# Binarized single cell RNA sequencing data clustering

## The impact of binarized scRNA-seq data on clustering through community detection algorithms

**Jurriën Theunisz**

**Supervisor(s): Marcel Reinders, Gerard Bouland**

EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 25, 2023

Name of the student: Jurriën Theunisz
Final project course: CSE3000 Research Project
Thesis committee: Marcel Reinders, Gerard Bouland, Bart Gerritsen

## Abstract

**Single-cell RNA sequencing data clustering is a valuable technique for demonstrating cell-to-cell heterogeneity and revealing cell dynamics within and amongst groups. Large up-scaling of scRNA-seq datasets in recent years pose computational challenges for existing state-of-the-art clustering techniques. A possible solution to tackle these challenges is to binarize the scRNA-seq data and perform clustering using optimized binary methods. Using a binary clustering pipeline we demonstrate that binary clustering solutions resemble conventional clustering solutions for large clusters, but show less resemblance for smaller clusters. We also show that the Leiden community detection algorithm can achieve higher cluster quality compared to the Louvain algorithm for the binarized data.**


## 1 Introduction

Single-cell RNA sequencing (scRNA-seq) technology has become an important method for deciphering the heterogeneity and complexity of RNA expressions within individual cells. scRNA-seq also reveals the composition of different cell types and functions within highly organized tissues, organs and organisms [1]. A major technique for getting valuable insight from the scRNA-seq data is the clustering of cells based on their gene expression levels. Nowadays, a single scRNA-seq dataset can easily contain over a million cells [2]. This large up-scaling of scRNA-seq datasets allows for more expressive clustering results, but does require clustering techniques to adapt to this data surge by running in a more time- and memory efficient manner.

The primary bottleneck in the clustering pipeline for large scRNA-seq datasets using state-of-the-art techniques is the calculation of the cell distance matrix. This step of the algorithm compares the gene expression data of every cell with every other cell and hence runs in quadratic time, scaling with the number of cells in the input dataset. Improvements to the time-efficiency of this step of the algorithm therefore result in a better runtime of the full pipeline and consequently enable processing larger datasets with the same amount of computational resources. Current approaches to reduce the runtime of clustering include methods such as principal component analysis (PCA), but these tend to result in a drop of clustering quality [3].

In previous research, Bouland et al. has proposed that it may be possible to run the clustering algorithm on a binarized representation of single cell RNA-seq data without greatly altering clustering results [4]. Using such technique would allow cell data to be stored in a binary format and comparisons to use a binary distance metric. This could lead to greatly improved time and memory efficiency of the clustering pipeline. This research is, however, mostly theoretical and requires more experimental evidence to prove it's value in real-world applications.

Here, we adapt the state-of-the-art scRNA-seq clustering pipeline to binarized input data in an attempt to minimize the changes in the output clusters generated through community detection and therefore maintain clustering quality. We pose the question; What community detection algorithm results in clusters most similar to current state-of-the-art clustering methods when applied to binarized scRNA-seq data?

If it can be shown that binary input data can be used without greatly altering the output clusters, it logically follows that the clustering pipeline can be greatly enhanced by using optimized binary comparison techniques. This research will not attempt to optimize the processing speed of the pipeline itself, but by showing that quality is largely maintained using a binarized representation of single cell RNA-seq data, it may open the door for future research to attempt to do so.

## 2 Methodology

To answer the research question posed in this paper we developed a binary clustering pipeline. The quality of this pipeline was evaluated by comparing the output clusters against clusters resulting from a conventional state-of-the-art clustering pipeline.

### 2.1 Binary clustering pipeline

Conventional state-of-the-art scRNA-seq clustering techniques usually consist of a five-step workflow [5]. This workflow is shown in figure 1 and is summarized as:

1. Read scRNA-seq data into an expression matrix
2. Convert expression matrix into component matrix using principal component analysis (PCA)
3. Find the distance matrix by calculating all cell to cell pairwise distances
4. Find the k-nearest-neighbour graph (KNN-graph) for this distance matrix
5. Find well-connected communities in the KNN-graph using a community detection algorithm and assign a cluster label to all cells per community

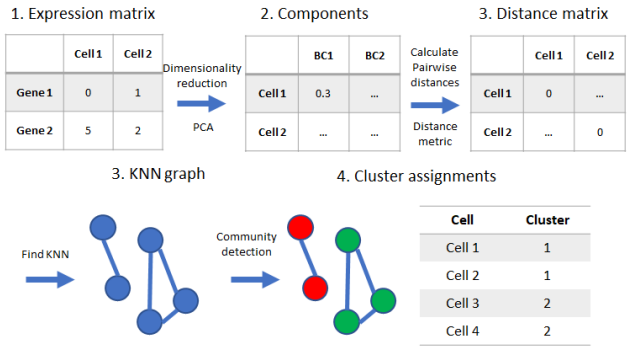



Figure 1: Conventional clustering workflow

The binary pipeline built for this research follows the same workflow as the conventional workflow, where the steps that are specific to non-binary data are replaced by a binary method that is most equivalent. Steps (4) and (5) of KNN-graph generation and community detection operate on the

non-binary distance matrix, hence these components stay the same. Step (1) of reading non-binary input data into an expression matrix is replaced by an equivalent step of reading the binary input data into a binary matrix. There are multiple options when replacing steps (2) and (3). The first option is to replace step (2) by a binary equivalent of PCA, turning the binary matrix into a component matrix. This would leave distance matrix calculation in step (3) untouched. A second option is to remove step (2) from the workflow and directly calculate the distance matrix from the binary matrix by replacing the pairwise distance metrics with their binary equivalents in step (3).

It is exactly in step (2) and (3) that the time and memory efficiency of the binary pipeline can be greatly enhanced by taking the second replacement option. The binary matrix created in step (1) of the binary workflow has greatly improved memory efficiency compared to the expression matrix and eliminating the need for PCA also removes the memory usage for the component matrix in step (2). The binary representations can also be compared more efficiently using binary distance metrics in step (3), as they can be optimized to compare multiple bits at a time by using efficient implementations for calculating the hamming weight [6].

The binary clustering pipeline used in this research implements the proposed four-step workflow by eliminating step (2) of the conventional pipeline and adapting step (3) by comparing binary data using binary distance metrics. A graphical summary of the binary clustering workflow used in this research is displayed below in figure 2.

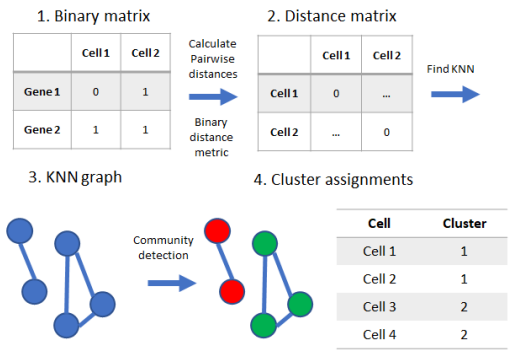



Figure 2: Binary clustering workflow

The binary clustering pipeline described in this section is entirely built in C++ as the programming language most well-known for its' efficiency and optimizability. The binary matrix, distance matrix as well as the KNN-graph steps are self-implemented. Community detection makes use of the IGraph library, because it is the most widely used graphing library that contains implementations of both the Louvain and Leiden algorithms and has an interface for C, C++, R and Python. Multiple binary distance metrics and clustering evaluation metrics have also been self-implemented. Finally, the pipeline has been made compatible with R4 and R5, being the languages most commonly used by researchers when working with statistics.

## 2.2 Experimental setup configuration

Clustering outcome of the binary clustering pipeline is largely decided by how the KNN-graph is connected and how the community detection algorithm identifies the different communities in the KNN-graph. Different cluster labels are assigned to every community detected in the KNN-graph, resulting in the clustering solutions that are evaluated against the ground truth.

### KNN-graph configuration

The effectiveness of the KNN-graph depends on the value of K, a parameter describing the number of nearest neighbours for every node in the graph, and the distance metric that is used to find these nearest neighbours. The K-value decides the degree of connectedness of the graph, where either too low or too high connectedness could lead to poor cluster quality. It is therefore essential that the value of K and the distance metric are selected with care when running clustering experiments.

Research has shown that the optimal choice of distance metric and the value of K are related, but mainly depend on properties of the scRNA-seq dataset operated on and whether PCA has been applied or not [7]. It is not guaranteed that the suggested distance metrics and K-value combinations that apply to conventional clustering methods also provide similar results when applied to binary clustering. Binary datasets have different properties from non-binary datasets and not every continuous distance metric has an equivalent binary distance metric available.

The objective of this research is to find binary clustering solutions that perform well relative to the state-of-the-art conventional methods and to compare the clustering quality of the community detection algorithms. It is therefore logical to pick a single binary distance metric and a K-value that are likely to provide the best results for binary datasets. Binary scRNA-seq datasets follow a discrete value distribution and are usually sparse in nature [4]. The binary clustering pipeline used to run experiments for this research does not apply PCA. Therefore, according to the flowchart for recommended metrics and neighbourhood sizes (k) given specific structural properties of an scRNA-seq dataset, provided in the research performed by Watson et al., the Cosine distance metric and Phi distance metric are both great candidates when applied to binarized scRNA-seq datasets [7]. Both the Cosine and Phi distance metrics have a binary equivalent and both provide good results for datasets that contain rare cell populations as well as those without [7] [8]. The Cosine distance metric appears to perform better with sparse datasets overall and also seems to perform well for low values of K as well as high K-values, providing more flexibility than the Phi distance metric.

The binary Cosine distance metric was therefore selected as the metric of choice for the experiments performed in this research. It is also known as the Ochiai dissimilarity metric and is the inverse of the Ochiai similarity metric [9]. It is defined as

$$D_o = 1 - S_o \tag{1}$$

where

$$S_o = \frac{X \cdot Y}{|X||Y|} \tag{2}$$

is the Ochiai similarity metric and $X$ and $Y$ are binary vector representations of the cell data.

The K-value used with the binary cosine distance metric is flexible (with recommended values of K=3,30,50,100) and is set to the K-value that was used to create the ground truth solutions [7]. Setting the K-value to the same K-value used for the ground truth guarantees that the KNN-graph is equally well connected while maintaining the memory usage of the KNN-graph.

**Community detection configuration**

Community detection is used to find the well-connected communities in the KNN-graph. Most conventional state-of-the-art clustering pipelines rely on the Louvain community detection algorithm for finding communities, but recent research has shown that the newly developed Leiden community detection algorithm may provide better clustering results [10]. It is these two community detection algorithms that our experimental setup deploys and were used to find the different communities in the KNN-graph and assign cluster values.

Both the Louvain and the Leiden community detection algorithms rely on modularity. Modularity is one of the most effective community detection methods nowadays. This method attempts to maximise the difference between the actual number of edges and expected number of edges in a community [10]. Modularity ($Q$) is calculated by

$$Q = \frac{1}{2m} \sum_c a_c - \gamma \frac{k_c^2}{2m} \tag{3}$$

where $a_c$ denotes the actual number of edges in community $c$, $k_c$ denotes the expected number of edges in community $c$ and $m$ is the total number of edges in the graph. $\gamma$ is a resolution parameter that influences the number of communities that are detected [10]. Higher values of $\gamma$ lead to more communities, whereas lower values lead to fewer communities.

The resolution parameter $\gamma$ is the most important optimization variable that was tweaked in the experiments conducted in this research. Both community detection algorithms were initialized as undirected graphs and the edges were assigned weights that further improve the performance of the modularity function. The edge weights were set to the inverse of the distance values on the edges of the KNN-graph and were calculated using the Ochiai similarity function defined in equation (2).

The Leiden algorithm furthermore has the option of configuring the parameter $\beta$, which indicates the degree of randomness. It also allows to set the maximum number of iterations $n$ that the algorithm will perform, which can be used to force the amount of communities detected for a specific value of $\gamma$. The parameter $\beta$ was set to 0.01 for all experiments to allow a small degree of randomness. $n$ was set to the default value for the experiments in this research as optimization is performed on $\gamma$.

## 2.3 Clustering outcome evaluation

Binary scRNA-data clustering is still a new concept and a lot is still unknown about resulting clustering composition. To study the different clustering compositions, evaluation metrics that look at different clustering traits were picked. Evaluating different clustering traits also proved useful when comparing the behaviour and clustering quality of the Leiden and Louvain community detection algorithms.

The most commonly used metrics for evaluating clustering similarity are the adjusted random index (ARI) and the normalized mutual index (NMI) [11]. Recently, the pairwise set index (PSI) has also become increasingly popular for evaluating clustering quality in a supervised manner [7]. We have selected these 3 evaluation metrics to compare the experimental clustering results against conventional clustering solutions for the same datasets.

ARI, NMI and PSI work fundamentally different and fall in different categories of cluster validity indexes. They each provide insight into different clustering characteristics. ARI is a pair-counting measure, whereas NMI is a information theoretic measure and PSI is a set matching measure [12].

**Pair-counting measures**

Pair-counting measures count the pairs of points that fall in the same clusters and those that fall in different clusters [12]. ARI is calculated as

$$ARI = \frac{RI - E(RI)}{1 - E(RI)} \tag{4}$$

where $RI$ denotes the random index and is defined as

$$RI = \frac{a + d}{N(N-1)/2} \tag{5}$$

and $E(RI)$ denotes the expected value of $RI$. $a$ represents the number of pairs that are in the same clusters and $d$ represents the number of pairs that are in different clusters. $N$ denotes the total number of points in a clustering.

**Information theoretic measures**

Information theoretic measures use the concept of entropy to compare two clusterings [12]. NMI is given by

$$NMI = \frac{MI(P, G)}{(H(P) + H(G))/2} \tag{6}$$

where $P$ and $G$ denote the two clusterings being compared, $H(P)$ and $H(G)$ give the entropy of clusterings $P$ and $G$ respectively and $MI(P, G)$ gives the mutual information score of the clusterings. The entropy of clustering $P$ with $K$ clusters is defined as

$$H = -\sum_{i=1}^{K} \frac{|P_i|}{N} \log \frac{|P_i|}{N} \tag{7}$$

where $N$ is the total number of points in a clustering and $|P_i|$ denotes the number of points in cluster $P_i$. The mutual information score $MI(P, G)$ of two clusterings $P$ and $G$ is given by

$$MI = \sum_{i=1}^{K} \sum_{i=1}^{K'} \frac{n_{ij}}{N} \log \frac{N n_{ij}}{|P_i||G_j|} \tag{8}$$

where $K$ and $K'$ denote the number of clusters in clusterings $P$ and $G$, $P_i$ and $G_j$ are two of those clusters, $|P_i|$ and $|G_j|$ are their respective sizes and $n_{ij}$ is the number of shared points between clusters $P_i$ and $G_j$.

**Set matching measures**

Set matching measures are based on matching entire clusters. Similar clusters are first aligned either by pairing or matching. Cluster pairs are then evaluated using set similarity metrics [12]. PSI is calculated as

$$PSI = \begin{cases} \frac{S - E(S)}{max(K,K') - E(S)} & \text{if } S \geq E(S) \\ 0 & \text{if } S < E(S) \\ 1 & \text{if } K = K' = 1 \end{cases} \tag{9}$$

where

$$S = \sum_{i=1}^{min(K,K')} \frac{n_{ij}}{max(|P_i|, |G_j|)} \tag{10}$$

and $K$ and $K'$ denote the number of clusters in clusterings $P$ and $G$, $P_i$ and $G_j$ are two of those clusters, $|P_i|$ and $|G_j|$ are their respective sizes and $n_{ij}$ is the number of shared points between clusters $P_i$ and $G_j$. $E(S)$ denotes the expected value of $S$.

## 3 Experiments

The experiments in this research were performed on two different scRNA-seq datasets to sketch a picture of how the Leiden and Louvain community detection algorithms impact the clustering results from a binary clustering pipeline relative to a ground truth solution resulting from a conventional clustering pipeline.

Each dataset had the normalized scRNA-seq data available and a ground truth solution was selected based on the dataset. The normalized data was binarized by setting every non-zero value to a binary 1 and every zero value to a binary 0.

The binarized data was used to run 1000 experiments on the experimental setup of our binary clustering pipeline for each community detection algorithm for increasing parameter $\gamma$. This resulted in 1000 clustering solutions for the Louvain algorithm and the Leiden algorithm respectively. These solutions were then evaluated against the ground truth for the ARI, NMI and PSI metrics to give a similarity score between 0% and 100% for each metric. This gave an overview of the behaviour of each community detection algorithm and narrowed down the $\gamma$-range for which each metric score peaked.

Exhaustive search was then performed to find the binary clustering solution with an optimal similarity score for each metric for each community detection algorithm over the narrowed down $\gamma$-ranges. The optimal similarity scores for each metric were used to quantify which community detection algorithm resulted in clusters most similar to the current state-of-the-art conventional pipeline. The optimal clustering solutions were further broken down into cluster partitions and visualized. The clustering solutions with interesting partitioning were also visualized in uniform manifold approximation and projection (UMAP) space using the Python UMAP library described in the paper by McInnes et al. [13]. These cluster visualizations were then empirically evaluated to argue which community detection algorithm seems to perform most similar to the current state-of-the-art for the corresponding ground truths.

The two datasets that were used in the experiments, including their respective ground truth solutions, are detailed below.

**Alzheimer's dataset**

The alzheimer's dataset contains scRNA-seq data from human brain cells. It is based on the alzheimer's dataset used in prior scRNA-seq research conducted by Grubman et al. where unidentified and hybrid cell types have been removed [14]. The dataset contains 11.884 cell data points.

The ground truth for this dataset was created using a conventional clustering pipeline by visual inspection of the resulting UMAPs. The resulting clustering solution was found using the jaccard distance metric and a k-value of 10. It is visualized in a UMAP in figure 7a.

**Xenopus' tail dataset**

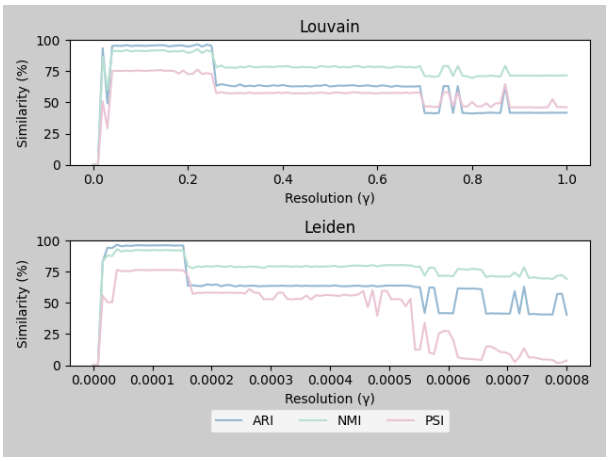
The xenopus' tail dataset contains scRNA-seq data from a tail of xenopus laevis tadpoles. The dataset was used in research conducted by Aztekin et al. and is openly accessible [15]. It contains 13.199 cell data points.

The ground truth for this dataset was found using a conventional clustering pipeline by optimizing on silhouette score using the cosine distance metric and a k-value of 10. The resulting clustering solution had a silhouette score of 0.809 and is visualized in a UMAP in figure 7b.

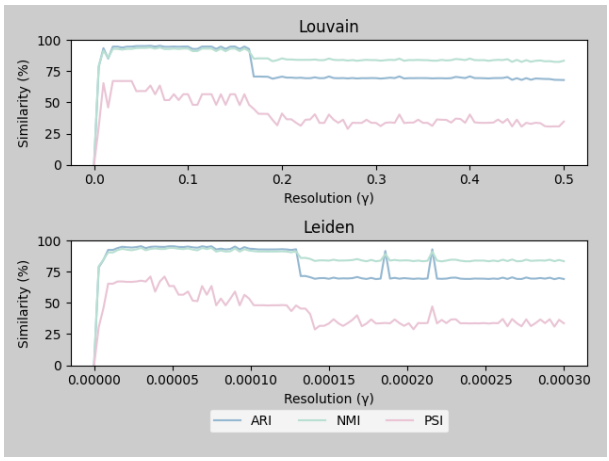
## 4 Results

**Similarity scores for all evaluation metrics peak for low $\gamma$ and then suddenly drop and gradually decline as $\gamma$ grows**

To get a general idea of the behaviour of the Louvain and Leiden community detection algorithms when applied to binarized scRNA-seq data we collected 1000 binary clustering solutions for each dataset for each algorithm. The solutions were collected for increasing parameter $\gamma$ and the ARI, NMI and PSI similarity scores were calculated relative to the ground truths. The similarity scores for each metric with regards to the ground truth peaked for low values of $\gamma$ after which they oscillated around the peak and then decreased as $\gamma$ increased (Figure 3). The same behaviour occurred for both community detection algorithms and both datasets. For the Alzheimer's dataset the similarity scores peaked for Louvain in the $\gamma$ range of [0, 0.27] and for Leiden in the range of [0, 0.00018] with approximate peak values of 96% (ARI), 93% (NMI) and 75% (PSI) (Figure 3a). In case of the Xenopus' tail dataset the peaking effect occurred for Louvain in the $\gamma$ range of [0, 0.18] and for Leiden in the range of [0, 0.00014] with approximate peak values of 95% (ARI), 94% (NMI) and 70% (PSI) (Figure 3b).




(a) Alzheimer's dataset




(b) Xenopus' tail dataset

Figure 3: Plot of the ARI, NMI and PSI similarity scores for binary clusterings resulting from the Louvain and Leiden community detection algorithms relative to the ground truths for increasing resolution ($\gamma$) for (a) the Alzheimer's dataset and (b) the Xenopus' tail dataset. The x-axis represents the resolution parameter $\gamma$, the y-axis represents the similarity score and the colors represent the different cluster evaluation metrics.

**Similarity scores for all evaluation metrics are highest when the number of clusters in the binary clustering solution and the ground truth is close**

The early peaking behaviour of the community detection algorithms was in line with the behaviour of the modularity function, defined in equation (3). By definition of the modularity function, low values of $\gamma$ result in a low number of clusters (n), where larger $\gamma$ values result in more clusters. The similarity scores were therefore expected to peak quickly, because the ground truths only contained few clusters and were then expected to drop gradually as the data points in the binary clustering solutions were divided over more and more clusters. The similarity scores relative to the ground truth (n=6) for the Alzheimer's dataset were highest when the binary clustering solutions contained 5 to 6 clusters with approximate peak values of 96% (ARI), 93% (NMI) and 75% (PSI) (Figure 4a). For the Xenopus' tail dataset the similarity scores peaked relative to the ground truth (n=8) when the binary clustering solutions contained 6 to 10 clusters with approximate peak values of 95% (ARI), 94% (NMI) and 81% (PSI) (Figure 4b). The centering of the similarity score peaks around the number of clusters in the ground truth is well-defined by the PSI evaluation metric as the PSI similarity score quickly decreases when the difference between the number of clusters in the binary clustering solution and the ground truth grows.

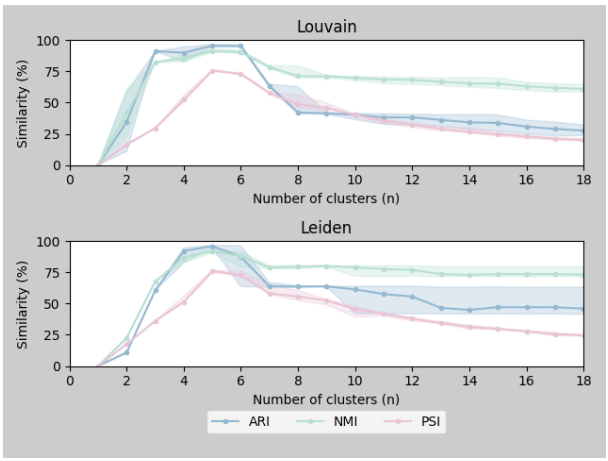
**Leiden community detection scores higher than Louvain community detection for all evaluation metrics**

To determine which community detection algorithm produces binary clustering solutions most similar to the conventional state-of-the-art, we have performed an exhaustive search to find the solutions with the optimal ARI, NMI and PSI scores for each community detection algorithm and each dataset. The optimal solutions for the Alzheimer's dataset resulted in similarity scores of 96.72% (ARI), 93.24% (NMI) and 76.58% (PSI) for the Louvain algorithm and in similarity scores of 96.84% (ARI), 93.42% (NMI) and 76.63% (PSI) for the Leiden algorithm (Figure 5a). For the Xenopus' tail dataset the optimal solutions resulted in similarity scores of 95.77% (ARI), 94.63% (NMI) and 71.82% (PSI) for the Louvain algorithm and in similarity scores of 95.80% (ARI), 94.79% (NMI) and 81.30% (PSI) for the Leiden algorithm (Figure 5b). In all cases the Leiden community detection algorithm scored higher than the Louvain community detection algorithm. The Leiden community detection algorithm, however does have a bias, as we set our $\beta$ parameter to 0.01. It is possible that the small amount of randomness resulted in clusters most similar to the ground truths when searching for the optimal solutions.

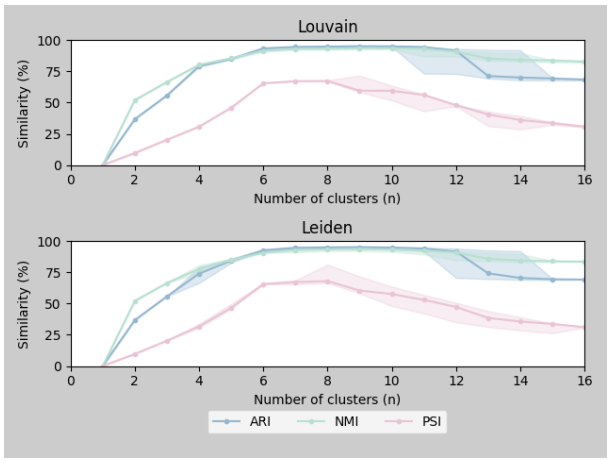
**Binary clustering shows great resemblance to conventional clustering for larger clusters, but less resemblance for smaller clusters**

To take an in-depth look at how binary clustering results relate to conventional clustering results we have analyzed the cluster compositions of the ground truth solution and the most similar binary clustering solutions for the ARI, NMI and PSI metrics.

The ground truth of the alzheimer's dataset and the optimal Leiden solution for the PSI metric contained 6 clusters each. The other optimal solutions only contained 5 clusters each. For the Alzheimer's dataset the largest cluster was nearly equal in size for all solutions, whereas the smaller clusters indicated more variation among the cluster sizes (Figure 6a). The high ARI score (>96%) and NMI score (>93%) proved that most of the data points were shared among clusters. The two largest clusters in the ground truth contained 9.603 out of 11.884 data points (80.1%) and therefore had more impact on the ARI and NMI metric scores. The clustering solutions with only 5 clusters have a maximum PSI score of 83.33% by definition. They were all missing the smallest cluster con-

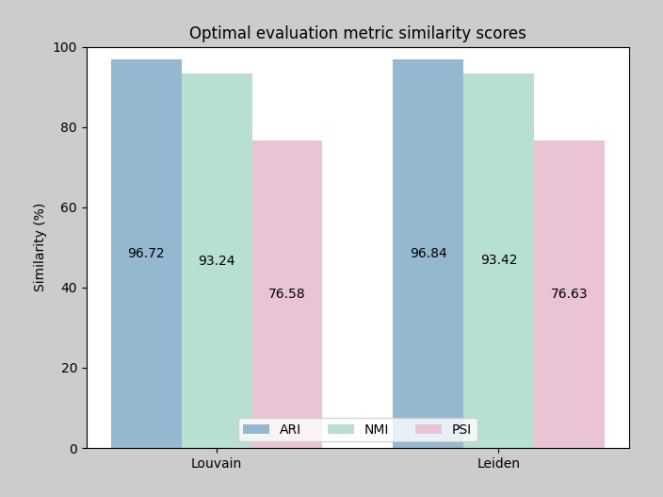



(a) Alzheimer's dataset




(b) Xenopus' tail dataset

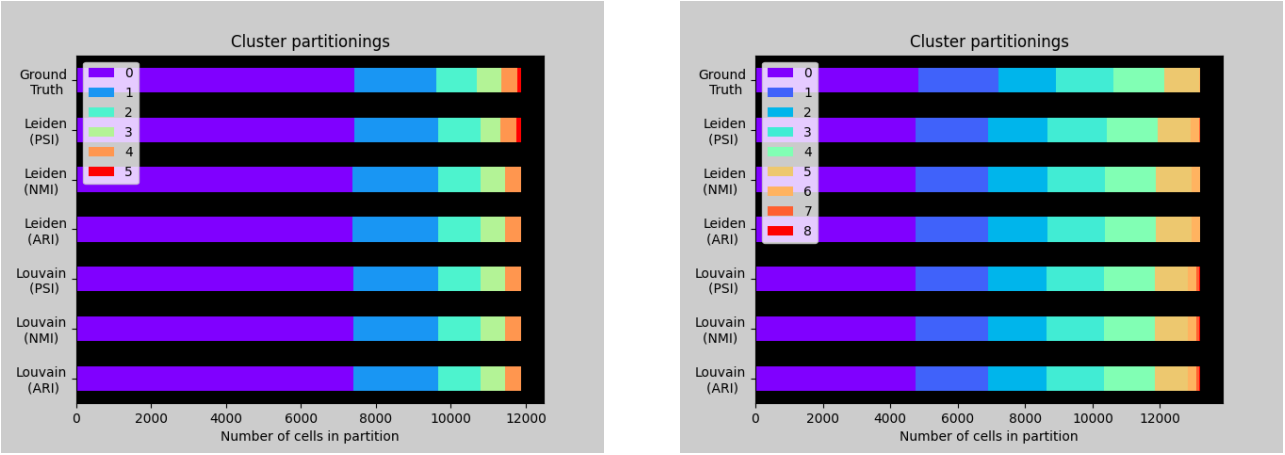
Figure 4: Plot of the mean, minimum and maximum ARI, NMI and PSI similarity scores by the number of clusters in the binary clustering solutions resulting from the Louvain and Leiden community detection algorithms relative to the ground truths for (a) the Alzheimer's dataset and (b) the Xenopus' tail dataset. The x-axis represents the number of clusters (n) in the binary clustering solution, the y-axis represents the similarity score and the colors represent the different cluster evaluation metrics.




(a) Alzheimer's dataset


(b) Xenopus' tail dataset

Figure 5: Histogram of the optimal ARI, NMI and PSI similarity scores of the binary clustering solutions resulting from the Louvain and Leiden community detection algorithms relative to the ground truths for (a) the Alzheimer's dataset and (b) the Xenopus' tail dataset. The x-axis represents the community detection algorithm, the y-axis represents the similarity score, the values inside the bars show the exact optimal similarity score and the colors represent the different cluster evaluation metrics.




(a) Alzheimer's dataset

(b) Xenopus' tail dataset

Figure 6: Stacked bar chart showing the cluster partitionings for the ground truth and binary clustering solutions with the optimal ARI, NMI and PSI metric scores for the Louvain and Leiden community detection algorithms for (a) the Alzheimer's dataset and (b) the Xenopus' tail dataset. The x-axis represents the number of datapoints in the partitioning, the y-axis represents the clustering solution and the colors represent the different clusters in the partitioning

firming that the impact of smaller clusters was much larger on the PSI metric score. The optimal binary clustering solution containing 6 clusters only had a PSI score of 76.63%. This indicated that there was little overlap between at least one of the small clusters. This was indeed the case as one of the smaller clusters had largely fused with one of the larger clusters in the binary clustering solution (Figure 7b) with respect to the ground truth (Figure 7a).

The ground truth of the The Xenopus' tail dataset consisted of 8 clusters and so did the optimal Leiden solution for the PSI metric. The optimal similarity scores for the Louvain algorithm resulted in the same solution for all metrics (ARI, NMI, PSI), containing 9 clusters. The optimal Leiden clustering solutions for the remaining metrics (ARI, NMI) were also the same solution and contained 7 clusters. There were 6 major clusters in all solutions that together contained more than 95% of the data points (Figure 6b). The minor clusters together, therefore, only contained less than 5% of all the data points. This imbalance supported the relatively high ARI and NMI scores and lower PSI scores that we found. Despite the optimal solution for the Leiden PSI metric containing an equal number of clusters as the ground truth we only found a PSI score of 81.3%. This indicated that the majority of data points in the minor clusters did not overlap between the binary solution and the ground truth. We verified that the smallest cluster in the ground truth (Figure 7c) was not detected in the binary clustering solution (Figure 7d), but a different minor cluster was detected instead.

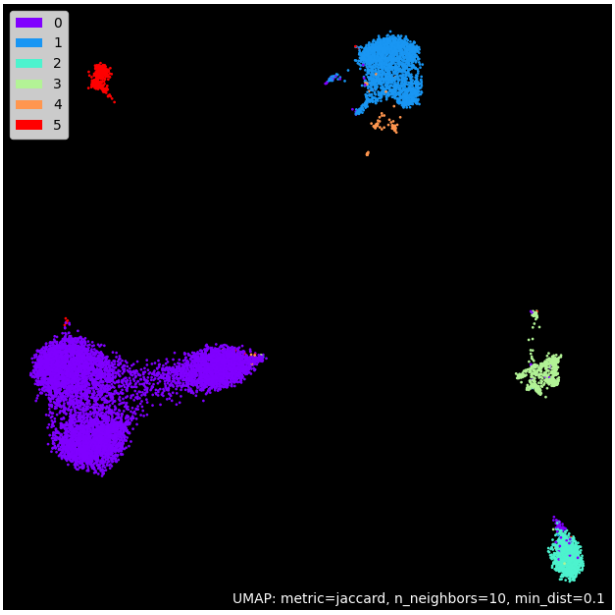
## 5 Discussion

From the results we have observed that the Louvain and Leiden community detection algorithms behave in a similar fashion. The similarity scores of the algorithms quickly peak with regards to the ground truth for low $\gamma$-values and then stabilize, or decline as the value of $\gamma$ increases.

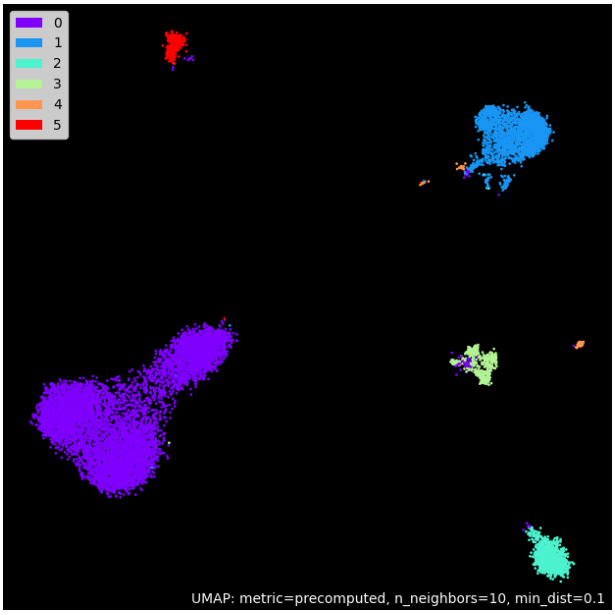
The optimal similarity scores for the ARI, NMI and PSI metrics turned out to be higher for the Leiden algorithm than the Louvain algorithm for all cases. The paper by Traag et al. suggests that the Leiden algorithm outperforms the Louvain algorithm in terms of quality for the conventional clustering pipeline [10]. The results seen in this research could provide supplementary evidence in favor of this suggestion and show that the Leiden algorithm also performs better than the Louvain algorithm for binary clustering. The Leiden algorithm was however used with the parameter $\beta$ set to 0.01 indicating a small degree of randomness. It is therefore possible that the optimal clustering results were only found based on chance. More experiments need to be conducted to verify these findings.

We have also observed that binary clustering shows a high level of resemblance for the ARI and NMI metrics with regards to the ground truth. With maximum ARI similarity scores of 96.72% and 95.77%, most data points for the binary case fall in the same clusters as in the conventional case. However, the maximum PSI scores of 76.63% and 81.3% with the visual assistance of the UMAPs suggested that smaller clusters had little overlap or were missing completely. This could be detrimental when looking for small clusters containing rare cell types.

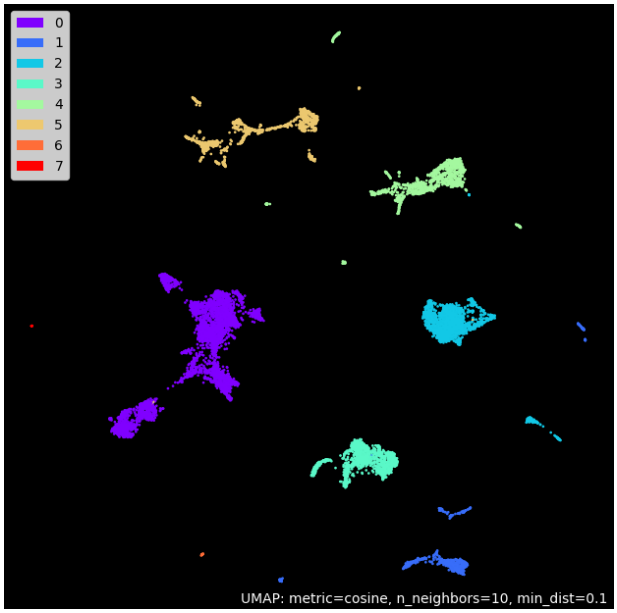
Nevertheless, the binary clustering experiments described in this research were only performed using two different datasets and evaluated against a single ground truth solution for each dataset. Binary clustering may show different behaviour for other datasets or perform better or worse when evaluated against other conventional clustering solutions. The application of unsupervised clustering techniques such as finding optimal clustering solutions based on silhouette score may also highlight yet unforeseen positive features of binary clustering with regards to conventional methods.



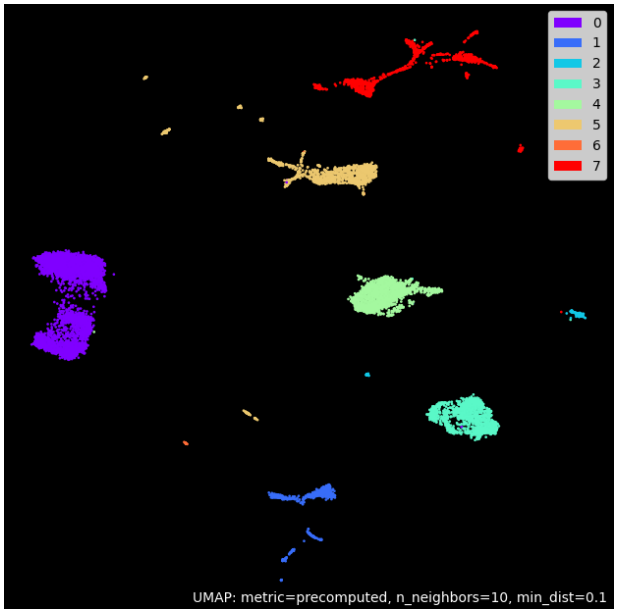

(a) Alzheimer's dataset (Ground truth)




(b) Alzheimer's dataset (Optimal PSI)




(c) Xenopus' tail dataset (Ground truth)




(d) Xenopus' tail dataset (Optimal PSI)

Figure 7: Uniform Manifold Approximation and Projection (UMAP) of (a) the ground truth of the Alzheimer's dataset and (b) the binary clustering solution with the highest PSI score for the Alzheimer's dataset and (c) the ground truth of the Xenopus' tail dataset and (d) the binary clustering solution with the highest PSI score for the Xenopus' tail dataset.

## 6 Conclusion

In this research we posed the question what community detection algorithm results in clusters most similar to the current state-of-the-art clustering methods when applied to binarized scRNA-seq data.

The experiments conducted on a binary clustering pipeline revealed that the Leiden community detection algorithm performed better than the Louvain community detection algorithm for 3 different clustering similarity metrics when compared to a ground truth solution resulting from a conventional clustering pipeline for all datasets.

The experimental results also showed that clusters resulting from the binary clustering pipeline were very similar in composition for bigger clusters, but showed less overlap for smaller clusters, or went completely undetected.

Further research into the impact of binary clustering on the clustering output is necessary to verify the observations made in this research. Applying the research method described in this paper to other datasets may also highlight different features of binary scRNA-seq data clustering that are yet unknown.

## 7 Responsible Research

Here we discuss the ethical aspects of this research. This research was built on top of the contributions of others and it is important that the experimental results are reproducible to prove their validity. The parts of this research that rely on the contributions of others each show a reference to the original source. In this way we acknowledge the intellectual property of others and distinguish it from our own. The experiments in this research were conducted on a self-implemented binary clustering pipeline. We have made the source code of this pipeline publicly available. This allows others to verify the validity of the code or to use the code to run their own experiments and verify our findings. This paper describes the experimental steps that were taken, the exact configuration settings of the pipeline and all the parameter values that were used to accumulate our results. Following this exact workflow guarantees that the results are valid and can be reproduced.

## References


[1] Jovic Dragomirka, Liang Xue, Zeng Hua, Lin Lin, Xu Fengping, and Luo Yonglun. Single-cell rna sequencing technologies and applications: A brief overview. *Clin Transl Med.*, 12(3), 2022.

[2] Philipp Angerer, Alexander F. Wolf, and Theis J. Fabian. Scanpy: large-scale single-cell gene expression data analysis. *Genome Biology*, 19(15), 2018.

[3] K. Y. Yeung and W. L. Ruzzo. Principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9), 2001.

[4] Gerard A. Bouland, Ahmed Mahfouz, and Marcel J. T. Reinders. Differential analysis of binarized single-cell rna sequencing data captures biological variation. *NAR Genomics and Bioinformatics*, 3(4), 2021.

[5] Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M. Mauck 3rd, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *Cell*, 177(7), 2019.

[6] Wojciech Muła, Nathan Kurz, and Daniel Lemire. Faster population counts using avx2 instructions. *Computer Journal*, 61(1), 2016.

[7] Ebony R. Watson, Ariane Mora, Atefeh T. Fard, and Jessica C. Mar. How does the structure of data impact cell–cell similarity? evaluating how structural properties influence the performance of proximity metrics in single cell rna-seq data. *Briefings in Bioinformatics*, 23(6), 2022.

[8] Seung-Seok Choi, Sung-Hyuk Cha, and Charles C. Tappert. A survey of binary similarity and distance measures. *J. Syst. Cybern. Inf.*, 8(1), 2009.

[9] Joëlle Barido-Sottani, Samuel D. Chapman, Evsey Kosman, and Arcady R. Mushegian. Measuring similarity between gene interaction profiles. *BMC Bioinformatics*, 20(435), 2019.

[10] Vincent A. Traag, Ludo Waltman, and Nees J. van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1), 2019.

[11] Lijia Yu, Yue Cao, Jean Y. H. Yang, and Pengyi Yang. Benchmarking clustering algorithms on estimating the number of cell types from single-cell rna-sequencing data. *Genome Biology*, 23(49), 2022.

[12] Mohammad Rezaei and Pasi Franti. Set matching measures for external cluster validity. *IEEE transactions on knowledge and data engineering*, 28(8), 2016.

[13] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection for dimension reduction. *The Journal of Open Source Software*, 3(29), 2018.

[14] Alexandra Grubman, Gabriel Chew, John F. Ouyang, Guizhi Sun, Xin Yi Choo, Catriona McLean, Rebecca K. Simmons, Sam Buckberry, Dulce B. Vargas-Landin, Daniel Poppe, Jahnvi Pflueger, Ryan Lister Owen J. L. Rackham, Enrico Petretto, and Jose M. Polo. A single-cell atlas of entorhinal cortex from individuals with alzheimer's disease reveals cell-type-specific gene expression regulation. *Nature Neuroscience*, 22(12), 2019.

[15] C. Aztekin, T. W. Hiscock, J. C. Marioni, J. B. Gurdon, B. D. Simons, and J. Jullien. Identification of a regeneration-organizing cell in the xenopus tail. *Science*, 364(6441), 2019.