

1 Supplementary information for **Differential cell-state**
2 **abundance testing using KNN graphs with *Milo***

3 Emma Dann, Neil C. Henderson, Sarah A. Teichmann, Michael D. Morgan,
4 John C. Marioni

5 27 March, 2021

6 **Contents**

7	1	Supplementary notes	2
8	1.1	Description of workflow for <i>Milo</i> analysis	2
9	1.1.1	Preprocessing and dimensionality reduction	2
10	1.1.2	Minimizing batch effects	2
11	1.1.3	Building the KNN graph	3
12	1.1.4	Definition of cell neighbourhoods and index sampling algorithm	4
13	1.1.5	Testing for differential abundance in neighbourhoods	4
14	1.2	Guidelines on parameter choice	7
15	1.3	Notes on experimental design	8

16 **2** **Supplementary Note Figures**

17 **References**

¹⁸ **1 Supplementary notes**

¹⁹ **1.1 Description of workflow for *Milo* analysis**

²⁰ Given a single-cell dataset of gene expression profiles of M cells collected from S experimental samples,
²¹ *Milo* aims to quantify systematic changes in the abundance of cells between biological conditions. Here we
²² provide a step-by-step description of the workflow for differential abundance analysis. Of note, we focus
²³ on the application to single-cell gene expression profiles, and we provide guidelines for pre-processing on
²⁴ this type of data. However, the core of the *Milo* framework, from KNN graph construction to differential
²⁵ abundance testing, is applicable to any kind of single-cell dataset that can be embedded in a low-dimensional
²⁶ space.

²⁷ **1.1.1 Preprocessing and dimensionality reduction**

²⁸ For pre-processing of scRNA-seq profiles we recommend following standard practices in single-cell analysis
²⁹ [1,2]: we normalize UMI counts by the total number of counts per cell, apply log-transformation and identify
³⁰ highly variable genes (HVGs). Then we project the $H \times M$ gene expression matrix, where M is the number
³¹ of cells and H is the number of HVGs, to the first d principal components (PCs). While downstream analysis
³² is generally robust to the exact choice of the number of HVGs [1], an optimal value for d can be selected by
³³ detecting the “elbow” in the variance explained by PCs or using the “jackstraw” method [3].

³⁴ **1.1.2 Minimizing batch effects**

³⁵ Comparing biological conditions often requires acquiring single-cell data from multiple samples, that can
³⁶ be generated with different experimental conditions or protocols. This commonly introduces batch effects,
³⁷ which can have a substantial impact on the data composition and subsequently the topology of any KNN
³⁸ graph computed across the single-cell data. Consequently, this will have an impact on the ability of *Milo*
³⁹ to resolve genuine differential abundance of cells between experimental conditions of interest. In addition,
⁴⁰ other biological nuisance covariates can impact DA analysis i.e. biological factors that are not of interest for
⁴¹ the analyst, such as donor of origin or sex of the donor. We recommend mitigating the impact of technical or
⁴² other nuisance covariates *before* building the KNN graph, by using one of the many *in silico* integration tools
⁴³ designed for this task in single-cell datasets. Defining the best tool for this task is beyond the scope of this

44 work; we refer the reader to a large number of integration methods that have been reviewed and benchmarked
45 in [4–6]. However, users should consider the type of output produced by their integration method of choice,
46 typically one of (A) a corrected feature space, (B) a joint embedding or (C) an integrated graph. The refined
47 neighbourhood search procedure in *Milo* relies on finding neighbors in reduced dimension space. Therefore
48 using a batch-correction method that produces an integrated graph (e.g. BBKNN [7], Conos [8]) may lead
49 to sub-optimal results in DA testing with *Milo*, as the refined neighbourhood search procedure would still
50 be affected by the batch effect.

51 In addition, the effect of nuisance covariates should be modelled in the generalized linear model used for DA
52 testing in *Milo* to minimize the emergence of false positives in case of imperfect batch correction (see Section
53 1.1.5) (Fig.2D, Supp Fig. 11).

54 We wish to emphasize that, in the presence of confounding factors, an appropriate experimental design is
55 crucial to obtain reliable results from differential abundance analysis: if nuisance factors are 100% confounded
56 with the biological condition used for differential abundance (e.g. if the samples from diseased and healthy
57 donors are processed in separate sequencing batches), there is no way to disentangle the abundance differences
58 that are truly driven by the biology of interest. In a similar case applying a batch integration strategy before
59 graph construction could lead to a loss of biological signal.

60 1.1.3 Building the KNN graph

61 *Milo* uses a KNN graph computed based on similarities in gene expression space as a representation of the
62 phenotypic manifold in which cells lie. While *Milo* can be used on graphs built with different similarity
63 kernels, here we compute the graph as follows: given the reduced dimension matrix X_{PC} of dimensions
64 $M \times d$, for each cell j , the Euclidean distances to its K nearest neighbors in X_{PC} are computed and stored in
65 a $M \times M$ adjacency matrix D . Then, D is made symmetrical, such that cells c_i and c_j are nearest neighbors
66 (i.e. connected by an edge) if either c_i is a nearest neighbor of c_j or c_j is a nearest neighbor of c_i . The
67 KNN graph is encoded by the undirected symmetric version \tilde{D} of D , where each cell has at least K nearest
68 neighbors.

69 **1.1.4 Definition of cell neighbourhoods and index sampling algorithm**

70 Next, we identify a set of representative cell neighbourhoods on the KNN graph. We define the neighbourhood
71 n_i 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 with
72 $i = 1, 2, \dots, N$ as the index cell of the neighbourhood, so that $N \leq M$. Formally, a cell c_j belongs to
73 neighbourhood n_i if $\tilde{D}_{i,j} > 0$.

74 In order to define neighbourhoods that span the whole KNN graph, we sample index cells by using an
75 algorithm previously adopted for waypoint sampling for trajectory inference [9,10]. Briefly, we start by
76 randomly sampling $p \cdot M$ cells from the dataset, where $p \in [0, 1]$ (we use $p = 0.1$ by default). Given the
77 reduced dimension matrix used for graph construction X_{PC} , for each sampled cell c_j we consider its K
78 nearest neighbors with PC profiles x_1, x_2, \dots, x_k and compute the mean position of the neighbors in PC space

79 \bar{x} :

$$\bar{x}_j = \frac{\sum_k x_k}{K}$$

80 Then, we search for the cell c_i such that the Euclidean distance between x_i and \bar{x} is minimized. Because the
81 algorithm might converge to the same index cell from multiple initial samplings, this procedure yields a set
82 of $N \leq p \cdot M$ index cells that are used to define neighbourhoods.

83 Having defined a set of N neighbourhoods from the sampled index cells, we construct a count matrix of
84 dimensions $N \times S$ which reports, for each sample, the number of cells that are present in each neighbourhood.

85 **1.1.5 Testing for differential abundance in neighbourhoods**

86 To test for differential abundance between biological conditions, *Milo* models the cell counts in neighbour-
87 hoods, estimating variability across biological replicates using a generalized linear model (GLM). We build
88 upon the framework for differential abundance testing implemented by *Cydar* [11]. In this section, we briefly
89 describe the statistical model and adaptations to the KNN graph setting.

90 **Quasi-likelihood negative binomial generalized linear models** We consider a neighbourhood n with
91 cell counts y_{ns} for each experimental sample s . The counts are modelled by the negative binomial (NB)
92 distribution, as it is supported over all non-negative integers and can accurately model both small and large
93 cell counts. For such non-Normally distributed data we use generalized-linear models (GLMs) as an extension

94 of classic linear models that can accomodate complex experimental designs. We therefore assume that

$$y_{ns} \sim NB(\mu_{ns}, \phi_n),$$

95 where μ_{ns} is the mean number of cells from sample s in neighbourhood n and ϕ_n is the NB dispersion
96 parameter.

97 The expected count value μ_{ns} is given by

$$\mu_{ns} = \lambda_{ns} M_s$$

98 where λ_{ns} is the proportion of cells belonging to experimental sample s in n and M_s is the sum of counts
99 of cells of s over all the neighbourhoods. In practice, λ_{ns} represents the biological variability that can be
100 affected by treatment condition, age or any biological covariate of interest.

101 We use a log-linear model to model the influence of a biological condition on the expected counts in the
102 neighbourhood:

$$\log \mu_{ns} = \sum_{g=1}^G x_{sg} \beta_{ng} + \log M_s \quad (1)$$

103 Here, for each possible value g taken by the biological condition of interest, x_{sg} is the vector indicating
104 the condition value applied to sample s . β_{ng} is the regression coefficient by which the covariate effects are
105 mediated for neighbourhood n , that represents the log fold-change between number of cells in condition g
106 and all other conditions. If the biological condition of interest is ordinal (such as age or disease-severity) β_{ng}
107 is interpreted as the per-unit linear change in neighbourhood abundance.

108 Estimation of β_{ng} for each n and g is performed by fitting the GLM to the count data for each neighbourhood,
109 i.e. by estimating the dispersion ϕ_n that models the variability of cell counts for replicate samples for each
110 neighbourhood. Dispersion estimation is performed using the quasi-likelihood method in `edgeR`[12], where
111 the dispersion is modelled from the GLM deviance and thereby stabilized with empirical Bayes shrinkage,
112 to stabilize the estimates in the presence of limited replication.

113 **Count model normalisation and compositional biases** In equation (1) above the $\log M_s$ term is
114 provided as an offset to the NB GLM which effectively normalises the cell counts in each neighbourhood by
115 the total number of cells in each sample S , thus accounting for variation in cell numbers across samples. If

116 there is a single strong region of differential abundance then the counts for these samples will increase, which
 117 can negatively bias the model log fold-change estimates. This results in an underestimate of the true log
 118 fold-changes and the appearance of false discoveries in the opposite direction to the true DA effect direction.
 119 To address this issue we turn to the RNA-seq literature, specifically the trimmed mean of M-values (TMM)
 120 method for estimating normalisation factors that are robust to such compositional differences across samples
 121 [13]. Under the assumption that the majority of neighbourhoods are not differentially abundant, the TMM
 122 approach first computes the per-neighbourhood log count ratios for a pair of samples s and s' (M values):

$$M_n = \log \frac{Y_{ns}/M_s}{Y_{ns'}/M_{s'}}$$

123 And the absolute neighbourhood abundance (A values):

$$A_n = \frac{1}{2} \log_2(Y_{ns}/M_s \cdot Y_{ns'}/M_{s'}), \text{ for } Y_n \neq 0$$

124 Both the M and A distribution tails are trimmed (30% for M, 5% for A by default) before taking a weighted
 125 average over neighbourhoods using precision weights, computed as the inverse variance of the neighbourhood
 126 counts, to account for the fact that more abundant neighbourhoods have a lower variance on a log scale.
 127 Thus, the normalisation factors are computed, with respect to a reference sample, r :

$$\log_2(TMM_s^{(r)}) = \frac{\sum_{n \in N} w_{ns}^r M_{ns}^r}{\sum_{n \in N} w_{ns}^r}$$

128 where, M_{ns}^r is computed as above for samples s and r , and:

$$w_{ns}^r = \frac{M_s - Y_{ns}}{M_s Y_{ns}} + \frac{M_r - Y_{nr}}{M_r Y_{nr}}$$

129 In practice, M_r and Y_{nr} are computed from the sample with the counts per million upper quartile that is
 130 closest to the mean upper quartile across samples.

131 **Adaptation of Spatial FDR to neighbourhoods** To control for multiple testing, we need to account for
 132 the overlap between neighbourhoods, that makes the differential abundance tests non-independent. We apply
 133 a weighted version of the Benjamini-Hochberg (BH) method, where p-values are weighted by the reciprocal

134 of the neighbourhood connectivity, as an adaptation to graphs of the Spatial FDR method introduced by
135 *Cydar* [11]. Formally, to control for FDR at a selected threshold α we reject null hypothesis i where the
136 associated p-value is less than the threshold:

$$\max_i p_{(i)} : p_{(i)} \leq \alpha \frac{\sum_{l=1}^i w_{(l)}}{\sum_{l=1}^n w_{(l)}}$$

137 Where the weight $w_{(i)}$ is the reciprocal of the neighbourhood connectivity c_i . As a measure of neighbourhood
138 connectivity, we use the Euclidean distance between the neighbourhood index cell c_i and its kth nearest
139 neighbour in PC space.

140 1.2 Guidelines on parameter choice

141 In this section we provide practical guidelines to select default parameters for KNN graph and neighbourhood
142 construction for DA analysis with Milo. We recognize that DA analysis will also be impacted by choices made
143 during feature selection and dimensionality reduction. However these depend strongly on the nature of the
144 single-cell dataset used as input. For example feature selection strategies suitable for UMI-based scRNA-seq
145 data might be suboptimal for data generated with non-UMI protocols, or dimensionality reduction methods
146 alternative to PCA might be used for single-cell epigenomics data. We point the reader to existing resources
147 and heuristics for the application to scRNA-seq in section 1.1.1.

148 **Selecting the number of nearest neighbors K** For construction of the KNN graph and neighbourhoods, the user has to select the number of nearest neighbors K to use for graph construction. The choice of K influences the distribution of cell counts within neighbourhoods, as K represents the lower limit in the number of cells in each neighbourhood ($\sum(y_{n,s})$). Hence, if K is too small the neighbourhoods might not contain enough cells to detect differential abundance. As we illustrate by testing for DA with increasing values for K in the mouse gastrulation dataset with synthetic condition labels (Supp Note Fig 1A-B) increasing K increases power, but can come at the cost of FDR control. In order to perform DA testing with sufficient statistical power, the analyst should consider the number of experimental samples S (that will correspond to the columns in the count matrix for DA testing) and the desired minimum number of cells per neighbourhood and experimental sample. The median number of cells per sample in each neighbourhood \hat{y}_{ns} increases with the total neighbourhood size (Supp Note Fig 1C), with:

$$\hat{y}_{ns} \sim \frac{\sum_s y_{ns}}{S}$$

159 Therefore a conservative approach to minimize false positives is to select $K \geq S \times 3-5$.
 160 We recommend users to inspect the histogram of neighbourhood sizes after sampling of neighbourhoods
 161 (Supp Note Fig 1D) and to consider the number of cells that would be considered a “neighbourhood”
 162 in the dataset at hand. As a heuristic for selecting a lower bound on K to increase the resolution of
 163 neighbourhoods for capturing rare sub-populations or states, the user can select K such that the mean
 164 neighbourhood size is no more than 10% of the expected size of the rare population. We provide the utility
 165 function `plotNhoodSizeHist` to visualize the neighbourhood size distribution as part of our R package.

166 **Selecting the proportions of cells sampled as neighbourhood indices p** The proportion of cells
 167 sampled for search of neighbourhood indices can affect the total number of neighbourhoods used for analysis,
 168 but this number will converge for high proportions thanks to the sampling refinement step described in section
 169 1.1.4 (Supp Fig. 1A). In practice, we recommend initiating neighbourhood search with $p = 0.05$ for datasets
 170 with more than 100k cells and $p = 0.1$ otherwise, which we have found to give appropriate coverage across
 171 the KNN graph while reducing the computational and multiple-testing burden. We recommend selecting
 172 $p > 0.1$ only if the dataset appears to contain rare disconnected subpopulations.

173 1.3 Notes on experimental design

174 Of key consideration when designing any single-cell experiment is how the sample collection relates to
 175 the biological variables of interest, and how these samples are processed and experiments are performed.
 176 Moreover, the experimenter (and analyst together), should design their experiment to minimise the impact
 177 of confounding effects on differential abundance testing, and incorporate appropriate replication to achieve
 178 enough power to detect the expected effect size for their experiment.

179 **Statistical power considerations** Increases in statistical power can be achieved by several means: (1)
 180 Increased cell numbers in neighbourhoods and (2) higher signal-to-noise ratio. The first can be achieved
 181 by collecting more cells for each sample, increasing K during graph building such that neighbourhoods are
 182 on average larger, and by increasing the number of replicate samples. Collecting more cells gives a greater

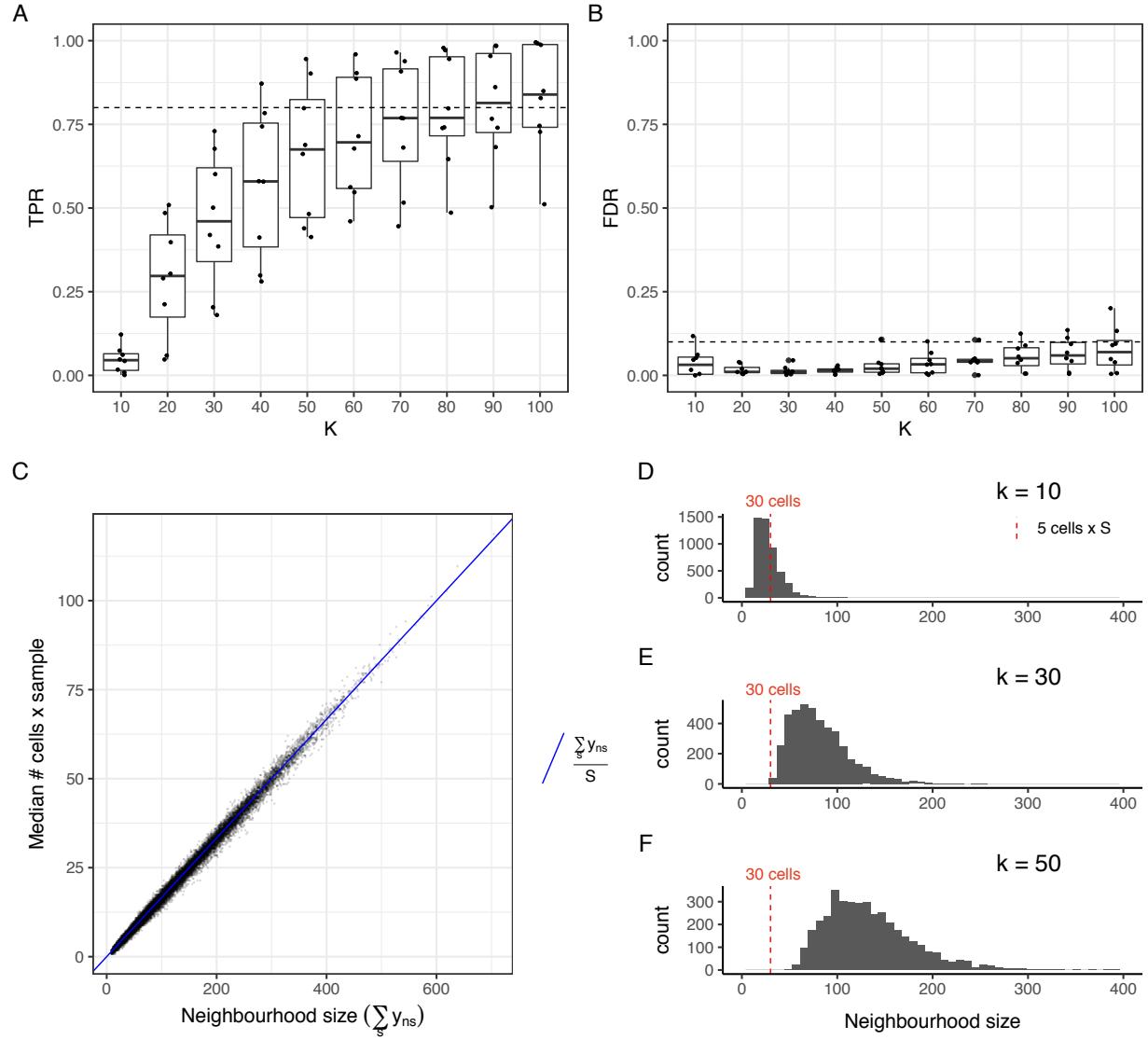
183 coverage of the cell-to-cell heterogeneity and different cell states/types, including increased detection for
184 rarer sub-populations. Increasing K increases power by constructing larger neighbourhoods, however, this
185 increase in power comes at a cost of reduced sensitivity for rarer sub-populations and an increased false
186 discovery rate (Supp Note Fig 1A-B). Designing an experiment with more replicate samples has multiple
187 benefits in terms of increasing statistical testing power, increasing the signal-to-noise ratio, and increasing the
188 accuracy of effect size estimates (Suppl Fig 8B). Therefore, in order of their impact on power and differential
189 abundance testing, we would recommend: (1) collecting more replicate samples, with a minimum of n=3,
190 (2) collecting more cells per sample, (3) increasing K to generate larger neighbourhoods.

191 **Batch effects and experimental design** Proper experimental design is crucial for answering scientific
192 questions, particularly in the presence of confounding effects. In single-cell experiments these can range
193 from batch effects introduced between samples processed on different days, owing to logistical constraints
194 or sample availability, to biological sample collections from a heterogenous population; the latter being
195 particularly apparent for genetically diverse non-model organisms.

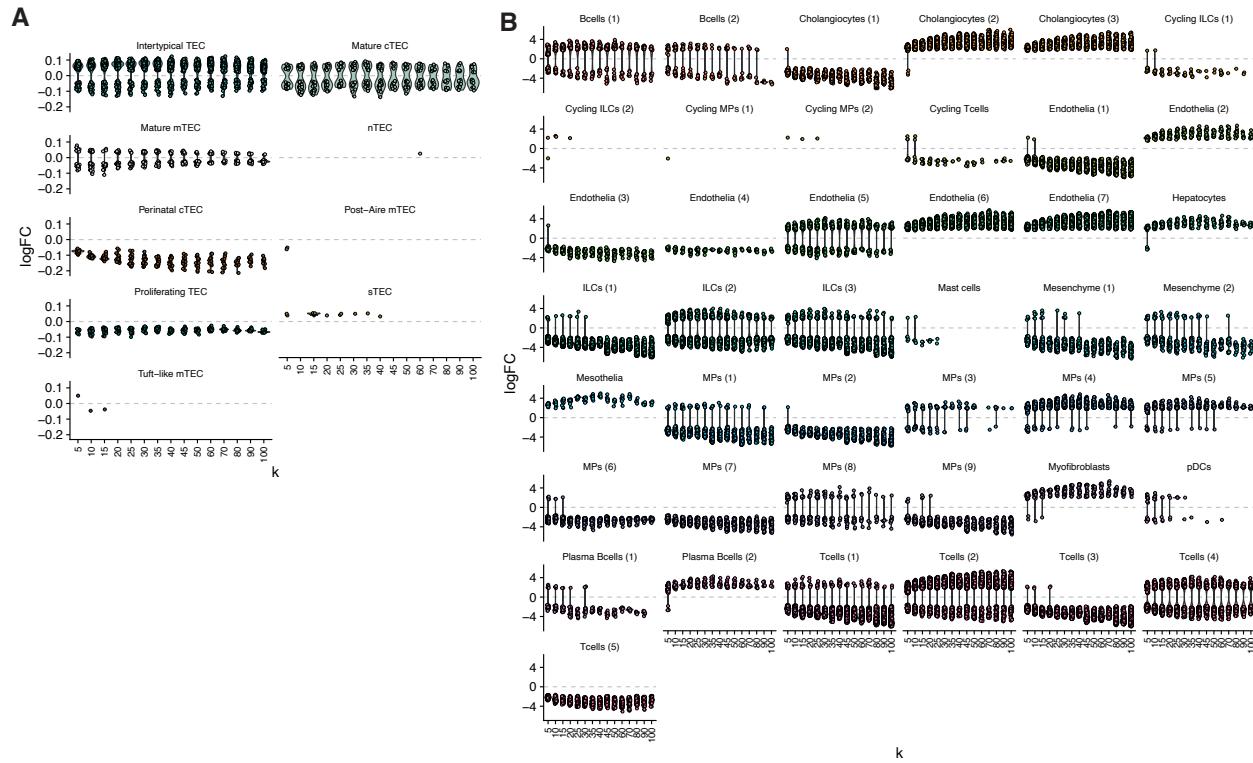
196 In the context of differential abundance testing with Milo, we recommend designing experimental procedures
197 and sample processing such that samples from different conditions are randomised across batches. One ex-
198 ample is to pair samples between conditions, such that during batch effect removal the variability between
199 these pairs of samples is minimally removed. This will help to facilitate removal of technical batch effects,
200 whilst retaining the relevant biological variability.

201 As described above, the exact choice of batch integration method should be carefully considered before ap-
202 plying Milo, with a preference for methods that generate a batch-integrated space (either reduced dimensions
203 or gene expression). The key point is that sample processing and experimental batches are not perfectly
204 confounded with the biological variable of interest. We expect *some* technical variability to remain (no batch
205 integration is perfect), which can be handled in Milo's GLM framework by including the batch identity as a
206 blocking factor in the design model. Examples of this correction are shown in the benchmarking in Fig. 2E
207 and supp Fig. 11.

208 **2 Supplementary Note Figures**



Supplementary Note Figure 1: **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 $\sum_s y_{ns}$ 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 Note Figure 2: **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.

209 **References**

- 210 1. Luecken, M.D., and Theis, F.J. (2019). Current best practices in single-cell RNA-seq analysis: A
211 tutorial. *Molecular Systems Biology* *15*, e8746.
- 212 2. Amezquita, R.A., Lun, A.T.L., Becht, E., Carey, V.J., Carpp, L.N., Geistlinger, L., Marini, F.,
213 Rue-Albrecht, K., Risso, D., Soneson, C., *et al.* (2020). Orchestrating single-cell analysis with
Bioconductor. *Nature Methods* *17*, 137–145.
- 214 3. Chung, N.C., and Storey, J.D. (2015). Statistical significance of variables driving systematic variation
215 in high-dimensional data. *Bioinformatics* *31*, 545–554.
- 216 4. Luecken, M.D., Büttner, M., Chaichoompu, K., Danese, A., Interlandi, M., Mueller, M.F., Strobl,
D.C., Zappia, L., Dugas, M., Colomé-Tatché, M., *et al.* (2020). Benchmarking atlas-level data
217 integration in single-cell genomics. *bioRxiv*, 2020.05.22.111161.
- 218 5. Chazarra-Gil, R., Dongen, S. van, Kiselev, V.Y., and Hemberg, M. (2020). Flexible comparison of
219 batch correction methods for single-cell RNA-seq using BatchBench. *bioRxiv*, 2020.05.22.111211.
- 220 6. Tran, H.T.N., Ang, K.S., Chevrier, M., Zhang, X., Lee, N.Y.S., Goh, M., and Chen, J. (2020). A
221 benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biology*
21, 12.
- 222 7. Polański, K., Young, M.D., Miao, Z., Meyer, K.B., Teichmann, S.A., and Park, J.-E. BBKNN: Fast
223 batch alignment of single cell transcriptomes. *Bioinformatics*.
- 224 8. Barkas, N., Petukhov, V., Nikolaeva, D., Lozinsky, Y., Demharter, S., Khodosevich, K., and
Kharchenko, P.V. (2019). Joint analysis of heterogeneous single-cell RNA-seq dataset collections.
225 *Nat Methods* *16*, 695–698.
- 226 9. Gut, G., Tadmor, M.D., Pe'er, D., Pelkmans, L., and Liberali, P. (2015). Trajectories of cell-cycle
227 progression from fixed cell populations. *Nature Methods* *12*, 951–954.
- 228 10. Setty, M., Tadmor, M.D., Reich-Zeliger, S., Angel, O., Salame, T.M., Kathail, P., Choi, K., Bendall,
S., Friedman, N., and Pe'er, D. (2016). Wishbone identifies bifurcating developmental trajectories
229 from single-cell data. *Nature Biotechnology* *34*, 637–645.

- 230 11. Lun, A.T.L., Richard, A.C., and Marioni, J.C. (2017). Testing for differential abundance in mass
231 cytometry data. *Nature Methods* *14*, 707–709.
- 232 12. Robinson, M.D., McCarthy, D.J., and Smyth, G.K. (2010). edgeR: A Bioconductor package for
233 differential expression analysis of digital gene expression data. *Bioinformatics* *26*, 139–140.
- 234 13. Robinson, M.D., and Oshlack, A. (2010). A scaling normalization method for differential expres-
235 sion analysis of RNA-seq data. *Genome Biology* *11*, R25. Available at: <http://genomebiology.biomedcentral.com/articles/10.1186/gb-2010-11-3-r25> [Accessed March 18, 2021].