

# Machine learning ideas in Seurat packages

Jan/16/2020

Sunho Park

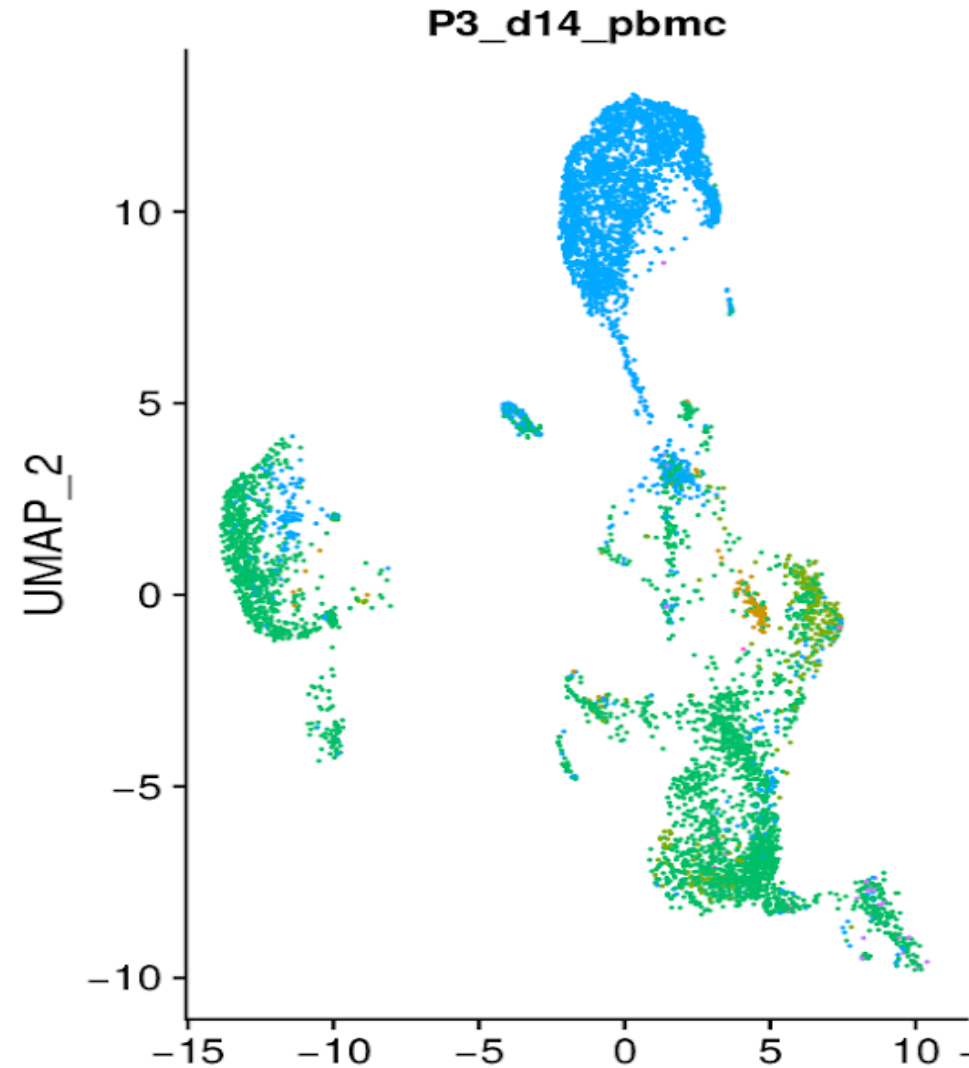
# Outline

- **Dimensionality reduction (visualization)**
  - t-SNE
  - Uniform Manifold Approximation and Projection (UMAP)
- **Clustering approaches**
  - Graph based methods: Louvain algorithm
  - Cell type identification
- **Multiple Dataset Integration**
  - Canonical correlation analysis & L2-norm normalization
  - Anchoring

# Outline

- **Dimensionality reduction (visualization)**
  - t-SNE
  - Uniform Manifold Approximation and Projection (UMAP)
- Clustering approaches
  - Graph based methods: Louvain algorithm
  - Cell type identification
- Multiple Dataset Integration
  - Canonical correlation analysis & L2-norm normalization
  - Anchoring

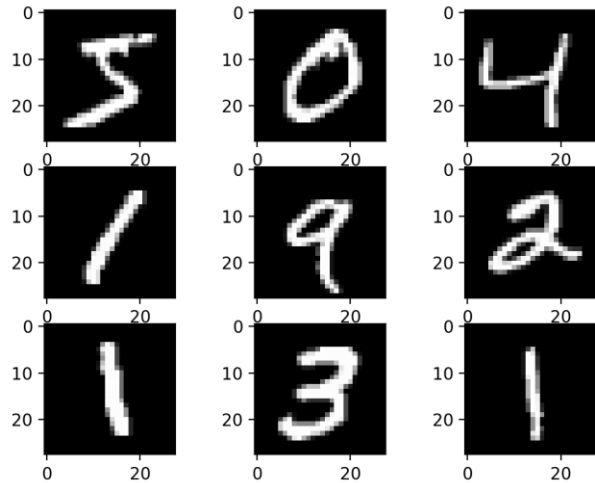
# Dimensionality reduction for single-cell data



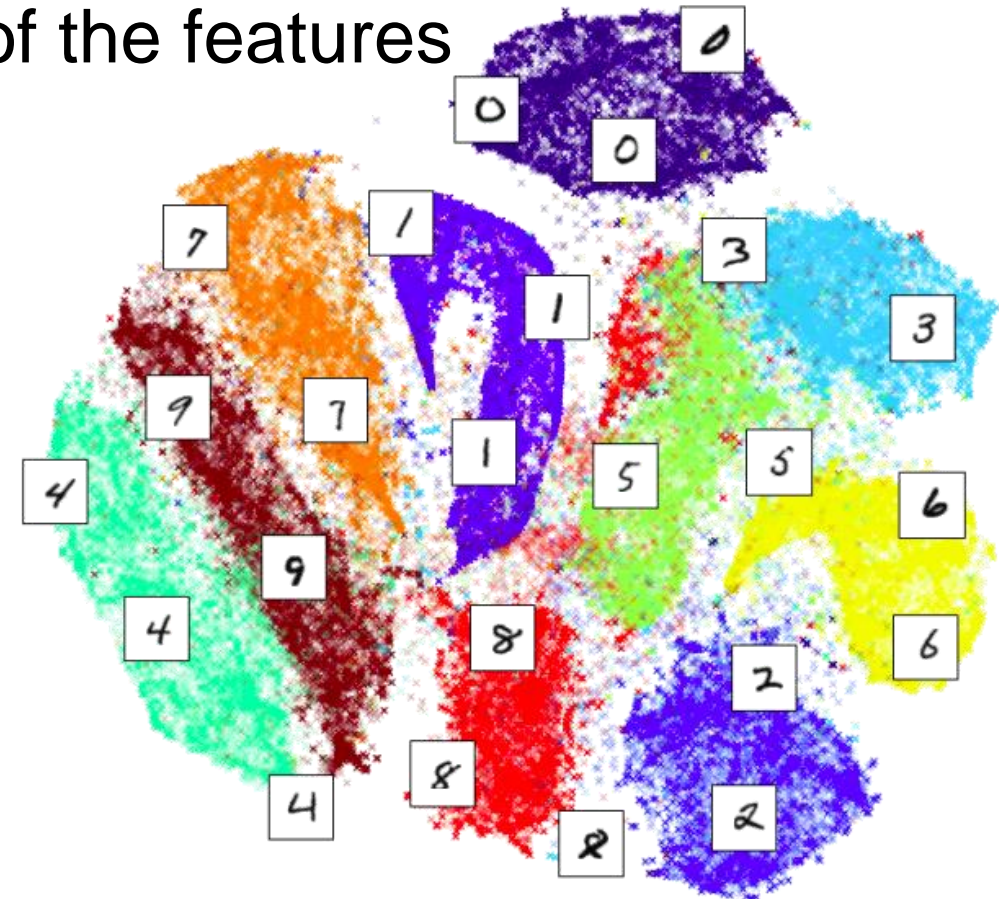
# Dimensionality reduction

Figure credit: liam schoneveld

- A process of reducing the number of the features

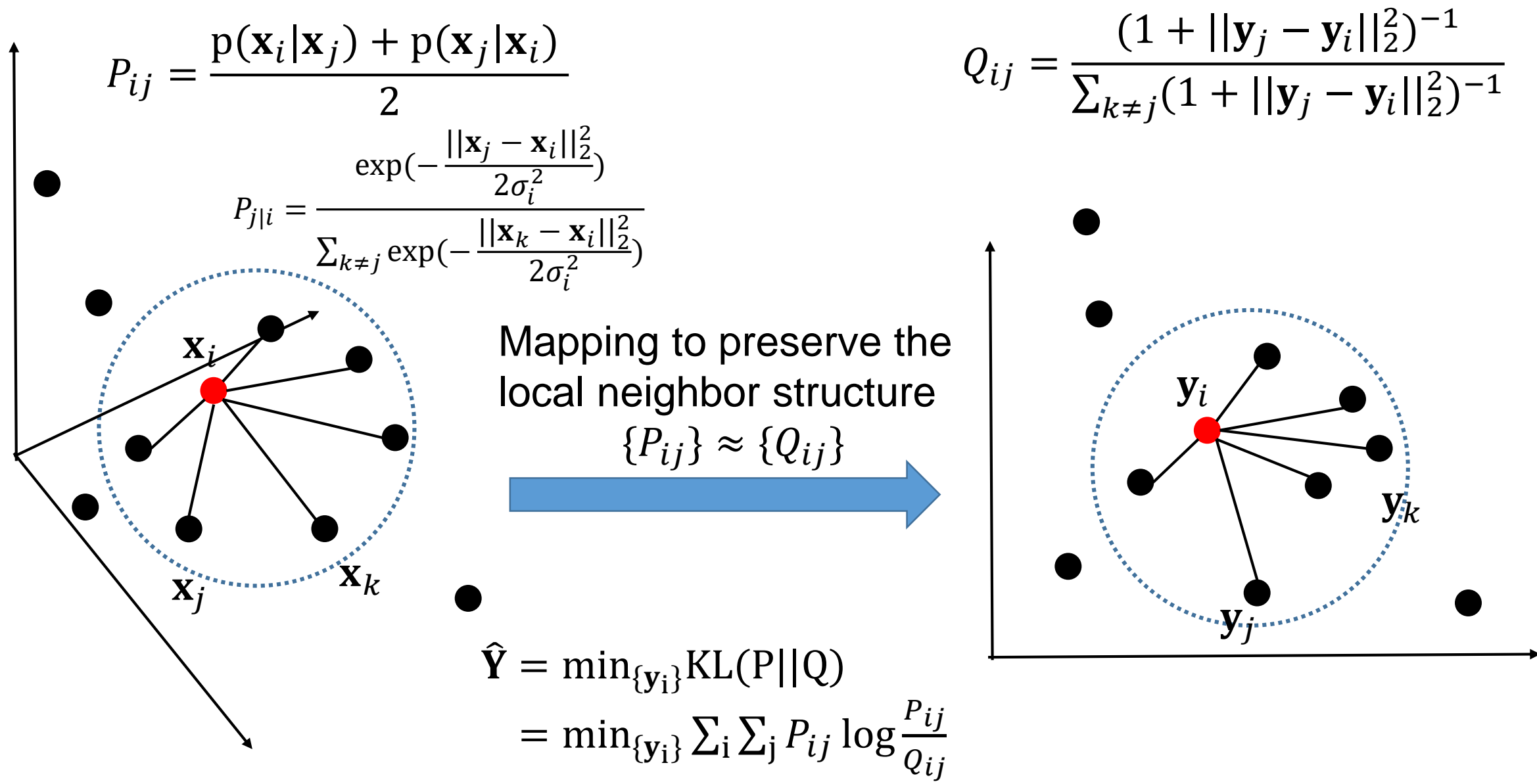


Original image (MNIST):  $\mathbf{x}_i \in \mathbb{R}^{784}$  ( $28 \times 28$ )



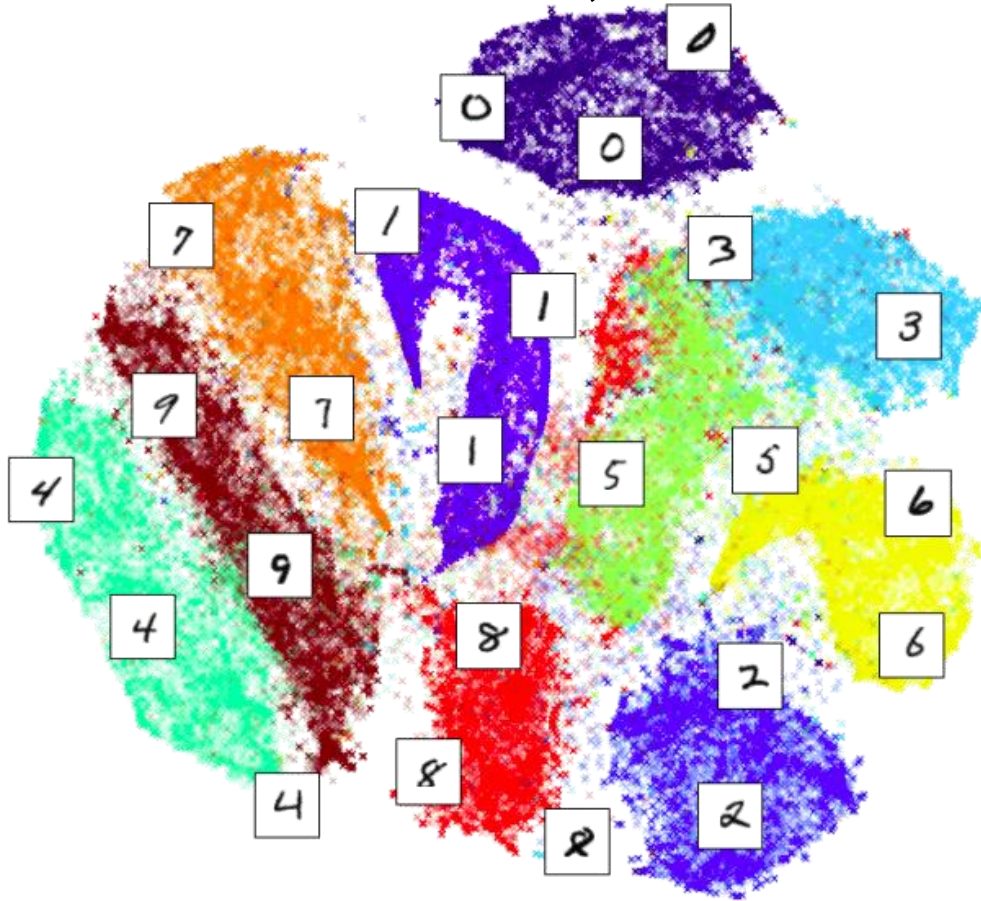
Reduced space (by t-SNE):  $\mathbf{x}_i \in \mathbb{R}^2$

# Dimensionality reduction: t-SNE



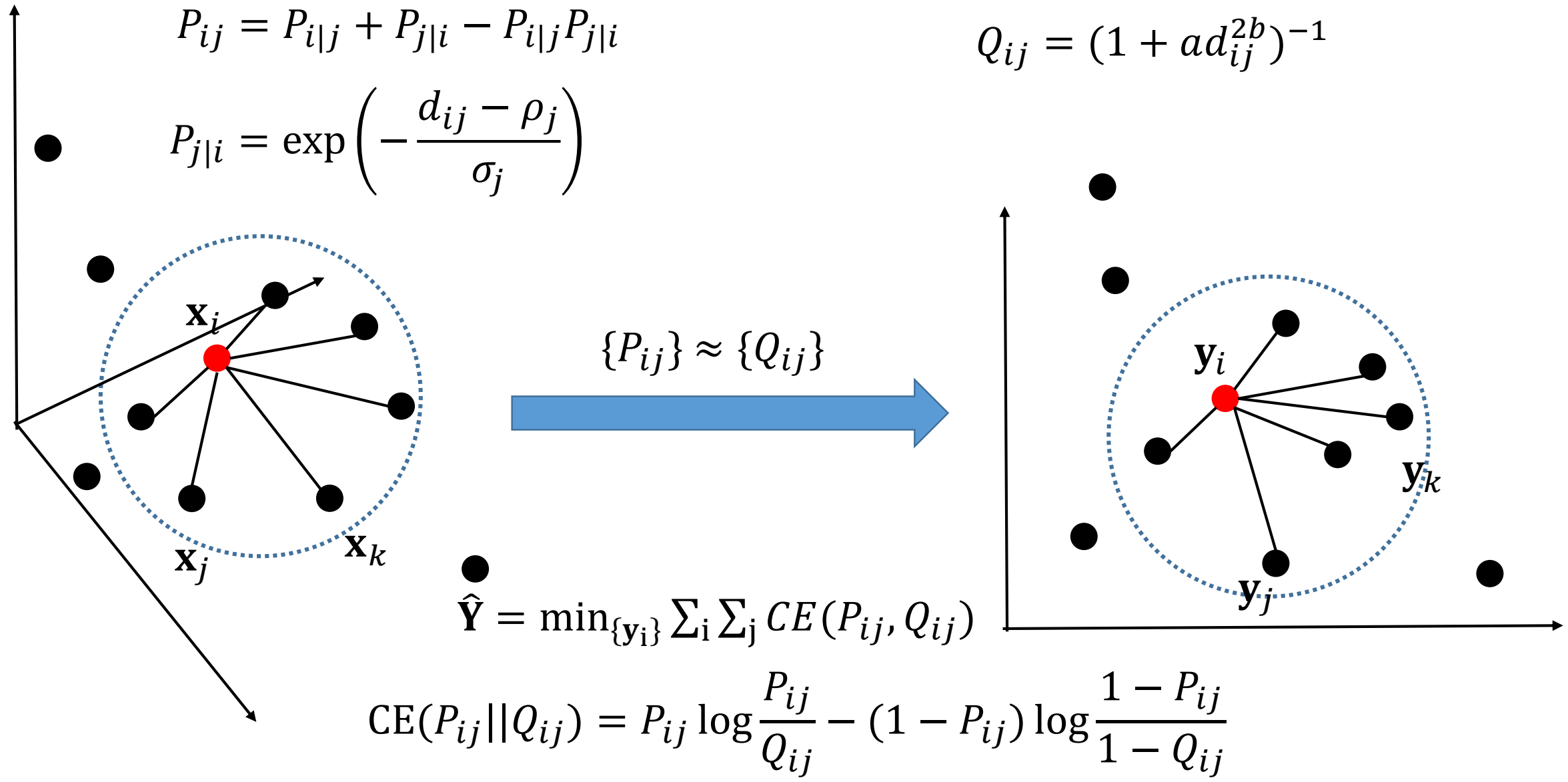
# Dimensionality reduction

- t-SNE has been widely used for visualization of high dimensional data, but it has some limitations:



- **The global structure is not preserved**
  - points in the same cluster are close to each other (local structure)
  - Similarities between clusters might not be accurate (global structure, e.g., consider clusters (0 and 3), (0 and 6))
- **It is not scalable to handle fast growing sample sizes in single cell data.**

# Uniform Manifold Approximation and Projection (UMAP)





# Dimensionality reduction

- UMAP works better for preserving the global structure of the data than t-SNE

- t-SNE

$$\text{KL}(P_{ij}||Q_{ij}) \approx \exp(-d_{ij}^X) \log(1 + (d_{ij}^Y)^2)$$

- $d_{ij}^X$  is large:  $d_{ij}^Y$  can be any value (i.e., the global structure is not guaranteed)

- UMAP

$$\text{CE}(P_{ij}||Q_{ij}) \approx \exp(-d_{ij}^X) \log(1 + (d_{ij}^Y)^2) + (1 - \exp(-d_{ij}^X)) \log\left(\frac{1 + (d_{ij}^Y)^2}{(d_{ij}^Y)^2}\right)$$

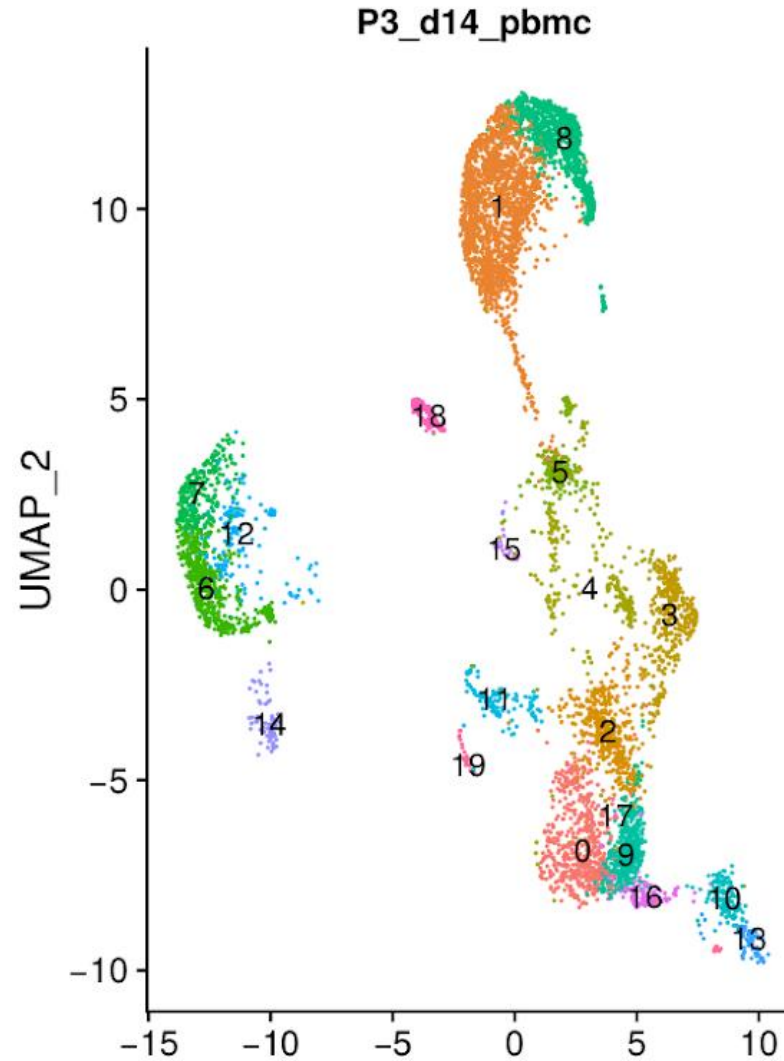
- $d_{ij}^X$  is small:  $\text{CE}(P_{ij}||Q_{ij}) \approx \log(1 + (d_{ij}^Y)^2)$

- $d_{ij}^X$  is large:  $\text{CE}(P_{ij}||Q_{ij}) \approx \log\left(\frac{1 + (d_{ij}^Y)^2}{(d_{ij}^Y)^2}\right)$  (i.e., it gives a high penalty for a small  $d_{ij}^Y$  and thus the global structure can be preserved)

# Outline

- Dimensionality reduction (visualization)
  - t-SNE
  - Uniform Manifold Approximation and Projection (UMAP)
- **Clustering approaches**
  - Graph based methods: Louvain algorithm
  - Cell type identification
- Multiple Dataset Integration
  - Canonical correlation analysis & L2-norm normalization
  - Anchoring

# Clustering approaches



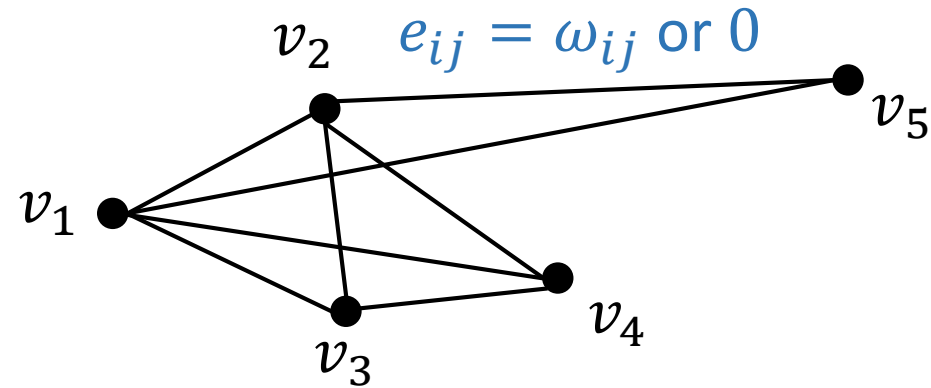
# Community detection (clustering) in networks (graphs)

## ■ Graph

It is used to represent complex systems (e.g., friend networks on Facebook, gene-gene interaction networks)

- $V = \{v_1, v_2, \dots, v_N\}$ : Nodes (vertices)
- $E = \{e_{ij}\}$ : Links (edges)

$$G = \{V, E\}$$



# Clustering on graphs

- Goal: divide a graph into multiple clusters where the nodes in the same cluster are more close to each other than to those in other clusters

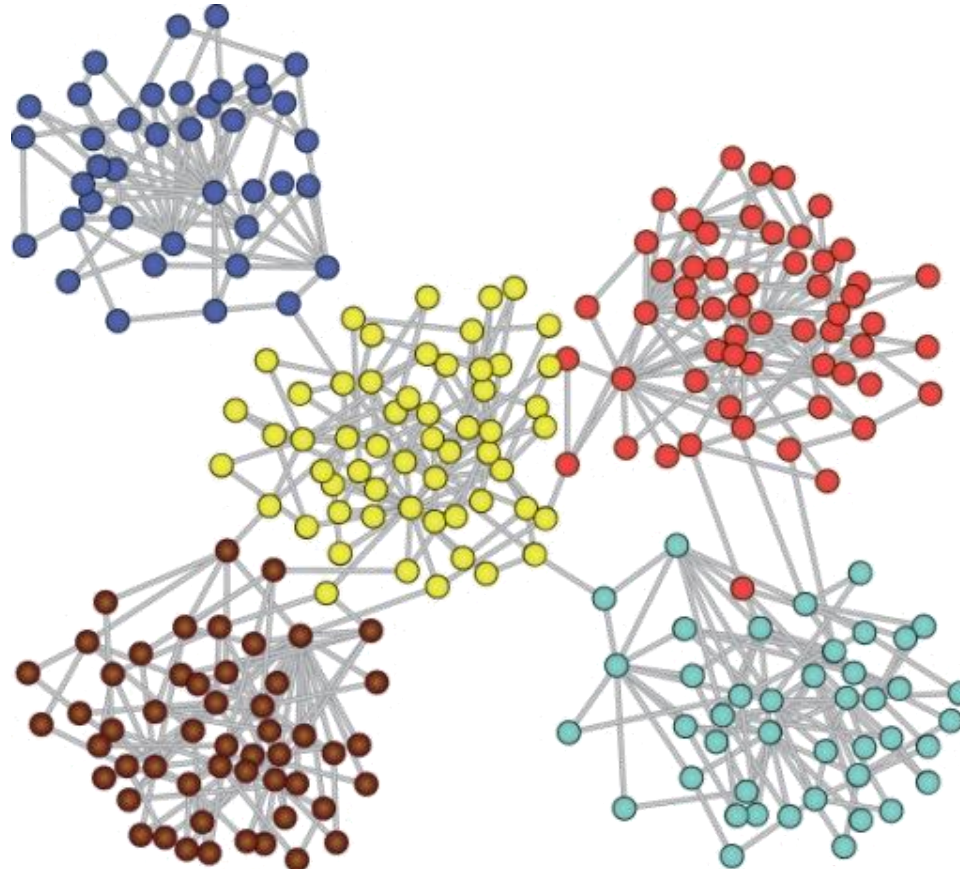


Figure credit: <https://github.com/benedekrozemberczki/awesome-community-detection>

# Graph clustering

## ■ Modularity

- Measurement of the strength of partitioning modules into modules (clusters)

$$Q(\gamma) = \frac{1}{2m} \sum_{ij} \left[ A_{ij} - \frac{d_i d_j}{2m} \right] \delta(\gamma(v_i), \gamma(v_j))$$

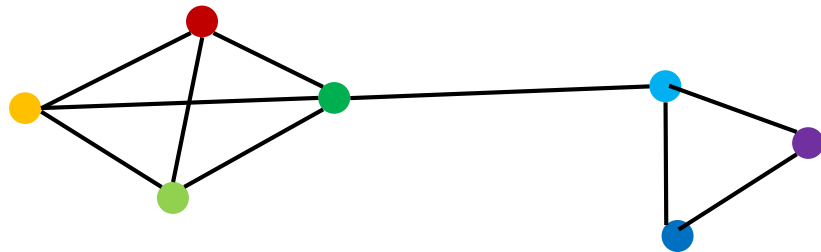
- $d_i$ : degree of the node  $i$  ( $d_i = \sum_j A_{ij}$ ) and  $m = \frac{1}{2} \sum_{ij} A_{ij}$
  - $\frac{d_i d_j}{2m}$ : probability of an edge existing between  $v_i$  and  $v_j$  in the random null model
  - the fraction of the edges inside cluster minus the expected fraction if edges were distributed at random
- 
- The high modularity means that there are dense connections between the nodes in the same cluster but sparse connections between nodes in different clusters

# Graph clustering

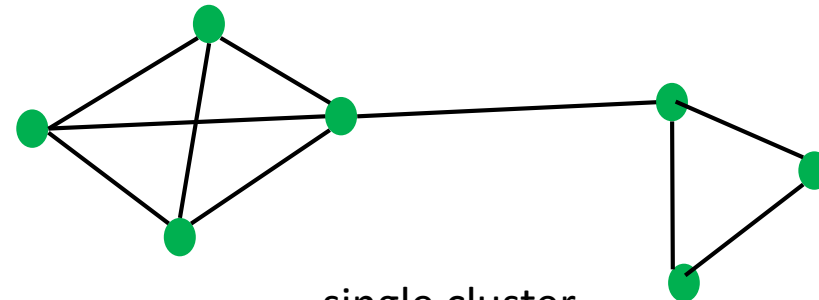
## ■ Modularity

$$Q(\gamma) = \frac{1}{2m} \sum_{ij} \left[ A_{ij} - \frac{d_i d_j}{2m} \right] \delta(\gamma(v_i), \gamma(v_j))$$

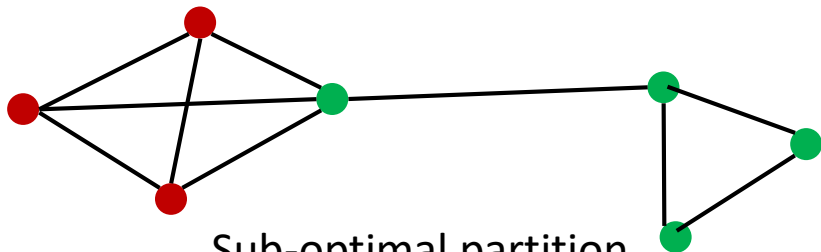
The high modularity means that there are dense connections between the nodes in the same cluster but sparse connections between nodes in different clusters



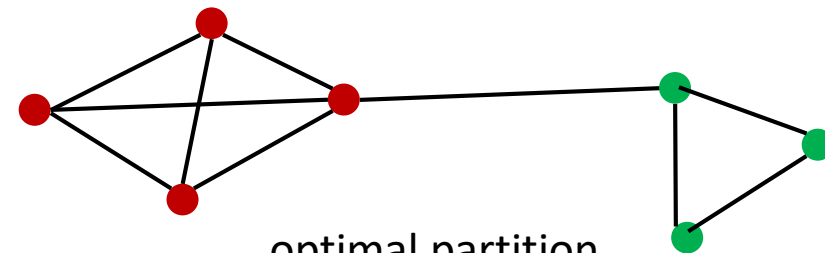
# of clusters = # of nodes



single cluster



Sub-optimal partition



optimal partition

# Clustering on graphs

- Louvain algorithm
  - greedily maximizes the modularity score by using an agglomerative approach
  - one of the fastest modularity-based algorithms and scalable

**Fast unfolding of communities in large networks**

**Vincent D. Blondel<sup>1;a</sup>, Jean-Loup Guillaume<sup>1,2;b</sup>, Renaud Lambiotte<sup>1,3;c</sup> and Etienne Lefebvre<sup>1</sup>**

<sup>1</sup>Department of Mathematical Engineering, Université catholique de Louvain, 4 avenue Georges Lemaitre, B-1348 Louvain-la-Neuve, Belgium

<sup>2</sup> LIP6, Université Pierre et Marie Curie, 4 place Jussieu, 75005 Paris, France

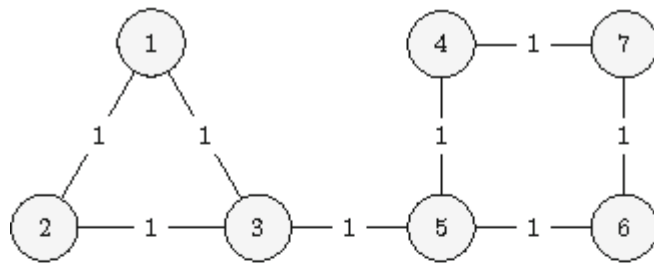
<sup>3</sup> Institute for Mathematical Sciences, Imperial College London, 53 Prince's Gate, South Kensington campus, SW72PG, UK

E-mail: <sup>a</sup>[vincent.blondel@uclouvain.be](mailto:vincent.blondel@uclouvain.be); <sup>b</sup>[jean-loup.guillaume@lip6.fr](mailto:jean-loup.guillaume@lip6.fr);

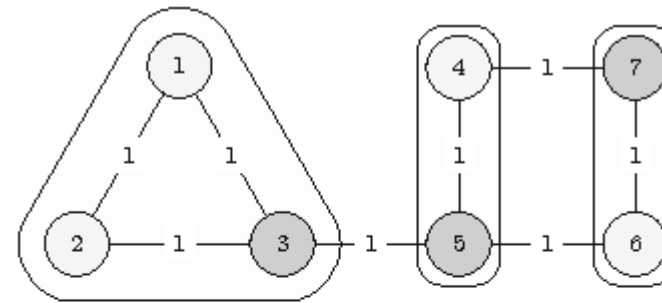
<sup>c</sup>[r.lambiotte@imperial.ac.uk](mailto:r.lambiotte@imperial.ac.uk);



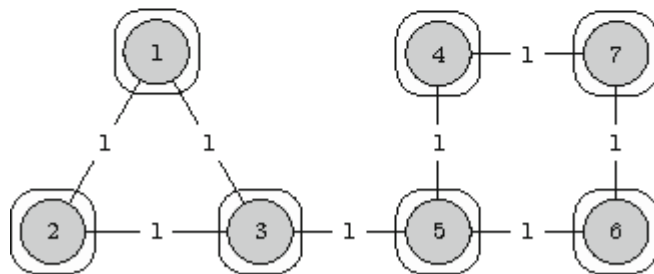
# Clustering on graphs (Louvain algorithm)



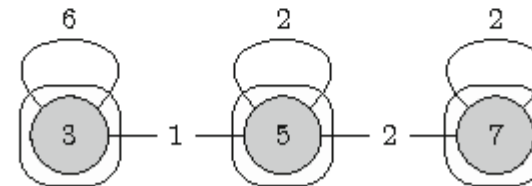
**(a)** original network



**(c)** step 1 of 1<sup>st</sup> iteration



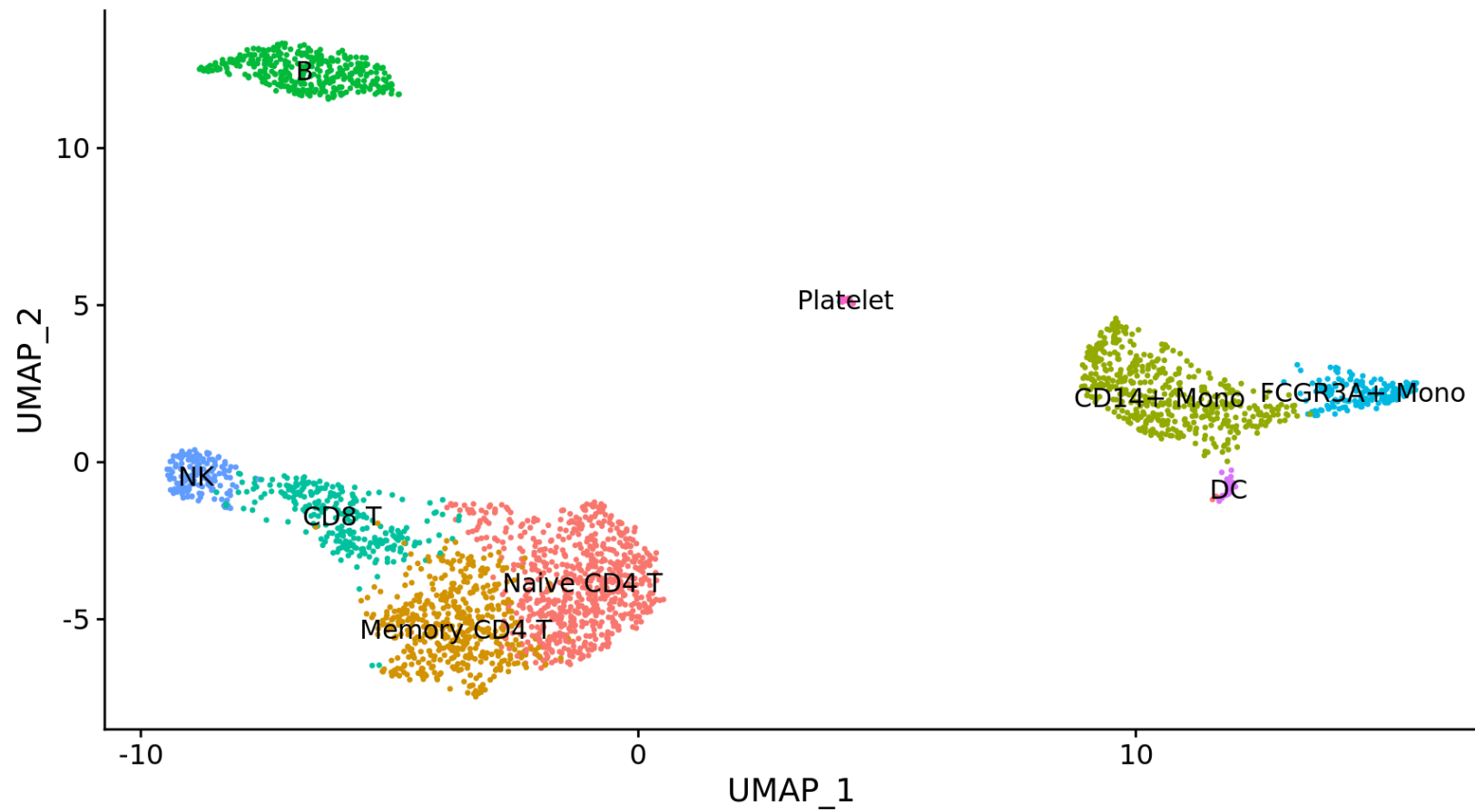
**(b)** initial communities



**(d)** step 2 of 1<sup>st</sup> iteration

# Cell type identification

## ■ Identify

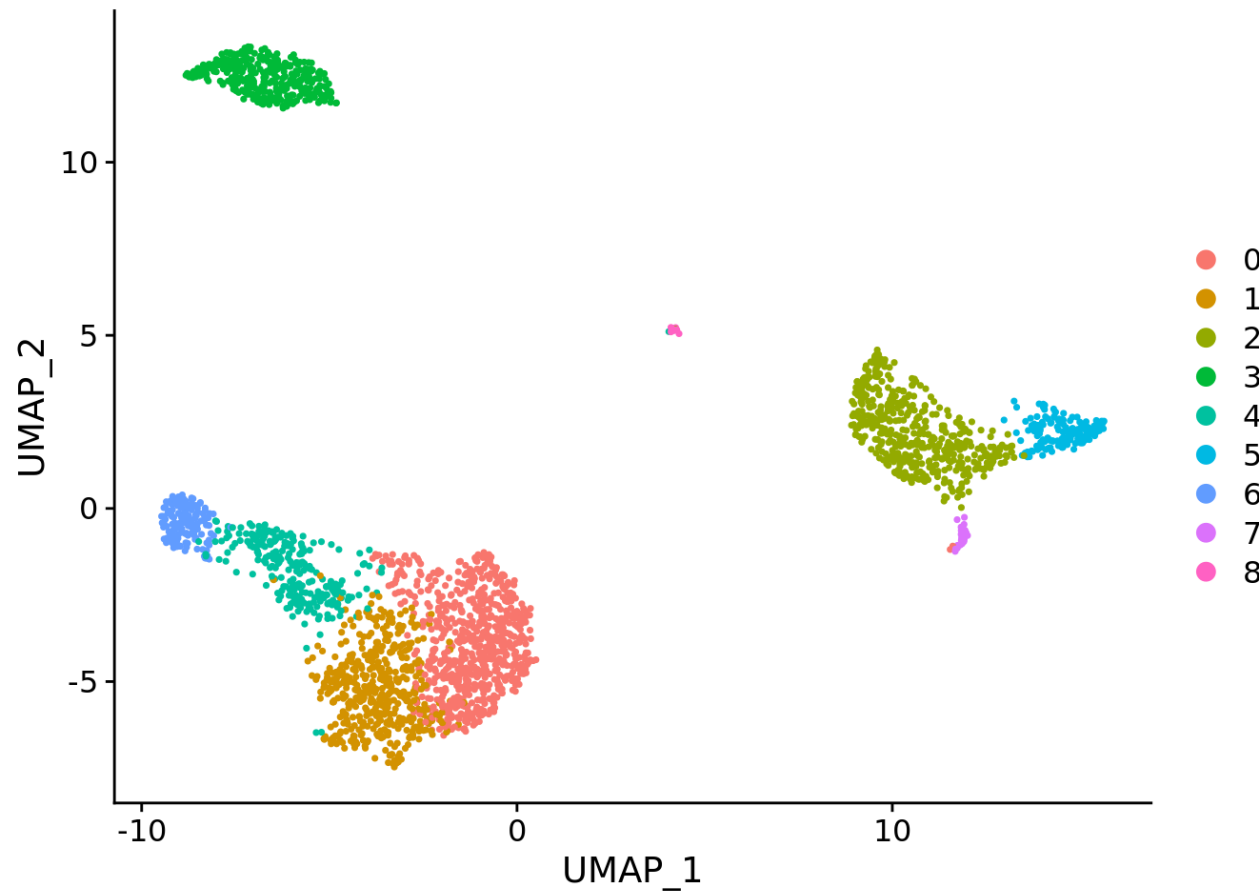


# Cell type identification

- Clustering step
  - Find sub-groups in single-cell data using any clustering methods (e.g., Louvain algorithm)
- Assignment step
  - Marker identification: finding differently expressed genes in a cluster compared to other cells (or to those in other clusters)
  - Assigning cell type identity to clusters by matching marker genes in known cell type signatures

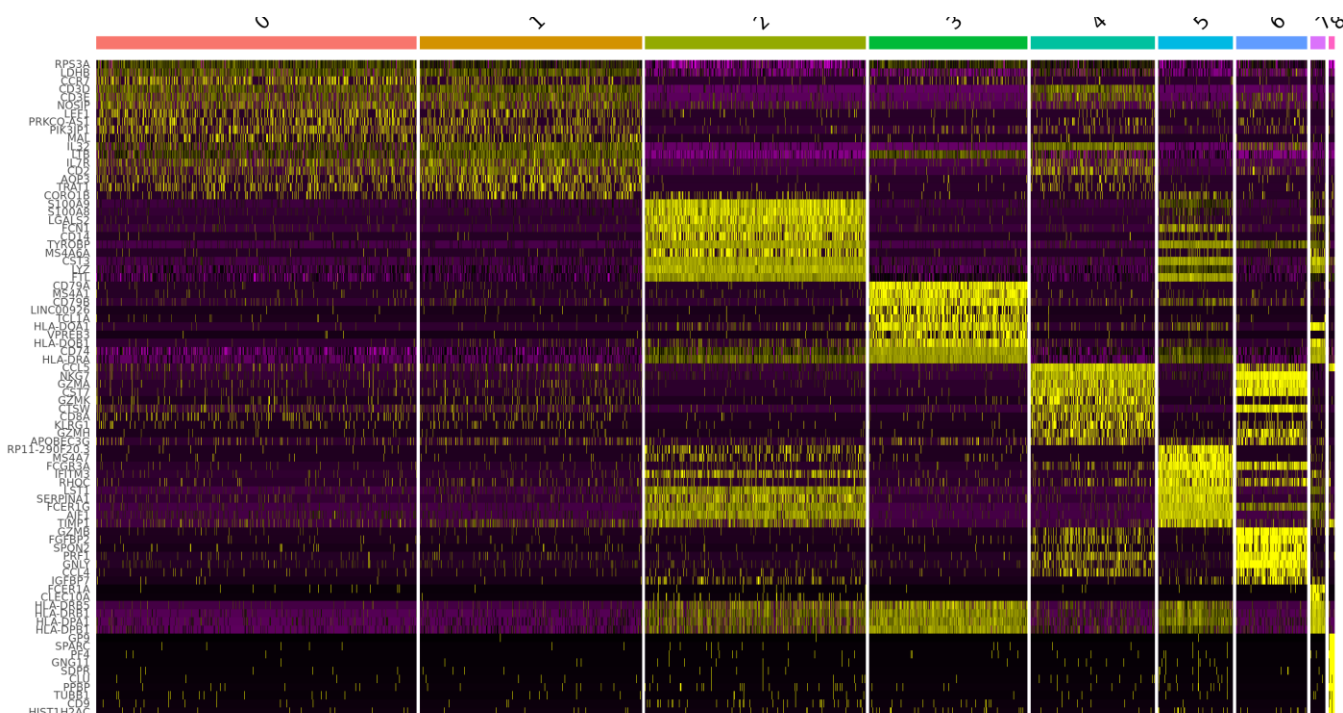
# Cell type identification

## ■ Clustering



# Cell type identification

- Marker gene identification: finding differently expressed genes
- Cell type assignment

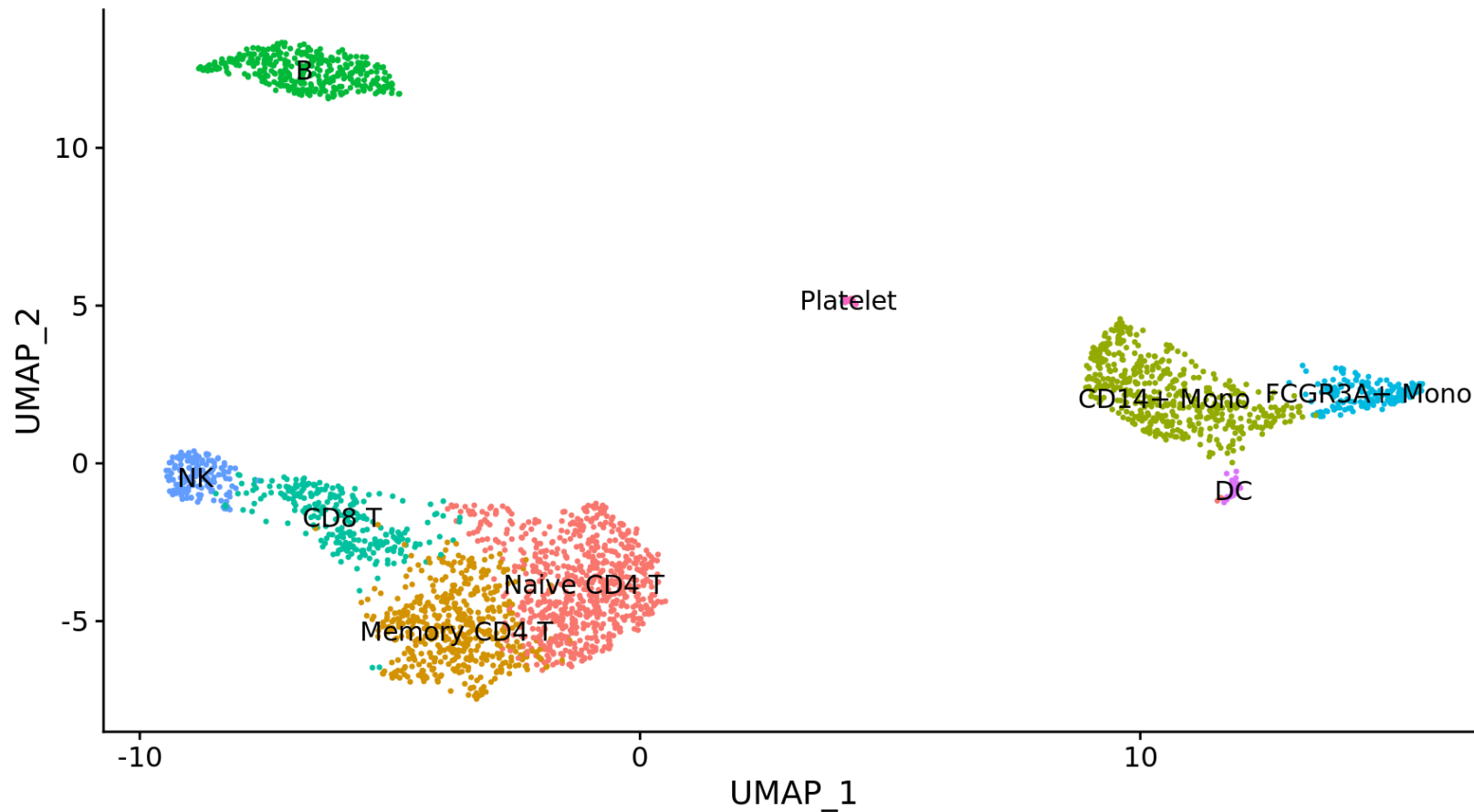


Markers	Cell Type
IL7R, CCR7	Naive CD4+ T
IL7R, S100A4	Memory CD4+
CD14, LYZ	CD14+ Mono
MS4A1	B
CD8A	CD8+ T
FCGR3A, MS4A7	FCGR3A+ Mono
GNLY, NKG7	NK
FCER1A, CST3	DC
PPBP	Platelet

Figure credit: Seurat package

# Cell type identification

- Final results



# Outline

- Dimensionality reduction (visualization)
  - t-SNE
  - Uniform Manifold Approximation and Projection (UMAP)
- Clustering approaches
  - Graph based methods: Louvain algorithm
  - Cell type identification
- **Multiple Dataset Integration**
  - Canonical correlation analysis & L2-norm normalization
  - Anchoring

# Multiple Dataset Integration

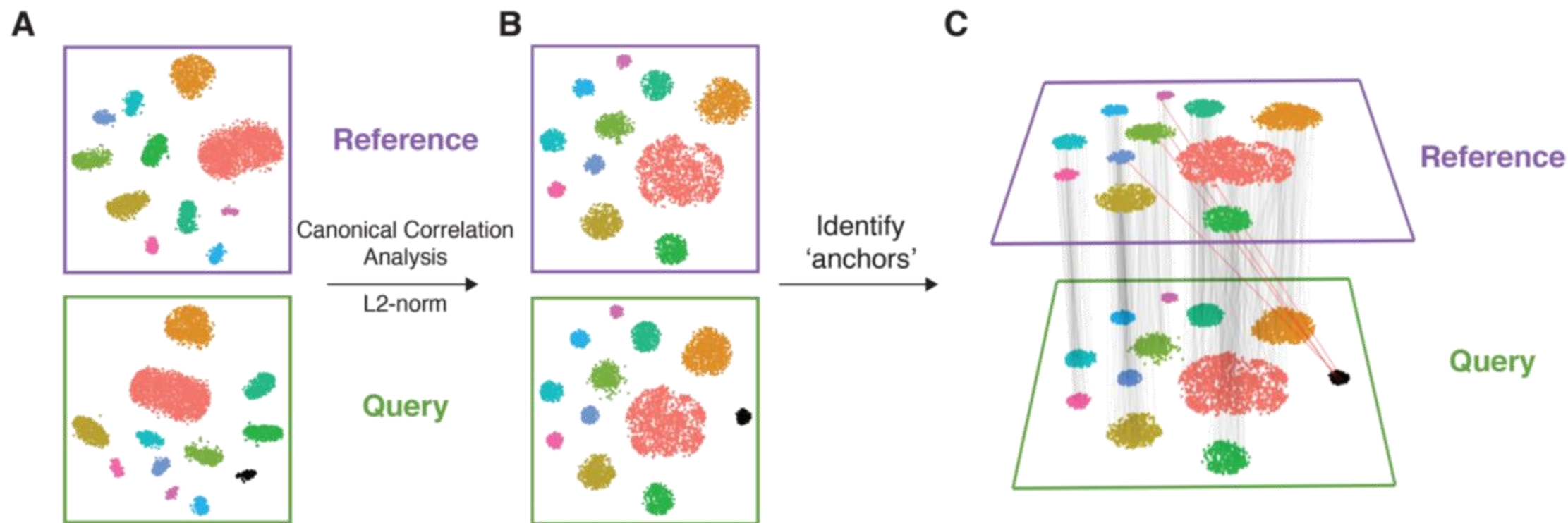


Figure: Tim Stuart et al. "Comprehensive Integration of Single-Cell Data", Cell 2019



# Multiple Dataset Integration

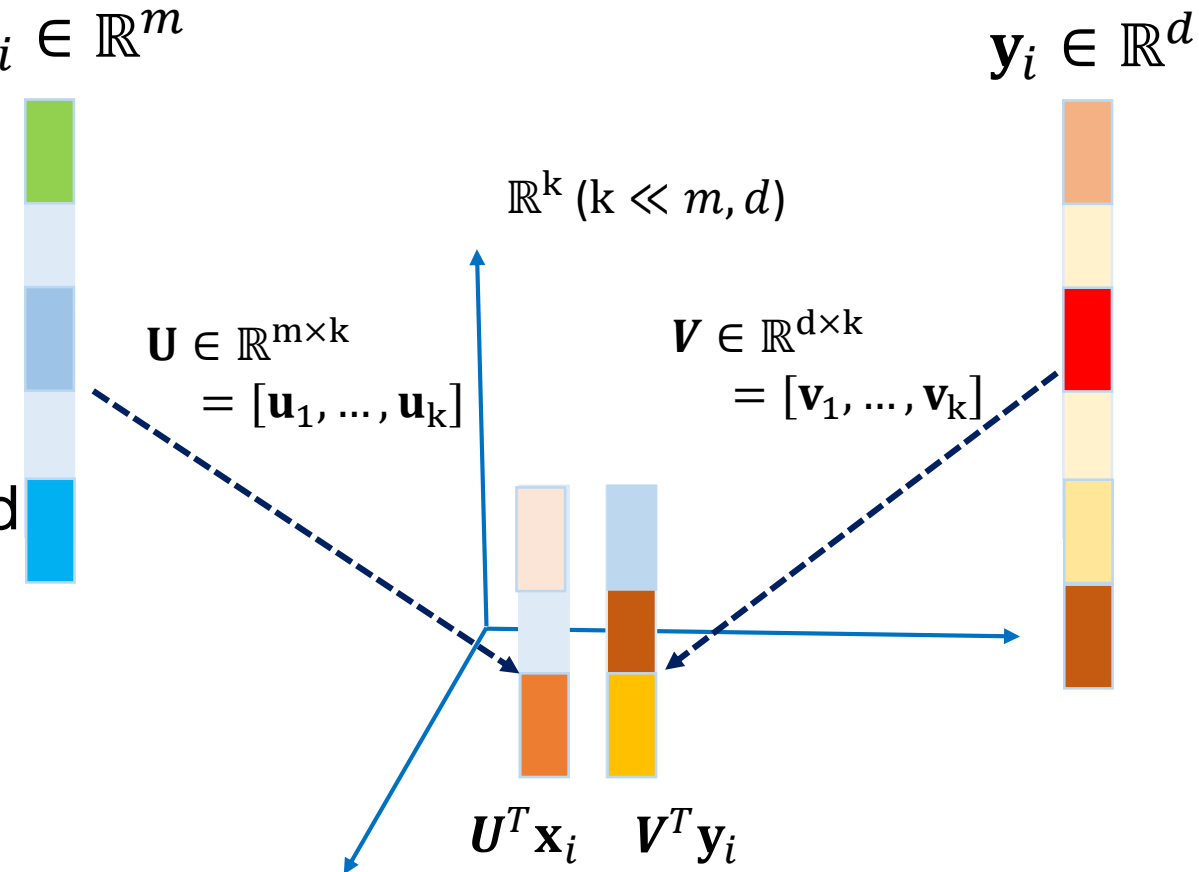
- Canonical correlation analysis (CCA)
- Finds a common space of two sets of data  $\mathbf{x}_i \in \mathbb{R}^m$

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^T$$

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]^T$$

- Finds a linear transformation of variables that makes the two variables maximally correlated

$$\hat{\mathbf{u}}, \hat{\mathbf{v}} = \operatorname{argmax}_{\mathbf{u}, \mathbf{v}} \mathbf{u}^T \mathbf{X}^T \mathbf{Y} \mathbf{v} \\ \text{s.t. } \mathbf{u}^T \mathbf{X}^T \mathbf{X} \mathbf{u} \leq 1, \mathbf{v}^T \mathbf{Y}^T \mathbf{Y} \mathbf{v} \leq 1$$



# Multiple Dataset Integration

- Canonical correlation analysis (CCA)
- Finds a common space of two sets of data  $\mathbf{x}_i \in \mathbb{R}^m$

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^T$$

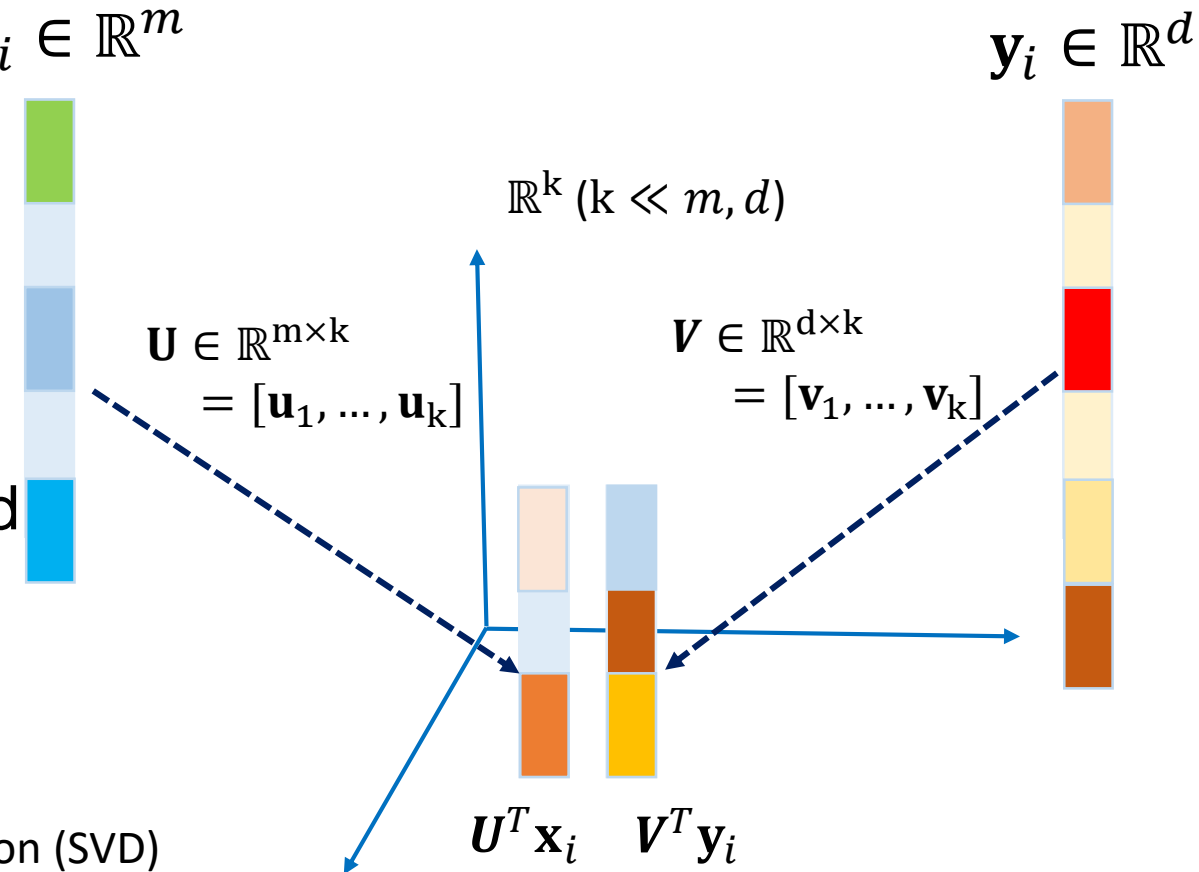
$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]^T$$

- Finds a linear transformation of variables that makes the two variables maximally correlated

In Seurat, the maximization problem is simplified as

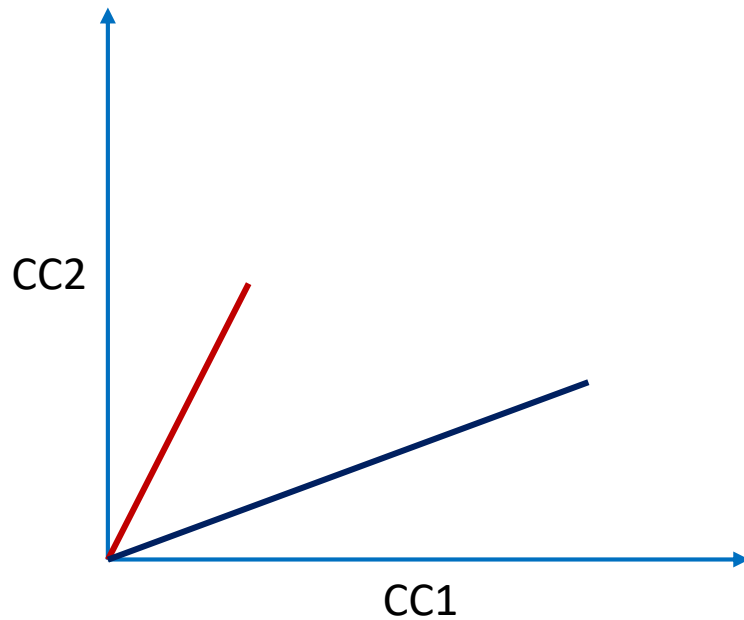
$$\begin{aligned} \hat{\mathbf{u}}, \hat{\mathbf{v}} &= \operatorname{argmax}_{\mathbf{u}, \mathbf{v}} \mathbf{u}^T \mathbf{X}^T \mathbf{Y} \mathbf{v} \\ \text{s.t. } &||\mathbf{u}||_2^2 \leq 1, ||\mathbf{v}||_2^2 \leq 1 \end{aligned}$$

The problem can be solved using singular value decomposition (SVD)

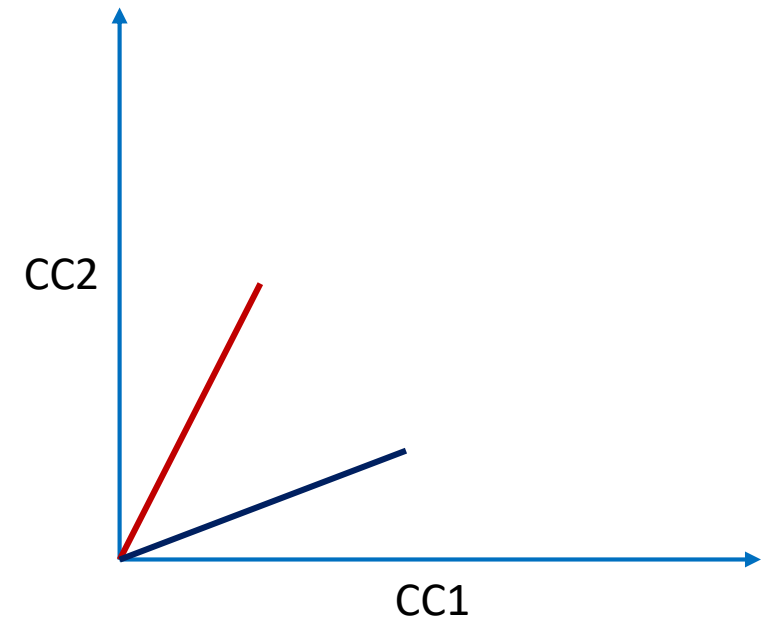


# Multiple Dataset Integration

- $L_2$ -norm normalization

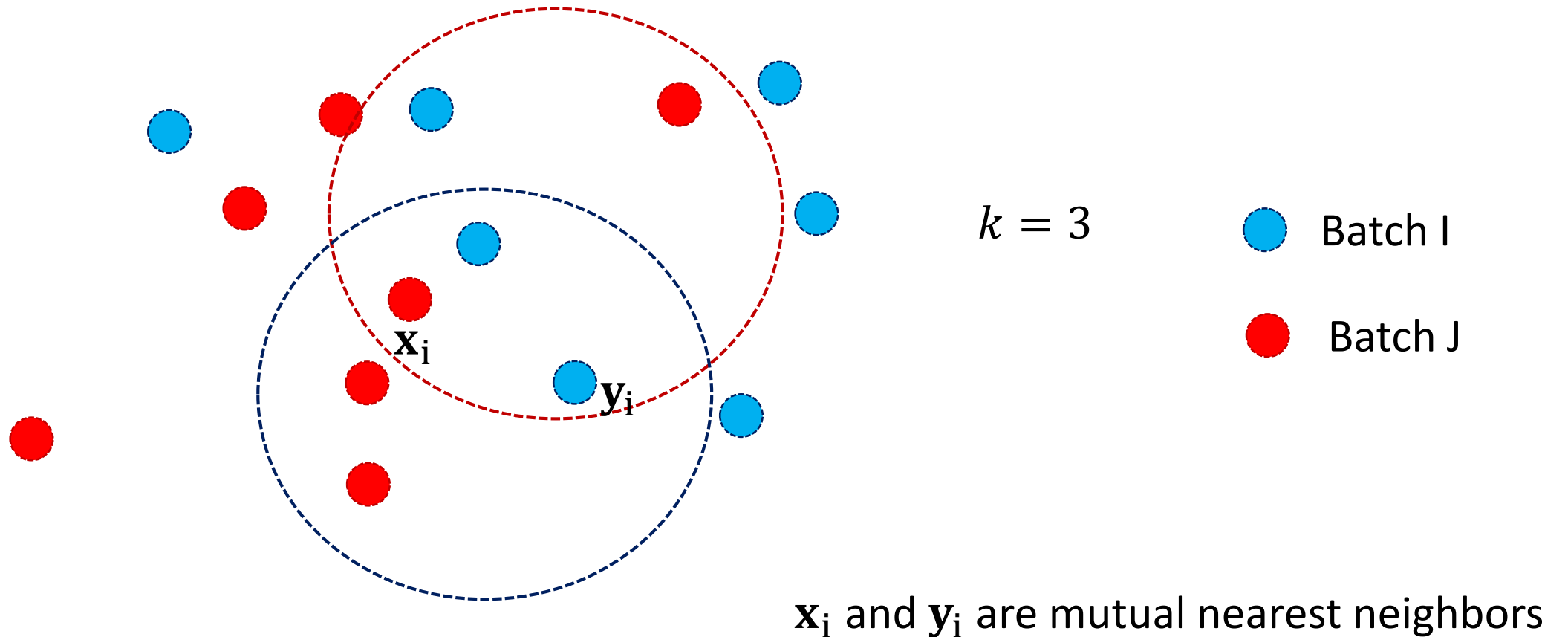


$$\frac{\mathbf{x}}{\|\mathbf{x}\|_2}$$



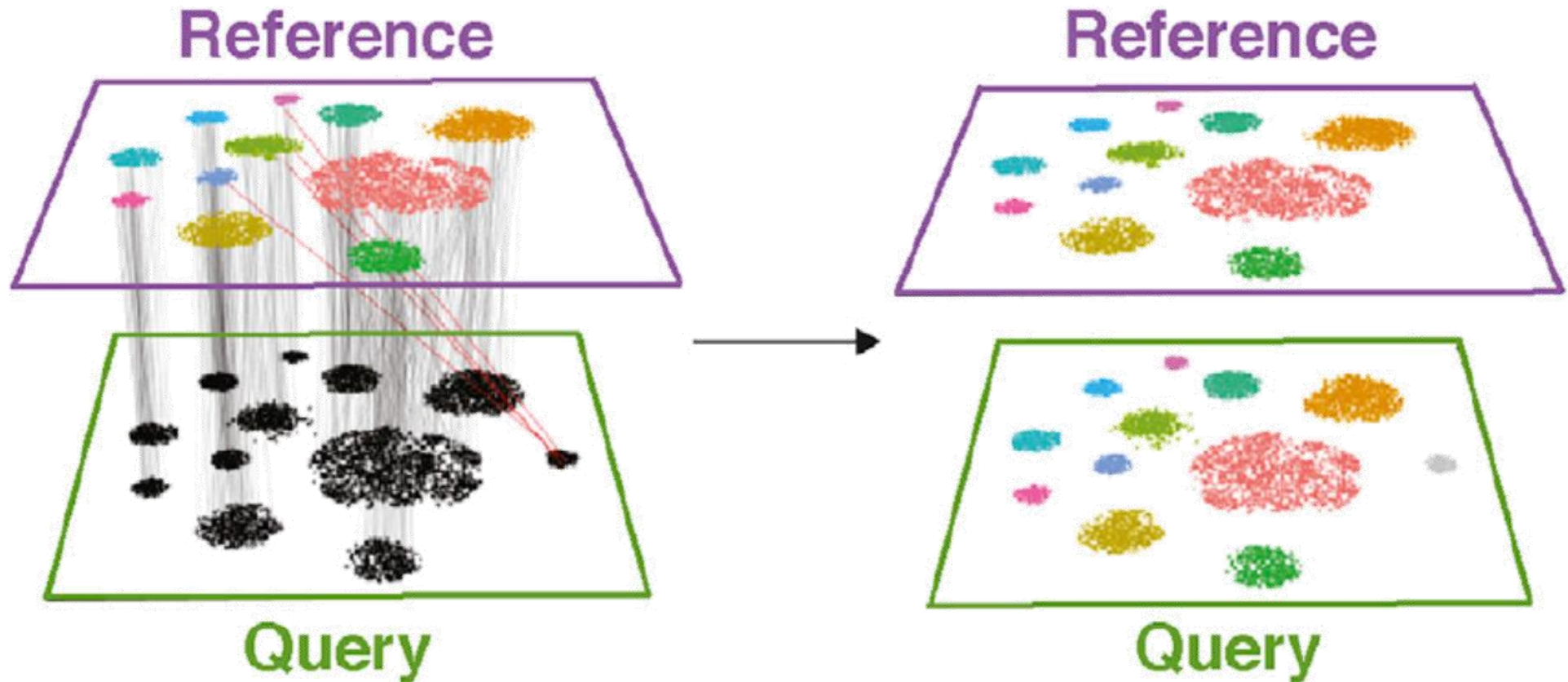
# Multiple Dataset Integration

- Anchors (Mutual nearest neighbors)



# Multiple Dataset Integration

- Label transfer



# References

- Seurat package: <https://satijalab.org/seurat/>
- Stuart, Tim et al. “Comprehensive Integration of Single-Cell Data” Cell, Volume 177, Issue 7, 1888 - 1902.e21, 2019
- How Exactly UMAP Works:  
<https://towardsdatascience.com/how-exactly-umap-works-13e3040e1668>
- V. Blondel et al., “Fast unfolding of communities in large networks”, Journal of Statistical Mechanics: Theory and Experiment 2008