Project Part 1

Jiahui Zhu

October 2024

Abstract

This project replicates the clustering and marker gene analysis from the study "Identification of a Regeneration-Organizing Cell in the Xenopus Tail" by C. Aztekin et al. (2019) using single-cell RNA sequencing (scRNA-seq) data. Both Leiden and Louvain clustering algorithms were applied to identify clusters, with the ROC (regeneration-organizing cell) consistently identified as cluster 10. Performance metrics, including silhouette score, Rand Index, Adjusted Rand Index, and Davies-Bouldin Index, were used to compare clustering methods. Wilcoxon and t-test methods were found to perform better in marker identification, showing higher overlap with the original study compared to logistic regression

1 Introduction

Tissue regeneration is a complex biological process that requires a coordinated response from various cell types. In the study "Identification of a Regeneration-Organizing Cell in the Xenopus Tail" by C. Aztekin et al. (2019), a novel cell type, the regeneration-organizing cell (ROC), was discovered using single-cell RNA sequencing (scRNA-seq). This cell type was found to be essential for the formation of the wound epidermis, a key structure that facilitates successful tissue regeneration in Xenopus laevis tadpoles. This project aims to replicate the clustering and marker gene analysis from the original study to identify ROCs and its associated marker genes within the dataset.

2 Methods

2.1 Data Processing

First, the dataset was filtered to retain only the data corresponding to the day of amputation, resulting in a matrix of cells $\times$ genes (5302 $\times$ 24,184). Median count depth normalization was performed, followed by a `log1p` transformation. Genes expressed in fewer than three cells and cells with fewer than 200 detected genes were excluded (a total of 7,351 genes were expressed in fewer than three cells). Highly variable genes (HVGs) were then selected based on the criteria

outlined in the supplementary material: "Highly variable genes (Fano factor $>$ 65th percentile) were chosen for clustering and visualization, with lowly expressed (mean expression $<$ 5th percentile) and highly expressed (mean expression $>$ 80th percentile) genes removed" (refer to section *scRNA-seq: Data visualization* in the supplementary material). After filtering, the remaining dataset comprised 5302 cells and 6905 genes. Subsequently, the data was scaled, with values exceeding a standard deviation of 10 truncated. Principal component analysis (PCA) was then applied to the scaled data before performing clustering.

2.2 Data Visualization

The neighborhood graph of cells is computed using the PCA representation of the data matrix with parameters `n_neighbors = 20`, `n_pcs = 40`, and `random_state = 12`. The graph is then embedded into two dimensions using the `sc.tl.umap` function in Scanpy with parameters `min_dist = 0.5` and `random_state = 12`. Finally, the clusters identified in the paper are visualized using `sc.pl.umap`.

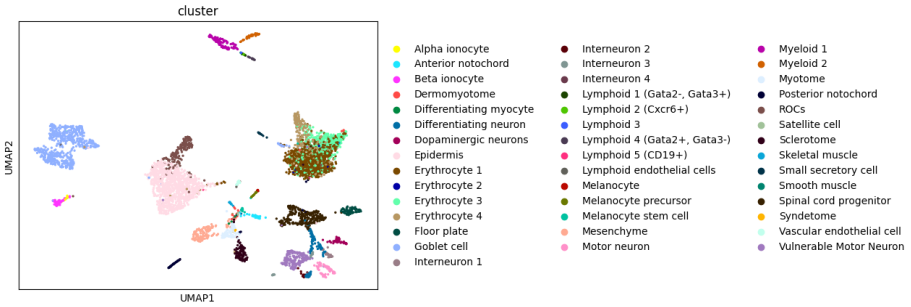

Figure 1: UMAP visualization of clusters in paper

2.3 Clustering analysis

The Leiden and Louvain clustering algorithms were subsequently applied to the precomputed neighborhood graph of cells from the previous section. The resolution parameter was optimized to 0.6, which most closely approximated the clusters observed in the original graph. The ROC (regeneration-organizing cell) cluster was identified as cluster 10.

2.4 Marker Identification

To identify cluster-specific markers, the function `sc.tl.rank_genes_groups` from the Scanpy library was utilized. Three different methods—logistic regression, t-test, and Wilcoxon rank-sum test—were applied and compared. The top 50 gene markers identified for cluster 10, corresponding to the ROC (regeneration-organizing cell) cluster, were selected for further result comparison.

2.5 Code Availability

The code used for analysis is available at: GitHub Repository.

3 Results

3.1 Clustering analysis

3.1.1 Clustering results

As shown in Figure 2, the Leiden clustering algorithm identified 21 clusters, whereas the Louvain algorithm identified 15 clusters under the same resolution. Despite this difference, both methods largely grouped the cells in a similar manner, with the ROC (regeneration-organizing cell) region consistently labeled as cluster 10 in both cases.

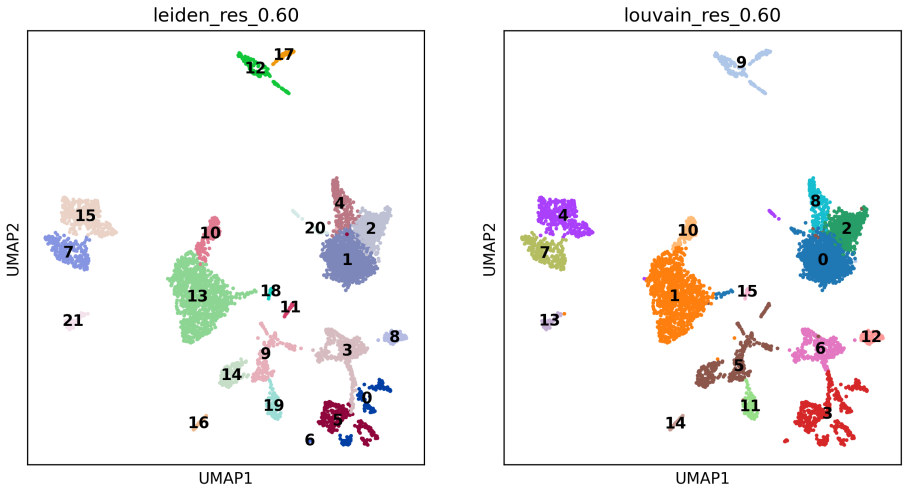

Figure 2: Comparison of Leiden and Louvain clustering methods with resolution parameter set to 0.6.

3.1.2 Metrics comparison

To evaluate the clustering performance, several metrics were computed for both Leiden and Louvain clustering algorithms. The silhouette score, Rand Index (RI), Adjusted Rand Index (ARI), and Davies-Bouldin Index (DBI) were used for comparison.

As shown in Table 1, the Louvain clustering achieved a silhouette score of 0.4520, while the Leiden clustering scored 0.4485. The Rand Index was 0.8917 for Louvain and 0.9255 for Leiden. For the Adjusted Rand Index, Louvain achieved 0.5950, whereas Leiden performed better with 0.6561. Regarding the

Davies-Bouldin Index, the Leiden clustering obtained a score of 0.7070, which was slightly higher than Louvain's score of 0.5947. For comparison, the original clustering from the paper had a DBI of 2.9943.

Since Louvain and Leiden each performed better in two of the four metrics, it is difficult to conclusively favor one over the other. As a result, both clustering methods will be utilized for marker identification in the subsequent analysis.

Clustering Method	**Silhouette Score**	**RI**	**ARI**	**DBI**
Paper Clustering	0.2060	-	-	2.9943
Louvain Clustering	**0.4520**	0.8917	0.5950	**0.5947**
Leiden Clustering	0.4485	**0.9255**	**0.6561**	0.7070

Table 1: Performance metrics comparison for clustering methods.

Note: A lower DBI value indicates better clustering quality.

3.2 Marker Identification

Markers identified using Wilcoxon and t-test methods under both the Leiden and Louvain clustering algorithms show significant overlap. Specifically, 47 and 48 markers were found to be the same for Wilcoxon and t-test, respectively, between the two clustering algorithms. In contrast, the logistic regression method exhibited more variation, with 27 different markers identified between the two clustering algorithms, indicating a less consistent performance compared to Wilcoxon and t-test.

When comparing with the original paper's markers, Wilcoxon identified 36 common markers (35 of which were identical between Leiden and Louvain). The t-test method identified 37 common markers in Leiden and 36 in Louvain, with 36 of them being identical between the two clustering algorithms. Logistic regression performed worse, identifying 29 common markers with Leiden and 34 with Louvain, with 21 markers being the same between the two.

Additionally, when comparing markers across methods (using Leiden as an example), Wilcoxon and t-test identified 49 common markers, while logistic regression shared 33 common markers with these two methods. This indicates that Wilcoxon and t-test are the most similar in terms of the markers they identify.

In summary, Wilcoxon and t-test show the highest level of agreement across both clustering algorithms and methods, making them the more reliable options for marker identification. Logistic regression, however, shows less consistency and performs worse in identifying common markers.

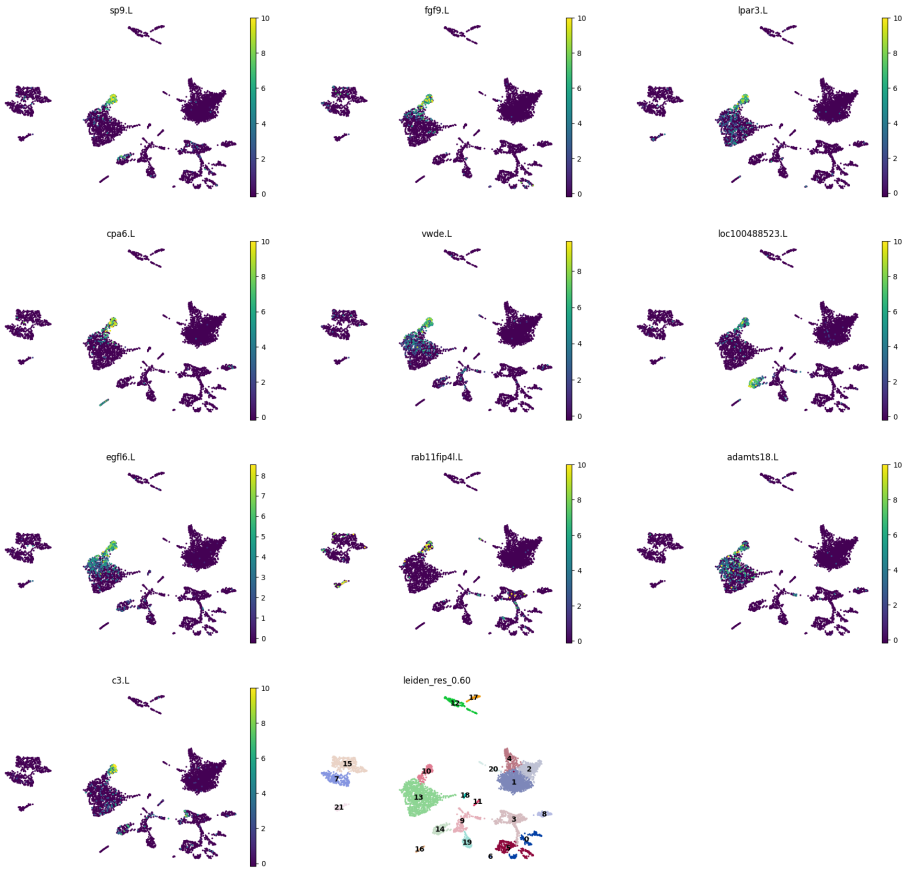

Figure 3: UMAP visualization of clusters using marker genes identified by Leiden clustering at resolution 0.6. The clusters are annotated with key markers such as *sp9*, *fgf9*, *lpar3*, *cpa6*, *vwde*, *loc100488523*, *egfl6*, *rab11fip4*, *adamts18*, and *c3*.

4 Conclusion

This study applied both Leiden and Louvain clustering algorithms, finding significant consistency between them in identifying the regeneration-organizing cell (ROC) cluster. The Wilcoxon and t-test methods demonstrated the highest consistency in marker identification, closely aligning with the results of the original study, whereas logistic regression showed less overlap. The marker identification results could potentially be improved by constructing the neighborhood graph in the same manner as the original paper and implementing the Walktrap clustering algorithm, as outlined in the supplementary material (refer to *scRNA-seq: clustering* in the supplementary material).