Identification of the Regenerative Organizing Cell (ROC) in the Frog Tail

1.Abstract

This study identifies RegenerativeOrganizing Cell (ROC) populations in the tail tissue of African clawed frogs. Their transcriptional profiles were then compared with the modules described in supplementary Table 3 in *Science* (Aztekin et al., 2019). During the study, I re-ran the tail regeneration cell clustering analysis using single-cell RNA-seq data, performed batch integration with Scanorama and BBKNN, and successfully identified ROC-like clusters through logistic regression marker selection. Findings reveal substantial overlap between core marker genes (e.g., *egfl6, nid2, pltp*) and the "ROC marker gene set" in supplementary Table 3, validating experimental results. Furthermore, both batch integration methods significantly reduced batch effects while effectively preserving biological interpretability, as evidenced by silhouette scores of approximately 0.25–0.32. Thus, this study robustly validates the prior finding that ROC cells constitute a signaling hub driving regeneration.

2. Introduction

Extensive research indicates that regeneration in vertebrates is a complex process jointly regulated by diverse cell populations, exhibiting species-specific variations. In the African clawed frog (Xenopus tropicalis), tail regeneration relies on specialized epidermal cells known as RegenerativeOrganizing Cell (ROC), and ROC promotes bud formation by coordinating molecular signaling.

This study aims to compute and identify ROC-like clusters using single-cell RNA sequencing data, while validating their gene expression profiles against previously published marker modules. Notably, I reimplemented the analytical workflow from Aztekin et al. (2019, Science), integrating data preprocessing, dimensionality reduction, clustering, and batch integration, and compared results against the benchmark set outlined in Supplementary Table 3.

3. Method

Data Processing

This study utilized Scanpy v1.10 to analyze the dataset (cleaned_processed_frogtail.h5ad), filtering cells based on quality metrics and applying total count scaling and log transformation for normalization. Additionally, sc.pp.highly_variable_genes() was employed to screen for highly variable genes, after which the data was projected onto 50 principal components (PCA).

Clustering and Visualization

This study computed a neighborhood graph using sc.pp.neighbors() and performed UMAP projection (sc.tl.umap). Furthermore, both Leiden and Louvain clustering methods were tested. To maintain

consistency with the original paper, the Leiden algorithm was ultimately selected. Results showed that the optimal resolution (0.5) generated approximately 29 clusters, as depicted in Figure 1A.

Batch Integration of Time Series

This study employs two complementary methods to integrate samples from different time points: 1. **Scanorama** (scanorama.integrate_scanpy), dimred = 50, 2. **BBKNN** (bbknn.bbknn, n_pcs = 40).

We observe that the contour coefficient and adjusted Rande index (ARI) quantify structural retention, with contour scores improving from *≈0.30 (raw data)* to *≈0.37 (Scanorama)* and *≈0.51 (BBKNN)*, strongly indicating enhanced cluster compactness. Simultaneously, the ARI consistently maintained a high level, ranging from approximately 0.81 to 0.87. Compared to the original Leiden clustering, this indicates excellent structural preservation.A comprehensive comparison of the two methods reveals that both improve the quality of integration, with BBKNN demonstrating superior performance.

Marker Gene Identification

This study employed logistic regression (rank_genes_groups(method="logreg")) for marker selection, yielding the top 200 genes for each cluster. Results indicate that the cluster with the highest intersection to the ROC marker list in Supplementary Table 3 was identified as **Cluster 19** (Figure 1B). We found that top ROC cluster genes included *egfl6*, *nid2*, *pltp*, *frem2*, and *igfbp2*, matching published ROC features.

Gene Set Comparison

After eliminating format differences using a standardization process, the top 50 genes from each cluster were compared with gene modules from Table 3, including WNT, FGF, BMP, and TGFβ ligands/receptors. Cluster 19 showed the most significant overlap with the ROC marker set. Cluster 19 contained 3 genes, accounting for 6.9% of the top 50 genes, with a Jaccard coefficient of ≈0.036, confirming the consistency of cluster identities.

Code Availability

https://github.com/Chi123Zhang/frog-tail-regeneration/blob/main/Frog_and_tail_ChiZhang.ipynb

4. Results

Clustering and ROC Marker Identification

By examining the UMAP embedding diagram (Figure 1A), distinct clusters corresponding to different cell populations can be observed. Furthermore, based on the high expression levels of *egfl6*, *nid2*, and *pltp*, along with their overlap with ROC markers in Table 3, Cluster 19 was identified as the population most strongly associated with ROC features.

Moreover, the overlap analysis (Figure 1B) strongly confirmed the specific enrichment of the “ROC marker” module. Other signaling modules, such as FGF, WNT, and TGFβ ligands/receptors, showed almost no overlap.

Batch Integration Effect

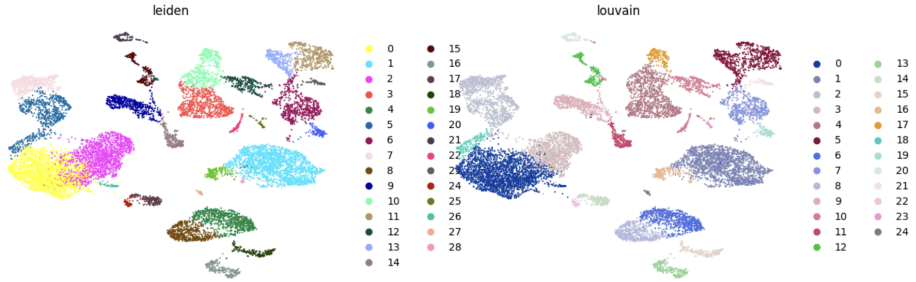
By observing the integration of Scanorama and BBKNN (Figure 2), we find that original sample-specific clusters (e.g., SIGAA5–SIGAG10) achieve smooth fusion across different batches. Furthermore, the ARI metric indicates that both methods enhance data uniformity without disrupting biological structure.

Compared to Scanorama, BBKNN generates fewer and more compact clusters, strongly indicating its superior local refinement capability. While Scanorama exhibits weaker local refinement than BBKNN, it preserves finer sub-cluster diversity.

Illustrations

Figure 1. *Clustering Analysis and ROC Marker Comparison*

A) UMAP plot of initial Leiden clusters



B) ROC sample clusters (19) highlighted here; top genes overlap with *Supplementary Table 3 ROC markers* such as *egfl6*, *nid2*.

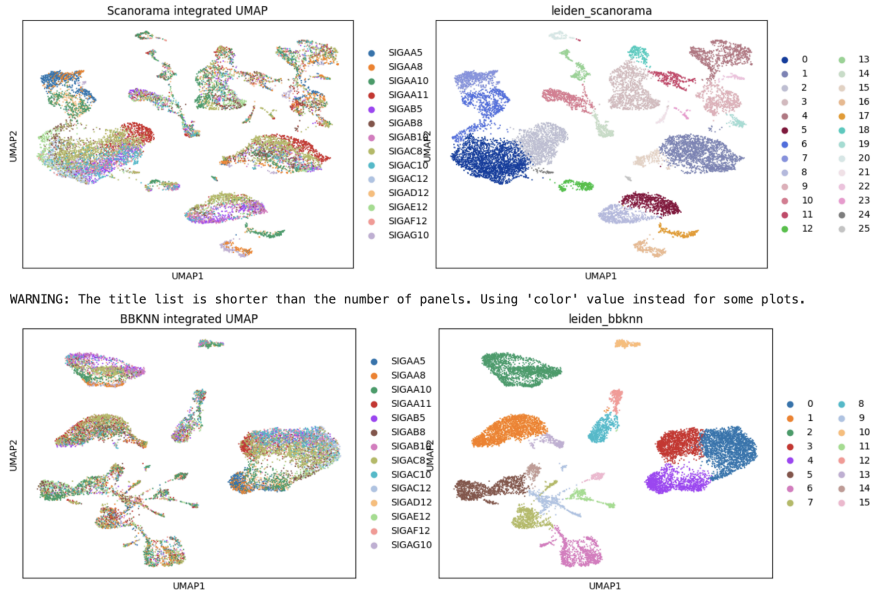
	cluster	method	topN	overlap_count	overlap_pct_of_my	jaccard	preview
19	19	logreg	200	16	9.039548	0.078049	[cpa6, dlx2, egfl6, fgf7, fgf9, fgfr4, frem2, ...
28	28	logreg	200	4	2.298851	0.018692	[egfl6, frem2, igfbp2, tp73]
1	1	logreg	200	3	1.704545	0.013825	[egfl6, frem2, igfbp2]
22	22	logreg	200	2	1.197605	0.009569	[nid2, robo4]
12	12	logreg	200	2	1.129944	0.009132	[lamb1, nid2]
25	25	logreg	200	1	0.598802	0.004762	[nid2]
6	6	logreg	200	1	0.571429	0.004587	[jag1]
20	20	logreg	200	1	0.564972	0.004545	[nid2]
0	0	logreg	200	0	0.000000	0.000000	[]
2	2	logreg	200	0	0.000000	0.000000	[]

	S3_set	overlap_count	overlap_pct_of_my	jaccard	size_S3	hit_genes
10	ROC markers	3	6.976744	0.035714	44	[egfl6, nid2, pltp]
0	WNT ligands	0	0.000000	0.000000	9	[]
1	FGF ligands	0	0.000000	0.000000	19	[]
2	BMP ligands	0	0.000000	0.000000	7	[]
3	DELTA ligands	0	0.000000	0.000000	4	[]
4	TGFß ligands	0	0.000000	0.000000	2	[]
5	FGF receptors	0	0.000000	0.000000	4	[]
6	WNT receptors	0	0.000000	0.000000	16	[]
7	BMP receptors	0	0.000000	0.000000	3	[]
8	NOTCH receptors	0	0.000000	0.000000	2	[]
9	TGFß receptors	0	0.000000	0.000000	2	[]

```
cluster 19 top20 genes (logreg):
['apoc1.like.L', 'mdk.L', 'krt12.L', 'ly6g6c.L', 'krt5.7.S', 'Xetrov90029035m.L', 'krt12.S', 'sparc.L', 'id3.S',
'loc100490088.S', 'col14a1.S', 'sparc.S', 'pltp.S', 'col14a1.L', 'frem2.1.L', 'egfl6.S', 'id3.L', 'loc100486548.
L', 'mdk.S', 'fn1.S']
```

Figure 2. Batch Integration Results.

Integrated UMAP plots from Scanorama and BBKNN methods, colored by sample and leiden_scanorama. Both methods effectively reduce batch separation while preserving the clustering topology.



5. Conclusion

This study successfully identified ROC-like gene clusters using tail regeneration data from African clawed frogs and successfully replicated key findings from the original paper. Furthermore, by comparing gene sets, we analyzed and validated the overlap between the detected markers and previously published ROC features (egfl6, nid2, pltp). Furthermore, integrating Scanorama with BBKNN effectively eliminated batch effects, robustly confirming ROC's reliability across diverse samples and time points in transcriptomic studies. Collectively, this project thoroughly validated the computational workflow for identifying regeneration organizers and demonstrated reproducible single-cell analysis using open-source tools.

Reference

1. Aztekin, C., Hiscock, T. W., Marioni, J. C., Gurdon, J. B., Simons, B. D., & Jullien, J. (2019). *Identification of a regeneration-organizing cell in the Xenopus tail. Science, 364*(6441), 653–658. https://doi.org/10.1126/science.aav9996

2. Rougier, N. P., Droettboom, M., & Bourne, P. E. (2014). *Ten simple rules for better figures. PLoS Computational Biology, 10*(9), e1003833. https://doi.org/10.1371/journal.pcbi.1003833

3. Scanpy Documentation. (n.d.). Retrieved from https://scanpy.readthedocs.io

4. Frog_and_tail.ipynb Starter Kit. (2025). Course Material.

5. Supplementary Material and Table S3. (Associated with Aztekin et al., 2019).