COMPARISON OF DIMENSIONALITY REDUCTION METHODS FOR SIMULATED SINGLE-CELL DATA

JUAN DIEGO ARIZA SÁNCHEZ
LILIANA LÓPEZ KLEINE

DEPARTMENT OF STATISTICS
UNIVERSIDAD NACIONAL DE COLOMBIA

UNIVERSIDAD
NACIONAL
DE COLOMBIA

AGENDA

1 INTRODUCTION

- Single-Cell Data

2 METHODOLOGY

- Workflow
- Simulation design

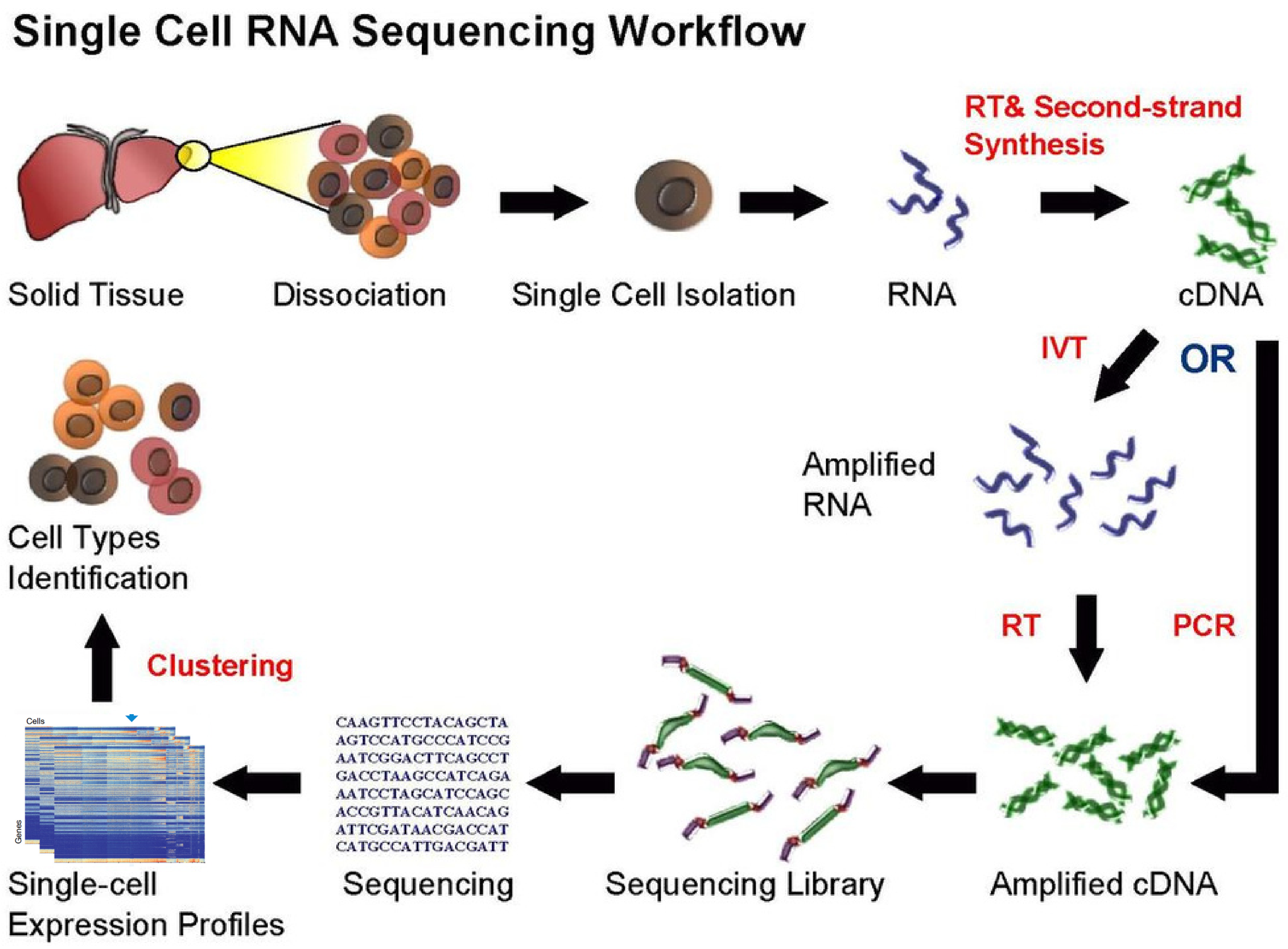
3 RESULTS

- Simulations and dimensionality reduction
- Clustering and assessment

4 DISCUSSION

- Future Work

SINGLE-CELL DATA



Single-Cell RNA sequencing

- Advantages:

Information of each cell individually

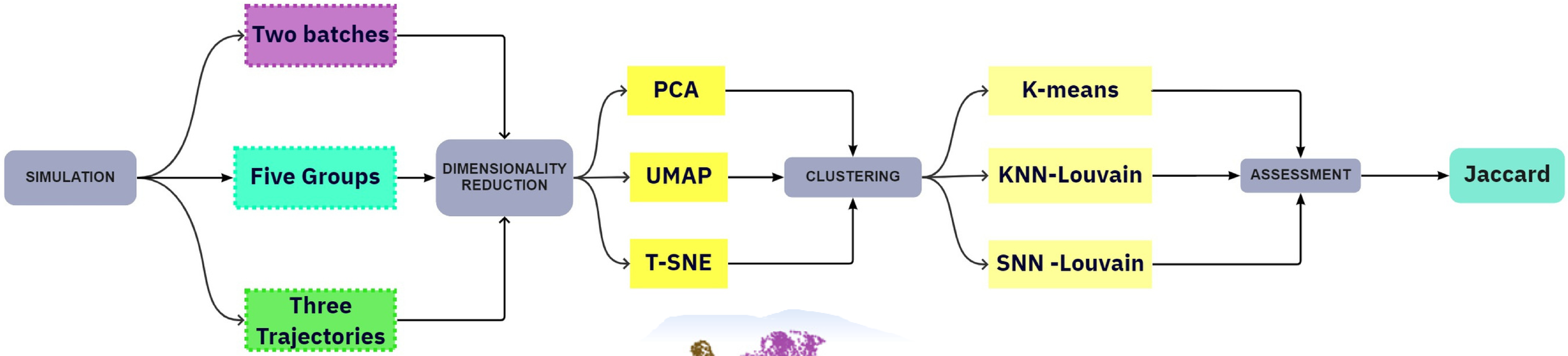
.

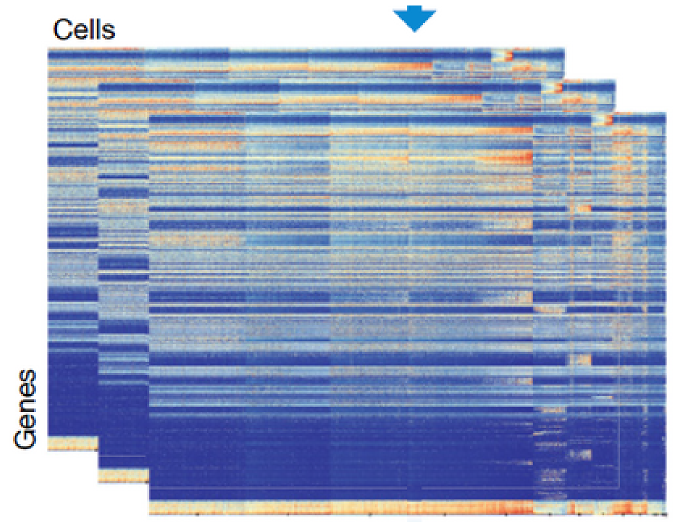
- Challenges:

Extract valuable information from sparse but high dimensional data to reveal new cell types, dynamics and regulation.

METHODOLOGY

WORKFLOW





$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

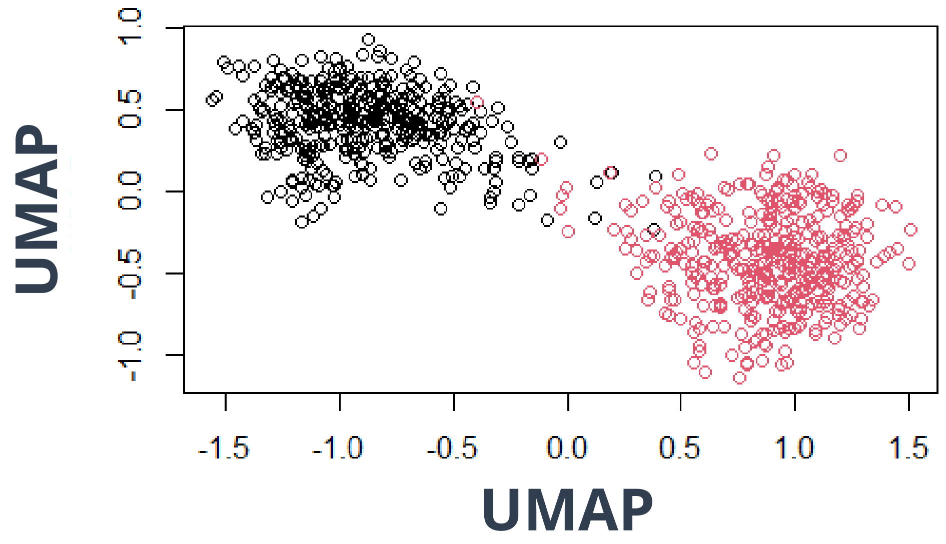
SIMULATION DESIGN

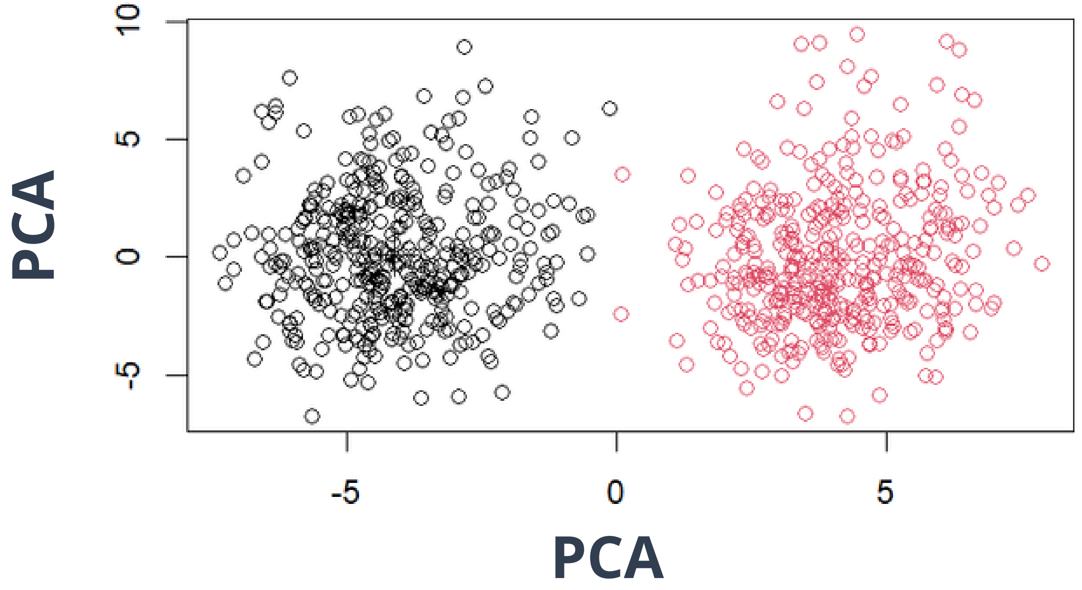
Types of simulation	Batches			Groups			Trajectories		
Clustering \Dimensionality reduction methods	*UMAP*	*PCA*	T-SNE	*UMAP*	*PCA*	T-SNE	*UMAP*	*PCA*	T-SNE
K-means	20	20	20	20	20	20	20	20	20
KNN - Louvain	20	20	20	20	20	20	20	20	20
SNN - Louvain	20	20	20	20	20	20	20	20	20

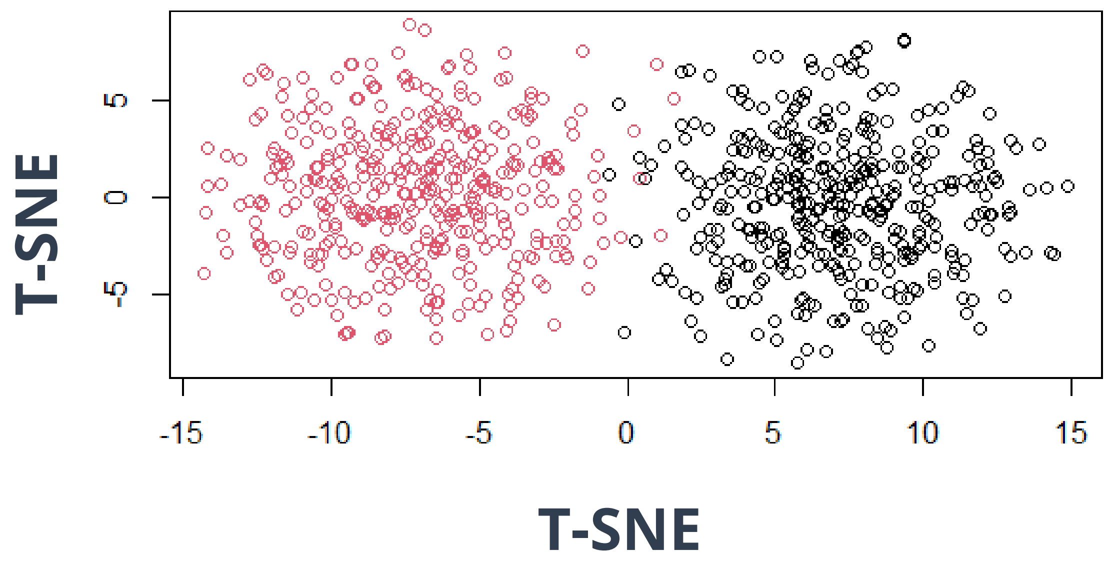
```
library("scater")
library("splatter")
library("scran")
```

RESULTS

CLUSTERING: TWO ORIGINAL SIMULATED BATCHES

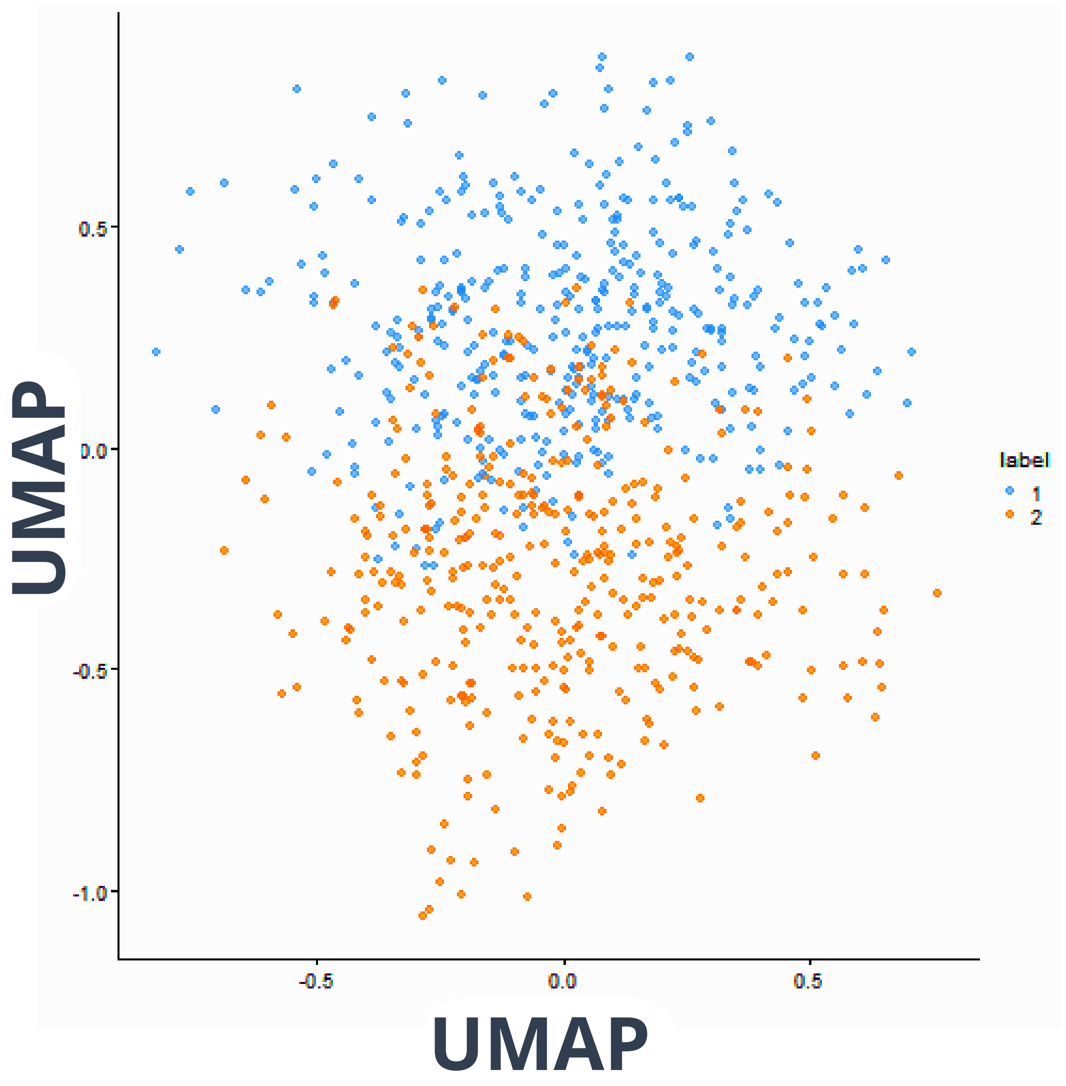


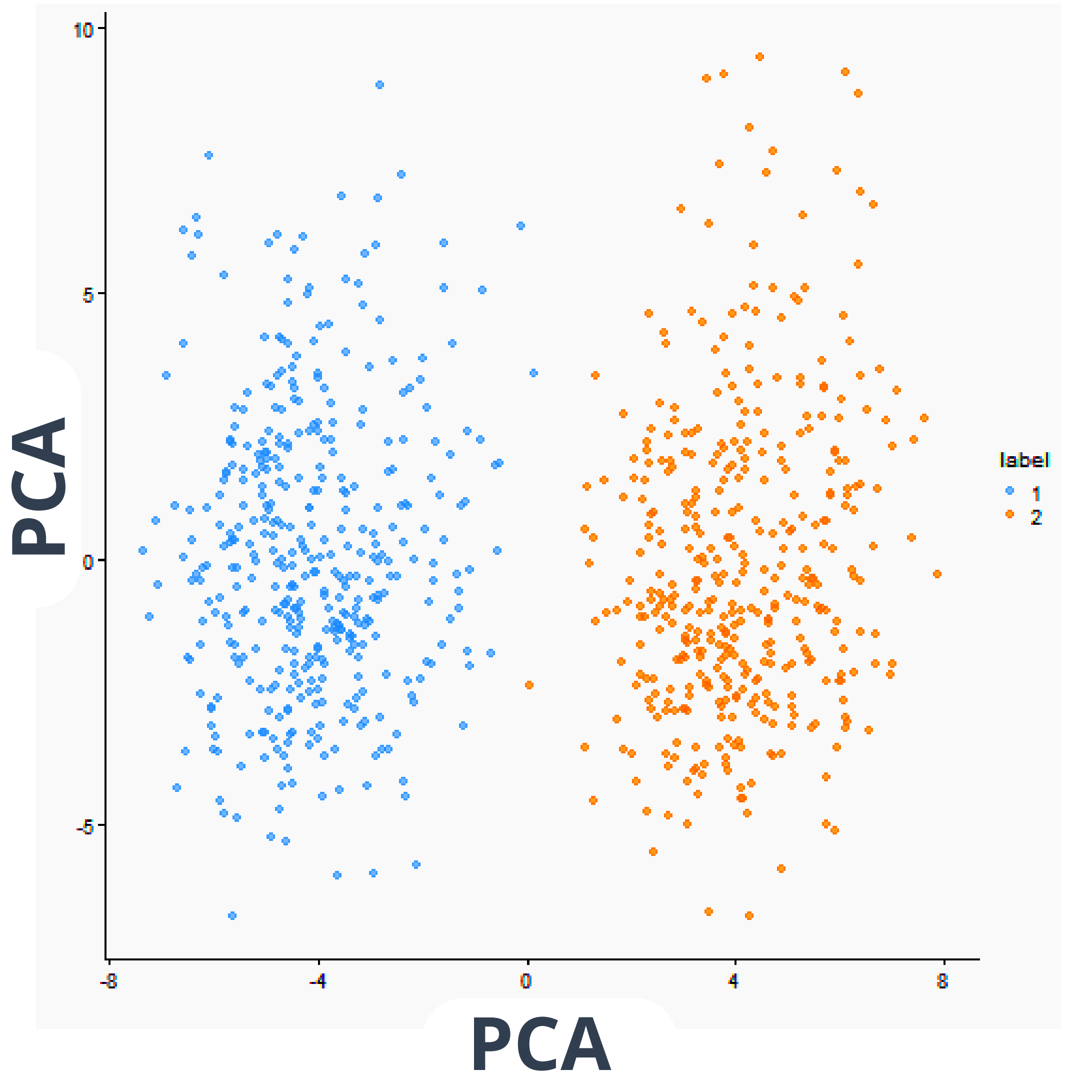


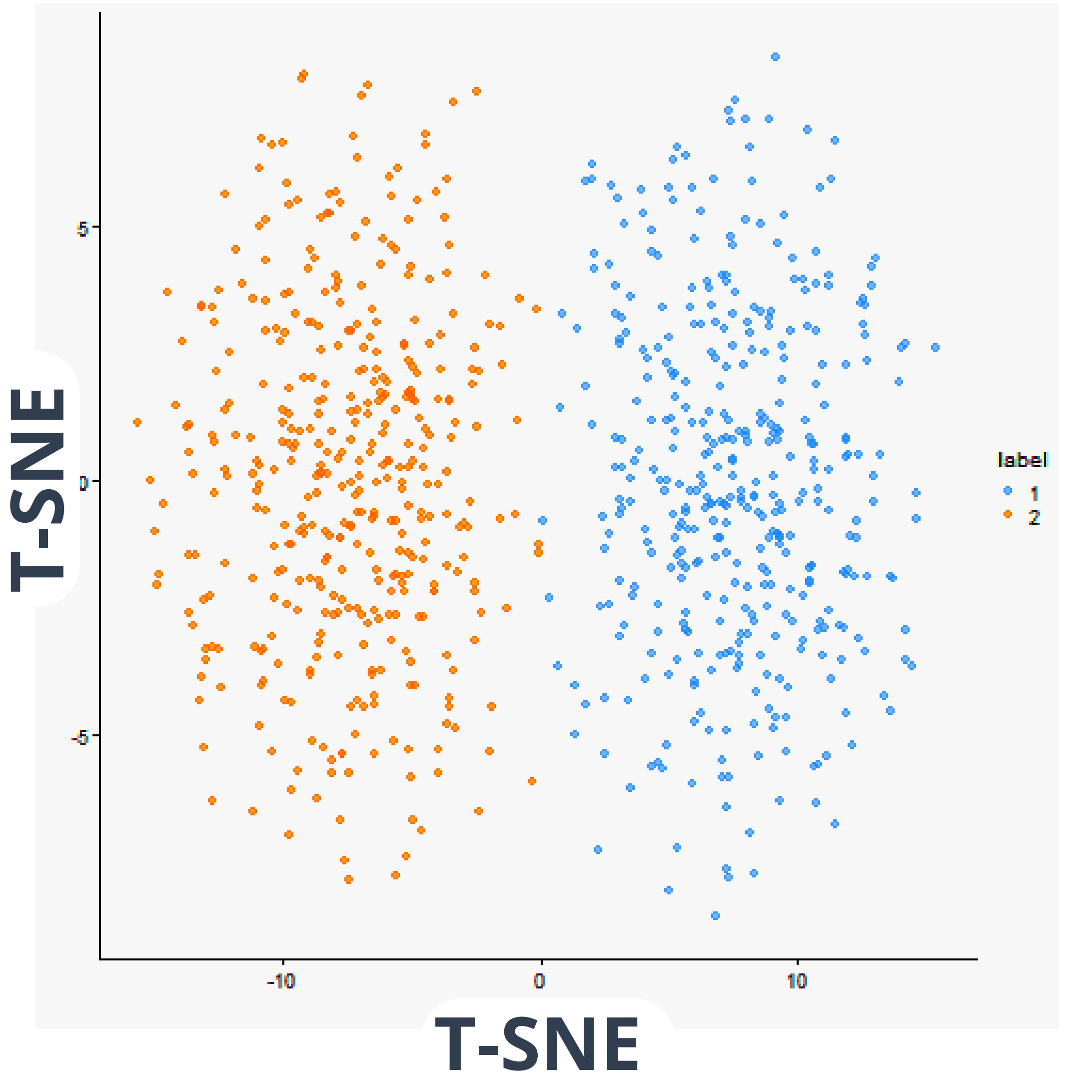


K-means

CLUSTERING: TWO ORIGINAL SIMULATED BATCHES

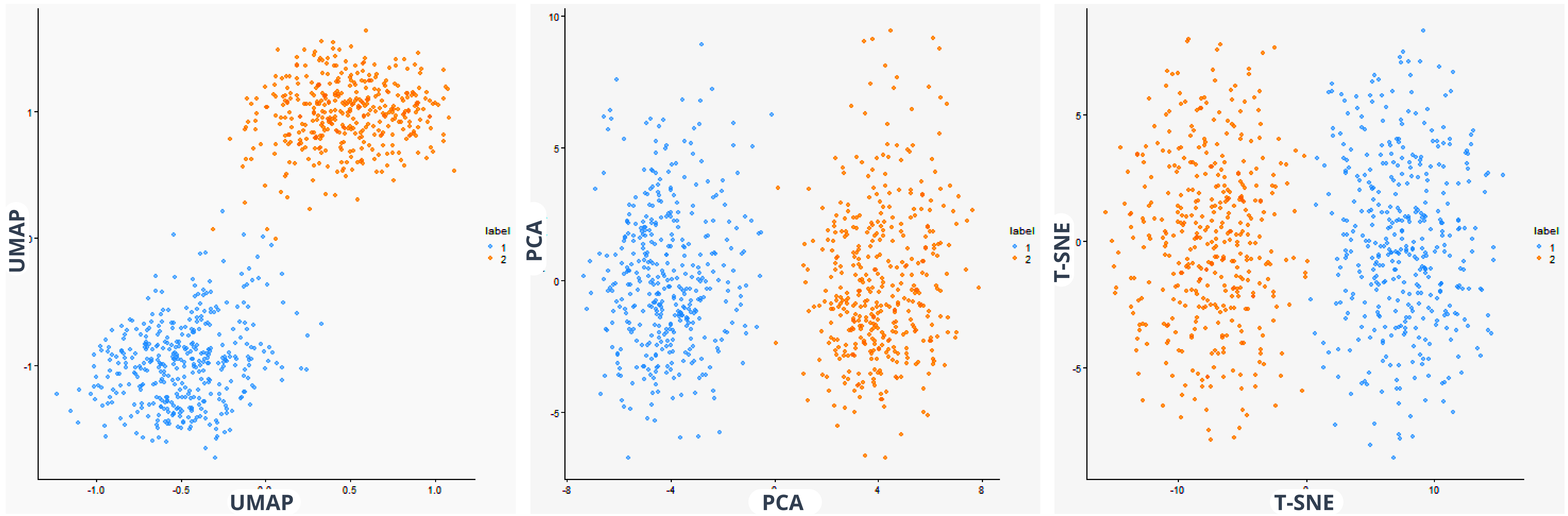






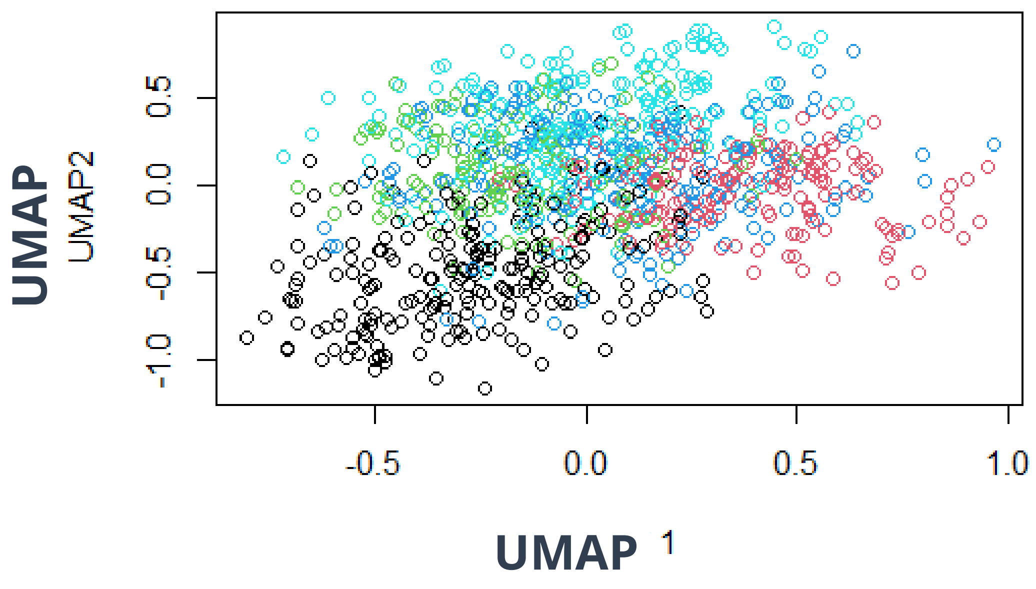
KNN + LOUVAIN

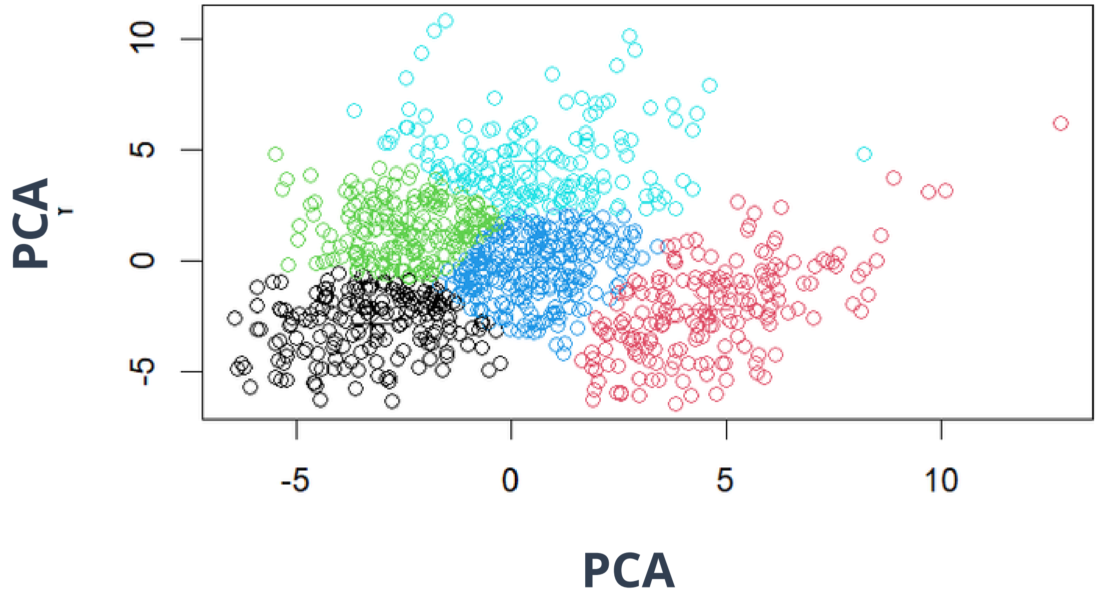
CLUSTERING: TWO ORIGINAL SIMULATED BATCHES

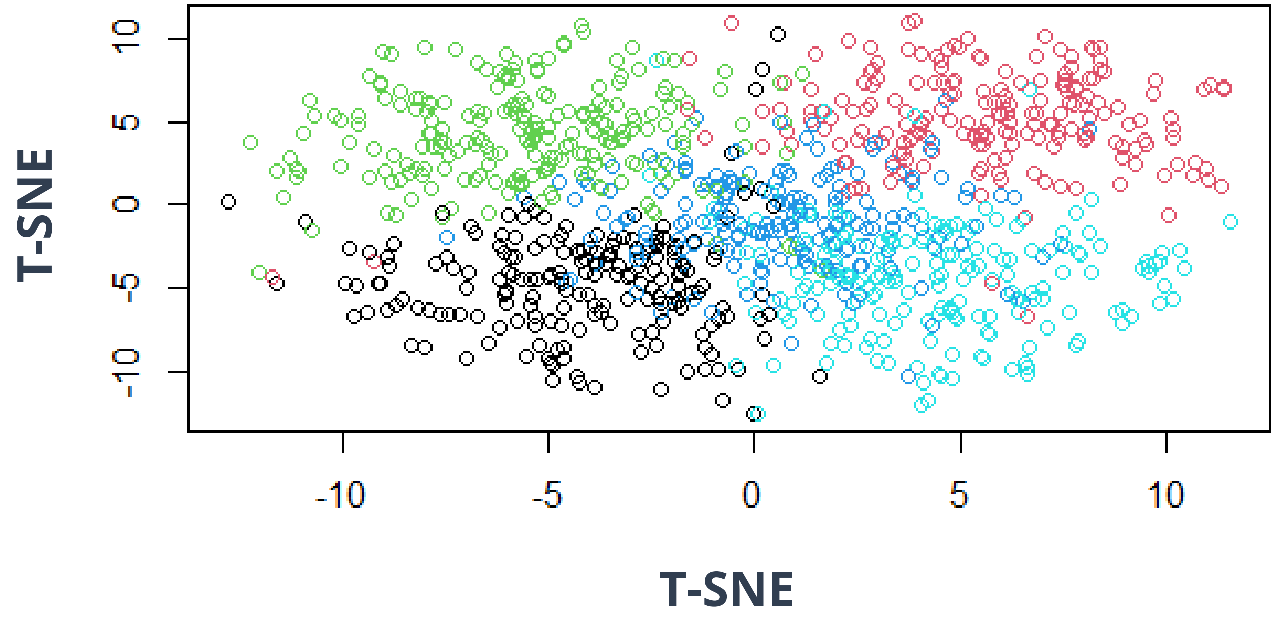


SNN + LOUVAIN

CLUSTERING: FIVE ORIGINAL SIMULATED GROUPS

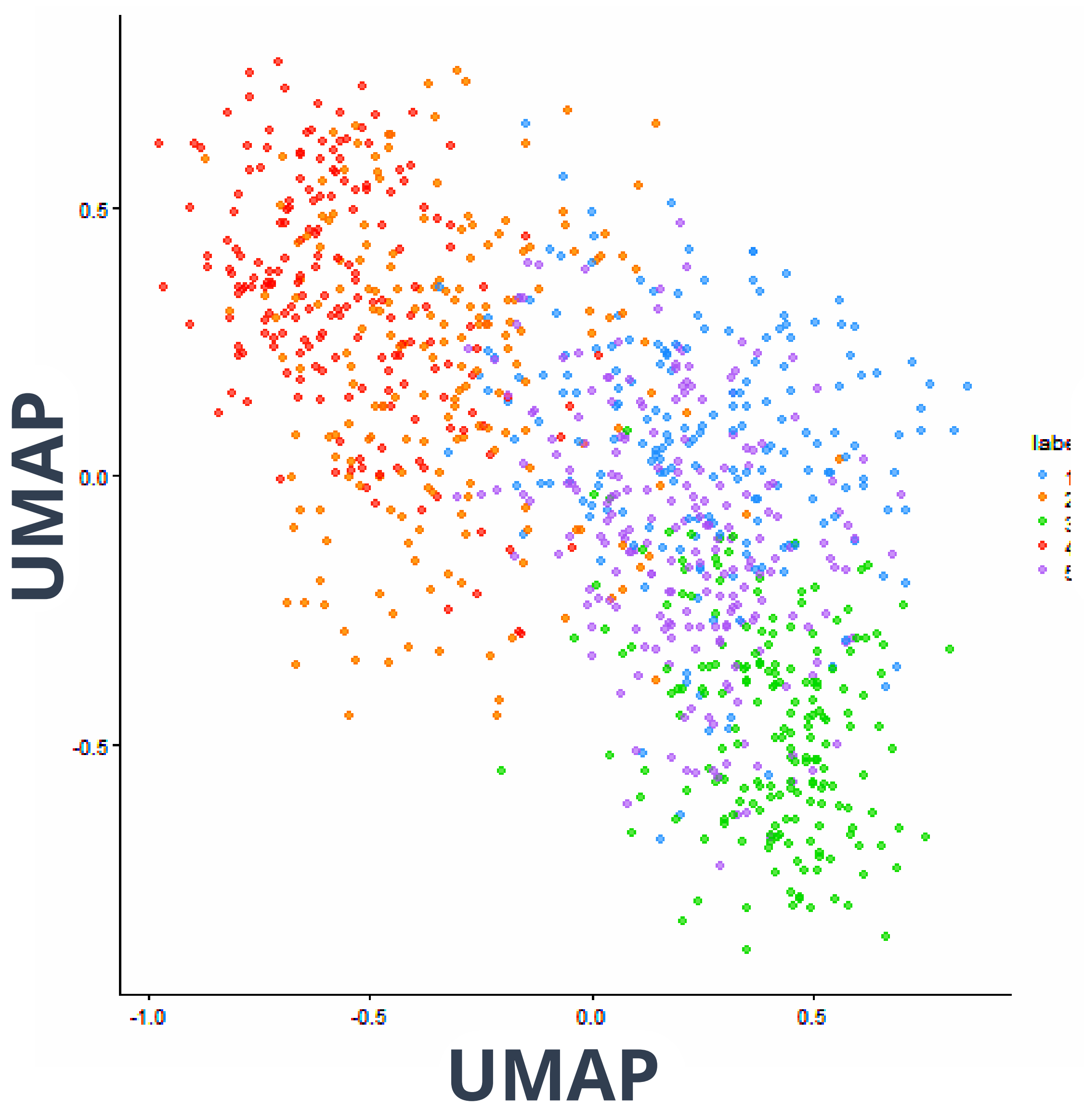


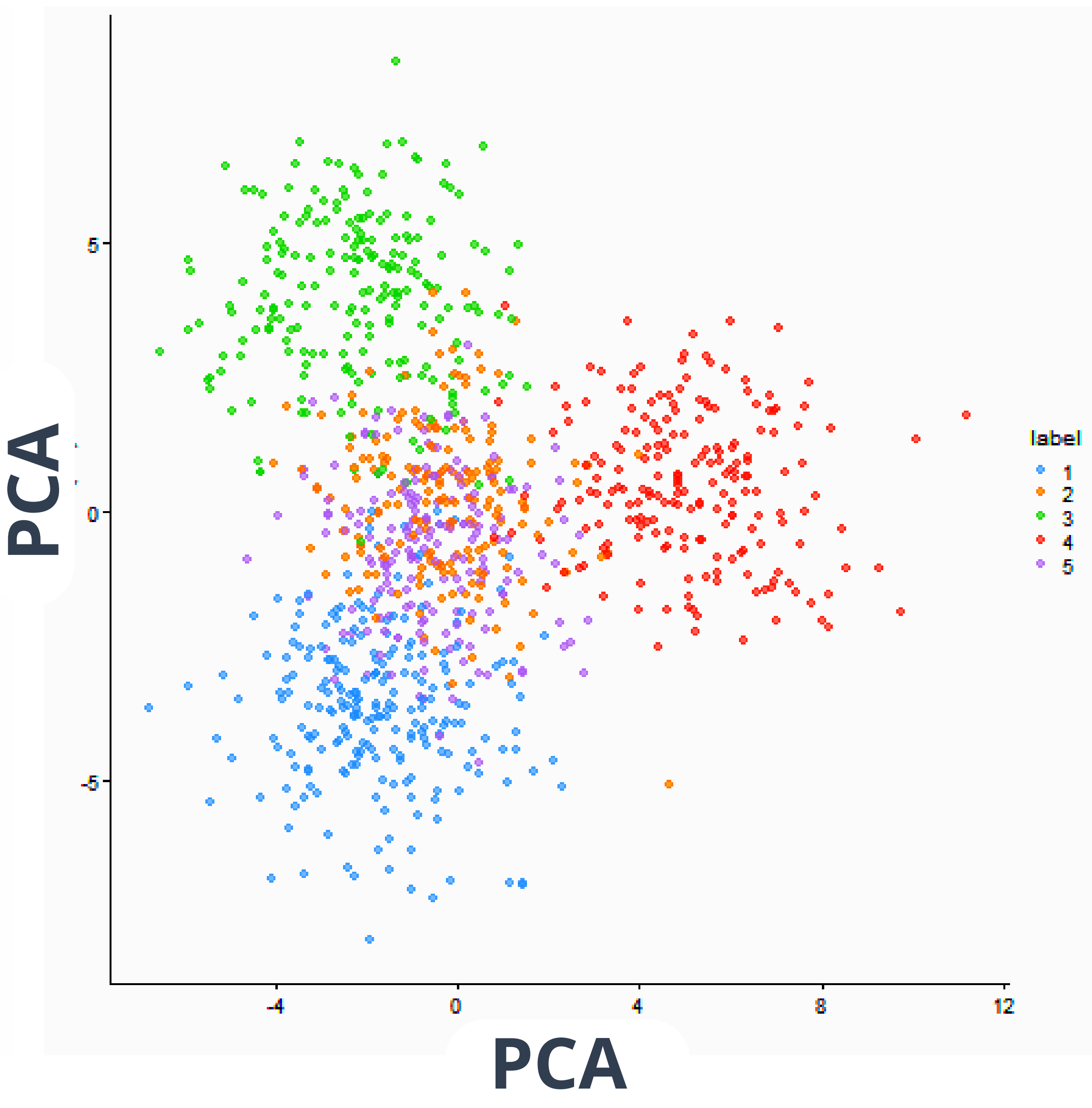


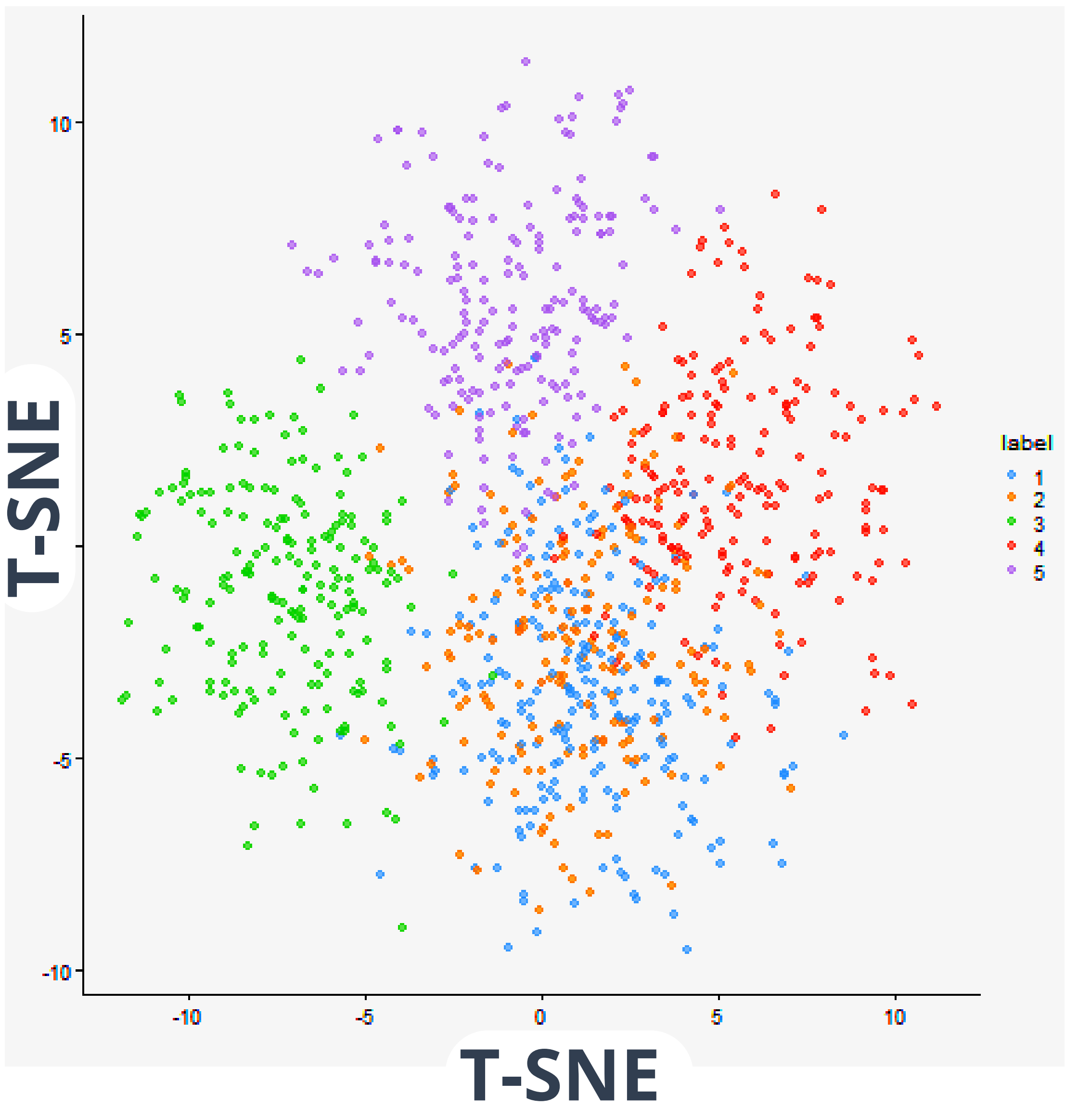


K-means

CLUSTERING: FIVE ORIGINAL SIMULATED GROUPS

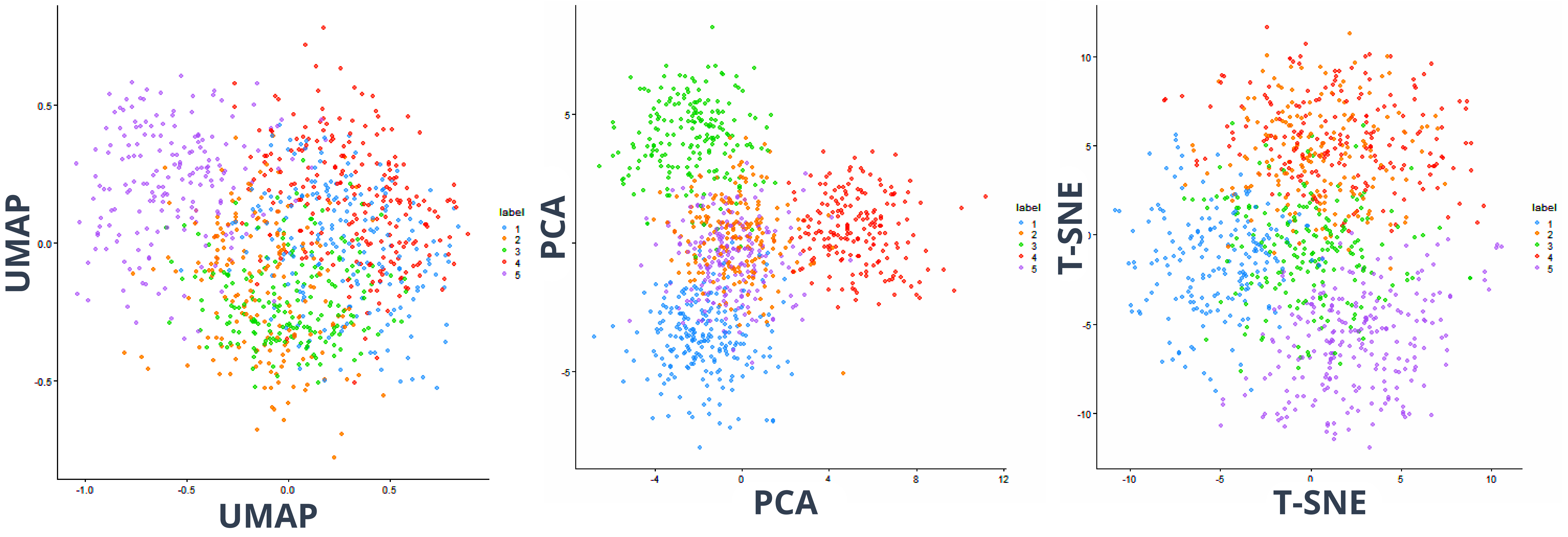






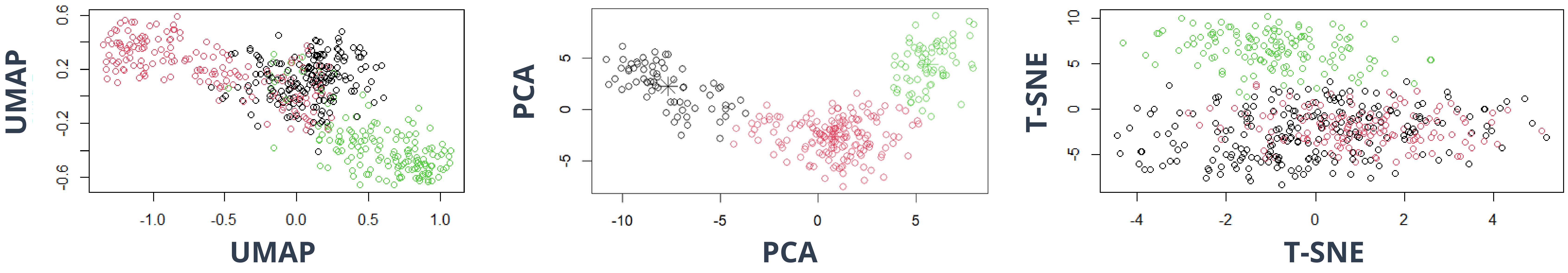
KNN + LOUVAIN

CLUSTERING: FIVE ORIGINAL SIMULATED GROUPS



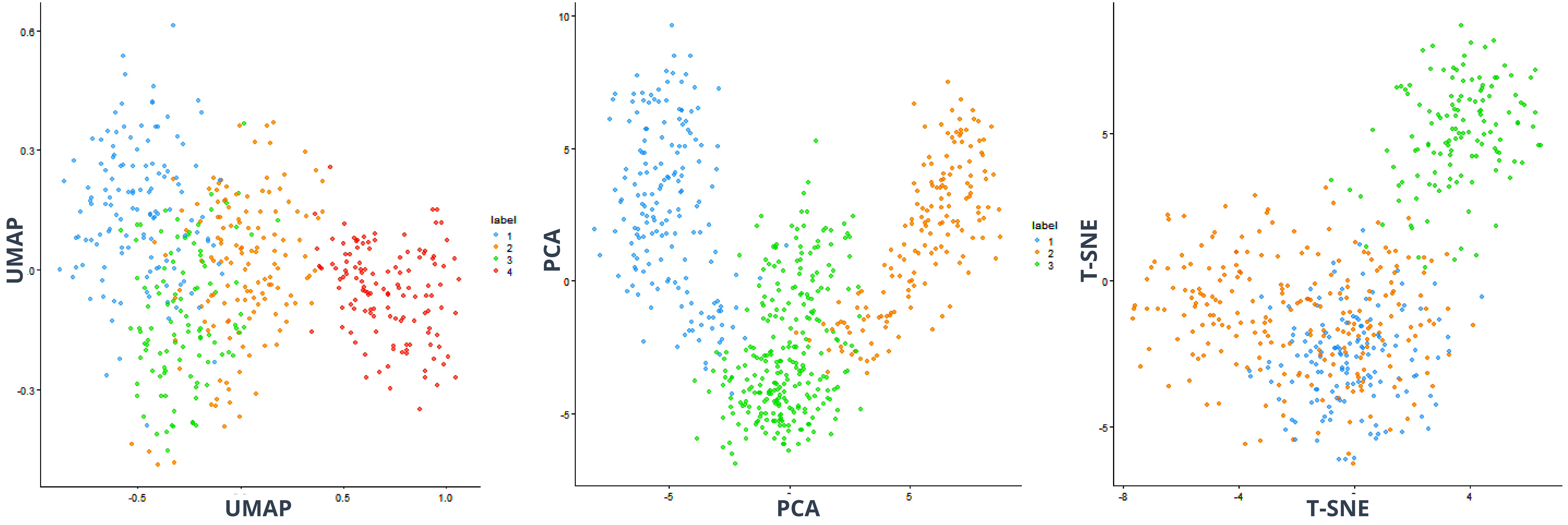
SNN + LOUVAIN

CLUSTERING: THREE ORIGINAL SIMULATED TRAJECTORIES



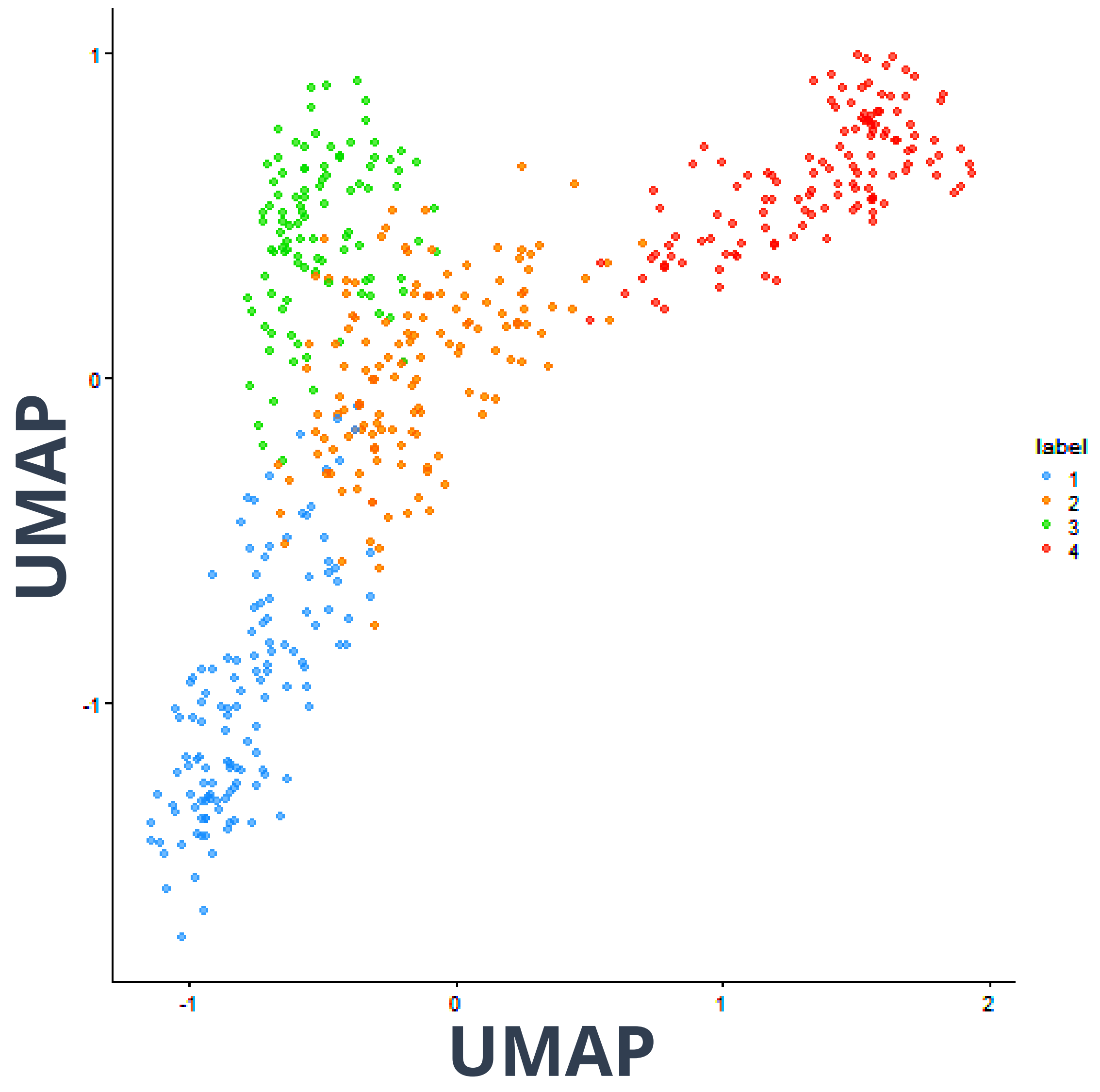
K-means

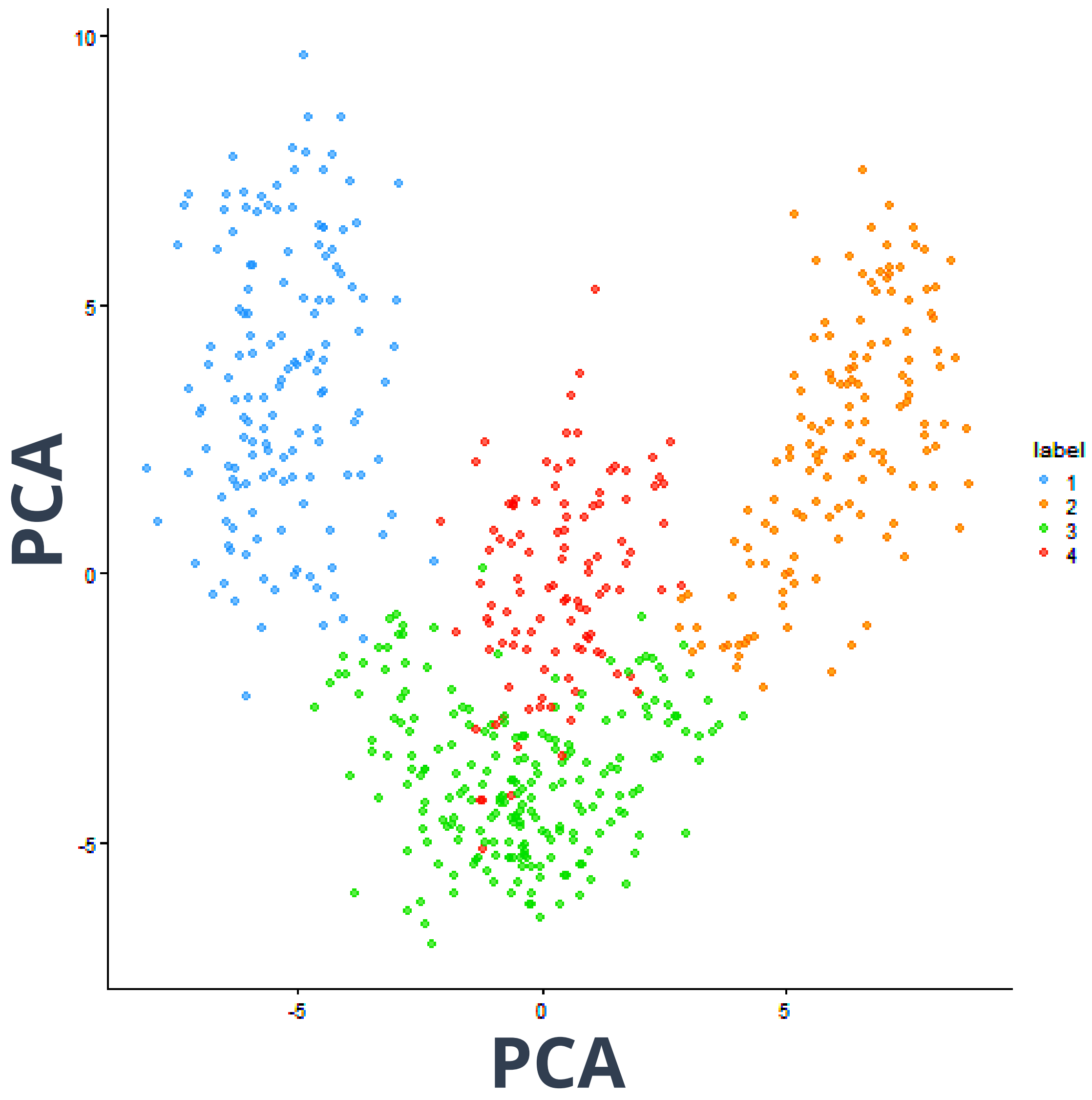
CLUSTERING: THREE ORIGINAL SIMULATED TRAJECTORIES

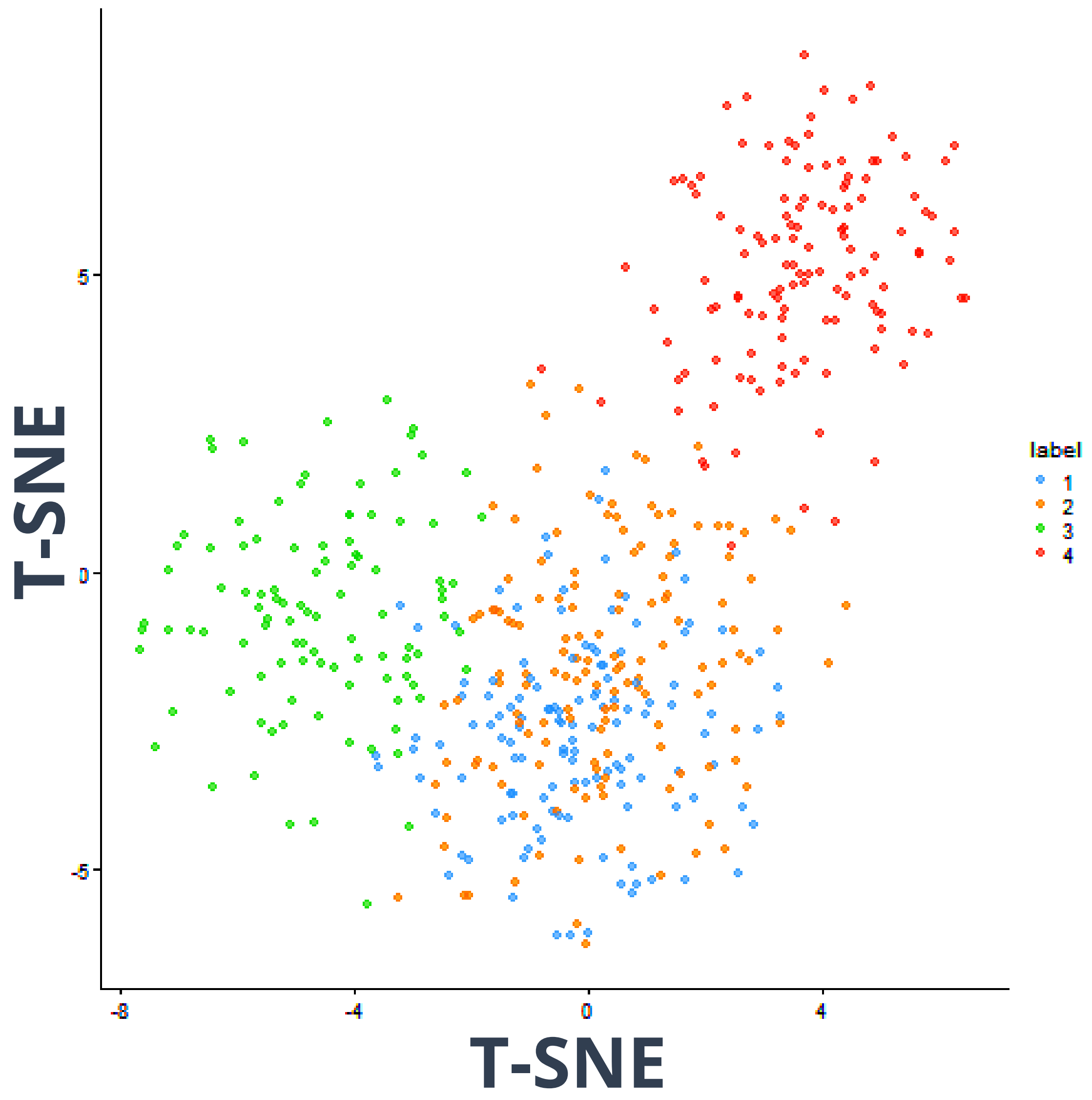


KNN + LOUVAIN

CLUSTERING: THREE ORIGINAL SIMULATED TRAJECTORIES







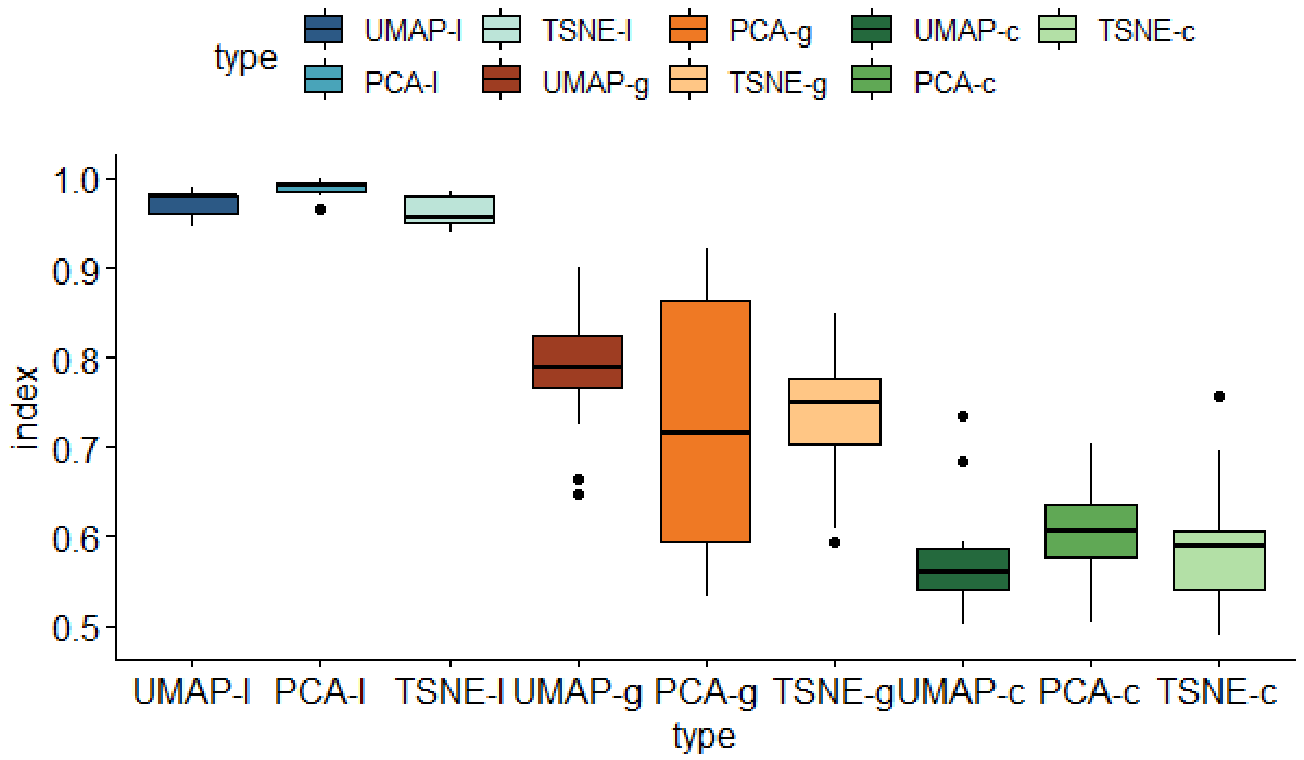
SNN + LOUVAIN

ASSESSMENT: JACCARD SCORE

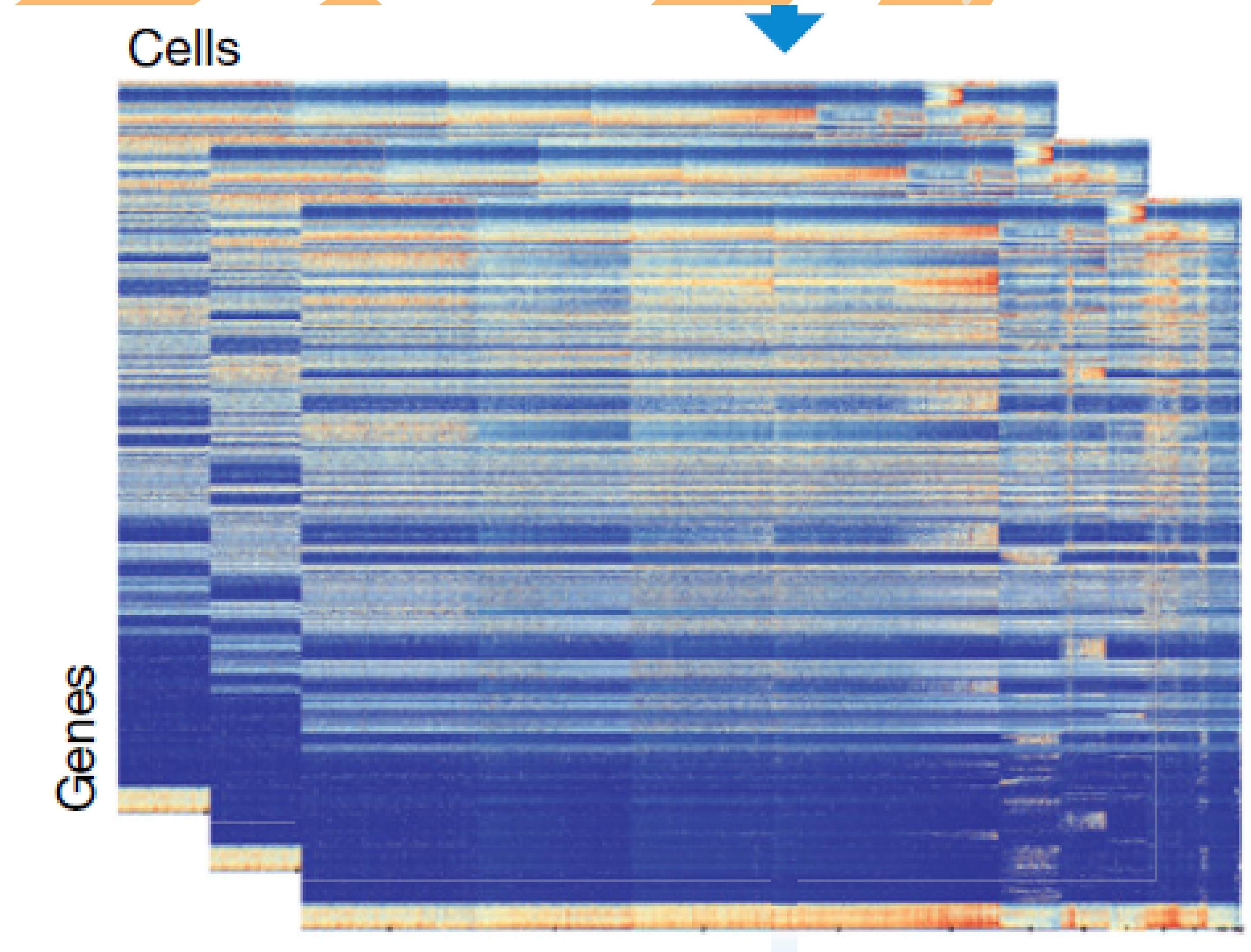
Type of simulation	Batches				Groups				Trajectories			
Clustering \Reduction	*UMAP*	*PCA*	T-SNE	Median	*UMAP*	*PCA*	T-SNE	Median	*UMAP*	*PCA*	T-SNE	Median
K-means	0.977	0.991	0.970	0.985	0.570	0.421	0.719	0.578	0.593	0.443	0.577	0.551
KNN - Louvain	0.974	0.990	0.961	0.980	0.787	0.729	0.736	0.765	0.570	0.605	0.589	0.582
SNN - Louvain	0.974	0.989	0.968	0.980	0.789	0.889	0.732	0.807	0.524	0.488	0.512	0.498
Median	0.980	0.99	0.966		0.764	0.714	0.738		0.553	0.493	0.551	

ASSESSMENT: BOXPLOTS OF JACCARD'S SCORE

Type of simulation	Batches				Groups				Trajectories			
Clustering \Reduction	*UMAP*	*PCA*	T-SNE	Median	*UMAP*	*PCA*	T-SNE	Median	*UMAP*	*PCA*	T-SNE	Median
K-means	0.977	0.991	0.970	0.985	0.570	0.421	0.719	0.578	0.593	0.443	0.577	0.551
KNN - Louvain	0.974	0.990	0.961	0.980	0.787	0.729	0.736	0.765	0.570	0.605	0.589	0.582
SNN - Louvain	0.974	0.989	0.968	0.980	0.789	0.889	0.732	0.807	0.524	0.488	0.512	0.498
Median	0.980	0.99	0.966		0.764	0.714	0.738		0.553	0.493	0.551	



KNN + LOUVAIN



DISCUSSION

- UMAP makes the best representations in low dimensions when the simulated data are closer to a real experiment.
- It is valid to use different clustering algorithms depending on prior information available to correctly use the dimensionality reduction of UMAP.
- Obtaining hyperparameters is crucial for the dimensionality reduction methods and also to clustering algorithms.

REFERENCES

Becht, E., L. McInnes, J. Healy, and C. Dutertre (2019). Dimensionality reduction for visualizing single-cell data using umap. Nature Biotechnology@(1), 38–44.

Hedlund, E. and Q. Deng (2018). Single-cell rna sequencing: Technical advancements and biological applications. Molecular Aspects of Medicine (59), 36–46.

Lun, A., D. McCarthy, and J. Marioni (2016). A step-by-step workflow for low-level analysis of single-cell rna-seq data with bioconductor. F1000Res. 5.

McCarthy, D., K. Campbell, A. Lun, and Q. Willis (2017). Scater: pre-processing, quality control, normalisation and visualisation of single-cell rna-seq data in r. Bioinformatics@(1), 1179–1186.

McInnes, L., J. Healy, and J. Melville (2018). Umap: uniform manifold approximation and projection for dimension reduction. arXiv [Preprint].

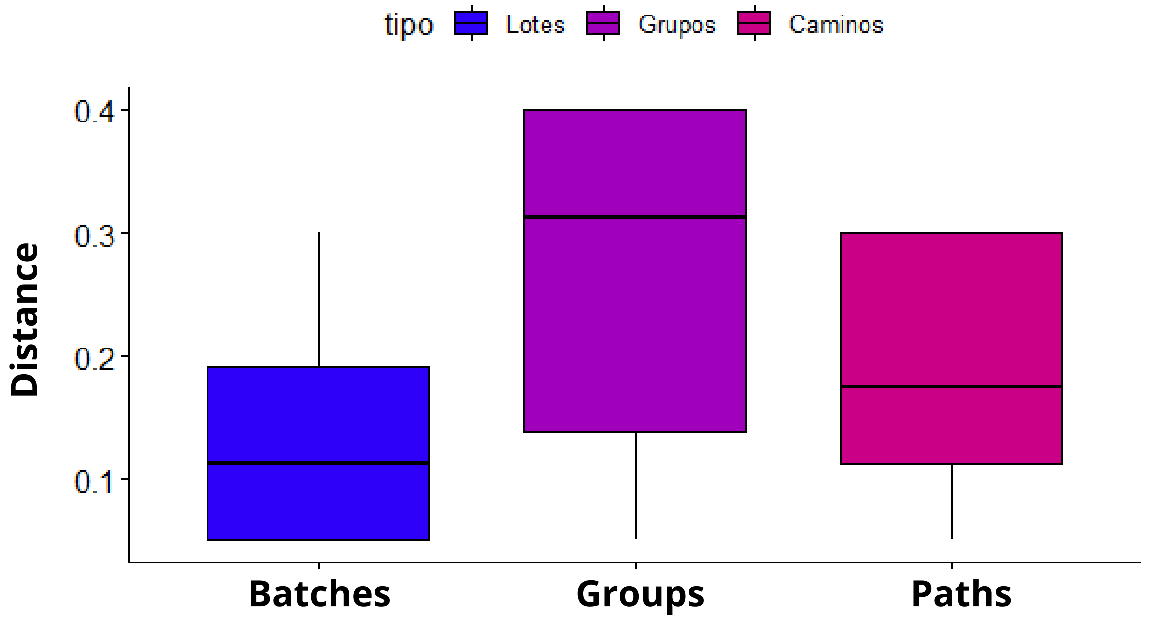
Wagner, A., A. Regev, and N. Yosef (2016). Revealing the vectors of cellular identity with single-cell genomics. Nat. Biotechnol. (34), 1145–1160.

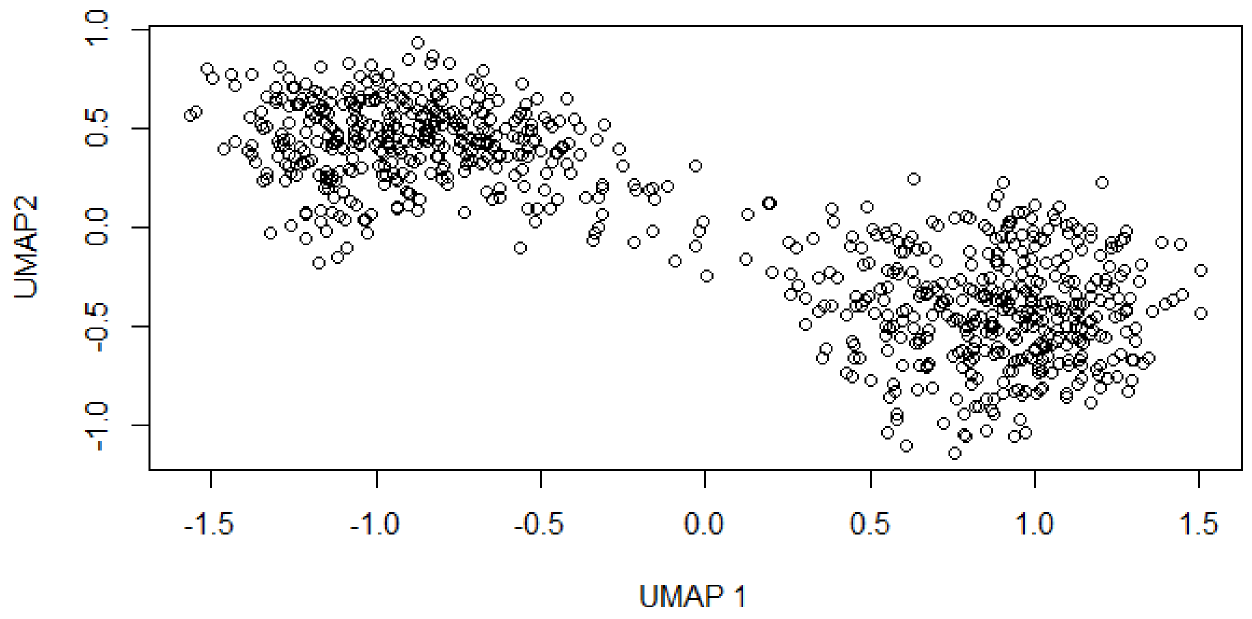
Xiang, R., W. Wang, L. Yang, S. Wang, C. Xu, and X. Chen (2021). A comparison for dimensionality reduction methods of single-cell rna-seq data. Front. Genet. 12.

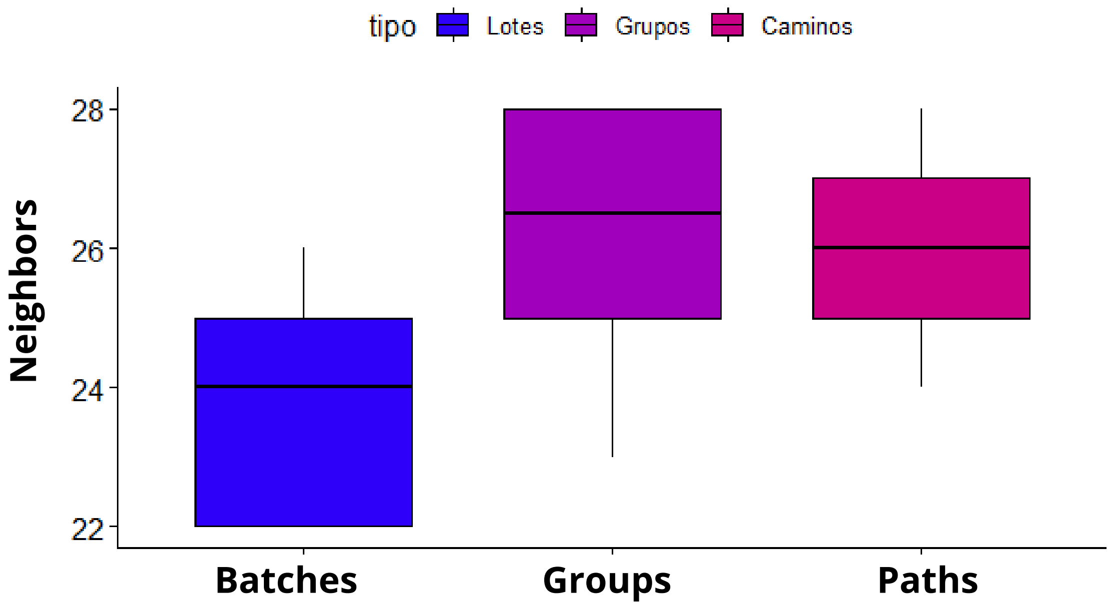
Zappia, L., B. Phipson, and A. Oshlack (2017). Splatter: simulation of single-cell rna sequencing data. Genome Biology.

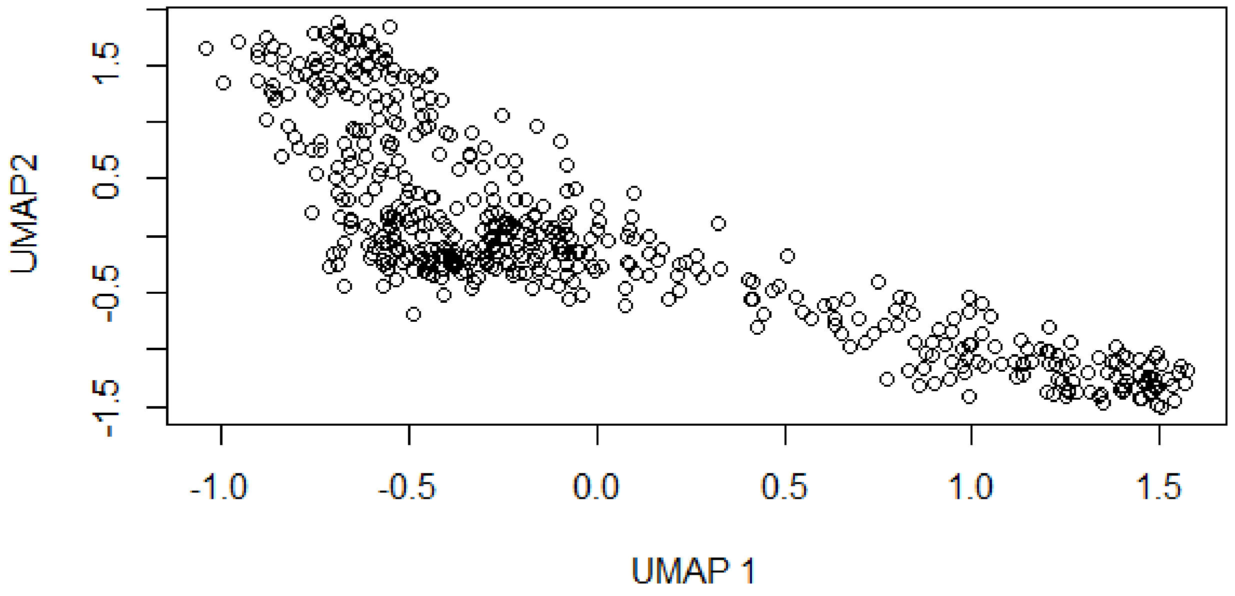
¡THANKS!

SIMULATIONS AND DIMENSIONALITY REDUCTION









UMAP

Algorithm 1 UMAP algorithm

function UMAP(X, n, d, min-dist, n-epochs)

 # Construct the relevant weighted graph

for all $x \in X$ **do**

fs-set$[x]$ ← LocalFuzzySimplicialSet(X, x, n)

top-rep ← $\bigcup_{x \in X}$ fs-set$[x]$ *# We recommend the probabilistic t-conorm*

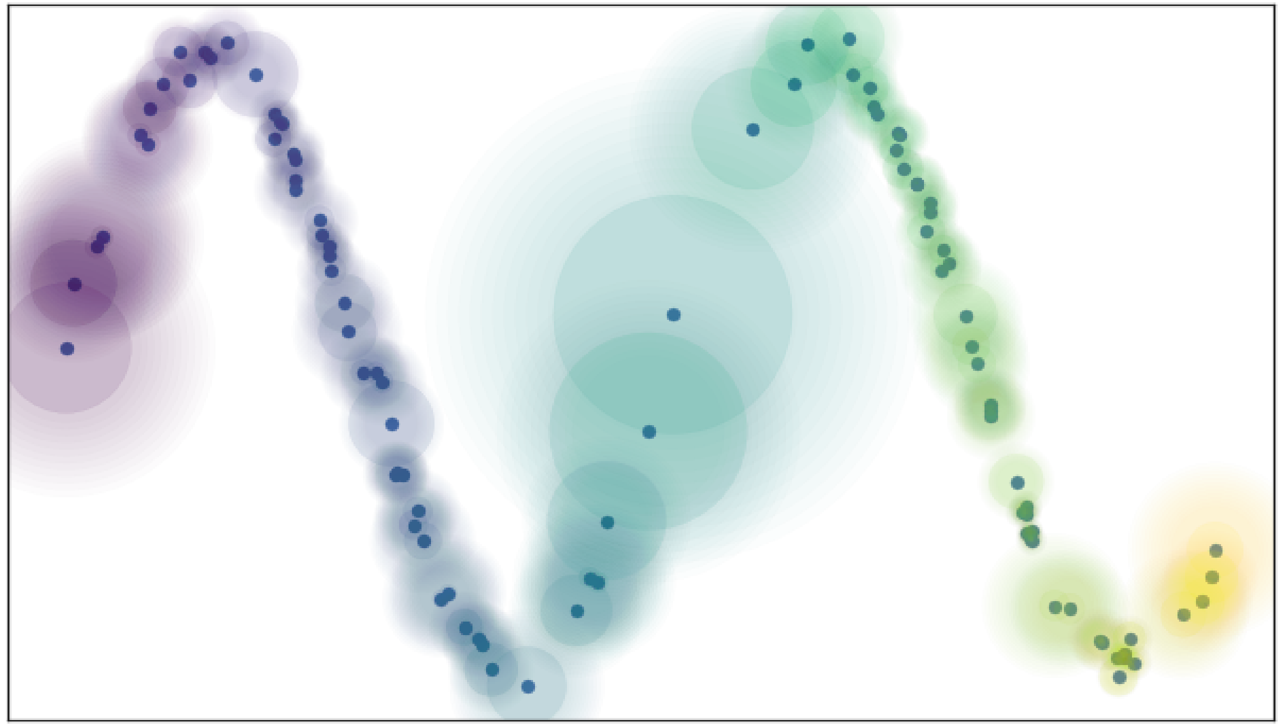
 # Perform optimization of the graph layout

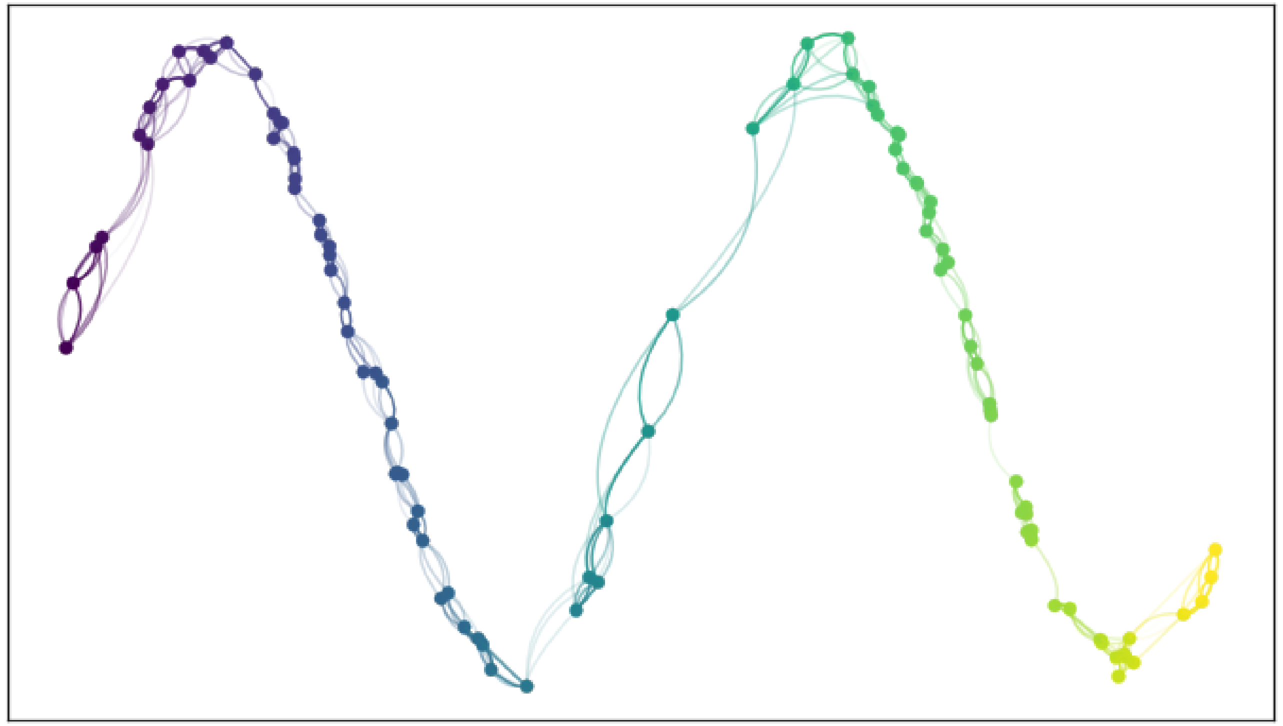
Y ← SpectralEmbedding(top-rep, d)

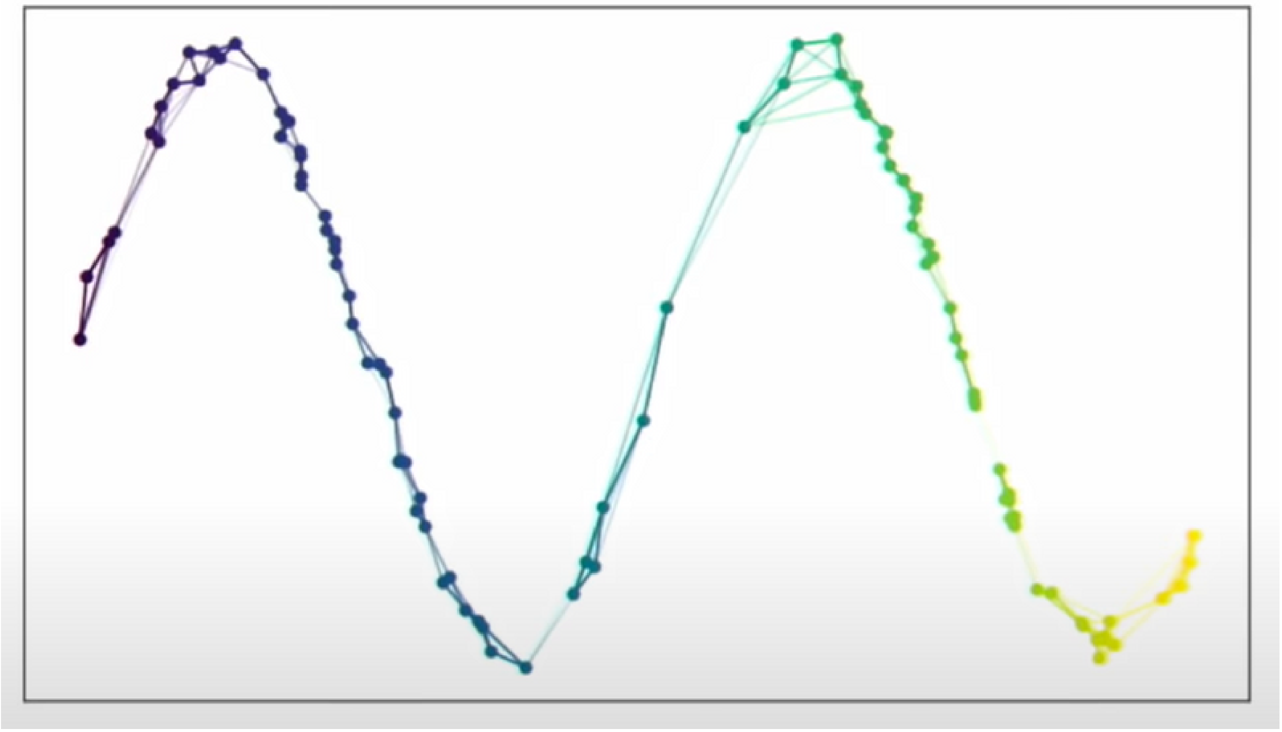
Y ← OptimizeEmbedding(top-rep, Y, min-dist, n-epochs)

return Y

UMAP CONSTRUCTION







PCA AND T-SNE

Algorithm 1: Principal component analysis

Input : X : Data matrix $X \in \mathbb{R}^{m \times n}$

d : Desired number of dimension $d < n$

Output: $\tilde{X}$: Data matrix $\tilde{X} \in \mathbb{R}^{m \times d}$

Center X, i.e., for each column in X substract the column mean

Compute scatter matrix $S = X \cdot X^T$

Compute eingen decomposition $S = V \cdot \Lambda \cdot V^T$

Sort the eigenvalues in Λ from largest to smallest such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$

Sort the eigenvectors in V following the order of the sorted eigenvalues

Compute the new features $\tilde{x}_{i,j} = X_i \cdot V_j^T$ where V_j^T = eigenvectors for $j = 1 \ldots d$

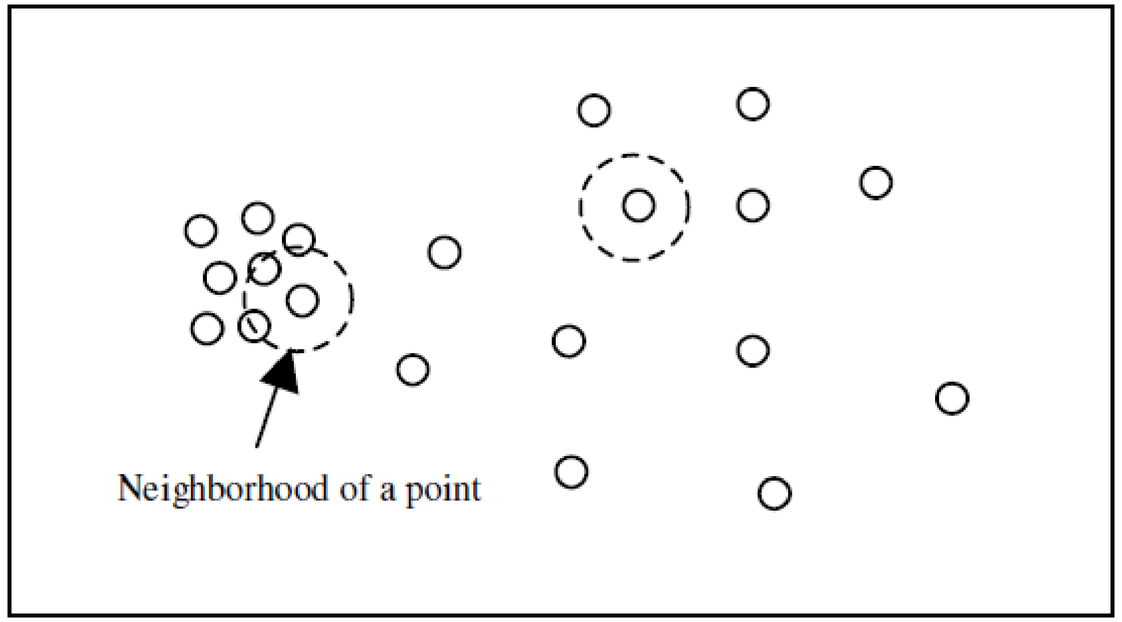
Return the transformed data matrix $\tilde{X}$

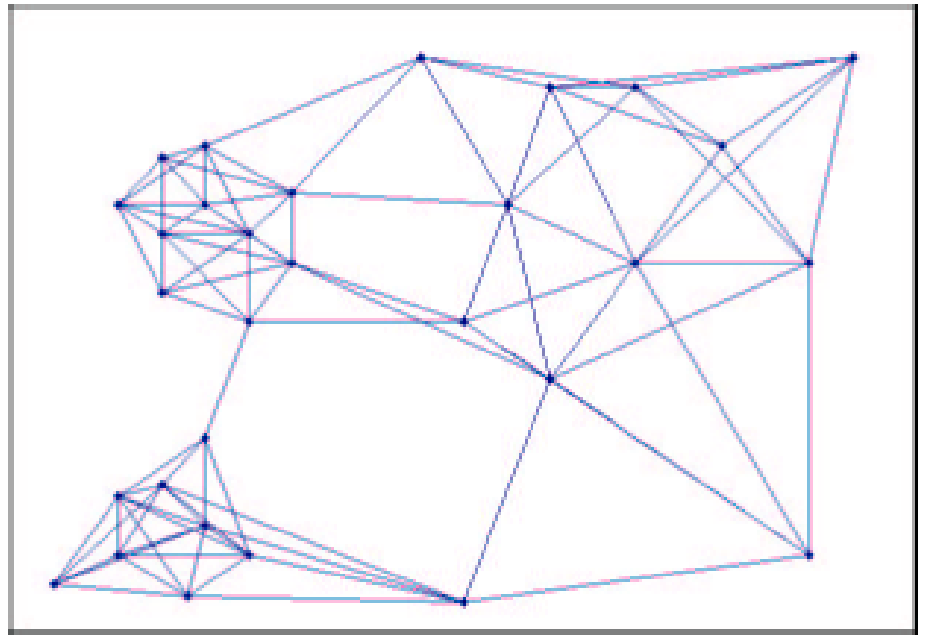
Algorithm 1: Simple version of t-Distributed Stochastic Neighbor Embedding.

Data: data set $X = \{x_1, x_2, \ldots, x_n\}$,
cost function parameters: perplexity *Perp*,
optimization parameters: number of iterations T, learning rate η, momentum $\alpha(t)$.
Result: low-dimensional data representation $\mathcal{Y}^{(T)} = \{y_1, y_2, \ldots, y_n\}$.

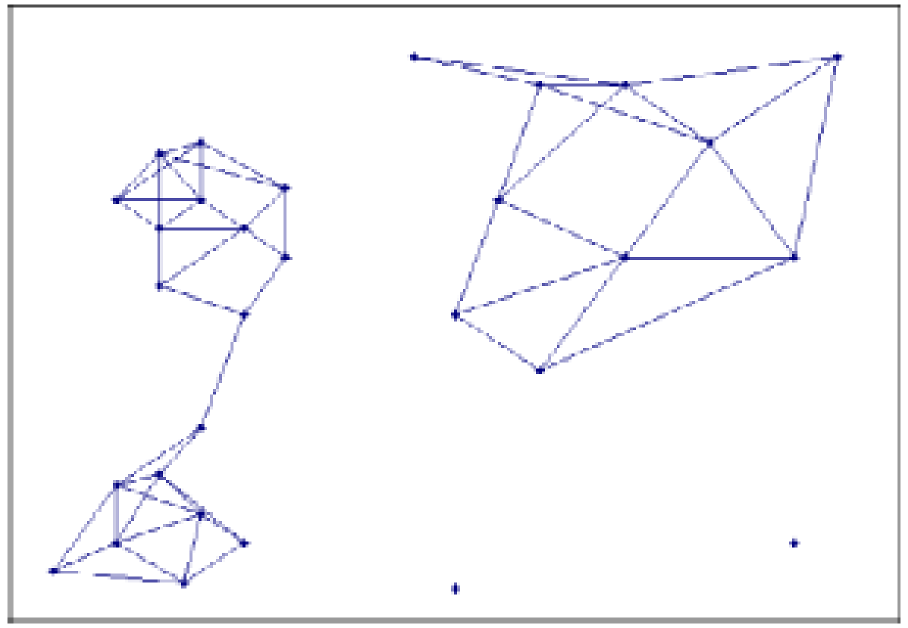
begin
compute pairwise affinities $p_{j|i}$ with perplexity *Perp* (using Equation 1)
set $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$
sample initial solution $\mathcal{Y}^{(0)} = \{y_1, y_2, \ldots, y_n\}$ from $\mathcal{N}(0, 10^{-4} I)$
for *t=1* **to** T **do**
compute low-dimensional affinities q_{ij} (using Equation 4)
compute gradient $\frac{\delta C}{\delta \mathcal{Y}}$ (using Equation 5)
set $\mathcal{Y}^{(t)} = \mathcal{Y}^{(t-1)} + \eta \frac{\delta C}{\delta \mathcal{Y}} + \alpha(t) \left(\mathcal{Y}^{(t-1)} - \mathcal{Y}^{(t-2)} \right)$
end
end

KNN/SNN GRAPH CONSTRUCTION





(a) Near Neighbor Graph.



(b) Unweighted Shared Nearest Neighbor.