5 Appendix: Additional results

5.1 Mouse embryonic stem cell differentiation

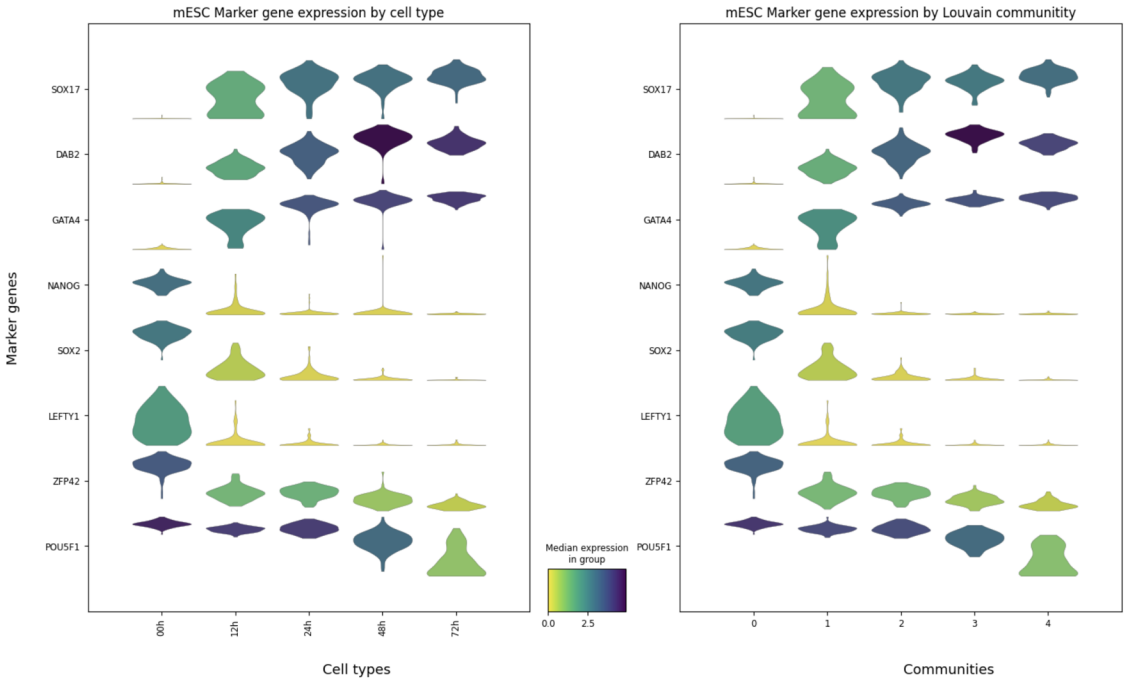

Figure 6: **Gene expression patterns of stage-specific markers during mESC differentiation.**

Figure 6 shows the comparison of marker gene expression in different cell states (left panel) to our inferred communities (right panel). We again observe that the stage specific markers follow the expected trend in gene expression over the inferred communities.

One noticeable difference between the two panels is the fact that many of the violin plots shown on the left, especially for the later time points (24h, 48h, 72h), have long tails, whereas the closest inferred communities (2-4) do not. This indicates that there are cells in these later states that do not adhere to the expression pattern of other cells in their state. BIRDccNEST mitigates these mis-classifications, giving a visual explanation of why the DBI is lower for the BIRDccNEST communities than the annotated time points.

5.2 Mouse bone-marrow-derived dendritic cells with immune stimulation

The mDC dataset examines over 1700 bone-marrow derived dendritic cells under various conditions, focusing primarily on cell-cell communication and its control of cellular heterogeneity. BEELINE benchmarks a portion of this dataset, namely LPS- stimulated wild-type dendritic cells over a time course measured at 1h, 2h, 4h, and 6h post-stimulation. Although this portion of the dataset comprises an inherent trajectory of an induced immune response, Shalek et al. [22] conclude that many cells are either ahead of, or behind, their expected schedules. Talking of a PCA plot of the data, their paper notes that while PC1 partially distinguishes early time points (1h, 2h) from later ones (4h, 6h), the expression levels of stage-specific markers vary substantially between cells within any single stimulus time point.

Figure 7 shows our results for this dataset. Our most noteworthy results are perhaps the substantial gain in clustering coherence that BIRDccNEST achieves, as this is the dataset with one of the greatest change in DBI and Silhoutte Score between the expected cell states and BIRDccNEST communities (Table 2). This observation is supported by the comments of Shalek et al.; Figure 7c shows that the BIRDccNEST communities largely do *not* correspond to the post-LPS time points, especially in the later stages.

We also can visually confirm from Figure 7d, which shows the recreated PCA plot for the first two principal components, that cells grouped by the measured time points indeed appear inseparable. However, Figures 7a, b, and e show that our cell network structurally embeds them in a manner that makes them more separable.

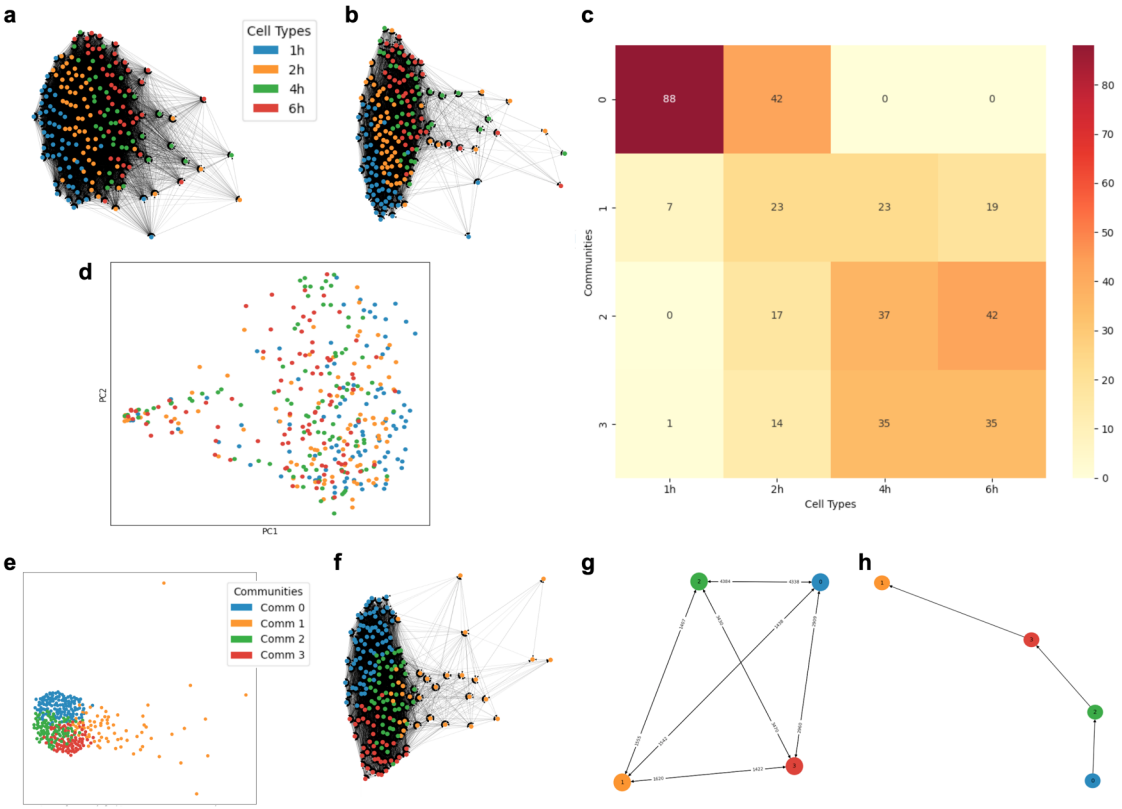

Figure 7: **Results for the mDC dataset.** Cell type legend applies to subfigures a, b, and d; Communities legend applies to subfigures e-h. a) Inferred cell-cell network, showing a sample of 50% of nodes. b) Pruned cell-cell network with top 40% quartile edge weights (in order to preserve connectedness of network), showing the same sample of nodes. c) Heatmap of overlap between known time points and inferred communities. d) PCA of dataset for first two principal components. e) All cell-cell network nodes colored by inferred communities displayed without edges. f) Pruned cell-cell network of inferred communities, showing the same sample of nodes. g) Cluster flow network with edge counts. h) Oriented maximum spanning tree showing inferred trajectory.

The inferred trajectory (7h) starts with the earliest cluster and progresses in a reasonable way given the network projection in Figure 7e.

5.3 Mouse hematopoietic stem cell lineages

The mHSC data set includes counts for 4773 genes profiled in 1656 hematopoietic stem and progenitor cells (HSPCs) from three lineages: erythroid (E), granulocyte-monocyte (GM), and lymphoid (L). BEELINE benchmarks the lineages separately, where mHSC-E corresponds to the erythroid lineage, mHSC-GM corresponds to the granulocyte-monocyte lineage and mHSC-L correspond to lymphoid lineage for gene regulatory network inference purposes. The combined dataset as presented by Nestorowa et al. [16] highlights the differentiation hierarchy that starts from hematopoietic stem cells (HSCs) to produce the full spectrum of mature blood cells via intermediate multipotent progenitor cells. Nestorowa et al. characterized the splits in the dataset that correspond to these distinct lineages by utilizing three-dimensional diffusion map embeddings to identify a start cell and end cells.

Figures 8, 9 and 10 show our results for datasets mHSC-E, mHCS-GM and mHCS-L, respectively. Cell types adapted for BEELINE from the original publication are broadest complete form: long-term hematopoietic stem cells (LT-HSCs or LT), hematopoietic stem and progenitor cells (HSPCs) that primarily range from ST-HSCs to MPPs, and "Prog", which represents lineage-specific commited progenitor cells such as MEPs for erythroid, GMPs for granulocyte-monocyte and LMPPs for lymphoid. Even these very coarse groupings were not as distinct as one might hope in the PCA plots based on the original data, suggesting that a more fine-grained investigation

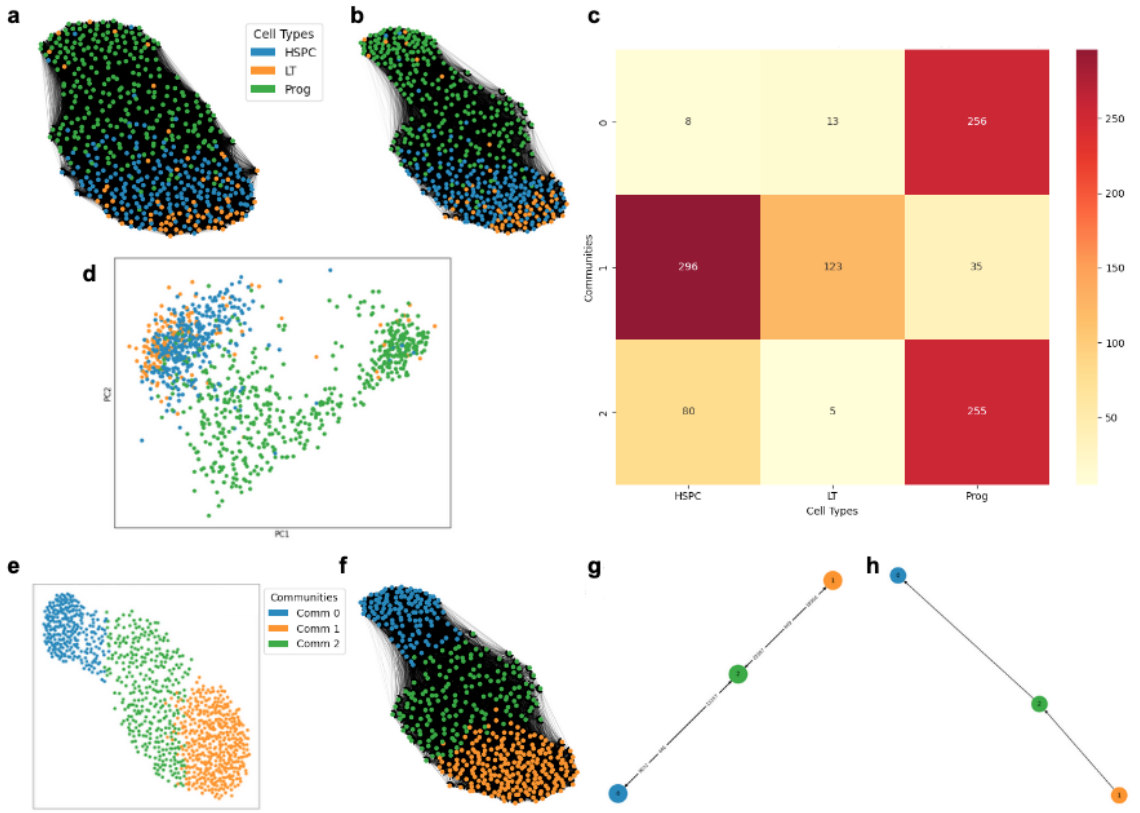

Figure 8: **Results for the mHSC-E dataset.** Cell type legend applies to subfigures a, b, and d; Communities legend applies to subfigures e-h. a) Inferred cell-cell network, showing a sample of 50% of nodes. b) Pruned cell-cell network with top quartile edge weights, showing the same sample of nodes. c) Heatmap of overlap between known time points and inferred communities. d) PCA of dataset for first two principal components. e) All cell-cell network nodes colored by inferred communities displayed without edges. f) Pruned cell-cell network of inferred communities, showing the same sample of nodes. g) Cluster flow network with edge counts. h) Oriented maximum spanning tree showing inferred trajectory.

of the subtypes might be appropriate. That said, both are more reasonably separated by the cell networks based on expression patterns and are then grouped linearly in apparent trajectories.

Given questions about the similarity of the progenitor cells from these lineages and their relative classifications, either drilling down into more specific cell types by favoring smaller communities or running BIRDccNEST on the combined multi-lineage data set might be informative directions for future work.

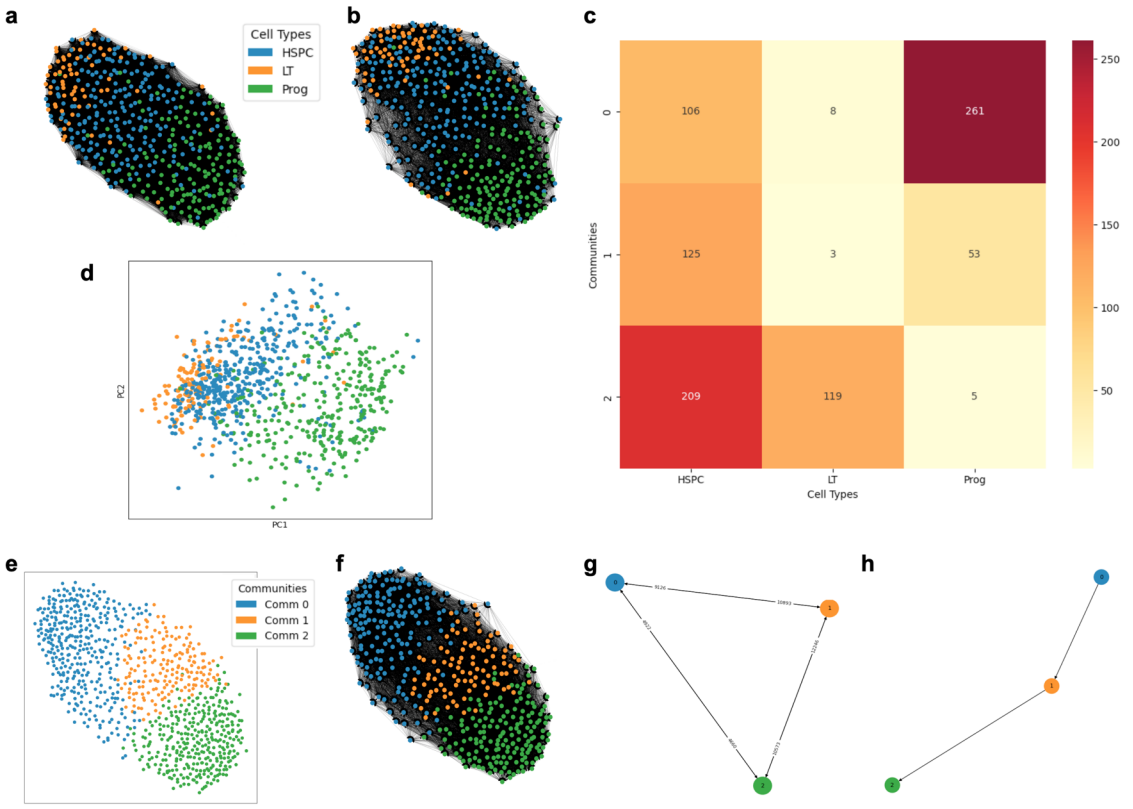

Figure 9: **Results for the mHSC-GM dataset.** Cell type legend applies to subfigures a, b, and d; Communities legend applies to subfigures e-h. a) Inferred cell-cell network, showing a sample of 50% of nodes. b) Pruned cell-cell network with top quartile edge weights, showing the same sample of nodes. c) Heatmap of overlap between known time points and inferred communities. d) PCA of dataset for first two principal components. e) All cell-cell network nodes colored by inferred communities displayed without edges. f) Pruned cell-cell network of inferred communities, showing the same sample of nodes. g) Cluster flow network with edge counts. h) Oriented maximum spanning tree showing inferred trajectory.

5.4 Comparison to other TI algorithms

To contextualize BIRDccNEST's strength and performance in trajectory inference, we compared its inferred trajectories to trajectories inferred by established TI algorithms PAGA+DPT and Slingshot. Partition-based graph abstraction (PAGA) generalizes the relationship between cell clusters found on a dimensionality reduced space via kNN graphs as an interpretable graph-like map that quantifies the connectivity between clusters [29]. PAGA is generally jointly used with Diffusion pseudotime (DPT) to construct the trajectory, where DPT operates on diffusion components of the diffusion map (rather than principal components in PCA) to assign transition probabilities between consecutive cell-cell pairs of a random walk [9]. As the PAGA map is undirected, to order the random walk through the PAGA+DPT algorithm requires a root cell to be specified. We chose this as one of the algorithms to compare BIRDccNEST to, as the PAGA map bears close resemblance to the functionality of our cluster flow networks, the difference being that the cluster flow networks abstracts connectivity between cell clusters with directionality.

Slingshot is considered as one of overall better performing TI algorithm (according to Dynoverse benchmarking [20]), that is a manifold-learning based approach that learns connections and lineage between cell clusters through a combination of constructing minimum spanning trees (MST) and fitting principal curves [24]. To assign ordering while constructing branching curves, Slingshot does require a starting cluster (i.e. root cluster) to be specified. It also allows for supervision in the form of specifying end points (i.e. terminal clusters).

In the sections below we show the trajectory inferred as pseudotemporal ordering by each method for all the BEELINE datasets. For each method we followed workflow and recommendations as outlined by their associated

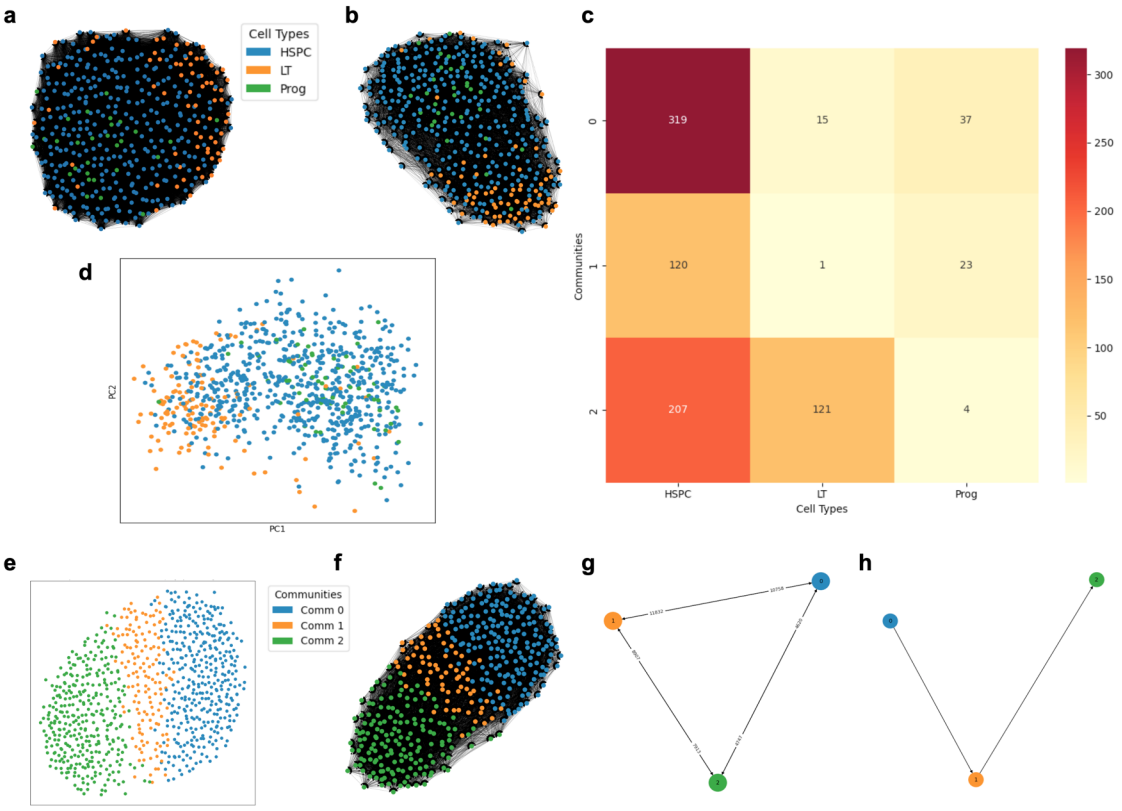

Figure 10: **Results for the mHSC-L dataset.** Cell type legend applies to subfigures a, b, and d; Communities legend applies to subfigures e-h. a) Inferred cell-cell network, showing a sample of 50% of nodes. b) Pruned cell-cell network with top quartile edge weights, showing the same sample of nodes. c) Heatmap of overlap between known time points and inferred communities. d) PCA of dataset for first two principal components. e) All cell-cell network nodes colored by inferred communities displayed without edges. f) Pruned cell-cell network of inferred communities, showing the same sample of nodes. g) Cluster flow network with edge counts. h) Oriented maximum spanning tree showing inferred trajectory.

papers and documentations, in which both started by projecting the dataset onto a lower dimensional representation with PCA. Both then recommended additional truncation of the representation to denoise, where PAGA+DPT used Diffusion maps and Slingshot used UMAP. Then kNN graph was computed to extract the subsequent Louvain clusters on the representation. Lastly, a root cell (for PAGA+DPT) or root cluster (for Slingshot) was specified to compute the psedotemporal ordering of cell describing the trajectory.

Overall, we observe that BIRDccNEST not only captures the underlying trajectory correctly when PAGA+DPT and/or Slingshot cannot, but it also categorizes transitory cell types/states connectivity and ordering more precisely without supervision.

5.4.1 PAGA+DPT and Slingshot for hESC

Figures 11 and 12 show the workflow and trajectory found for hESC with PAGA+DPT and Slingshot, respectively. We notice that both methods does not find the expected trajectory for the dataset, when BIRDccNEST can. With PAGA+DPT, the diffusion components spreads cells too apart, to the point that the PAGA graph in Figure 11c is computed as disconnected that means it assumes that the cells are on separate lineages, when in fact the underlying trajectory is a continuous differentiation. Consequently, when a root cell is specified from the 00bh4s state, DPT can only compute psedotemporal ordering for the cells that were in clusters the PAGA graph were connected (i.e. clusters 8, 11, 12) as seen in Figure 11d.

Unlike PAGA+DPT, Slingshot at least infers a continuous trajectory across all cells rather than as separate lineages, but mistakes transitory cell types/states to be terminal points as shown in Figure 12c. The inferred trajectory

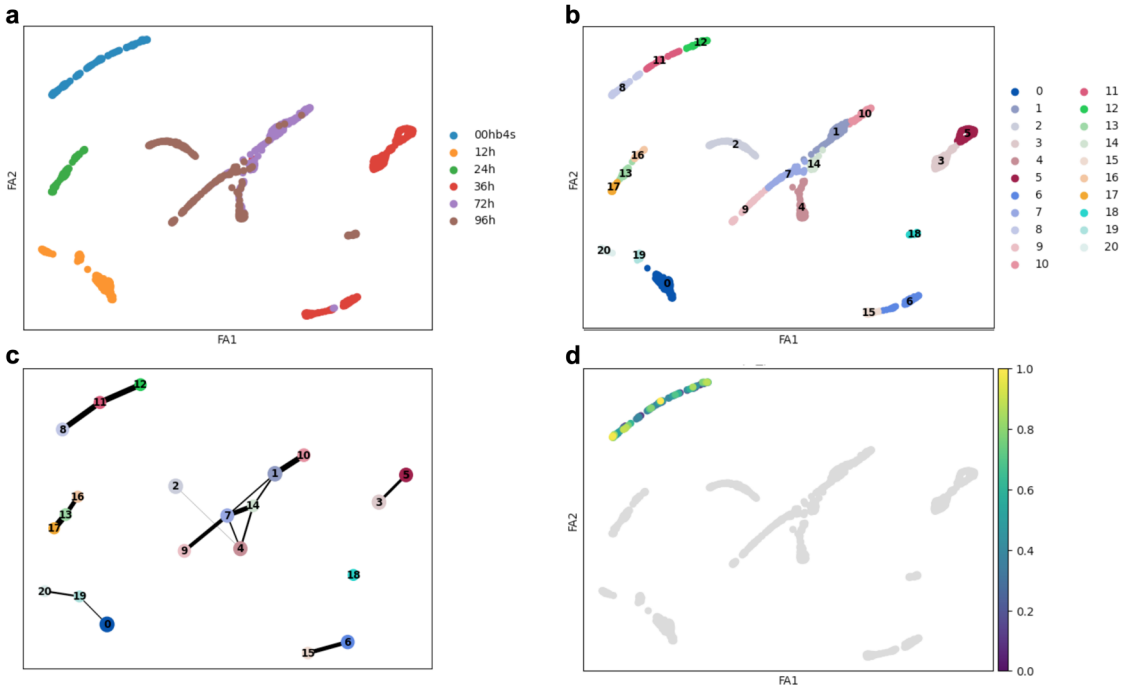

Figure 11: **PAGA+DPT applied on hESC** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell states. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from 00hb4s state.

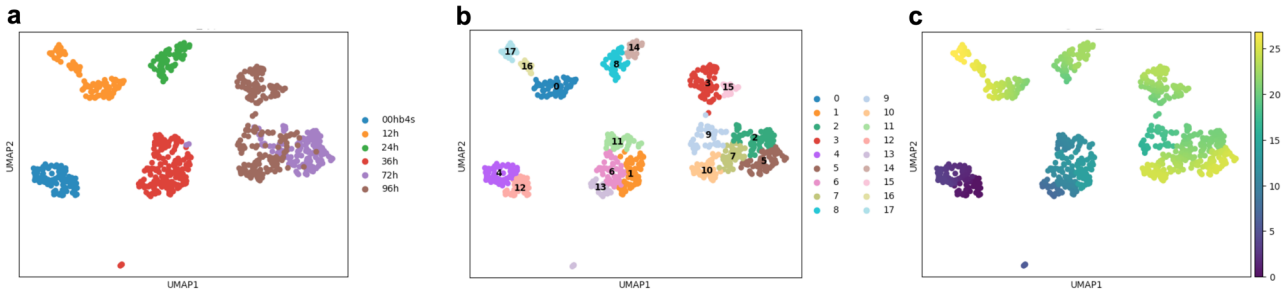

Figure 12: **Slingshot applied on hESC** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell states. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 4.

also psedotemporally orders cells from 36h right after cells in 00hb4s, when cells in 12h and 24h are ordered as succeeding cells from 36h and some even being terminal. Although Slingshot allows for specifying endpoints to mitigate such issues, it is not required, and in the absence of this form of supervision fails to order the transitory cell types/states precisely, when BIRDccNEST does correctly without any supervision as shown on Figure 1h. This shows that BIRDccNEST can capture a closer representation for cells that are in adjacent cell states with the cell-cell network, when 2d representation of cells in PCA, UMAP, or Diffusion map space can miss the closeness.

5.4.2 PAGA+DPT and Slingshot for mESC

Figures 13 and 14 show the workflow and trajectory found for mESC with PAGA+DPT and Slingshot, respectively. We again observe that PAGA+DPT and Slingshot do not find the expected trajectory for the mESC dataset and makes similar types of inference mistakes as seen on the inference on the hESC dataset. With PAGA+DPT,

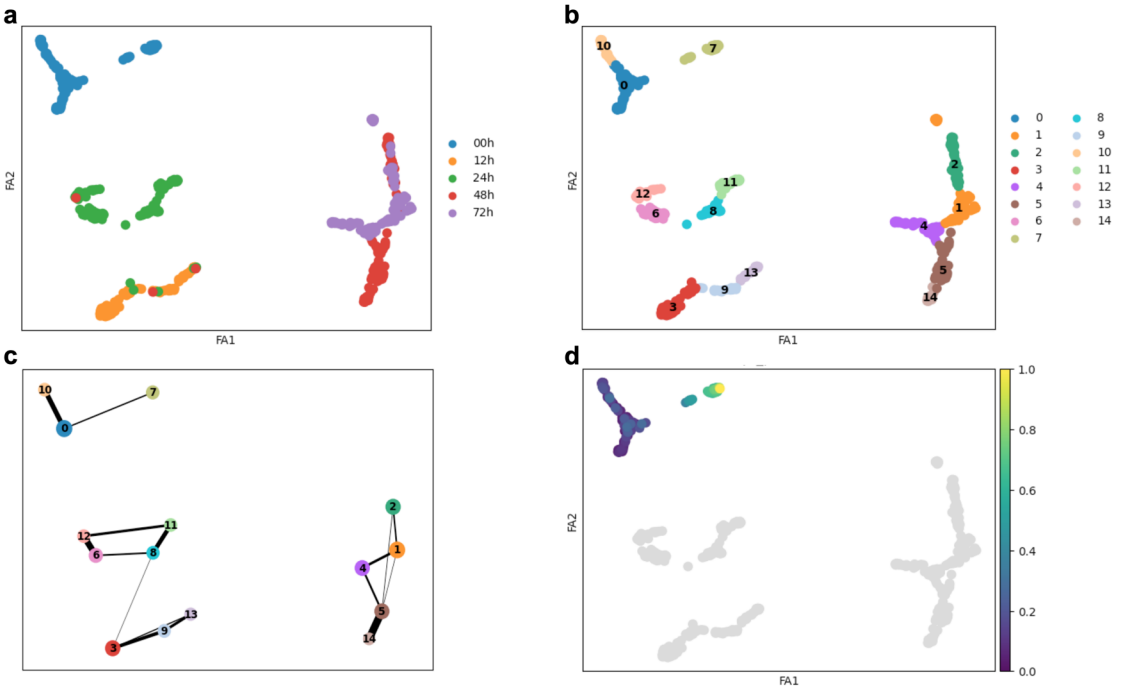

Figure 13: **PAGA+DPT applied on mESC** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell states. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from 00h state.

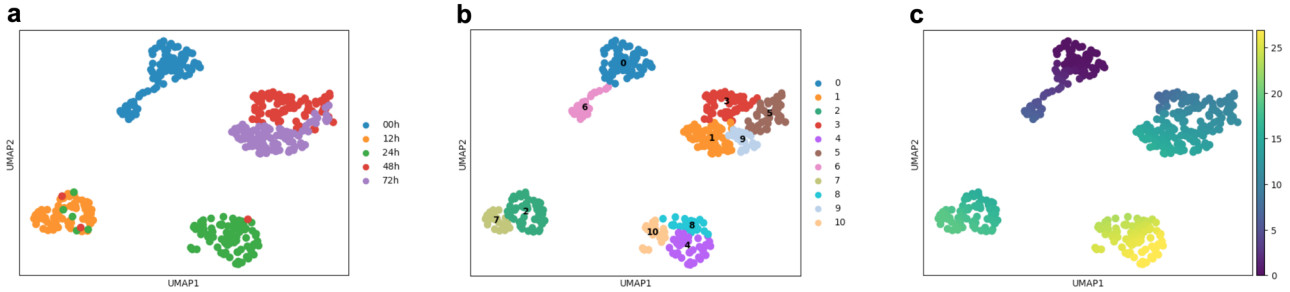

Figure 14: **Slingshot applied on mESC** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell states. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 0.

we again see that the cells are inferred to be on separate lineages in Figures 13 c and d, missing the continuous trajectory across all cells.

With Slingshot, we again see that in the absence of addition optional supervision, fails to order adjacent cell states correctly as they are represented to be father away in the 2d space (Figure 12c). Comparatively, we see in Figure 3h that BIRDccNEST does not make this mistake and infers the expected correct trajectory.

5.4.3 PAGA+DPT and Slingshot for hHep

Figures 15 and 16 show the workflow and trajectory found for hHep with PAGA+DPT and Slingshot, respectively. With PAGA+DPT, we see that the constructed PAGA graph in Figure 15c is mostly connected, which yields to a continuous trajectory across most cells being inferred in Figure 15d. Although the psedotemporal ordering correctly identifies the cells of type mh1 and mh2 as terminal, it lacks the precise ordering of the transitory cell types. Namely, it orders cell types he1 and he2 right after ipsc cells, rather than the expected next cell type of de

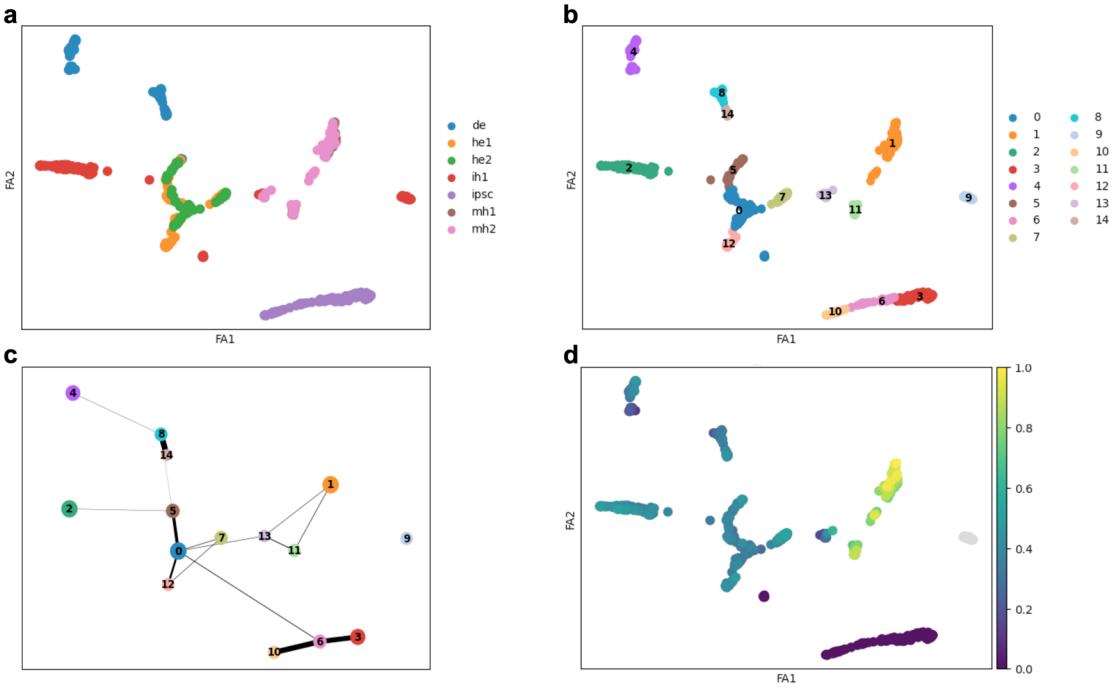

Figure 15: **PAGA+DPT applied on hHep** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell types. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from ipsc type.

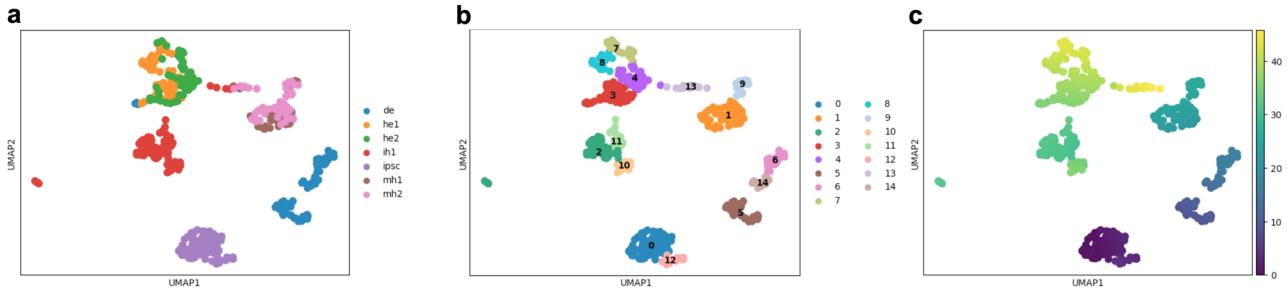

Figure 16: **Slingshot applied on hHep** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell types. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 0.

cells. This is due to the fact that on the PAGA graph the only edge between clusters that have ipsc cells is to a cluster that have he cells. In contrast, in Figure 4g we observe that BIRDccNEST does show directed connectivity from the cluster that has most ipsc cells to the clusters with he1 and he2 cells, however a closer directed edge from the cluster with ipsc cells to the cluster with de cells exist. This enables BIRDccNEST to identify the correct association between these inherently adjacent cell types in Figure 4h and ordering them appropriately. With Slingshot, we observe in Figure 16c the correct terminal state is not identified and the precise ordering of transitory states is lacking.

5.4.4 PAGA+DPT and Slingshot for mDC

Figures 17 and 18 show the workflow and trajectory found for mDC with PAGA+DPT and Slingshot, respectively. Arguably both methods perform reasonably well in inferring the expected trajectory for this dataset. Compared to the PCA for the dataset shown in 7d, the denoising done by diffusion map in PAGA+DPT and by UMAP in Slingshot allows the cells to spread more making distinct cell populations more separable. Although, BIRDcc-

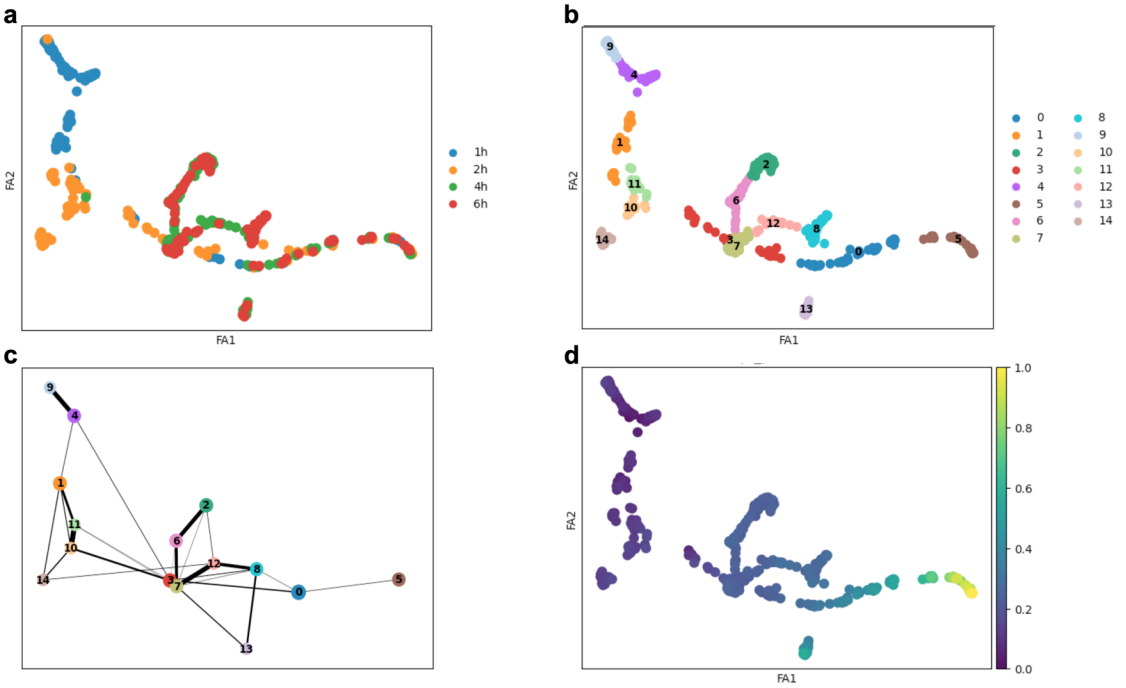

Figure 17: **PAGA+DPT applied on mDC** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell states. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from 1h state.

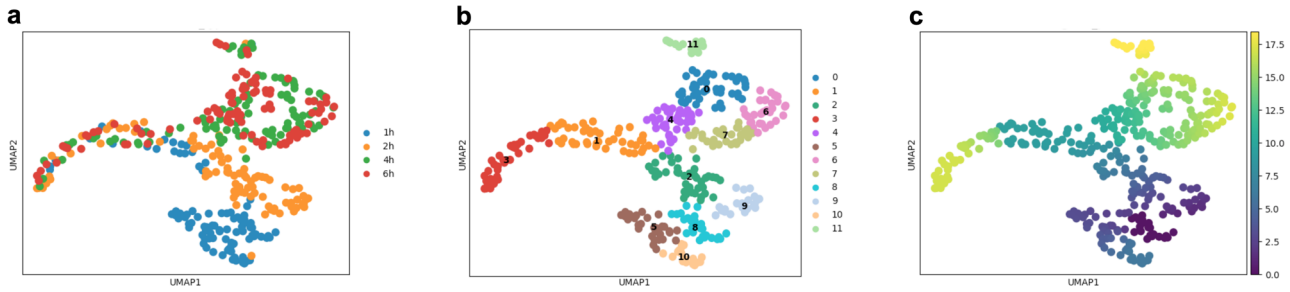

Figure 18: **Slingshot applied on mDC** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell states. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 10.

NEST improves on the separability of cell population compared to PCA, that fact that less of its cell-cell network edges gets pruned may hinder further separability to be observed. Hence, why we hypothesize that the trajectory inferred by PAGA+DPT and Slingshot has a leg up. For future work, we plan to inquire more about varying the percentage of pruning and possibly the effects of pruning to a degree in which the network becomes weakly connected or even disconnected, rather than simply prioritizing for strongly connected network. We do note that BIRDccNEST still extracts a reasonable trajectory without the need of supervision, while both PAGA+DPT and Slingshot would fail in its absence.

5.4.5 PAGA+DPT and Slingshot for mHSC-E,GM,L

Figure pairs (19 and 20), (21 and 22), (23 and 24) show the workflow and trajectory found by PAGA+DPT and Slingshot for the datasets mHSC-E, mHSC-GM, and mHSC-L, respectively. Both methods perform similarly across datasets as BIRDccNEST in terms of trajectory inference. One thing of note in Figure 20c, which shows

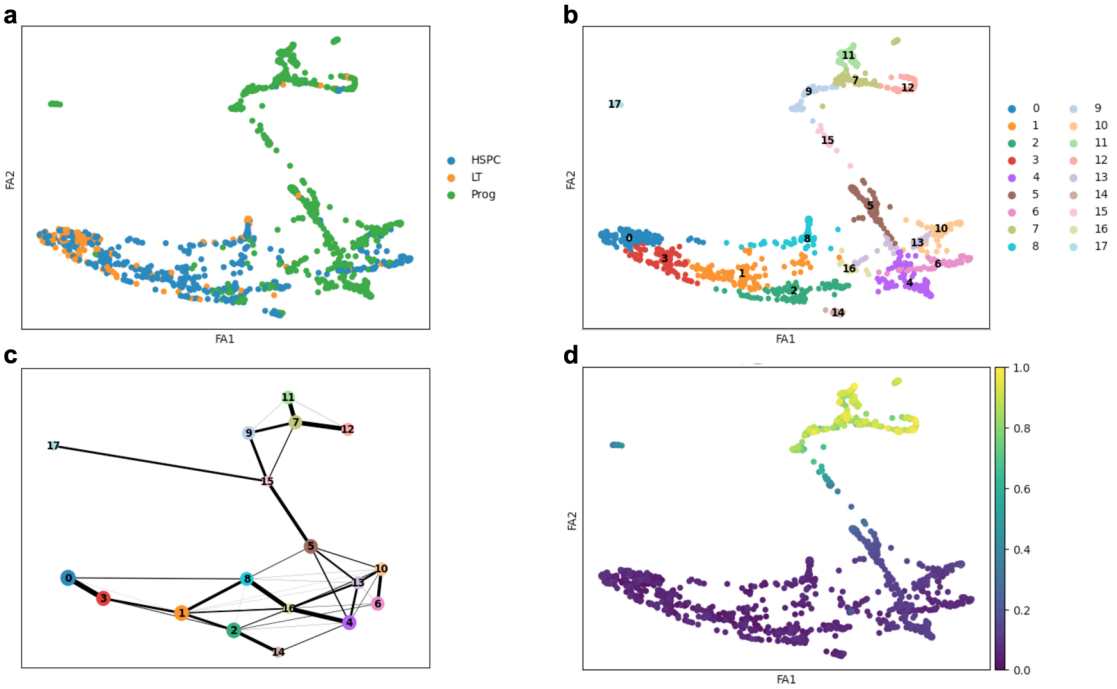

Figure 19: **PAGA+DPT applied on mHSC-E** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell types. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from LT type.

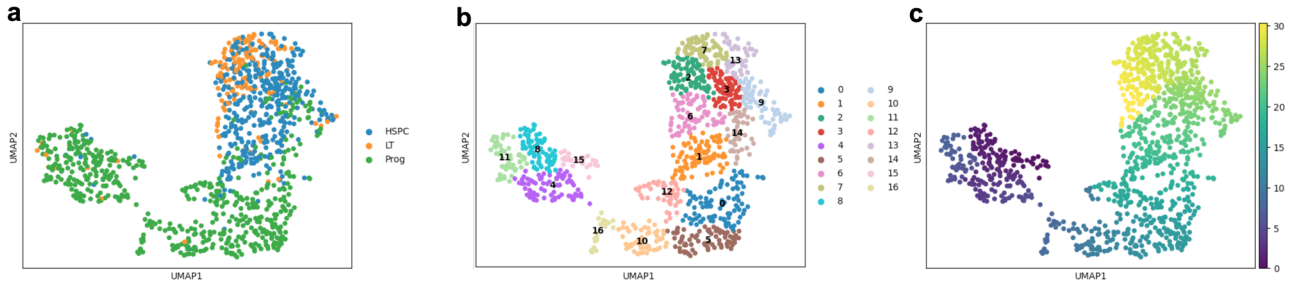

Figure 20: **Slingshot applied on mHSC-E** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell types. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 7.

the trajectory found for mHSC-E by Slingshot, is that although a start cluster is specified the pseudotemporal ordering found is in reverse. It's not apparent if this simply is an indication that for this dataset significantly more epochs were needed to run during fitting, as the resultant trajectory did not change when run for more. Overall, PAGA+DPT and Slingshot infers a more granular trajectory, which highlights further inspection needed into smaller communities to be found with BIRDccNEST for the mHSC dataset.

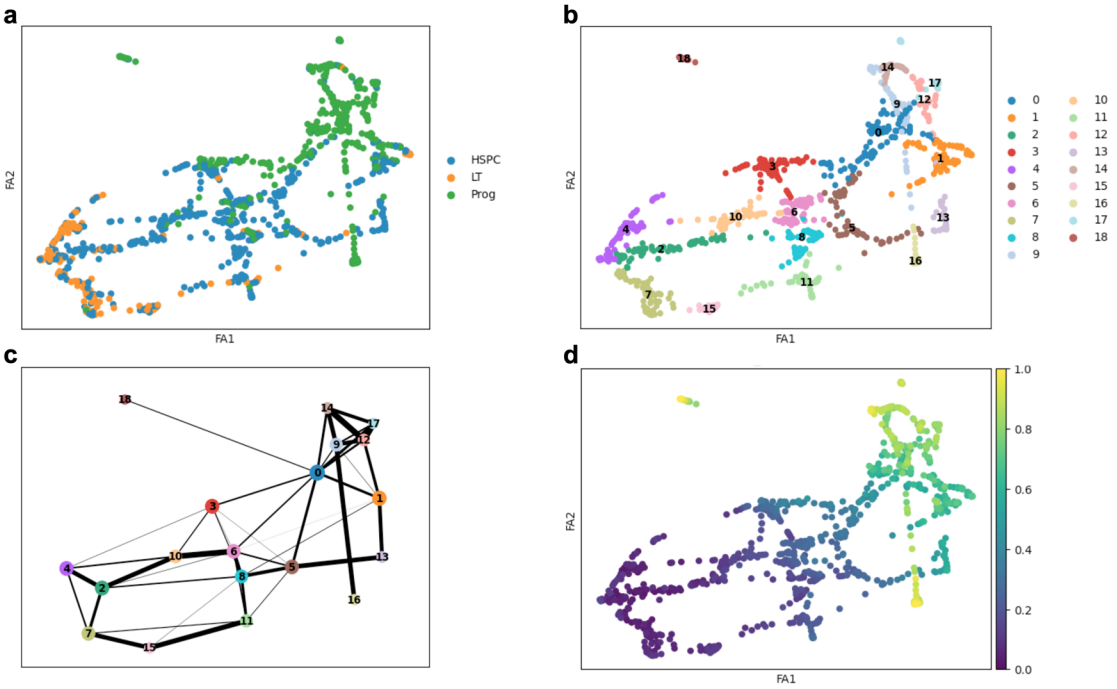

Figure 21: **PAGA+DPT applied on mHSC-GM** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell types. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from LT type.

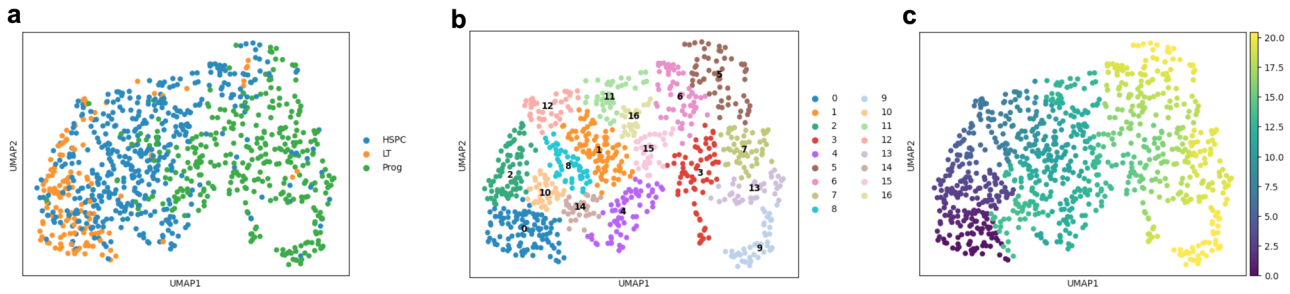

Figure 22: **Slingshot applied on mHSC-GM** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell types. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 0.

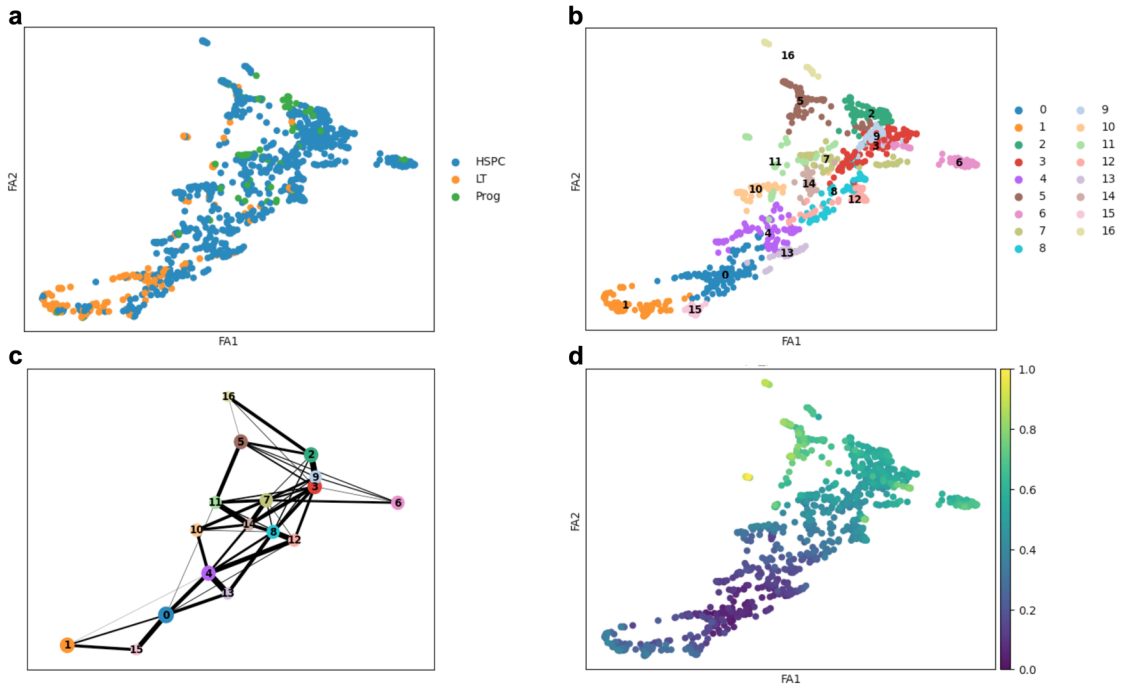

Figure 23: **PAGA+DPT applied on mHSC-L** All subfigures are plotted in diffusion map components. a) Diffusion map of dataset colored by cell types. b) Louvain clusters (resolution 1) found on diffusion map. c) PAGA graph computed for Louvain communities. d) Pseudotime ordering of cells with DPT, when given a root cell from LT type.

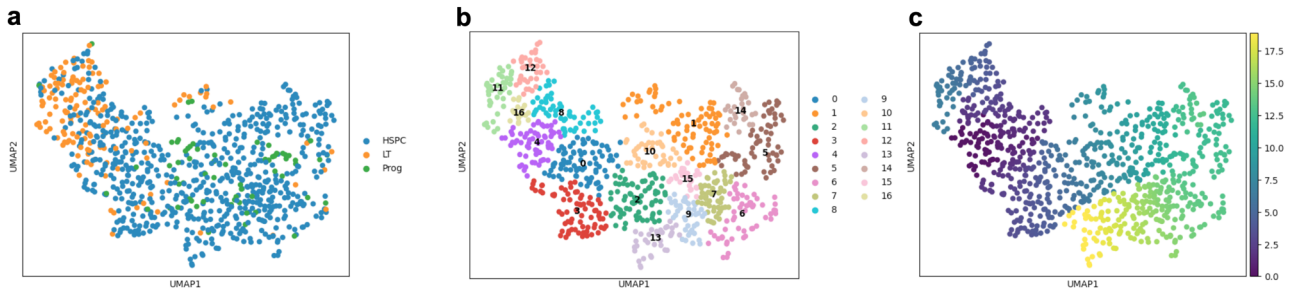

Figure 24: **Slingshot applied on mHSC-L** All subfigures are plotted in UMAP components. a) UMAP of dataset colored by cell types. b) Louvain clusters (resolution 1) found on UMAP. c) Pseudotime ordering of cells with Slingshot, when given start cluster as Cluster 11.

5.5 Specification of parameter settings for reproducibility

Data set	Top percentage kept	Louvain resolution
hESC	25%	2.2
hHep	25%	1.9
mESC	25%	1.5
mDC	40%	1
mHSC-L	25%	1
mHSC-GM	25%	0.9
mHSC-E	25%	1

Table 3: BIRDccNEST parameter settings used on BEELINE datasets

All results for BIRDccNEST can be reproduced by the provided code at https://bcb.cs.tufts.edu/BIRDccNEST.html , that was written in Python version 3.12.11. The code was run in environment with packages numpy 2.0.2, pandas 2.2.2, networkx 3.5, scanpy 1.11.4, and pyvis 0.3.2.

For cell-cell network inference RegDiffusion 0.1.1 was utilized. For each dataset, we used the default training parameters for `RegDiffusionTrainer` as listed in paper and code documentation.

The only two parameters that we set explicitly in the BIRDccNEST framework on a dataset basis, is the top percentage to prune for the cell-cell network and the Louvain resolution parameter when finding communities. Table 3 lists the values set each parameter for each dataset.