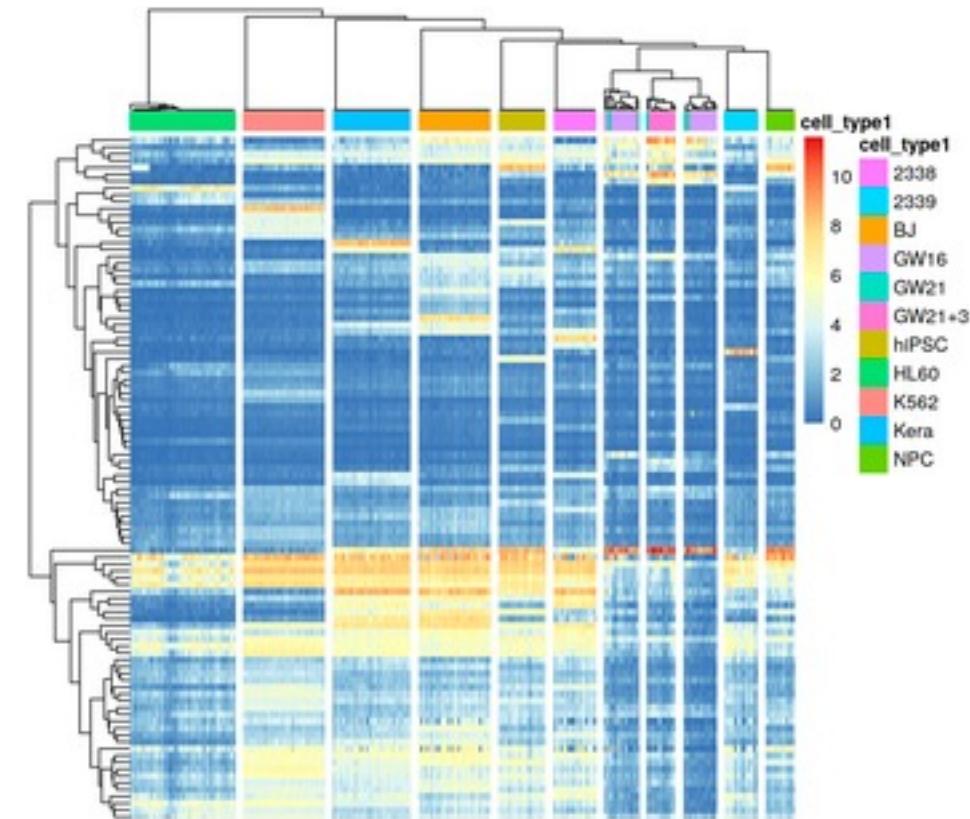
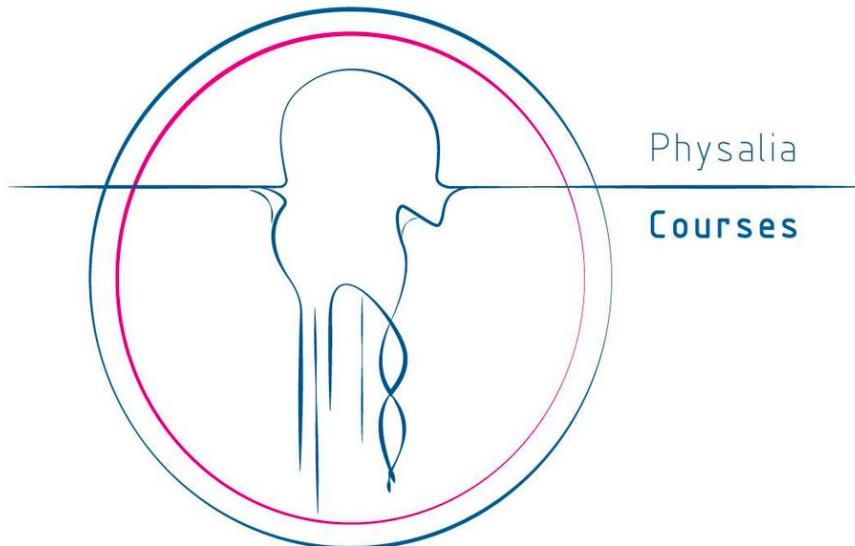


Analysis of Single Cell RNA-Seq Data: Data integration and batch effect correction

Orr Ashenberg, Jacques Serizay, Fabricio Almeida-Silva

November 6, 2024

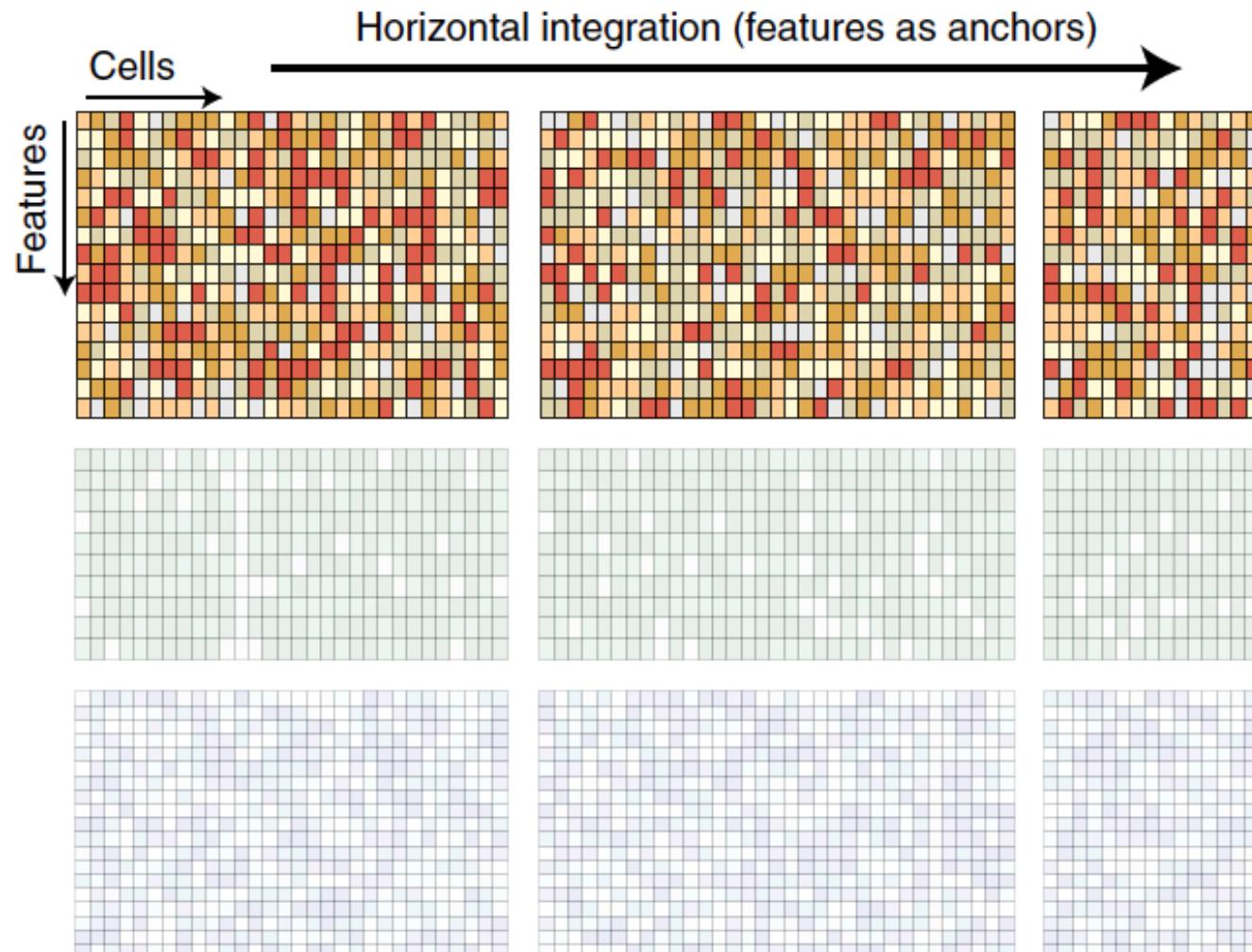
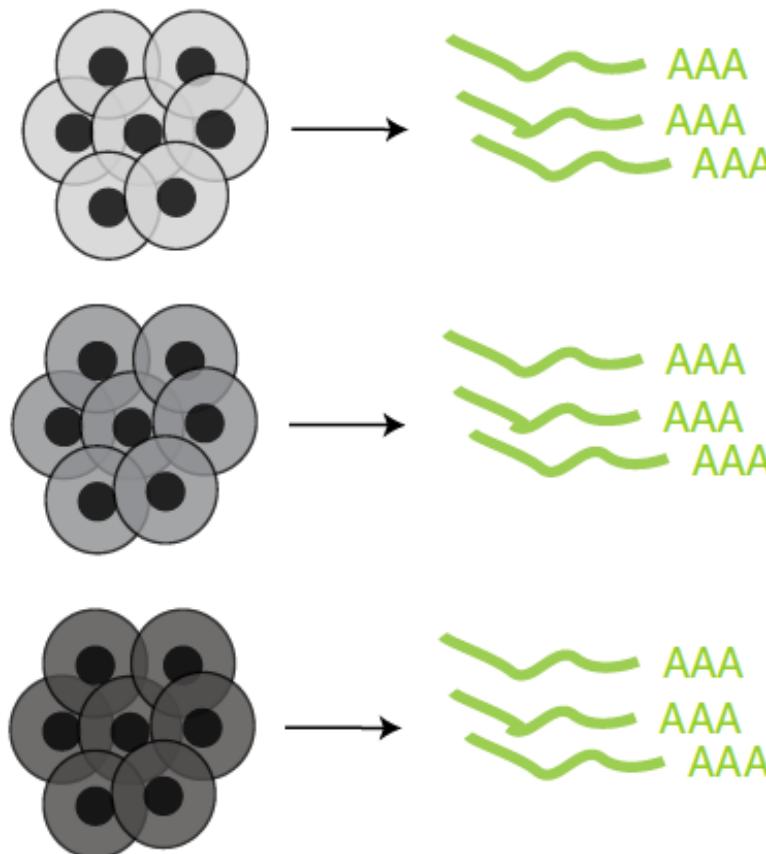


Outline: Batch effects and data integration

- Types of data integration.
- Where batch effects come from.
- Computational approaches to correct for batch effects and integrate data.
- Making biological comparisons after data integration.

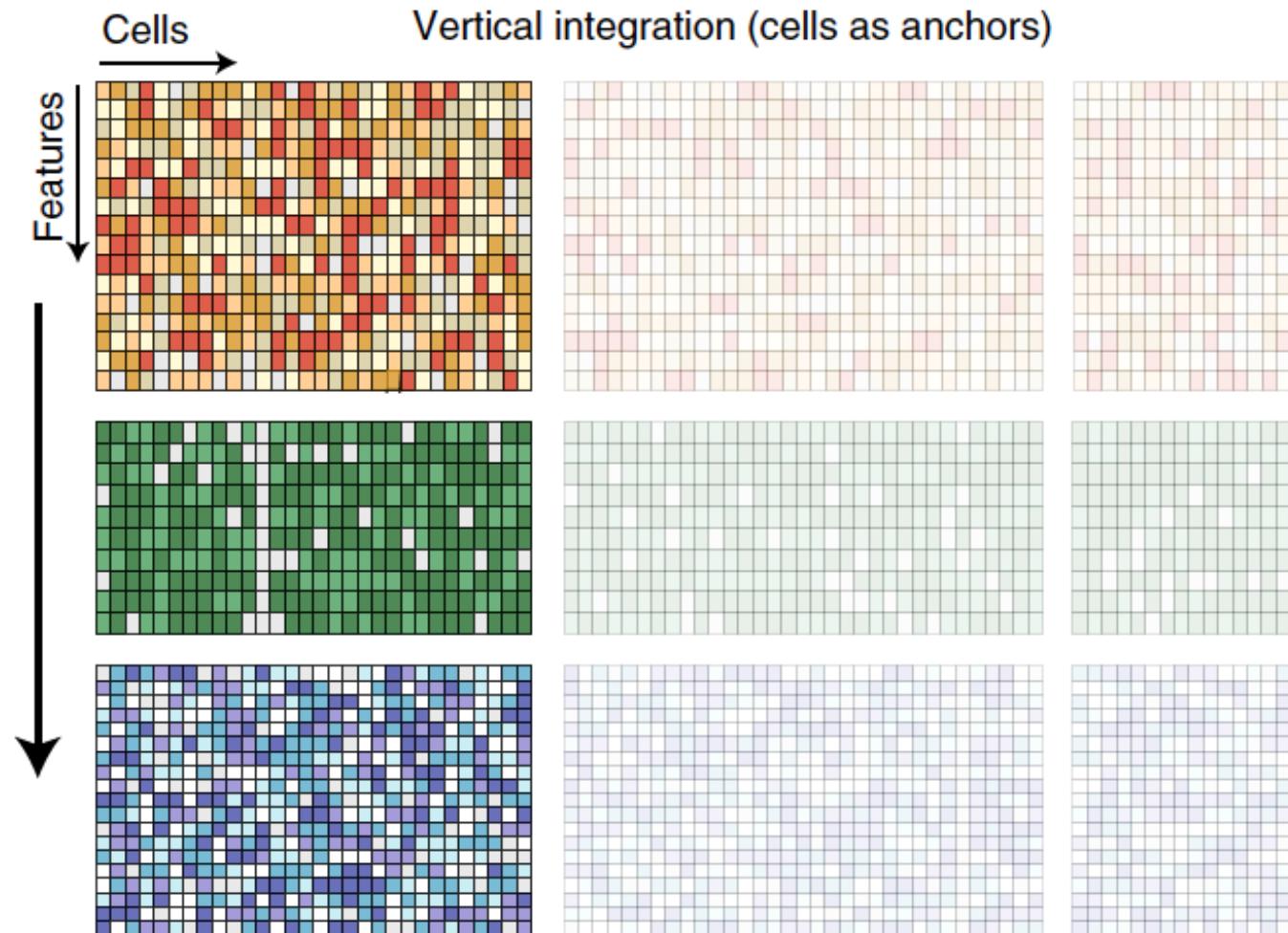
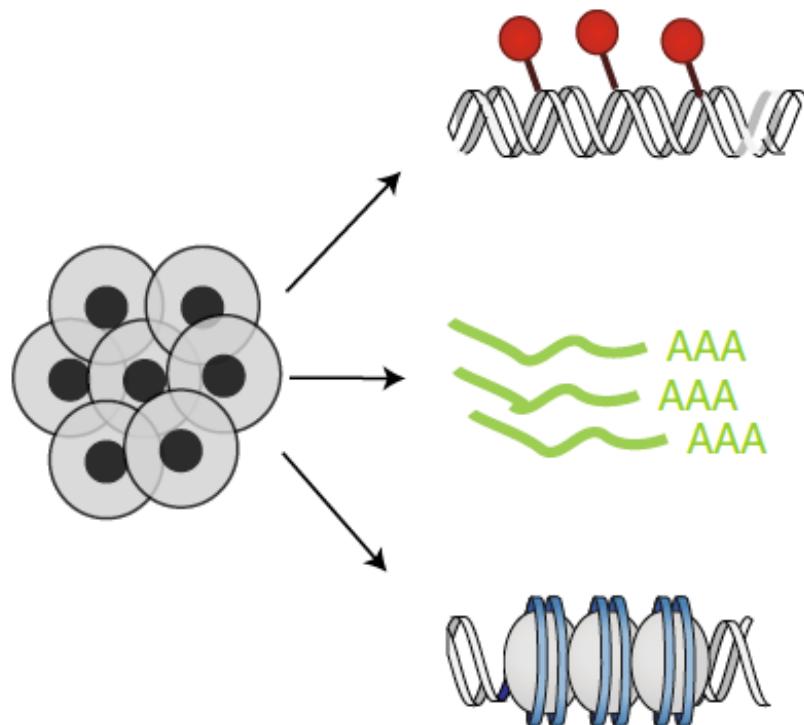
Data integration

Shared features across different batches of cells



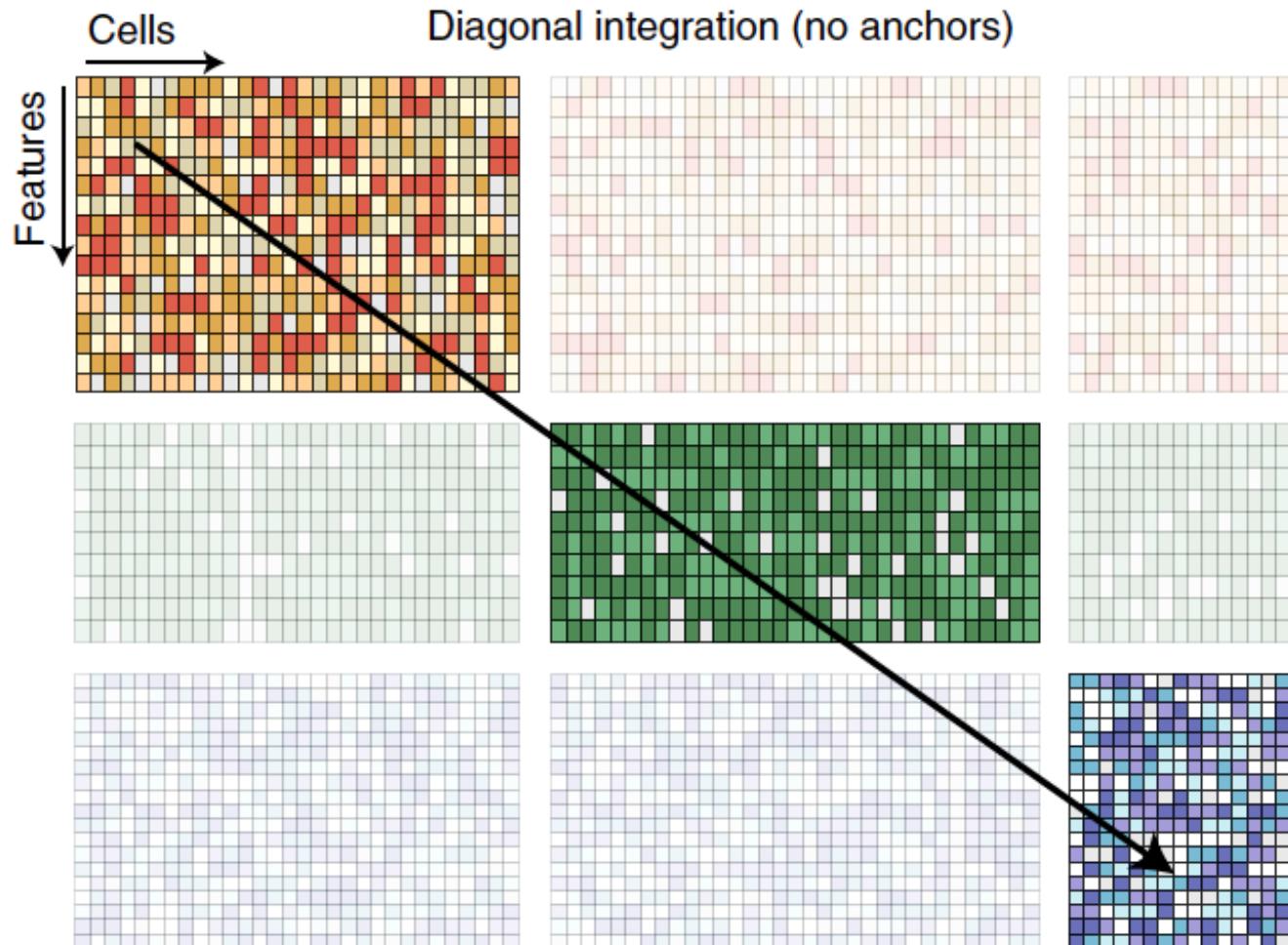
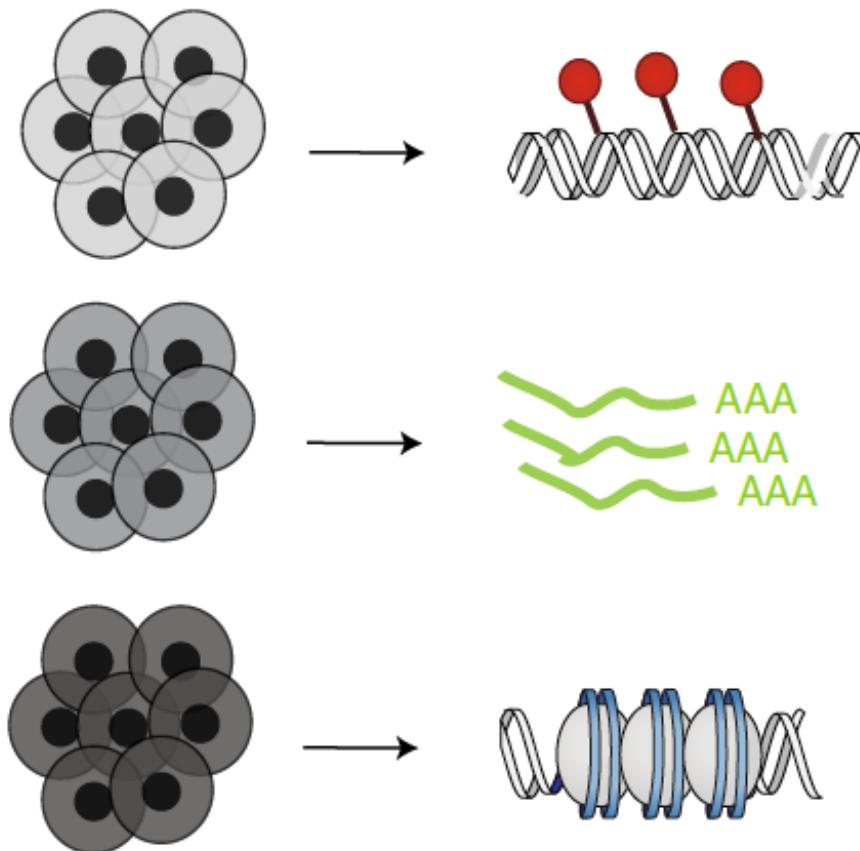
Data integration

Different features measured from same cells



Data integration

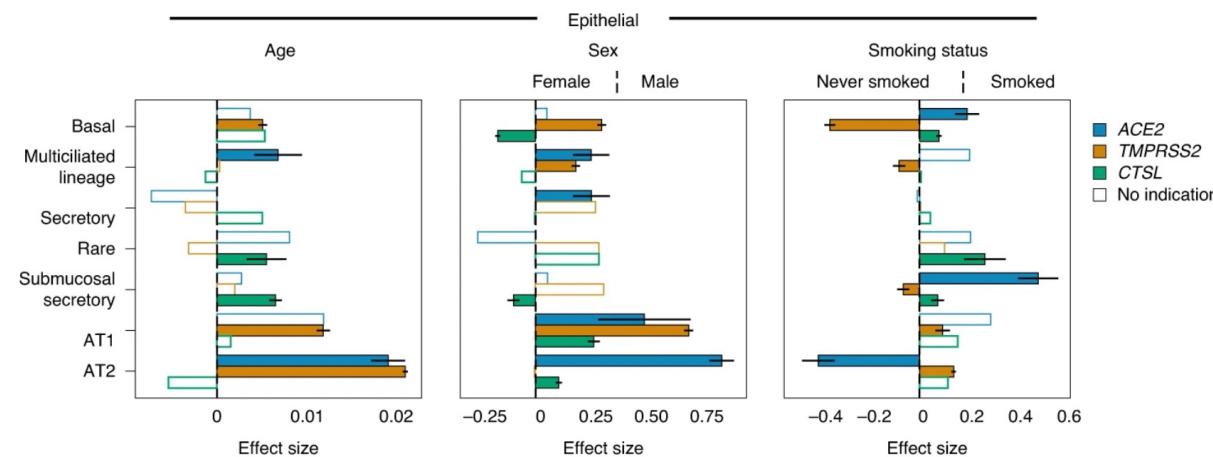
Different cell batches and features in each experiment



Why perform data integration and batch effect correction?

We often require large cohorts of individuals ($n > 10$) to make biological comparisons (case vs control, mutant vs wildtype etc...).

- As an example, how do smoking status and age contribute to SARS-CoV-2 entry-related genes in lung cells?

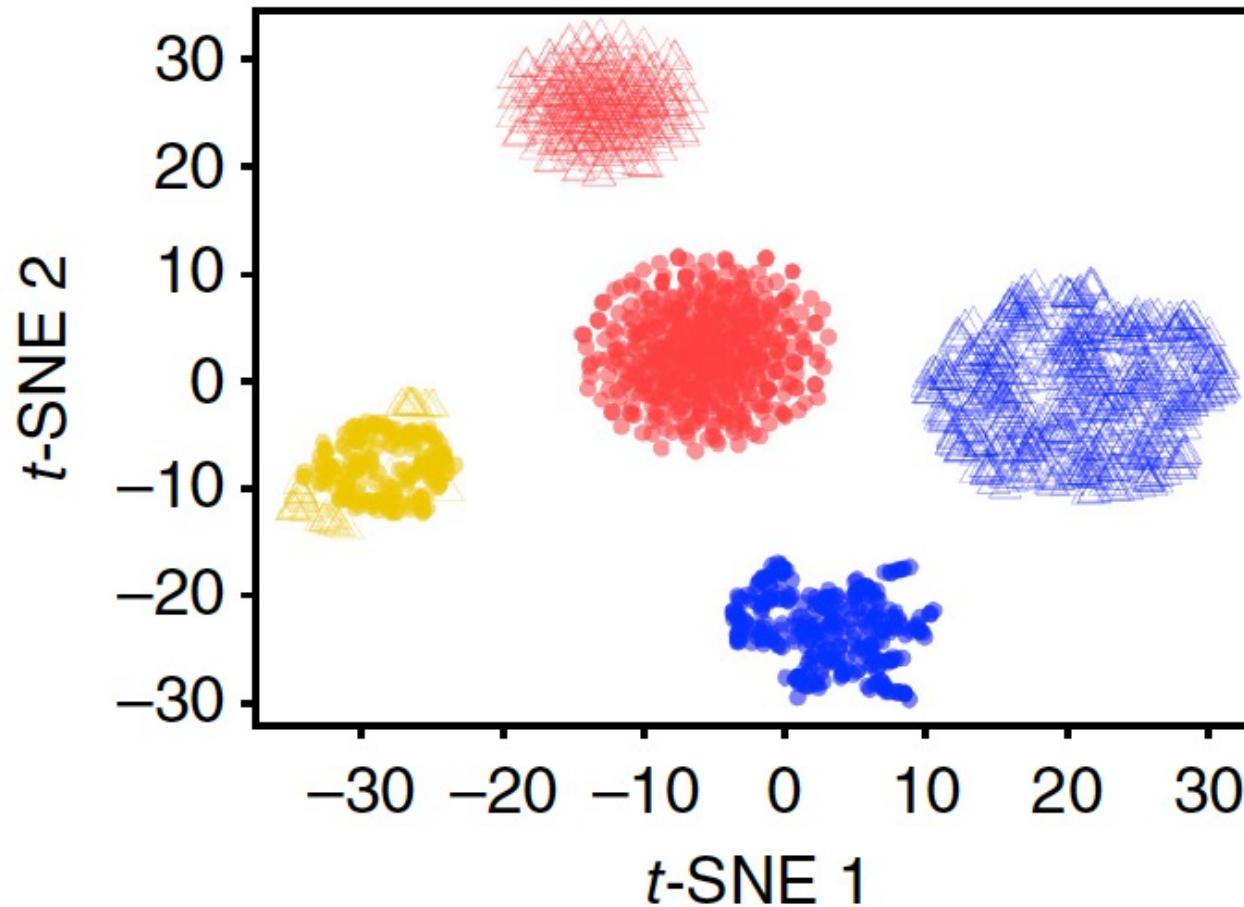


1,320,896 cells from 228 individuals

Challenges

- Technical batch effects confound biological variation of interest (e.g. site and time of collection, experimental protocol used...)
- Different sources of biological variation confound biological variation of interest (e.g. cell cycle, treatment status...)

Distinguishing biological effects from technical, batch effects is a difficult problem



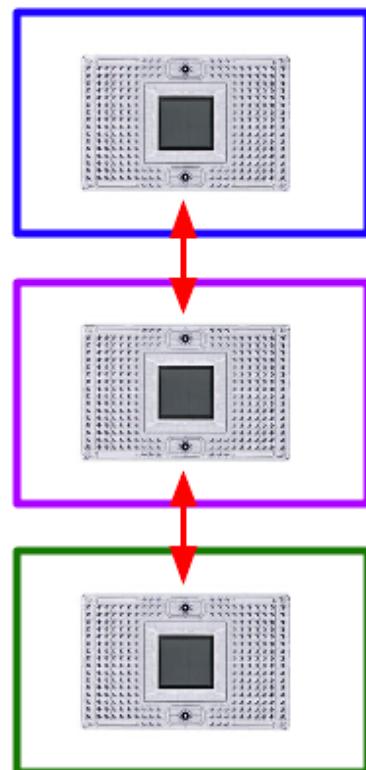
Cells are colored by cell type.

Symbols represent different batches.

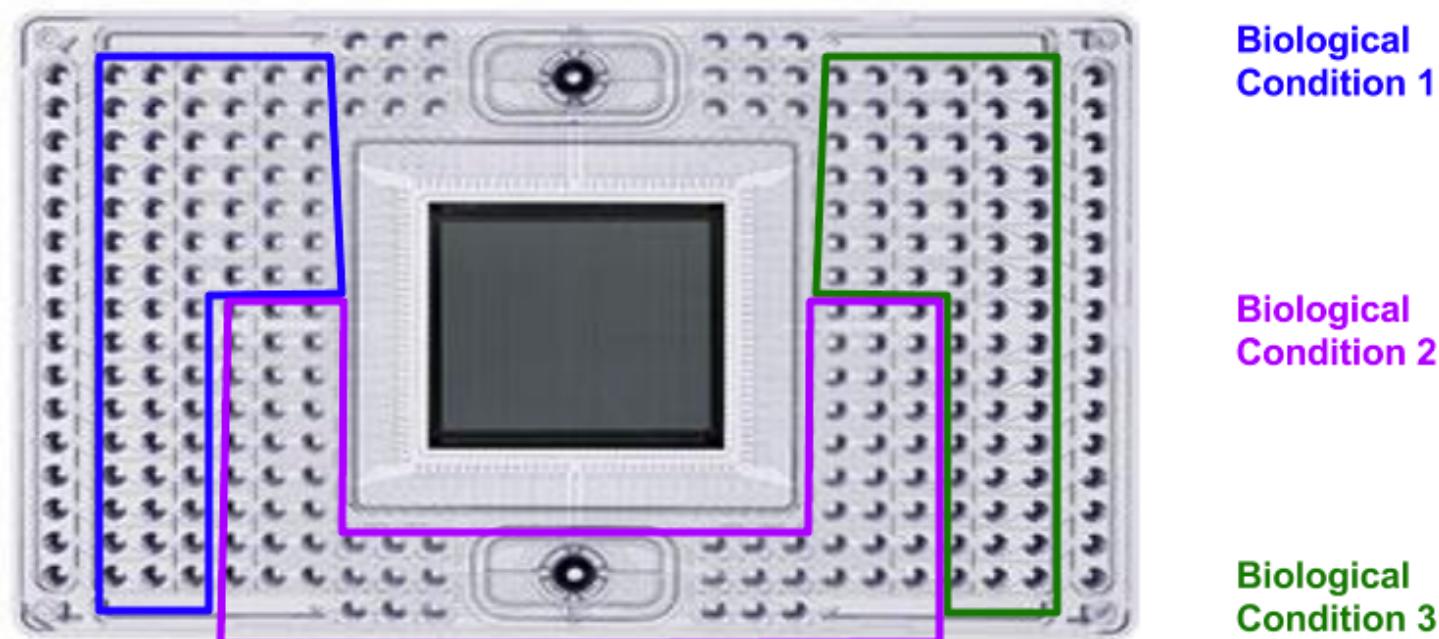
Correcting for batch effects allows us to combine datasets and boost biological signal, while reducing technical confounders

The most powerful way to control batch effects is with careful experimental design

Completely Confounded



Unconfounded



Sound experimental design : Replication, Randomization and Blocking
- R. A. Fisher, 1935

Batch effects: technical sources

- Differences in how samples are sequenced
 - sequencing depth and saturation
 - sequencing instrument



Miseq

~ 20M reads total



Nextseq

~ 500M reads total

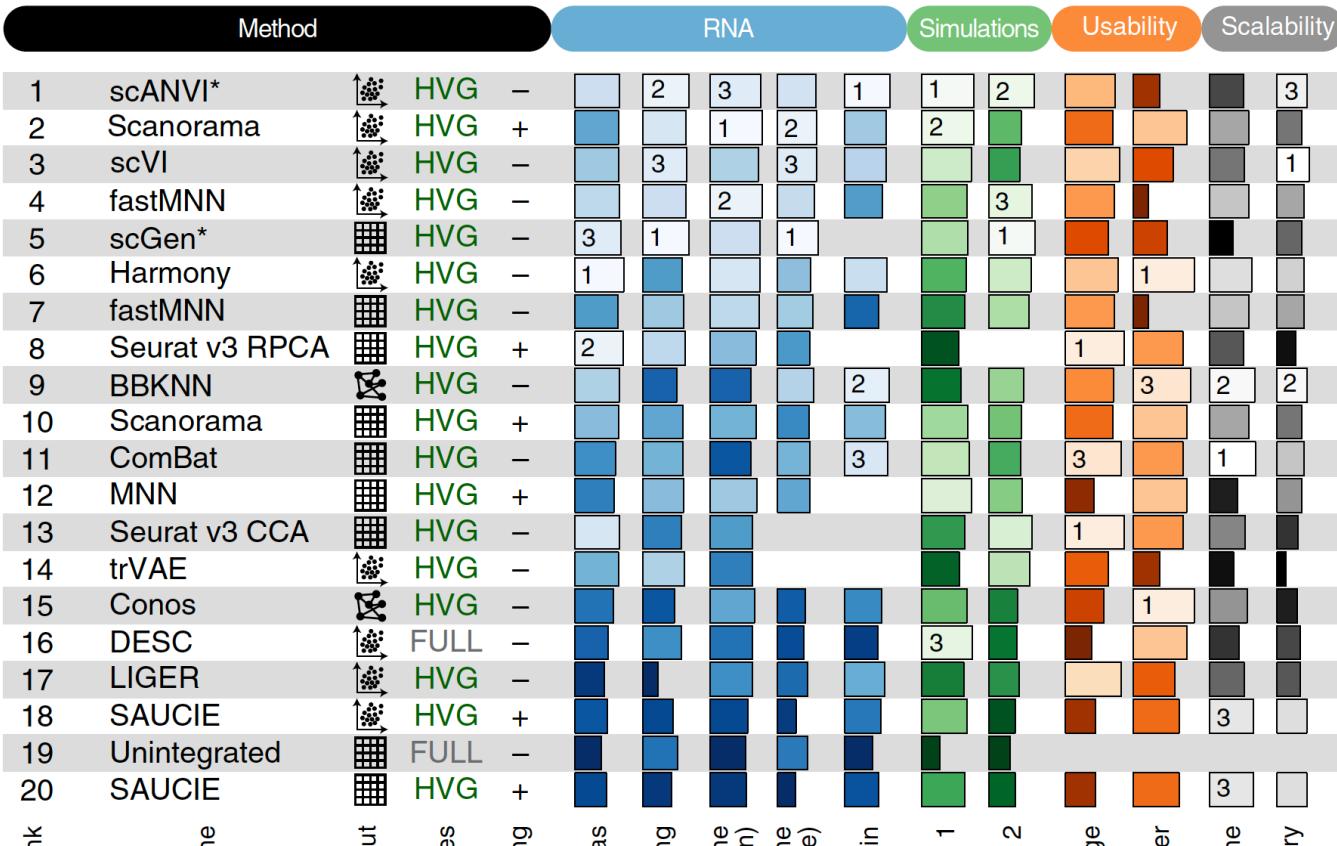
HiSeq 4000

4 billion reads

A few basic approaches to batch correction

- Down sampling of sequencing reads
- Normalization
- Using variable genes common to multiple samples
- Removing genes correlated with batch
- Regression of residuals with technical covariates
 - batch id
 - number of UMI per cell
 - number of genes per cell
 - % mitochondrial reads
- ComBat (developed for microarray experiments)

Benchmarking of data integration methods



Output
 Genes
 Embedding
 Graph

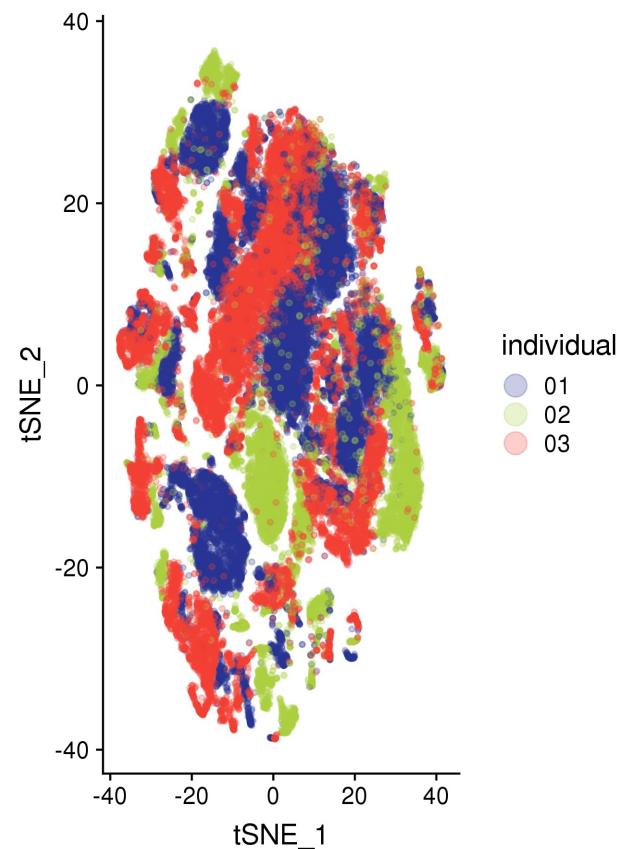
Scaling
+ Scaled
- Unscaled

Ranking

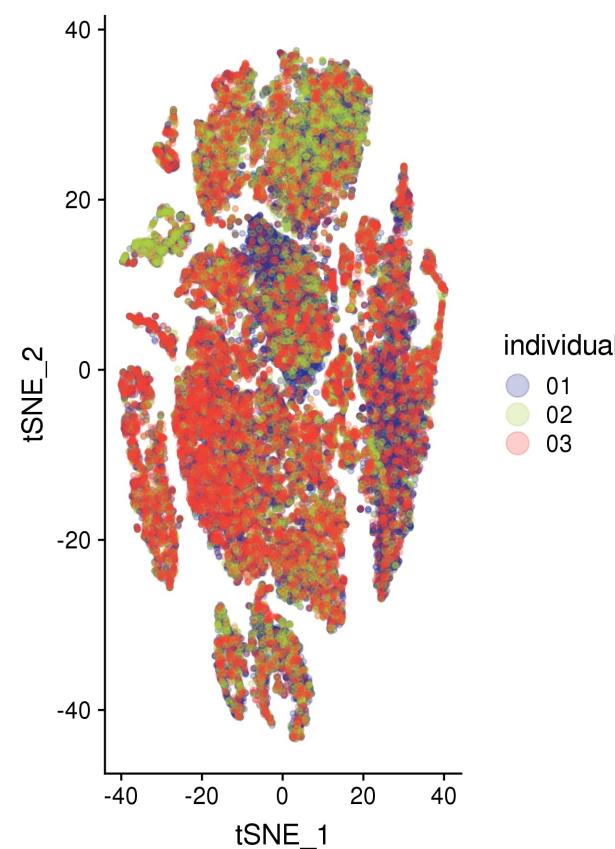
↑ 1
↓ 20

Batch correction and data modality integration

Batch effects often arise when patient samples are analyzed together



batch correction
→

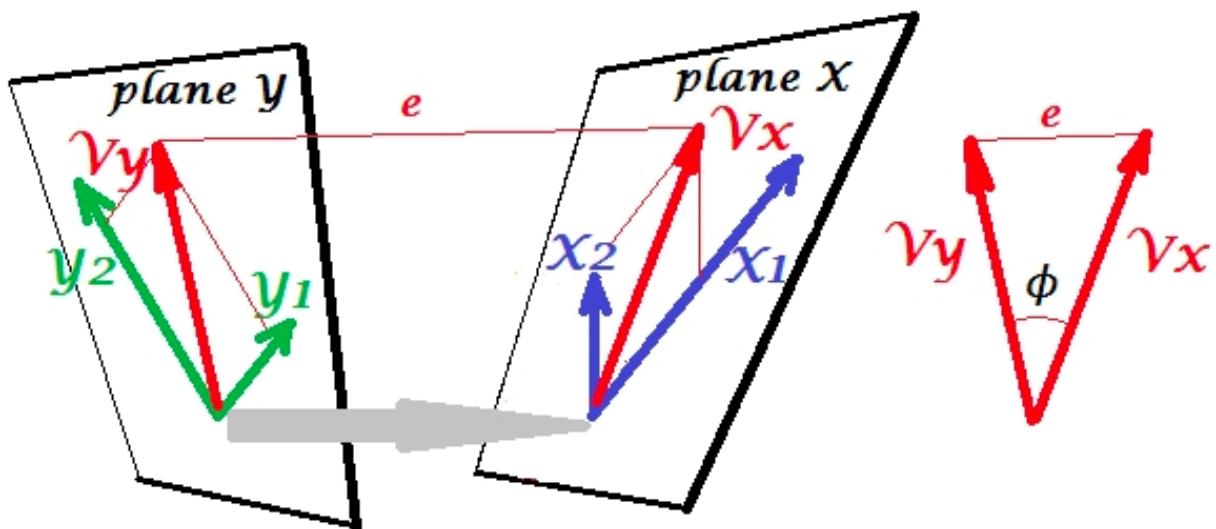


Batch correction and data modality integration

- Seurat v3
- LIGER (Linked Inference of Genomic Experimental Relationships)
- Conos (Clustering on Network of Samples), Harmony
- scVI (deep learning approach)

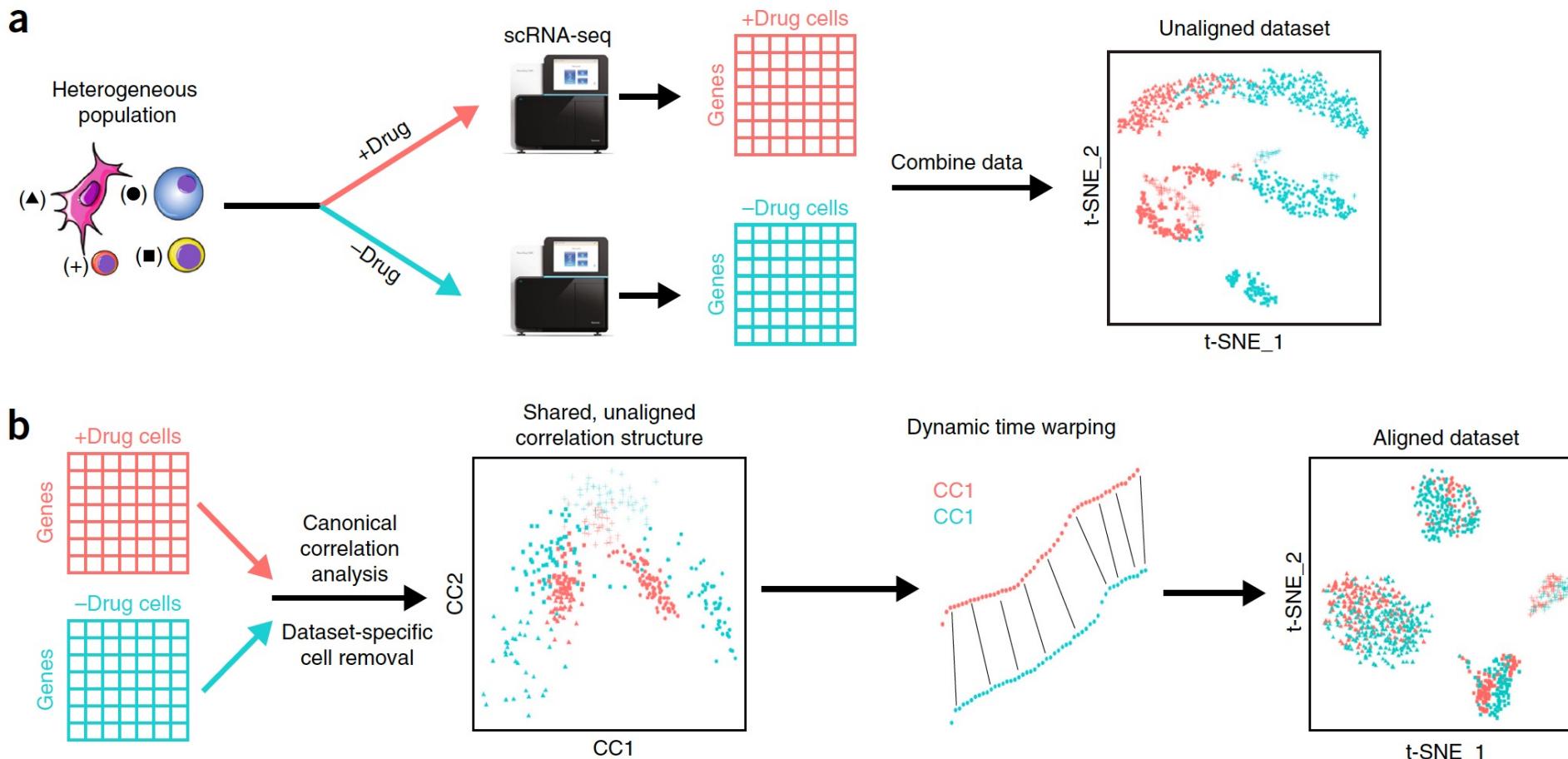
Canonical Correlation Analysis (CCA)

- CCA finds the linear combinations of variables across two datasets that are maximally correlated with one another.
 - The first pair of canonical variables maximizes the correlation across datasets.
 - The second pair of canonical variables maximizes the correlation subject to the constraint of not being correlated with the first pair, and so on.



- Goals of CCA
 - Similar to Principal Components Analysis (PCA)
 - Dimensional reduction: explain covariation between datasets with a small number of linear combinations of variables

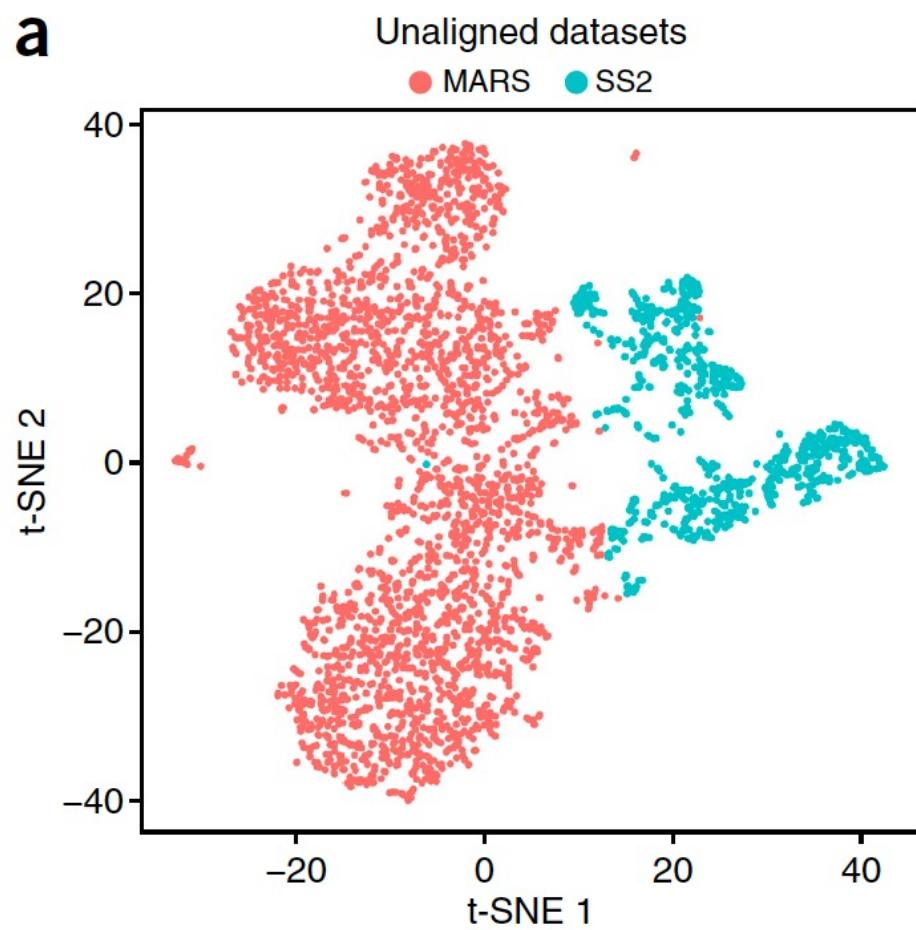
Canonical Correlation Analysis (CCA)



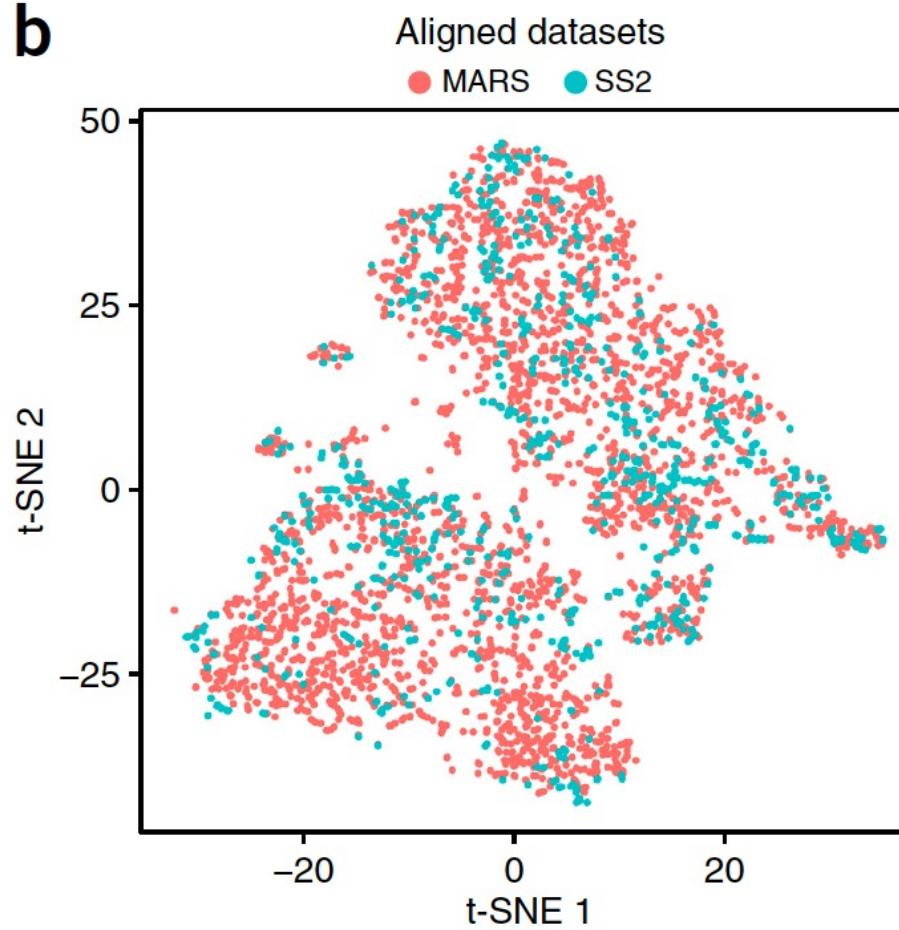
"Effectively, we treat the data sets as multiple measurements of a gene–gene covariance structure, and search for patterns that are common to the data sets."

Canonical Correlation Analysis (CCA)

a



b

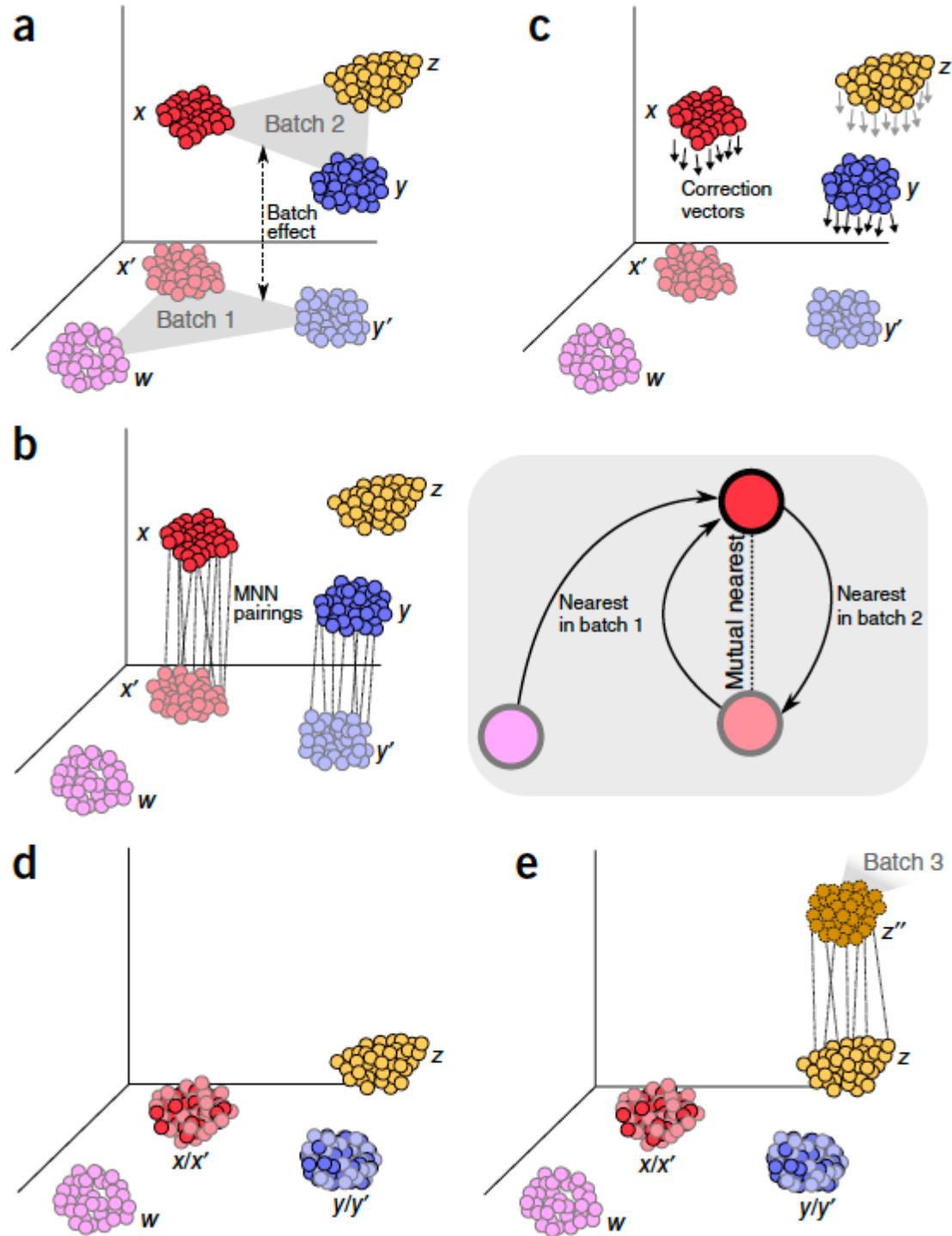


Assessing the performance of batch correction

"For every cell, we calculate how many of its k nearest-neighbors belong to the same data set and average this over all cells. If the data sets are well-aligned, we would expect that each cells' nearest neighbors would be evenly shared across all data sets."

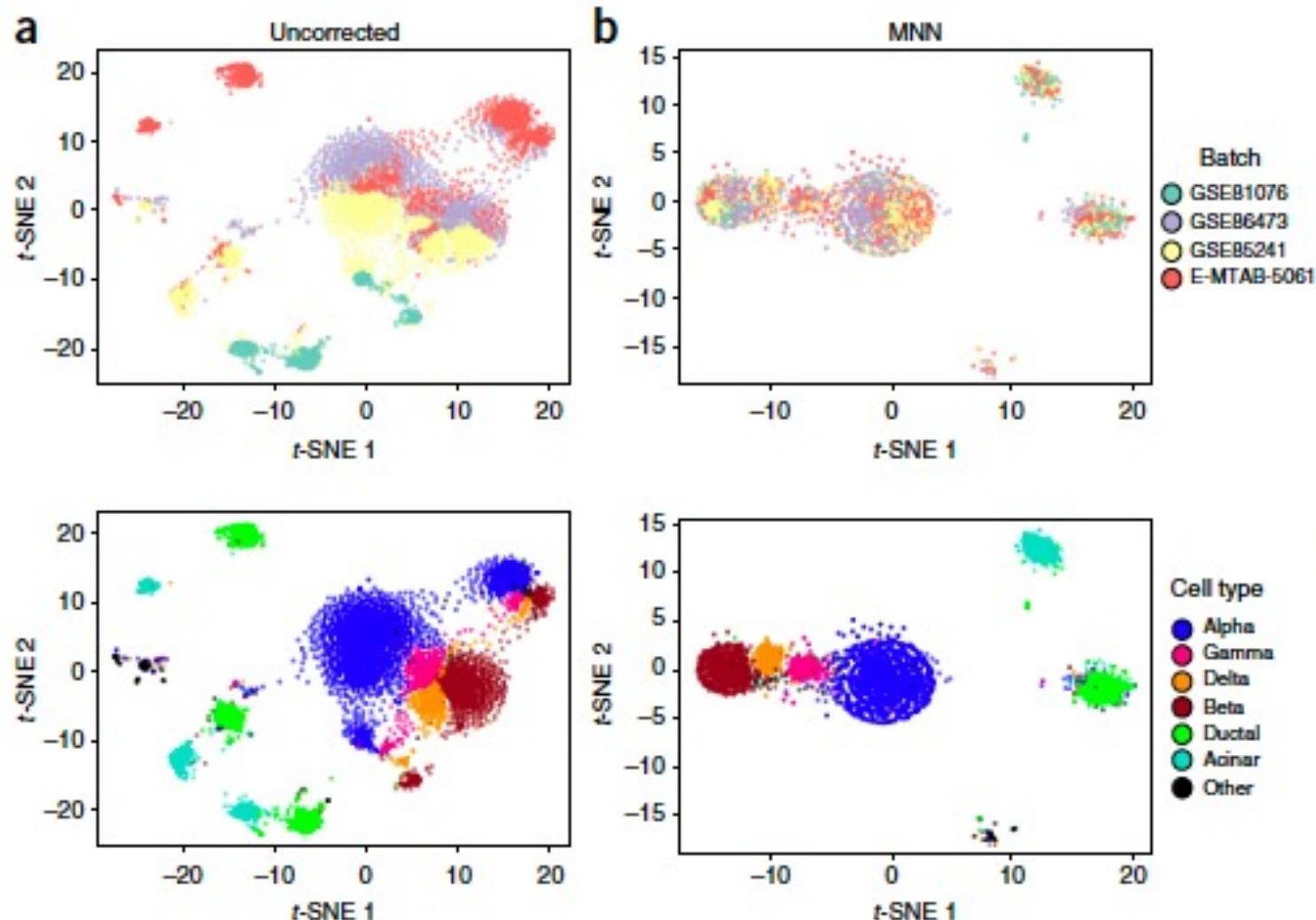
Mutual Nearest Neighbors

"If a pair of cells from each batch is contained in each other's set of nearest neighbors, those cells are considered to be mutual nearest neighbors. We interpret these pairs as containing cells that belong to the same cell type or state despite being generated in different batches. Thus, any systematic differences in expression level between cells in MNN pairs should represent the batch effect."

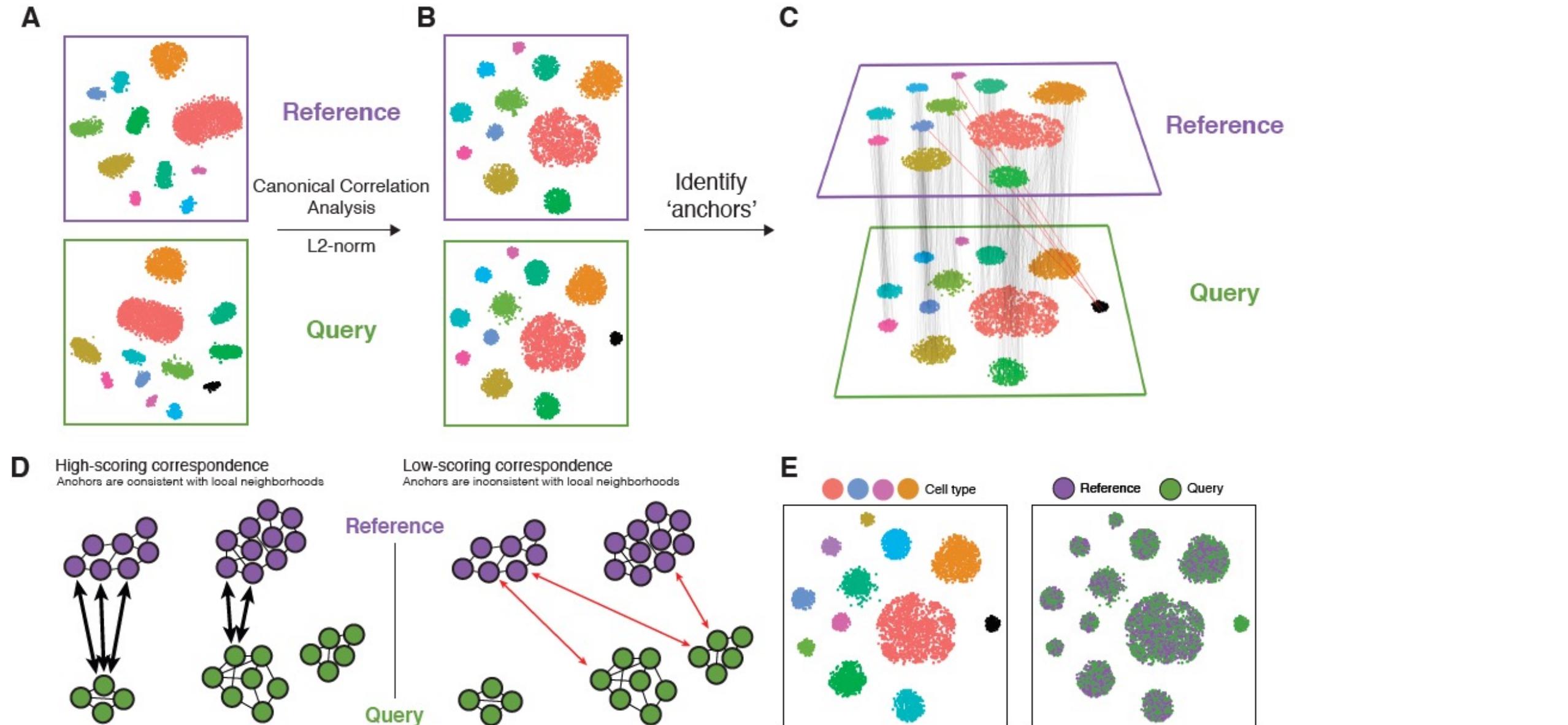


Mutual Nearest Neighbors

4 pancreas datasets



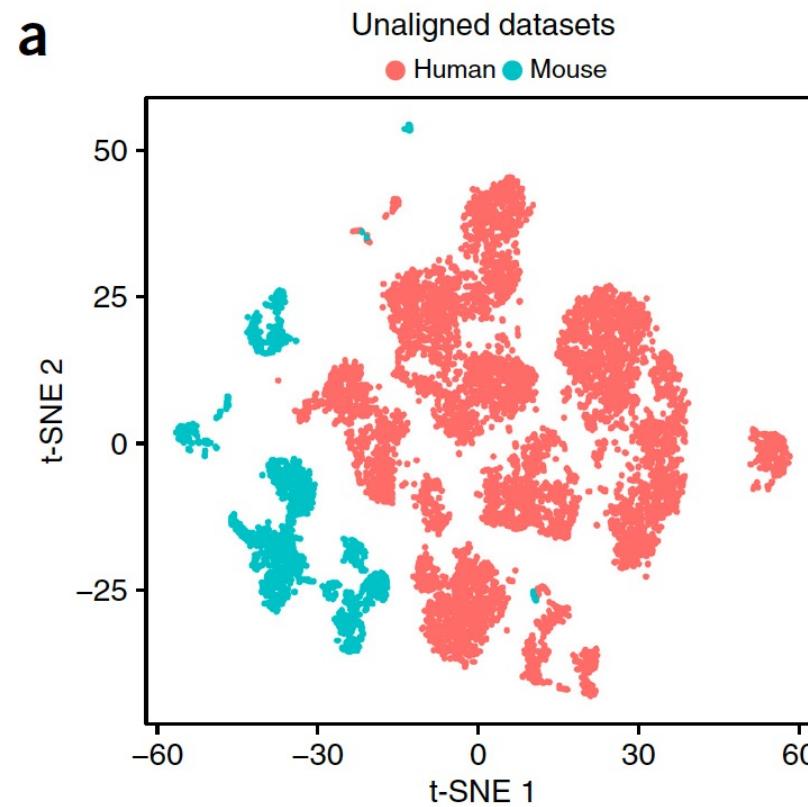
Combining CCA and Mutual Nearest Neighbors (Seurat v3)



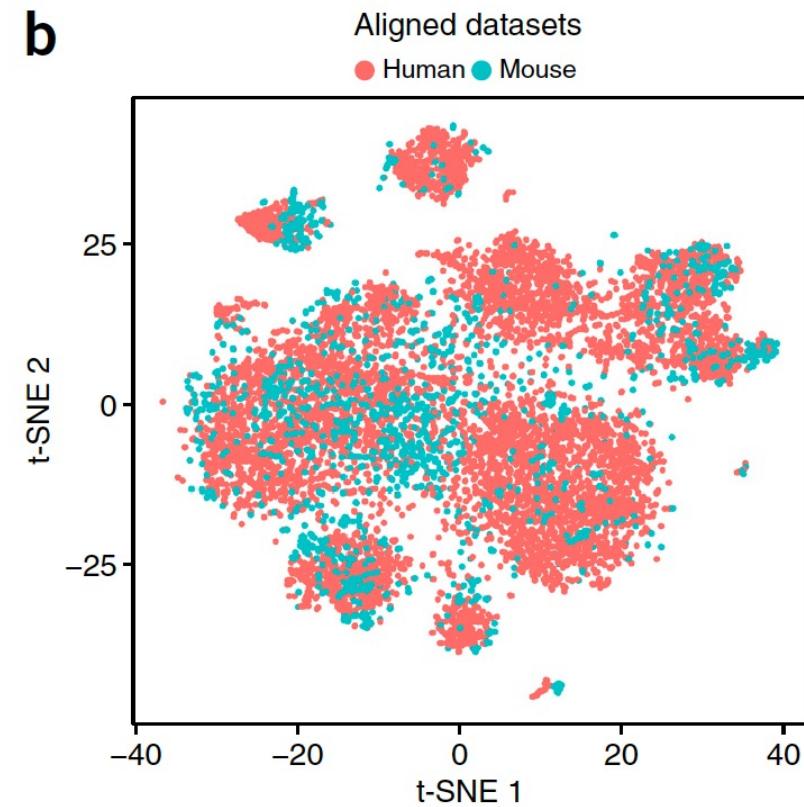
Combining CCA and Mutual Nearest Neighbors (Seurat v3)

Human and mouse pancreas datasets

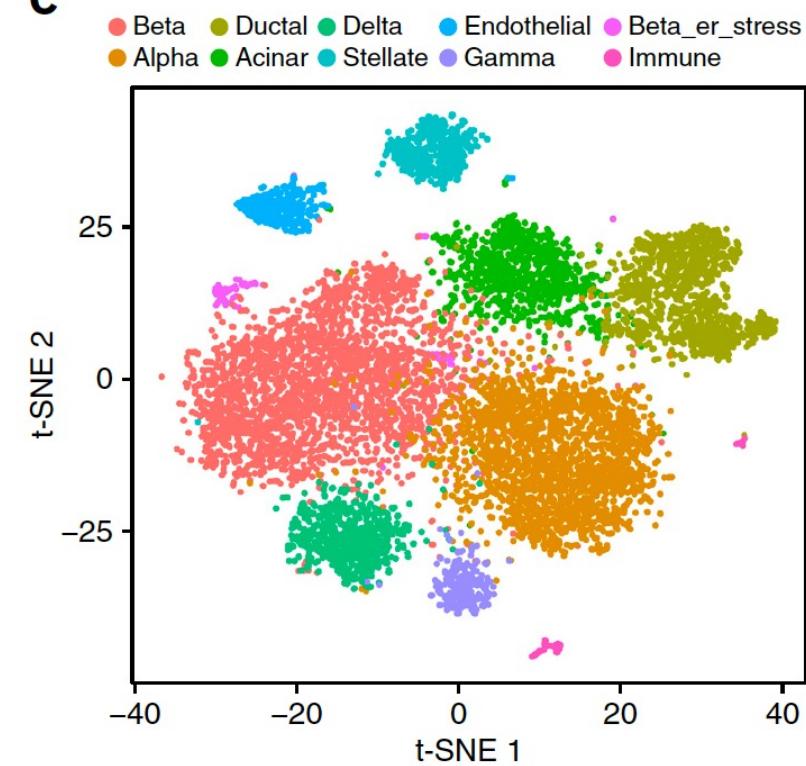
a



b



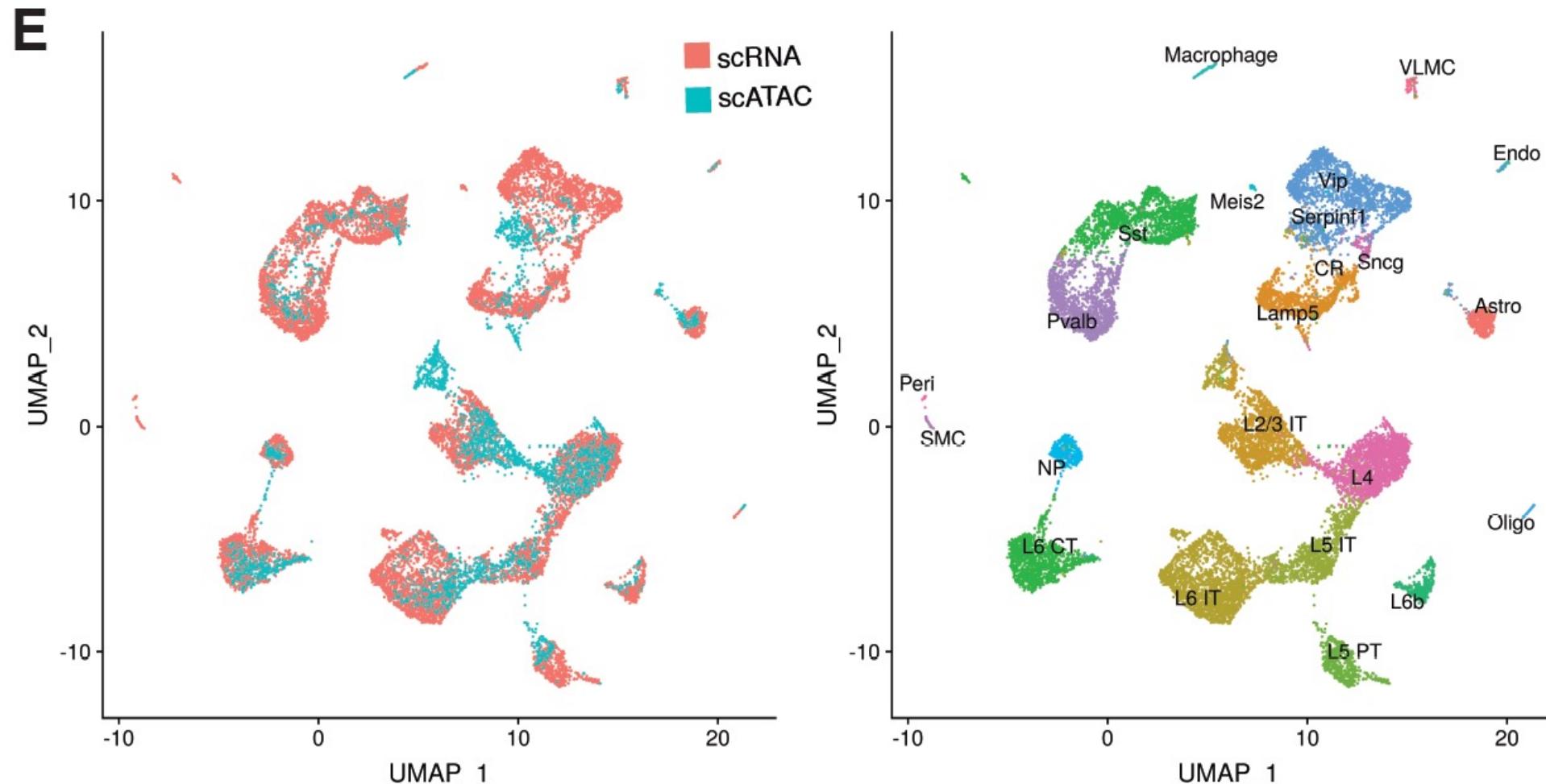
c



Classifying nuclei from a single cell ATAC-seq experiment using single cell RNA-seq data as a reference

Integrating data modalities

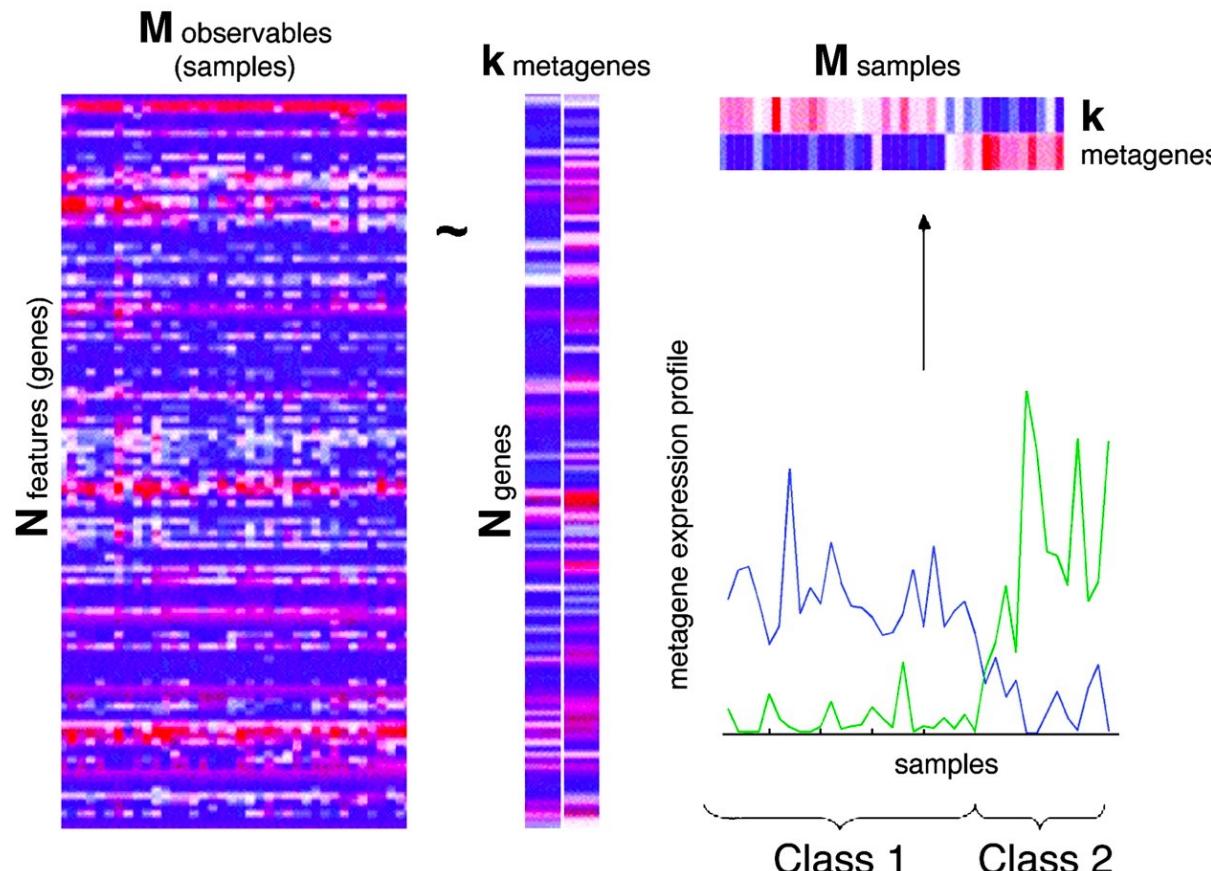
14,249 cells from scRNA-seq and 2,548 cells from scATAC-seq



Batch correction and data modality integration

Nonnegative Matrix Factorization (NMF)

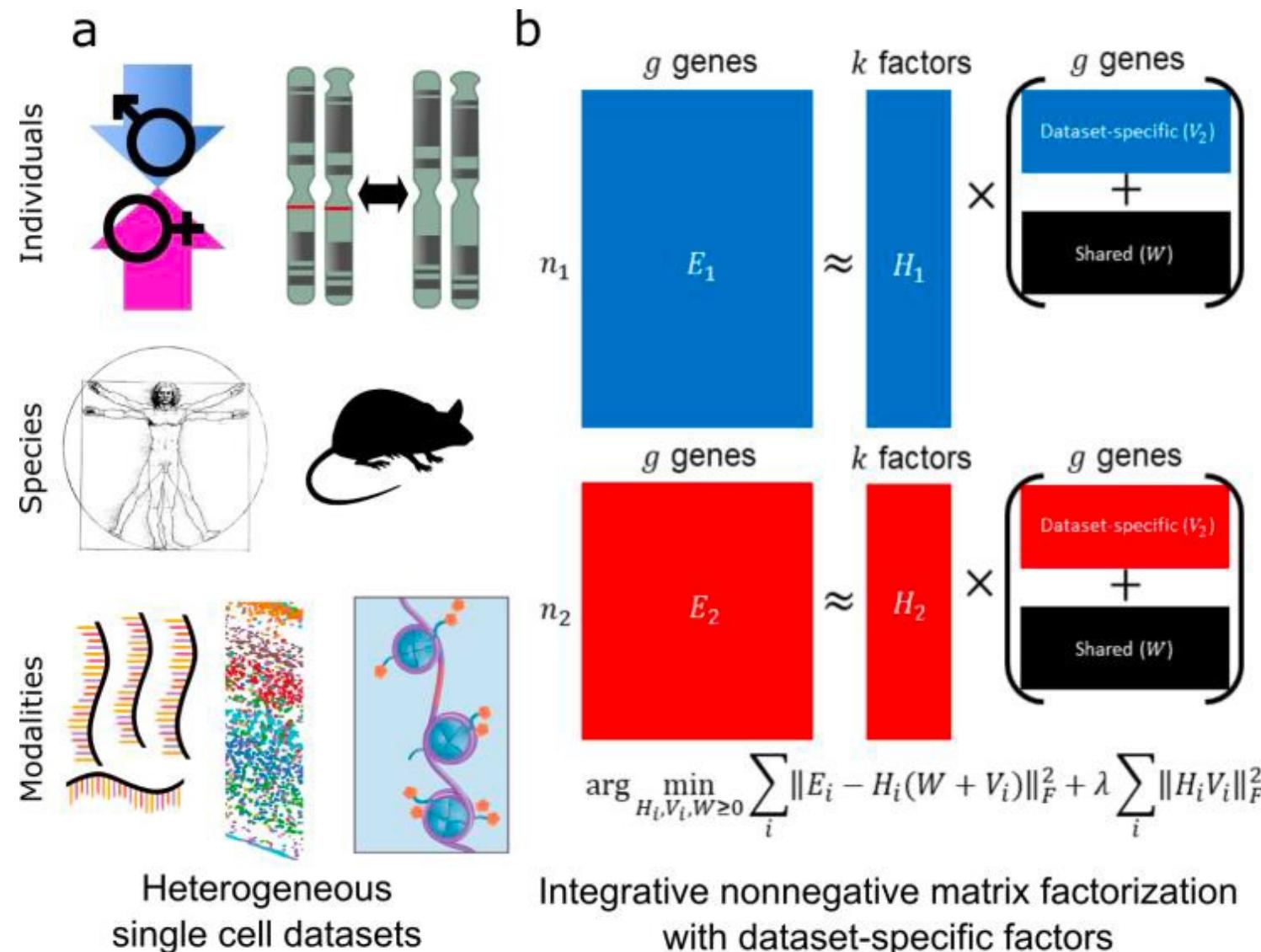
$$A \text{ (rank } M) \sim W H \text{ (rank } k=2)$$



"Our goal is to find a small number of metagenes, each defined as a positive linear combination of the N genes. We can then approximate the gene expression pattern of samples as positive linear combinations of these metagenes. Mathematically, this corresponds to factoring matrix A into two matrices with positive entries, $A \sim WH$."

Batch correction and data modality integration using LIGER

LIGER implements
non-negative
matrix factorization



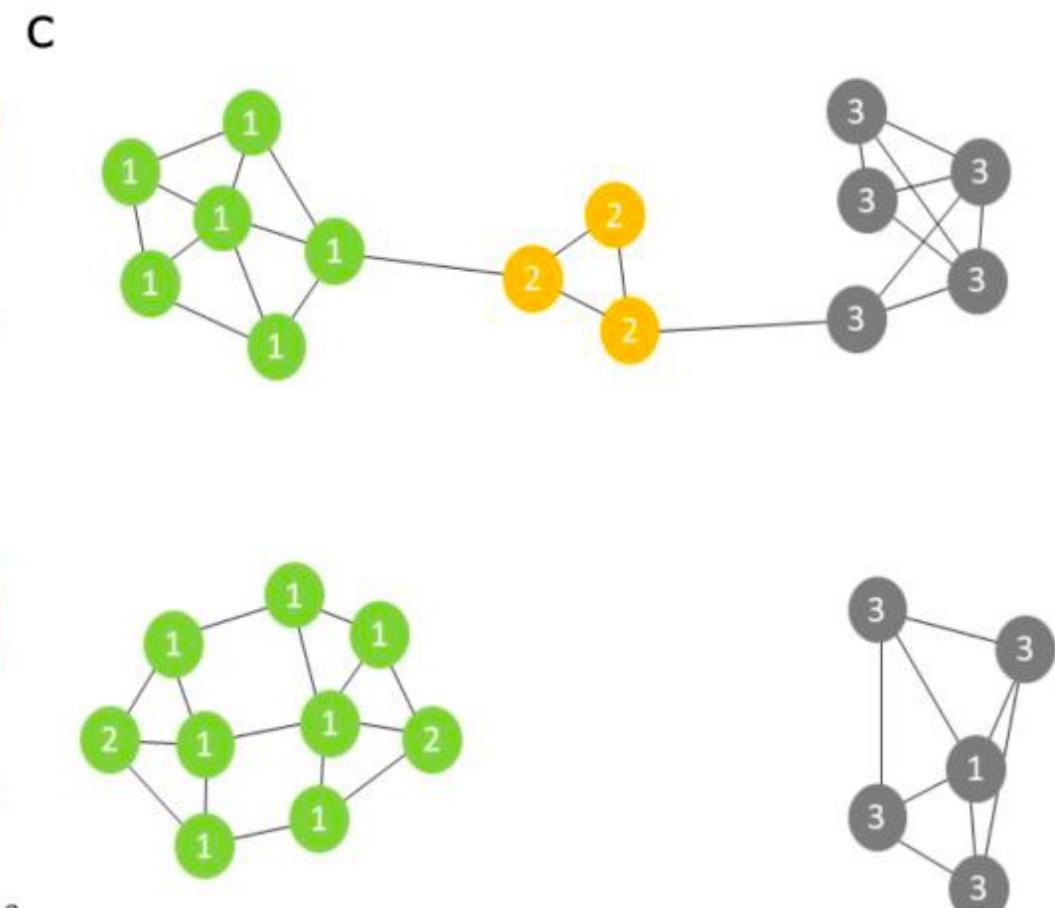
Batch correction and data modality integration using LIGER

LIGER implements
non-negative
matrix factorization

b

$$\begin{array}{ccc} g \text{ genes} & k \text{ factors} & \\ n_1 \quad E_1 & \approx & H_1 \\ & & \times \end{array}$$
$$\begin{array}{ccc} g \text{ genes} & k \text{ factors} & \\ n_2 \quad E_2 & \approx & H_2 \\ & & \times \end{array}$$
$$\arg \min_{H_i, V_i, W \geq 0} \sum_i \|E_i - H_i(W + V_i)\|_F^2 + \lambda \sum_i \|H_i V_i\|_F^2$$

Integrative nonnegative matrix factorization
with dataset-specific factors

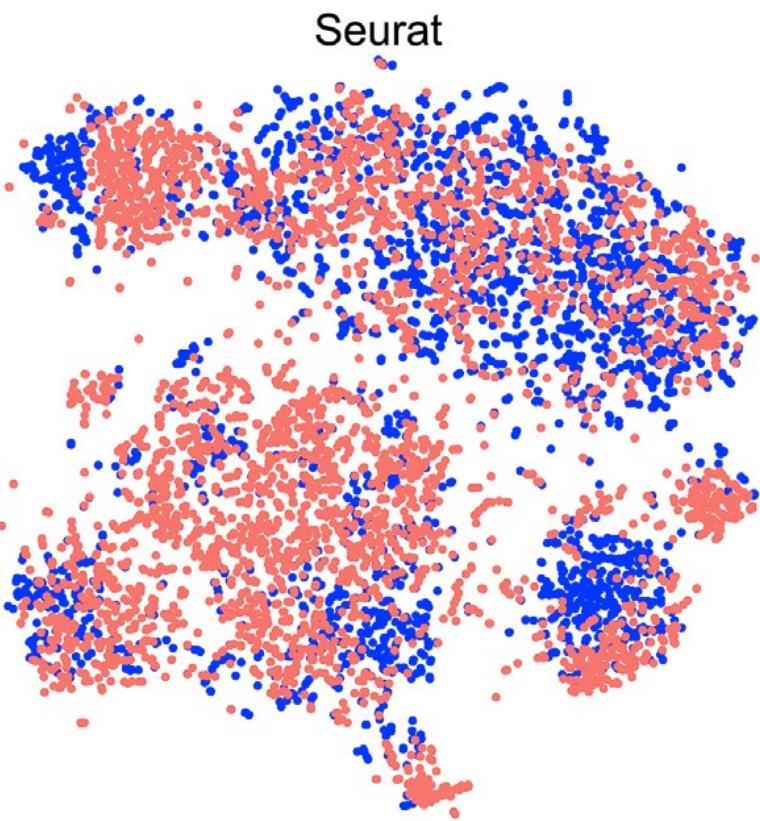


Joint clustering using
shared factor neighborhood graph

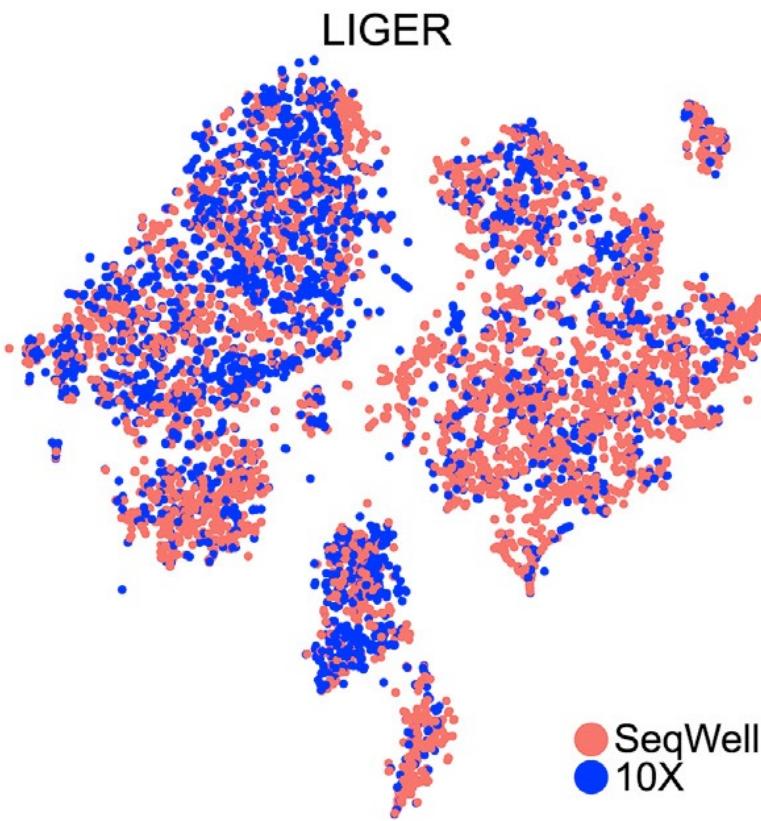
Integrating blood cell datasets using LIGER

Higher alignment score =
better data integration

A

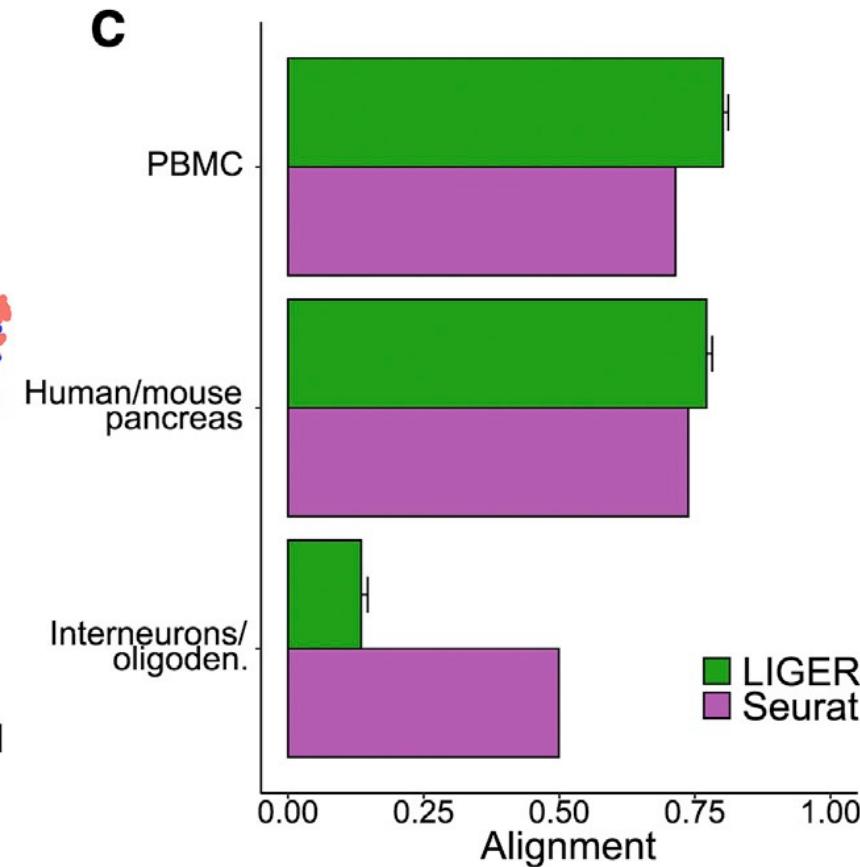


B

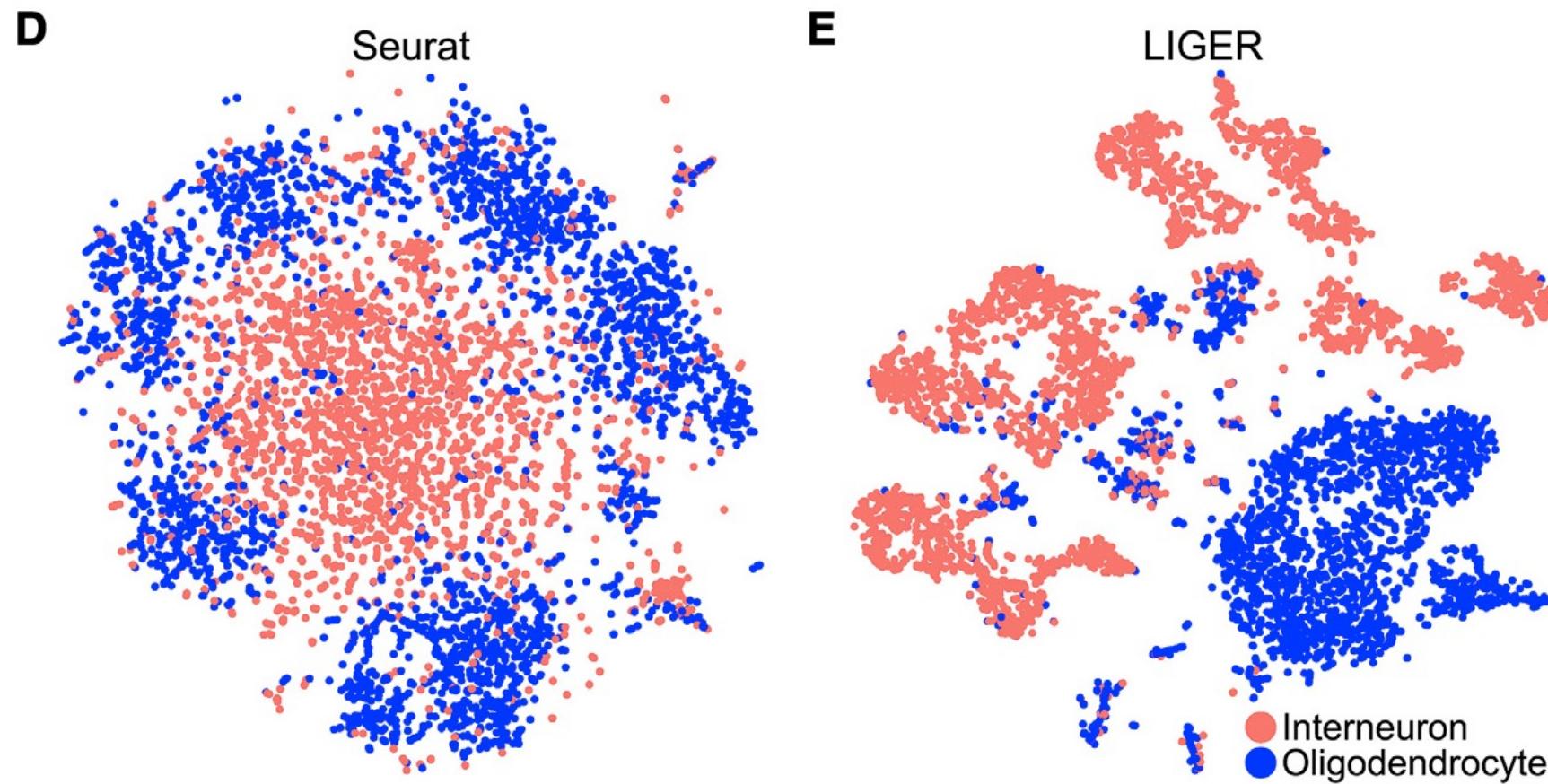


SeqWell
10X

C



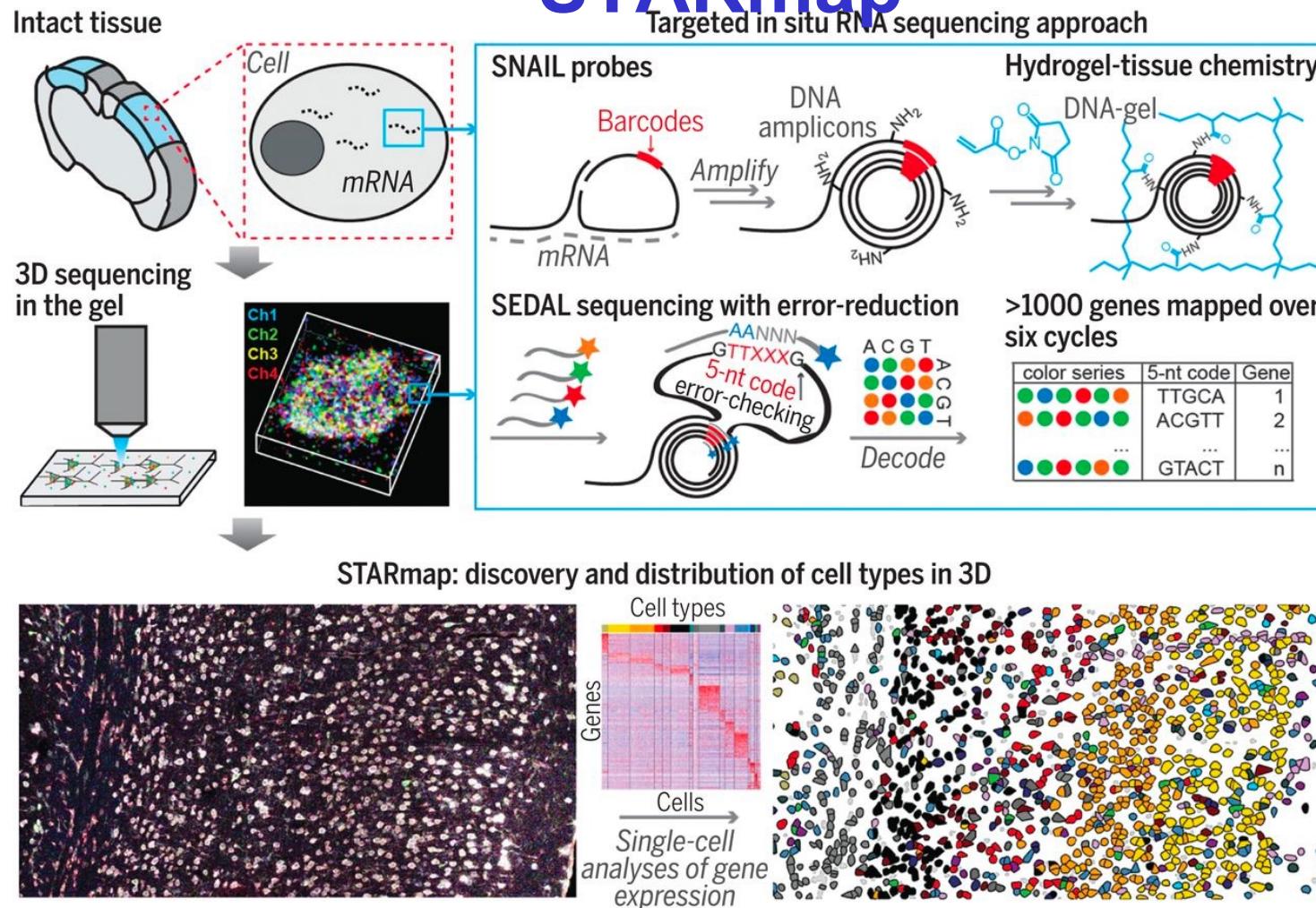
Integrating blood cell datasets using LIGER



Ideally, divergent cell types should not cluster together after batch correction.

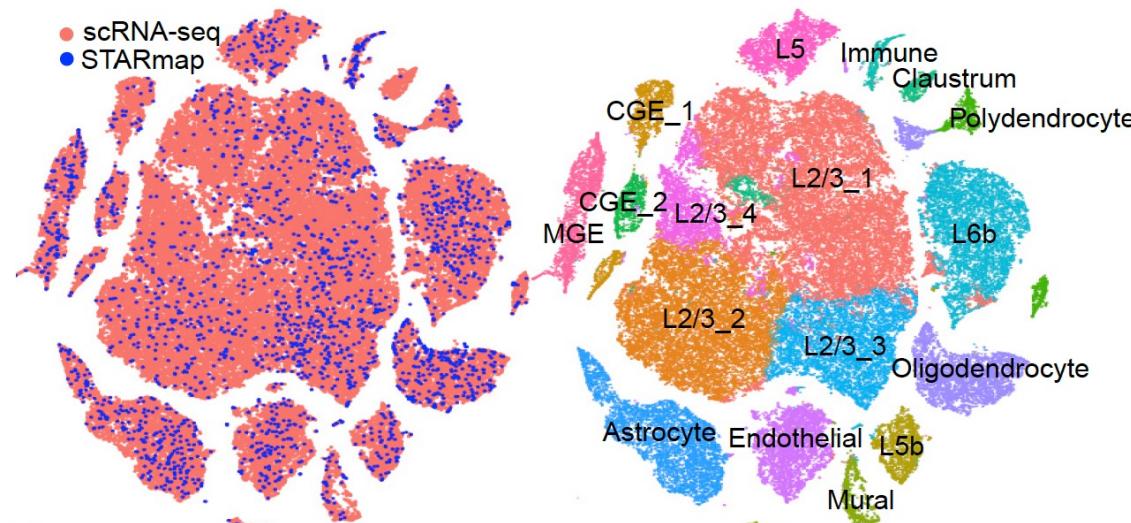
In situ spatial transcriptomic data in mouse frontal cortex

STARmap



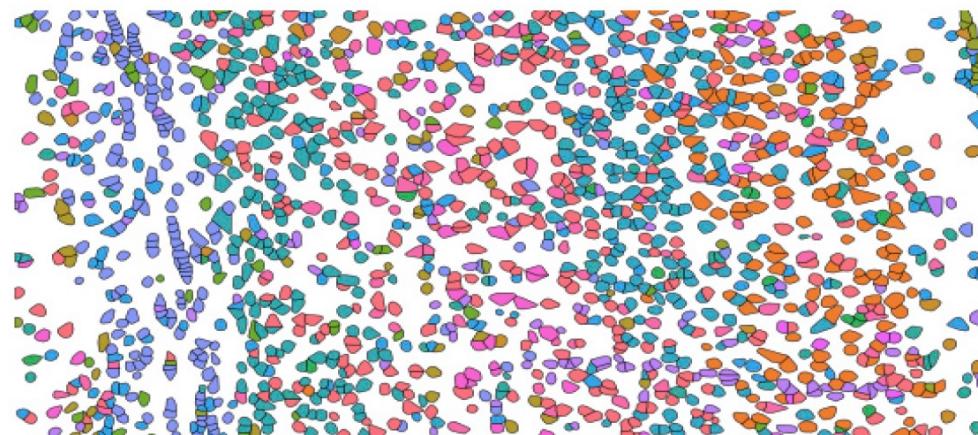
Using LIGER to integrate single-cell transcriptomic and *in situ*^B spatial transcriptomic data

A



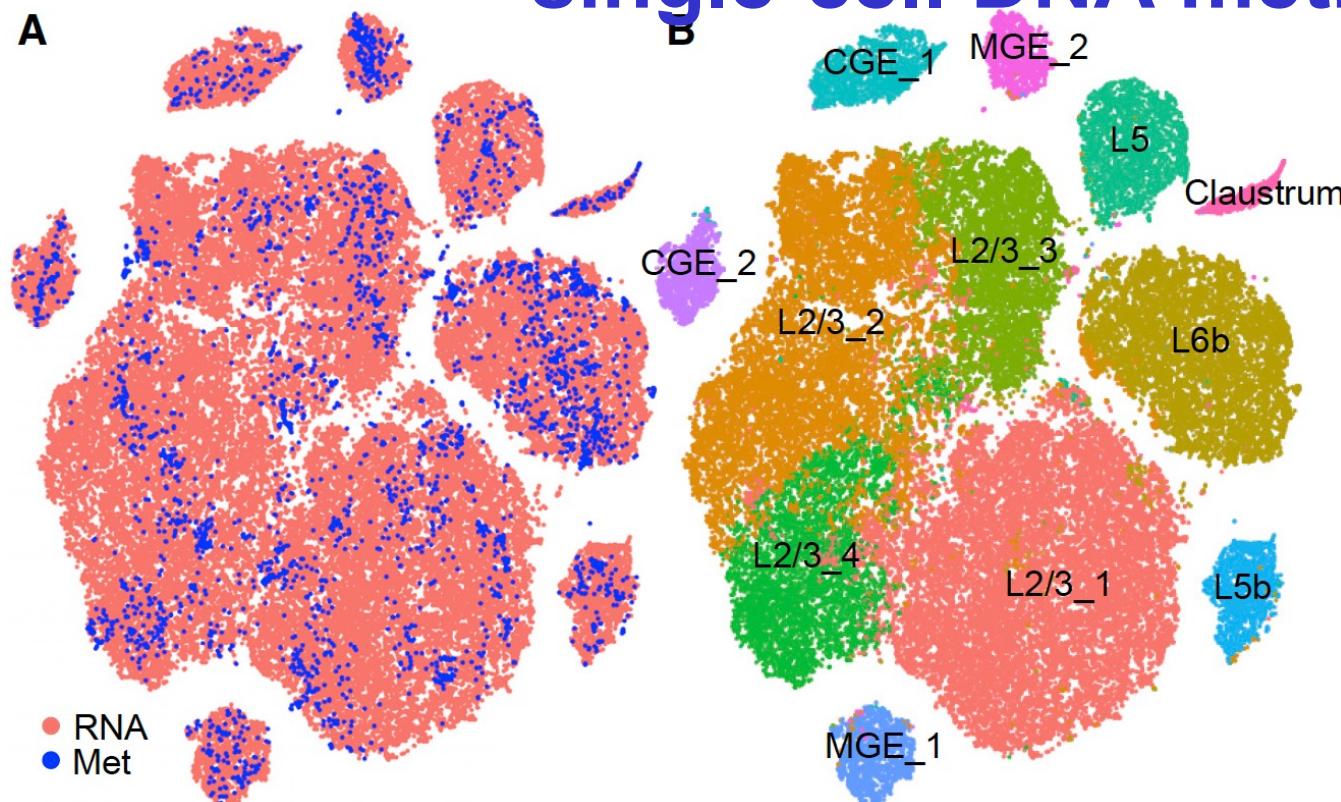
71,000 cells from scRNA-seq
2,500 cells from STARmap

D



What are advantages of integrated analysis of these 2 datasets?

Using LIGER to integrate single-cell transcriptomic and single-cell DNA methylation data



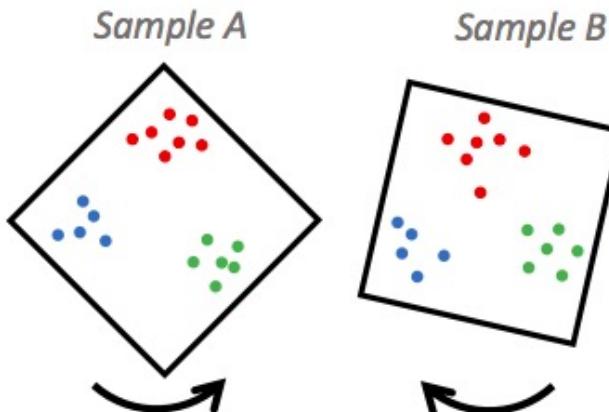
56,000 cells from scRNA-seq
3,000 cells from DNA methylation

"We reasoned that, because gene body methylation is generally anticorrelated with gene expression, reversing the direction of the methylation signal would allow joint analysis."

Batch correction and data modality integration using Conos

Conos (Clustering on Network of Samples)

Error-prone pair alignment



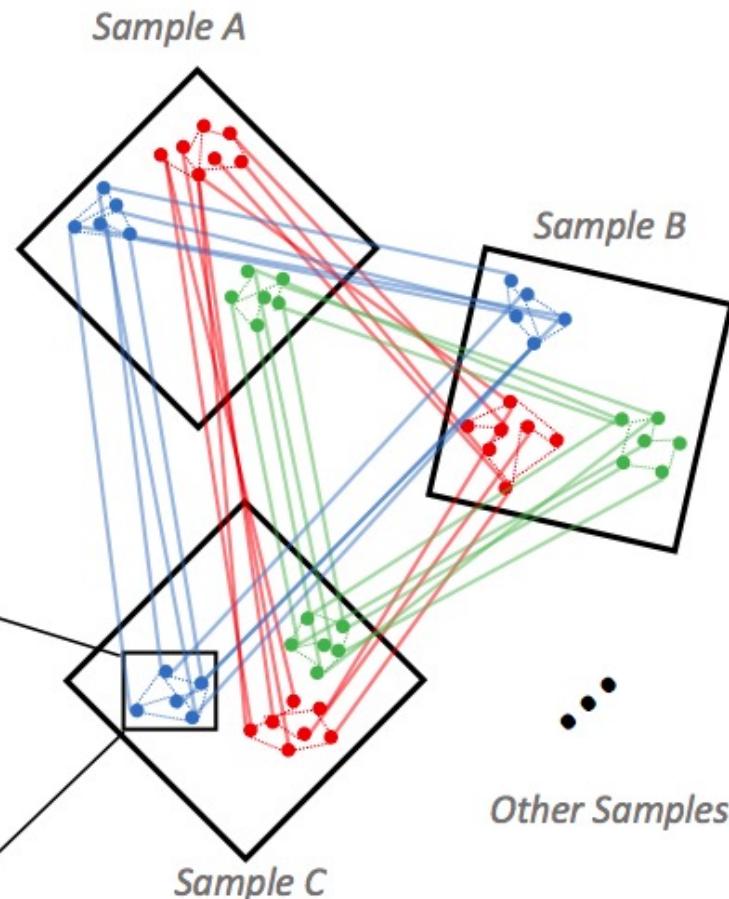
rotations:

- CPCA
- GSVD
- JNMF

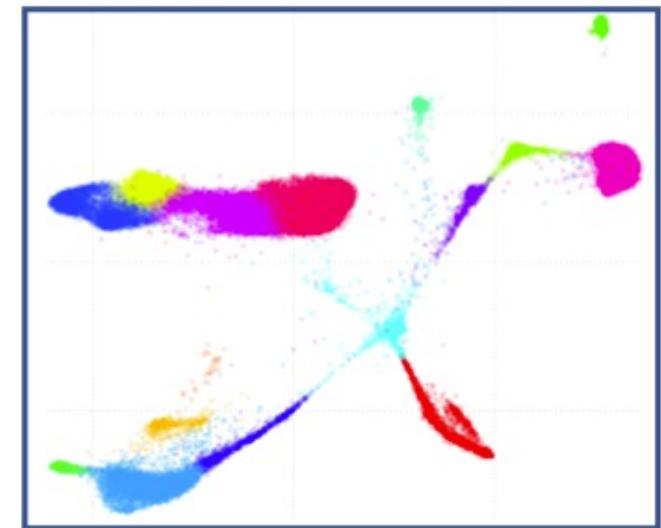
mapping:

- nearest neighbor
- mutual NN

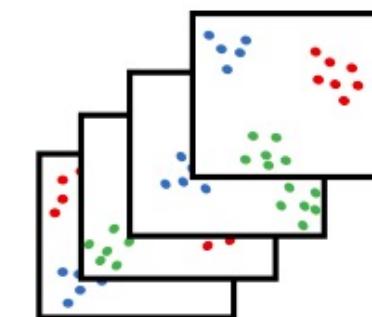
All-to-all pair (joint) graph



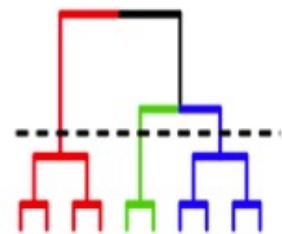
Joint graph



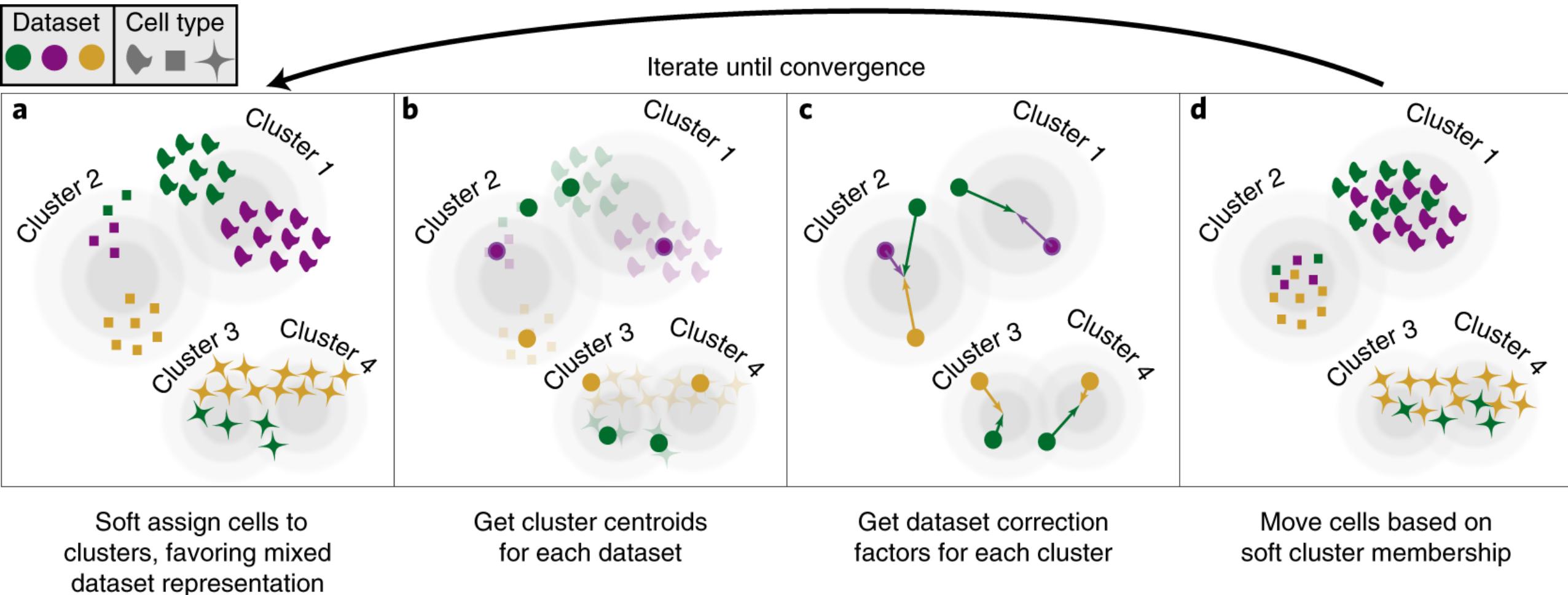
Joint clustering



Breadth analysis

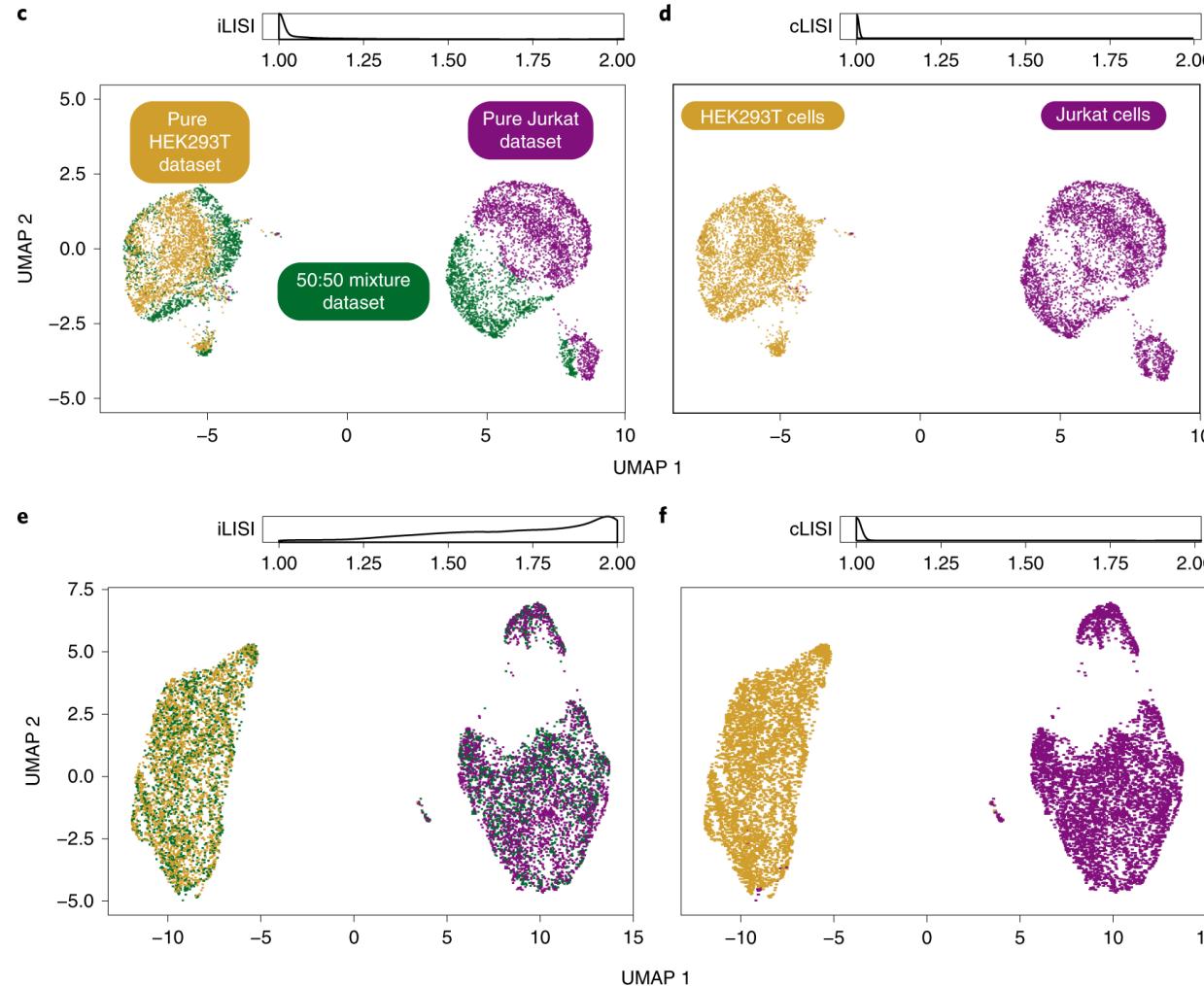


Batch correction and data modality integration using Harmony



Harmony integrates different datasets while maintaining cell type differences

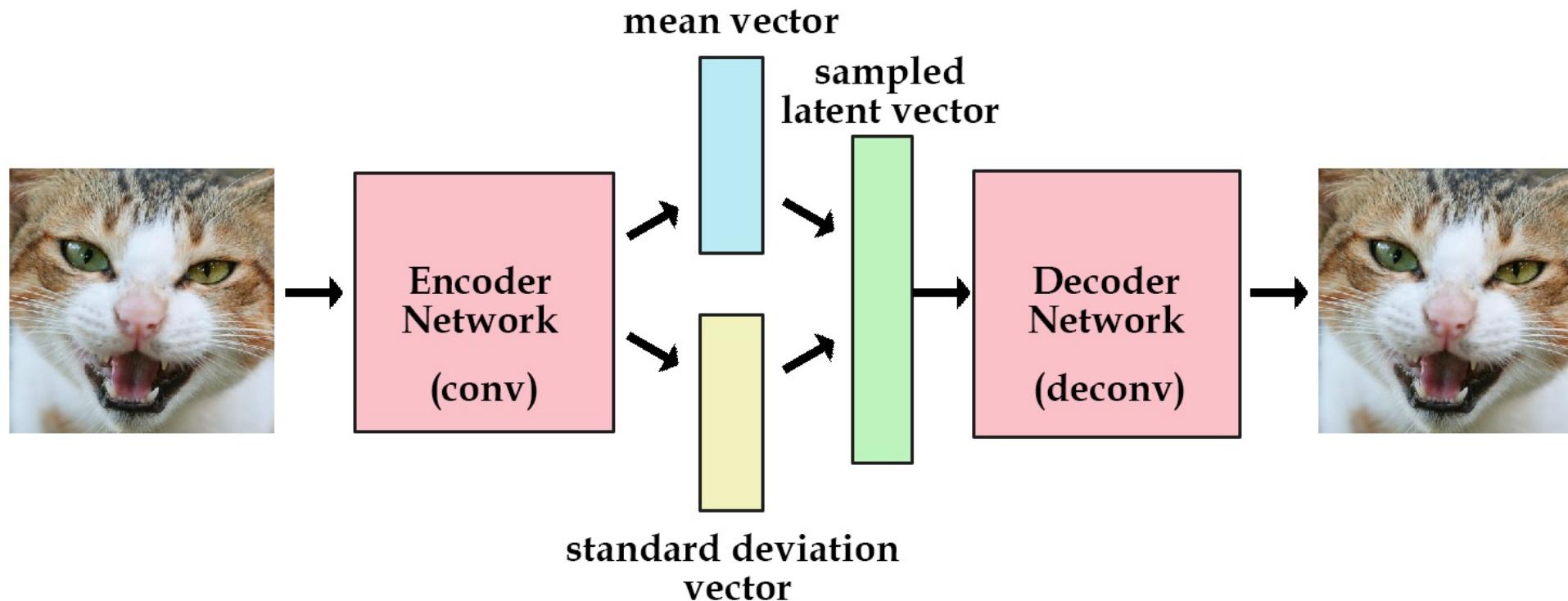
Before integration



After integration

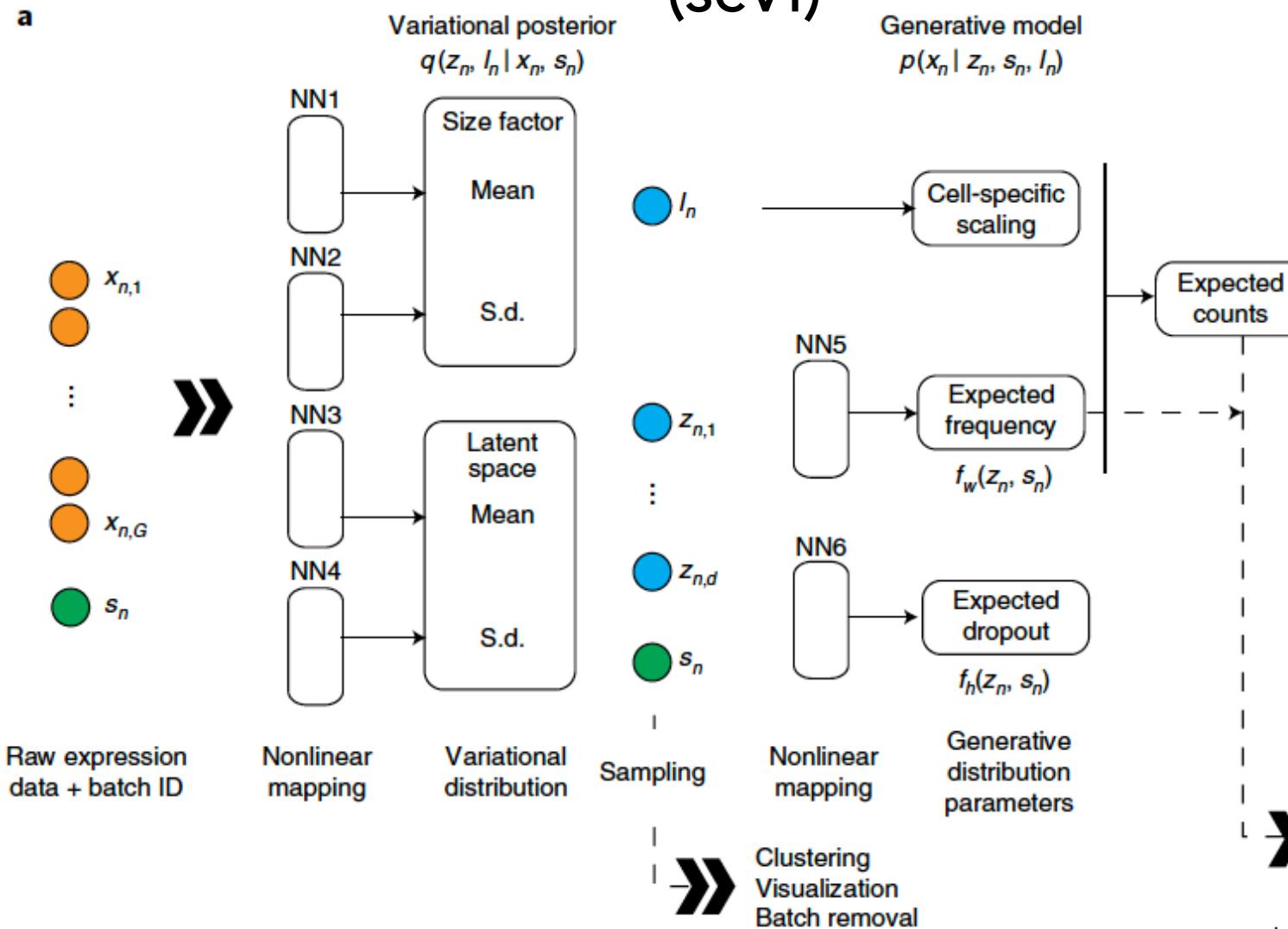
Batch correction with deep learning approaches

Autoencoders and variational autoencoders are popular frameworks for embedding single cell genomics datasets.



Batch correction with deep learning approaches

single-cell variation inference (scVI)



Models follow a train/validate/test framework

The “integrated data” is the embedding in the latent space.

Comparison of data integration methods

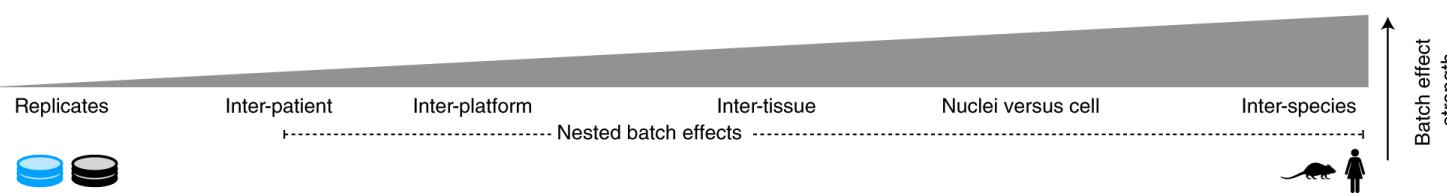
a

Considerations	scANVI	Scanorama embed	scVI	FastMNN embed	scGen	Harmony	FastMNN gene	Seurat v3 RPCA	BBKNN	Scanorama gene	ComBat	MNN	Seurat v3 CCA	trVAE	Conos	DESC	LIGER	SAUCIE embed	SAUCIE gene
Input																			
Programming language	Python	Python	Python	R	Python	R	R	Python	Python	Python	Python / R	Python / R	R	Python	R	Python	R	Python	
Method runs without additional information	X				X														
Consistent top performer	✓	✓	✓			✓													
Scib results																			
Top method on small/simple tasks		✓		✓	✓	✓	✓												
Top method on large/complex tasks	✓	✓	✓		✓		✓												
Top method on ATAC data	—	—	—			✓											✓		
Task details																			
Integrates strong batch effects	✓	—	—	—	✓			—	—										
Top method for recovery cell states or modules	✓	✓								✓	✓	✓							
Confounding of bio and batch variance	✓	—			✓														
Top method for trajectories	—	✓	—	✓	✓	✓													
Method deals with varying compositions											X								
Speed																			
Fast method for quick results								✓			✓								
Scales well to large datasets on CPU	✓	—	✓					✓	—							✓	✓		
Method has GPU support	✓		✓	✓												✓	✓		
Scales well to feature spaces beyond genes																✓	✓		
Output																			
Method shows corrected expression					✓		✓	✓		✓	✓	✓	✓					✓	
Method gives relative cell embeddings							X					X							

Legend:

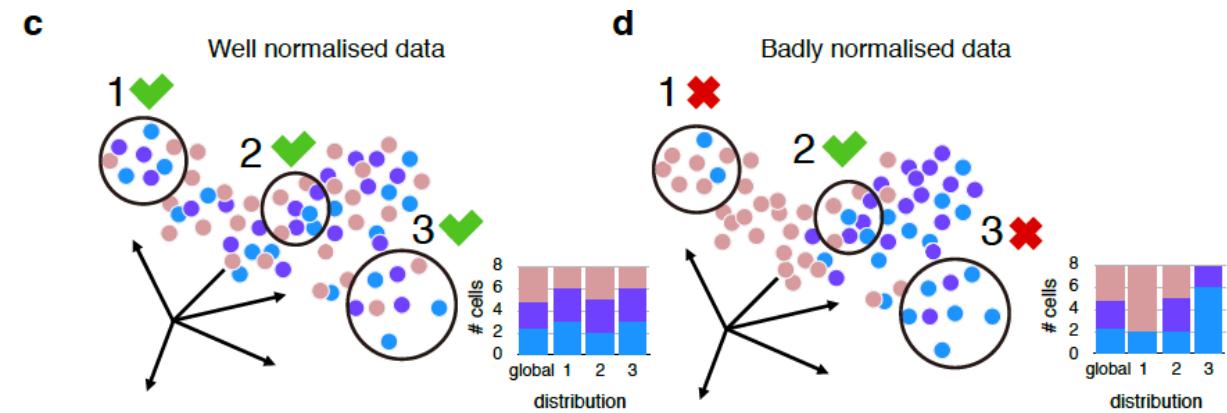
- ✓ Fulfills the criterion
- Partial fulfillment of criterion
- X Does not fulfill criterion

b



Methods to assess performance of batch correction

- Entropy of batch mixing
- kBET - k-nearest neighbor batch effect test



- Silhouette coefficient
 - cells of the same cell type are close together and far from other cells of a different type
- Adjusted rand index
 - do batch labels and cluster labels agree with one another?
- Biological significance

Common assumptions during batch correction

- At least one cell population is found shared in both datasets.
- The batch effect and the biological differences do not overlap with one another (ie orthogonal).
- The magnitude and variation of the biological effect you care about is greater than that of the batch effect.

Will batch correction remove biological differences in single cell transcriptomic data? (ChatGPT response)

Batch correction in single-cell transcriptomics is designed to remove technical variations that arise due to differences in the experimental procedures such as sample preparation, sequencing, and data analysis. The goal of batch correction is to reduce technical noise in the data and allow biological differences between cells to be better distinguished.

However, batch correction can potentially remove some biological differences in the data, particularly if the biological differences are correlated with the batch effect. For example, if cells from different batches are isolated from different tissues or at different stages of development, then the batch correction algorithm may inadvertently adjust the expression levels of genes that are differentially expressed between these groups.

Therefore, it is important to carefully evaluate the results of batch correction and to consider the biological context of the data. One approach is to compare the results of batch-corrected data with uncorrected data and to ensure that the biological differences identified in the uncorrected data are still present after batch correction. It is also important to choose an appropriate batch correction method that is suitable for the specific experimental design and to use quality control metrics to ensure that the batch correction has not removed important biological information.

Recent helpful articles on data integration methods

Argelaguet, Ricard, et al. "Computational principles and challenges in single-cell data integration." *Nature biotechnology* (2021).

- Review

Luecken, Malte D., et al. "Benchmarking atlas-level data integration in single-cell genomics." *Nature methods* (2022).

- Methods comparison

Tran, Hoa Thi Nhu, et al. "A benchmark of batch-effect correction methods for single-cell RNA sequencing data." *Genome biology* (2020).

- Methods comparison

Once data is integrated, how do we compare across biological conditions?

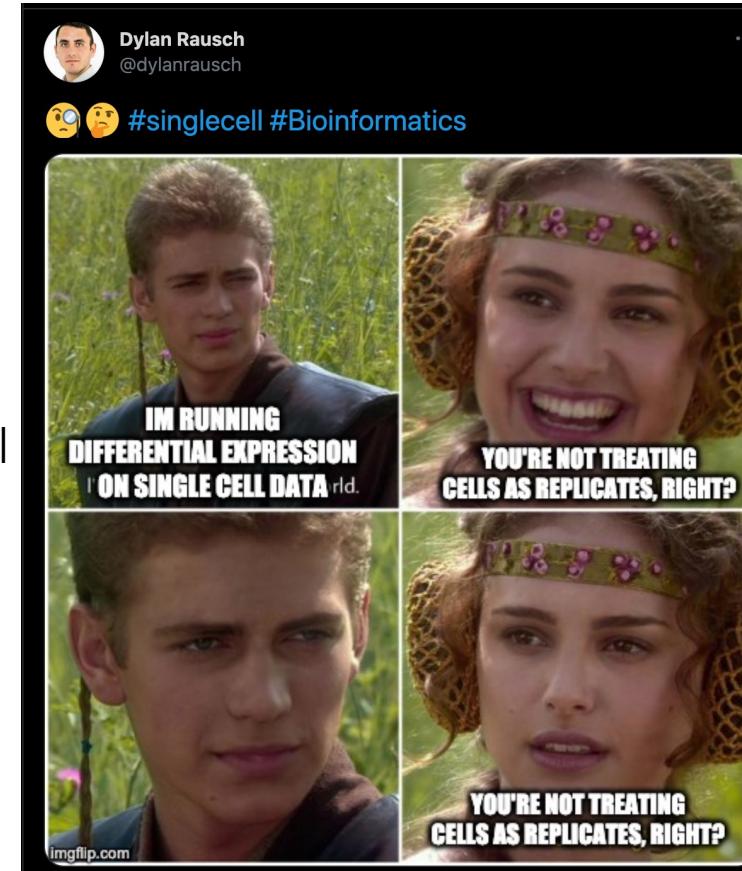
When detecting gene differentially expressed across different biological conditions (e.g. case vs control, treated vs untreated, etc...), what is the replicate? A single cell? A sample?

Treating a single cell as a replicate will lead to false discoveries.

Strongly consider using pseudobulk models or mixed effect models when comparing samples across conditions.

- Crowell, Helena L., et al. "Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data." *Nature communications* (2020).
- Squair, Jordan W., et al. "Confronting false discoveries in single-cell differential expression." *Nature communications* (2021).
- <https://www.nxn.se/valent/2019/2/15/handling-confounded-samples-for-differential-expression-in-scrna-seq-experiments>

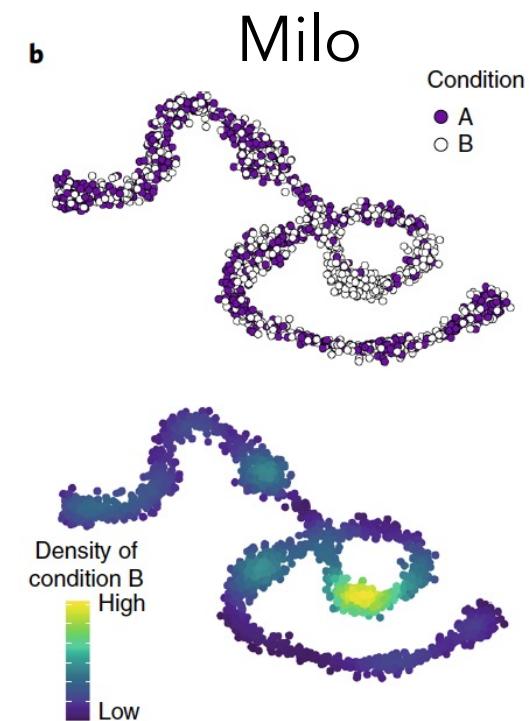
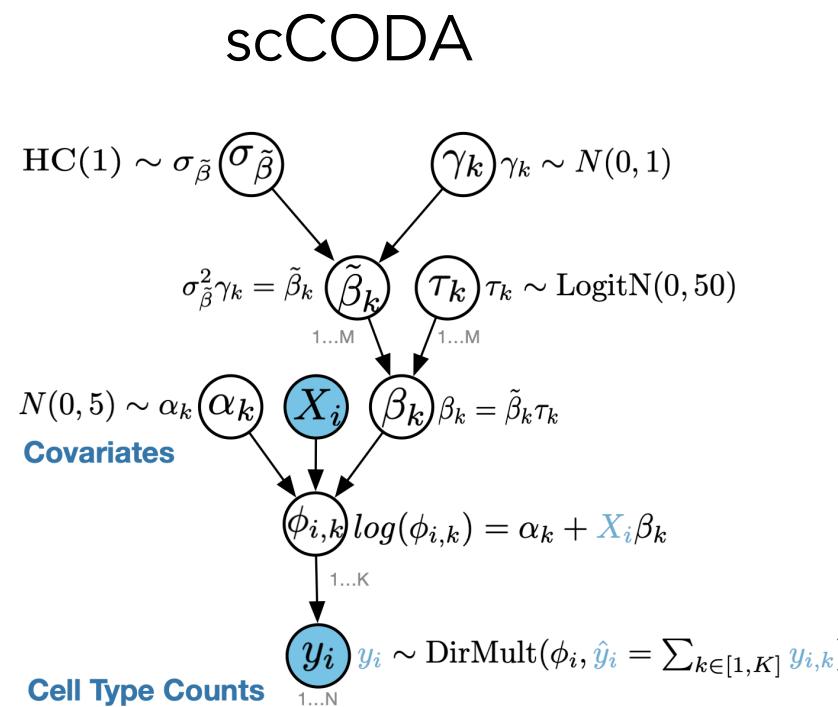
Remember: go back to normalized (non batch-corrected) gene expression values for DE analyses! Add covariates for your batches in the DE statistical model.



Testing for significant changes in cell type abundance across biological conditions

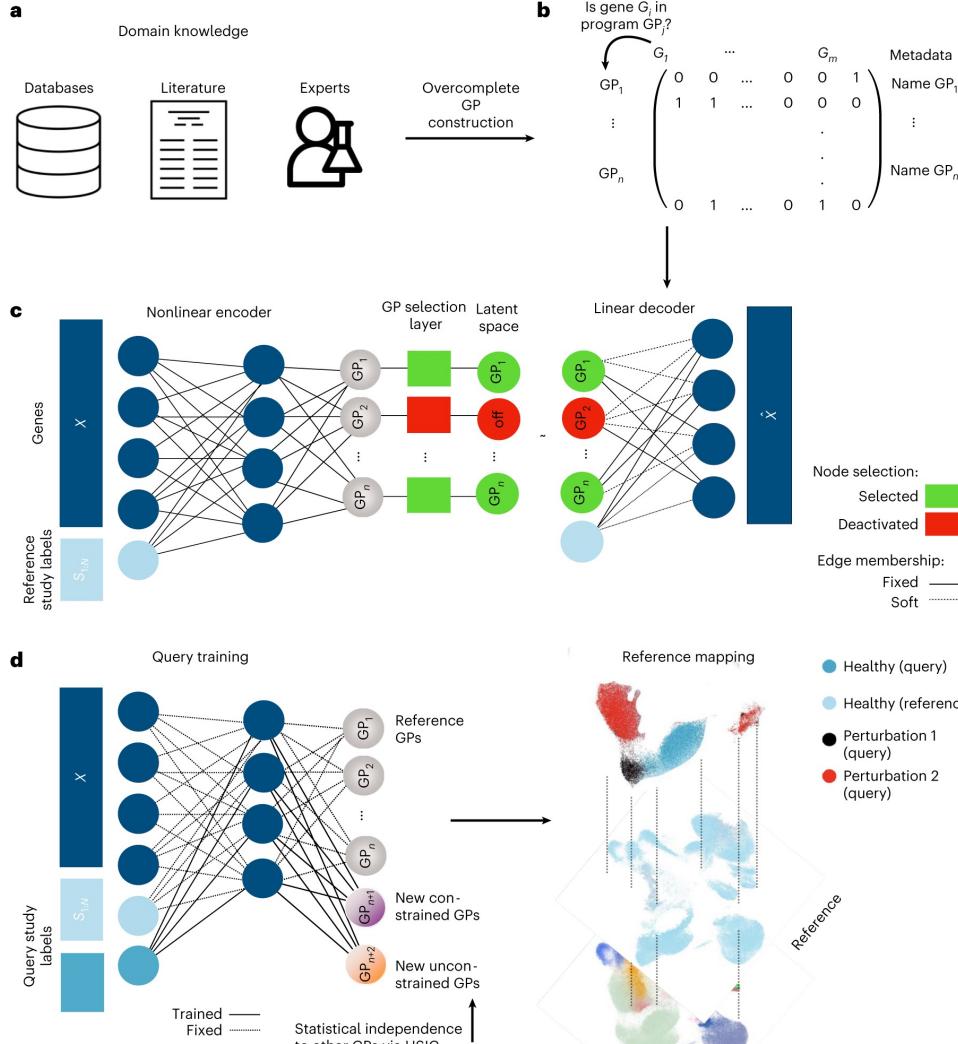
Assume we have 3 cell types (A, B, C) in a treated and untreated sample. If the frequency of cell type A increases in the treated sample due to the treatment, what will happen to the frequencies of cell types B and C?

The compositional nature of cell type abundance data means the abundances of cell type A, cell type B, and cell type C are not independent of one other. Their frequencies must always sum to 1.



Integrating datasets and mapping to gene programs

ExpiMap (deep learning)



Spectra (matrix factorization)

