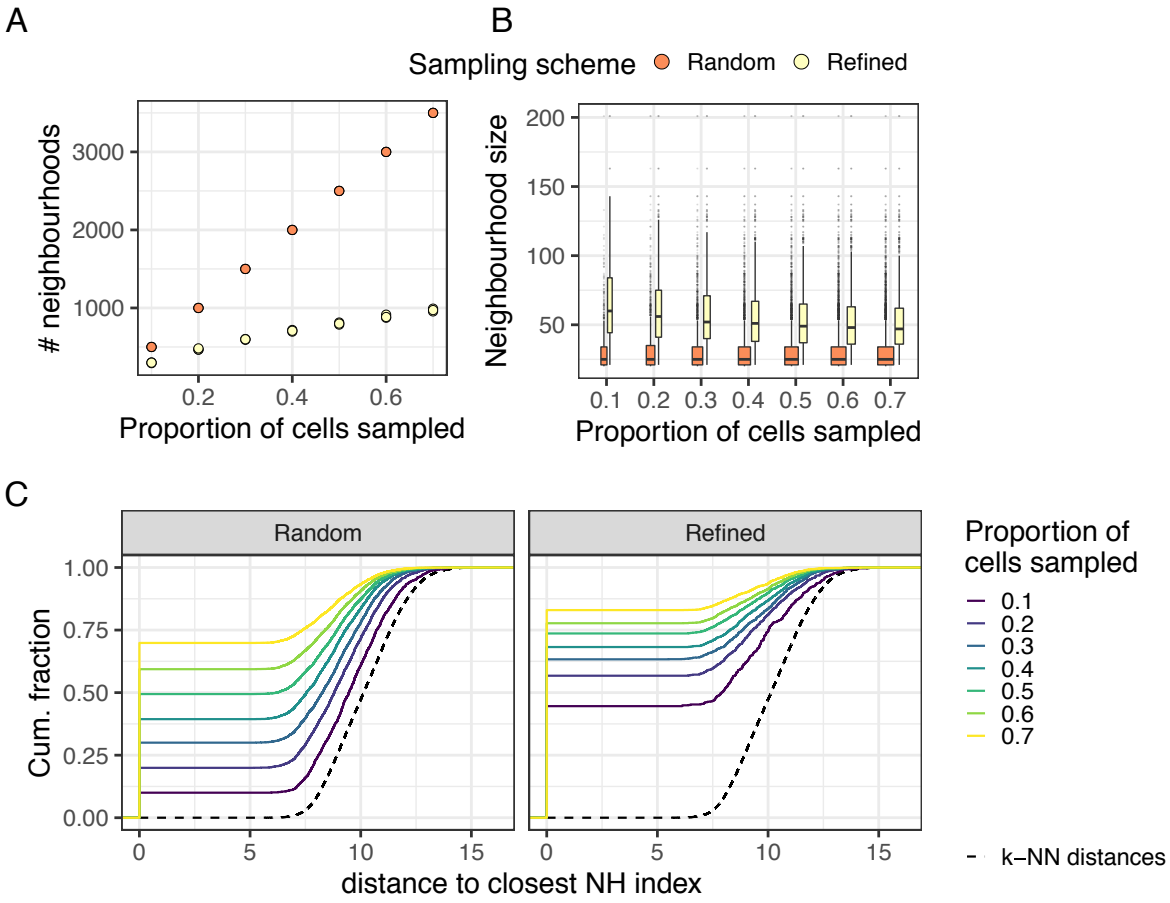
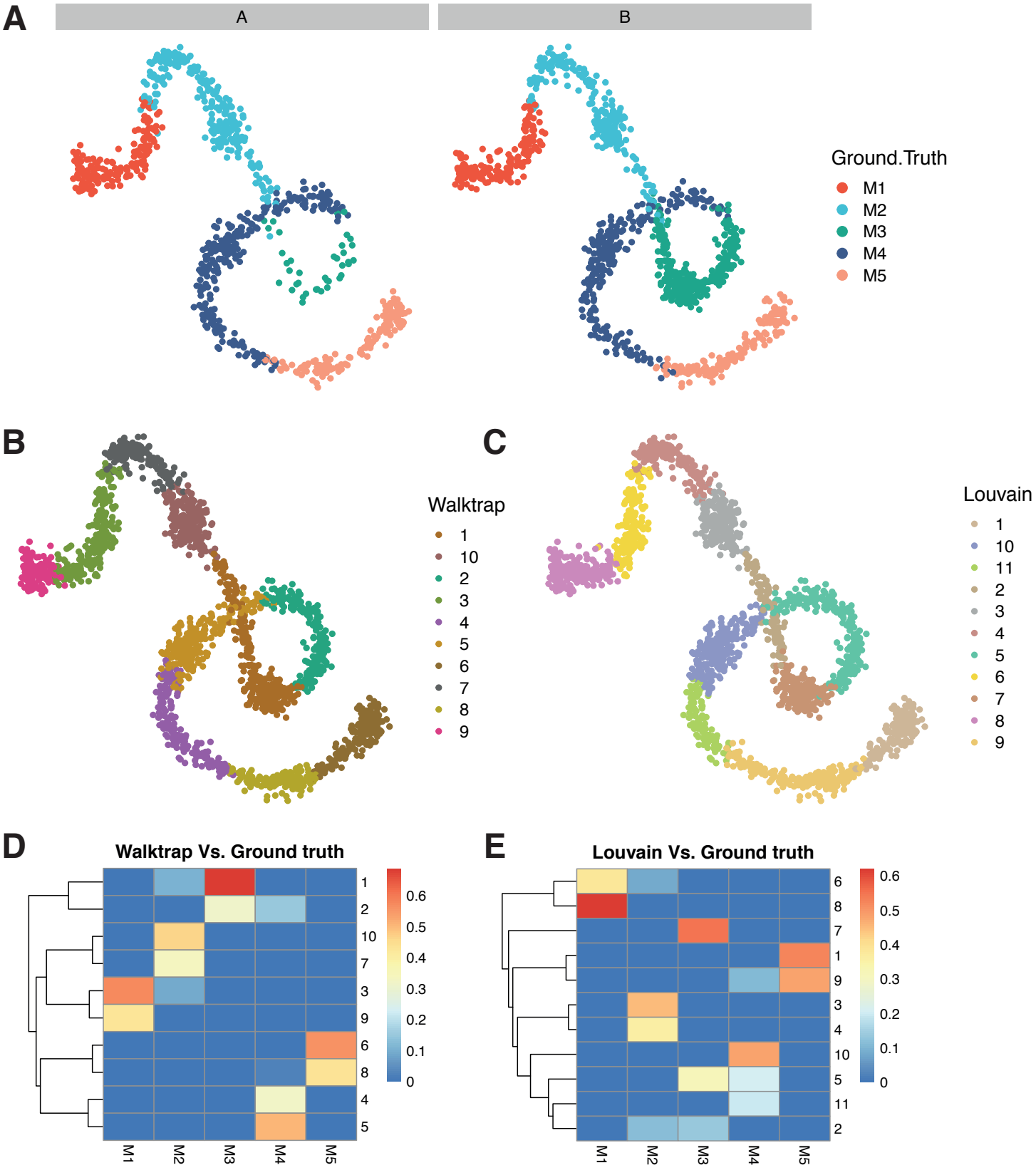
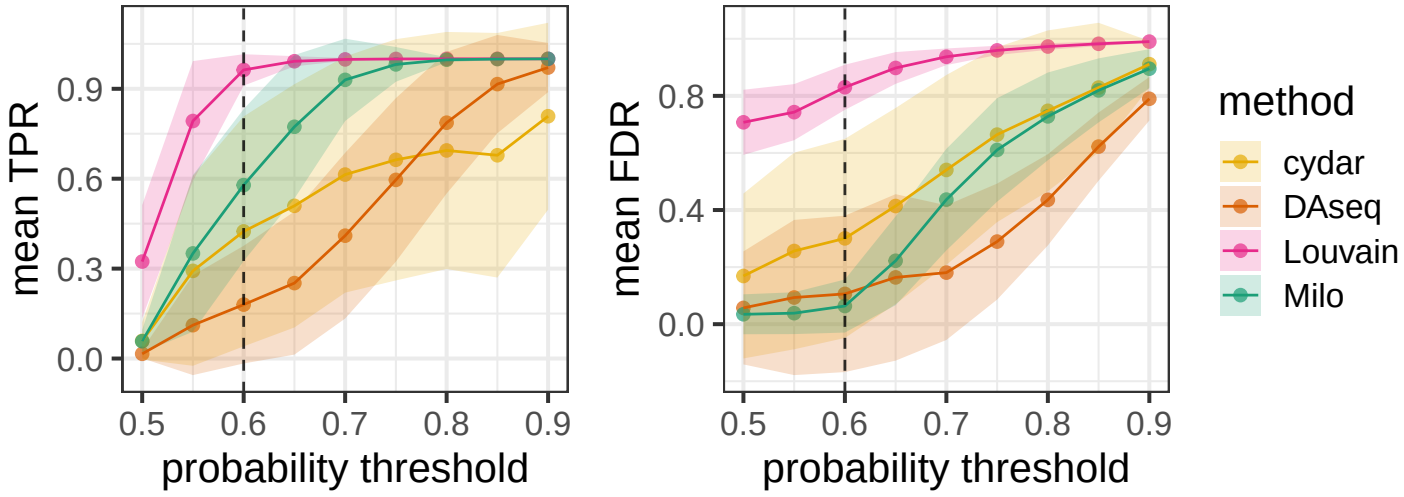
Supplementary Figures



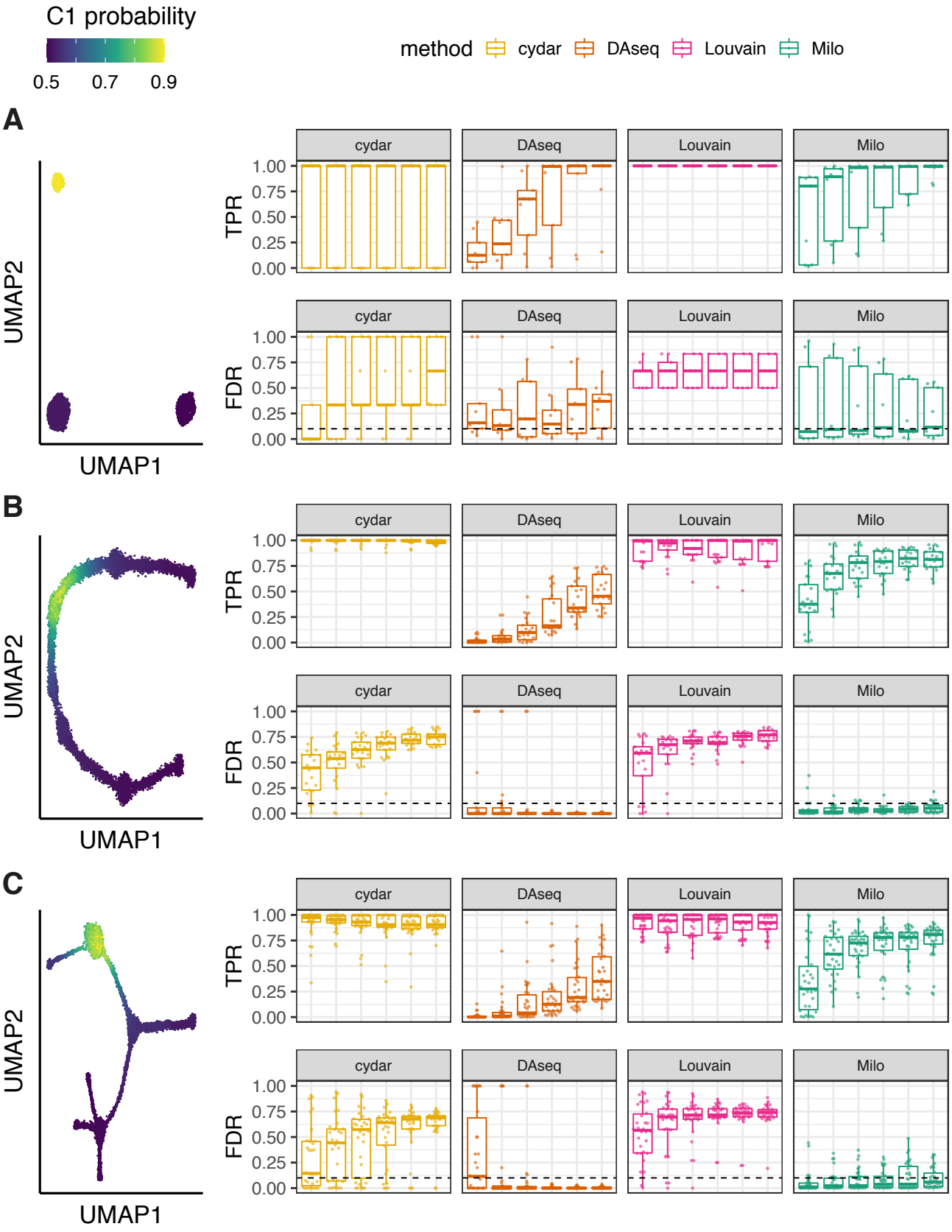
Supplementary Figure 1: **Random sampling of KNN graph vertices is suboptimal compared to sampling with refinement.** (A) Sampling with refinement leads to selection of fewer neighbourhoods (B) Sampling with refinement leads to selection of bigger neighbourhoods for DA testing, independently of the initial proportion of cells sampled (C) Sampling with refinement generates robust neighbourhoods across initializations: for each index cell we calculate the distance from the closest index in a sampling with different initialization. The cumulative distribution of distances to the closest index is shown. The black dotted line denotes the distribution of distances between K nearest neighbors in the dataset (K=30) (NH: neighbourhood). Neighbourhood statistics were calculated using a simulated trajectory dataset of 5000 cells. All plots show results from three sampling initializations for each proportion.



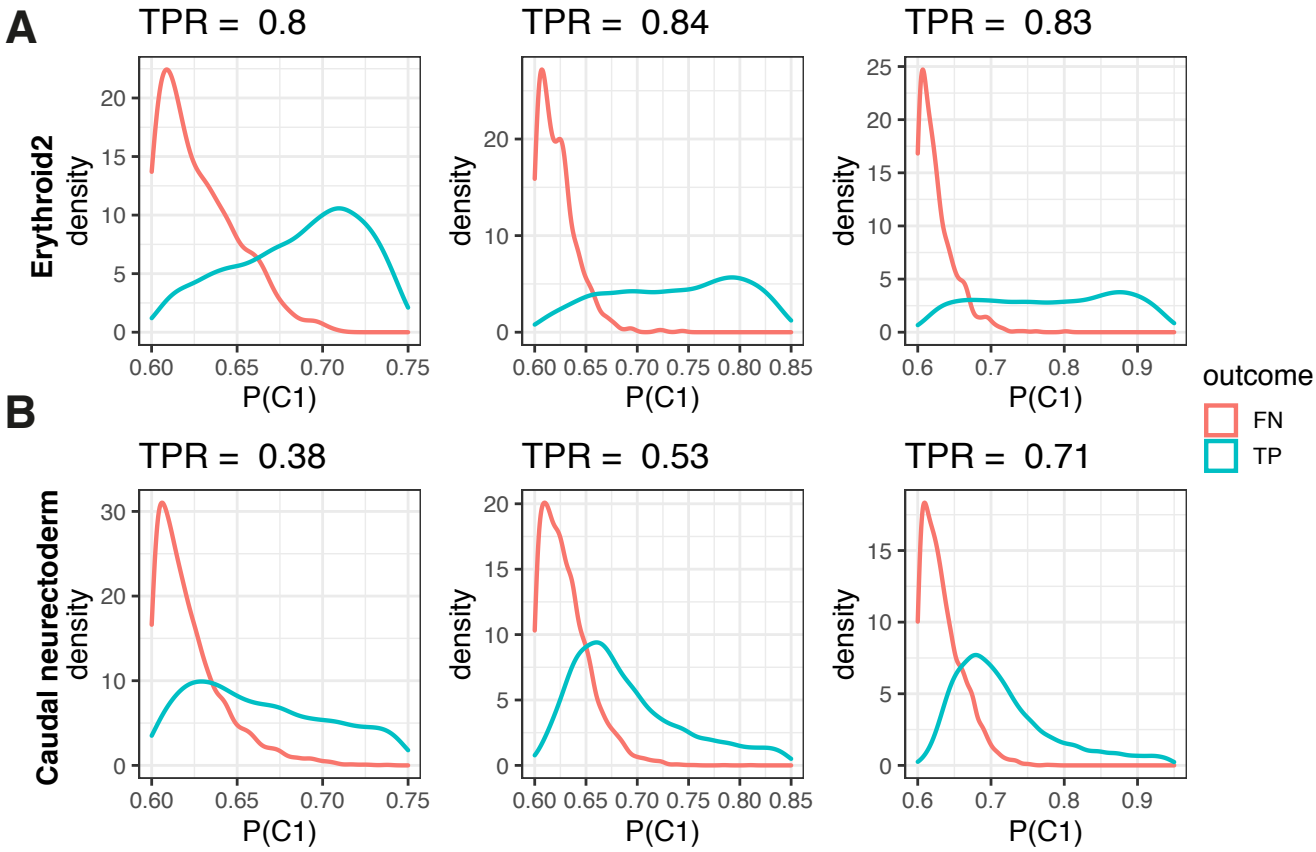
Supplementary Figure 2: **Graph-clustering does not faithfully capture simulated groups and differentially abundant subpopulations in a simulated continuous trajectory.** (A) A simulated linear trajectory of 2000 single-cells generated from 5 different groups, with cells assigned to either condition 'A' (left) or condition 'B' (right). (B) A Walktrap clustering of the data in (A) using the same KNN graph. Cells are coloured by Walktrap cluster identity. (C) A Louvain clustering of the data in (A) using the same KNN graph. Cells are coloured by the Louvain clustering identity. (D-E) Heatmaps comparing the numbers of cells in each cluster with respect to the ground truth groups in (A). Each cell is coloured by the proportion of cells from the column groups (ground truth) that are assigned to the respective cluster.



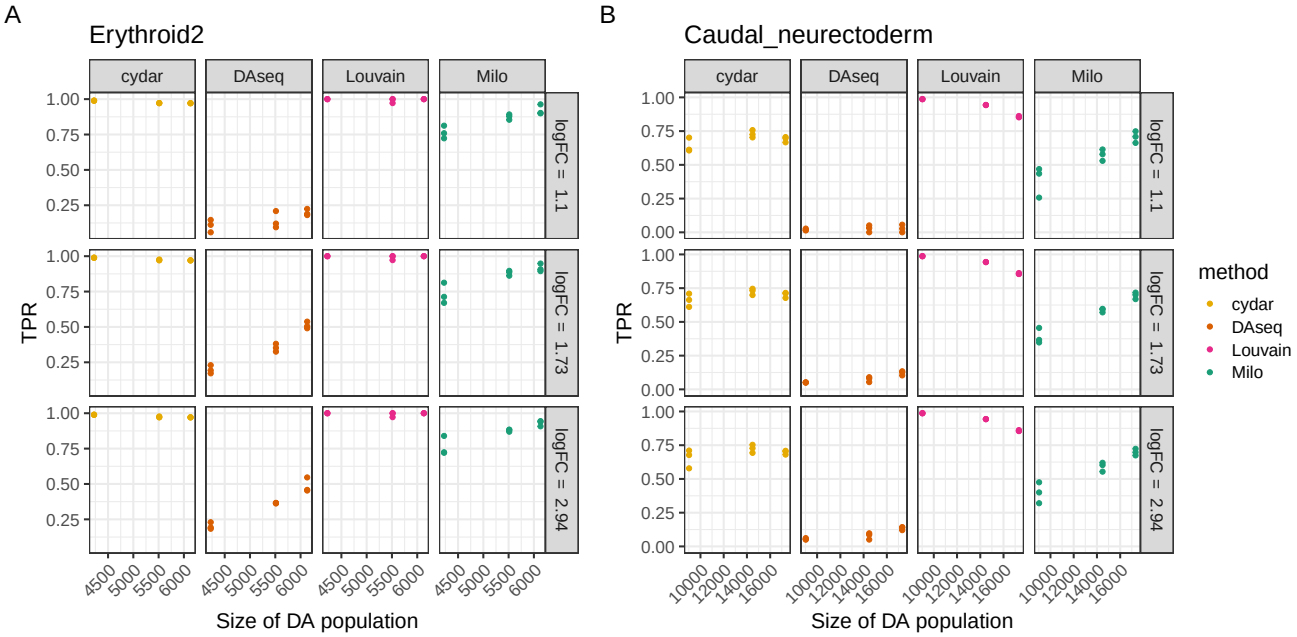
Supplementary Figure 3: **Selection of probability threshold for DA benchmarking.** Mean True Positive Rate (left) and False Discovery Rate (right) for recovery of cells in simulated DA regions as a function of probability threshold t picked to define true DA. The dashed line indicates $t = 0.6$, that was selected for benchmarking analyses. The mean is calculated over simulations on 8 populations. Line shading indicates the standard deviation of the mean.



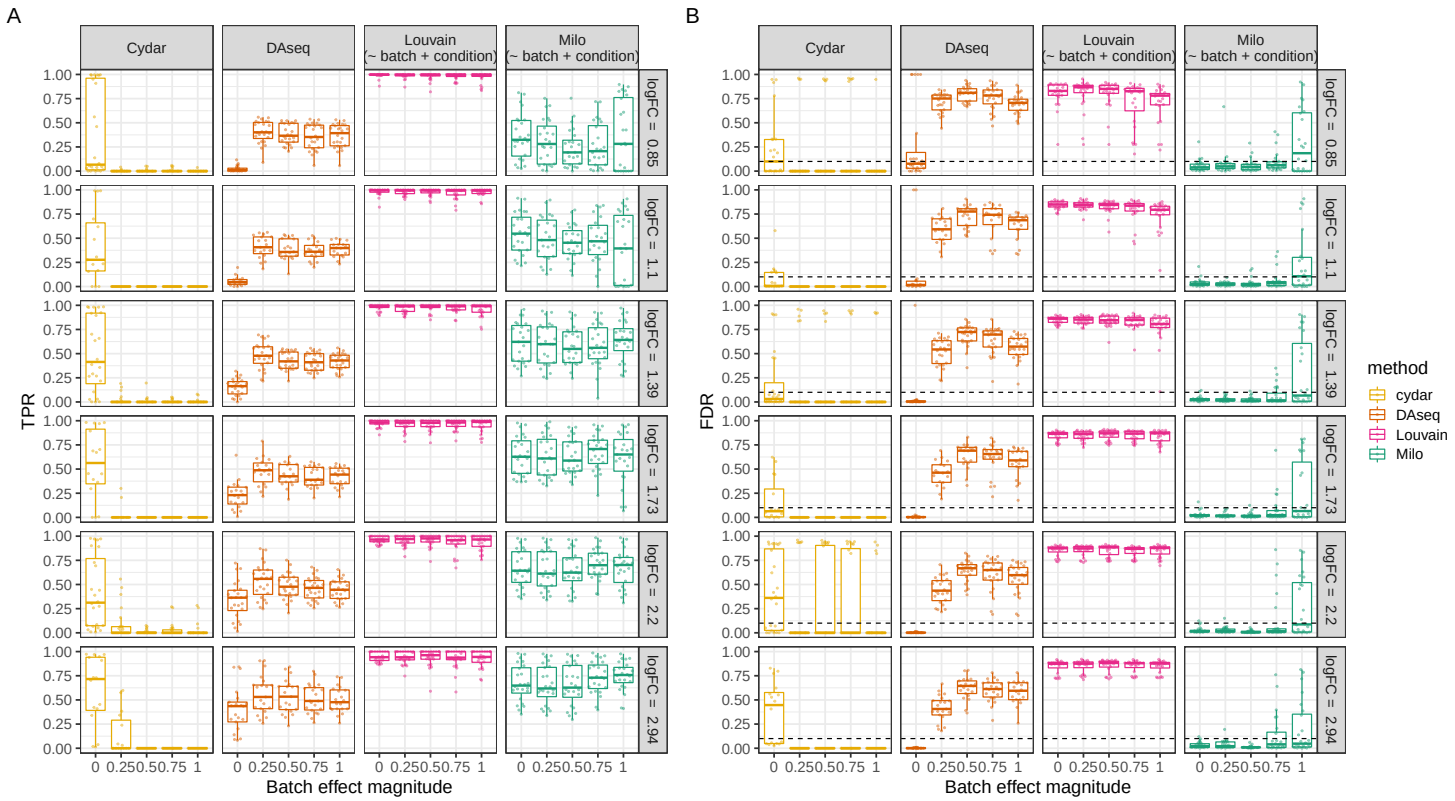
Supplementary Figure 4: **Benchmarking DA methods on simulated data.** DA analysis performance on KNN graphs from simulated datasets of different topologies: (A) discrete clusters (1800 cells, 3 populations); (B) 1-D linear trajectory (7500 cells, 7 populations); (C) Branching trajectory (7500 cells, 10 populations).



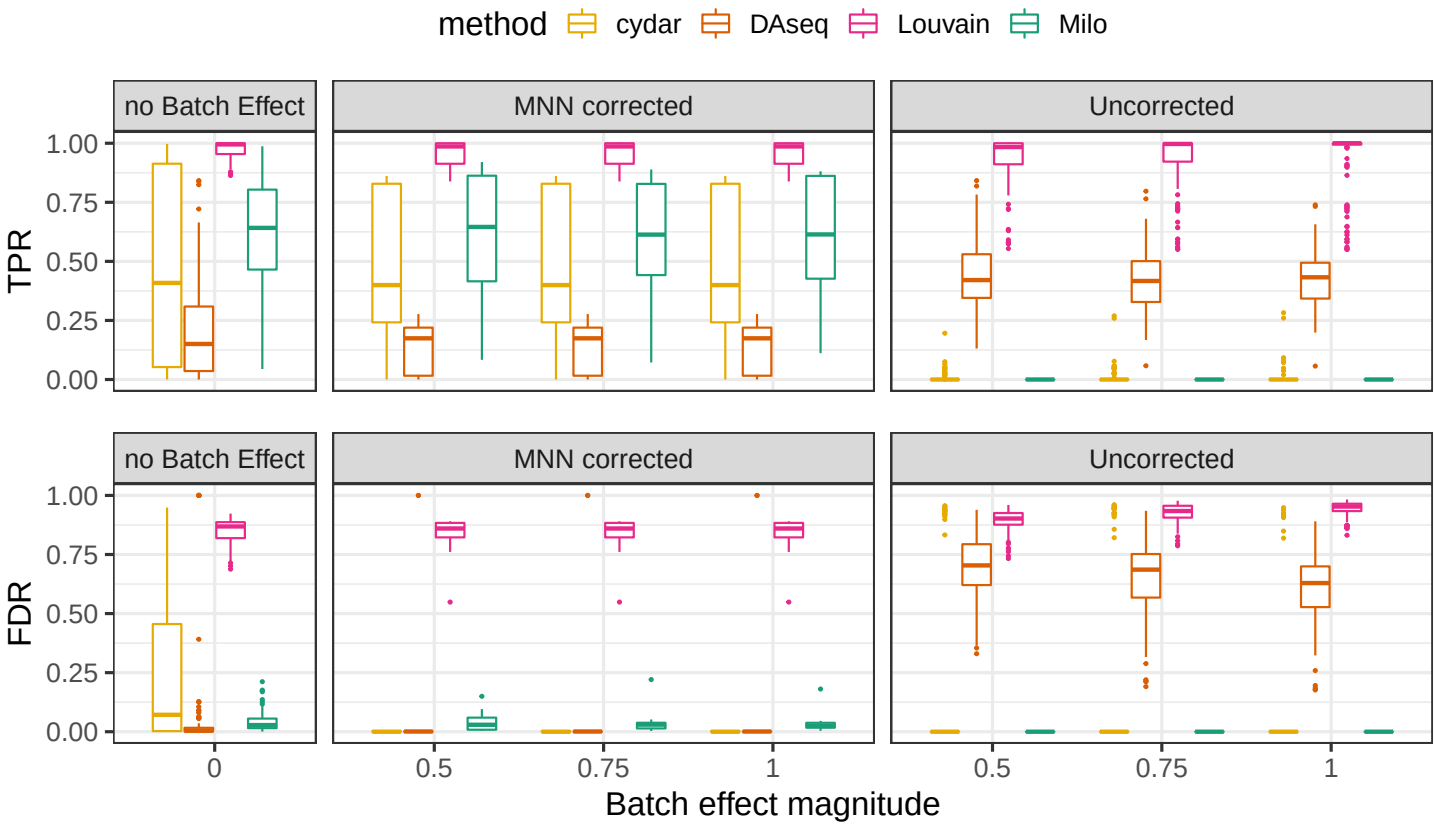
Supplementary Figure 5: **Variability in Milo power is explained by the fraction of true positive cells close to DA threshold** example distributions of underlying probability of Condition1 (P(C1)) for cells detected as True Positives (TP) or False Negatives (FN) by Milo. Examples for simulations on 2 populations (rows) and 3 simulated fold changes (columns) are shown.



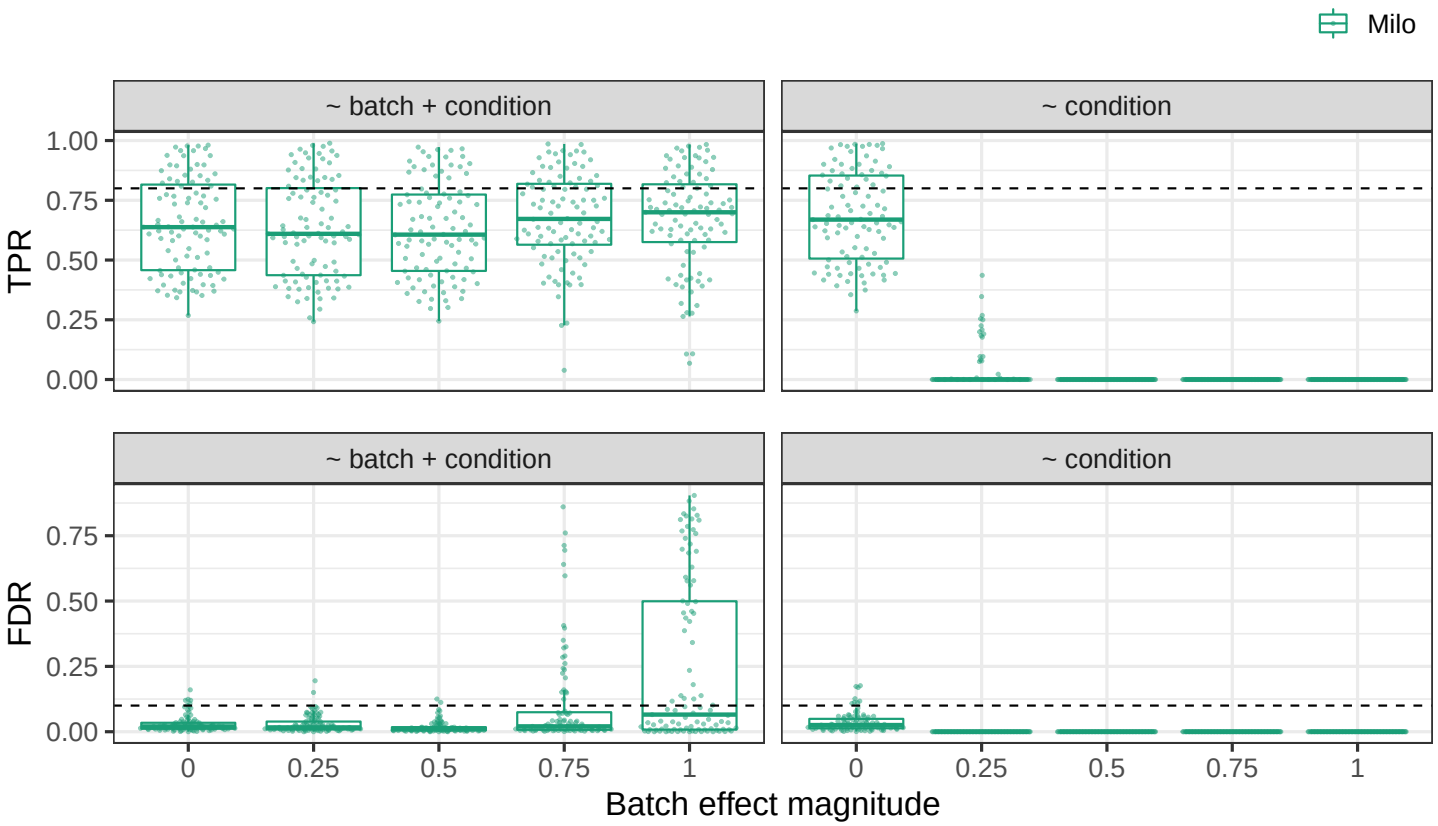
Supplementary Figure 6: **DA testing power increases with the size of the DA population** True Positive Rate (TPR) of DA detection for simulated DA regions of increasing size centered at the same centroid (Erythroid2 (A) and Caudal neuroectoderm (B)). Results for 3 condition simulations per population and fold change are shown.



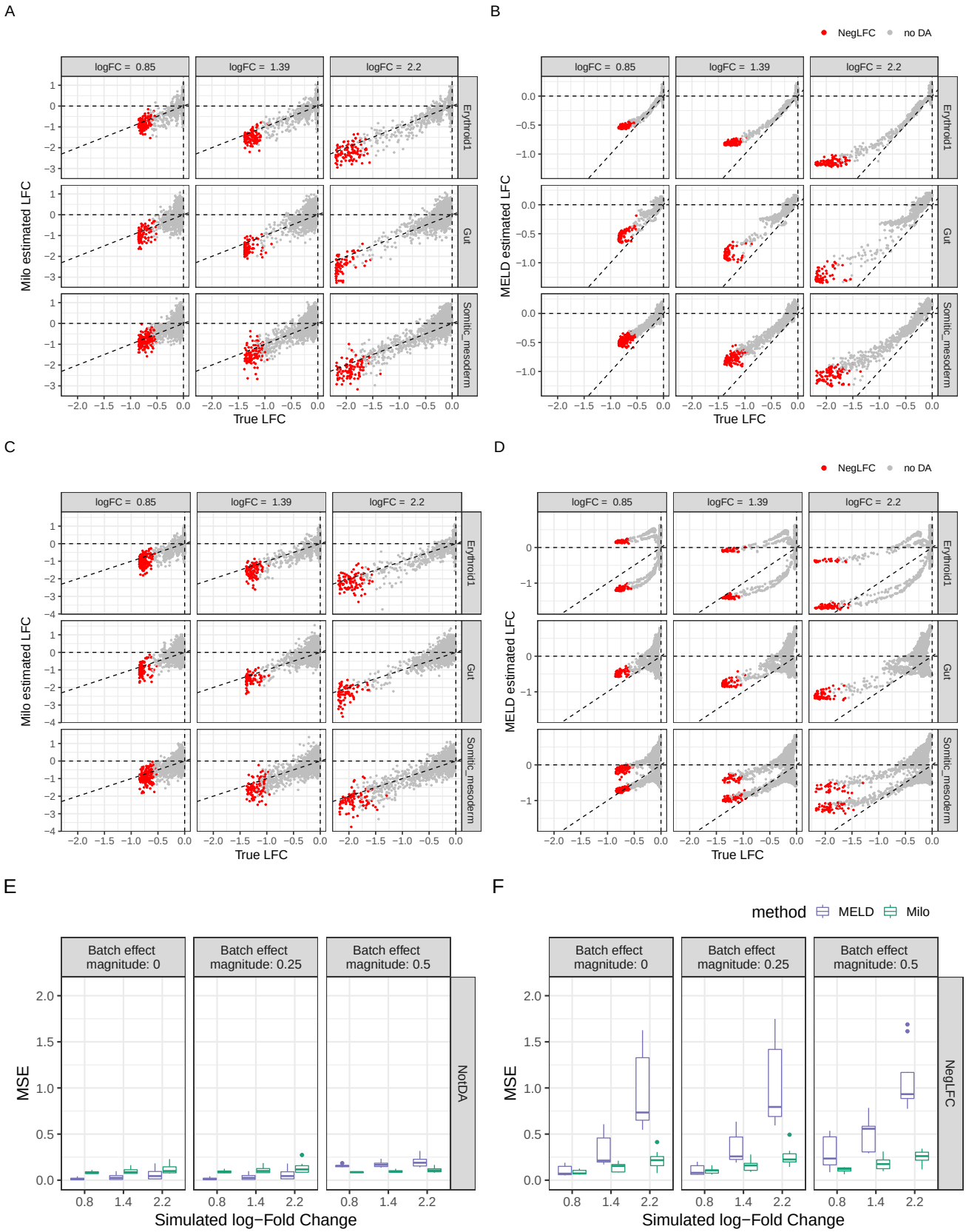
Supplementary Figure 7: **Batch effect control across DA effect sizes.** True Positive Rate (TPR, left) and False Discovery Rate (FDR, right) for recovery of cells in simulated DA regions for DA populations with increasing batch effect magnitude on the mouse gastrulation dataset. For each boxplot, results from 8 populations and 3 condition simulations per population are shown. Each panel represents a different DA method and a different simulated log-Fold Change.



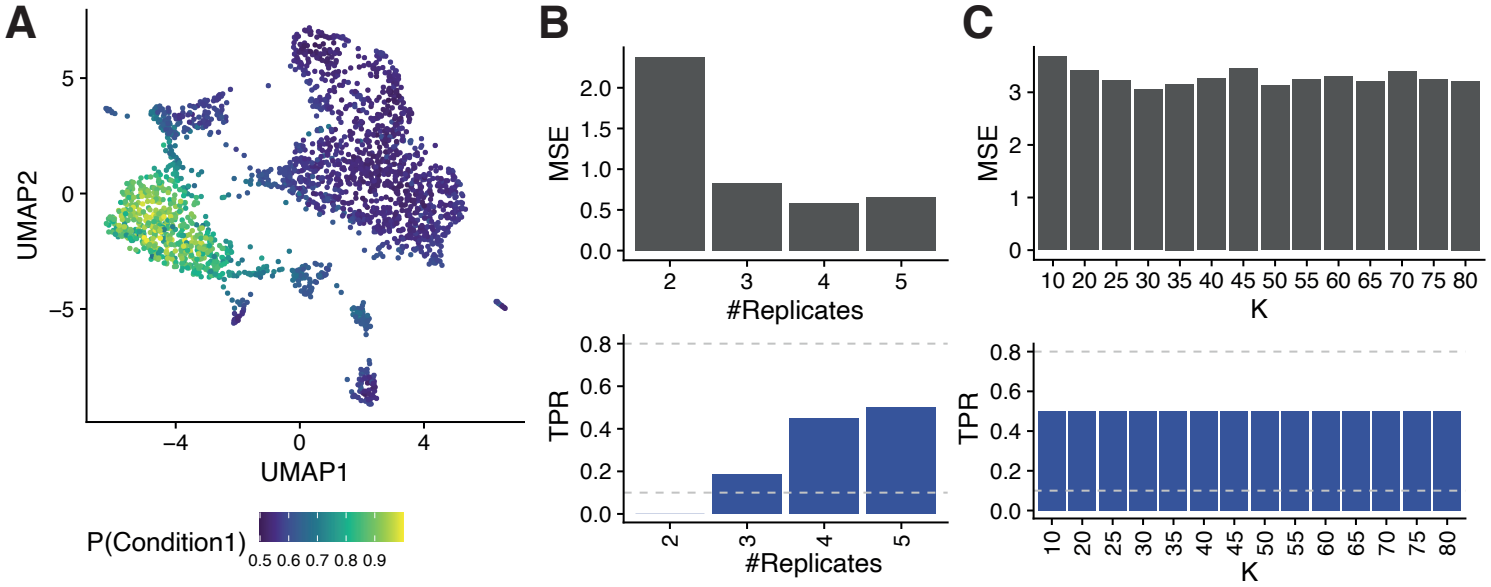
Supplementary Figure 8: **In silico batch correction enhances the performance of DA methods in the presence of batch effects.** Comparison of performance of DA methods with no batch effect, with batch effects of increasing magnitude corrected with MNN, and uncorrected batch effects.



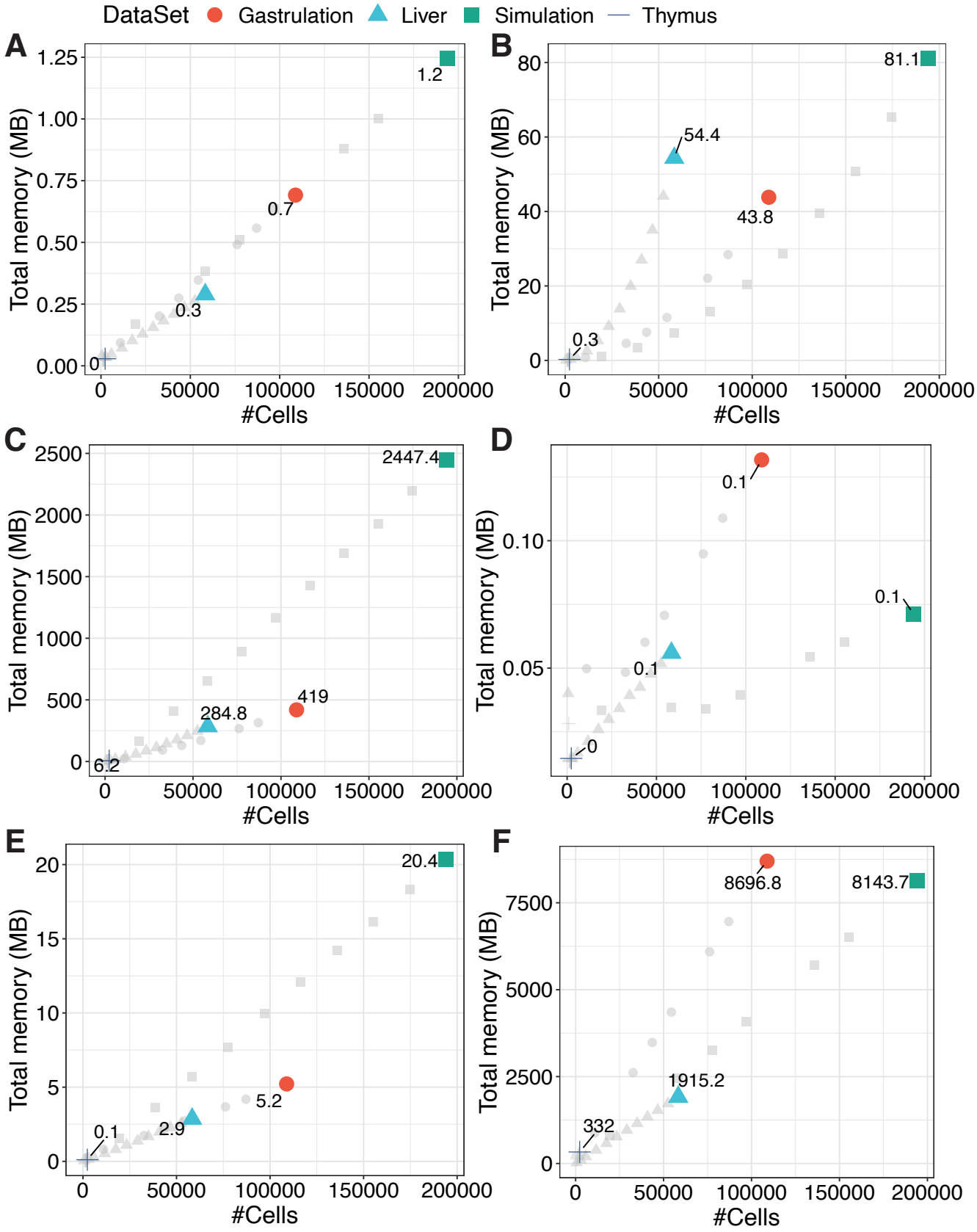
Supplementary Figure 9: **Modelling batch effects in DA analysis with Milo.** (B) Comparison of Milo performance with (~ batch + condition) or without (~ condition) accounting for the simulated batch in the GLM. For each boxplot, results from 8 populations, simulated fold change > 1.5 and 3 condition simulations per population and fold change are shown (72 simulations per boxplot).



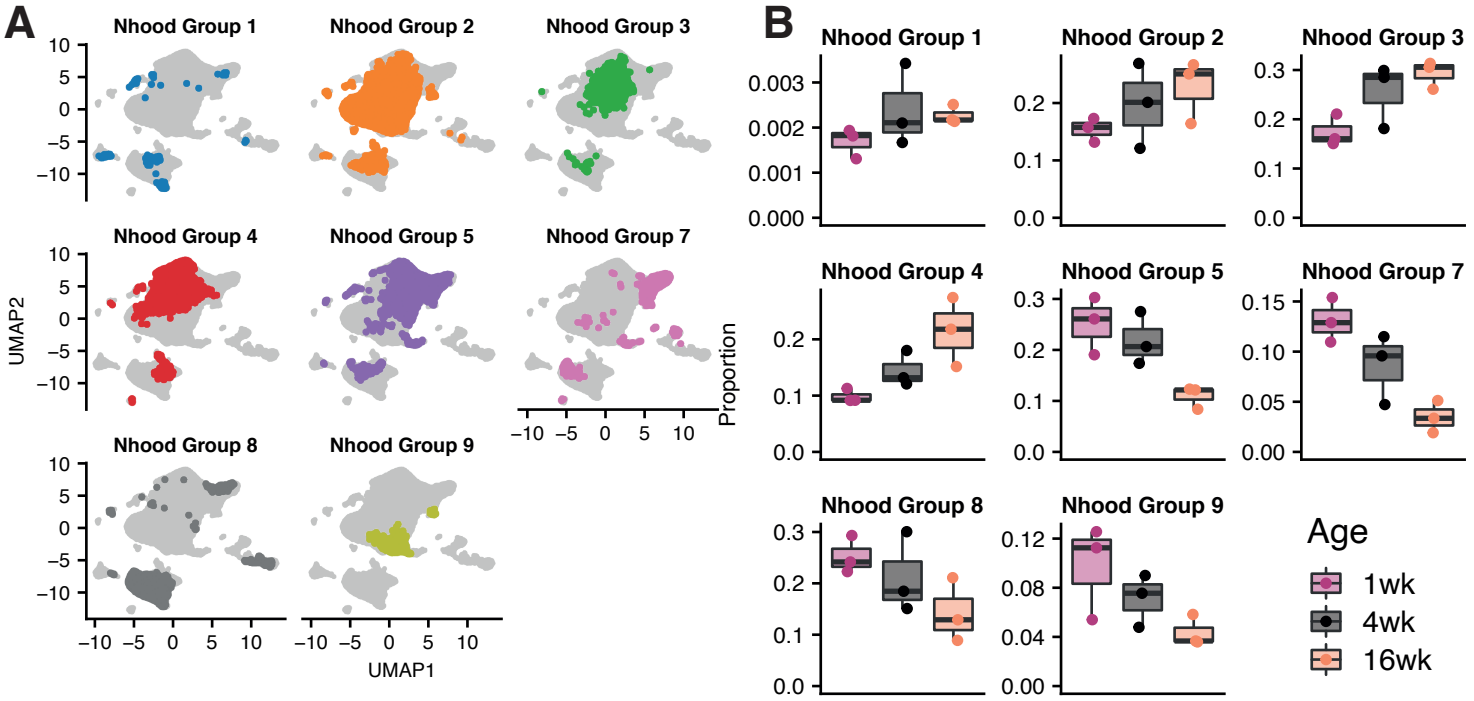
Supplementary Figure 10: **Comparison of Milo and MELD for abundance fold-change estimation.** (A-D) Scatter-plots of the true fold-change at the neighbourhood index against the fold-change estimated by Milo (A,C) and MELD (B,D), without batch effect (A-B) and with batch effect (magnitude = 0.5) (C-D), where $LFC = log(p_{c'}/(1 - p_{c'}))$. The neighbourhoods overlapping true DA cells ($p_{c'} > 0.6$) are highlighted in red. (C-D) Mean Squared Error (MSE) comparison for MELD and Milo for true negative neighbourhood (C) and true positive neighbourhoods (D), with increasing simulated log-Fold Change and magnitude of batch effect.



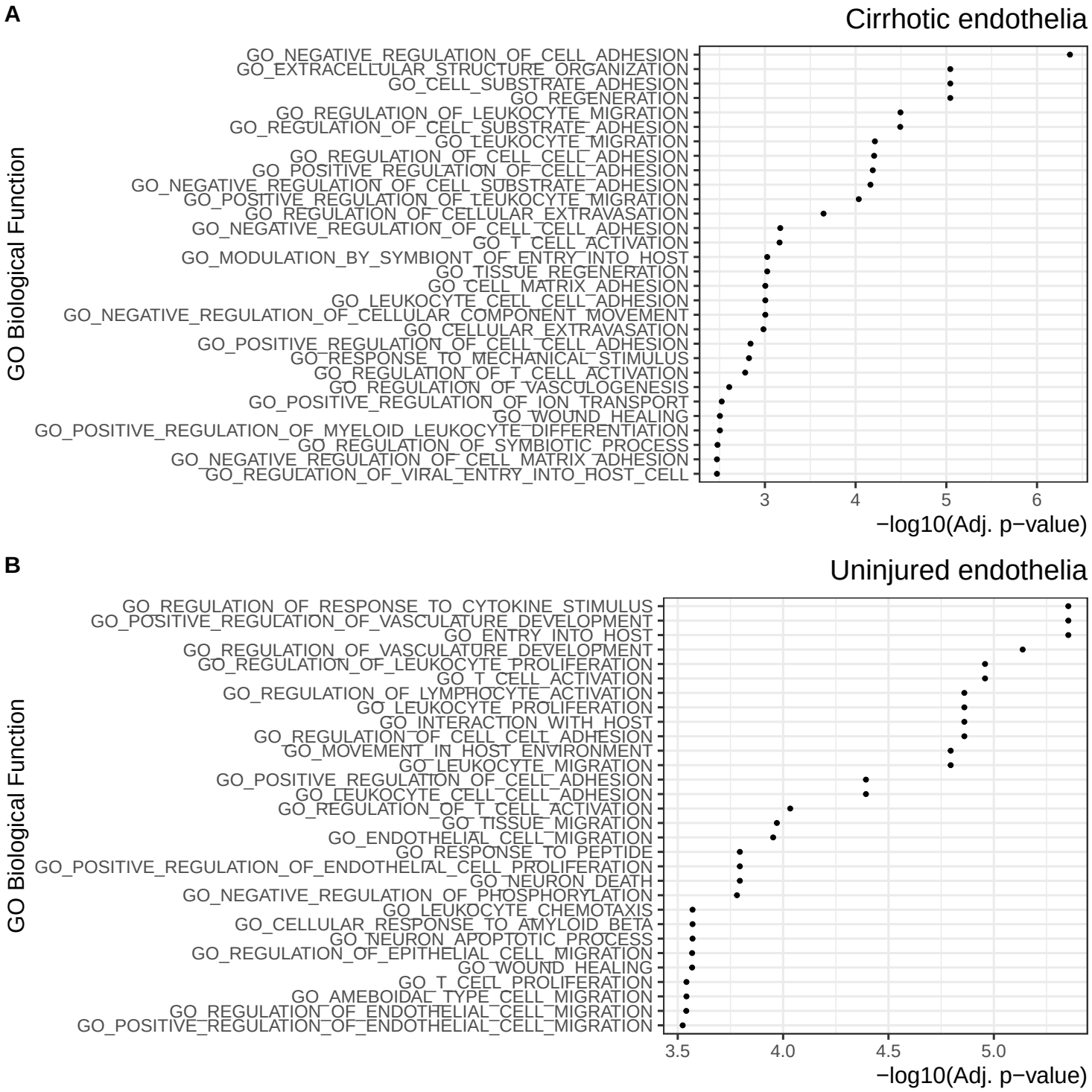
Supplementary Figure 11: **The impact of replication and k selection on effect size estimation variance.** (A) A UMAP of the mouse thymus data with a single simulated DA. Points are single cells coloured by the P(Condition1). (B) Increasing the number of replicates reduces the difference between the true simulated and estimated effect sizes, using the mean squared error (MSE; top panel) and increases the testing true positive rate (TPR; bottom panel). (C) Increasing k marginally reduces the estimation variance (top panel), and has less of an impact on power (bottom panel) compared to increased replication.



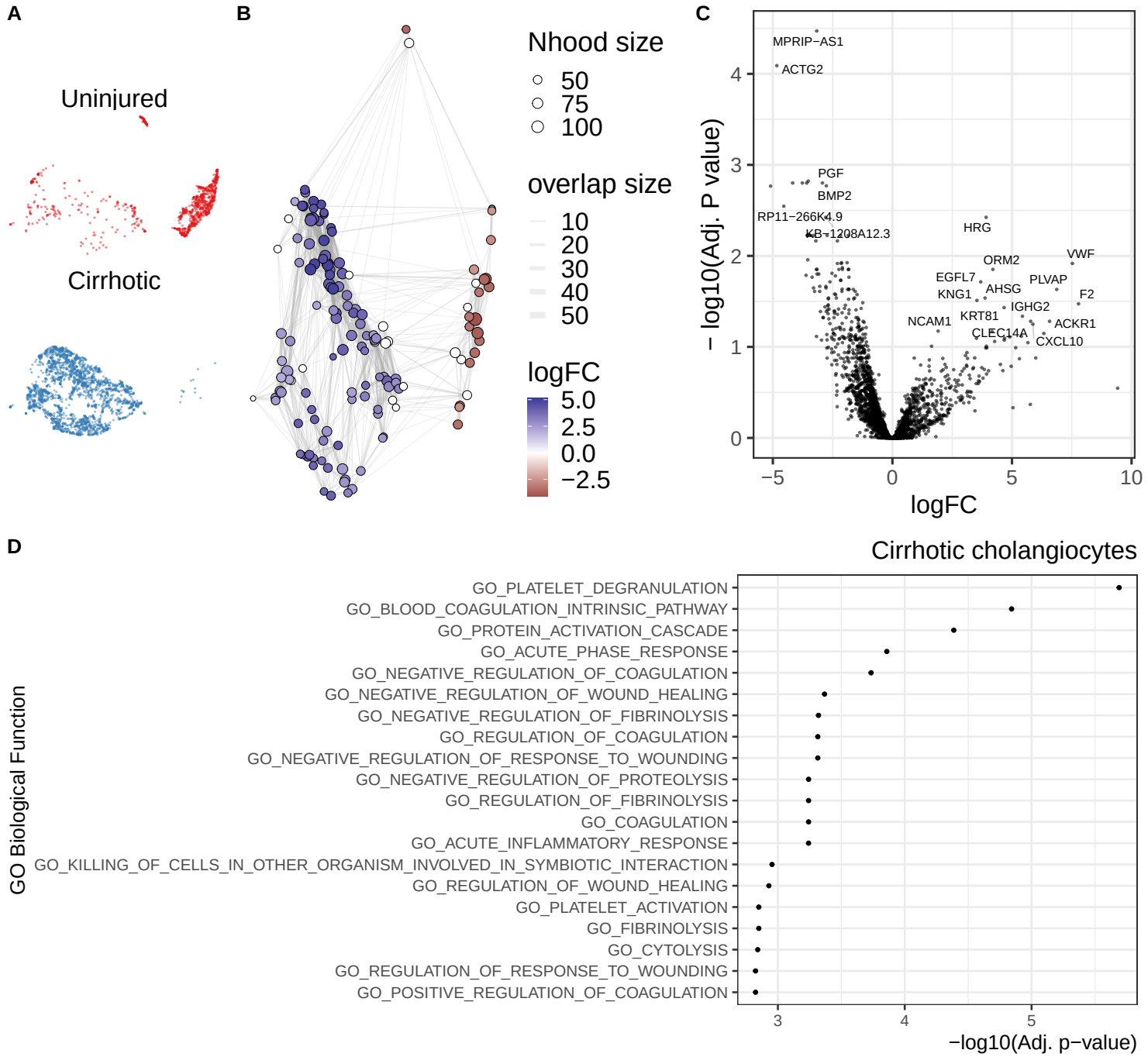
Supplementary Figure 12: **Memory usage across the Milo analysis workflow.** Total memory usage across the steps of the Milo analysis workflow in 4 datasets containing different numbers of cells (Gastrulation: circles, Liver: triangles, Thymus: crosses, Simulation: squares). Grey pointed denot down-sampled datasets of the corresponding type. Coloured points denote the total number of cells for the respective dataset. Total memory usage (y-axis) is shown in megabytes (MB). (A) KNN graph building, (B) neighbourhood sampling and construction, (C) within-neighbourhood distance calculation, (D) cell counting in neighbourhoods according to the input experimental design, (E) differential abundance testing, (F) total in memory R object size. A fixed value was used in all datasets for graph building and neighbourhood construction (k=30).



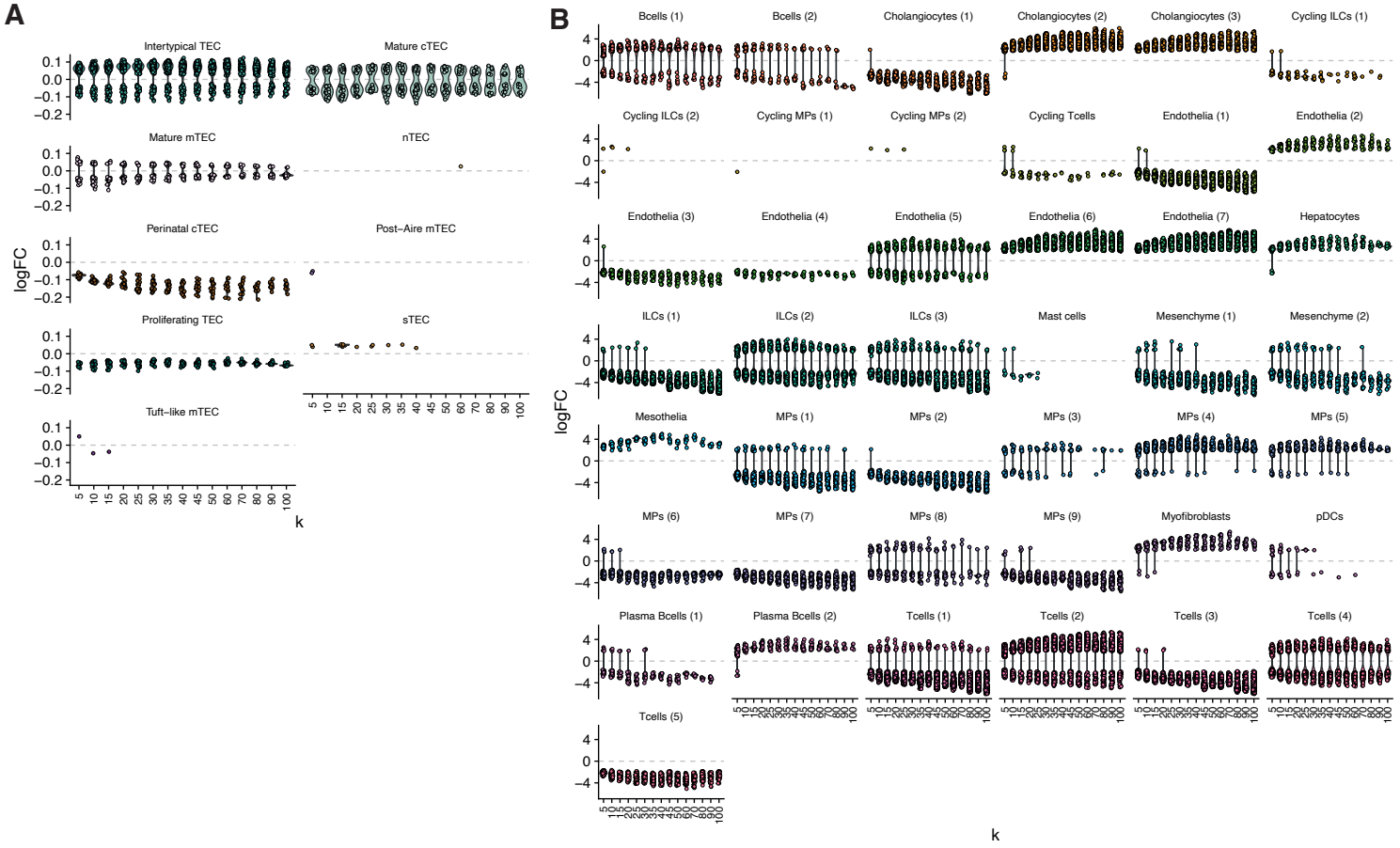
Supplementary Figure 13: **Label transferred neighbourhood groups onto droplet scRNA-seq cells.** (A) Joint UMAP embedding for SMART-seq and droplet scRNA-seq datasets, points are coloured by label-transferred neighbourhood groups for the droplet scRNA-seq cells. (B) Proportions of label-transferred neighbourhood groups across mouse ages (n=3 replicates per age), corresponding to (A).



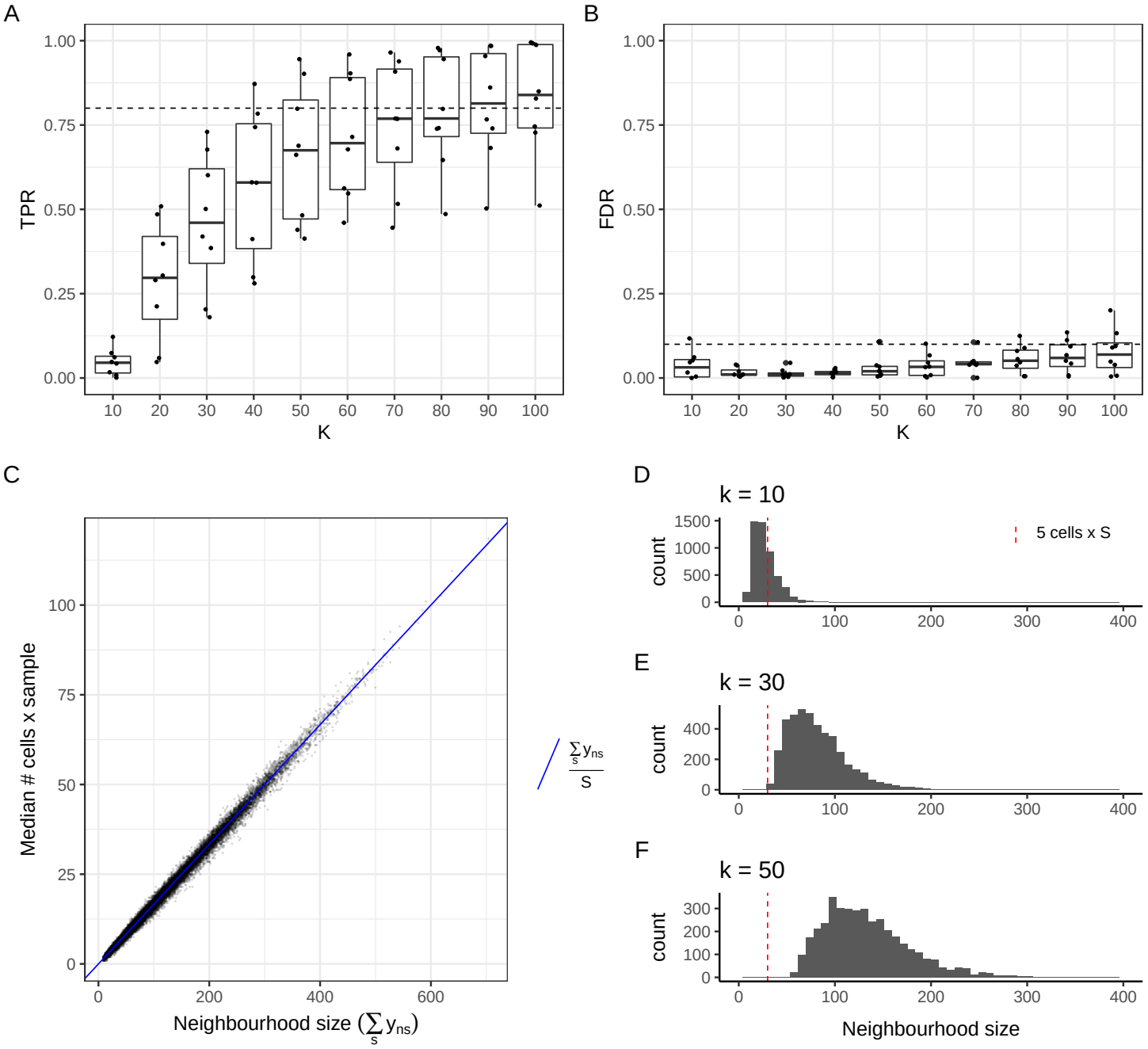
Supplementary Figure 14: **Downstream analysis of disease-specific endothelial subpopulations in liver cirrhosis** (A) GO term enrichment analysis on marker genes of cirrhosis-enriched endothelia. (B) GO term enrichment analysis on marker genes of healthy-enriched endothelia. The top 30 significant terms are shown.



Supplementary Figure 15: **Downstream analysis of disease-specific cholangiocyte subpopulations in liver cirrhosis** Downstream analysis of disease-specific cholangiocyte subpopulations in liver cirrhosis (A-B) UMAP embedding (A) and graph representation (B) of neighbourhoods of 3369 cells from cholangiocyte lineage. (C) Volcano plot for DGE test on cholangiocytes DA subpopulations: the x-axis shows the log-fold change between expression in cirrhotic and healthy cholangiocytes. The y-axis shows the adjusted p-value. (D) GO term enrichment analysis on marker genes of cirrhosis-enriched cholangiocytes. The top 20 significant terms are shown.



Supplementary Figure 16: **Robustness of Milo DA testing to varying K**. Distributions of DA neighbourhoods across values of K for the mouse ageing thymus (A) and human cirrhotic liver (B) data sets. Shown are the distributions of log fold-changes (y-axis) for DA (FDR 10%) neighbourhoods using different values of K (x-axis) from 5-100, illustrating that DA testing is robust across a broad range of values of K.



Supplementary Figure 17: **Selection of K parameter** (A-B) Example trends for TPR and FDR for increasing values of K used for KNN graph building on simulated DA on 8 regions ($P(C1) = 0.8$). Dotted lines highlight TPR=0.8 and FDR=0.1 thresholds. (C) The median number of cells per experimental sample is a function of the neighbourhood size $\sum_s n_s$ divided by the total number of samples S. (D-F) Histogram of neighbourhood sizes for different choices of K. The red dotted line denotes the minimum neighbourhood size to obtain 5 cells per sample on average.

Supplementary tables

	Milo	**MELD**	**DAseq**	**Cydar**	**Louvain + GLM**
Framework	R	python	R	R	R
Unit for DA estimate	KNN graph neighbourhood	Cell	Cell	PC hypersphere	Cluster
Hyperparameters	d	d	d	d	d
	k	k	minimum k	hypersphere radius	k
	prop		maximum k	downsampling fraction	(resolution)
			k step		
Clustering-free	yes	yes	yes	yes	no
Representative sampling across dataset	yes	yes	yes	no	yes
Output					
Effect size estimate	log-Fold change	condition likelihood	DA score	log-Fold change	log-Fold change
Statistical testing	yes	no	yes	yes	yes
Modelling variation between replicates	yes	no	no	yes	yes
Spatial FDR control	yes	no	no	yes	NA
DA testing experimental design					
two condition	yes	yes	yes	yes	yes
multi-condition	yes	yes	no	yes	yes
continuous condition	yes	no	no	yes	yes
nuisance covariate control	yes	no	no	yes	yes
interaction	yes	no	no	yes	yes

Supplementary Table 1: **Qualitative comparison of evaluated methods for DA analysis**

Parameter name	**Description**	**Range of tested values**			
KNN graph	Underlying k-NN graph, generated from simulated or real scRNA-seq datasets	Clusters [dyntoy]	1D trajectory [dyntoy]	Branching trajectory [dyntoy]	Mouse gastrulation atlas [Pijuan-Sala et al. 2019]
DA population	cell population selected as centroid for differential abundance region	M1	M1	M1	Caudal_neurectoderm
		M2	M2	M2	Erythroid2
		M3	M3	M3	Gut
			M4	M4	Somitic_mesoderm
			M5	M5	Pharyngeal_mesoderm
			M6	M6	Erythroid1
			M7	M7	Mesenchyme
				M8	ExE_endoderm
				M9	
				M10	
Logit parameter	Coefficient of logit transformation	0.5	0.5	0.5	0.5
Max C1 probability	maximum probability of Condition 1	0.75 - 0.95	0.75-0.95	0.75-0.95	0.75-0.95
Seed for label sampling	Random seed for sampling of condition labels and assignment to replicates	43, 44, 45	43, 44, 45	43, 44, 45	43, 44, 45
Batch effect magnitude	Standard deviation of gaussian vector added to all cells in the same batch	0	0	0	0, 0.25, 0.5, 0.75, 1
	Total # simulations	54	126	180	810

Supplementary Table 2: **Summary of parameters used for DA simulations**

Method	Parameters	Values - clusters	Values - 1D trajectory	Values - branching trajectory	Values - mouse gastrulation	Significance threshold
Milo	K	15	20	20	50	10% FDR
	d	30	30	30	30	
	prop	0.1	0.1	0.1	0.1	
MELD	K	NA	NA	NA	50	NA
	d	NA	NA	NA	30	
Cydar	tol	0.8	6	6	1	10% FDR
	downsample	3	3	3	3	
	d	30	30	30	30	
DAseq	k.vector	15-500, steps of 50	20-500, steps of 50	20-500, steps of 50	50-500, steps of 50	DA score > permutation threshold (pred.thres = NULL)
	d	30	30	30	30	
Louvain + GLM	k	15	20	20	50	10% FDR
	d	30	30	30	30	

Supplementary Table 3: **Summary of parameters used for benchmarking of DA methods**