science.sciencemag.org/cgi/content/full/science.aaw3381/DC1

Supplementary Materials for

Lineage tracing on transcriptional landscapes links state to fate during differentiation

Caleb Weinreb*, Alejo Rodriguez-Fraticelli*, Fernando D. Camargo†, Allon M. Klein†‡

*These authors contributed equally to this work.
†These authors contributed equally to this work.
‡Corresponding author. Email: Allon_Klein@hms.harvard.edu

Published 23 January 2020 on *Science* First Release
DOI: 10.1126/science.aaw3381

This PDF file includes:

Other Supplementary Material for this manuscript includes the following:
(available at science.sciencemag.org/cgi/content/full/science.aaw3381/DC1)

MATERIALS AND METHODS

1 Data availability

Data is available at https://bit.ly/2z2F8jX

2 Cell culture and lentiviral barcoding

2.1 Barcoded lentiviral library synthesis and amplification

The pLARRY vector was constructed by DNA synthesis and Gateway cloning (Vectorbuilder) using a protocol loosely adapted from (*29*) and (*30*). The barcoded linker was created by annealing two DNA primers (forward, 5'-CCC CGGATCCAGACATNNNNCTNNNNACNNNNTCNNNNGTNNNNTGNNNNCANNNNCATATGAGCAAT CCCCACCCTCCCACCTAC-3'; reverse, 5'-GTAGGTGGGAGGGTGGGGATTGCT-3'; IDT DNA). N was a hand mix of 25% A, 25% C, 25% T and 25% G. Primers (10 pmoles of each) were mixed in 50 µl 1× NEB buffer 4 (New England Biolabs). After heating the mixture for 5 minutes at 95°C, the primers were allowed to anneal down to 37ºC gradually decreasing temperature (0.5ºC/minute). Then, 1U of Klenow DNA polymerase (3'-5' Exonuclease mutant) and 50 nmoles of dNTPs was added to the mixture and incubated for 2 hours at 37ºC. After Klenow inactivation for 20 minutes, the barcoded linker was then digested with a mixture of NdeI and BamHI (New England Biolabs) and ligated into the *NdeI-Bam*HI site of the pLARRY vector at 3:1 ratio. The resulting ligation mix was purified by columns (NucleoSpin Gel and PCR Clean-up, Macherey-Nagel) and transformed into 10-beta electroporation ultracompetent *Escherichia coli* cells (New England Biolabs) and grown overnight on LB plates supplemented with 50 µg/mL ampicillin (Sigma-Aldrich). From 8 plates, ~0.5-1×10^6 colonies were pooled by flushing plates with LB supplemented with 50 µg/mL ampicillin. After 6h of culture, plasmid DNA was extracted with a Maxiprep endotoxin-free kit (Macherey-Nagel). The pLARRY vector map and plasmid, as well as a sample of the library will be made available through Addgene before publication.

2.2 Lentivirus production and barcode labeling

Barcoded GFP plasmid and lentivirus packaging components (https://www.addgene.org/browse/article/836/) were transfected into HEK293T cells using the Lipofectamine 3000 kit (Thermofisher). Lentivirus was harvested every 12 hours for 72 hours and concentrated using ultra-centrifugation at 80,000g. HEK cells were grown in DMEM with 10% fetal bovine serum (FBS) and 1% PenStrep but switched to OptiMEM (Thermofisher) with 10% FBS for transfection and lentivirus harvest. Hematopoietic progenitor cells (HPCs) were transduced using spin infection (800g for 90 minutes) in virus concentrate with DAEA-dextran hydrochloride (Sigma).

2.3 Estimate of barcode diversity

Though the theoretical diversity of the barcode is 4^{28}, the actual diversity is much lower since each barcode must be transformed, amplified and harvested from a bacterial colony to appear in the final library. We estimated the overall diversity of our library be resampling 10,000 unique barcodes from the two main datasets in this paper (the in vitro and in vivo datasets described below) and asking how many barcodes were shared between these samples. If T is the total number of unique barcodes in the library, then (for T>>10,000) the expectation for the number of shared barcodes N is

$$N \approx \frac{10000^2}{T} \quad \text{hence} \quad T = \frac{10000^2}{N}.$$

We found N=241±13 barcodes shared between the resampled groups of 10,000, implying T= 414,811±22,194, or approximate 0.5 x 10^6 as noted in the main text. This calculation sets a lower bound on barcode diversity by assuming that all barcodes are uniformly represented.

2.4 Cell isolation for state-fate experiments

After euthanasia, bone marrow from femur, tibia, pelvis and sternum was isolated by crushing with pestle and mortar to obtain all cells. Collected bone marrow cells were filtered through a 40-µm strainer and washed in cold 'Easy Sep' buffer (PBS; 2% fetal bovine serum (FBS); 1 mM EDTA; Pen/Strep). Red blood cells were then lysed using ammonium chloride (STEMCELL) and mature lineage cells were depleted magnetically using the EasySep Lineage Depletion Kit (STEMCELL). The resulting Lin- fraction was stained for Kit (CD117-PE, clone 2B8, Biolegend), Sca-1 (Ly6a-FITC, clone D7, Biolegend) and lineage markers (antibody mix from the same EasySep kit), and Lin-Kit+Sca-1+ (LSK) or Lin-Kit+ (LK) cells were isolated by flow activated cell sorting (FACS) on a Sony SH800 with a 130uM nozzle.

2.5 Cell culture for state-fate experiment in vitro

In three separate in vitro experiments, sorted LK (two experiments) or LSK cells (one experiment) were barcoded as described above and then plated in round-bottom 96-well plates in media designed to support pan-myeloid differentiation, consisting of StemSpan media (STEMCELL), Pen/Strep, IL-3 (20ng/mL), FLT3-L (50ng/mL), IL-11 (50ng/mL), IL-5 (10ng/mL), EPO (3U/mL), TPO (50ng/mL) (Peprotech), and mSCF (50ng/mL) and IL-6 (10ng/mL) (R&D Systems). The number of cells plated varied from 5,000 – 10,000. After two days in culture, cells were split evenly, with one half being sequenced immediately and the other half re-plated in two separate culture wells. The decisions of how many cells to plate and how long to culture before the first sequencing step represent key tradeoffs in experimental design, and the choices made here are described below. After two further days in culture, cells were again split, with 30% taken for inDrops and the remainder re-plated in 6 wells. After two further days in culture, GFP+ cells were sorted using FACS and profiled using inDrops. For the latter pair of inDrops experiments, cells were dissociated from the well by 5-minute treatment with 2.5% Trypsin (Thermofisher).

In carrying out the experiment described above, our main goal was to maximize the yield of clones detected both at the initial (day 2) time point, and also at later time points after several days of differentiation. The probability that a clone appears in both time points depends on three factors: (1) the probability P_{split} that when cells are split for sequencing vs. re-plating, members of the clone are physically present in both fractions; (2) the probability $P_{detect\ early}$ that cells in the fraction designated for immediate sequencing are actually detected; (3) the probability $P_{detect\ late}$ that cells in the re-plated fraction survive cell culture and appear in late timepoint dataset. For N initially barcoded cells, the final yield is proportional to

$$P_{split}\ P_{detect\ early}\ P_{detect\ late} N$$

When choosing the amount of time to culture barcoded cells before initially splitting them, there is a trade-off between the quality and quantity of data. A longer culture time results in higher P_{split}, hence more data. But during the culture period, sister cells diverge in gene expression space, thus decreasing the extent to which clones sampled over time resemble single-cell trajectories. We chose 2 days as the minimum culture period that provided a reasonable yield of state-fate clones.

When choosing the number of cells N for initial barcoding, the key consideration is to avoid having so many clones that they far exceed the capacity for sequencing them, thus resulting in low values of $P_{detect\ early}$ and $P_{detect\ late}$. For example, doubling N but halving the sampling probabilities would result in the same number of total cells sequenced but halve the yield of state-fate clones. We choose 5,000-10,000 cells for initial plating based on a capacity to sequence ~100,000 cells and on considerations of proliferation rate in culture (~1-2 cell cycles per day) as well as the efficiency of barcoding (30%) and ability to isolate barcoded cells by FACS.

2.6 Cell culture for state-fate experiment in vivo

Cell isolation, barcoding and culturing for 2 days was carried out as in the in vitro experiment (above), but with a very stringent Sca1 gate to enrich for stem cells. After 2 days in culture, 20% of the cells were analyzed by inDrops scSeq, and the rest were transplanted retro-orbitally into ten C57BL/6J sub-lethally irradiated host mice (500 cGy) as described in (*10*). After one and two weeks, the transplanted cells were harvested from total bone marrow and spleen (six mice on week 1, four mice on week 2). Gr1-biotin (biolegend) and anti-biotin (Miltenyi) magnetic MACS beads were used to partially deplete Neu/Mo lineage cells (~10-fold depletion) and all remaining cells were labeled with Gr1-eFluor450 and Streptavidin-eFluor450 (to label remaining Gr1 cells), Ter119-PE/Cy5, cKit-APC and CD19/B220-APC/Cy7. All GFP+ cells were collected in their entirety, with the exception of Ter119+ cells, which were only sampled at 1/2 of the total (~2-fold depletion). Thus in final cell sample profiled by indrops included ~10% of Gr+ cells and 50% of Ter119+ cells and 100% of all other subsets.

2.7 Techniques for preventing cell loss

At all steps during cell culture, extreme care was taken to prevent loss of cells, since the number of barcodes shared between time points is highly dependent on maximizing the yield of profiled GFP+ cells from the total that are in culture. To minimize cell loss: all spins were performed at 500g for 5 minutes in 1.5mL tubes in a swinging-bucket centrifuge; any transfer of cells out of a well involved several washes of the well with PBS; and washing steps, including those for inDrops and re-plating, were performed using PBS with 0.5% BSA. Since measuring cell density with a hemocytometer involves loss of cells, all experiments were accompanied by an identical 'decoy' copy of the experiment carried out in parallel that was used to assess cell density, thus leaving the 'real' experimental sample unperturbed by cell counting.

3 Single-cell encapsulation and data pre-processing

3.1 Single-cell encapsulation and library preparation for sequencing

For single-cell RNA sequencing (scSeq), we used inDrops (*31*) following the protocol described in (*32*), with a modification to allow targeted sequencing of the LARRY barcode. In brief, single cells were encapsulated into 3-nl droplets with hydrogel beads carrying barcoding reverse transcription primers. After reverse transcription in droplets, the emulsion was broken and the bulk material was taken through: (i) second strand synthesis; (ii) linear amplification by in vitro transcription (IVT); (iii) amplified RNA fragmentation; (iv) reverse transcription; (v) PCR. To specifically amplify barcode-containing GFP transcripts, we followed a protocol similar to (*26*). We split the amplified RNA fraction (after step (ii)) and used one half for standard library preparation and the other half for targeted lineage barcode enrichment. To target the barcode, we modified the subsequent steps of library prep by (i) skipping RNA fragmentation; (ii) priming reverse transcription using a transcript specific primer at 10mM (TGAGCAAAGACCCCAACGAG); (iii) introducing an extra PCR step using a targeted primer (8 cycles using Phusion 2X master mix; Thermofisher; primer sequence = TCG TCG GCA GCG TCA GAT GTG TAT AAG AGA CAG NNN Ntaa ccg ttg cta gga gag acc atat) and 1.2X bead purification (Agencourt AMPure XP). Targeted and non-targeted final libraries were pooled at equimolar ratios for initial sequencing. Subsequent re-sequencing was performed on non-targeted libraries alone.

3.2 Read alignment and cell filtering

FASTQ sequence files were demultiplexed and aligned to the GRCm38 mouse reference genome using the inDrops pipeline (https://github.com/indrops/indrops), generating cell-by-gene counts tables for each experiment and condition. Cells were filtered to include only abundant barcodes on the basis of visual inspection of the histograms of total transcripts per cell (see Table 1 for UMI thresholds for each sample, as well as statistics reporting the median UMIs and median genes detected for each cell). The data were further filtered to eliminate putatively stressed or dying cells, defined by having >20% of transcripts coming from mitochondrial genes. We then applied the SCRUBLET algorithm (https://github.com/AllonKleinLab/scrublet; Wolock et al., Cell Systems 2018) to identify putative doublet cells, i.e. cell barcodes that likely represented the combination of two or more actual cells. SCRUBLET produces a 'doublet score' for each cell, and these scores were thresholded based on manual inspection, being set in each case to eliminate apparent doublets – defined by co-expression of mature marker genes for different lineages – while retaining the maximum number of apparent non-doublets. Doublet filtering eliminated ~10% of cells per dataset, in accordance with an experimentally estimated doublet rate of 5-10%. Cells within each experiment were then normalized to have the same total number of transcripts for all subsequent analyses.

3.3 Calling of lineage barcodes

To call lineage barcodes, we began with an intermediate output of the indrops pipeline: a list of reads with annotated cell barcode and unique molecular identifier (UMI). From this list, we extracted all (Cell-BC, UMI, lineage-BC) triples that were supported by at least 10 reads, collapsed all Lineage-BC's within a hamming distance of 3 using a graph-connected-components based algorithm, and carried forward the (Cell-BC, Lineage-BC) pairs supported by at least 3 UMIs. To call clones, we then applied a set of rigorous filtering steps: (i) Cells with the exact same set of barcodes were classified as clones; (ii) Pairs of cells in separate sequencing libraries with the same Cell-BC and Lineage-BC were discarded, since statistically these could only arise from instability of the droplet emulsion; (iii) Clones that were statistically over-abundant within a single sequencing library compared to their frequency in other libraries of the same sample were discarded, since these could also only arise through emulsion instability. These steps have been implemented in a pipeline available online: https://github.com/AllonKleinLab/LARRY.

4 Data visualization and cell type annotation

4.1 Definition of terms: "PCA-batch-correction" and "graph-smoothing"

Here we define terms that will be used repeatedly in the following methods sections. PCA-batch-correction is a method for co-embedding separate scSeq datasets in a way that minimizes their global or 'batch' differences. One dataset is chosen as a 'reference' and used to establish a set of principal components that define the embedding space. Both datasets are embedded in this principal component space and then carried forward for subsequent analysis.

We use "graph-smoothing" to refer to a tunable method for local averaging in a single-cell dataset, which we adapted from (*23*). The input is a graph in which cells are nodes and edges represent similarity between cells, as well as a vector of values for each cell which is to be smoothed. In each of N iterative rounds of smoothing, cells adopt a weighted average of their graph neighbors' values and their own value, where the weight on their own value is β. (Formally, this can be understood to implement a diffusion process with a source term at each cell that preserves the local information, and the weighted average preserves density). To be precise, at each smoothing iteration, cells update their value to be $(1-\beta)V_n \; + \beta V_s$ where V_n is the average value of their graph neighbors and V_s is their own value. In the following Methods sections, we use $\beta = 0.1$ and report the value of N when smoothing is invoked. The choice of β = 0.1 is intended to allow values to diffuse while maintaining their overall localization in the graph. In general, β and N are partially redundant, in the sense that higher values of β decrease the amount of smoothing, while higher values of N increase it. Where smoothing plays a key role in an analysis, we have tested that the key results hold across a broad range of N values (see Supp Fig 11c and Supp Fig 12a,b)

4.2 Generation of SPRING plot layouts

We used SPRING (*18*) [see https://github.com/AllonKleinLab/SPRING_dev] for single-cell data visualization. The default SPRING pipeline was used in all cases except two, where we elaborated on the default pipeline (see below). In each case, we began with total-counts-normalized gene expression data, filtered for highly variable genes using the SPRING gene filtering function ("filter_genes" from https://github.com/AllonKleinLab/SPRING_dev/blob/master/data_prep/spring_helper.py using parameters (85 ,3, 3)), and further filtered to exclude cell cycle correlated genes – defined as those with correlation R>0.1 to the gene signature defined by Ube2c, Hmgb2, Hmgn2, Tuba1b, Ccnb1, Tubb5, Top2a, and Tubb4b. Data was then reduced to 50 principal components and a k-nearest neighbor graph was constructed with k=4. A 2D layout was then generated using the ForceAtlas2 algorithm with 500 steps.

For two of the largest SPRING plots, we could not directly follow the standard pipeline because the large number of cells made running ForceAtlas2 prohibitively slow. We therefore initialized the layout with a subset of cells, and then extended the layout to the remaining cells, as follows. For the in vitro data (Fig 1e), we sampled 40,000 and generated a SPRING layout in the usual way. Coordinates for the remaining ~90,000 cells were learned by allowing each cell to choose its 40 nearest neighbors from among the initial 40,000 and then take on the average position of those neighbors. For the post-transplantation data (Fig 1j) we subsampled 50,000 and generated a SPRING plot as usual. Coordinates for the remaining ~110,000 cells were learned by allowing each cell to choose its 25 nearest neighbors from among the initial 50,000 and then take on the average position of those neighbors.

4.3 Annotation of cell types in vitro (Fig 1e)

Mature cell types in vitro were manually identified on the basis of marker gene expression (see Table 2). Annotation was restricted to cells that had high expression of the respective markers and belonged to a region of the SPRING graph that was well separated from alternative lineage branches. We took care to ensure that the manual selection process did not affect the conclusions of the paper by performing it 'blindly' – before any subsequent analyses were done – and not changing the annotations from that point onward. We have posted annotations for all cells online (see Data Availability).

4.4 Annotation of cell types in vivo (Fig 1j)

For cells post-transplantation, we used clustering rather than manual selection, since the late stage of the cells (1-2 weeks of differentiation) resulted in well-separated mature states. Clusters were generated using Louvain-Jaccard clustering (the default implementation in SPRING was used; see https://github.com/AllonKleinLab/SPRING_dev) and labeled on the basis of marker gene expression (see Supp Fig 3).

5 Sister cell similarity on day 2

5.1 Clustering analysis of sister cell similarity (Fig 2b and Supp Fig 4b,c)

To assess the similarity of sister cells on day 2, we clustered day 2 cells from the in vitro dataset and computed the fraction of sister cell pairs that appear in each pair of clusters. Clustering was performed using either K-means applied to the SPRING coordinates (Fig 2b; K=20 clusters) or spectral clustering applied to PCA coordinates (Supp Fig 4b; K=20 and K=40). For each of these clusterings, we computed a probability matrix P, defined by:

$$P_{ij} = \text{probability that a cell is in cluster } j \text{ given that its sister is in cluster } i$$

To calculate the proportion of sisters falling in the same or nearest N clusters (Supp Fig 4c), we calculated the nearness of each pair of clusters based on the SPRING graph as follows

$$\text{Nearness of cluster } i \text{ and cluster } j = \frac{\text{Number of edges between cluster } i \text{ and cluster } j}{(\text{Size of cluster } i) * (\text{Size of cluster } j)}$$

5.2 Gene expression correlation analysis of sister cell similarity (Supp Fig 4d)

We computed the correlation of gene expression for pairs of sisters in the in vitro state-fate dataset, restricting to highly variable genes (n=3844 genes). For comparison, we repeated the analysis for cells that were nearest neighbors in PCA space. A size-matched set of nearest neighbors was constructed by randomly sampling cells and then identifying the cell's nearest neighbor. Supp Fig 4d shows the distribution of correlation values for clonal sister pairs and nearest-neighbor pairs compared to random pairs of cells from day 2.

6 Regulator discovery

6.1 Identifying genes correlated with fate choice in vitro (Fig 2g)

To discover early regulators of fate choice in vitro, we first restricted analysis to the most immature cells, defined by high expression of the progenitor marker Cd34. Since Cd34 expression is sparse and noisy, we used a smoothened map of Cd34 expression and then selected the early cells as those with a smoothened score above a set cutoff. Smoothing was performed on the SPRING graph with N=10 iterations, and a cut-off of 1 (units of smoothed UMIs) was used to select immature cells. The choice of smoothing parameter N and threshold establish the extent of the CD34+ domain selected. An overly permissive parameter combination (high N or low threshold) lead to inclusion of fate-committed cells, while an overly strict threshold leads to restriction to a narrow subset that excludes putatively uncommitted cells, including regions where multiple clone fate potentials overlap, and where cells express additional markers of immaturity such as Sca1 and Kit. Because the threshold and smoothing parameter N both act to control the size of the selected region, we fixed N=10 and then adjusted the threshold according to the criteria described above.

Having defined a region of immaturity, we proceeded to detect genes correlated with fate choice in this region. For each immature day 2 cell with a sister on days 4 or 6, we classified whether any of its day 4 or 6 sisters belonged to each respective lineage. This produced, for each lineage, two groups of day 2 cells – those with a later (day 4/6) sister in the lineage and those without. Differential expression between these two groups was performed using a T-test applied to log-transformed transcript counts (a pseudo-count of 1 was used for the log transform), and the resulting p-values were adjusted for multiple hypotheses using the Benjamini–Hochberg correction.

6.2 Identifying genes correlated with fate choice in vivo (Fig 2j)

All day 2 cells were used for regulator discovery in vivo. Differential expression analysis was performed as described in the preceding paragraph.

6.3 Analysis of functional gene categories and enriched gene ontology terms

We asked whether the set of genes enriched among the progenitors of each cell type belonged to specific functional categories. We obtained lists of genes for specific molecular functions from the Mouse Genome Informatics (MGI) Gene Ontology Browser (http://www.informatics.jax.org/vocab/gene_ontology/) (see Table 4 for ontology term names and IDs). The intersection between each molecular function gene list and the set of progenitor-enriched genes was obtained, and fold-enrichment was calculated as ratio between the percentage progenitor-enriched genes in a given list compared to the expected percentage for a size-matched random sample from the genome. P-values were assigned using a binomial test of proportion. All results are reported in Table 4.

In addition to molecular function, we asked whether progenitor-enriched genes represented specific biological processes. The set of all enriched genes (Benjamin-Hochberg adjusted p-val < 0.05) for each cell type progenitor in vivo were submitted for gene ontology enrichment to the PANTHER classification system (*33*) (http://geneontology.org/; “biological process” category). Enriched terms with FDR < 0.001 are reported in Table 5 (a maximum of 20 terms is reported for each cell type).

7 Comparison of fate bias in vitro and in vivo

7.1 Mapping between in vivo and in vitro cell state manifold (Supp Fig 6)

To understand how the fate of cells differs in vivo and in vitro, we mapped day 2 cells from the in vivo experiment to the SPRING plot of day 2 cells from the in vitro experiment (Supp Fig 6a) and also directly compared how cells in a similar transcriptional state differed in their output (Supp Fig 6b).

To map cells in SPRING, day 2 cells from the in vivo experiment were projected using principal components calculated from day 2 in vitro cells, and then chose their 5 nearest neighbors from among the day 2 in vitro cells in 50-dimensional PCA space (Euclidean distance) PCA was performed for the in vitro data as described in Supp Methods 4.2 for SPRING visualization. A 2D projection (Supp Fig 6a) was computed by assigning each day 2 cell in vivo the average (x,y) coordinates its 5 nearest neighbors from the in vitro dataset.

To directly compare the shift in observed fate among transcriptionally similar cells from each dataset, we projected the day 2 in vivo cells using PCA as above, and then identified their 100 nearest neighbors in the in vitro dataset. For each cell fate in vivo, we selected the day 2 cells that exclusively gave rise to that fate, and then recorded the average in vitro fate distribution for the union of their respective 100 nearest neighbors. The result of this average produces an N x M matrix where N is the number of in vitro fates, M is the number of in vivo fates, and the (i,j)-th entry is the average fraction of progenitors with fate-i in vitro are transcriptionally similar to progenitors with fate-j in vivo. A null distribution of such matrices was computed by shuffling the fates of day 2 cells in vivo, and the Z-score of the real matrix with respect to this null is shown in Supp Fig 6b.

8 Clonal fate under cytokine perturbations

8.1 Experimental design and analysis (Supp Fig 7)

To investigate how cell fate shifts under perturbation, we barcoded HSPCs and then profiled them after culture in several cytokine conditions. Lin-Kit+Sca1- and Lin-Kit+Sca1+ cells were harvested as described in Supp Methods 2.4, combined them in proportions 33% Lin-Kit+Sca1+, 66% Lin-Kit+Sca1- (the native proportions are 10%/90%) and then barcoded and plated them in a broad multilineage-supporting cytokine cocktail, as described in Supp Methods 2.5 (n=20,000 initially plated cells). On day 2, 25% of cells were profiled for inDrops and the remaining cells were split evenly into four groups and re-plated in the following conditions respectively: (i) continuation of the same cytokine mixture; (ii) M-CSF (20ng/mL) in StemSpan media; (iii) G-CSF (20ng/mL) in StemSpan; (iii) EPO (3U/mL) in StemSpan. Pen/strep was included in the media for all conditions. Cells from all conditions were harvested at day 4 and day 6 for profiling by indrops. Bioinformatic processing of transcriptomes and barcodes was performed as described in Supp Methods 3. Cell type annotation was performed as described in Supp Methods 4.3.

9 Fate prediction with machine learning

9.1 Assessing predictability of cell fate from day 2 gene expression state (Fig 3a,b)

A panel of machine learning methods was used to test the predictability of cell fate from day 2 gene expression. Each method was tasked with predicting a categorical variable **Y**, defined as the dominant fate outcome (most common fate among all day 4 and 6 sisters), from a vector **X**, representing total-counts normalized gene expression for the day 2 cell, restricting to a defined set of genes. The choice of assigning each cell a single dominant fate is motivated by the maturity of machine learning approaches for predicting categorical variables, and by the fact that 85% of cells gave rise to a single lineage in vitro, and 97% gave rise to clones where >50% of cells belonged to a single lineage (comparable figures for the in vivo data are 72% and 94%). Several gene sets for **X** were tested: all highly variable genes, a curated list of transcription factors (n=1181, obtained from Riken Transcription factor database(*34*): http://genome.gsc.riken.jp/TFdb/tf_list.html), a sized match random gene set (n=1181), and the set of all differentially expressed genes (see Table 3 and Methods section “Regulator discovery”). The highly variable genes and differentially expressed genes were distinct in vivo and in vitro. Two machine learning methods were applied to each gene set in vivo and in vitro: logistic regression and multilayer perceptron (neural network). We used implementations from the python package sklearn (version 0.19.2). The parameters used for each function are also shown below. All parameters were default except for the “hidden_layer_sizes” and “alpha” parameters of the MLPClassifier, and the “C” parameter of the LogisticRegression classifier, which were chosen through a hyperparameter scan (Supp Fig 8a-f). The hyperparameter scans were performed for a single train-test split of the data, and the values achieving the greatest performance were used for the remaining train-test splits.

```
sklearn.neural_network.MLPClassifier(hidden_layer_sizes=[(100, 20) in vitro;
(200,20) in vivo], activation=’relu’, solver=’adam’, alpha=[0.72 in vitro; 10 in vivo], batch_size=’auto’,
learning_rate=’constant’, learning_rate_init=0.001, power_t=0.5, max_iter=200, shuffle=True,
random_state=None, tol=0.0001, verbose=False, warm_start=False, momentum=0.9,
nesterovs_momentum=True, early_stopping=False, validation_fraction=0.1, beta_1=0.9, beta_2=0.999,
epsilon=1e-08)
```

```
sklearn.linear_model.LogisticRegression(penalty=’l2’, dual=False, tol=0.0001, C=[0.008 in
vitro;  0.004 in vivo], fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None,
solver=’warn’, max_iter=100, multi_class=’warn’, verbose=0, warm_start=False, n_jobs=None)
```

For each classifier and condition, we performed 100 splits of the data into training and testing groups and computed accuracy as the fraction of correct predictions for dominant fate. Fig 3a,b show the mean and standard deviation of accuracy across 100 trials. To focus on prediction of uncommitted states, analysis of the in vitro data was restricted to LSK cells.

In addition to predicting the dominant fate for each clone, we also predicted the distribution of fates for each clone (Supp Fig 8g-j). Thus, the outcome variable **Y** was a normalized vector indicting the proportion of mature cells in the clone belonging to each lineage. We restricted this analysis to the in vitro data since the models were observed to overfit the sparser vivo dataset. Training, testing and choice of hyperparameters was carried out as above, with the following differences: (1) To measure accuracy, we used the mean-square deviation between the predicted and measured clonal proportions; (2) We used machine learning methods that can predict continuous rather than categorical variables: `sklearn.linear_model.Ridge` and `sklearn.neural_network.MLPRegressor`.

9.2 Analyzing predictability at different stages of differentiation (Supp Fig 8k-n)

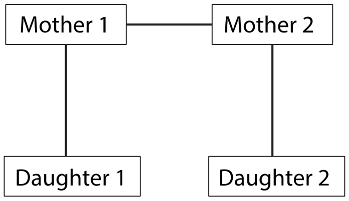
To understand how the predictability of cell fate varies across stages of differentiation, we assessed the accuracy of fate prediction from machine learning at a single-cell level. The dominant fate of each day 2 cell learned from the expression of all highly variable genes using logistic regression, exactly as described in Supp Methods 9.1, with the modification that prediction in vitro applied to all cells, rather than just LSK cells. Across 500 trials, cells were randomly divided into train and test sets. Each cell was then assigned an overall prediction accuracy, defined as the proportion of correct predictions across all trials where it belonged to the test set (Supp Fig 8k,m). To visualize how prediction accuracy varies across stages of differentiation, cells were clustered and colored by the average prediction accuracy for all cells in their cluster (Supp Fig 8l,n). Clustering was performed using spectral clustering applied to the SPRING graph for the in vitro and in vivo datasets. The number of clusters was set to k=20; this choice of parameter only affects visualization of the results.

10 Analysis of 'hidden properties' that influence fate choice

We used our combined state and lineage data to ask whether there are hidden properties, such as chromatin state or protein abundance, that influence cell fate but are not evident in the measurable transcriptome. In the language of probability, these properties would cause the trajectories of cells through gene expression space to be non-Markovian (*4*). The analysis tests for evidence of non-Markovian dynamics. Below, we present the theoretical justification for our approach, and then describe its practical application to data.

10.1 Theoretical approach for analysis of hidden properties (Fig 4c,f)

Our approach to detect hidden properties is motivated by the data processing inequality (DPI), which formalizes the idea that in a Markov chain, information about the starting state can decrease over time but never increase. Formally, the DPI states that in a Markovian sequence of random variables $X \rightarrow Y \rightarrow Z$, the following inequality holds: $MI(X,Y) \geq MI(X,Z)$, where MI denotes mutual information.



Supplementary Figure 13: Graphical model showing dependencies between clonally related cells at different time

To apply the DPI in the context of our experiment, we consider four random variables that represent the transcriptomes of two starting sister cells ("Mother 1" and "Mother 2") and their differentiated progeny at a later time point ("Daughter 1", "Daughter 2") (Supp Fig 13). If cell differentiation were Markovian with respect to scSeq measurements, then these four variables would form a Markov random field with respect to the graphical model shown in Supp Fig 13, meaning that any pair of non-adjacent nodes are conditionally independent given the nodes that connect them. Intuitively, this means (for example) that the relationship between "Daughter 1" and "Daughter 2" is entirely mediated by their shared relationship with "Mother 1". Applying the DPI, the Markov assumption would imply

$$MI(\text{Daughter 1}, \text{Daughter 2}) \leq MI(\text{Mother 1}, \text{Daughter 2})$$

Thus, a violation of the above inequality, by contradicting the assumption of Markovity, would imply the existence of hidden properties. Due to the high-dimensional nature of scSeq data, as well as the continuity of early progenitor cell states, we found that calculating mutual information directly was impractical. Therefore, we took the alternative approach of applying machine learning algorithms to ask how well "Daughter 1" and "Mother 1" could respectively predict the state of "Daughter 2". The intuition is that higher prediction accuracy would imply greater shared information about state.

10.2 Data analysis for assessment of hidden properties in vitro (Fig 4e)

Following the theoretical discussion above, the existence of hidden properties could be inferred from the following inequality:

$$\text{Accuracy("Daughter 1" predicting "Daughter 2")} \geq \text{ Accuracy("Mother 1" predicting "Daughter 2")}$$

Since most machine learning algorithms perform discrete classification based on continuous, high-dimensional input data, we specifically assessed how well the gene expression states of "Daughter 1" and "Mother 1" could respectively predict the discrete lineage identity of "Daughter 2". In our data, "Mother 1" and "Mother 2" would refer to clonally related cells from day 2, and "Daughter 1" and "Daughter 2" would refer to clonally related cells found in different wells on day 6 (the cells need to have been plated in separate wells on day 2 to ensure that their most recent common ancestor existed before day 2). Note that the terms "mother" and "daughter" are used for simplicity to refer to the early and late time points, but there is no requirement that mother cells divide.

We constructed training sets for each machine learning algorithm as follows. For each clone that appeared in two separate wells on day 6, we recorded the most common fate of the clone in one well, and the average transcriptome of the clone in the other well. Likewise, for each clone that appeared on day 6 and day 2, we recorded the most common fate of the day 6 cells and the average transcriptome of the day 2 cells. In each case, we then asked how well the most common fate could be predicted from the average transcriptome by applying a panel of machine learning algorithms (see below). We performed 100 splits into training and testing sets, and show the average accuracy of each algorithm in Fig. 4e, where accuracy is defined as the fraction of correct guesses for most common fate. Analysis of the in vitro data was restricted to LSK cells, and to lineages with adequate statistical power (>2% of mature cells, including megakaryocytes, mast cells, basophils, neutrophils, monocytes, and lymphoid cells).

For machine learning algorithms, we applied the following functions from the python 2.7 module sklearn (version 0.19.2). The parameters used for each function are also shown below. Parameters for the MLPClassifier and LogisticRegression classifier are described in Methods section 9.1.

```
sklearn.ensemble.RandomForestClassifier(n_estimators='warn', criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None)
```

```
sklearn.neighbors.KNeighborsClassifier(n_neighbors=5, weights='uniform', algorithm='auto', leaf_size=30, p=2, metric='minkowski', metric_params=None, n_jobs=1, **kwargs)
```

```
sklearn.naive_bayes.GaussianNB(priors=None)
```

10.3 Data analysis for assessment of hidden properties in vivo (Fig 4e)

Analysis of hidden variables in the in vivo data was identical to that in vitro, with the following accommodations for the altered experimental design. First, whereas the information theoretic approach for quantifying hidden variables assumes that cells have been cultured in two separate wells, the in vivo experiment involved transplantation into ten separate mice. Therefore, we partitioned the mice into two groups (A and B) and compared forward prediction from day 2 cells to group A cells to lateral prediction from group B to group A cells. Partitioning into A and B was repeated randomly 50 times. Second, because there were only 9 clones that appeared in multiple mice at the two-week time point, versus 69 at the one-week time point, we restricted the analysis to cell fates at one week. Finally, as in the in vitro analysis, we restricted to common lineages (>2% of mature cells, including neutrophils, monocytes, erythrocytes, MPPs, and dendritic cells).

10.4 Analysis of functional purity of scSeq-defined cell states (Fig 3 i-l, Supp Fig 9)

We define an equipotent, or functionally pure state, as one in which each cell has the same set of fate probabilities when marginalizing over interactions with the environment. In theory, the fate probability of a cell could be estimated by allowing it to divide several times before fate commitment, and then observing the independent fate choice of each daughter cell. A simple observation of the abundance of clonally-derived cell types in a single well does not represent the clonal fate probabilities of the clone founder cell, since division can (and usually does) occur after fate commitment. The functional purity of scSeq-defined cell states can however be assessed by analyzing clones that appear both on day 2 and in two separate wells on days 4/6, since the outcomes in the two wells are independent.

Specifically, for each two-way fate choice (between, say, fates A and B) we classified the clones that appeared in two wells on days 4/6 as either

- **AA** if only cells of type A were produced in both wells
- **BB** if only cells of type B were produced in both wells
- **AB** if only type A cells appeared in well 1 and only type B cells appeared in well 2
- **BA** if only type B cells appeared in well 1 and only type A cells appeared in well 2
- **OTHER** If any combination of fates occurred that is not covered above

Let N_{AA} denote the number of "AA" type clones, and likewise for the other types. Let $T = N_{AA} + N_{AB} + N_{BA} + N_{BB}$ be the total number of clones and define $p_A = (2N_{AA} + N_{AB} + N_{BA})/T$ as the overall probability of generating type A cells and likewise for p_B. The following relation holds for clones from a functionally pure starting state:

$$E(N_{AB} + N_{BA}) = 2p_A p_B T$$

This result holds because, by conditioning on the outcome where only a single fate is produced in each well, we may define p_A as the phenomenological probability that the fate is A and likewise for p_B. Because differentiation in each well is independent, if all cells are equipotent then $P(AB \text{ or } BA) = 2p_A p_B$, which completes the argument.

If the number of clones with different outcomes in each well falls below expectation, then fate choice is correlated between sister cells in isolated wells when compared to all cells with co-clustered (transcriptionally-similar) day 2 progenitors of the clone, implying that the starting cells were already biased toward one outcome or the other, hence not functionally pure.

We produced an ensemble of clustering of the day 2 fate space using Louvain clustering on the SPRING graph (https://python-louvain.readthedocs.io/en/latest/) with a resolution parameter ranging between 0.6 and 4.8 in increments of 0.2 (Supp Fig 9a). Each cluster from each clustering, we analyzed clones that appeared on two wells on days 4/6 and had a day 2 cell in the cluster. If there were fewer than 10 such clones, the cluster was discarded. We then counted the number of clones in each type, and computed the observed $(N_{AB} + N_{BA})$ as well as the expected $(2p_A p_B T)$ number of clones with two different fates in each well. These are compared in Supp Fig 9b-d. The two fate outcomes "A" and "B" were defined for three different fate choices as

- A=Neu, B=Mo
- A=[Er,Mk,Ma,Ba], B=[Neu,Mo]
- A=[Er,Mk,Ma,Ba,Neu,Mo], B=[Ly,DC]

11 Multiple paths of monocyte differentiation

11.1 Classifying Neu-like and DC-like monocytes (Fig 4c)

We used graph smoothing to classify monocytes as DC-like or Neu-like for differential expression and clonal progenitor analysis. First a raw score was computed for each monocyte as the proportion of its non-monocyte sisters in the neutrophil fate minus the proportion of its non-monocyte sisters in the DC fate. Cells with no barcode or no sisters in either fate received a score of 0. This raw score was then smoothed as described in Supp Methods 4.1 with N=10 rounds of smoothing. Cells with a smoothed score above the median were classified as Neu-like, and those with a score below the median were classified as DC-like.

11.2 Differential expression analysis and functional enrichment in vitro (Fig 4d,e)

Genes enriched among DC-like or Neu-like monocytes (Fig 4e; Table 7), or the day 2 progenitors of these subsets (Fig 4d; Table 6) were identified by differential gene expression analysis. Gene expression values for each highly variable gene were compared using a t-test, with the resulting p-values corrected using the method of Benjamini-Hochberg. When comparing the mature monocytes, analysis was restricted to genes that were highly variable among monocytes. When comparing day 2 progenitors, highly variable genes among all day 2 cells were used. The set of all enriched genes (adjusted p-val < 0.05) for each comparison were submitted for gene ontology enrichment to the PANTHER classification system (*33*) (http://geneontology.org/; "biological process" category).

11.3 Cell-type shared barcode analysis (Fig 4f)

Analysis of clonal relationships between mature fates in vivo was performed exactly as described in Supp Methods 15.1 and presented in Fig 5p, with the modification that all DC fates (CD11 cDCs, CD8 cDCs, pDCs and migDCs) were grouped together as one fate called "DC".

11.4 Differential expression between Neu-related and DC-related monocytes in vivo (Fig 4g,f; Supp Fig 10e)

To investigate phenotypic heterogeneity among monocytes after transplantation, we calculated for each monocyte the fraction of its non-monocyte sisters that were neutrophils ($\mathrm{frac}_{\mathrm{Neu}}$; Supp Fig 10e, left) and the fraction that were DCs ($\mathrm{frac}_{\mathrm{DC}}$; Supp Fig 10e right). Monocytes were then classified as Neu-related if $\mathrm{frac}_{\mathrm{Neu}} > \mathrm{frac}_{\mathrm{DC}}$ and as DC-related if $\mathrm{frac}_{\mathrm{DC}} > \mathrm{frac}_{\mathrm{Neu}}$. Differential expression analysis between Neu-related and DC-related monocytes was performed using a t-test applied to each highly variable gene with Benjamini/Hochberg correction of p-values (Fig 4g). This analysis resulted in a group of genes enriched in DC-related cells (N=6 at FDR<0.05) and a group of genes enriched in Neu-related cells (N=6 at FDR<0.05). The expression of these genes was visualized on a SPRING plot of post-transplant monocytes using a signature score computed from each group respectively (Fig 4h). The signature score was calculated by Z-scoring the gene expression values for each gene, and then adding the Z-scored expression values for all genes in the group.

11.5 Gene expression heterogeneity of monocytes in steady-state hematopoiesis (Fig 4i; Supp Fig 10f,g)

To investigate the heterogeneity of monocytes in unperturbed stead-state hematopoiesis, we investigates two populations: mature bone-marrow monocytes from adult mice and peripheral blood monocytes from humans. Mouse bone marrow was harvested as described in Supp Methods 12.2 and monocytes were sorted as Cd115+,Cd11c-,Cd43-,Ly6c(hi) cells. Human monocytes were manually selected from a human peripheral blood mononuclear cell (PBMC) dataset provided 10X genomics (https://support.10xgenomics.com/single-cell-gene-expression/datasets; Cell Ranger 2.1.0, "4k PBMCs from a Healthy Donor").

We performed PCA and observed that the second-ranking principal component (PC) for each dataset distinguished DC marker and neutrophil marker expression. We therefore divided the monocytes from each dataset into two groups: Neu-like monocytes, defined as those with a PC score greater than 0, and DC-like monocytes, defined as those with a PC score less than 0 (Supp Fig10f,g; note that the sign of the PC is arbitrary). Differential expression between these groups in each respective monocyte dataset was performed by applying a t-test to each highly variable gene and correcting p-values using the Benjamini-Hochberg method. Significant genes are reported in Table 8 and a subset is shown in Fig 4i.

11.6 Isolation, and culture, and visualization of MDP and GMP cells (Supp Fig 10c)

We investigated whether the distinct monocyte differentiation pathways in our vitro dataset corresponded to the monocyte-dendritic progenitor (MDP) and granulocyte-monocyte progenitor (GMP) progenitor cell types. MDPs and GMPs were obtained from adult mouse bone marrow as described in Supp Methods 12.2. A subset of each population was immediately profiled with inDrops and the remainder was plated in a broad multilineage cocktail for 2 days, then transferred to M-CSF (20ng/mL in StemSpan media) for 4 days until profiling with inDrops. To visualize MDPs and GMPs (before and after differentiation) on the SPRING plot of the main in vitro dataset, we projected them into the principal component space defined by the in vitro dataset, and then assigned each cell an x- and y-coordinate based on the average coordinate valued of its 20 nearest neighbors in the in vitro dataset.

12 Analysis of monocyte ontogeny with sleeping beauty transposon system

12.1 Labeling of Sleeping Beauty Transposon Mice

The Sleeping Beauty Transposon mouse model was used as described previously (Sun et al. 2014, Rodriguez-Fraticelli et al. 2018). Briefly, we crossed mice carrying homozygous knock-in insertions of the DsRed2-Sleeping Beauty Transposon reporter in the *colla1* locus (Tn/Tn) and mice carrying homozygous knock-in insertions of the Hyperactive Sleeping Beauty Transposase in the *col1a1* locus (SB/SB) and knock-in insertion of the M2-rtTA tetracyclin-responsive transactivator in the *Rosa26* locus (M2/M2) to generate barcodable *col1a1*$^{SB/Tn}$*Rosa26*$^{M2rtTA/+}$ mice. Two 3 month-old mice were labeled by 2mg/ml Doxycycline-hyclate (Sigma-Aldrich) with 5mg/ml sucrose in drinking water for 1 week. Thereafter, Dox was removed and successful labelling (~15% DsRed) was verified by retroorbital sinus peripheral blood collection (70 µl) after 1 week. Mice were euthanized 12 weeks after labeling. All animal procedures were approved by the Boston Children’s Hospital Institutional Animal Care and Use Committee.

12.2 Bone Marrow and Spleen Cell Isolation

After euthanasia, whole BM (excluding the cranium) and spleen were dissected and processed separately. Cellular fractions were prepared in 2% fetal bovine serum in phosphate buffered saline, filtered with 70 µm filters, and then erythrocytes were removed with red blood cell lysis buffer. Lineage depletion was performed using Magnetic Assisted Cell Separation (Miltenyi Biotec) with anti-cKit magnetic beads. The cKit+ fraction was stained with CD117 APC (cKit, ACK2, Biolegend, 1:100), Ly6a PE/Cy7 (Sca-1, D7, eBiosciences, 1:100), CD34 FITC (RAM34, eBiosciences, 1:50), CD135 PE/Cy5 (Flt3, A2F10, eBiosciences, 1:50), CD115 BV605 (CSF-1R, AFS98, Biolegend, 1:100), CD48 APC/Cy7 (HM48-1, eBiosciences, 1:100), CD3e eFluor450 (145-2C11, eBiosciences, 1:100), CD19 eFluor450 (MB19-1, eBiosciences, 1:100), Ter119 eFluor450 (TER119, eBiosciences, 1:100), Gr1 eFluor450 (RB6-685, eBiosciences, 1:100) and CD11b eFluor450 (Mac1, M1/70, eBiosciences, 1:100). The fractions sorted were MPP (Lin^{-}cKit^{+}Sca1^{+}CD48^{+}), EryP (Lin^{-}cKit^{+}Sca1^{-}CD34^{-}FcgRII/III^{-}Flt3^{-}), GMP (Lin^{-}cKit^{+}Sca1^{-}CD34^{+}CD115^{lo}FcgRII/III^{+}Flt3^{-}) and MDP (Lin^{-}cKit^{+}Sca1^{-}CD34^{+}CD115^{+}FcgRII/III^{lo}Flt3^{+}). The cKit- fraction was stained with CD115 BV605 (CSF-1R, AFS98, Biolegend, 1:50) , Biolegend, 1:100), Ly6C APC (HK1.4, Biolegend, 1:200), Ly6G-AF700 (1A8, eBiosciences, 1:100), Ter119 eFluor450 (TER119, eBiosciences, 1:100), CD19 APC/Cy7 (1D3, eBiosciences, 1:100), Cd11c FITC (HC3, Biolegend, 1:100), CD74 BV711 (MHC-II, In-1, Biolegend, 1:100) and CD43-PECy7 (S11, Biolegend, 1:100). Classic Monocytes were isolated as CD115^{+}CD19^{-}Ter119^{-}Ly6C^{hi}Ly6G^{-}CD43^{-} cells from the bone marrow, Neutrophils were isolated as CD115^{-}CD19^{-}Ter119^{-}Ly6C^{lo}Ly6G^{+}CD43^{-}CD74^{-} cells from the bone marrow, and DCs were isolated as CD115^{lo}CD19^{-}Ter119^{-}Ly6C^{-}Ly6G^{-}CD11c^{hi}CD74^{+} cells from the spleen. For Tn tag content extraction and analysis, only DsRed2^{+} cells were sorted from each fraction. We FACS-sorted all the available cells from the whole BM or spleen extract using purity modes (~95% purity) at ~75–80% sorting efficiency.

12.3 Transposon integration retrieval and analysis

Cells of interest were sorted into 1.7 ml tubes and concentrated into 5–10 µl of buffer by low speed centrifugation (700 g for 5 minutes). Transposon insertion site retrieval and analysis was performed with an adapted version of TARIS (Rodriguez-Fraticelli et al. 2018). Library indexing was used with Illumina library construction kit primers and sequencing was carried out on NextSeq (Illumina) at the Biopolymers Facility (Harvard Medical School). Tag identification and alignment was performed with a custom script as previously described (Sun et al. 2014). Briefly, we grep the Tn-containing reads from each fastq file, trim the adaptor and Tn sequences, and align the integration sites to the reference mouse genome (Ensembl mm9) using bowtie 1.2. Samples with fewer than 10,000 mapping reads were discarded. Then, reads are normalized between samples (per million reads) and compared with at least one additional independently labelled mouse with libraries prepared in parallel and sequenced in the same NextSeq lane to account for contaminations. Tags present in the control mouse samples or only in one of the split samples were filtered out (contaminating reads). Next, read frequencies were log normalized and plotted using a heatmap (rows: unique barcodes, columns: sorted cell populations). Custom primers used were: Tn1-C primer (5′-CTT GTG TCA TGC ACA AAG TAG ATG TCC-3’), MAF-Tn1-1F primer (5′-ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT NNN NCG AGT TTT AAT GAC TCC AAC T-3’), and MAR-LCII (5’-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATCTAGTGGCACAGCAGTTAGG-3’). All primers were ordered from IDT DNA technologies, at 100 nmole scale and HPLC-purified.

13 Application of fate prediction algorithms

13.1 Classification of neutrophil/monocyte trajectory cells

Putative neutrophil/monocyte trajectory cells are defined as cells spanning the continuum of states from MPPs to mature neutrophils/monocytes. There are multiple computational methods available to parameterize trajectories from single cell data alone. Access to clonal data offers a principled way of determining which cells belong on the trajectory, by including as many cells with Neu or Mo fate outcomes while minimizing inclusion of other fates. Here we implemented a custom method to define cells on the trajectory. The motivation for using a custom method, rather than published piplines, was in the flexibility to tune results to best match the clonal data from this study. We note that the approach for incorporating clonal information in order to best extract trajectories could be automated in future work.

We implemented a custom method to assign cells to the MPP-to-Mo/Neu trajectory by a stepwise process of graph smoothing and thresholding, with manual tuning of parameters to include as many cells with Neu or Mo fate outcomes while minimizing inclusion of other fates. Briefly, indicator variables representing each mature lineage as well as MPPs (see the previous methods section “Graph connectivity score” for a definition of indicator variables) were smoothed (N=250 smoothing iterations) and summed together to form an aggregate score per cell. In the summation, smoothed scores for monocytes, neutrophils and MPPs were given a positive coefficient, and the other lineages got a negative coefficient. Formally, let S_{ij} denote the smoothed score for the j-th lineage in the i-th cell, and let a_j be the coefficient for the j-th lineage. Then the aggregate score Z was computed for each cell as $Z_i = \sum_j a_j S_{ij}$. The aggregate score (Z) was then thresholded at the 40th percentile to generate an indicator I_i of the Neu/Mo trajectory. The resulting indicator can be noisy, so another round of smoothing was performed on I_i (N=50 iterations) and the smoothened indicator was re-thresholded, again at the 40th percentile, to generate a final annotation of neutrophil/monocyte trajectory cells. By manual inspection of clonal fates, the following values for a_i and thresholds were found to provide good specificity of the trajectory to Neu/Mo-committed cells: Neu=0.1, Mo=0.1, MPP=1, Meg = -2, All other lineages = -1;

13.2 Population Balance Analysis

We performed PBA as described in (*4*), using the python pipeline available at (https://github.com/AllonKleinLab/PBA). PBA was performed on a random 20,000-cell subset of the full dataset to limit the computational run-time. To extend the predicted fate probabilities to all cells, a 50-nearest-neighbor graph was constructed on the full dataset, and each cell inherited the mean value assigned to the nearest neighbors that were part of the 20,000-cell subset.

13.3 Waddington-OT

We performed Wadding-OT analysis (*35*) on the neutrophil/monocyte trajectory cells (Fig 5e) and on the whole dataset (Supp. Fig 7c) using python code available online (https://github.com/broadinstitute/wot). We did the analysis on a subset of cells (20,000 for neu/mo, 40,000 for all lineages) and then extended fate probabilities to the full dataset by averaging over 50 nearest neighbors (see above). The input to Wadding-OT is a set of time-point labeled transcriptomes and a proliferation score for each cell, and the output is a transition map between each pair of consecutive time points. We calculated proliferation scores in two steps. First, we used our state-fate barcoding data to calculate the clonal expansion downstream of each barcoded cell. Since these labels were very noisy, we then applied graph smoothing (N=15 rounds). Using the resulting proliferation scores, we ran Wadding-OT with parameters $\lambda_1 = 50$ and ϵ ranging from 0.0001 to 50, where λ_1 and ϵ are regularization parameters that control the fidelity of proliferation score constraints and the entropy of optimal transport map respectively. The output was a pair of transition maps $T_{2,4}$ and $T_{4,6}$ from day 2 to day 4 and day 4 to day 6 respectively, where $\left(T_{2,4}\right)_{i,j}$ denotes the total density transported from cell i to cell j.

Terminal fate probabilities for each day 2 cell were calculated by (1) Row normalizing the $T_{2,4}$ and $T_{4,6}$ to generate transition probabilities; (2) Composing the normalized transition maps; (3) Collapsing the target cells of the transition map by lineage; (4) Linearly rescaling each lineage probability to ensure a uniform average distribution over lineages among MPPs, to be consistent with PBA. Formally, these steps involved the following operations:

Step 1: Row normalizing the $T_{2,4}$ and $T_{4,6}$

$$(\bar{T}_{2,4})_{i,j} = \frac{\left(T_{2,4}\right)_{i,j}}{\sum_j \left(T_{2,4}\right)_{i,j}} \quad \&. \quad (\bar{T}_{4,6})_{i,j} = \frac{\left(T_{4,6}\right)_{i,j}}{\sum_j \left(T_{4,6}\right)_{i,j}}$$

Step 2: Composing the normalized transition maps

$$(\bar{T}_{2,6})_{i,j} = \sum_k (\bar{T}_{2,4})_{i,k} (\bar{T}_{4,6})_{k,j}$$

Step 3: Collapsing the target cells of the transition map by lineage

Let X_{ij} be an indicator variable where $X_{ij} = 1$ if cell i is in lineage j, otherwise $X_{ij} = 0$. We calculate P_{ij}, the probability that cell i (from day 2) gives rise to lineage j, as follows:

$$P_{i,j} = \sum_k (\bar{T}_{2,6})_{i,k} X_{kj}$$

Step 4: Linearly rescaling each lineage probability to ensure a uniform distribution over lineages among MPPs

To set all three of the computational methods on an equal footing, we implemented a final rescaling step on the resulting probabilities that ensures MPPs in all cases have the same average predicted fate probabilities for Mo and Neu fates. This rescaling step allows focusing on changes in predicted fate commitment point, rather than on the net biases predicted by each method. We note that the biases in each method depend on the assumed fluxes of cells towards each terminal fate, which are adjustable parameters (e.g. in Waddington-OT, this free parameter is the net expansion rate per transcriptional state).

Let X_i be an indicator variable where $X_i = 1$ if cell i is an MPP, otherwise $X_i = 0$, and let $Q_j = (\sum_k X_k P_{kj})/(\sum_k X_k)$ be the average predicted probability for an MPP to give rise to lineage j. Starting with lineage probabilities P_{ij}, we calculate rescaled probabilities $\tilde{P}_{ij}$ as follows:

$$\tilde{P}_{ij} = \frac{P_{ij}}{Q_j} \Big/ \sum_k \frac{P_{ik}}{Q_k}$$

13.4 FateID

We performed FateID analysis (*7*) using the R package (https://github.com/dgrun/FateID). We did the analysis on a subset of 20,000 cells and then extended to the full dataset. The top 1000 most highly variable genes were used. As with PBA and Wadding-OT, lineage probabilities were linearly rescaled to ensure a uniform average distribution of fates for MPPs (see step 4 above).

13.5 Fate prediction with held-out data

The poor correlation between clonal fate outcomes the computational fate prediction methods arises from a combination of noise in the clonal data and genuine differences between predicted and experimental fate boundaries. To understand the relative contribution of these factors, we performed a fate prediction analysis using held-out clonal data and then applied it to the remaining clonal data. Since the noise between these two subsets of the clonal data is independent, the resulting correlation provides an upper bound for the other methods tested. To perform fate prediction from held-out data, we sampled 50% of clones to act as the held-out clones, and then assigned predicted fate probabilities to the non-held-out cells by taking the average fate outcome for held-out cells that were among their 400 nearest neighbors in the whole day 2 dataset. Here fate outcome is defined as the relative proportion of neutrophil vs. monocyte sisters on days 4 and 6.

14 Pseudotime analysis

14.1 Classification of neutrophil trajectory cells

To classify cells as belonging to the neutrophil trajectory, we followed a similar procedure to that described above in section “Classification of neutrophil/monocyte trajectory cells”, but with the following weights. a_i: Neu=0.1, MPP=0.1, Meg = -2, All other lineages = -1, and the threshold for the aggregrate score set to the 60th percentile rather than the 40th, since the neutrophil trajectory has fewer cells than the branching neutrophil+monocyte trajectory. As above, the parameter choices were guided by manual inspection that the trajectory captured neutrophil clones while minimizing other fates.

14.2 Pseudotime ordering assignment

We assigned a pseudotime value to each cell in the neutrophil trajectory using graph smoothing. An initial indicator vector with 1 for MPP cells and 0 for all other cells was smoothed (N=300 rounds), generating a gradient of values that was highest near the MPPs and decreased along the trajectory. These values were quantile normalized to generate a final set of pseudotimes. Pseudotime was extremely robust to the number (N) of smoothing rounds (Supp Fig 11c), remaining correlated ($R > 0.975$) across a broad range of N varying from 50 to 1000.

14.3 Calculating the timing of differentiation for a typical cell

Let t represent real (calendar) time, and F represent pseudotime. To calculate the function $F(t)$ from clonal barcoding data (Fig 5f), we began by calculating the average change in pseudotime over 2 days for cells starting at different points in the trajectory. The resulting function $D(f) = 2\,\frac{dF}{dt}|_{F(t)=f}$, is proportional to the derivative of F with respect to t, written as a function of F. To transform D(f) into F(t), we can use the following general formula for taking the derivative of an inverse function:

$$[F^{-1}]'(f) = \frac{1}{F'\left(F^{-1}(f)\right)} = \frac{2}{D(f)}$$

It follows that

$$F^{-1} = \int \frac{2}{D(f)} df$$

And therefore

$$F(t) = \left[\int \frac{2}{D(f)} df\right]^{-1}$$

In practice, we calculated $D(f)$ by first collecting all pairs of clonally related cells that appear two days apart and recording the change in pseudotime (ΔT) between them, as well as the pseudotime f of the starting cell. Note that, $D(f) = \frac{1}{2}\Delta T(f)$. Since the ΔT values were noisy, we used curve fitting to estimate $D(f)$. Curve fitting was performed in python using the function UnivariateSpline from package scipy.interpolate, with smoothing factor = 10^9 and all other parameters default. We then numerically integrated $D(f)$ to arrive at an estimate for $F(t)$.

15 Analysis of lineage hierarchy

15.1 Observed/Expected clonal coupling score (Fig 5k,p)

We computed an observed/expected ratio of shared barcodes for each pair of lineages in the in vitro state-fate dataset. A barcode is considered shared if it appears in at least one cell from both lineages. From the observed shared barcode matrix O_{ij}, we derived an expected shared barcode matrix E_{ij} under the assumption of no lineage couplings, as follows:

$$E_{ij} = \left(\sum_k O_{kj}\right)\left(\sum_k O_{ik}\right)/\left(\sum_{k,l} O_{kl}\right)$$

To avoid artifacts from particularly large or atypical clones, we re-computed these matrices 500 times, each time using a random 25% sample of clones. The lineage coupling scores shown in Fig 5k,p represent the median O_{ij}/E_{ij} from these 500 randomized trials.

15.2 Graph connectivity score (Fig 5l,q)

Graph connectivity between each pair of lineages in the in vitro state-fate dataset was computed by smoothing an indicator vector centered on each lineage, as follows: For each lineage, i, we initialized an indicator vector X_i defined by:

$$(X_i)_k = \begin{cases} 1 \text{ if cell } k \text{ is a member of lineage } i. \\ 0 \text{ if cell } k \text{ is not a member of lineage } i \end{cases}$$

We smoothed the indicator vector using graph smoothing with N iterations applied to a k-nearest neighbor graph with k=100, generating a smoothened vector S_i (N=100 iterations were used for Fig 5l,q, and a range of N values are shown in Supp Fig 12a,b). The graph connectivity between lineages i and j was then calculated as the mean value of S_i among the cells of lineage j. For lineage membership we used the annotation described above (see section “Annotation of cell types”).

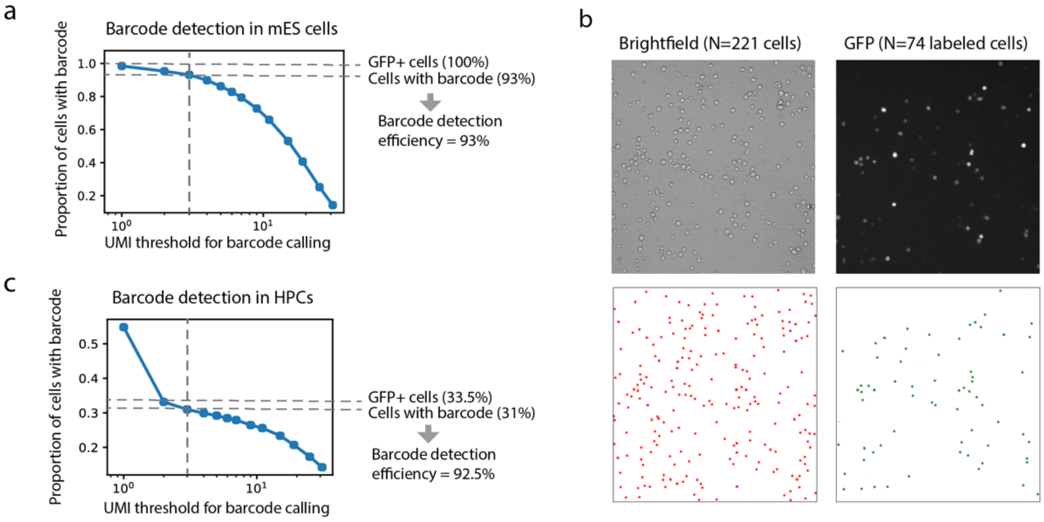
15.3 Alternative measures of cell state connectivity (Supp Fig 12c-h)

To test the robustness of the state-distance fate-distance correlation presented in Fig 5l,q, we tested two alternative metrics of state-distance. **End-state correlation** (Supp Fig 12c) was calculated as the Pearson correlation between the average expression vectors for each lineage, after Z-scoring of gene expression. **Number of shared edges** (Supp Fig 12f) was calculated as the number of edges that cross from one lineage to another in a knn-graph, and is proportional to the ‘graph connectivity score’ described above with N=1 round of smoothing.

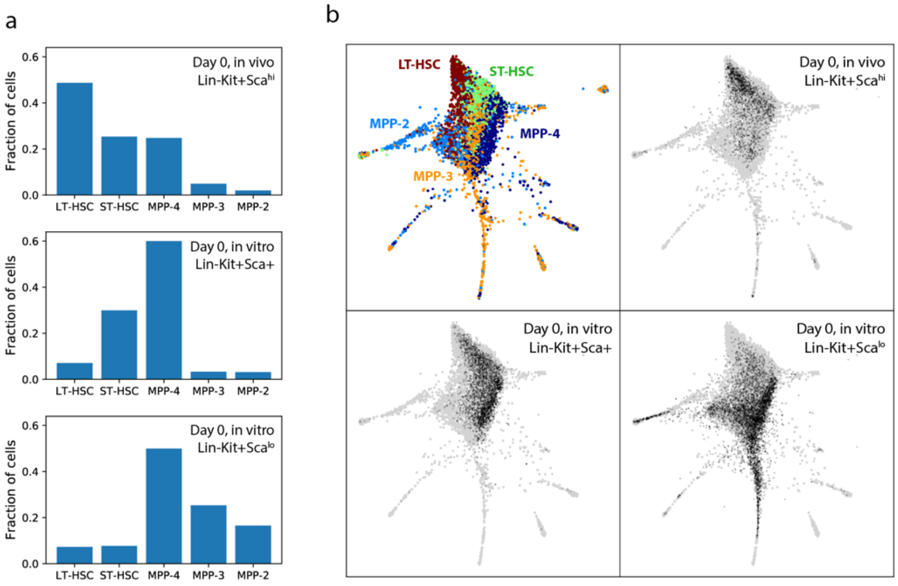
15.4 Lineage hierarchy construction and comparison (Fig 5n,o,s,t; Supp Fig 12b,e,h)

Lineage hierarchies were constructed from matrices of lineage distance (both state distance and fate distance) by hierarchical clustering with average linkage. The lineage trees shown are dendrograms of the clustering operation. The trees produced from state- and fate- distance were formally compared using the Robinson-Foulds metric, which can be defined as follows. For any edge in a tree, the removal of the edge will sever the tree into two connected components, forming a bipartition of the leaf nodes. The Robinson-Foulds metric counts the number of such bipartitions that are unique to one or the other of the trees being compared. We calculated Robinson-Foulds distance using the “symmetric_difference” function from the Dendropy package.

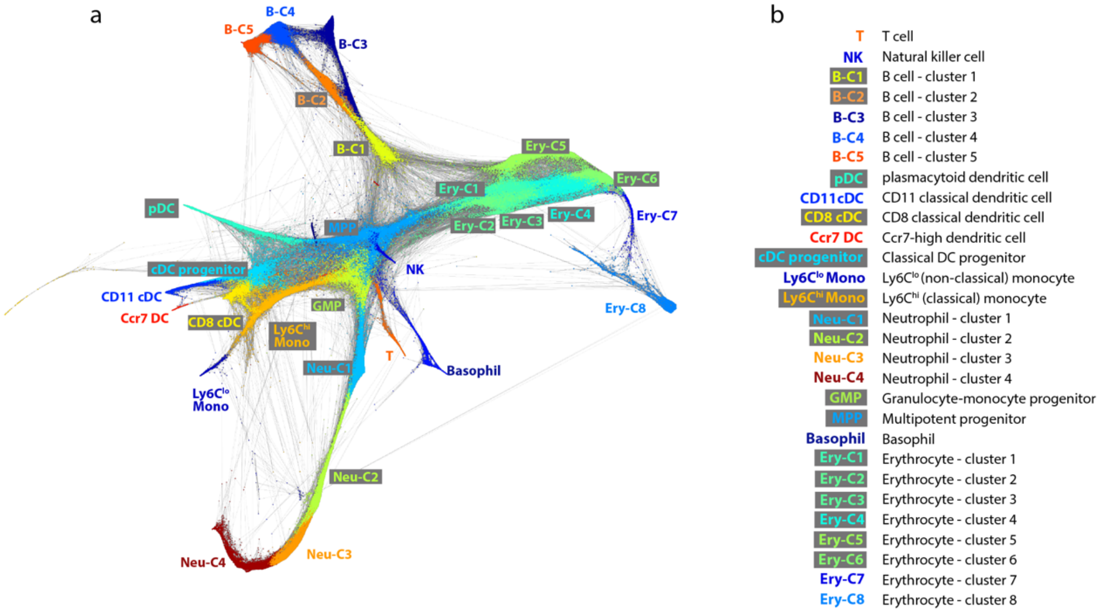
Supplementary Figure 1



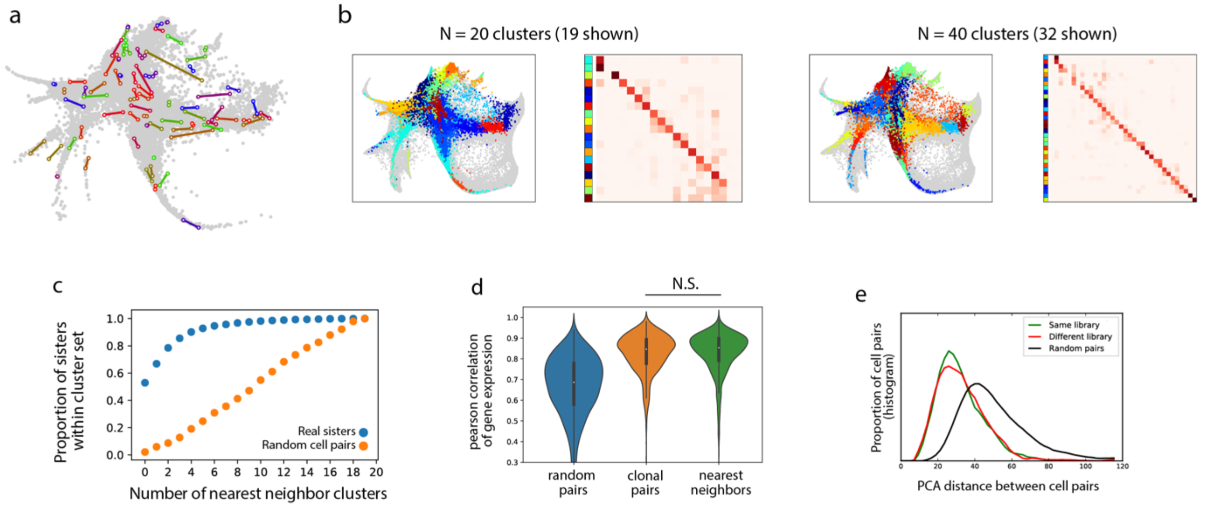
Supplementary Figure 1: Lineage and RNA Recovery (LARRY) detection efficiency and effect on cells. (a) Proportion of cells with $> n$ UMIs of a LARRY barcode as a function of n in GFP-sorted embryonic stem cells. Dashed line indicates filtering threshold to accept a LARRY barcode (minimum 3 UMIs). (b) Microscopy-based measurement of viral transduction efficiency in HPCs. Cell counts in brightfield and FITC images (top) are computationally scored (bottom) to measure the fraction of GFP+ cells (74/225 cells). (c) Proportion of cells with $> n$ UMIs of a LARRY barcode as a function of n in HPCs from the same pool shown in panel (b). Dashed line threshold as in (a).



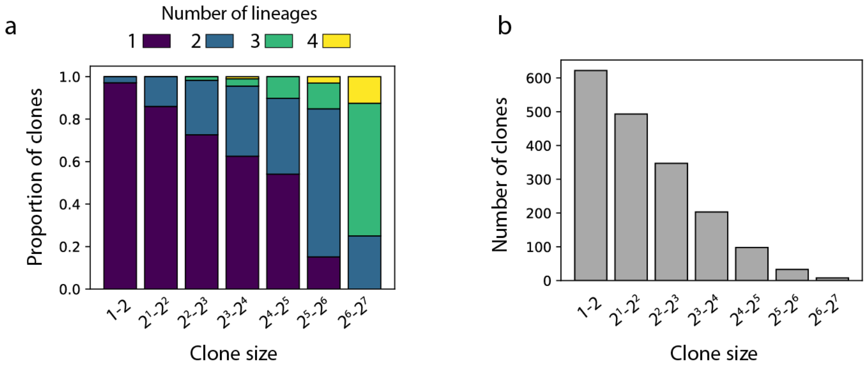
Supplementary Figure 2: Cell types of day 0 starting populations. (a,b) Day 0 populations mapped to scSeq data from a previous paper (Rodriguez-Fraticelli, 2018) with distinct MPP subsets identified through FACS. (a) Bar-chart indicating the proportion of cells that mapped to each of 5 FACS gates from three day 0 starting populations: Lin-Kit+Sca^hi cells used for the in vivo experiment, Lin-Kit+Sca+ cells as well as Lin-Kit+Sca- cells used for the in vitro experiment. (b) Cells from each of the day 0 starting populations mapped onto a reference SPRING plot. The SPRING includes cells from 5 MPP FACS gates (see top-left panel).



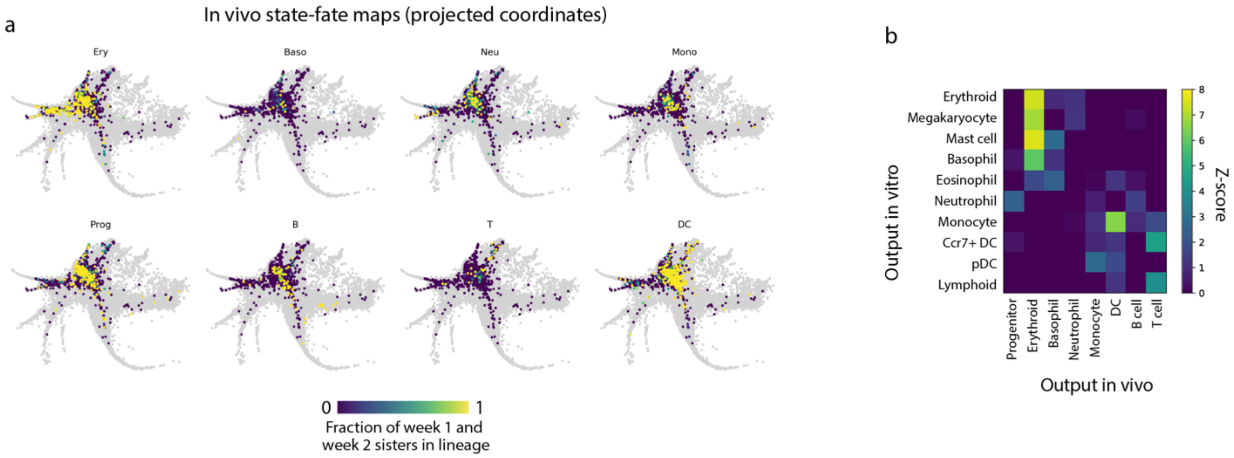
Supplementary Figure 3: Cell states after transplantation. (a) SPRING plot of cells from samples collected one and two weeks after transplantation. A clustering is indicated by color, with the cell type of each clusters labeled according to manual inspection of marker genes. Cluster names are abbreviated in (a), and written fully in (b).



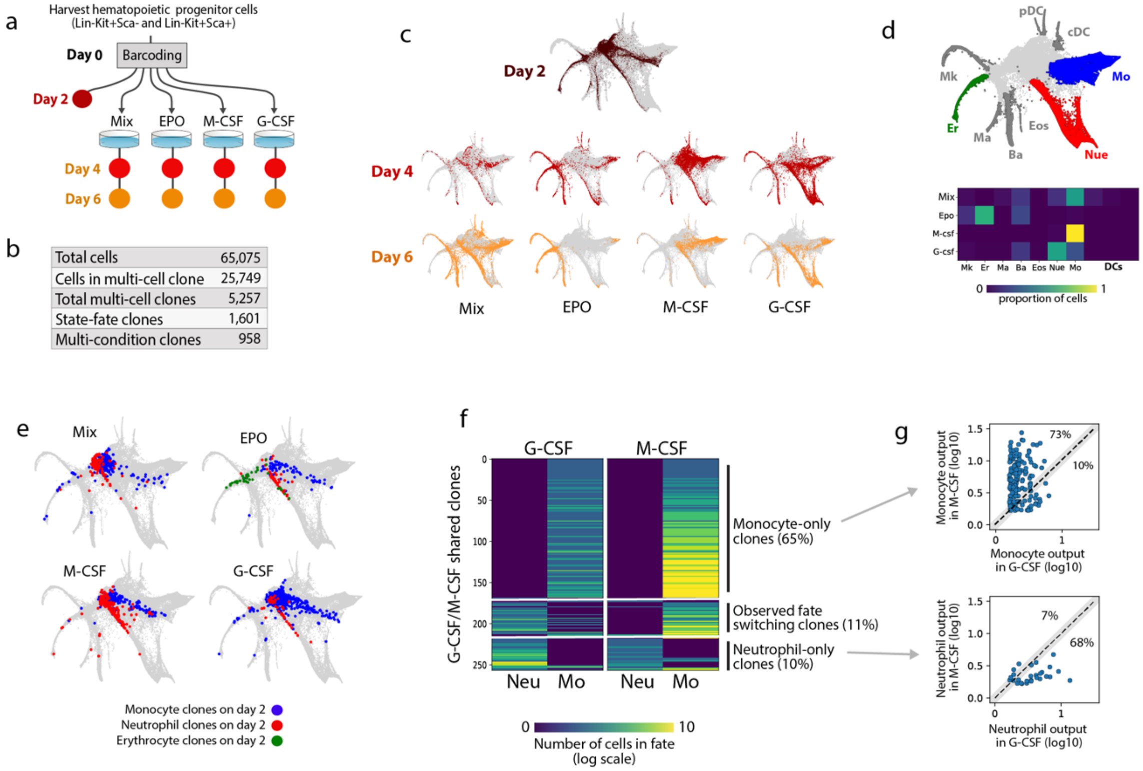
Supplementary Figure 4: Analysis of sister cell differences on day 2. (a) SPRING plot visualization of a sample of sister cell pairs 2 days post-barcoding. Each colored line spans between two sister cells. The sister cell pairs were sampled so that they would be non-overlapping, but also have 2D distances representative of the full distribution. Grey points show the full data set across all time points. (b) Co-localization of sister cells in clusters shown for two different clustering granularities. In each case, the cluster labels for each cell are plotted on the left, and a heatmap showing the joint probability for a pair of sisters to appear in two different clusters is shown on the right. Out of N total clusters, only a subset is shown in the heatmap, since we excluded clusters with <0.1% of the cells. (c) Probability that a sister cell will lie within the same cluster or among the N nearest clusters for variable N. The clustering from Fig 2b was used. (d) Distribution of gene expression correlation between random pairs of cells (left), sister cells (middle) and nearest neighbors in PCA space (right), showing no significant different between the latter two conditions. (e) Distribution of high dimensional distances between: random cell pairs (black); sister cell pairs separated and subject to separate inDrop cell encapsulation runs (red); and sister cell pairs from the same library (green). The difference between the green and red curves indicates the degree to which sister cell similarity is driven by single-cell encapsulation artifacts such as gel doublets.



Supplementary Figure 5: Relationship between clone size and number of lineages in vitro. (a) Stacked bar chart showing the prevalence of uni-potent, bi-potent, tri-potent, and 4-potent clones, stratified by clone size. Clone size is based on cells from days 4 and 6. Lineages were grouped for the analysis as follows: [Er,Mk,Ba,Ma,Eos], [Neu], [Mo], [Ly,DC]. (b) Overall distribution of clone sizes, where size is based on the number of cells on days 4 and 6.

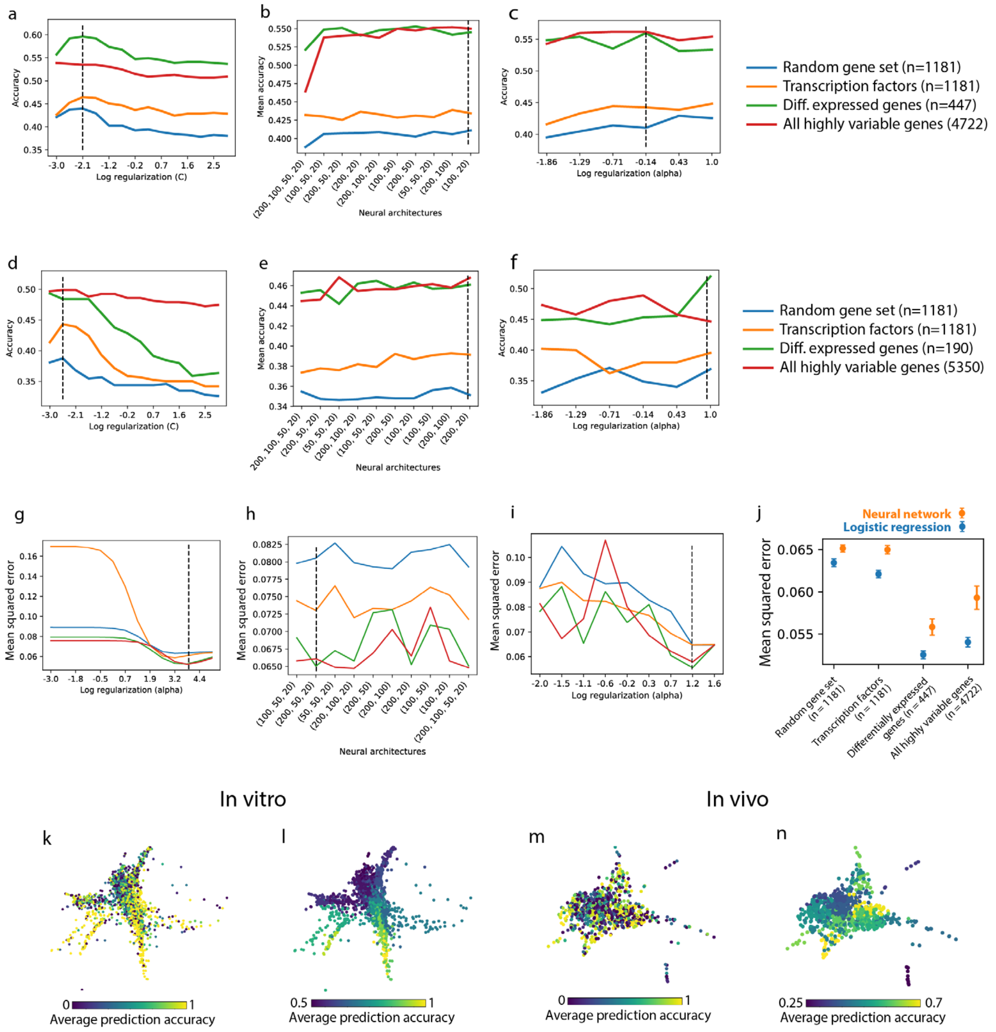


Supplementary Figure 6: Comparison of clonal fate bias in vivo and in vitro. (a) Cells from day 2 o the in vivo dataset were projected onto the day 2 landscape of the in vitro data. Gray dots are day 2 in vitro-dataset cells, colored dots are day 2 in vivo-dataset cells colored by their clonal lineage output after transplantation. (b) Heatmap comparing in vitro output to in vivo output. Each column represents the progenitor cells of a lineage in vivo, colored by the in vitro output of the cells they have been projected into. Thus, a high value in the heatmap at entry (i,j) indicates the that the progenitor of (j) in vivo give rise to (i) in vitro.

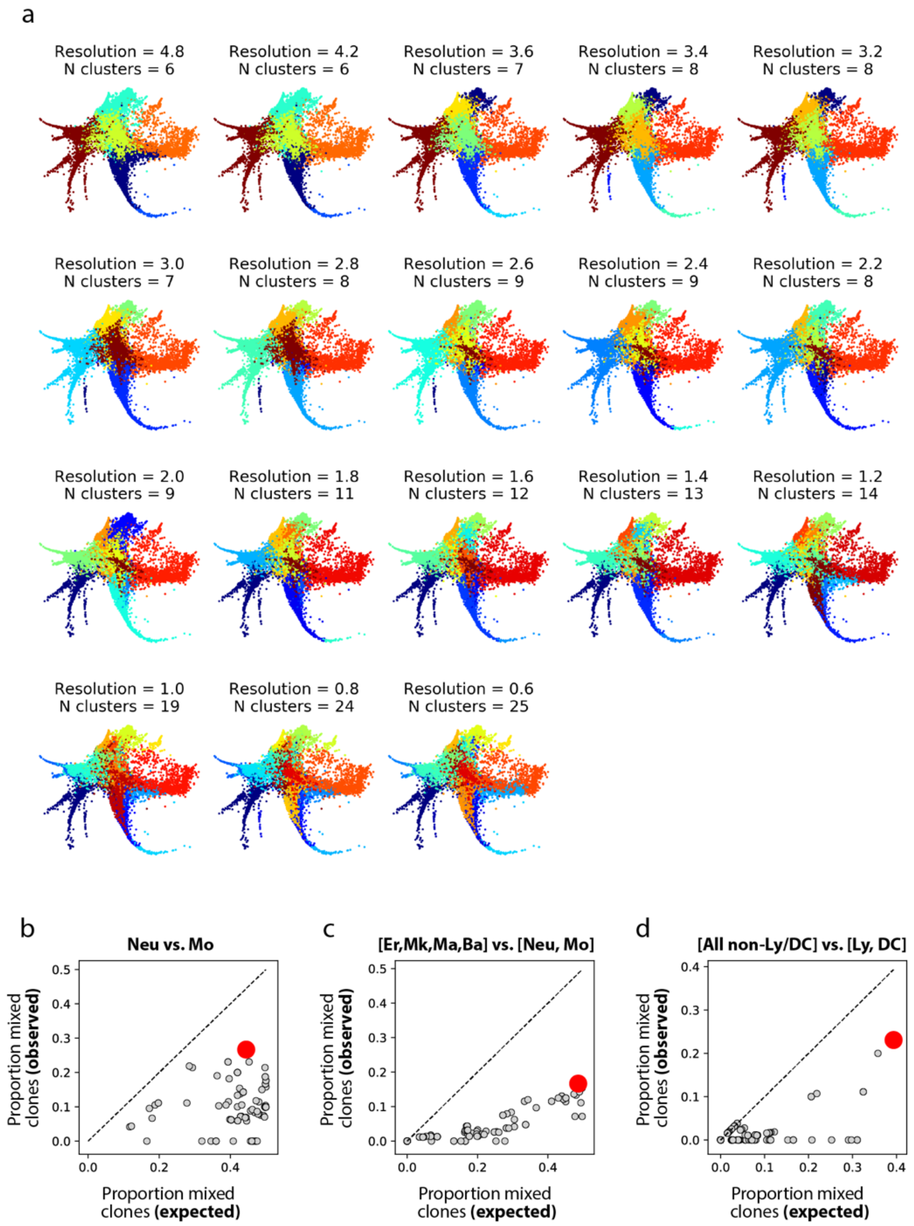


Supplementary Figure 7: Clonal cell fate under cytokine perturbations. (a) Experimental design for measuring the effect of cytokine perturbation on clonal cell fate. Cells are harvested, barcoded and plated. After two days of culture to allow cell division, cells are split into five groups, including one group for immediate profiling by scSeq and four groups for continued culture in different cytokine conditions. Each cytokine condition consists of a single cytokine, except "mix" which refers to a broad multilineage cocktail. Colored circles represent scSeq samples. (b) Numbers of cells and clones sampled. (c) SPRING plots showing cells from each sample (colored dots) superimposed on all cells from the experiment (gray dots). (d) Cell type annotations for mature cells (top), and quantification of the proportion of mature cells in each fate for each cytokine condition (bottom) showing an enrichment of erythrocytes in EPO, increased neutrophils in G-CSF and increased monocytes in M-CSF. (e) Day 2 cells colored by the observed fate of their sisters on days 4 and 6. Each panel shows a different cytokine condition. Day 2 cells are colored if they gave rise to only one observed fate in that cytokine condition, and the identity of the fate determines the color. Comparison of M-CSF and G-CSF shows arms of maturing cells that produce the same lineage across conditions, as well as a core of immature cells that that lie in a region with shifting fate boundaries that depend on the condition. (f) Heatmap showing clones that were detected on days 4/6 of in both G-CSF (left) and M-CSF (right). Each row represents a clone, and rows are shared between the two heatmaps. Color indicates number of cells. Three dominant behaviors are apparent: clones that produced monocytes in both conditions (65%), clones that produced neutrophils in both conditions (10%) and clones that produced monocytes in M-CSF and neutrophils in G-CSF (11%).

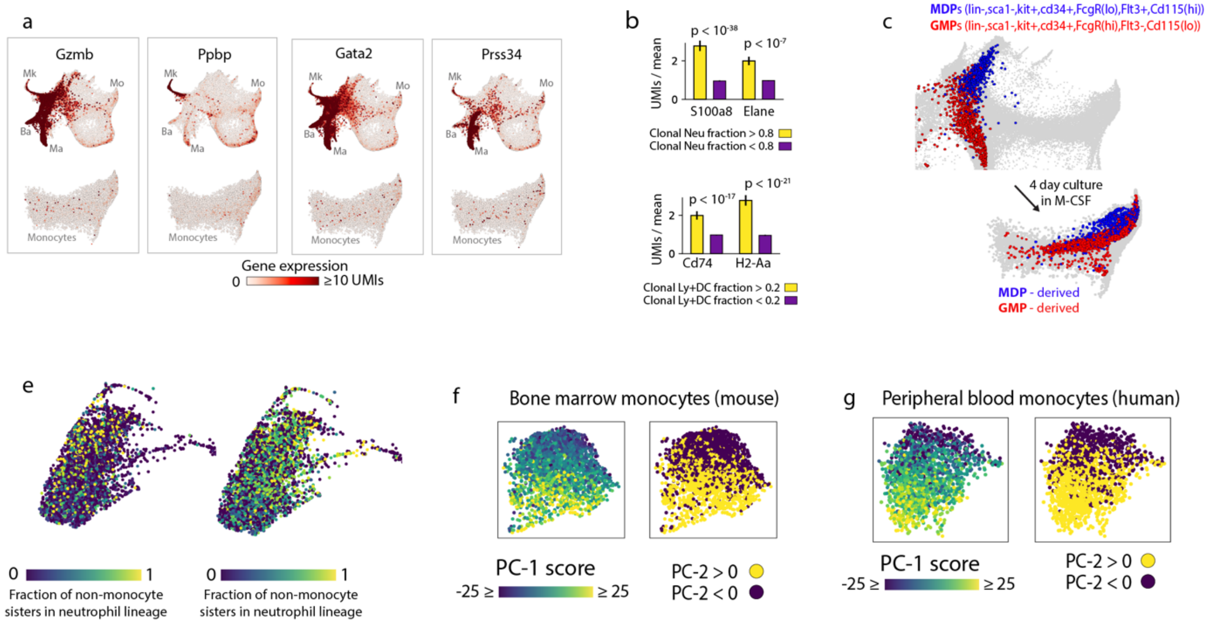
(g) Scatter plots showing clone size in M-CSF vs. G-CSF, showing either monocyte-only clones (top) or neutrophil-only clones (bottom). Monocyte-only clones expand more in M-CSF than G-CSF, whereas the reverse behavior occurs in neutrophil-only clones. Percentages indicate the proportion of clones in the indicated region. Clones lying on the line of equality (gray) don't contribute to either percentage. Points in the scatter plot have been jittered to avoid overlap.
(e-g) Taken together, these data show cytokines altering cell fate through regulating proliferation or survival in committed cells, and switching the observed fate of more immature cells. The switching of observed clonal fates could be explained an additional effect in biasing the fate choice of uncommitted cells, but we cannot rule out that these data result from changes in survival alone.



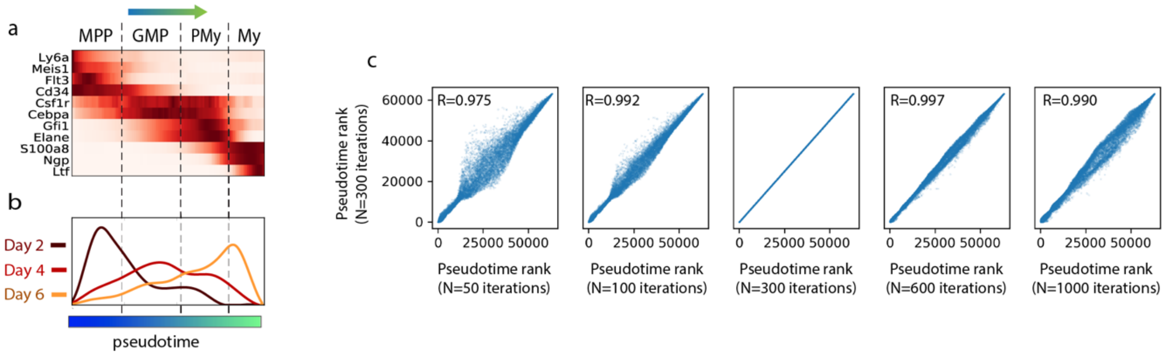
Supplementary Figure 8: Predictability of fate across stages of differentiation. (a-c) Scan over hyperparameters for logistic regression (a) and multilayer perceptron classifier (b,c) for the in vitro data. Dotted line indicates the parameters chosen for downstream analysis. (d-f) as (a-c) for in vivo data. (g-i) Scan over hyperparameters for ridge regression (g) and multilayer perceptron regressor (h,i). Dotted line as in (a-f). (j) Accuracy of regressors for predicting the distribution of fate outcomes for each clone. Errorbars represent standard deviation. (k) Day 2 cells colored by accuracy of fate prediction using logistic regression. Cells were randomly partitioned into train and test sets 500 times and the accuracy of prediction recorded for each cell. Each cell is colored by the proportion of correct fate predictions across all trials where the cell belonged to the test set. (m) Same as (k) applied to fates one-week post-transplantation. (l,n) Values from (k) and (m) respectively averaged across clusters reveals a pattern of increasing prediction accuracy with greater cell maturity.



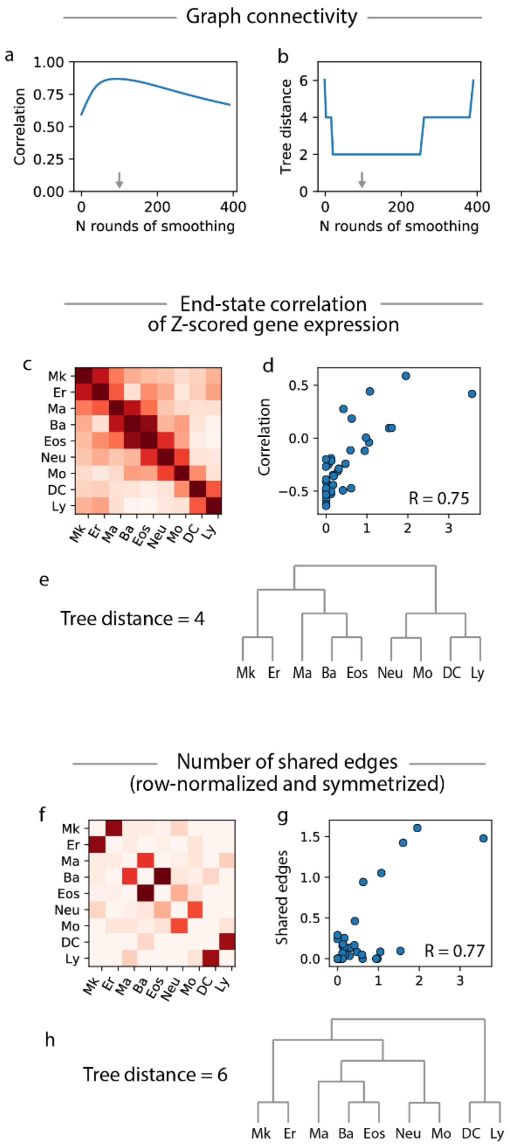
Supplementary Figure 9: Functional purity of scSeq-defined cell states. (a) Ensemble of clusterings of cells from day 2, using a range of values for the resolution parameter in Louvain-Jaccard clustering. (b-d) For each cluster from each clustering in (a), the proportion of 'mixed clones' with a day 2 cell in that cluster was assessed. A clone is defined as 'mixed' if it is detected in two separate wells after 6 days in culture, and the fate in each well is distinct (see Fig 3i). In each plot, the clusters (gray dots) are plotted according to the observed proportion of mixed clones (y-axis) compared to that expected for a pure bi-potent population (x-axis; see Fig 3i). The red dot indicates the cluster with the highest proportion of mixed clones. The red dot clusters are used for Fig 3j.



Supplementary Figure 10: Heterogeneity in the gene expression and ontogeny of monocytes. (a) Expression of basophil, mast cell and megakaryocyte marker genes in the whole dataset (top plot in each panel) or just monocytes (bottom plot in each panel) shows, by comparison to Fig 5a, that only neutrophil and DC markers are enriched among monocytes. Since basophil, mast cells and megakaryocytes combined are 70% as abundant as neutrophils, the neutrophil and DC marker gene expression in monocytes is unlikely to be caused by cell doublets, which would affect all lineages equally. (b) Quantifying the link between gene expression and lineage history by differential expression of neutrophil markers among monocytes with or without significant clonal coupling to neutrophils (top), and the same for DC markers (bottom). Systematic differential expression analysis (Fig 4e) revealed a large number of genes also matching this segregated pattern. Functional enrichments between these groups revealed that beyond the differential expression of key marker genes, DC-like and neutrophil-like monocyte express batteries of genes that may distinguish their function in immunity. Gene set enrichment revealed terms such as 'inflammatory response' ($p < 10^{-12}$) 'leukocyte migration' ($p < 10^{-5}$), and 'wound healing' ($p < 10^{-5}$) associated with neutrophil-like monocytes, compared to terms such as 'response to virus' ($p < 10^{-20}$), and 'response to interferon beta' and gamma ($p < 10^{-15}$). (c) The location of freshly sorted GMPs and MDPs projected onto the SPRING plot (top), and the projected positions of MDP-derived and GMP-derived cells after 4 days of culture in M-CSF (bottom). (e) Monocytes one week post-transplantation colored by the fraction of their sisters that are neutrophils (left) or DCs (right). (f,g) The value of the second principal component (PC) is plotted for bone marrow monocytes in mouse (f) and peripheral monocytes in human (g) (left panels). These PC scores correlate with a neutrophil vs. DC axis of gene expression, as revealed by differential expression of cell groups defined by positive vs. negative values of the PC score (right panels).



Supplementary Figure 11: Supporting data for pseudotime progression analysis. (a) Marker genes used to establish boundaries between stages [MPP=multipotent progenitor; GMP=granulocyte monocyte progenitor; PMy=promyelocyte, My=myelocyte]. (b) Distribution of pseudotimes for each time point. Cells within a time point are asynchronous, but accumulate toward the end of the trajectory over time. (c) Robustness test for parameters used to compute pseudotime. Pseudotime was defined by graph smoothing – an iterative process where the outcome depends on the number of iterations chosen. Scatter plots comparing the pseudotime orderings for different values of N iterations are shown. In each case, the result for a given value of N is compared to that for N=300, which is used for all other analyses.



Supplementary Figure 12: Robustness analysis for the comparison of state and fate distance in in vitro hematopoiesis. (a) Sensitivity analysis showing that correlation between fate distance and state distance remains stable across graph-smoothing iterations in the calculation of state distance. Correlation predictably drops at high iterations as all states become equally close. The gray arrow indicates the value used in Fig 5k-m. (b) Trees generated from state and fate distance also remain similar across parameter values. Tree distance (y-axis) is defined using the Robinson-Foulds metric. (c) Heatmap of correlation-based state distances, their relationship with fate distances (d), and the tree that they produce (e). Note that this metric directly compares gene expression similarity, which is expected to be a less reliable indicator of developmental relationship than the graph connectivity via transitional states. (f) State distance based on number of shared edges between lineages in a nearest neighbor graph, their relationship with fate distances (g), and the tree that they produce (h). Note that this distance corresponds to N=1 iterations in panel (a).

References and Notes

1. P. Jensen, S. M. Dymecki, Essentials of recombinase-based genetic fate mapping in mice. *Methods Mol. Biol.* **1092**, 437–454 (2014). doi:10.1007/978-1-60327-292-6_26 Medline

2. M. B. Woodworth, K. M. Girskis, C. A. Walsh, Building a lineage from single cells: Genetic techniques for cell lineage tracking. *Nat. Rev. Genet.* **18**, 230–244 (2017). doi:10.1038/nrg.2016.159 Medline

3. C. A. Herring, B. Chen, E. T. McKinley, K. S. Lau, Single-cell computational strategies for lineage reconstruction in tissue systems. *Cell. Mol. Gastroenterol. Hepatol.* **5**, 539–548 (2018). doi:10.1016/j.jcmgh.2018.01.023 Medline

4. C. Weinreb, S. Wolock, B. K. Tusi, M. Socolovsky, A. M. Klein, Fundamental limits on dynamic inference from single-cell snapshots. *Proc. Natl. Acad. Sci. U.S.A.* **115**, E2467–E2476 (2018). doi:10.1073/pnas.1714723115 Medline

5. G. Schiebinger, J. Shu, M. Tabaka, B. Cleary, V. Subramanian, A. Solomon, J. Gould, S. Liu, S. Lin, P. Berube, L. Lee, J. Chen, J. Brumbaugh, P. Rigollet, K. Hochedlinger, R. Jaenisch, A. Regev, E. S. Lander, Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell* **176**, 1517 (2019). doi:10.1016/j.cell.2019.02.026 Medline

6. G. La Manno, R. Soldatov, A. Zeisel, E. Braun, H. Hochgerner, V. Petukhov, K. Lidschreiber, M. E. Kastriti, P. Lönnerberg, A. Furlan, J. Fan, L. E. Borm, Z. Liu, D. van Bruggen, J. Guo, X. He, R. Barker, E. Sundström, G. Castelo-Branco, P. Cramer, I. Adameyko, S. Linnarsson, P. V. Kharchenko, RNA velocity of single cells. *Nature* **560**, 494–498 (2018). doi:10.1038/s41586-018-0414-6 Medline

7. J. S. Herman, D. Sagar, D. Grün, FateID infers cell fate bias in multipotent progenitors from single-cell RNA-seq data. *Nat. Methods* **15**, 379–386 (2018). doi:10.1038/nmeth.4662 Medline

8. K. Akashi, D. Traver, T. Miyamoto, I. L. Weissman, A clonogenic common myeloid progenitor that gives rise to all myeloid lineages. *Nature* **404**, 193–197 (2000). doi:10.1038/35004599 Medline

9. S. Nestorowa, F. K. Hamey, B. Pijuan Sala, E. Diamanti, M. Shepherd, E. Laurenti, N. K. Wilson, D. G. Kent, B. Göttgens, A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation. *Blood* **128**, e20–e31 (2016). doi:10.1182/blood-2016-05-716480 Medline

10. L. Perié, K. R. Duffy, L. Kok, R. J. de Boer, T. N. Schumacher, The branching point in erythro-myeloid differentiation. *Cell* **163**, 1655–1662 (2015). doi:10.1016/j.cell.2015.11.059 Medline

11. F. Notta, S. Zandi, N. Takayama, S. Dobson, O. I. Gan, G. Wilson, K. B. Kaufmann, J. McLeod, E. Laurenti, C. F. Dunant, J. D. McPherson, L. D. Stein, Y. Dror, J. E. Dick, Distinct routes of lineage development reshape the human blood hierarchy across ontogeny. *Science* **351**, aab2116 (2016). doi:10.1126/science.aab2116 Medline

12. L. Kester, A. van Oudenaarden, Single-cell transcriptomics meets lineage tracing. *Cell Stem Cell* **23**, 166–179 (2018). doi:10.1016/j.stem.2018.04.014 Medline

13. L. Tian, J. Schreuder, D. Zalcenstein, J. Tran, N. Kocovski, S. Su, P. Diakumis, M. Bahlo, T. Sargeant, P. D. Hodgkin, M. E. Ritchie, S. H. Naik, SIS-seq, a molecular 'time machine', connects single cell fate with gene programs. bioRxiv 403113 [Preprint]. 29 August 2018. https://doi.org/10.1101/403113.

14. D. T. Montoro, A. L. Haber, M. Biton, V. Vinarsky, B. Lin, S. E. Birket, F. Yuan, S. Chen, H. M. Leung, J. Villoria, N. Rogel, G. Burgin, A. M. Tsankov, A. Waghray, M. Slyper, J. Waldman, L. Nguyen, D. Dionne, O. Rozenblatt-Rosen, P. R. Tata, H. Mou, M. Shivaraju, H. Bihler, M. Mense, G. J. Tearney, S. M. Rowe, J. F. Engelhardt, A. Regev, J. Rajagopal, A revised airway epithelial hierarchy includes CFTR-expressing ionocytes. *Nature* **560**, 319–324 (2018). doi:10.1038/s41586-018-0393-7 Medline

15. R. Lu, N. F. Neff, S. R. Quake, I. L. Weissman, Tracking single hematopoietic stem cells in vivo using high-throughput sequencing in conjunction with viral genetic barcoding. *Nat. Biotechnol.* **29**, 928–933 (2011). doi:10.1038/nbt.1977 Medline

16. D. S. Lin, A. Kan, J. Gao, E. J. Crampin, P. D. Hodgkin, S. H. Naik, DiSNE movie visualization and assessment of clonal kinetics reveal multiple trajectories of dendritic cell development. *Cell Rep.* **22**, 2557–2566 (2018). doi:10.1016/j.celrep.2018.02.046 Medline

17. B. A. Biddy, W. Kong, K. Kamimoto, C. Guo, S. E. Waye, T. Sun, S. A. Morris, Single-cell mapping of lineage and identity in direct reprogramming. *Nature* **564**, 219–224 (2018). doi:10.1038/s41586-018-0744-4 Medline

18. C. Weinreb, S. Wolock, A. M. Klein, SPRING: A kinetic interface for visualizing high dimensional single-cell expression data. *Bioinformatics* **34**, 1246–1248 (2018). doi:10.1093/bioinformatics/btx792 Medline

19. L. Velten, S. F. Haas, S. Raffel, S. Blaszkiewicz, S. Islam, B. P. Hennig, C. Hirche, C. Lutz, E. C. Buss, D. Nowak, T. Boch, W.-K. Hofmann, A. D. Ho, W. Huber, A. Trumpp, M. A. G. Essers, L. M. Steinmetz, Human haematopoietic stem cell lineage commitment is a continuous process. *Nat. Cell Biol.* **19**, 271–281 (2017). doi:10.1038/ncb3493 Medline

20. J. B. Kinney, G. S. Atwal, Equitability, mutual information, and the maximal information coefficient. *Proc. Natl. Acad. Sci. U.S.A.* **111**, 3354–3359 (2014). doi:10.1073/pnas.1309933111 Medline

21. A. Yáñez, S. G. Coetzee, A. Olsson, D. E. Muench, B. P. Berman, D. J. Hazelett, N. Salomonis, H. L. Grimes, H. S. Goodridge, Granulocyte-monocyte progenitors and monocyte-dendritic cell progenitors independently produce functionally distinct monocytes. *Immunity* **47**, 890–902.e4 (2017). doi:10.1016/j.immuni.2017.10.021 Medline

22. J. Sun, A. Ramos, B. Chapman, J. B. Johnnidis, L. Le, Y.-J. Ho, A. Klein, O. Hofmann, F. D. Camargo, Clonal dynamics of native haematopoiesis. *Nature* **514**, 322–327 (2014). doi:10.1038/nature13824 Medline

23. B. K. Tusi, S. L. Wolock, C. Weinreb, Y. Hwang, D. Hidalgo, R. Zilionis, A. Waisman, J. R. Huh, A. M. Klein, M. Socolovsky, Population snapshots predict early haematopoietic and erythroid hierarchies. *Nature* **555**, 54–60 (2018). doi:10.1038/nature25741 Medline

24. C. Trapnell, D. Cacchiarelli, J. Grimsby, P. Pokharel, S. Li, M. Morse, N. J. Lennon, K. J. Livak, T. S. Mikkelsen, J. L. Rinn, The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat. Biotechnol.* **32**, 381–386 (2014). doi:10.1038/nbt.2859 Medline

25. D. Gupta, H. P. Shah, K. Malu, N. Berliner, P. Gaines, Differentiation and characterization of myeloid cells. *Curr. Protoc. Immunol.* **104**, 1, 28 (2014). doi:10.1002/0471142735.im22f05s104 Medline

26. D. E. Wagner, C. Weinreb, Z. M. Collins, J. A. Briggs, S. G. Megason, A. M. Klein, Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo. *Science* **360**, 981–987 (2018). doi:10.1126/science.aar4362 Medline

27. M. M. Chan, Z. D. Smith, S. Grosswendt, H. Kretzmer, T. M. Norman, B. Adamson, M. Jost, J. J. Quinn, D. Yang, M. G. Jones, A. Khodaverdian, N. Yosef, A. Meissner, J. S. Weissman, Molecular recording of mammalian embryogenesis. *Nature* **570**, 77–82 (2019). doi:10.1038/s41586-019-1184-5 Medline

28. B. Spanjaard, B. Hu, N. Mitic, P. Olivares-Chauvet, S. Janjuha, N. Ninov, J. P. Junker, Simultaneous lineage tracing and cell-type identification using CRISPR-Cas9-induced genetic scars. *Nat. Biotechnol.* **36**, 469–473 (2018). doi:10.1038/nbt.4124 Medline

29. S. H. Naik, T. N. Schumacher, L. Perié, Cellular barcoding: A technical appraisal. *Exp. Hematol.* **42**, 598–608 (2014). doi:10.1016/j.exphem.2014.05.003 Medline

30. A. Gerrits, B. Dykstra, O. J. Kalmykowa, K. Klauke, E. Verovskaya, M. J. C. Broekhuis, G. de Haan, L. V. Bystrykh, Cellular barcoding tool for clonal analysis in the hematopoietic system. *Blood* **115**, 2610–2618 (2010). doi:10.1182/blood-2009-06-229757 Medline

31. A. M. Klein, L. Mazutis, I. Akartuna, N. Tallapragada, A. Veres, V. Li, L. Peshkin, D. A. Weitz, M. W. Kirschner, Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161**, 1187–1201 (2015). doi:10.1016/j.cell.2015.04.044 Medline

32. L. W. Plasschaert, R. Žilionis, R. Choo-Wing, V. Savova, J. Knehr, G. Roma, A. M. Klein, A. B. Jaffe, A single-cell atlas of the airway epithelium reveals the CFTR-rich pulmonary ionocyte. *Nature* **560**, 377–381 (2018). doi:10.1038/s41586-018-0394-6 Medline

33. H. Mi, A. Muruganujan, P. D. Thomas, PANTHER in 2013: Modeling the evolution of gene function, and other gene attributes, in the context of phylogenetic trees. *Nucleic Acids Res.* **41** (D1), D377–D386 (2013). doi:10.1093/nar/gks1118 Medline

34. M. Kanamori, H. Konno, N. Osato, J. Kawai, Y. Hayashizaki, H. Suzuki, A genome-wide and nonredundant mouse transcription factor database. *Biochem. Biophys. Res. Commun.* **322**, 787–793 (2004). doi:10.1016/j.bbrc.2004.07.179 Medline

35. G. Schiebinger, J. Shu, M. Tabaka, B. Cleary, V. Subramanian, A. Solomon, S. Liu, S. Lin, P. Berube, L. Lee, J. Chen, J. Brumbaugh, P. Rigollet, K. Hochedlinger, R. Jaenisch, A. Regev, E. S. Lander, Reconstruction of developmental landscapes by optimal-transport analysis of single-cell gene expression sheds light on cellular reprogramming. bioRxiv 191056 [Preprint]. 27 September 2017. https://doi.org/10.1101/191056.