nature biotechnology

ARTICLES

https://doi.org/10.1038/s41587-021-01033-z

Differential abundance testing on single-cell data using *k*-nearest neighbor graphs

Emma Dann[1], Neil C. Henderson[2,3], Sarah A. Teichmann[1,4], Michael D. Morgan[5,6] ✉ and John C. Marioni[1,5,6] ✉

Current computational workflows for comparative analyses of single-cell datasets typically use discrete clusters as input when testing for differential abundance among experimental conditions. However, clusters do not always provide the appropriate resolution and cannot capture continuous trajectories. Here we present Milo, a scalable statistical framework that performs differential abundance testing by assigning cells to partially overlapping neighborhoods on a *k*-nearest neighbor graph. Using simulations and single-cell RNA sequencing (scRNA-seq) data, we show that Milo can identify perturbations that are obscured by discretizing cells into clusters, that it maintains false discovery rate control across batch effects and that it outperforms alternative differential abundance testing strategies. Milo identifies the decline of a fate-biased epithelial precursor in the aging mouse thymus and identifies perturbations to multiple lineages in human cirrhotic liver. As Milo is based on a cell–cell similarity structure, it might also be applicable to single-cell data other than scRNA-seq. Milo is provided as an open-source R software package at https://github.com/MarioniLab/miloR.

The advent and expansion of high-throughput and high-dimensional single-cell measurements has empowered the discovery of specific cell state changes associated with disease, development and experimental perturbations. Perturbed cell states can be detected by quantifying shifts in abundance of cell types in response to a biological insult. A common analytical approach for quantitatively identifying such shifts is to ask whether the composition of cells in pre-defined and discrete clusters differs among experimental conditions[1–5]. However, assigning single cells to discrete clusters can be problematic, especially in the context of continuous differentiation, developmental or stimulation trajectories, thus limiting the power and resolution of such differential abundance testing strategies.

Alternative approaches to perform differential abundance analysis without requiring clusters have been proposed for high-throughput mass cytometry data[6] or for scRNA-seq data[7,8]. However, these have different limitations, where they either do not model variability in cell numbers among replicate samples or are primarily designed for pairwise comparisons. This limits their application to datasets with more complex experimental designs, including continuous covariates (such as age or disease severity) and confounding sources of variation.

To solve these challenges, we developed a computational framework to perform differential abundance testing without relying on clustering cells into discrete groups. We make use of a common data structure that is embedded in many single-cell analyses: *k*-nearest neighbor (KNN) graphs. We model cellular states as overlapping neighborhoods on such a graph, which are then used as the basis for differential abundance testing. We account for the non-independence of overlapping neighborhoods by applying a weighted version of the Benjamini–Hochberg method, where *P* values are weighted by the reciprocal of the neighborhood connectivity—an adaptation to graphs of a previously described strategy to control the spatial false discovery rate (FDR)[6].

Our method, which we call Milo, leverages the flexibility of generalized linear models (GLMs), thus allowing complex experimental designs, such as the inclusion of nuisance and technical covariates, including accounting for samples ascertained from different batches. Moreover, by modeling cell states as overlapping neighborhoods, we are able to accurately pinpoint the perturbed cellular states, enabling the identification of the underlying molecular programs. We demonstrate the power of our approach by identifying perturbed cellular states from publicly available datasets in the context of human liver cirrhosis and by uncovering a fate-biased progenitor in the aging murine thymus. Furthermore, we demonstrate the speed and scalability of our open-source implementation of Milo and demonstrate its superiority to alternative approaches.

Results

Modeling cell states as neighborhoods on a KNN graph. We propose to model the differences in the abundance of cell states among experimental conditions using graph neighborhoods (Fig. 1). Our computational approach allows overlapping neighboring regions, which alleviates the principal pitfalls of using discrete clusters for differential abundance testing. We make use of a refined sampling implementation[9], which leads to high coverage of the graph while simultaneously controlling the number of neighborhoods that need to be tested. For each neighborhood, we then perform hypothesis testing among biological conditions to identify differentially abundant cell states while controlling the FDR across the graph neighborhoods.

Our method works on a KNN graph that represents the high-dimensional relationships among single cells, a common scaffold for many single-cell analyses[1–4] (Fig. 1a; practical guidance for

[1]Wellcome Sanger Institute, Wellcome Genome Campus, Hinxton, Cambridge, UK. [2]Centre for Inflammation Research, The Queen's Medical Research Institute, University of Edinburgh, Edinburgh, UK. [3]MRC Human Genetics Unit, Institute of Genetics and Molecular Medicine, University of Edinburgh, Edinburgh, UK. [4]Theory of Condensed Matter Group, The Cavendish Laboratory, University of Cambridge, Cambridge, UK. [5]European Molecular Biology Laboratory European Bioinformatics Institute, Hinxton, Cambridge, UK. [6]Cancer Research UK Cambridge Institute, Li Ka Shing Centre, University of Cambridge, Cambridge, UK. ✉e-mail: michael.morgan@cruk.cam.ac.uk; marioni@ebi.ac.uk

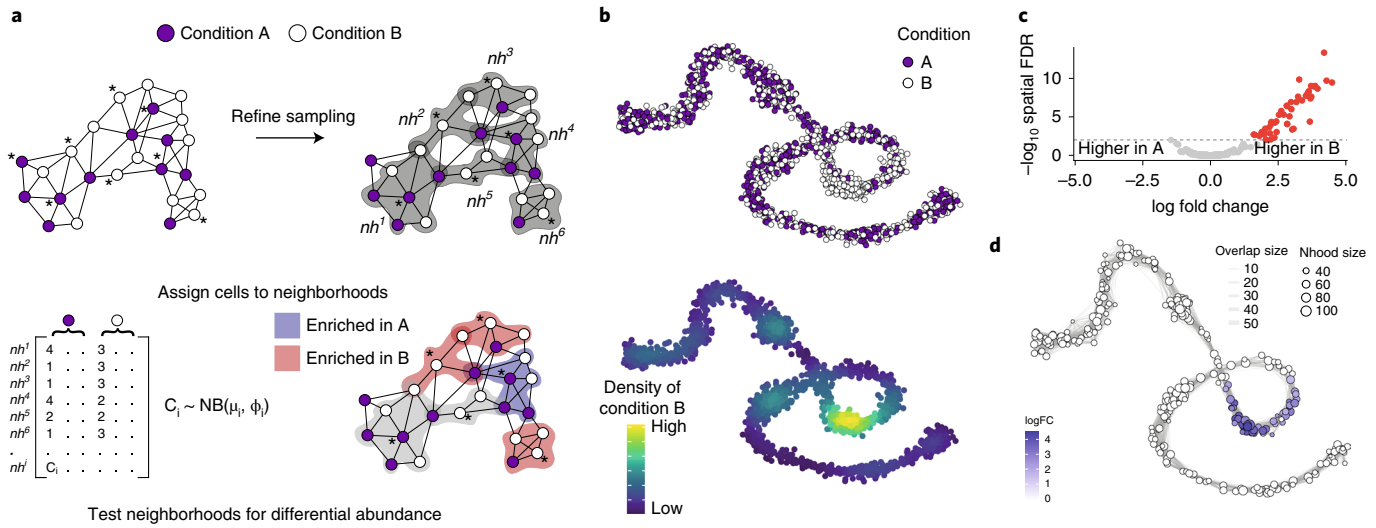


Fig. 1 | Detecting perturbed cell states as differentially abundant graph neighborhoods. a, Schematic of the Milo workflow. Neighborhoods are defined on index cells, selected using a graph sampling algorithm. Cells are quantified according to the experimental design to generate a counts table. Per-neighborhood cell counts are modelled using a negative binomial GLM, and hypothesis testing is performed to determine differentially abundant neighborhoods. **b**, A force-directed layout of a KNN graph representing a simulated continuous trajectory of cells sampled from two experimental conditions (top panel: condition A, purple; condition B, white; bottom panel: kernel density of cells in condition B). **c**, Hypothesis testing using Milo accurately and specifically detects differentially abundant neighborhoods (FDR 1%). Red points denote differentially abundant neighborhoods. **d**, A graph representation of the results from Milo differential abundance testing. Nodes are neighborhoods, colored by their log fold change. Non-differentially abundant neighborhoods (FDR 1%) are colored white, and sizes correspond to the number of cells in a neighborhood. Graph edges depict the number of cells shared between adjacent neighborhoods. The layout of nodes is determined by the position of the neighborhood index cell in the force-directed embedding of single cells. Nhood, neighborhood.

selection of parameters for KNN graph construction are provided in the Supplementary Notes). The first step in our method is to define a set of representative neighborhoods on the KNN graph, where a neighborhood is defined as the group of cells that are connected to an index cell by an edge in the graph. Consequently, we need to sample a subset of single cells to use as neighborhood indices. Adopting a purely random sampling approach means that the number of neighborhoods required to sample a fixed proportion of cells scales linearly with the total number of index cells (Supplementary Fig. 1a). This leads to an increased multiple testing burden, with the potential to reduce statistical power. To solve this problem, we implemented a refined sampling scheme (Fig. 1a)[9]. Concretely, we perform an initial sparse sampling, without replacement, of single cells and compute the KNNs for each sampled cell. We then calculate the median position of each set of nearest neighbors and find the nearest cell to this median position. These adjacent cells become the set of indices from which we compute the final set of neighborhoods. This procedure has three main advantages: (1) fewer, yet more representative, neighborhoods are selected, as initial random samplings from dense regions of the KNN graph will often converge to the same index cell (Supplementary Fig. 1a); (2) the representative neighborhoods include more cells on average (Supplementary Fig. 1b); and (3) neighborhood selection is more robust across initializations (Supplementary Fig. 1c).

Next, we count the numbers of cells present in each neighborhood (per experimental sample) and use these for differential abundance testing among conditions. To incorporate complex experimental designs (for example, the presence of multiple conditions), we test for differences in abundance using a negative binomial (NB) GLM framework[10,11], normalizing for differences in cell numbers across samples[12] (Methods and Supplementary Notes). By doing this, we can borrow information across neighborhoods, allowing robust estimation of dispersion parameters. We employ a quasi-likelihood (QL) *F*-statistic[13] for comparing different hypotheses, which has been shown to be powerful in single-cell differential expression testing[14]. To account for multiple hypothesis testing, we use a weighted FDR procedure[13] that accounts for the spatial overlap of neighborhoods, building upon an approach initially introduced in Cydar[6]. We adapt this procedure for a KNN graph and weight each hypothesis test *P* value by the reciprocal of the *k*th nearest neighbor distance.

To illustrate the Milo workflow, we generated a simulated trajectory[15] composed of cells sampled from two experimental conditions: 'A' and 'B' (Fig. 1b). Cells in a defined subpopulation of this trajectory were simulated to be more abundant in the 'B' condition (Fig. 1b); this region of differential abundance is not defined as a distinct cluster by widely used clustering algorithms (Supplementary Fig. 2). However, applying Milo to these simulated data specifically detects that this region contains different abundances of cells from the two conditions (Fig. 1c,d).

Milo outperforms existing methods for differential abundance testing and controls for false discoveries in the presence of batch effects. To illustrate the power and accuracy of Milo, we tested its performance against alternative methods for differential abundance analysis, simulating regions of differential abundance between two conditions (C1 or C2) on real and simulated single-cell datasets. The methods that we compare define regions for differential abundance testing in the high-dimensional single-cell space in different ways (Methods). Therefore, to provide a fair comparison between them, we reasoned that generating single-cell probabilities and using these to assign cells to a differential abundance region based on a threshold defines a ground truth that does not favor a specific method. To do this, we generated a smooth probability for each cell that it was sampled from C1 over a defined region of a KNN graph (hereafter referred to as *P*(C1)). We then simulate a condition label (C1 or C2) for each single cell using this probability (Methods). Cells from each condition are then assigned to one of three simulated replicates, thus

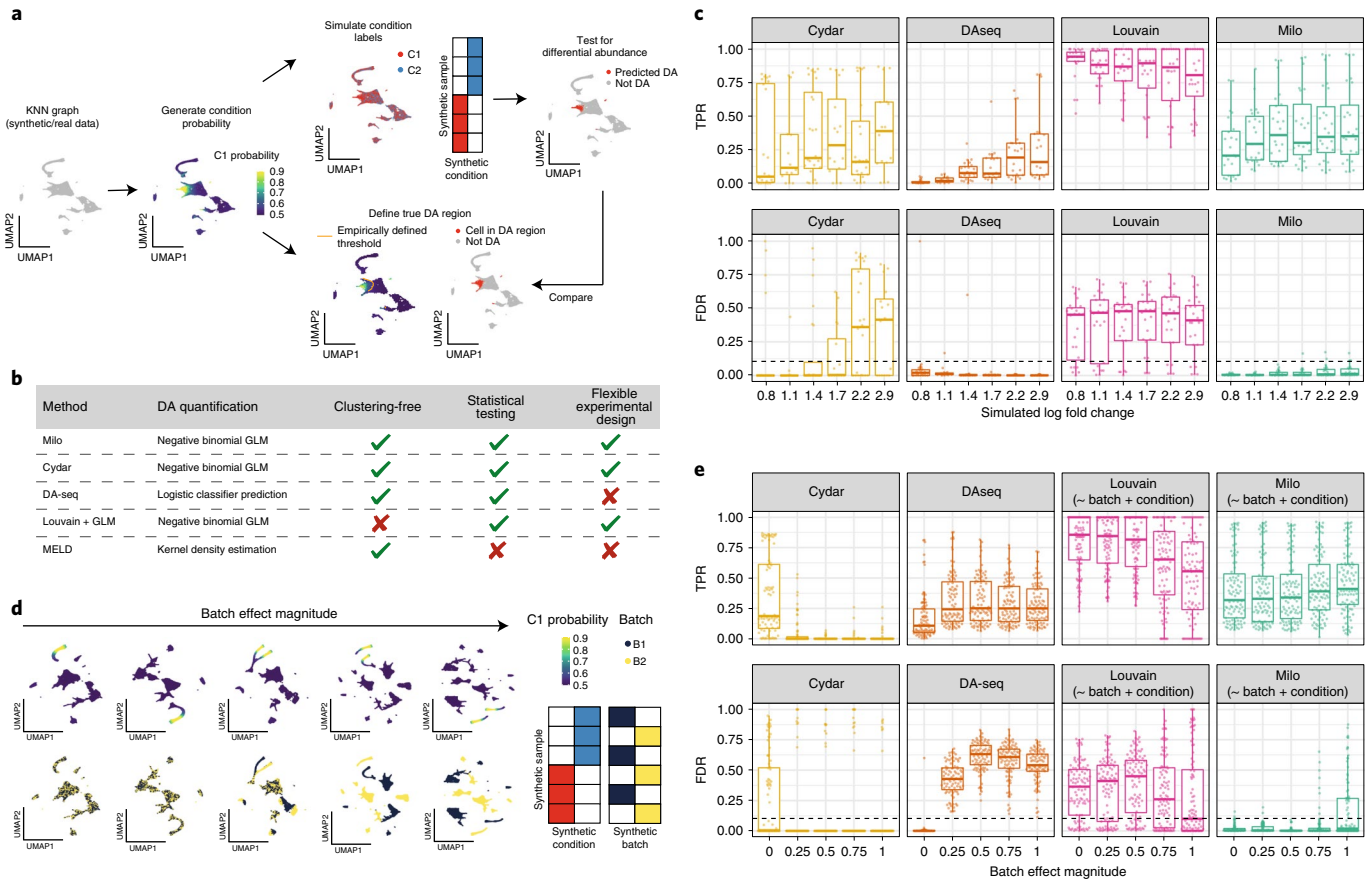


Fig. 2 | Milo outperforms alternative differential abundance testing approaches and controls for false discoveries in the presence of batch effects. **a**, Schematic of the strategy used to simulate ground truth regions of differential abundance to evaluate differential abundance testing methods: on a KNN graph from real or simulated single-cell profiles we generate a smooth probability for each single-cell to be in condition 1 (P(C1)), that is used to assign condition labels to cells (cells in each condition are randomly split into three synthetic replicates for testing). Differential abundance testing methods are run counting the cells in each synthetic condition. The cells assigned to a predicted differential abundance region are compared to a ground truth differential abundance region, defined by cells over an empirically determined threshold for each dataset topology. **b**, A table outlining the characteristics of the methods compared to Milo (see Supplementary Table 3 for a more extensive comparison). **c**, True positive rate (TPR, top) and false discovery rate (FDR, bottom) for recovery of cells in simulated differential abundance regions for differential abundance populations with increasing simulated fold change on the mouse gastrulation dataset. For each boxplot, results from eight populations and three condition simulations per population are shown ($n = 24$ simulations). Each panel represents a different differential abundance method. **d**, Example uniform manifold approximation and projection (UMAP) visualization of simulated batch effects of increasing magnitude. **e**, TPR (top) and FDR (FDR) for recovery of cells in simulated differential abundance regions for differential abundance populations with simulated batch effects of increasing magnitude. For each boxplot results from eight populations, simulated fold change > 1.5 and three condition simulations per population and fold change are shown ($n = 72$ simulations). Each panel represents a different differential abundance method. The batch covariate is included in the test design for GLM-based methods (~ batch + condition). In **c** and **e**, boxplots show the median with interquartile ranges (25–75%); whiskers extend to the largest value no further than 1.5X the interquartile range from the box, with outlier data points shown beyond this range. DA, differential abundance.

mimicking a balanced experimental design with a minimal number of replicates required to estimate a variance parameter. These datasets with simulated condition labels provide a ground truth against which the performance of differential abundance testing approaches can be compared (Fig. 2a). We define differential abundance using an empirical threshold, based on the distribution of P(C1), for each simulated dataset. Specifically, cells with P(C1) > 75th percentile across all cells and simulations for a given topology are assigned to a differentially abundant region (Supplementary Fig. 3a). This choice of threshold does not penalize detection of small shifts in differential abundance while filtering out false positives due to noise around P(C1) = 0.5 (Supplementary Fig. 3b).

We created three simulated datasets, to which we applied the above condition labeling: three discrete clusters (2,700 cells; Extended Data Fig. 1a), a linear trajectory (7,500 cells; Extended Data Fig. 1b) and a branching trajectory (7,500 cells; Extended Data Fig. 1c). In addition, to provide a comparison using a more realistic dataset, we simulated differential abundance labels on a real dataset based on a single-cell atlas of mouse gastrulation (64,018 cells; Fig. 2a)[4]. For each dataset, we simulated labels varying the location of the differential abundance population in the graph, as well as the maximum P(C1), thus mimicking different differential abundance fold changes. Parameters used to simulate multi-condition datasets for benchmarking are summarized in Supplementary Table 1 and described in detail in the Methods. This framework allows us to evaluate Milo's performance on a range of KNN graph geometries, including real-world data representing complex developmental trajectories, while varying both the size and location of differential abundance regions and effect sizes, representing a range of challenging scenarios for differential abundance analysis.

To benchmark the performance of Milo, we compared the results to two alternative methods for clustering-free differential abundance analysis (Fig. 2b and Supplementary Table 2): Cydar, originally designed to model differential abundance in mass cytometry data[6], and DA-seq[7], which uses the structure of single-cell KNN graphs to identify differential abundance regions. In addition, we applied the current standard-of-practice differential abundance analysis for single-cell experiments: graph clustering followed by differential abundance testing among conditions within clusters. To do this, we applied the Louvain clustering algorithm to the same KNN graph as used for Milo and tested for differential abundance using the same NB GLM framework employed by Milo. To ensure comparability between methods, we used the same reduced dimensional space as the input for all methods and the same parameter values, where these were shared—for example, the value of k for KNN graph building. Where parameters were specific to a method, we made use of the recommended practice by the method developers to select an appropriate value (Supplementary Table 3).

Milo detected the simulated differential abundance regions with high sensitivity and maintained FDR control across benchmarking scenarios (Fig. 2c and Extended Data Fig. 1). Although, in specific simulated datasets, other methods achieve similar performance, Milo outperformed all methods on simulated data generated using the real KNN graph. In contrast, DA-seq showed a consistently lower sensitivity, even for large simulated fold changes (Fig. 2c and Extended Data Fig. 1). In the discrete cluster simulation, we found that multiple methods had an inflated FDR due to compositional biases; this was most evident where the differentially abundant cluster comprised ≥50% of the dataset cells (Extended Data Fig. 1a). Notably, Milo mitigates against these compositional biases using a combination of trimmed mean of M-values (TMM) normalization and a graph spatial FDR, which was demonstrated by a median FDR of ~10% across these specific simulations (Extended Data Fig. 1a).

Although Milo generally outperforms alternative methods, we observed some variability in the true-positive rate (TPR) from Milo when attempting to identify differential abundance populations in different regions of the KNN graph (Fig. 2c). We found that, although Milo is consistently sensitive to changes at higher fold changes (Extended Data Fig. 2a), the variability in power between simulations from different population centroids is primarily accounted for by the fraction of true-positive cells with $P(\mathrm{C1})$ close to the threshold used to define true differential abundance—that is, $P(\mathrm{C1}) = 0.551$ (Extended Data Fig. 2a,b)—rather than by other factors, such as the coverage or the size of the differential abundance region (Extended Data Fig. 2c,d).

Burkhardt et al. recently published MELD, a method that quantifies shifts in abundance between conditions over a KNN graph[8]. MELD estimates the probability of observing each cell in each of the experimental conditions, while averaging the density over replicates from the same condition, but does not provide any statistical measure of confidence. This hinders the computation of TPR and FDR because differential abundance would be based on arbitrary probability thresholds. Instead, we separately compared the accuracy of the effect size estimates from Milo and MELD relative to the true effect sizes from our mouse gastrulation simulations. We found that MELD consistently underestimates the fold change of true differential abundance regions (Extended Data Fig. 3a,b). By contrast, the effect size estimates from Milo were unbiased and were more accurate for true differential abundance regions, especially for higher fold changes. Notably, because Milo models sample-to-sample variability, increasing the number of replicates increases the accuracy of the effect estimates (Supplementary Fig. 4).

Technical batch effects have the potential to pose a particular challenge to differential abundance testing methods, including all of those benchmarked above. Moreover, biological sources of variation might confound analyses, such as the co-variation of age, gender and other biological factors with the experimental variable of interest.

To assess the ability of Milo to detect differential abundance among conditions of interest in the presence of confounding factors, we extended our benchmarking using synthetic condition labels by introducing synthetic batch effects with increasing magnitude (Methods and Fig. 2d). Current best practice recommendations suggest applying batch correction methods during dataset pre-processing, to obtain a KNN graph embedding where differences due to batch are minimized (see refs. [16–18] for a systematic review and benchmarking of different batch correction strategies). We confirm that using an in silico batch correction maintains the sensitivity of all differential abundance methods, similarly to the case of no batch effect, across different batch effect magnitudes (Extended Data Fig. 4a; batch effects corrected using fastMNN[19]; further practical considerations on minimizing batch effects are discussed in Supplementary Note 3.1.2). However, no batch correction algorithm is perfect, and uncorrected or incompletely corrected batch effects might lead to false discoveries in differential abundance analysis. The GLM framework implemented in Milo can model nuisance covariates, enabling a direct adjustment for such batch effects.

To assess the effect of such an adjustment, we tested for differential abundance in the datasets with uncorrected synthetic batch effects, incorporating the batch covariate in the NB GLM model used in both Milo and for testing on Louvain clusters. We found that Milo was the only method to maintain FDR control across batch effects of increasing magnitude (Fig. 2e). Even in the presence of partial confounding between the experimental variable and batch, Milo is still able to identify differential abundance and control FDR, particularly for large effect sizes, despite relatively strong batch effects (Extended Data Fig. 4b). The inability of methods such as MELD to incorporate the experimental design in the likelihood estimation severely hinders performance in the presence of uncorrected batch effects (Extended Data Fig. 3c–f). By contrast, explicit modeling of the batch effect as a regression term in the Milo GLM maintains the sensitivity of the differential abundance (Extended Data Fig. 4c).

In summary, we show that Milo outperforms alternative methods for differential abundance testing across a range of experimental scenarios, including in the presence of residual batch effects.

Milo is fast and scalable. The benchmarking datasets discussed above are fairly typical in size for current single-cell experiments. However, moving forward, the number of cells assayed is likely to increase with advances in experimental sample multiplexing[20,21]. As such, we tested the scalability of the Milo workflow and profiled the memory usage across multiple steps. For this, we ran Milo on three published datasets of differing sizes, from ~2,000 to ~130,000 cells, representing differences in both biological and experimental complexity[2–4], as well as a dataset of 200,000 simulated single cells from a linear trajectory (Methods). Using these four datasets, we measured the amount of time required to execute the Milo workflow, from graph building through to differential abundance testing (Fig. 3a). In parallel, we profiled the amount of memory used across the entire workflow (Fig. 3b) and at each defined step (Supplementary Fig. 5). Notably, the amount of time taken increased linearly with the total size of the dataset (Fig. 3a), which, for a large set of 200,000 cells, was less than 60 min. Moreover, the total memory usage across all steps of the Milo workflow scaled primarily with the size of the input dataset (Fig. 3b), indicating that the complexity and composition of the single cells largely determines the memory requirements (Supplementary Fig 5). These memory requirements are within the resources of common desktop

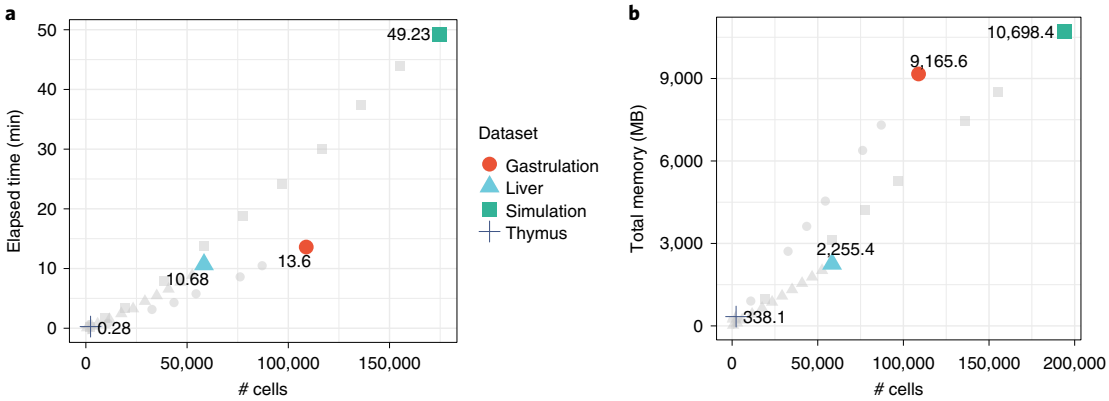


Fig. 3 | Milo efficiently scales to large datasets. a, Run time (*y*-axis) of the Milo workflow from graph building to differential abundance testing. Each point represents a down-sampled dataset, denoted by shape. Colored points show the total number of cells in the full dataset labelled by the elapsed system time (min). **b**, Total memory usage (*y*-axis) across the Milo workflow. Each point represents a down-sampled dataset, denoted by shape. Colored points are the full datasets labelled with the total memory usage (MB, megabytes).

computers (that is, <16 GB). This benchmarking analysis demonstrates that Milo is able to perform differential abundance analysis in large and complex datasets at a scale and speed that is feasible on a desktop computer.

Milo identifies the decline of a fate-biased epithelial precursor in the aging mouse thymus. To demonstrate the utility of Milo in a real-world setting, we applied it to an scRNA-seq dataset of mouse thymic epithelial cells (TECs) sampled across the first year of mouse life, which were previously clustered into nine distinct TEC subtypes (Fig. 4a)[3]. These data, generated using plate-based SMART-seq2, consist of 2,327 single cells equally sampled from mice at five different ages: 1, 4, 16, 32 and 52 weeks old (Fig. 4b). Moreover, the experimental design included five replicate experimental samples of cells for each age. The goal of the study was to identify TEC subtypes that change in frequency during natural aging.

To this end, we first constructed a KNN graph ($k=21$), before assigning cells to 363 neighborhoods, which were then used to test for differential abundance of TEC states across time. At a 10% FDR, we identified 208 differentially abundant neighbourhoods (101 showed a decreased abundance with age; 107 showed an increased abundance with age) spanning multiple TEC states (Fig. 4c). We compared our results to those generated in the original publication, which demonstrated that we were able to identify all previously identified differentially abundant states (Fig. 4d), including changes in the abundance of the structural thymic epithelial cell (sTEC) population, which consisted of just 24 cells. Moreover, while we recovered the previously reported accumulation of intertypical TECs with age, we also identified an additional subset of these cells that were depleted with age (Fig. 4c,d).

To understand the function of the substate of intertypical TECs identified using Milo, we first grouped the differential abundance neighborhoods with overlapping cells and concordant differential abundance fold change (Fig. 4e and Methods) and then performed marker gene expression identification among the neighborhood groups corresponding to intertypical TECs enriched or depleted in younger mice (FDR = 10%; Fig. 4f). This analysis indicated that the cells from younger mice upregulated multiple cytokine response genes (for example, *Stat1*, *Stat4* and *Aff3*), illustrated by the enriched Gene Ontology term GO:0034097 'response to cytokine' (enrichment adjusted $P=0.047$). Cytokine signaling is key to medullary thymic epithelial cell (mTEC) differentiation[22,23], indicating that these TECs from younger mice might be differentiating more efficiently to the mTEC lineage.

Independent evidence in support of the observation that a subpopulation of intertypical TECs are depleted with age was previously described by Baran-Gale et al.[3], using a dataset comprised of ~90,000 single-cell transcriptomes (profiled with the 10x Genomics platform) coupled with lineage tracing. To further validate the robustness of our observation in the small SMART-seq-based dataset, we first integrated the SMART-seq and droplet scRNA-seq datasets using the mutual nearest neighbors (MNN) algorithm[19]. To identify the subpopulations in the droplet scRNA-seq data that are transcriptionally similar to the cells in the neighborhood groups identified by Milo, we transferred neighborhood group labels from the SMART-seq dataset onto the droplet scRNA-seq dataset (Fig. 4g–i, Supplementary Fig 6a and Methods). Using these groups, we examined how these subpopulations vary across ages by calculating the proportions of cells assigned to each neighborhood group across ages (Fig. 4h and Supplementary Fig 6b). This analysis revealed that the cells transcriptionally similar to the mTEC-biased intertypical TEC ('Neighborhood Group 5') were indeed depleted in older mice, providing independent validation of our findings in a dataset comprising ~90,000 single cells (Fig. 4i). In summary, these analyses demonstrate the sensitivity of Milo by identifying that an mTEC progenitor state is depleted with age, a finding that was not resolved using clustering approaches.

Milo identifies compositional disorder in cirrhotic human liver. To demonstrate the applicability of our method in multiple biological contexts, we next applied Milo to a dataset of 58,358 hepatic cells isolated from five healthy and five cirrhotic human livers[2]. The original study assigned cells to multiple lineages, including immune, endothelial and mesenchymal cells (Fig. 5a,b). A key goal of the study was to determine whether different cell types were differentially abundant among experimental samples taken from healthy and cirrhotic tissue. In the original study, cells from each lineage were subclustered, and these subclusters were interrogated using a Poisson GLM to determine whether there were differential contributions from cirrhotic and healthy donors.

To explore whether more subtle differences could be detected, we applied Milo, identifying 2,677 neighborhoods spanning the KNN graph, of which 1,351 showed evidence of differential abundance (FDR = 10%; Fig. 5c). To assess performance, we compared differential abundance results with those from the compositional analysis performed by Ramachandran et al.[2]. Milo recovered differentially

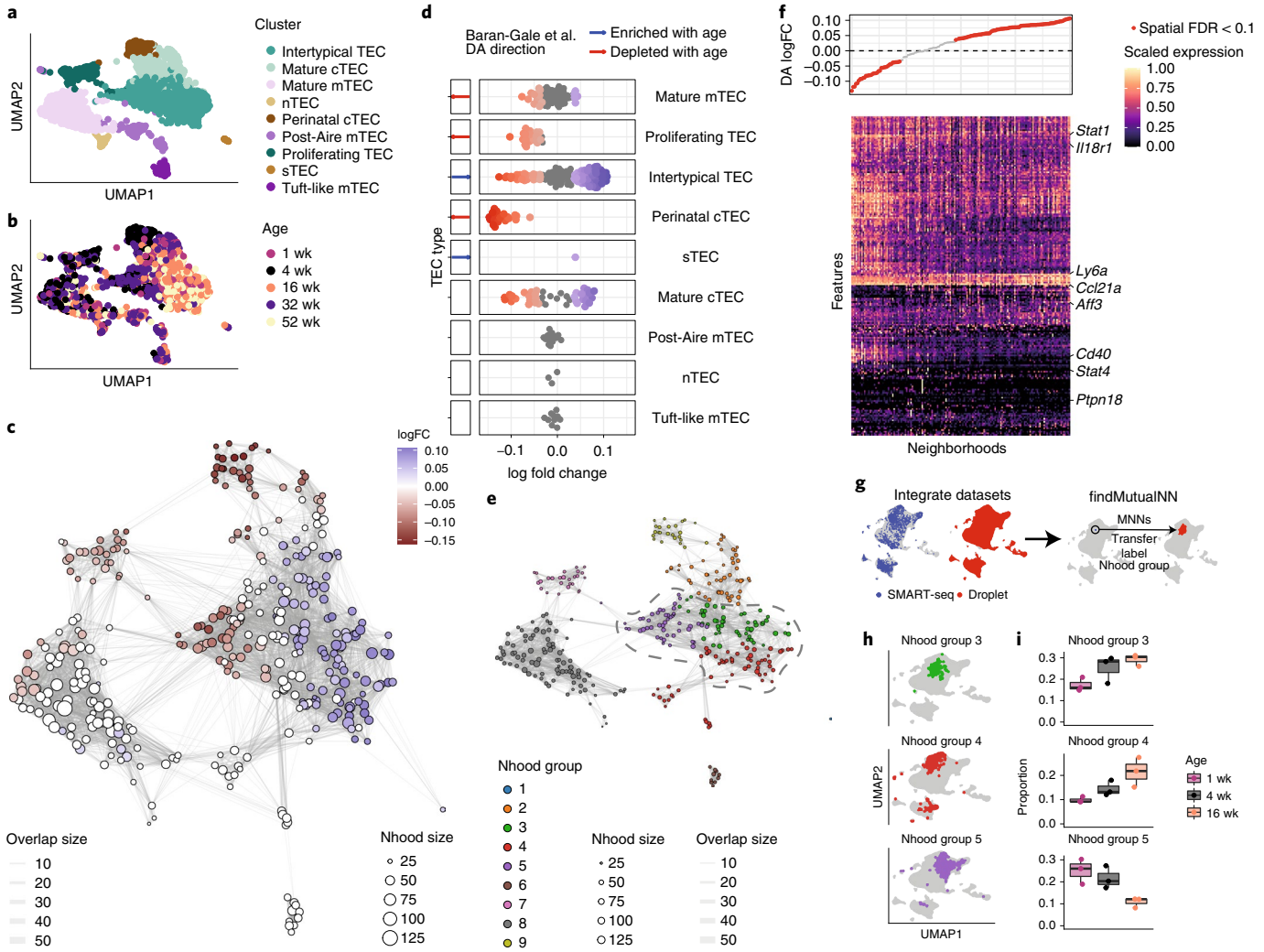


Fig. 4 | Milo identifies the decline of a fate-biased precursor in the aging mouse thymus. a, **b**, A UMAP of murine thymic epithelial cells. Points are labelled according to their annotation in Baran-Gale et al.[3] (**a**) and mouse age (**b**). **c**, A neighborhood graph of the results from Milo differential abundance testing. Nodes are neighborhoods, colored by their log fold change across ages. Non-differential abundance neighborhoods (FDR 10%) are colored white, and sizes correspond to the number of cells in each neighborhood. Graph edges depict the number of cells shared between neighborhoods. The layout of nodes is determined by the position of the neighborhood index cell in the UMAP in panel **a**. **d**, Beeswarm plot of the distribution of log fold change across age in neighborhoods containing cells from different cell type clusters. Differential abundance neighborhoods at FDR 10% are colored. Cell types detected as differential abundance through clustering by Baran-Gale et al.[3] are annotated in the left side bar. **e**, Neighborhood grouping, overlaid on the neighborhood graph as in panel **c**. Colors denote assignment of neighborhoods to discrete groups using Louvain clustering. The region encircled by the dashed line denotes neighborhood groups that correspond to the intertypical TEC subpopulation. **f**, A heatmap of differentially expressed genes between differential abundance neighborhoods in the intertypical TEC cluster. Columns are neighborhoods and rows are differentially expressed genes (FDR 5%). Expression values for each gene are scaled between 0 and 1. The top panel denotes the neighborhood log fold change. **g**, Schematic showing the transfer of neighborhood group labels to an independent droplet scRNA-seq dataset of aging mouse TEC from Baran-Gale et al.[3]. **h**, A joint UMAP of SMARTseq and droplet scRNA-seq datasets, highlighting label-transferred neighborhood groups, as colored in panel **e**. **i**, Boxplots showing the proportion of cells from the droplet scRNA-seq dataset for each neighborhood group across ages. Individual points represent replicates ($n = 3$ mice per age). Boxplots show the median with interquartile range (IQR) (25–75%); whiskers extend 1.5X the IQR. DA, differential abundance; Nhood, neighborhood; wk, week.

abundant neighborhoods in all clusters identified as differentially abundant among cirrhotic or uninjured tissue in the original study (Fig. 5d).

Moreover, Milo identified multiple groups of neighborhoods within the same pre-defined subclusters that showed opposing directions of differential abundance between the control and cirrhotic liver experimental samples (Fig. 5d). In other words, within a subcluster, some neighborhoods were enriched for control experimental samples, whereas others were enriched for disease experimental samples. These patterns, exemplified by the T cell (2) and the endothelial (5) compartments, were obscured in the previous study due to the reliance on pre-clustering (Fig. 5d).

To further explore the biological meaning of these neighborhoods, we first focused on the hepatic endothelial cells, where we resolved disease-specific subpopulations at higher resolution than was possible by clustering-based analysis (Fig. 5d). Milo identified a gradient of changes in neighborhood abundance across this compartment, suggestive of a continuous transition between healthy and diseased cell states (Fig. 5e). To identify gene expression signatures associated with this change, we performed differential

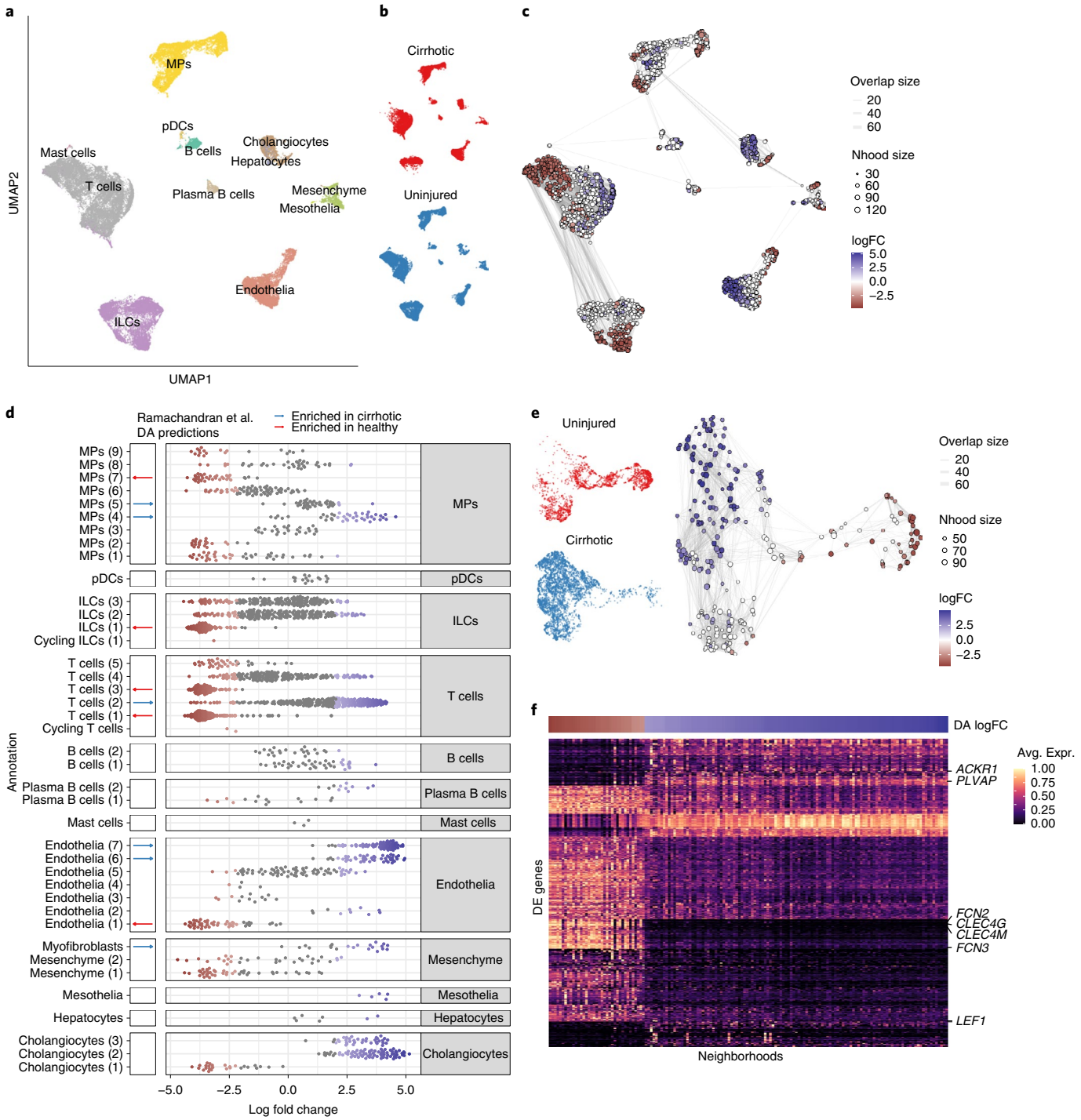


Fig. 5 | Milo identifies the compositional disorder in cirrhotic liver. a, **b**, UMAP embedding of 58,358 cells from healthy ($n = 5$) and cirrhotic ($n = 5$) human livers. Cells are colored by cellular lineage (**a**) and injury condition (**b**). ILCs, innate lymphoid cells; MPs, mononuclear phagocytes; pDCs, plasmacytoid dendritic cells. **c**, Graph representation of neighborhoods identified by Milo. Nodes are neighborhoods, colored by their log fold change between cirrhotic and healthy samples. Non-differential abundance neighborhoods (FDR 10%) are colored white, and sizes correspond to the number of cells in a neighborhood. Graph edges depict the number of cells shared between adjacent neighborhoods. The layout of nodes is determined by the position of the neighborhood index cell in the UMAP embedding of single cells. **d**, Beeswarm plot showing the distribution of log fold change in abundance between conditions in neighborhoods from different cell type clusters. Differential abundance neighborhoods at FDR 10% are colored. Cell types detected as differential abundance through clustering by Ramachandran et al.[2] are annotated in the left side bar. **e**, UMAP embedding and graph representation of neighborhoods of 7,995 cells from endothelial lineage, colored by differential abundance log fold change. **f**, Heatmap showing average neighborhood expression of genes differentially expressed between differential abundance neighborhoods in the endothelial lineage (788 genes). Expression values for each gene are scaled between 0 and 1. The top panel denotes the neighborhood differential abundance log fold change. DA, differential abundance; DE, differentially expressed; Nhood, neighborhood.

expression analysis of cells in differentially abundant neighborhoods with positive and negative log fold changes (LFCs), identifying 788 differentially expressed genes (FDR = 5%; Methods) (Fig. 5f). In the cirrhosis-enriched neighborhoods, we recovered overexpression of known markers of scar-associated endothelium, including *ACKR1* and *PLVAP* (Fig. 5f)[2]. We also recovered overexpression of genes associated with regulation of leukocyte recruitment, confirming the validated immunomodulatory phenotype displayed by scar-associated tissue (Supplementary Fig. 7a)[24]. In addition, cirrhotic endothelium displayed a downregulation of genes involved in response to infection, endocytosis and immune complex clearance, including *FCN2* and *FCN3* (Supplementary Fig. 7b), which has been suggested as an additional component of cirrhosis-associated immune dysfunction[25,26].

Milo also identified strong differential abundance between healthy and cirrhotic cells in lineages that were unexplored in the original study, such as the cholangiocyte compartment (Fig. 5d). Cholangiocytes are epithelial cells that line a three-dimensional network of bile ducts known as the biliary tree, and cholangiocyte proliferation can be induced by a broad range of liver injuries in a process termed the ductular reaction[27]. However, the gene signatures associated with this process in human cirrhosis are largely unexplored. Milo recovered an enrichment of disease-specific cholangiocytes (Supplementary Fig. 7c,d). Performing differential gene expression analysis restricted to this subset, we detected overexpression of genes associated with fibrosis, wound healing and angiogenesis (Supplementary Fig. 7e,f), which has been shown to accompany the ductular reaction[28,29].

These analyses demonstrate the potential of using differentially abundant subpopulations detected by Milo to recover known and novel signatures of disease-specific cell states.

Discussion

Given the increasing number of complex single-cell datasets where multiple conditions are assayed[20,21], Milo tackles a key problem: determining sets of cells that are differentially abundant among conditions without relying on pre-existing sets of clusters. Moreover, Milo is fully interoperable with established single-cell analysis workflows and is implemented as an open-source R software package[30] with documentation and tutorials at https://github.com/MarioniLab/miloR.

The definition of neighborhoods, as implemented in Milo, overcomes the main limitations of standard-of-practice clustering-based differential abundance analysis while using a common data structure in single-cell analysis—graphs. A strength of our approach is that it is applicable to a wide range of datasets with different topologies, including gradual state transitions, thus removing the need for time-consuming iterative subclustering and identifying subtle differences in differential abundance that would otherwise be obscured (Fig. 5d).

Recently, other clustering-free methods have been proposed to detect compositional differences among experimental conditions[7,8]. However, these methods do not exploit the replication structure in the experimental design to account for technical variability among samples. In addition, MELD and DA-seq are designed for pairwise comparisons between two biological conditions and cannot be easily extended to detect differential abundance with complex experimental designs, including continuous variables (age and timepoints), multifactorial conditions or nuisance covariates. By modeling cell counts with an NB GLM, Milo can incorporate arbitrarily complex experimental designs, as demonstrated by our application of Milo to detect compositional changes in the aging mouse thymus (Fig. 4). This flexibility is illustrated in the ability of Milo to account appropriately for batch effects of varying magnitude, controlling false discoveries, which is not possible using other differential abundance testing algorithms (Fig. 2e).

Although we have addressed several challenges, several qualifiers should be considered when performing differential abundance analysis with Milo. First, reliable results from differential abundance testing depend on well-designed single-cell experiments. Biological replicates are required to estimate the NB dispersion parameter for each neighborhood. Moreover, when considering experimental design, it is vital to avoid complete confounding among technical sources of variation and experimental variables of interest, including the number of cells acquired for each condition. Although we have shown that nuisance effects can be minimized by applying batch integration before graph building (Extended Data Fig. 4a) and by incorporating known confounders in the testing framework (Fig. 2e and Extended Data Fig. 4c), these strategies can lead to loss of biological signal if the condition of interest and the confounders are strongly correlated. Second, cells in a single neighborhood do not necessarily represent a unique biological subpopulation; a cellular state might span multiple neighborhoods. Accordingly, we search for marker genes of differential abundance states by aggregating cells in adjacent and concordantly differential abundance neighborhoods (Figs. 4e and 5f). One challenge of this approach is that rare cell states might be represented by a small subset of neighborhoods, thus making identification of marker genes challenging. To overcome this problem, one can either choose a smaller value of k or, alternatively, construct a graph on cells from a particular lineage of interest. Third, there will be cases where differential abundance analysis on clusters or pre-defined cell populations will be preferable—for example, where differences are apparent in large clusters by visualization—or to compare abundances of pre-annotated cell populations across datasets without requiring integration of single cells on a common manifold. Alternative methods that model cell type proportions might be more suitable for these applications[31].

Following the generation of reference single-cell atlases for multiple organisms and tissues, an increasing number of studies now focus on quantifying how cell populations are perturbed in disease, aging and development, using, for example, large-scaled pooled CRISPR screens[32–34]. We envision that Milo will see use in all of these contexts. Milo might also be applicable to single-cell assays other than scRNA-seq, including multi-omic assays[35–39]. Thus, Milo has the potential to facilitate the discovery of fundamental biological and clinically relevant processes across multiple layers of molecular regulation when they are assayed at single-cell resolution.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41587-021-01033-z.

Received: 20 November 2020; Accepted: 26 July 2021;
Published online: 30 September 2021

References

1. Kiselev, V. Y., Andrews, T. S. & Hemberg, M. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat. Rev. Genet.* **20**, 273–282 (2019).
2. Ramachandran, P. et al. Resolving the fibrotic niche of human liver cirrhosis at single-cell level. *Nature* **575**, 512–518 (2019).
3. Baran-Gale, J. et al. Ageing compromises mouse thymus function and remodels epithelial cell differentiation. *eLife* **9**, e56221 (2020).
4. Pijuan-Sala, B. et al. A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature* **566**, 490–495 (2019).
5. Haber, A. L. et al. A single-cell survey of the small intestinal epithelium. *Nature* **551**, 333–339 (2017).
6. Lun, A. T. L., Richard, A. C. & Marioni, J. C. Testing for differential abundance in mass cytometry data. *Nat. Methods* **14**, 707–709 (2017).

7. Zhao, J. et al. Detection of differentially abundant cell subpopulations discriminates biological states in scRNA-seq data. *Proc. Natl Acad. Sci. USA* **118**, e2100293118 (2021).
8. Burkhardt, D. B. et al. Quantifying the effect of experimental perturbations at single-cell resolution. *Nat. Biotechnol.* **39**, 619–629 (2021).
9. Gut, G., Tadmor, M. D., Pe'er, D., Pelkmans, L. & Liberali, P. Trajectories of cell-cycle progression from fixed cell populations. *Nat. Methods* **12**, 951–954 (2015).
10. McCarthy, D. J., Chen, Y. & Smyth, G. K. Differential expression analysis of multifactor RNA-seq experiments with respect to biological variation. *Nucleic Acids Res.* **40**, 4288–4297 (2012).
11. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2010).
12. Robinson, M. D. & Oshlack, A. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biol.* **11**, R25 (2010).
13. Benjamini, Y. & Hochberg, Y. Multiple hypotheses testing with weights. *Scand. J. Statist.* **24**, 407–418 (1997).
14. Soneson, C. & Robinson, M. D. Bias, robustness and scalability in single-cell differential expression analysis. *Nat. Methods* **15**, 255–261 (2018).
15. Cannoodt, R., Saelens, W., Deconinck, L. & Saeys, Y. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nat. Communications* **12**, 1–9 (2021).
16. Luecken, M. et al. Benchmarking atlas-level data integration in single-cell genomics. Preprint at https://www.biorxiv.org/content/10.1101/2020.05.22.111161v2 (2020).
17. Tran, H. T. N. et al. A benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biol.* **21**, 12 (2020).
18. Chazarra-Gil, R., van Dongen, S., Kiselev, V. Y. & Hemberg, M. Flexible comparison of batch correction methods for single-cell RNA-seq using BatchBench. *Nucleic Acids Res.* **49**, e42 (2021).
19. Haghverdi, L., Lun, A. T. L., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **36**, 421–427 (2018).
20. Stoeckius, M. et al. Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. *Genome Biol.* **19**, 224 (2018).
21. McGinnis, C. S. et al. MULTI-seq: sample multiplexing for single-cell RNA sequencing using lipid-tagged indices. *Nat. Methods* **16**, 619–626 (2019).
22. Akiyama, T. et al. The tumor necrosis factor family receptors RANK and CD40 cooperatively establish the thymic medullary microenvironment and self-tolerance. *Immunity* **29**, 423–437 (2008).
23. Hikosaka, Y. et al. The cytokine RANKL produced by positively selected thymocytes fosters medullary thymic epithelial cells that express autoimmune regulator. *Immunity* **29**, 438–450 (2008).
24. Wilkinson, A. L., Qurashi, M. & Shetty, S. The role of sinusoidal endothelial cells in the axis of inflammation and cancer within the liver. *Front. Physiol.* **11**, 990 (2020).
25. Foldi, I. et al. Lectin-complement pathway molecules are decreased in patients with cirrhosis and constitute the risk of bacterial infections. *Liver Int.* **37**, 1023–1031 (2017).
26. Ganesan, L. P. et al. FcγRIIb on liver sinusoidal endothelium clears small immune complexes. *J. Immunol.* **189**, 4981–4988 (2012).
27. Sato, K. et al. Ductular reaction in liver diseases: pathological mechanisms and translational significances: liver injury and regeneration. *Hepatology* **69**, 420–430 (2019).
28. Morell, C. M., Fabris, L. & Strazzabosco, M. Vascular biology of the biliary epithelium: biliary epithelium vascular biology. *J. Gastroenterol. Hepatol.* **28**, 26–32 (2013).
29. Mariotti, V., Fiorotto, R., Cadamuro, M., Fabris, L. & Strazzabosco, M. New insights on the role of vascular endothelial growth factor in biliary pathophysiology. *JHEP Rep.* **3**, 100251 (2021).
30. R Core Team. R: A Language and Environment for Statistical Computing. https://www.R-project.org (R Foundation for Statistical Computing, 2017).
31. Büttner, M., Ostner, J., Müller, C. l., Theis, F. J. & Schubert, B. scCODA: a Bayesian model for compositional single-cell data analysis. Preprint at https://www.biorxiv.org/content/10.1101/2020.12.14.422688v2 (2020).
32. Dixit, A. et al. Perturb-seq: dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell* **167**, 1853–1866 (2016).
33. Datlinger, P. et al. Pooled CRISPR screening with single-cell transcriptome readout. *Nat. Methods* **14**, 297–301 (2017).
34. Jaitin, D. A. et al. Dissecting immune circuits by linking CRISPR-pooled screens with single-cell RNA-seq. *Cell* **167**, 1883–1896 (2016).
35. Stoeckius, M. et al. Simultaneous epitope and transcriptome measurement in single cells. *Nat. Methods* **14**, 865–868 (2017).
36. Cao, J. et al. Joint profiling of chromatin accessibility and gene expression in thousands of single cells. *Science* **361**, 1380–1385 (2018).
37. Chen, S., Lake, B. B. & Zhang, K. High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nat. Biotechnol.* **37**, 1452–1457 (2019).
38. Zhu, C. et al. An ultra high-throughput method for single-cell joint analysis of open chromatin and transcriptome. *Nat. Struct. Mol. Biol.* **26**, 1063–1070 (2019).
39. Ma, S. et al. Chromatin potential identified by shared single-cell profiling of RNA and chromatin. *Cell* **183**, 1103–1116 (2020).

Methods

Milo. Milo detects sets of cells that are differentially abundant between conditions by modeling counts of cells in neighborhoods of a KNN graph. The workflow includes the following steps:

(A) Construction of the KNN graph. Milo uses a KNN graph computed based on similarities in gene expression space as a representation of the phenotypic manifold on which cells lie. To construct a KNN graph, we follow best practices in single-cell analysis[40] by re-scaling unique molecular identifier counts by per-cell sequencing depth, applying log-transformation and projecting the gene expression matrix of M cells onto the d leading principal components (PCs). Then, we construct the KNN graph by calculating the Euclidean distance between each cell and its k nearest neighbors in PC space. We provide practical guidance for selection of d and k in the Supplementary Notes.

We assume that the KNN graph is a faithful representation of the single-cell phenotypes, where cell–cell similarities are driven by true biological effects rather than technical/batch effects. Although technical covariates can be accounted for in the experimental design of the differential abundance test, we recommend mitigating batch effects among cells from different experimental batches before graph building to maximize the power of differential abundance testing. A large number of methods to integrate single cells from different experimental samples have been reviewed and benchmarked in refs. [16–18]. We provide further practical considerations on how to account for batch effects in the Supplementary Notes.

(B) Definition of cell neighborhoods. We define the neighborhood of cell c_i as the group of cells that are connected to c_i by an edge in the graph. We refer to c_i as the index of the neighbourhood. To define a representative subset of neighborhoods that span the whole KNN graph, we implement a previously developed algorithm to sample the index cells in a graph[9,41] (see Supplementary Note 3.1.2 for a detailed description).

(C) Counting cells in neighborhoods. For each neighborhood, we count the number of cells from each experimental sample, S, constructing an $N \times S$ (neighborhood × experimental sample count) matrix.

(D) Testing for differential abundance in neighborhoods. To test for differential abundance, we analyze neighborhood counts using the QL method in edgeR, similarly to the implementation in Cydar[6]. We fit an NB GLM to the counts for each neighborhood, accounting for different numbers of cells across samples using TMM normalization[12], and use the QL F-test with a specified contrast to compute a P value for each neighborhood. Further details of the statistical framework are provided in Supplementary Note 3.1.5.

(E) Controlling the spatial FDR in neighborhoods. To control for multiple testing, we adapt the spatial FDR method introduced by Cydar[6]. The spatial FDR can be interpreted as the proportion of the union of neighborhoods that is occupied by false-positive neighborhoods. To control the spatial FDR in the KNN graph, we apply a weighted version of the Benjamini–Hochberg method, where P values are weighted by the reciprocal of the neighborhood connectivity. As a measure of neighborhood connectivity, we use the Euclidean distance to the kth nearest neighbor of the index cell for each neighborhood.

Visualization of differential abundance neighborhoods. To visualize results from differential analysis on neighborhoods, we construct an abstracted graph, where nodes represent neighborhoods and edges represent the number of cells in common among neighborhoods. The size of nodes represents the number of cells in the neighborhood. The position of nodes is determined by the position of the sampled index cell in the single-cell uniform manifold approximation and projection (UMAP), to allow qualitative comparison with the single-cell embedding. Graphical visualization is implemented using the R packages ggplot and ggraph.

Benchmarking of differential abundance methods. To evaluate methods for differential abundance analysis using a ground truth, we applied a semi-synthetic approach where we simulated condition labels on KNN graphs from real and simulated single-cell datasets.

Benchmarking datasets. For benchmarking using in silico-generated datasets (Extended Data Fig. 1), we simulated single-cell data representing different trajectory geometries (one-dimensional trajectory and branching trajectory) using the R package dyntoy[15] as well as discrete clusters. For benchmarking on the KNN graph from real data, we downloaded the raw count matrix and the batch-corrected principal component analysis (PCA) matrix for the mouse gastrulation atlas[4] via the R package MouseGastrulationData[42]. We subset the dataset to embryos at developmental stages E7.75–E8.5 (64,018 cells) and used the batch-corrected PCA representation of the data provided in the package for KNN graph construction.

Generation of condition probability. To assign cells in a KNN graph to two simulated experimental conditions (C1 and C2) and to define a ground truth for differential abundance, we generate for each cell x_i a probability $P(\mathrm{C1})$ of belonging to condition C1. For datasets representing continuous trajectory geometries (one-dimensional trajectory, branching trajectory and mouse gastrulation data), we generate $P(\mathrm{C1})_i$ using the following procedure:

1. We select one cell population, q, in which to generate the maximum differential abundance among conditions. For the simulated datasets, we use the cell clusters defined by the simulation. For the mouse gastrulation data, we use the cell-type annotations provided by the publication[4].
2. We identify the centroid $\bar{x}_q$ of the cell population based on the average position of cells in q in PC space.
3. For each cell x_i, following the approach taken in fuzzy clustering, we calculate a weighted distance from $\bar{x}_q$ as:

$$w_i = \frac{1}{\sum_{j=1}^{Q} \left(\frac{x_i - \bar{x}_q}{x_i - \bar{x}_j} \right)^{\frac{2}{m-1}}}$$

where m is a hyper-parameter controlling the strength of membership to the cell population (the higher m, the weaker membership). Unless otherwise stated, we used $m = 2$.

We use a logit transformation to normalize w_i:

$$w_i' = \frac{1}{1 + e^{-\alpha w_i}}$$

where $\alpha = 0.5$ unless otherwise stated.

We then re-scale w_i between 0.5 and f, with $0.5 < f < 1$, to obtain:

$$p_i = \frac{w_i' - \min\left(w_i'\right)}{\max\left(w_i'\right) - \min\left(w_i'\right)} (f - 0.5) + 0.5$$

Here, $P(\mathrm{C1})_i = 0.5$ indicates the absence of differential abundance (equal probability of being in condition C1 or C2), and f indicates the maximum enrichment of condition C1 in population q (the differential abundance effect size).

In the dataset with discrete clusters (Extended Data Fig. 1a), we select one cluster q and assign the same probability of being in condition C1 $P(\mathrm{C1})_i > 0.5$ to all the cells in cluster q. To all other cells, we assign $P(\mathrm{C1})_i = 0.5$.

Assignment of simulated experimental condition labels. Cells were assigned to one of two condition labels (C1 or C2) by randomly sampling a label for each cell based on the probability $P(\mathrm{C1})_i$. For each condition, cells were then randomly assigned to one of the three simulated replicates. This resulted in a total of six simulated experimental samples.

Definition of ground truth for differential abundance testing. To define a region displaying true differential abundance among conditions, we define a probability threshold of $0.5 < t < f$ and assign, to each cell, a true differential abundance outcome label o_i based on the simulated probability:

$$o_i = 0 \text{ (not DA) if } P(\mathrm{C1})_i \leq t$$

$$o_i = 1 \text{ (enriched in C1) if } P(\mathrm{C1})_i > t$$

For datasets representing a continuous trajectory geometry, we set t equal to the 75th percentile of the $P(\mathrm{C1})$ distribution for all simulations for each dataset topology (Supplementary Fig. 3). For the cluster dataset, we set $t = 0.5$.

Simulation of batch effects. To recreate a batch effect with an unbalanced experimental design, we randomly assign experimental samples to two simulated batches. We simulate batch effects by generating a random 0-centered Gaussian vector of length d and adding the same vector to the PC profile of all cells in the same batch. We simulate batch effects of increasing magnitude by increasing the standard deviation of the Gaussian vector (from 0 to 1, steps of 0.25). To demonstrate the effect of in silico batch correction before differential abundance analysis (Extended Data Fig. 4a), we performed batch correction using the MNN method, as implemented in the R package batchelor by the function fastMNN, using default parameters[19].

Benchmarked methods. We benchmark the performance of Milo against four other methods designed to quantify differential abundance in single-cell datasets. We provide details on how each method was run and how we assigned each single cell, i, an outcome label to compare with the true differential abundance label o_i:

1. Louvain: Louvain clustering was performed using the function cluster_louvain from the R package igraph[43]. We tested for differential abundance within clusters using a GLM with NB likelihood, using the QL method implemented in edgeR, using TMM normalization, thereby replicating the testing framework used by Milo on neighborhoods. FDR correction was performed using the Benjamini–Hochberg procedure. We used 10% FDR as a threshold for

significance to assign an outcome label to each cluster. We then assigned the same outcome label to all of the cells that are a member of that cluster.
2. Cydar:[6] We use Cydar by constructing hyperspheres in PC space and asking whether the abundance of cells from different conditions varies in each hypersphere, using the implementation in the Bioconductor package Cydar. To select an appropriate value for the radius parameter for each dataset, we examined the distribution of distances from each cell to its nearest neighbors, as recommended by the authors.
3. DA-seq:[7] DA-seq computes, for each cell, a differential abundance score based on the relative prevalence of cells from both biological states in the cell's neighborhood, using a range of k values. The scores are used as input for a logistic classifier to predict the biological condition of each cell. The method is implemented in https://github.com/KlugerLab/DAseq. To choose a range of k values (k.vector parameter of getDAcells function), we use the same value of k used for differential abundance analysis with Milo and MELD as the lower limit (which represents the smallest number of cells that a user will consider a meaningful region), $k = 500$ as the upper limit and step = 50 as used by default. Of note, the authors demonstrate that the upper limit has limited effect on the method's performance. As recommended by the authors, we select as cells showing significant enrichment/depletion cells with absolute differential abundance measure values larger than the maximum differential abundance measure obtained with randomly permuted labels.
4. Milo: Milo was run as previously described. To convert neighborhood-level outcomes to single-cell-level outcomes, we consider the average outcome in neighborhoods to which a cell belongs.
5. MELD:[8] MELD estimates the probability density of each sample over a KNN graph, which is then used to quantify the relative likelihood that each cell would be observed in one condition relative to the others. The average probability among all samples in a given condition is taken as the condition probability. We used the functions and tutorials implemented in https://github.com/KrishnaswamyLab/MELD. MELD does not perform any statistical inference; instead, the user selects a threshold on the per-cell likelihoods to define a cell as being in a differential abundance region or not.

Details on parameters used for all benchmarking datasets are provided in Supplementary Table 3.

Evaluation metrics. We evaluate method performance by quantifying the TPR and FDR when comparing the predicted single-cell outcomes to the true differential abundance labels o_i. To compare Milo and MELD (Extended Data Fig. 3), we converted the simulated probability to a ground truth LFC, such that $LFC = log\left(P\left(C1\right) / \left(1 - P\left(C1\right)\right)\right)$. We use the same formula to convert the MELD-estimated single-cell probabilities to an estimated LFC. We converted the simulated ground truth and MELD-estimated single-cell probabilities to LFCs. We computed the mean squared error between the ground truth LFCs and the estimates generated by Milo and MELD at the neighborhood index cells.

Scalability analysis. We assessed the scalability of Milo by profiling the time taken to execute the workflow, starting with the KNN graph-building step and concluding with the differential abundance testing. We simulated a dataset of 200,000 single cells using the dyntoy package implemented in R[15]. With this large simulation, we downsampled to specific proportions, ranging from 1% to 100%, and recorded the elapsed system time to complete the Milo workflow using the system.time function in R[30]. In addition, we performed an equivalent analysis using the published datasets included in this manuscript: mouse thymus[3], human liver[2] and mouse gastrulation[4]. All timings are reported in minutes.

To assess the memory usage of the Milo workflow, we made use of the Rprof function in R to record the total amount of memory used at each step. We followed the same approach as above, downsampling simulated and published datasets from 1% to 100% of the total cell numbers. All memory usage is reported in MB.

For both the system timing and memory usage, we ran the simulated and published datasets, downsampling analyses on a single node of the high-performance computing cluster at the Cancer Research UK Cambridge Institute. Each node has 2× Intel Xeon E5-2698 2.20-GHz processors with 40 cores per node and 384 GB of DDR4 memory; cluster jobs were run using a single core.

Mouse thymus analysis. Single-cell data are available from ArrayExpress (accession E-MTAB-8560). Additional metadata were acquired from Baran-Gale et al.[3], including cluster identity and highly variable genes (HVGs). The dataset consists of 2,327 single TECs that passed quality control (see ref. [3] for details). Following the pre-processing steps from the original publication, we used log-normalized gene expression values as input, along with 4,906 HVGs, to estimate the first 50 PCs using a randomized PCA implemented in the R package irlba, the first 40 of which were used for KNN graph building and UMAP embedding. The refined sampling, using an initial random sampling of 30% of cells, identified 363 neighborhoods. Differential abundance testing used the mouse age as a linear predictor variable; thus, LFCs are interpreted as the per-week linear change in neighborhood abundance. Neighborhood cluster identity was assigned by taking the most abundant cluster identity among neighborhood cells.

Differential expression (DE) testing was performed on cells within neighborhoods containing a majority of cells from the intertypical TEC cluster. Neighborhoods were first aggregated into groups by constructing a neighborhood adjacency matrix, where the rows and columns represent the graph neighborhoods, and the elements of the matrix are the number of cells shared between each pair of neighborhoods. The adjacency matrix elements were then censored at 0, where the number of overlapping cells was <5 and where the difference in LFC between neighborhoods was >0.1. This adjacency matrix, representing a neighborhood graph, was then used as input to group neighborhoods using Louvain clustering. DE testing was performed comparing the log-normalized gene expression of neighborhood cells between the enriched and depleted abundant neighborhood groups from the larger intertypical TEC population (that is, Neighborhood Groups 3 and 4 versus 5; Fig. 4f) using a linear model implemented in the Bioconductor[44,45] package limma[46], using 5% FDR. Gene Ontology Biological Process term analysis was performed on the 407 DE genes (FDR = 10%) using the R package enrichR[47].

Droplet and SMART-seq scRNA-seq datasets were integrated using the MNN approach[19] implemented in the batchelor package function fastMNN ($k = 60$). Neighborhood group labels were transferred from the SMART-seq cells onto the droplet cells using the following procedure: (1) MNNs were identified between the two datasets in the integrated space (30 dimensions); (2) for each cell in the SMART-seq dataset, the neighborhood group label was transferred onto the corresponding set of 150 MNNs in the droplet scRNA-seq dataset; and (3) the frequency of each transferred neighborhood group label (Fig. 4i and Supplementary Fig. 6b) was then computed in each experimental replicate and age in the droplet scRNA-seq data ($n = 3$ mice per age).

Liver cirrhosis analysis. The dataset including cell type annotations was downloaded from https://datashare.is.ed.ac.uk/handle/10283/3433 (Gene Expression Omnibus accession no. GSE136103)[2]. The dataset comprises 58,358 cells, obtained from five healthy and five cirrhotic liver samples. We followed the pre-processing steps from the original publication. Namely, dimensionality reduction with PCA was performed on the 3,000 top HVGs, calculated using modelGeneVar and getTopHVGs from the R package scran[48], and the top 11 PCs were retained for KNN graph building and UMAP embedding. Refined sampling identified 2,676 neighborhoods ($k = 30$; initial proportion of sampled cells = 0.05). We ran Milo to test for differential abundance between cirrhotic and healthy experimental samples. To assign cell type annotations to neighborhoods, we took the most frequent annotation between cells in each neighborhood. Neighborhoods were generally homogeneous, with an average of 80% of cells belonging to the most abundant cell type label.

For the focused analysis on the endothelial and cholangiocyte lineages, DE testing was performed on the subset of neighborhoods from the selected lineage. Neighborhoods displaying significant differential abundance were aggregated into two groups based on similarity of LFC direction. DE testing was performed summing the gene expression counts for each experimental sample and neighborhood group between the more and less abundant groups using the QL test implemented in edgeR[11], using 5% FDR. Gene Ontology term analysis was performed on the significant DE genes using the R package clusterProfiler[49].

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Code availability

Milo is implemented as an open-source package in R (https://github.com/MarioniLab/miloR) and is installable from Bioconductor (≥3.13; http://www.bioconductor.org/packages/release/bioc/html/miloR.html). Code used to generate figures and perform analyses can be found at https://github.com/MarioniLab/milo_analysis_2020.

References

40. Luecken, M. D. & Theis, F. J. Current best practices in single-cell RNA-seq analysis: a tutorial. *Mol. Syst. Biol.* **15**, e8746 (2019).
41. Setty, M. et al. Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nat. Biotechnol.* **34**, 637–645 (2016).
42. Griffiths, J. & Lun, A. MouseGastrulationData: single-cell transcriptomics data across mouse gastrulation and early organogenesis. https://github.com/MarioniLab/MouseGastrulationData (2021).
43. Csardi, G. & Nepusz, T. The igraph software package for complex network research. *InterJournal* http://www.interjournal.org/manuscript_abstract.php?361100992 (2006).
44. Huber, W. et al. Orchestrating high-throughput genomic analysis with Bioconductor. *Nat. Methods* **12**, 115–121 (2015).
45. Gentleman, R. C. et al. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol.* **5**, R80 (2004).
46. Ritchie, M. E. et al. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.* **43**, e47 (2015).
47. Kuleshov, M. V. et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Res.* **44**, W90–W97 (2016).

48. Lun, A. T. L., McCarthy, D. J. & Marioni, J. C. A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor. *F1000Res.* **5**, 2122 (2016).
49. Yu, G., Wang, L.-G., Han, Y. & He, Q.-Y. clusterProfiler: an R package for comparing biological themes among gene clusters. *OMICS* **16**, 284–287 (2012).

Acknowledgements

We thank S. Ghazanfar for feedback on the method; N. Kumasaka for comments on the manuscript; C. Suo, V. Kedlian, R. Elmentaite, J. P. Pett, K. Tuong and B. Stewart for feedback on the software package; and D. Burkhardt, M. Luecken and W. Lewis for discussions on benchmarking. J.C.M. acknowledges core funding from the European Molecular Biology Laboratory and core funding from Cancer Research UK (C9545/A29580), which supports M.D.M. E.D. and S.A.T. acknowledge Wellcome Sanger core funding (WT206194). N.C.H. is supported by a Wellcome Trust Senior Research Fellowship in Clinical Science (ref. 219542/Z/19/Z), the Medical Research Council and a Chan Zuckerberg Initiative Seed Network Grant.

Author contributions

E.D., M.D.M. and J.C.M. conceived the method idea. E.D. and M.D.M. developed the method, wrote the code and performed analyses. E.D., M.D.M., S.A.T. and N.C.H. interpreted the results. E.D., M.D.M., S.A.T., N.C.H. and J.C.M. wrote and approved the manuscript. M.D.M. and J.C.M. oversaw the project.

Competing interests

In the last three years, S.A.T. has received remuneration for consulting and Scientific Advisory Board membership from Genentech, Roche, Biogen, ForesiteLabs and Qiagen. All other authors have no competing interests to declare.

Additional information

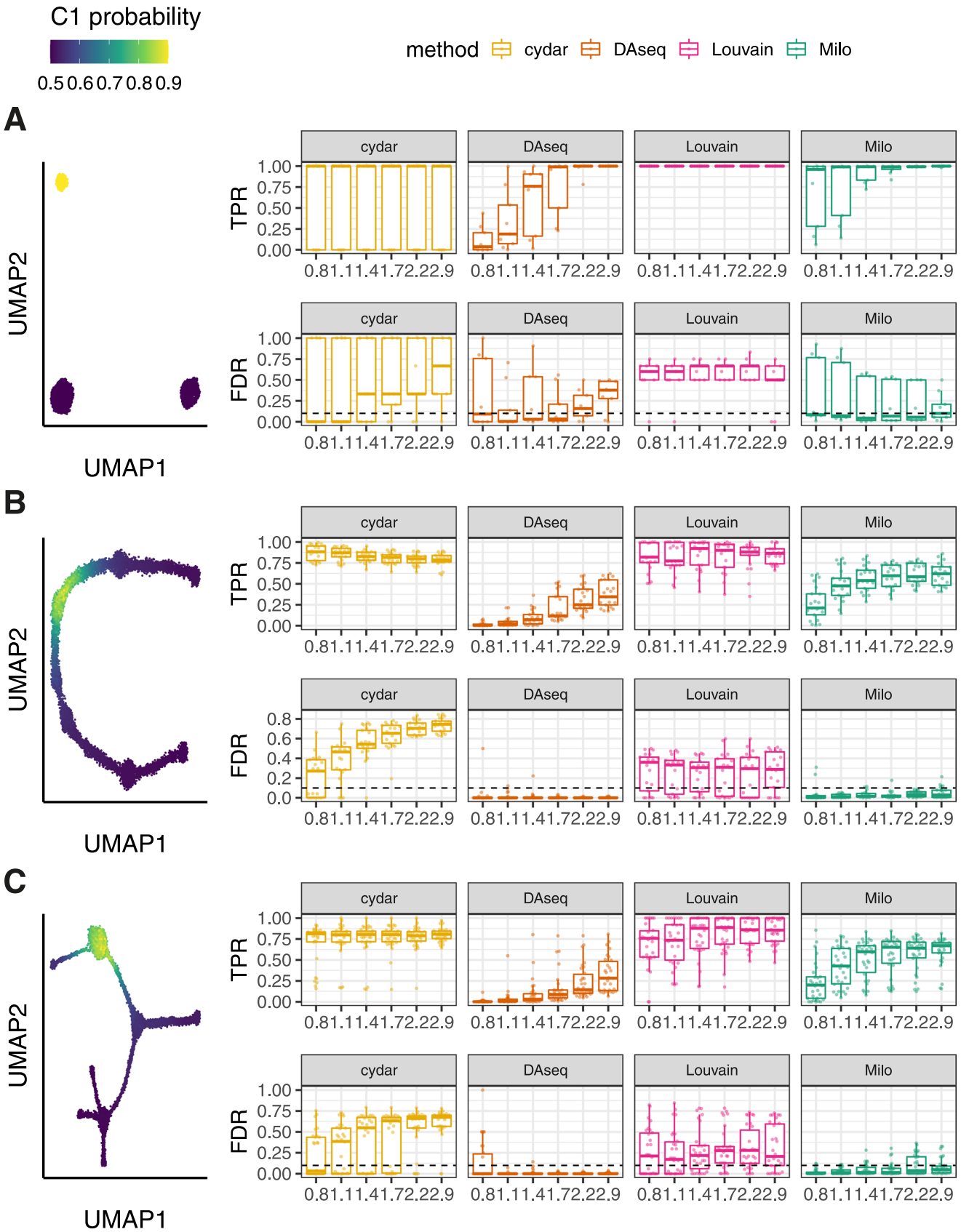
Extended data is available for this paper at https://doi.org/10.1038/s41587-021-01033-z.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41587-021-01033-z.

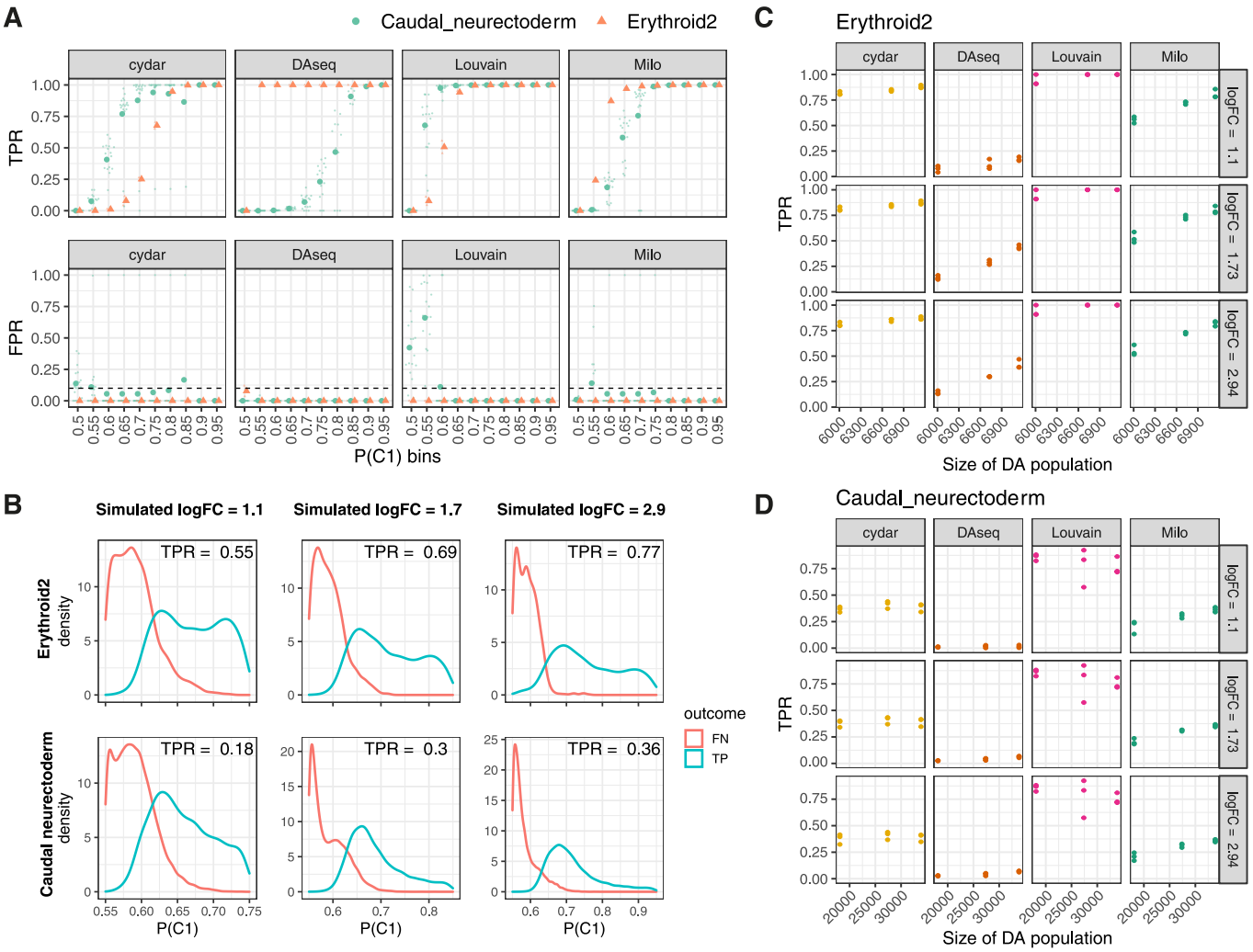
Correspondence and requests for materials should be addressed to Michael D. Morgan or John C. Marioni.

Peer review information *Nature Biotechnology* thanks Dana Pe'er, Michael Love and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

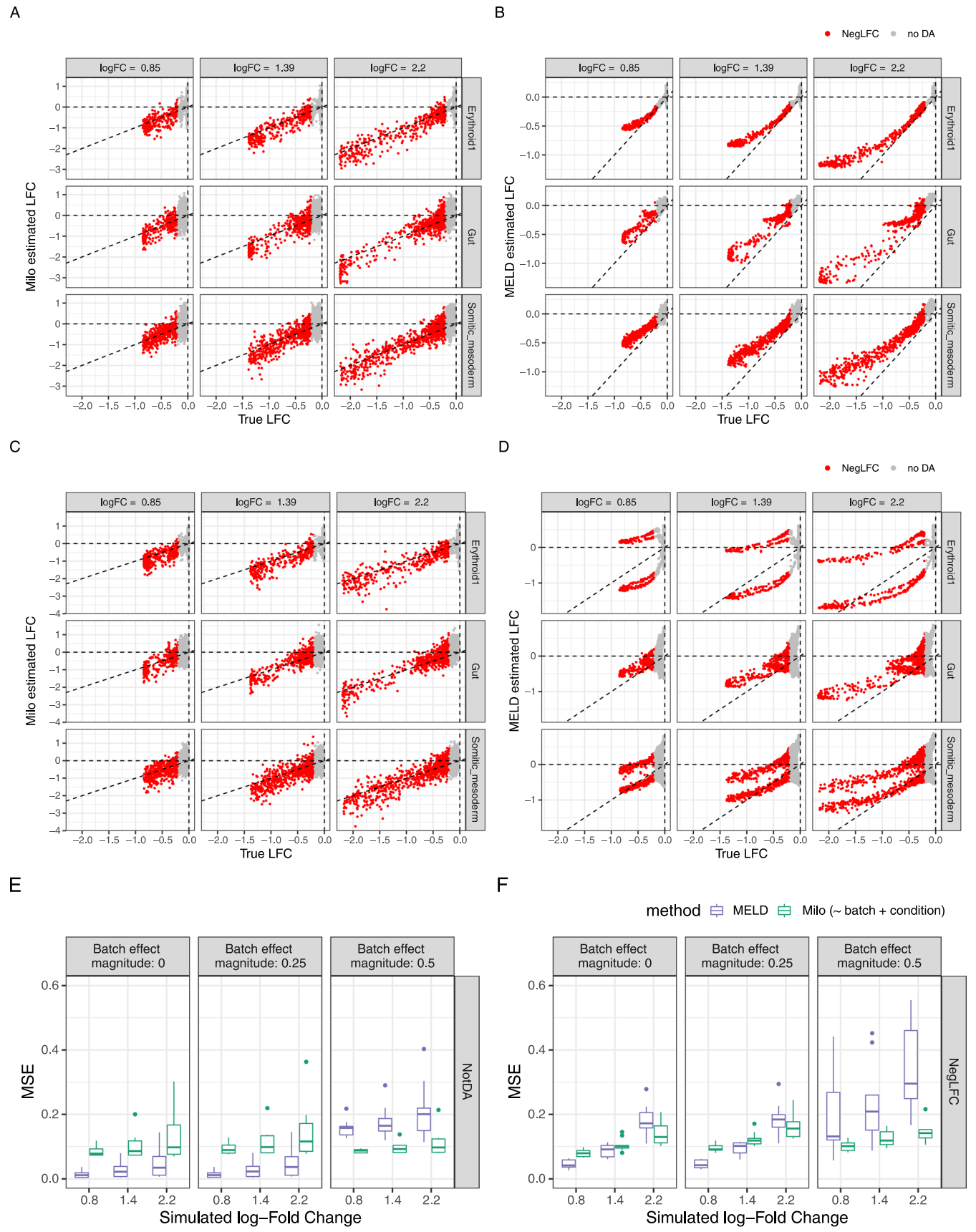
Reprints and permissions information is available at www.nature.com/reprints.



Extended Data Fig. 1 | Benchmarking DA methods on simulated data. DA analysis performance on KNN graphs from simulated datasets of different topologies: (**a**) discrete clusters (2700 cells, 3 populations); (**b**) 1-D linear trajectory (7500 cells, 7 populations); (**c**) Branching trajectory (7500 cells, 10 populations). Boxplots show the median with interquartile ranges (25-75%); whiskers extend to the largest value no further than 1.5x the interquartile range from the distance from the box, with outlier data points shown beyond this range.



Extended Data Fig. 2 | Sensitivity of DA methods to low fold change in abundance. (**a**) True positive rate (TPR, top) and false positive rate (FPR, bottom) of DA methods calculated on cells in different bins of P(C1) used to generate condition labels (bin size = 0.05, the number on the x-axis indicates the lower value in the bin). The results for 36 simulations on 2 representative populations (colors) are shown. The filled points indicate the mean of each P(C1) bin. (**b**) Variability in Milo power is explained by the fraction of true positive cells close to the DA threshold for definition of ground truth. Example distributions of 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. (**c**, **d**) True Positive Rate (TPR) of DA detection for simulated DA regions of increasing size centred at the same centroid (Erythroid2 (**c**) and Caudal neuroectoderm (**d**)). Results for 3 condition simulations per population and fold change are shown.



Extended Data Fig. 3 | See next page for caption.

Extended Data Fig. 3 | 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(pc′/(1 - pc′)). The neighbourhoods overlapping true DA cells (pc′ greater than the 75% quantile of P(C1) in the mouse gastrulation dataset) are highlighted in red. (**e**, **f**) Mean Squared Error (MSE) comparison for MELD and Milo for true negative neighbourhood (**e**) and true positive neighbourhoods (**f**), with increasing simulated log-Fold Change and magnitude of batch effect. Each boxplot summarises the results for *n*=27 simulations. Box plots show the median with interquartile ranges (25–75%); whiskers extend to the largest value no further than 1.5x the interquartile range from the distance from the box, with outlier data points shown beyond this range.

Extended Data Fig. 4 | Controlling for batch effects in differential abundance analysis. (**a**) 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. Each boxplot summarises results from simulations on *n*=9 populations. (**b**) 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 (*n*=24 simulations). Each panel represents a different DA method and a different simulated log-Fold Change. (**c**) Comparison of Milo performance *with* (~ batch + condition) or *without* (~ condition) accounting for the simulated batch in the NB-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). In all panels, boxplots show the median with interquartile ranges (25–75%); whiskers extend to the largest value no further than 1.5x the interquartile range from the distance from the box, with outlier data points shown beyond this range.

nature research

Corresponding author(s): John C. Marioni
Michael D. Morgan

Last updated by author(s): Jul 12, 2021

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	☐	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☐	☒	A description of all covariates tested
☐	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection: No software was used during data collection

Data analysis:

R (>= version 3.6.1) packages:
irlba (v2.3.1)
igraph (v1.2.4)
ggplot2 (v3.3.0)
DAseq (https://github.com/KlugerLab/DAseq)
ggraph (v2.0.1)
dyntoy (v1.0.0)
enrichR (v3.0)

miloR: https://github.com/MarioniLab/miloR

Bioconductor packages (Bioc 3.12):
edgeR (v3.32.0)
clusterProfiler (v3.17.5)
SingleCellExperiment (v1.12.0)
scran (v1.18.0)
batchelor (v1.6.0)
limma (v3.46.0)
MouseGastrulationData (v1.4.0)
Cydar (v1.14.1)

python (version 3.7.8) packages (pypi):
MELD

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

Mouse ageing thymus data can be downloaded from ArrayExpress (E-MTAB-8560), human liver cirrhosis data are available from Gene Expression Omnibus (GSE136103). Mouse gastrulation data are directly accessible from the Bioconductor package MouseGastrulationData (https://bioconductor.org/packages/devel/data/experiment/html/MouseGastrulationData.html)

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	No sample size calculations were performed. Sample sizes were determined based on the availability of datasets with at least 3 replicated measurements per condition
Data exclusions	No data were excluded
Replication	Biological replicates were used as determined by the original studies for real-world data. For simulations, single-cells were randomly assigned to replicate samples (n=3 per condition).
Randomization	Allocation of samples into experimental groups was determined by the original studies. No randomisation was performed as data were derived from observational studies.
Blinding	No blinding was performed for any analyses. Real-world data were derived from published observational studies, and simulations were generated by the research team.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging