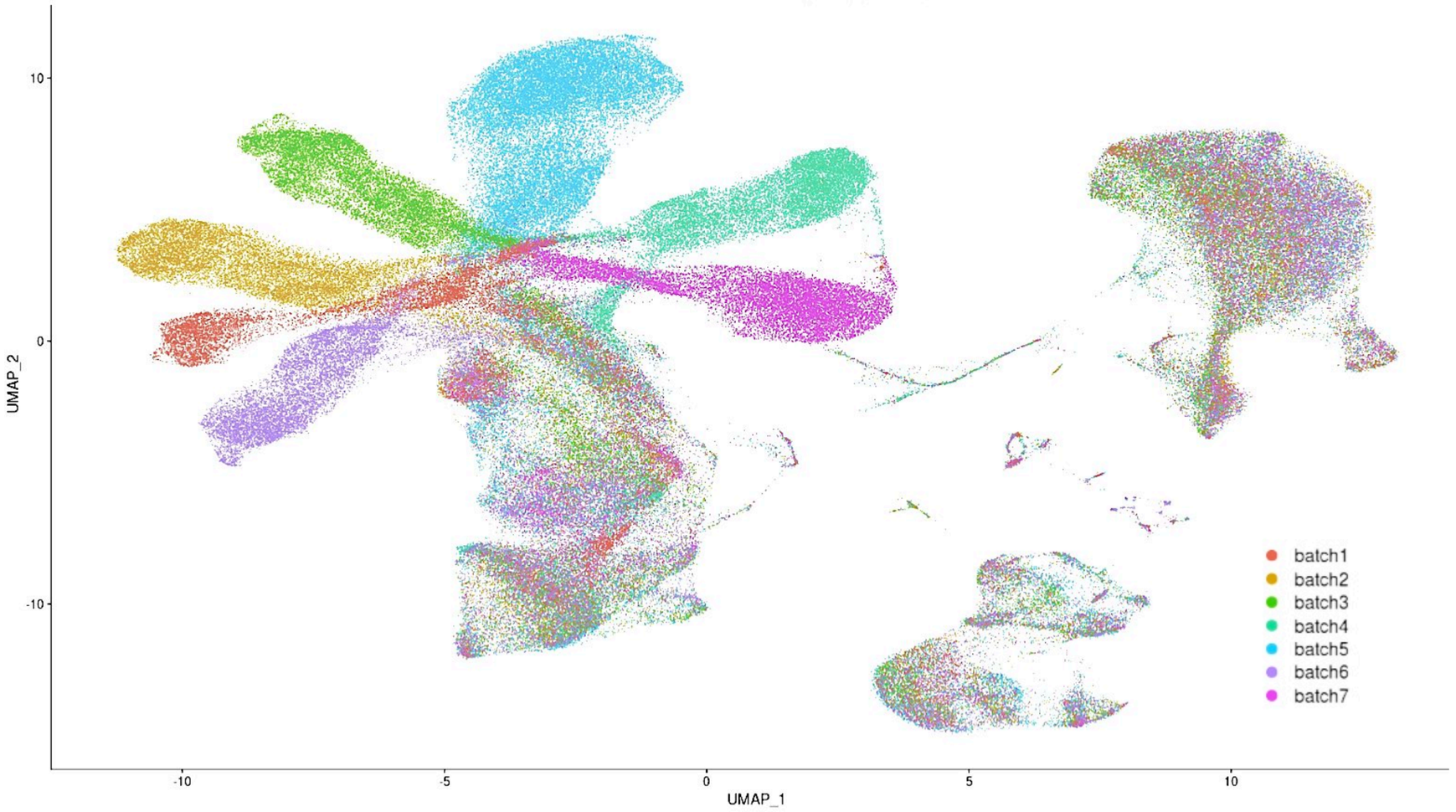
Batch Effects - Example

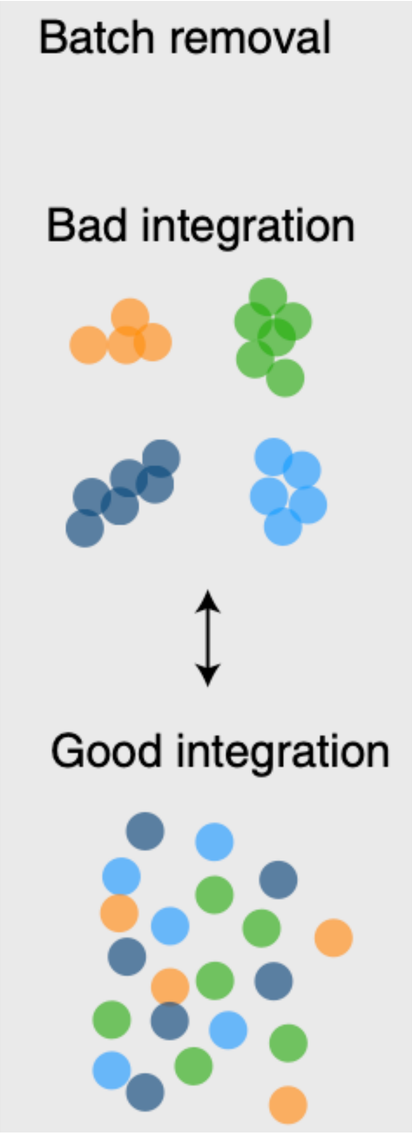
nature methods
ANALYSIS
https://doi.org/10.1038/s41592-021-01336-8
Check for updates

OPEN

Benchmarking atlas-level data integration in single-cell genomics

Malte D. Luecken[1], M. Büttner[1], K. Chaichoompu[1], A. Danese[1], M. Interlandi[2], M. F. Mueller[1], D. C. Strobl[1], L. Zappia[1,3], M. Dugas[4], M. Colomé-Tatché[1,5,6] ✉ and Fabian J. Theis[1,3,5] ✉





Considerations & Strategies when performing integration

To integrate or not to integrate?

Should integration be between studies, batches or samples?

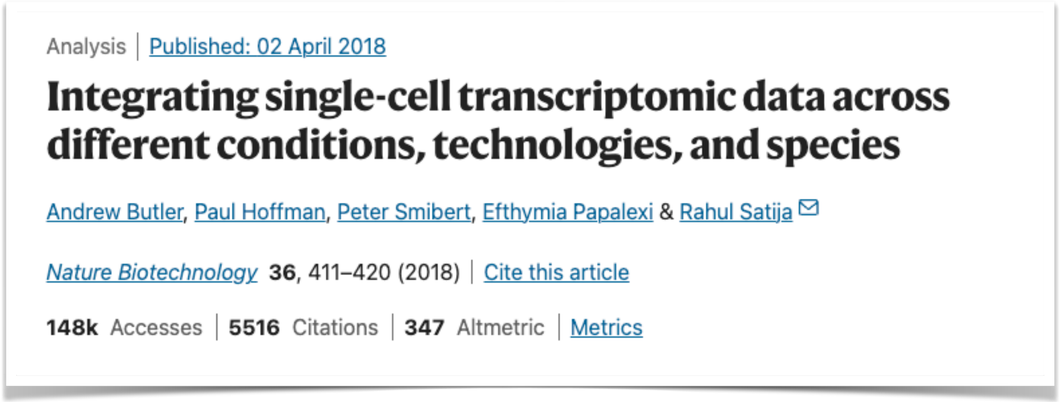
One popular approach has been to perform an initial integration of all cells to align ‘major’ cell types and then re-cluster individual cell types without integration

Often requires some prior understanding of biological and technical variability in your data:

- **How were samples processed (day, protocol, treatments, conditions)**
- **If you are looking at different conditions, do you expect it to affect all cell types or a subset of cell types?**
- **Important to have biological positive controls to guide integration process (e.g. known cell types, pathways or marker genes)**

Seurat Supported Integration Methods

Canonical Correlation Analysis (CCA)



Analysis | Published: 02 April 2018

Integrating single-cell transcriptomic data across different conditions, technologies, and species

Andrew Butler, Paul Hoffman, Peter Smibert, Efthymia Papalexi & Rahul Satija

Nature Biotechnology **36**, 411–420 (2018) | Cite this article

148k Accesses | **5516** Citations | **347** Altmetric | Metrics

Perform canonical correlation analysis (CCA):

CCA identifies shared sources of variation between the conditions/groups.

Using the 3000 most variant genes from each sample
This step roughly aligns the cells using the greatest shared sources of variation.

Identify anchors or mutual nearest neighbors (MNNs) across datasets (sometimes incorrect anchors are identified)

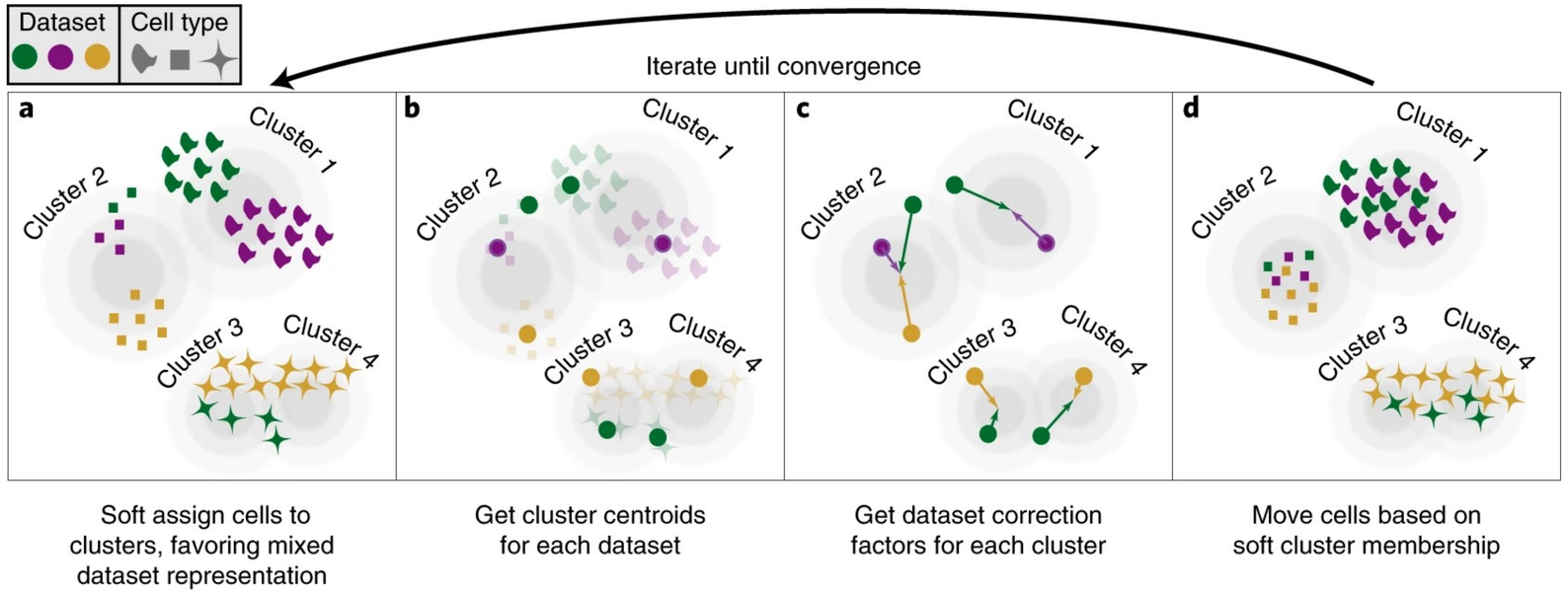
Filter anchors to remove incorrect anchors:
Assess the similarity between anchor pairs by the overlap in their local neighborhoods (incorrect anchors will have low scores)

Integrate the conditions/datasets:
Use anchors and corresponding scores to transform the cell expression values, allowing for the integration of the conditions/datasets (different samples, conditions, datasets, modalities)

Seurat Supported Integration Methods

Harmony Integration

Harmony iteratively learns a cell-specific linear correction function



Article | Published: 18 November 2019

Fast, sensitive and accurate integration of single-cell data with Harmony

Ilya Korsunsky, Nghia Millard, Jean Fan, Kamil Slowikowski, Fan Zhang, Kevin Wei, Yuriy Baglaenko, Michael Brenner, Po-ru Loh & Soumya Raychaudhuri

Nature Methods **16**, 1289–1296 (2019) | Cite this article

84k Accesses | **2169** Citations | **147** Altmetric | Metrics

Faster and more efficient than CCA;

More reliable for Batch effected studies;

Suffers less from over-correction;

Seurat Supported Integration Methods

CCA

RPCA

Harmony

FastMNN

scVI

```
obj <- IntegrateLayers(
  object = obj, method = CCAIntegration,
  orig.reduction = "pca", new.reduction = "integrated.cca",
  verbose = FALSE
)
```

```
obj <- IntegrateLayers(
  object = obj, method = RPCAIntegration,
  orig.reduction = "pca", new.reduction = "integrated.rpca",
  verbose = FALSE
)
```

```
obj <- IntegrateLayers(
  object = obj, method = HarmonyIntegration,
  orig.reduction = "pca", new.reduction = "harmony",
  verbose = FALSE
)
```

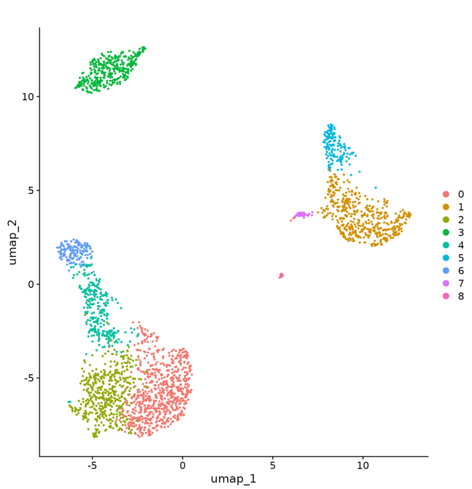
```
obj <- IntegrateLayers(
  object = obj, method = FastMNNIntegration,
  new.reduction = "integrated.mnn",
  verbose = FALSE
)
```

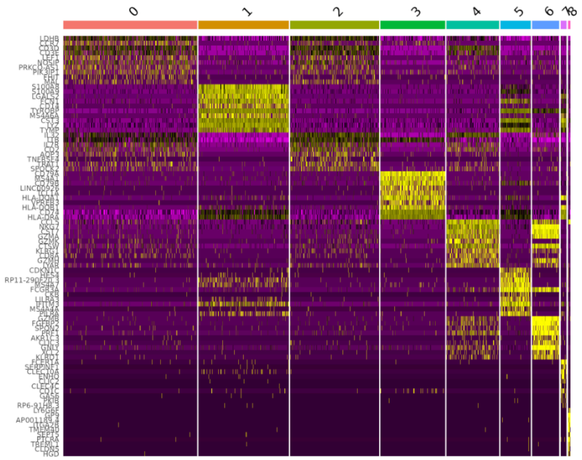
```
obj <- IntegrateLayers(
  object = obj, method = scVIIntegration,
  new.reduction = "integrated.scvi",
  conda_env = "../miniconda3/envs/scvi-env", verbose = FALSE
)
```

3. Downstream Analysis

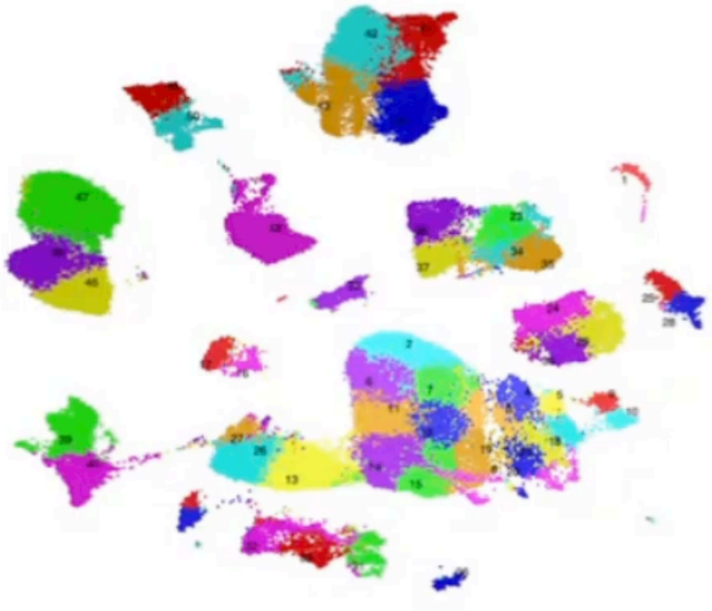
Approaches for FindMarker and Differential Expression

Tries to find ‘up-regulated’ genes by comparing each cluster to all other clusters





Can be challenging for complex datasets with many cell types and subpopulations



Seurat FindMarkers

In Seurat we can use Find Markers function to perform DE ***between*** clusters and also ***within*** clusters

Uses ***Wilcoxon Rank Sum*** Test for DE by default

Finds markers (differentially expressed genes) for each of the identity classes in a dataset

```
FindAllMarkers(
  object,
  assay = NULL,
  features = NULL,
  logfc.threshold = 0.1,
  test.use = "wilcox",
  slot = "data",
  min.pct = 0.01,
  min.diff.pct = -Inf,
  node = NULL,
  verbose = TRUE,
  only.pos = FALSE,
  max.cells.per.ident = Inf,
  random.seed = 1,
  latent.vars = NULL,
  min.cells.feature = 3,
  min.cells.group = 3,
  mean.fxn = NULL,
  fc.name = NULL,
  base = 2,
  return.thresh = 0.01,
  densify = FALSE,
  ...
)
```

DE *between clusters example*

```
# find markers for every cluster compared to all remaining cells, report only the positive
# ones
pbmc.markers <- FindAllMarkers(pbmc, only.pos = TRUE)
pbmc.markers %>%
    group_by(cluster) %>%
    dplyr::filter(avg_log2FC > 1)
```

```
## # A tibble: 7,046 × 7
## # Groups:   cluster [9]
##        p_val avg_log2FC pct.1 pct.2 p_val_adj cluster gene
##        <dbl>      <dbl> <dbl> <dbl>     <dbl> <fct>   <chr>
##  1 1.74e-109       1.19 0.897 0.593 2.39e-105 0       LDHB
##  2 1.17e- 83       2.37 0.435 0.108 1.60e- 79 0       CCR7
##  3 8.94e- 79       1.09 0.838 0.403 1.23e- 74 0       CD3D
##  4 3.05e- 53       1.02 0.722 0.399 4.19e- 49 0       CD3E
##  5 3.28e- 49       2.10 0.333 0.103 4.50e- 45 0       LEF1
##  6 6.66e- 49       1.25 0.623 0.358 9.13e- 45 0       NOSIP
##  7 9.31e- 44       2.02 0.328 0.11  1.28e- 39 0       PRKCQ-AS1
##  8 4.69e- 43       1.53 0.435 0.184 6.43e- 39 0       PIK3IP1
##  9 1.47e- 39       2.70 0.195 0.04  2.01e- 35 0       FHIT
## 10 2.44e- 33       1.94 0.262 0.087 3.34e- 29 0       MAL
## # ℹ 7,036 more rows
```

https://satijalab.org/seurat/articles/de_vignette

Seurat FindMarkers

In Seurat we can use Find Markers function to perform DE ***between*** clusters and also ***within*** clusters

Uses ***Wilcoxon Rank Sum*** Test for DE by default

DE *within cluster example*

```
# Find Markers case v ctrl in cluster 1

case.v.ctrl <- FindMarkers(seurat_object, ident.1 = “case”,
                           ident.2 = “ctrl”, group.by = “condition”,
                           subset.ident = 1, only.pos = F)
```

```
group.by = A column name present in meta.data
ident.1 & ident.2  = Factor values of the group.by column
subset.ident = Cluster Number or Name
```

Seurat FindMarkers

In Seurat we can use Find Markers function to perform DE ***between*** clusters and also ***within*** clusters

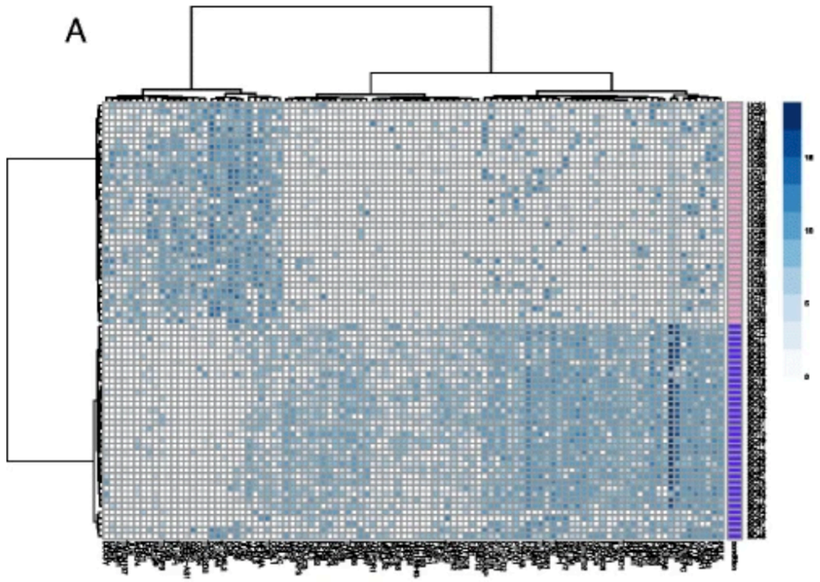
For Complicated DE (datasets with many co-variates)

Method | Open access | Published: 10 December 2015

MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data

Greg Finak, Andrew McDavid, Masanao Yajima, Jingyuan Deng, Vivian Gersuk, Alex K. Shalek, Chloe K. Slichter, Hannah W. Miller, M. Juliana McElrath, Martin Prlic, Peter S. Linsley & Raphael Gottardo

Genome Biology **16**, Article number: 278 (2015) | Cite this article



Finds markers (differentially expressed genes) for each of the identity classes in a dataset

```
FindAllMarkers(
  object,
  assay = NULL,
  features = NULL,
  logfc.threshold = 0.1,
  test.use = "wilcox",
  slot = "data",
  min.pct = 0.01,
  min.diff.pct = -Inf,
  node = NULL,
  verbose = TRUE,
  only.pos = FALSE,
  max.cells.per.ident = Inf,
  random.seed = 1,
  latent.vars = NULL,
  min.cells.feature = 3,
  min.cells.group = 3,
  mean.fxn = NULL,
  fc.name = NULL,
  base = 2,
  return.thresh = 0.01,
  densify = FALSE,
  ...
)
```

Seurat FindMarkers

```
# Find Markers case v ctrl in cluster 1

case.v.ctrl <- FindMarkers(seurat_object, ident.1 = “case”,
                           ident.2 = “ctrl”, group.by = “condition”,
                           subset.ident = 1, only.pos = F)
```

```
group.by = A column name present in meta.data
ident.1 & ident.2  = Factor values of the group.by column
subset.ident = Cluster Number or Name
```

Biological Replicates?

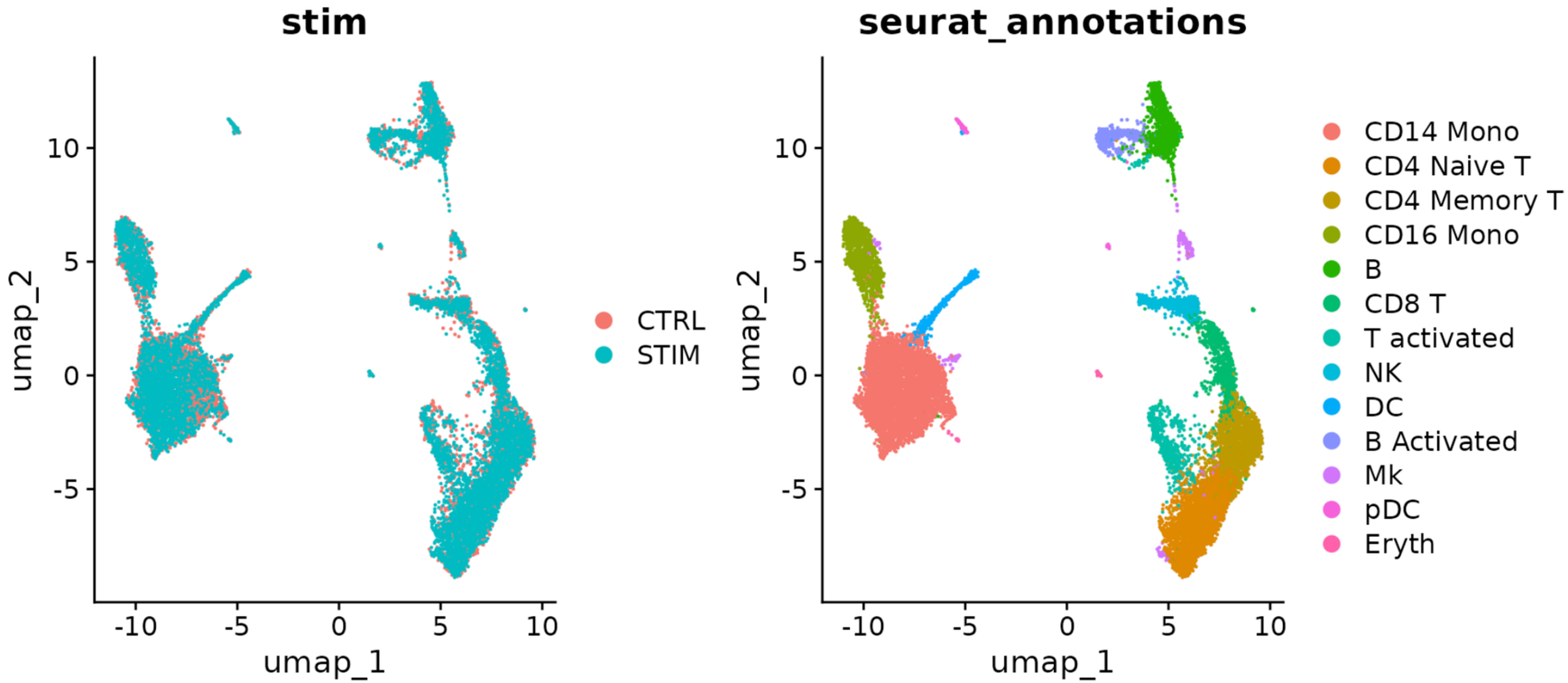
Seurat’s DE tests treat each cell as an independent replicate and ignore inherent correlations between cells originating from the same sample.

Pseudo-bulk Approach

Create a Pseudo-bulk Profile

For each cell type / cluster :

Sums together gene counts from all cells within a sample (cluster) and returns a ***gene x sample matrix*** instead of ***gene x cell matrix***



Pseudo-bulk Approach

Create a Pseudo-bulk Profile

For each cell type / cluster :

Sums together gene counts from all cells within a sample (cluster) and returns a ***gene x sample matrix*** instead of ***gene x cell matrix***

Returns summed counts ("pseudobulk") for each identity class.

```
AggregateExpression(
  object,
  assays = NULL,
  features = NULL,
  return.seurat = FALSE,
  group.by = "ident",
  add.ident = NULL,
  normalization.method = "LogNormalize",
  scale.factor = 10000,
  margin = 1,
  verbose = TRUE,
  ...
)
```

For each cell type

gene	ctrl_1	ctrl_2	exp_1	exp_1
geneA	10	11	56	45
geneB	0	0	128	54
geneC	42	41	59	41
geneD	103	122	1	23
geneE	10	23	14	56
geneF	0	1	2	0
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…

DESeq2

EdgeR

Differential Abundance Analysis

RESEARCH ARTICLE | BIOPHYSICS AND COMPUTATIONAL BIOLOGY |

sccomp: Robust differential composition and variability analysis for single-cell data

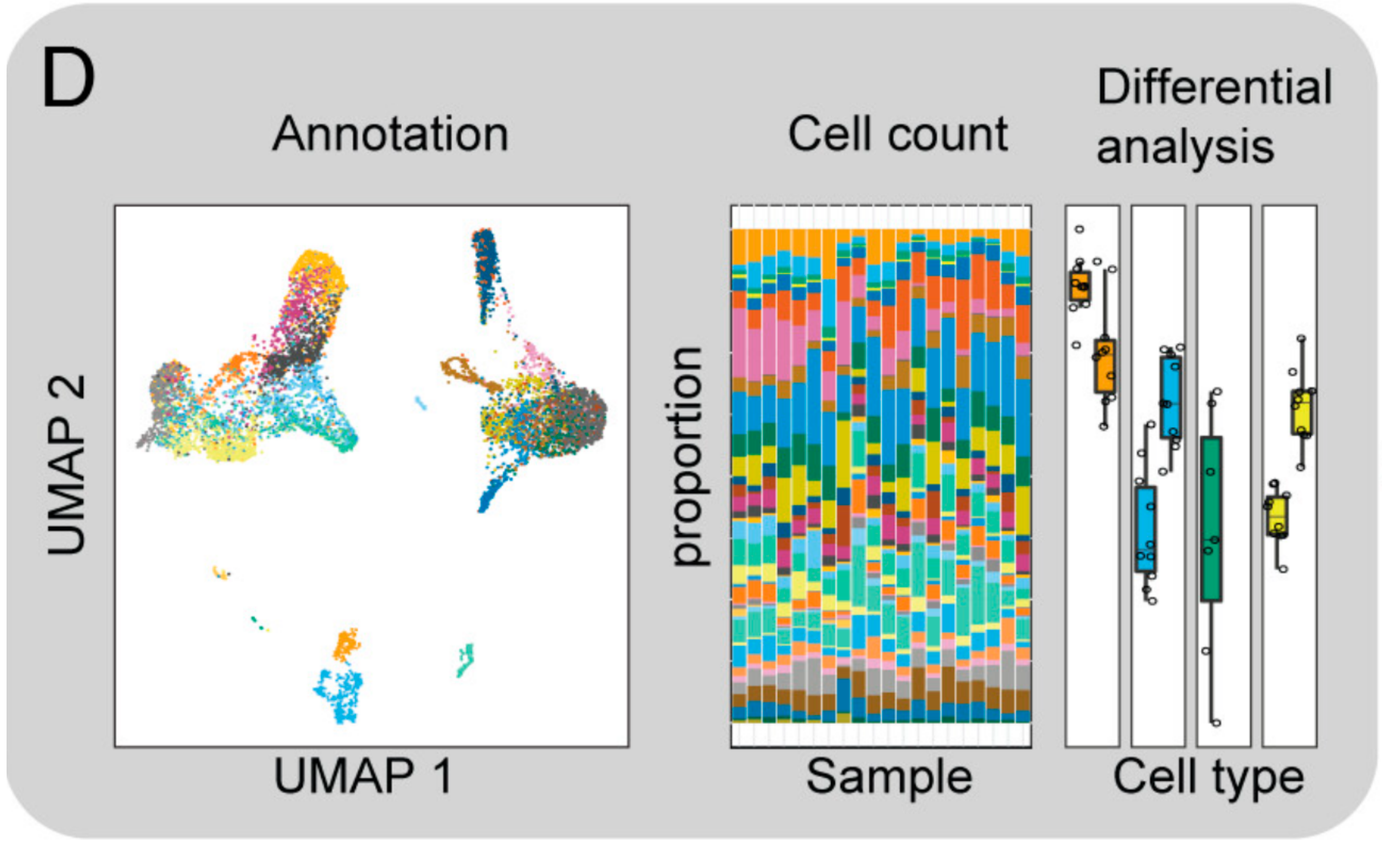
Stefano Mangiola, Alexandra J. Roth-Schulze, Marie Trussart, +6, and Anthony T. Papenfuss Authors Info & Affiliations

Edited by Simon Tavare, University of Cambridge, Cambridge, United Kingdom; received March 5, 2022; accepted May 18, 2023

August 7, 2023 | 120 (33) e2203828120 | https://doi.org/10.1073/pnas.2203828120

THIS ARTICLE HAS BEEN UPDATED

Tests to check if proportion of cells within clusters are associated with a particular phenotype e.g. cases v ctrls



Gene Set Enrichment Analysis

Software/Web server Article

UCell: Robust and scalable single-cell gene signature scoring

Massimo Andreatta [a,b,*], Santiago J. Carmona [a,b]

[a] Ludwig Institute for Cancer Research, Lausanne Branch, and Department of Oncology, CHUV and University of Lausanne, Epalinges 1066, Switzerland
[b] Swiss Institute of Bioinformatics, Lausanne, Switzerland

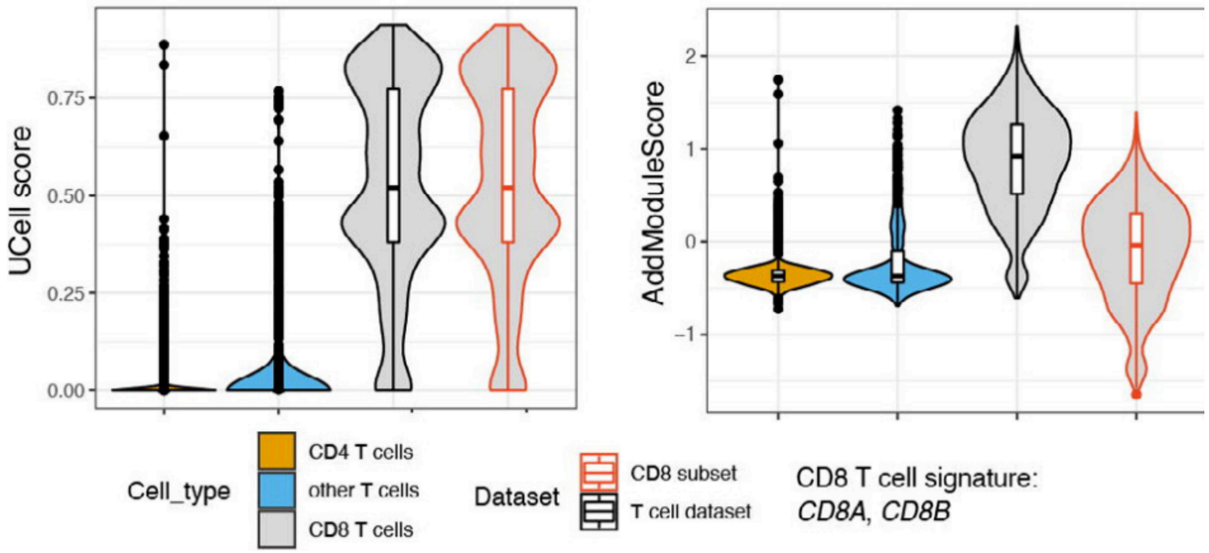
ARTICLE INFO

Article history:
Received 11 May 2021
Received in revised form 22 June 2021
Accepted 22 June 2021
Available online 30 June 2021

Keywords:
Single-cell
Gene signature
Module scoring
Cell type
Gene set enrichment

ABSTRACT

UCell is an R package for evaluating gene signatures in single-cell datasets. UCell signature scores, based on the Mann-Whitney U statistic, are robust to dataset size and heterogeneity, and their calculation demands less computing time and memory than other available methods, enabling the processing o large datasets in a few minutes even on machines with limited computing power. UCell can be applie to any single-cell data matrix, and includes functions to directly interact with Seurat objects. The UCel package and documentation are available on GitHub at https://github.com/carmonalab/UCell.



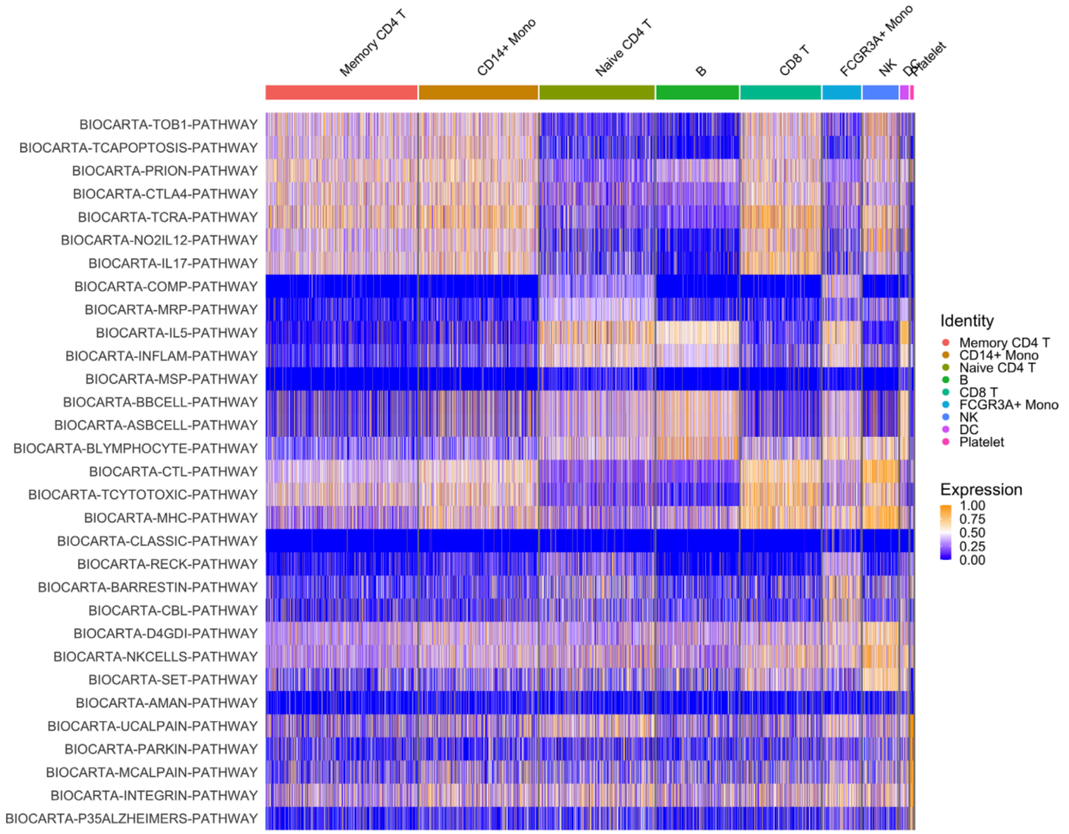
Published online 7 July 2020

Nucleic Acids Research, 2020, Vol. 48, No. 16 ***e94***
doi: 10.1093/nar/gkaa582

Variance-adjusted Mahalanobis (VAM): a fast and accurate method for cell-specific gene set scoring

Hildreth Robert Frost *

Department of Biomedical Data Science, Geisel School of Medicine, Dartmouth College, Hanover, NH 03755, USA



Automated Cell Type Labeling with a reference

Seurat

Azimuth is a web application that uses an annotated reference dataset to **automate the processing, analysis, and interpretation of a new single-cell RNA-seq or ATAC-seq experiment**. Azimuth leverages a **'reference-based mapping'** pipeline that inputs a counts matrix and performs normalization, visualization, cell annotation, and differential expression (biomarker discovery). All results can be explored within the app, and easily downloaded for additional downstream analysis.

The development of Azimuth is led by the New York Genome Center Mapping Component as part of the NIH Human Biomolecular Atlas Project (HuBMAP). Thirteen molecular reference maps are currently available, with more coming soon.

SingleR

Article | Published: 14 January 2019

Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage

Dvir Aran, Agnieszka P. Looney, Leqian Liu, Esther Wu, Valerie Fong, Austin Hsu, Suzanna Chak, Ram P. Naikawadi, Paul J. Wolters, Adam R. Abate, Atul J. Butte & Mallar Bhattacharya

Nature Immunology **20**, 163–172 (2019) | Cite this article

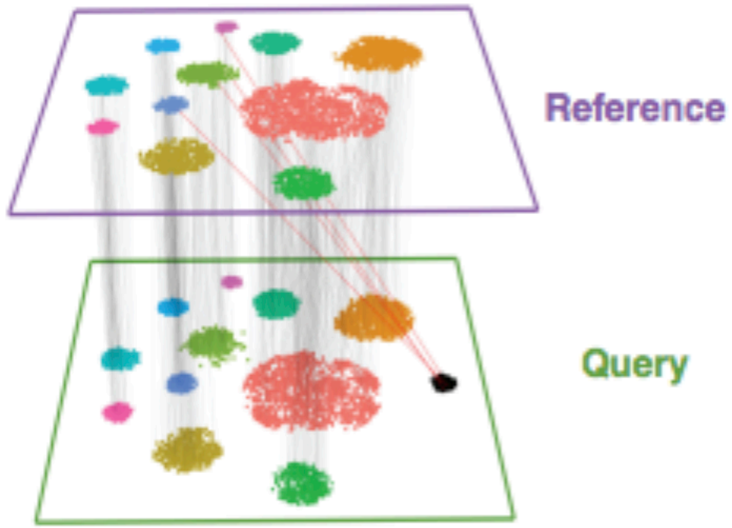
scArches

Analysis | Open access | Published: 30 August 2021

Mapping single-cell data to reference atlases by transfer learning

Mohammad Lotfollahi, Mohsen Naghipourfar, Malte D. Luecken, Matin Khajavi, Maren Büttner, Marco Wagenstetter, Žiga Avsec, Adam Gayoso, Nir Yosef, Marta Interlandi, Sergei Rybakov, Alexander V. Misharin & Fabian J. Theis

Nature Biotechnology **40**, 121–130 (2022) | Cite this article



Cell-cell Communication analysis for predicting interactions between cell types

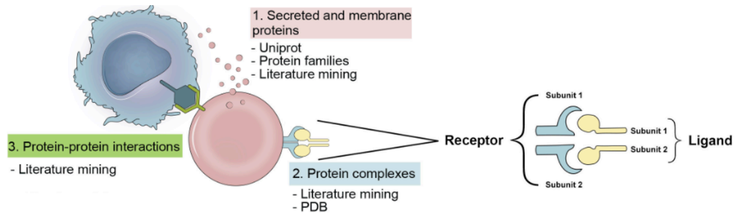
› Nat Protoc. 2020 Apr;15(4):1484-1506. doi: 10.1038/s41596-020-0292-x. Epub 2020 Feb 26.

CellPhoneDB: inferring cell–cell communication from combined expression of multi–subunit ligand–receptor complexes

Mirjana Efremova [1], Miquel Vento-Tormo [2], Sarah A Teichmann [1 3], Roser Vento-Tormo [4]

Affiliations + expand

PMID: 32103204 DOI: 10.1038/s41596-020-0292-x

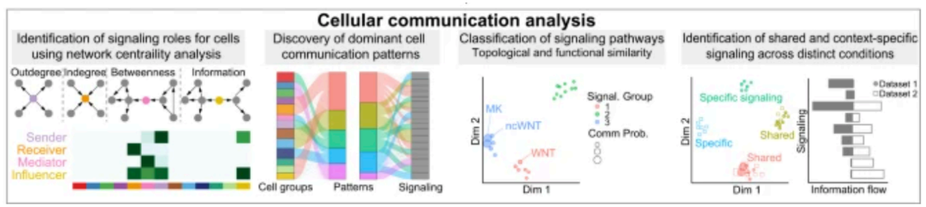


Article | Open access | Published: 17 February 2021

Inference and analysis of cell-cell communication using CellChat

Suoqin Jin, Christian F. Guerrero-Juarez, Lihua Zhang, Ivan Chang, Raul Ramos, Chen-Hsiang Kuan, Peggy Myung, Maksim V. Plikus & Qing Nie

Nature Communications **12**, Article number: 1088 (2021) | Cite this article



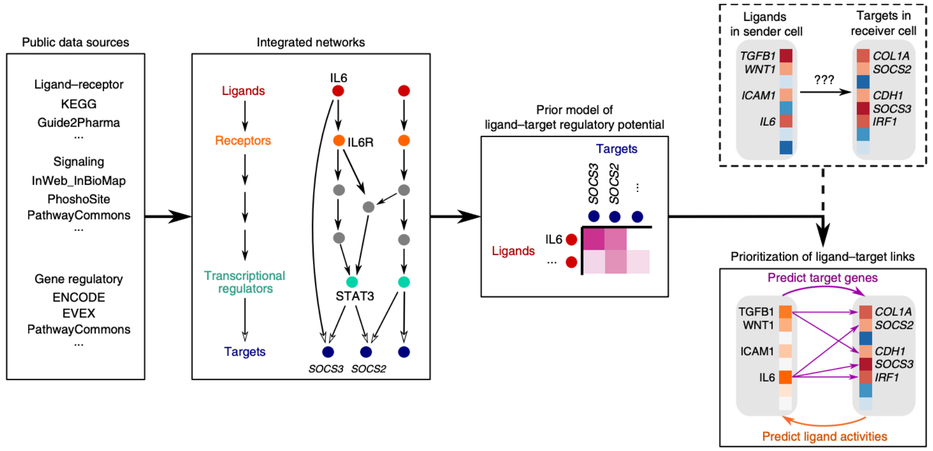
nature methods

BRIEF COMMUNICATION

https://doi.org/10.1038/s41592-019-0667-5

NicheNet: modeling intercellular communication by linking ligands to target genes

Robin Browaeys [1,2], Wouter Saelens [1,2,3] and Yvan Saeys [1,2,3*]



4. Interactive Analysis

Automated Cell Type Labeling with a reference

Seurat

Azimuth is a web application that uses an annotated reference dataset to **automate the processing, analysis, and interpretation of a new single-cell RNA-seq or ATAC-seq experiment**. Azimuth leverages a **'reference-based mapping'** pipeline that inputs a counts matrix and performs normalization, visualization, cell annotation, and differential expression (biomarker discovery). All results can be explored within the app, and easily downloaded for additional downstream analysis.

The development of Azimuth is led by the New York Genome Center Mapping Component as part of the NIH Human Biomolecular Atlas Project (HuBMAP). Thirteen molecular reference maps are currently available, with more coming soon.

SCTK