Supplemental Figure Legends

Figure S1, related to Figure 1. Robustness analysis of Wanderlust. Robustness analysis of the Wanderlust algorithm, depicted in Figure 1. **A**) Wanderlust is robust to increasing amounts of noise. Each plot corresponds to a Wanderlust run on one of the synthetic datasets. The magnitude of the noise and the Pearson correlation between the solution trajectory and the Wanderlust trajectory are indicated in each plot's title. The X-axes are the solution trajectory (cells are ordered by the solution) and the Y-axes are the Wanderlust trajectories (ordered by Wanderlust). Each dot is a cell. The Wanderlust trajectories are highly correlated with the solution trajectory even when the noise's magnitude equals half of the maximum. **B**) Wanderlust is resilient to short circuits. The dataset with magnitude = 0.01 served as the basis for 8 synthetic datasets with varying amounts of short circuits (top, N=50, 100, 500, 1,000). The short circuit lengths were exponentially distributed with increasing μ (μ= 0.05, 0.09). (*left*) The adjacency matrix of the 30-nearest-neighbors graph of each N=1,000 dataset. Each point is an edge. The X- and Y-axes are node numbers, ordered by the solution trajectory. The diagonal is composed of the real neighbors while the surrounding cloud is the short circuits. As μ increases (*right*) the short circuits connect points that are more distant across the solution trajectory. Wanderlust runs of the synthetic datasets. The X-axes are the solution trajectory and the Y-axes are the Wanderlust trajectories. Pearson's correlation values are given at each plot's title. As the number of short circuits increases, Wanderlust remains well correlated with the solution trajectory unless too many short circuits are long-range. Even when the data includes many long-range short circuits, the algorithm provides a reasonable trajectory. **C**) Wanderlust is robust to early-cell parameter choice. The Wanderlust algorithm has been rerun four times with the early-cell parameter advancing across the baseline trajectory. X-axes are the baseline trajectory. Y-axes are Wanderlust's output for given 's'. Each dot is a cell along the trajectory. Pearson's correlation is given in title. The Wanderlust trajectory is well correlated with the baseline for s= 0.2 and 0.3, and is inversely correlated (the algorithm detects the reverse trajectory) for s=0.8. For s= 0.4, Wanderlust broke the trajectory in half and modeled each half correctly.

Figure S2, related to Figure 2. Wanderlust does not rely on any individual marker and can detect the distribution of marker expression across development. **A**) Correlation to original trajectory (Figure 2A) when indicated marker is omitted from the construction of the trajectory. Correlation values are generally high (greater than 0.97). One exception is HLA-DR (ρ=0.796), which did not affect the overall B cell ordering, rather only their position on the Wanderlust axis, moving everything earlier on the axis. **B**) Scatterplot representing correlation between original trajectory (X axis) and trajectory constructed using only 11 markers, when 6 classical B cell markers were omitted (CD10, CD19, CD20, CD79b, IgMi, IgMs). Data was uniformly downsampled to 50,000 cells to make plot clearer. Correlation remains high (ρ=0.95). Only 2.4% of the total cells reside in the pink box representing uncorrelated off-diagonal cells. **C**) Wanderlust traces after omission of 6 key markers demonstrates that general trends are maintained when compared to the original trajectory (Figure 2A) however, trajectory becomes noisy toward the end. Such noisy "trendless" behavior of the trace is a telltale sign for poor reconstruction of that segment. These cells at the end are exactly those represented in the pink box in panel B. **D**) By omitting these cells (Figure S2B, pink box), the resulting Wanderlust trace was now closer to the original trace in Figure 2A and correctly place the remaining 97.6% of cells. Note that only 3 markers included in this trace were used to build the trajectory (denoted by "*"). Taken together, Wanderlust is able to capture the progression of the system even without the inclusion of definitional markers. **E**) Distribution of marker expression across trajectory from overlapping windows. Green line indicates standard deviation of expression across trajectory. **F**) Cross correlation analysis showing Wanderlust traces for additional

analysis markers from 4 healthy individuals (sample A-blue, sample B-green, sample C-red, sample D-teal) demonstrating consistency of Wanderlust to determine the progression of these markers in healthy individual replicates.

Figure S3, related to Figure 3. Standard curves for qPCR analysis of IgH rearrangement and sorting strategy for prospective isolation of Populations II-V. A) Movement of CD24 and TdT expression when taken as increments of 0.05 across the early half (0-0.5) of the Wanderlust trajectory. **B**) Standard curve showing relative rearrangement versus qPCR fold change for differing concentrations of purified mature (CD20 positive) B cells (65% rearranged), CD3+ T cells (10% rearranged), and CD33+ myeloid cells (0.05% rearranged) as determined by flow cytometry. Correlation for $V_H(D)J_H$ rearrangement R^2=0.99809. **C**) Correlation for DJ rearrangement R^2=0.99982. **D**) Sorting gates for prospective isolation of populations II-V, as described in Figure 3, by FACS.

Figure S4, related to Figure 4. The nature of STAT5 phosphorylation in human B cell populations emerging in the marrow. A) phosphorylation of STAT5, normalized to the basal control across populations I-V. The response is reported as arcsinh difference for the treatment versus each individual's basal control. Maximum arcsinh differences were 1.7, 2.2, 1.1, 0.8, and 1 for the top left (JAK inhibitor – tofacitinib - and IL7Rα ligands), bottom left (Src family kinase inhibitor - dasatinib), and replicates A to C (pervanadate), respectively. **B**) Biaxial contour plots displaying IL-7rα and pSTAT5 across populations II-V in the basal, unstimulated condition (top row) and in response to treatment with IL-7 (bottom row). **C**) Biaxial contour plots displaying CD19 and pSTAT5 across populations II-V in the IL-7 stimulated condition.

Figure S5, related to Figure 5. Derivative analysis of additional bone marrow replicates. Results of the first derivative analysis of the Wanderlust trajectory for additional biological replicates (A) and (B), first replicate presented in Figure 5. Markers were hierarchically clustered based on absolute values. Following clustering, both replicates had the same marker ordering. Wanderlust was run independently on each replicate.

Figure S6, related to Figure 6. Flow cytometry analysis strategy and cell counts for B cell co-culture experiments A) Gating strategy for flow cytometry analysis of primary B cells after 6 weeks of OP-9 co-culture. **B**) Live cell counts in $CD34^+CD38^+$ fraction following 6 weeks of co-culture in presence of DMSO control or targeted kinase inhibitors. Two-tailed t-test, n=4.

Figure S7, related to Figure 5 and 7. Relative timing of marker expression coordination is conserved in biological replicates. To demonstrate consistency of developmental events represented in Figure 7 (coordinated marker increases or decreases in expression) across independently analyzed biological replicates, the CD20 rise, TdT decline, CD10 rise, and CD24 rise were set as key developmental pivot points on the Wanderlust trajectory. The relative timing of these pivot points in relation to other developmental events (columns) are indicated by their Wanderlust distance. Red indicates given event occurs prior to pivot point. Blue indicates given event occurs following pivot point. White indicates event occurring within a +/- five percentile window from pivot point. Biological replicates are A through D (Figure 2C and S2C) and H, the representative individual in the main figures. For the developmental events in the columns, the 'up' or 'down' arrow indicates the initial 'rise' or 'fall' of the indicated parameter observed in the first derivative analysis of that trace (Figure 5).

Supplemental Tables

Table S1, related to experimental procedures. Mass Cytometry Antibody Reagents. This table supports the mass cytometry analysis described in the experimental methods. Cellular antibody staining panel clones, suppliers, isotope reporter, and staining concentration. All DVS antibodies were purchased pre-conjugated. All other supplier antibodies were purchased in a 'carrier-free' format and conjugated with the respective metal isotope using the MaxPar-X8 conjugation kit (DVS Sciences) and custom ordered metal isotopes (Trace Sciences). The last column indicates whether the marker was used for Wanderlust's (WL) trajectory detection.

Protein	Clone	Manufacturer	Metal Isotope	Conc. (ug/mL)	Used in WL
CD10	HI10a	Biolegend	Gd156	1.5	X
CD117	104D2	Biolegend	Yb171	0.5	X
CD11c	Bu15	Biolegend	Yb173	2	
CD127	HCD127	Biolegend	Dy162	2	
CD16	3G8	Biolegend	Yb173	1	
CD179a	HSL96	Biolegend	Sm149	1	X
CD179b	HSL11	Biolegend	Gd158	1	X
CD19	H1B19	BD Biosciences	Nd142	2	X
CD20	2H7	DVS	Sm147	2	X
CD21	LT21	Biolegend	Nd150	3	
CD22	HIB22	Biolegend	Nd143	2	
CD23	EBVCS-5	Biolegend	Yb172	2	
CD235	HIR2	Biolegend	In113	2	
CD24	ML5	Biolegend	Gd160	1.5	X
CD27	O323	Biolegend	Tb159	0.75	
CD33	WM53	Biolegend	Yb173	2	
CD34	8G12	BD Biosciences	Nd148	2	X
CD38	HIT2	Biolegend	Er168	1	X
CD40	5C3	Biolegend	Er170	1	
CD43	CD43-10G7	Biolegend	Er167	1	
CD45	HI30	Biolegend	In115	2	X
CD45RA	HI100	Biolegend	La139	1	
CD49d	9F10	Biolegend	Gd160	1	
CD5	SK3	Biolegend	Nd144	1	
CD61	VI-PL2	BD Biosciences	In113	1	
CD62L	DREG-56	Biolegend	Tm169	2	
CD66b	G10F5	Biolegend	In113	2	
CD7	M-T701	BD Biosciences	Yb176	1	
CD72	3F3	Biolegend	Eu151	2	X
CD79b	CB3-1	Biolegend	Nd146	2	X
CXCR4	12G5	Biolegend	Lu175	3	
HLADR	L243	Biolegend	Yb174	2	X
IgD	IA6-2	Biolegend	Nd145	2	X
IgMi/IgMs (IgH)	polyclonal	Invitrogen	Eu153/Lu175	1	X
Kappa	MHK-49	Biolegend	Sm154	1	X
Lambda	MHL-38	Biolegend	Gd157	0.75	X
Pax5	1H9	eBioscience	Ho165	2	
PreBCR	HSL2	Biolegend	Ho165	1.5	
RAG1	D36B3	Cell Signaling Technology	Dy163	4	
TdT	E17-1519	BD Biosciences	Dy164	4	
Functional Markers					
Akt(pS473)	193H12	Cell Signaling Technology	Tb159	3	
Btk/Itk (pY551)	24a/BTK	BD Biosciences	Yb171	2	
cParp	F21-852	BD Biosciences	La139	1.5	
Creb (pS133)	87G3	Cell Signaling Technology	Yb176	2	
Erk1/2 (pT202/pY204)	D13	Cell Signaling Technology	Er167	1.5	
IkBα	L35A5	Cell Signaling Technology	Er166	2	
Ki67	B56	BD Biosciences	Sm152	2	
p38 (pT180/pY182)	36/p38	BD Biosciences	TM169	1.5	
PLCg2 (pY759)	K86-689.37	BD Biosciences	Pr141	2	
S6 (pS235/pS236)	N7-548	BD Biosciences	Yb172	1	
Src (pY418)	K98-37	BD Biosciences	Nd144	1	
STAT3 (pY705)	4/P-STAT3	BD Biosciences	Gb158	2	
STAT5 (pY694)	47	BD Biosciences	Nd150	2	

ZAP70/Syk (pY319/pY352)	17a	BD Biosciences	Er166	1	

Table S2, related to Figure 4. Human Marrow Stimulations. Stimulation conditions for mass cytometry experiments. Final concentrations for approximately 1 to 5 million cells in 1 ml volume for the indicated duration.

Perturbation	Final Concentration	Time-point	Manufacturer
Basal	----	----	----
IgM (BCR crosslink)	4 ug/mL	5 min.	Invitrogen
PVO	125 µM	15 min.	Sigma
IL-7	20 ng/mL	15 min.	BD Biosciences
Tofacitinib	500nM	30 min.	Selleck Chemicals
Dasatinib	100nm	30 min.	ChemieTek

Table S3, related to experimental procedures. Lineage Depletion Antibodies. Non-B cell lineage depletion antibody staining panel clones, suppliers, and staining concentration. *IgM, Kappa, and Lambda were only applied to the samples for the *ex vivo* BMMC 6 week cell culture.

Antibody	Clone	Manufacturer	Final Concentration
CD16	3G8	Biolegend	2ug/mL
CD14	HCD-14	Biolegend	2ug/mL
CD11c	3.9	Biolegend	2ug/mL
CD56	HCD56	Biolegend	2ug/mL
CD3	UCHT1	Biolegend	2ug/mL
CD66	G10F5	Biolegend	2ug/mL
Anti-kappa*	MHK-49	Biolegend	2ug/mL
Anti-lambda*	MHL-38	Biolegend	2ug/mL
Anti-IgM*	MGM-88	Biolegend	2ug/mL

Table S4, related to Figure 6. Ex Vivo B cell inhibitor culture conditions.. Small molecule inhibitors used in primary B cell co-cultures and molecular target. Final concentrations are given for 1ml of growth media.

Drug	Final Concentration (in 1ml)	Primary Target	Manufacturer
Control	DMSO (1:1000)	---	Sigma
Tofacitinib	100nM	Jak 3 Kinase	Selleck Chemicals
Ruxolitinib	2uM	Jak1&2 Kinase	ChemieTek
SB203580	1uM	p38 Kinase	Cell Signaling Technology

Supplementary Experimental Methods

Primary human bone marrow

For mass cytometry analyses, fluorescence activated cell sorting (for qPCR), and *ex vivo* cell cultures, fresh human bone marrow (BM) was obtained from healthy donors from AllCells, Inc. Samples were treated with Ficoll and processed as described in Bendall et al., *Science*, 2011.(Bendall et al., 2011) and were used immediately or were cryopreserved. Cryopreserved cells were thawed in 90% RPMI, 10% fetal calf serum (FCS) supplemented with 20 U/mL sodium heparin (Sigma) and 0.025 U/mL benzonase (Sigma), 1X L-glutamine, and 1X penicillin/streptomycin (Invitrogen). Freshly prepared samples were resuspended in 90% RPMI with 10% FCS.

Lineage depletion

Where indicated, BMMC preparations were lineage-depleted prior to cell culture or cytometric analyses. In all cases, except for the *ex vivo* cell culture, the preparations were fixed with 1.6% paraformaldehyde (PFA; Electron Microscopy Sciences, Hatfield, PA) for 10 min prior to depletion. Cells were then washed with staining media (CSM: PBS with 0.5% BSA, 0.02% sodium azide). For depletion, cells were stained with biotin-conjugated antibodies (Table S3) for 30 min at a concentration of 5 million cells per 100 µl. Cells were washed with CSM twice then incubated with BD Streptavidin Particles Plus (BD Biosciences) at the manufacturer's recommended concentration for 30 min at room temperature. Particle-labeled cells were resuspended in CSM to approximately 2-8x10^7 cells/ml and placed in a magnetic holder for 7 min. The supernatant was transferred to a new tube, and the beads/cells were washed and resuspended and placed back in the magnetic holder for an additional round of depletion and supernatant recovery. This washing procedure was repeated twice. Cells from the supernatant were then concentrated by centrifugation at 250 *g* for 5 min.

Mass cytometry analysis

BMMCs were first stained for viability using cisplatin as described by Feinberg et al.(Fienberg, Simonds, Fantl, Nolan, & Bodenmiller, 2012). Cells were then rested for 30 min at 37 °C and perturbed with various stimuli and inhibitors prior to analysis (Table S2). Following perturbation, cells were immediately fixed with 1.6% PFA (Electron Microscopy Sciences) for 10 min. Cells were then washed with CSM and Fc receptor block was performed using Human TruStain FcX (Biolegend) following manufacturer's instructions. Cells were then stained for surface proteins at room temperature for 30 min. Following staining, cells were washed twice with CSM and permeabilized with methanol pre-cooled to 4°C for 10 min. Cells were then washed twice with CSM and stained for intracellular proteins for 30 min at room temperature. Surface and intracellular staining cocktails are listed in Table S3. Cells were washed with CSM and stained with 1 mL of 2000x Ir DNA intercalator (diluted 1:5000 in PBS with 1.6% PFA; DVS Sciences) for 20 min at room temperature or overnight at 4 °C. Prior to CyTOF analysis, cells were washed once with CSM and then twice with ddH_2O.

Mass cytometry analysis data pre-processing

To make all samples maximally comparable, data were acquired using internal metal isotope bead standards as previously described(Finck et al., 2013). Cell events were acquired at approximately 500 events per second on a CyTOF I (DVS Sciences) using instrument settings previously described(Finck et al., 2013). Each patient sample was individually normalized to the internal bead standards prior to analysis. To remove dead cells, debris, and non-B cell types, cells were gated based on cell length and DNA content as described in Bendall et al. (Bendall et al., 2011) and for cisplatin negativity. Remaining cells were filtered for high expression levels of CD3 (T cells), CD33 (myeloid), CD11c (dendritic cells), CD16 (NK cells), CD235 (erythrocytes), and CD61 (platelets) prior to Wanderlust analysis or manual population gating.

Mass cytometry contour plots, histograms, signaling heatmaps, and signaling inductions analysis were performed on www.cytobank.org(Kotecha, Krutzik, & Irish, 2010). All Wanderlust-based analyses (including marker traces and derivative analyses), were performed using MATLAB (Mathworks) and our customized MATLAB GUI Cyt (Amir et al., 2013)..

Cell sorting and DNA extraction

Lin^- BMMCs were stained for surface antigens using FITC-CD34 clone 8G12 (BD Biosciences) and BV711-CD38 clone HIT2 (BioLegend) in 100 µl total volume for 30 min at room temperature in the dark. Cells were washed with cell staining mdia (CSM) then permeabilized with methanol pre-chilled to 4 °C for at 10 min at 4 °C. Cells were then washed twice in CSM and stained intracellularly with BV421-CD24 clone ML5 (BioLegend) and PE-TdT clone E17-1519 (BD Biosciences) in a total volume of 100 µl for 30 min at room temperature in the dark. Cells were washed with CSM prior to sorting on a BD FACS Aria. Supplemental Figure 3C shows the cell sorting gating schema. Following sorting, DNA was extracted from cellular subsets using RecoverAll Total Nucleic Acid Isolation Kit for FFPE as per the manufacturer's instructions (Invitrogen).

RT-PCR analysis of IgH rearrangements

Relative IgH rearrangement status in genomic DNA was assessed using quantitative PCR using with primer cocktails adapted from Van Dongen *et. al*.(van Dongen et al., 2003). $V_H(D)J_H$ rearrangements were detected using the "A" and "B" primer cocktails and DJ_H rearrangements were detected using the "D" primer cocktail. A control primer set to *PLZF* was also used to assess the quality of DNA and calculate the ΔCT. Primers (10 pmol) were used in a reaction with ~20 ng of unamplified genomic DNA from sorted populations. DNA from each sorted population was combined with primers and SYBR Green PCR Master Mix (Qiagen, Inc.) and each reaction was performed in triplicate. Samples were analyzed using RotorGene RG3000 (Corbett Research, Inc.) with the following program: 15 min hold at 95 °C followed by 45 cycles

of 30 s at 95 °C, 30 s at 60 °C, and 45 s at 72 °C. Data shown are representative of triplicate analyses of two donors analyzed on separate occasions. For each sample, fold-change was calculated relative to a germline (non-B cell) genomic DNA control. For validation, a standard curve representing relative rearrangement versus qPCR fold change was established utilizing differing concentrations of purified mature (CD20 positive) B cells as confirmed by flow cytometry (Figure S3). For the relative rearrangement of sorted , populations II to V, qPCR fold change values were normalized to fraction II.

Primary B cell co-culture

OP-9 progenitor cell co-cultures for B cell specification were performed as previously described(Sanz et al., 2010).. Briefly, live, Lin^- BMMC from two separate donors, were each resuspended at 10^6 cells/ml and divided into four wells in RPMI, 2 mM L-glutamine with 1x Pen/Strep, 30 µM BME, and 3% FBS. Cells were treated as indicated in Table S4. A 24-well, flat bottom plate was pre-coated with a ~60% confluent layer of OP-9 cells. Cells were maintained at 37 °C in 5% CO_2 for 6 weeks. Media and inhibitors were changed twice weekly. Cells were harvested and pelleted by centrifugation at 300 *g* for 5 min. Cells were resuspended in CSM and fixed with PFA at a final concentration of 1.6% for 10 min at room temperature. For analysis, non-adherent cells were removed from wells by rinsing with cell staining media. To increase cell numbers for analysis, four wells were pooled into two for each condition before analysis. Cells were characterized by flow cytometry using the staining scheme described in the cell sorting methods and analyzed on a BD LSRII.

Additional Computational Methods and Optimization

The Wanderlust algorithm

The development of Wanderlust was premised on three assumptions about the data. First, a tissue sample taken at a single time point will include sufficient cells to represent an entire continuous developmental process, including intermediate and rare cell populations. Second, the developmental process is non-branching and linear: cells can either proceed along development, stop progression, or “fall off” the progression (i.e. leave the tissue, undergo apoptosis, etc.). Third, cell maturation is continuous; changes in marker expression are gradual, and therefore the transitions between stages are gradual. Given this, the algorithm uses marker expression levels to best order developmental stages from a set of cells to enable tracing of a most likely trajectory. In this report the B cell maturation process of human was used as a template on which to test the robustness of the approach.

The main conceptual framework that underlies Wanderlust is the representation of the data as a graph. The data is converted into a k-nearest neighbor graph (k-NNG). Each cell is represented as a node and is connected to its k neighbors, the cells most similar to it, via an edge whose weight is set by the similarity, defined below, between the cells. We define the shortest-path distance between a pair of cells as the length of the path between the nodes that minimizes the sum of weights of its constituent edges. The shortest-path distance is composed of transitions through neighbors, where each transition constitutes a gradual “step”. To find the trajectory, we would start from the earliest cell in the data and order the rest of the cells according to their shortest-path distance from it. Based on the third assumption, above, this approach would provide an approximation to the developmental order between cells.

A graph representation of the data addresses several problems. The conversion into a graph is computationally cheap, as shortest-path calculations are fast due to the graph’s

sparsity; this enables the analysis of large datasets in reasonable time. The data model is based on similarity between cells rather than on distance relationships between parameters and can therefore handle non-linearity. The magnitude of the noise is expected to be proportional to the distance; therefore since the local neighborhood is based on short distances, this approach is less susceptible to noise (since short distances are reliable, and our representation of long distances as graph traversal mitigates noise from long distances). Finally, even if two developmentally close cells are not direct neighbors they still reside in the same region in the high-dimensional space. Therefore, they will be separated by a small number of close neighbors and the shortest-path distance between them will be low, circumventing much of the variability in the data.

However since markers are rarely monotone along development, the k-NNG is still susceptible to "short circuits": cells connected via an edge simply due to close proximity in the high dimensional space and not due to close developmental chronology. With over ten thousand cells in a typical dataset, since developmentally distant cells bear similar levels for at least some markers along noise typical of high throughput biology, it is virtually impossible to avoid at least a few such "short circuits". Even with only one short circuit in the graph, a shortest-path between developmentally distant cells will use the short circuit routes, leading to incorrect trajectories. A possible solution for this problem is to use a random-walk based distance measure. However, random walks are computationally intensive as they go both towards and away from the target and are not practical in large graphs, such as those contemplated here. To address this we utilize randomness to address the noise in the data. Given good data, short circuits are rare in our construction; therefore, a random subset of the graph is likely to include only a few short circuits.

An algorithmic innovation of our technique overcomes this obstacle by extending the graph representation into an ensemble of l-out-of-k-nearest neighbor graphs (l-k-NNG)[1]. An l-k-NNG is generated by starting with the k-NNG and iterating over each node in the graph, randomly keeping only l of its k-nearest neighbors. On average, a given short circuit will only exist in 2l/k of the graphs (2l/k being the probability that that specific edge will be one of the l edges chosen out of the k edges, for each of the two nodes that the edge connects); in these graphs the shortest-path distances will be distorted by that short circuit. By picking l lower than k, each short circuit only appears in a few l-k-NNGs and influences a different set of cells. Its effect is averaged away by taking the mean trajectory over all graphs.

Shortest-path distances raise two important complications. First, distances do not have a direction while the trajectory does. In order to address this issue we require a user-defined "initiator" cell. The initiator cell is assumed to reside toward the beginning of the trajectory and is used for orientation. Second, the shortest-path distance variability increases with the distance. As nodes get farther from each other, the accuracy of their short-path distance decreases, since mistakes accumulate. Therefore, the distance between two distant cells is less reliable than between two close cells.

A second major innovation of the algorithm is to utilize randomly chosen cells from the entire dataset as "waypoints" that support trajectory assembly by breaking it into shorter distances. To increase the accuracy of the cell ordering along the entire trajectory, we randomly (following a uniform distribution) flag a set of cells as waypoints. In the naïve approach (without waypoints) only the initiator cell was used to mold the distance along the path to cells along the trajectory. The waypoints serve as reinforcements to the early cell: the position of each cell will now be calculated as the average of its distance from all of the waypoints. Additionally, we weigh the contribution of each waypoint according to its distance from the cell, further

[1] Please note that l stands for lowercase l, not for the digit 1.

reinforcing the influence of short distances and reducing the influence of long distances. Since at least some of the waypoints will be close to the cell, this allows us to get a better approximation of its position in the trajectory.

An outline of Wanderlust

Wanderlust begins with a two-step initialization step (Figure 1c, top left). First, a set of cells is randomly chosen as waypoints. Then, the data is transformed into an ensemble of l-k-NNGs. The algorithm proceeds by iteratively calculating the trajectory in each of the graphs separately: for each cell (referred to as a target), the target's position along the trajectory is first set to the shortest-path distance from a user-defined initiator cell *s*. The target's position is refined according to the shortest-path distance from each waypoint. The distances are weighed so that waypoints closer to the target contribute more to the calculation (as they are less susceptible to the noise inherent in the shortest-path distance). However, the waypoints are themselves cells. Therefore, their position will change following the refinement based on the same calculation that was applied to the rest of the cells. Since all cell positions depend on waypoint positions, the shift in waypoints might obsolete the newly calculated positions. Therefore, the refinement step is repeated with the new waypoint positions until the positions of all cells converge. Once the trajectory calculation completes in all of the graphs, the output trajectory is set to the average over all graph trajectories.

Formal description of Wanderlust

Wanderlust receives as input a list of *N* points in *D* dimensions. Each point is a cell represented by a vector of length *D*, where each element is a measurement of the intensity of one marker. The algorithm assumes that the cells lie upon a one-dimensional developmental trajectory. In addition, Wanderlust receives an early initiator cell *s* that serves as a starting point for the trajectory detection. As its name implies, *s* is expected to originate from the beginning of the trajectory. For each cell, Wanderlust outputs a continuous trajectory score which provides

the cell's temporal position across development: *s* has a score of zero and the most mature cell has a score of one, with the rest of the cells in between.

Wanderlust is composed of two steps: an initialization step and an iterative trajectory detection step. In the initialization step, Wanderlust flags a set of cells to serve as waypoints. The waypoint selection is done uniformly at random and therefore uses no prior information about the data or its underlying developmental process. The waypoints buffer against noise: each cell is going to have a waypoint nearby, reducing the variability in calculating its position across the trajectory.

Next, the data is converted to a k-nearest-neighbor graph (k-NNG): each cell is represented by a node and is connected via an edge to the k cells most similar to it. The edge weights are equal to the distance between the two nodes. The graph is represented as an adjacency matrix, where each row and each column are a cell, and the value at position *(k, l)* corresponds to the weight of the edge between nodes *k* and *l*. The k-NNG is used as a template for the generation of an ensemble of l-out-of-k-nearest-neighbor graphs (l-k-NNG). A single l-k-NNG is generated by randomly and uniformly picking l neighbors out of the k-nearest-neighbors for each cell. As with the waypoints, the construction of a random ensemble mitigates noise, since a spurious edge in the k-NNG will be absent from most l-k-NNGs randomly derived from it.

After waypoints have been chosen and the l-k-NNG ensemble constructed, the trajectory calculation step begins. This is an iterative process that is done separately for each l-k-NNG. First, we define the shortest-path distance between each pair of nodes *(s, t)* as:

$$shortest - path - distance(s,t) = min_P \sum_{e \epsilon P} w(e)$$

where *P* is a path between *s* and *t*, *e* is an edge and *w(e)* is the weight of *e*. We calculate the shortest-path distances using Dijkstra's algorithm which has a running time of O(|E|+|V|log|V|),

when |E| is the number of edges and |V| is the number of nodes. Briefly, Dijkstra's algorithm initializes the distance from *s* to all other nodes in the graph to infinity. The algorithm recursively scans the graph, at each step updating the distances to any nodes that can be reached. The algorithm stops when *t* is reached.

For each cell, the trajectory score is initialized to the shortest-path distance to that cell from the initiator cell that was supplied as a parameter to the algorithm. We define this score as the initial orientation trajectory.

Next, for each cell (referred to as target), the shortest-path distance is calculated between each waypoint and the target. However, distance does not have a direction: we cannot separate between the cases where the target precedes a landmark and where the target follows a landmark (figure M1, top and center). Therefore, an orientation step follows, where we utilize the initial orientation trajectory to decide on cell ordering relative to each waypoint (figure M1, bottom):

given initiator cell s then for each target cell t and waypoint l,

if d(s,t) < d(s,l): t precedes l
otherwise : t follows l

Additionally, graph traversal is in itself a source of noise and is proportional to the shortest-path distance: as the distance between two nodes increases many more possible paths exist between them, leading to higher variance in the traversed distance. Therefore, the distance from the target to a nearby waypoint has lower variability than the distance to a distant one. This can be leveraged in minimizing the noise by defining a weight for each waypoint:

$$w_{l,t} = \frac{d(l,t)^2}{\sum_m d(l,m)^2}$$

The summation at the denominator is over all target cells *m*; the weight of each waypoint is exponentially proportional to its distance from the target. The trajectory score for *t* is the weighted average over all waypoint distances:

$$traj_t = \sum_l \frac{d(l,t)}{nl} w_{l,t}$$

where the summation is over all waypoints *l* and *nl* is the total number of waypoints. In this weighing scheme closer waypoints, whose distance to the target is less noisy, have a higher weight in its trajectory score.

During this step the initiator cell and each waypoint also become target cells and their trajectory score changes. We use this trajectory score as a new orientation trajectory and repeat the orientation step. Since the graph itself does not change, the shortest-path distances do not change and we can use the existing distances in the repeated trajectory score calculation. Waypoint positions continue to change with each orientation step, which is repeated until waypoint positions converge:

$$converge?(\square_{t-1}, L_t) = \rho_{L_{t-1},L_t} == 1$$

where L_i is a vector of waypoint positions at orientation step *i* and ρ is Pearson's correlation.

The l-k-NNG's trajectory is equal to the trajectory of the last orientation iteration.

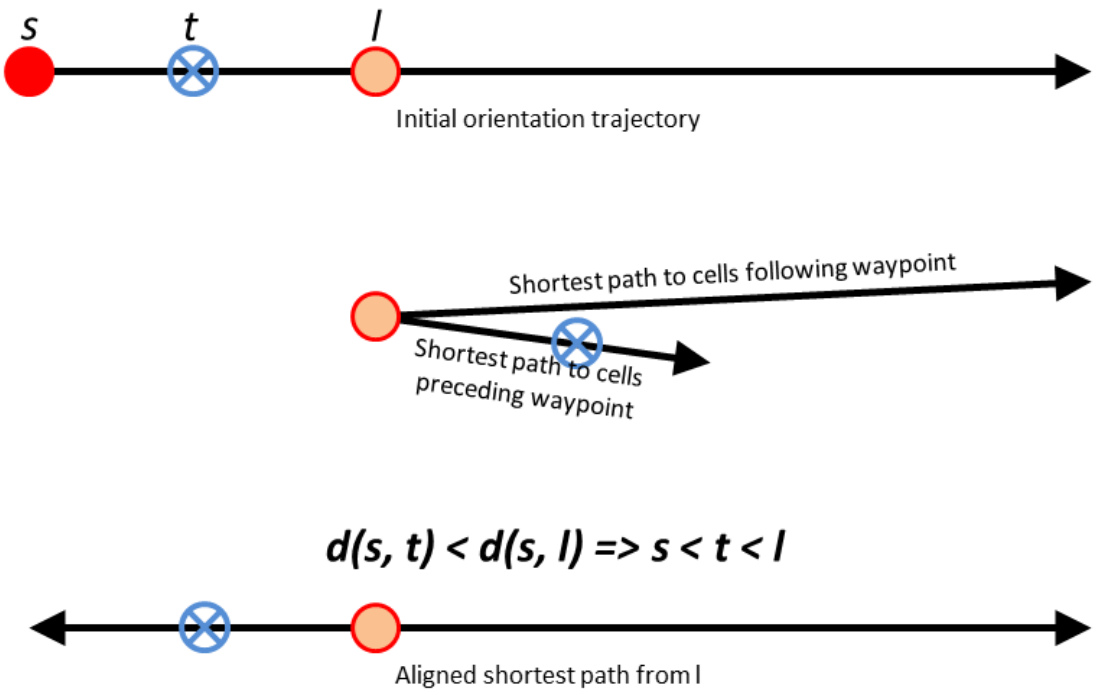


Figure M1. The orientation step of the Wanderlust algorithm.
Top: the initial orientation trajectory from the initiator cell (*s*) to a target cell (*t*) and a waypoint (*l*). center: the shortest-path distance from the waypoint to the target cell. Since distance has no direction we cannot identify whether the target cell precedes or follows the waypoint. Bottom: According to the initial orientation trajectory, the distance from the initiator cell to the target is lower than its distance to the waypoint. Therefore, the target must be between the initiator cell and the waypoint and we can orient the shortest-path distance from l accordingly.

Finally, after a trajectory is iteratively calculated for each l-k-NGG graph, the output trajectory is

set to the average over the trajectory scores of all l-k-NNG graphs.

Pseudo-code of the algorithm

Input: data set of cells $X = \{x_1, \ldots, x_n\}$, initiator cell *s*, number of waypoints *nl*, distance function *dist*, ensemble parameters: size of ensemble *ng*, number of nearest neighbors *k*, subset size *l*
Output: trajectory score $S = \{s_1, \ldots, s_n\}$ for each cell x_i
Initialization:
pick from *X* *nl* cells uniformly at random to serve as waypoints; set *s* as first waypoint → $\{l_1 = s, \ldots, l_{nl}\}$
calculate *k*-nearest-neighbor graph of *X* → *G*
randomly generate *ng* *l*-out-of-*k*-nearest-neighbor graphs → $G_{1, \ldots,} G_{ng}$
Trajectory calculation:
for each *l*-out-of-*k*-nearest-neighbor graph
calculate shortest-path distance from each waypoint l_j to each point x_i → $D = d_{ij}$
set w_{ij} to $|d_{ij}|^2 / \Sigma_k |d_{ik}|^2$

realign: calculate realigned distance T = t_{ij}. for each waypoint l_j

$t_{ij} = d_{ij}$ if $d_{1i} > d_{1j}$, $-d_{ij}$ otherwise

$t_{ij} = t_{ij} + d_{1j}$

set $traj_{1,i}$ to $\Sigma_j t_{ij}/nl * w_{ij}$

repeat until convergence

realign t_{ij} using $traj_{(iter-1),i}$

set $traj_{iter,i}$ to $\Sigma_j t_{ij}/n_l * w_{ij}$

set the graph's trajectory to *traj* of the last iteration

return average over all graph trajectories in ensemble → S

Wanderlust recapitulates the trajectory in noisy synthetic data

To evaluate Wanderlust's performance we first applied it to synthetic data composed of a series of simulated datasets. A curved, one-dimensional simulated trajectory, embedded in 3-dimensions, was generated by starting at position (1, 1, 1) and randomly traversing the space for 10,000 steps. After each step the current position was added to the trajectory as a point. All datasets had the same solution trajectory (figure M2, top). Each dataset additionally included seven dimensions of normally-distributed noise. The mean of each noise dimension was zero. The magnitude of each noise dimension was defined as the standard deviation divided by the range of the solution trajectory. Each dataset used the same magnitude for all seven noise channels. A total of eight datasets were generated with increasing magnitude, from zero (no noise) to one (noise magnitude equals the range of the solution trajectory). In total, this synthetic data included eight datasets, each of which had 10,000 points and ten dimensions; regardless of the noise magnitude, each dataset included seven noise dimensions and only three trajectory dimensions. The algorithm parameters were the same as those used in later analysis of biological data.

We computed a Wanderlust trajectory for each synthetic dataset and compared the resulting trajectory to the solution trajectory (figure M2, bottom). When there was no noise, the algorithm's output was almost identical to the solution (Pearson's $\rho=1$). Wanderlust continued to faithfully recapture the trajectory as noise levels increased to magnitude as high as 0.2 (Pearson's $\rho=0.97$). When the magnitude was increased to 0.5, output quality decreased as the entire first half of the trajectory was given a similar score of 0.2 by the algorithm. The second half, however, was well modeled, giving a reasonable view of the system (Pearson's $\rho=0.86$). As expected, when the magnitude reached 1, the algorithm was no longer able to detect the solution trajectory. Over the eight datasets, Wanderlust perfectly detected the solution trajectory in six datasets, reached high correlation with one dataset, and failed to detect the trajectory in

the last dataset (where the magnitude of the noise was equal to seven times the solution trajectory).

We next tested whether Wanderlust continued to correctly detect the trajectory even when the data included short circuits. We chose the synthetic dataset with the lowest noise magnitude, 0.01 (Pearson's ρ=0.99), as a template for the generation of sixteen short-circuit datasets, since we wanted to isolate the effect of the short circuits (figure M3a). Dataset generation included two parameters. The first parameter, N, was the number of short circuits (N=50, 100, 500, 1,000). The length of each short-circuit was randomly sampled from an exponential distribution whose mean, μ, was the second parameter (μ=0.01, 0.05, 0.09, 0.13). When μ=0.01, very few of the short circuits were long-range. However, as μ increased, short-circuit length increases, and, more specifically, more long-range short circuits appeared. We expected the detection quality to be inversely correlated with the number of short circuits and the proportion of long-range short circuits.

The Wanderlust trajectories were well-correlated with the solution trajectory (Pearson's ρ>0.95) in ten of the sixteen datasets (Figure M3b). As long as most of the short circuits were short-range (μ=0.01, 0.05) the number of short circuits (N) had only a slight effect on the algorithm's output. When μ=0.09, Wanderlust detected the trajectory well until N increased to 500 (Figure M3b, third row). Even when N was equal to 500 or 1,000, the algorithm gave a reasonable solution (Pearson's ρ=0.92).

We observed an interesting inversion observed when μ=0.13 and most of the short circuits were long-range (Figure M3b, fourth row): with a small number of short circuits (N=50), Wanderlust modeled the first half of the trajectory well, but then backtracked and included the second half of the solution trajectory as a plateau, leading to a ridge in the scatter plot (Pearson's ρ=0.59); however, as N increased, so did the correlation between the algorithm's

trajectory and the solution (Pearson's ρ=0.74, 0.83 and 0.85 when N=100, 500 and 1,000, respectively). When examining the scatter plots we saw that as more short circuits were added, less of the trajectory was modeled well, with the rest becoming a plateau (similar to the magnitude=1 dataset in Figure M2). The improved correlation is an artifact caused as the ridge in N=50 changes into noise.

Overall, Wanderlust detected the solution trajectory in the synthetic data in almost all of the datasets. Despite increasing levels of noise and the incorporation of varying quantities of short circuits of varying lengths, the algorithm reached a high correlation with the embedded solution trajectory. Wanderlust's high degree of robustness allowed it to overcome many of the challenges we expect trajectory detection to face in real data.

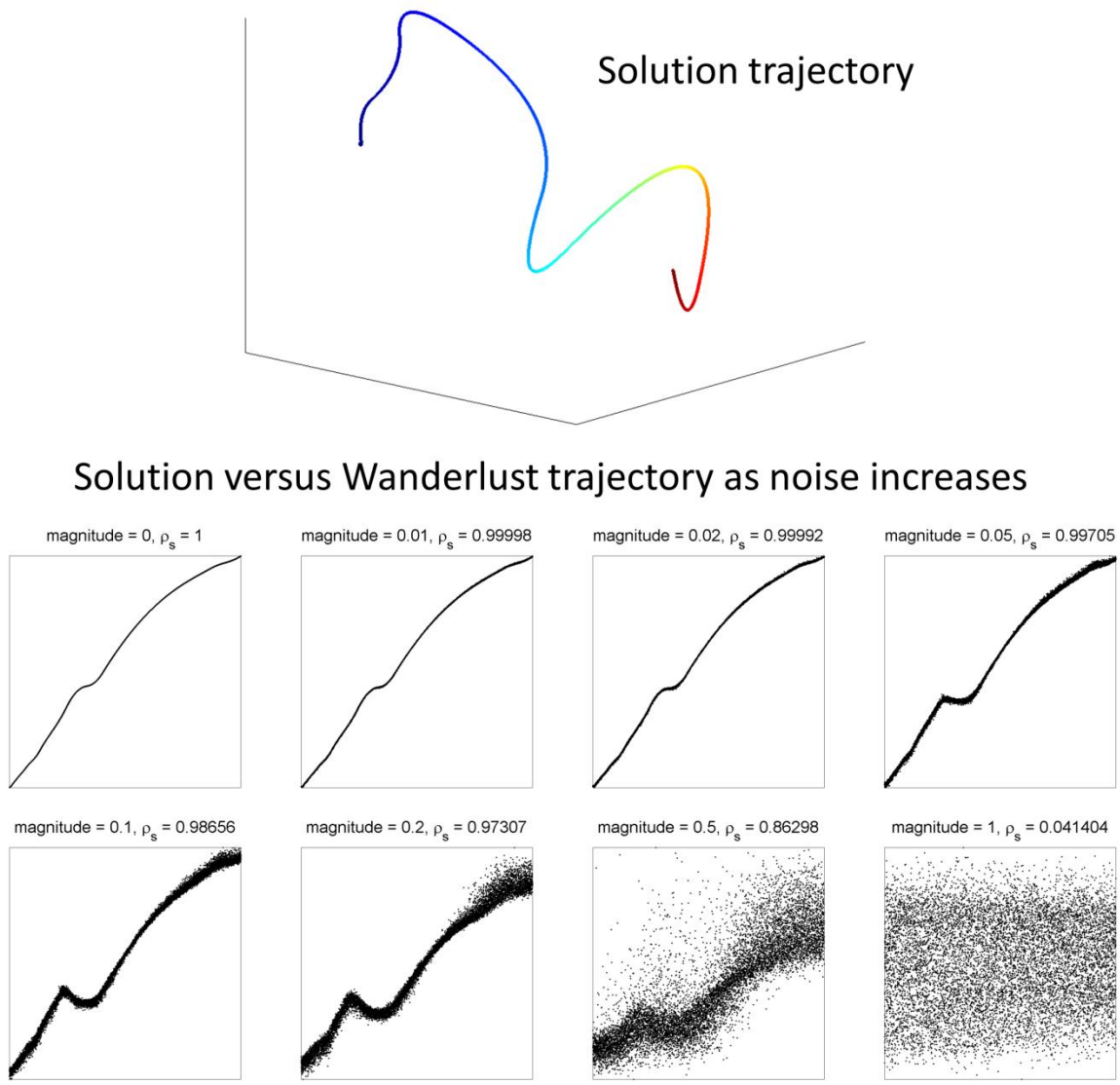


Figure M2. Trajectory detection in synthetic data with increasing amounts of noise. Top: The synthetic datasets are composed of a 1-dimensional curve (trajectory) embedded three-dimensions (beginning colored in blue, end colored in red) and seven dimensions of normally distributed noise with μ=0 and increasing magnitude (σ/range of solution trajectory; magnitude = 0, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1). Bottom: Each plot corresponds to a Wanderlust run on one of the synthetic datasets. The magnitude of the noise and the Pearson correlation between the solution trajectory and the Wanderlust trajectory are indicated in each plot's title. The X-axes are the solution trajectory (cells are ordered by the solution) and the Y-axes are the Wanderlust trajectories (ordered by Wanderlust). Each dot is a cell. The Wanderlust trajectories are highly correlated with the solution trajectory even when the noise's magnitude equals half of the maximum.

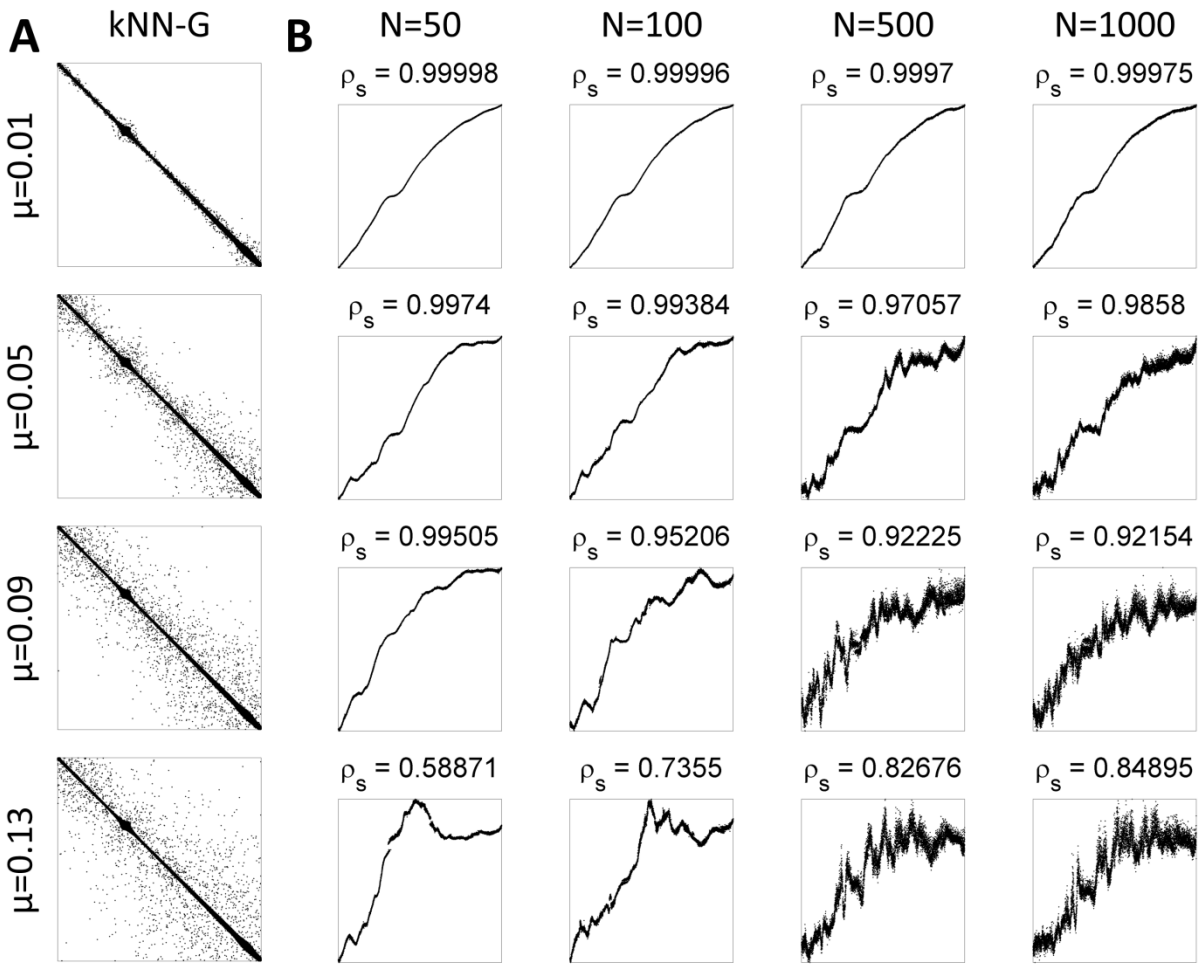


Figure M3. Wanderlust is resilient to short circuits.
The dataset with noise magnitude = 0.01 served as the basis for 16 synthetic datasets with varying amounts of short circuits (top, N=50, 100, 500, 1,000). The short circuit lengths were exponentially distributed with increasing μ (left, μ=0.01, 0.05, 0.09, 0.13). (a) The adjacency matrix of the 30-nearest-neighbors graph of each N=1,000 dataset. Each point is an edge. The X- and Y-axes are node numbers, ordered by the solution trajectory. The diagonal is composed of the real neighbors while the surrounding cloud is the short circuits. As μ increases the short circuits connect points which are more distant across the solution trajectory. (b) Wanderlust runs on these synthetic datasets. The X-axes are the solution trajectory and the Y-axes are the Wanderlust trajectories. Pearson's correlation values are given at each plot's title. As the number of short circuits increases, Wanderlust remains well correlated with the solution trajectory unless too many short circuits are long-range. Even when the data includes many long-range short circuits, the algorithm provides a reasonable trajectory.

Wanderlust parameters

The following parameters were used in all Wanderlust runs unless otherwise stated.

Parameter name	**Description**	**Value**
s	The early cell that is used in calculating the initial orientation trajectory Synthetic data: (1, 1, 1), the point from which the simulated trajectory originates. Mass cytometry healthy bone marrow data: the cell with *max(CD34-IgM)* *	
k	Number of neighbors of each node in k-nearest neighbors graph.	30
l	Number of neighbors selected for each node in each l-k-nearest neighbors graph in the ensemble	5
ng	Number of graphs in the l-k-nearest neighbors graph ensemble	20
nl	Number of landmarks	20
* This criterion was set in order to choose a stem cell (CD34+) while avoiding a possible stem cell-mature cell doublet (which would be CD34+IgM+, a non-physiological condition).		

Table M1. Default Wanderlust parameters
A list of the Wanderlust parameters, explanation of each parameter and the default parameter used in all experiments (unless otherwise stated).

All ten dimensions were used as input for the synthetic data. For the B-lineage data, the following surface markers were used: CD45, CD19, CD22, IgD, CD79b, CD20, CD34, CD179a, CD72, intracellular and surface IgM, Kappa, CD10, Lambda, CD179b, CD49d, CD24, CD127, CD38, CD40, CD117, HLADR.

The output trajectory was normalized using the following equation:

$$\square_{norm} = \frac{w - p_5}{p_{95} - p_5}$$

where *w* is the output trajectory and p_i is the *ith* percentile of the output trajectory. By using the 5th/95th percentiles we avoid cases where a distant outlier cells skews the rest of the trajectory.

Cosine distance

The variable scale and dynamic range of the values representing each marker do not account for (and may be counter to) the marker's importance in the biological system for both technical and biological reasons. On the technical side, each antibody has differences in affinity and biochemical properties and therefore, similar to microarray technology, it is inappropriate to compare quantitative values between two different protein epitopes, but rather only compare the same protein epitope between different cells or samples. Moreover, abundance does not imply physiological importance (for example CD45 vs. CD10) – markers appearing in lower copy number can sometimes have a more important phenotypic impact. Most standard metrics such as the L^1 norm, cityblock distance, or the L^2 norm, Euclidean distance, are sensitive to scale and give more weight to higher scales.

As such, even though Euclidean distance was used in Wanderlust runs on the synthetic data, measuring cell-to-cell distances in biological data required a different metric. To address this problem, we provide Wanderlust with the cosine distance, which is defined for a pair of cells (*s, t)* as:

$$d_{(s,t)} = 1 - \frac{x_s x_t{}'}{\sqrt{(x_s x_s')(x_t x_t')}}$$

where x_i is the vector representing cell *i*. $d_{(s,t)}$ is equal to *1-cos(θ)*, where *θ* is the angle between x_s and x_t in the high-dimensional space, hence the name cosine distance. Cosine distances are scale-independent and their distance distribution remains relatively constant after scaling changes (figure M4b, bottom).

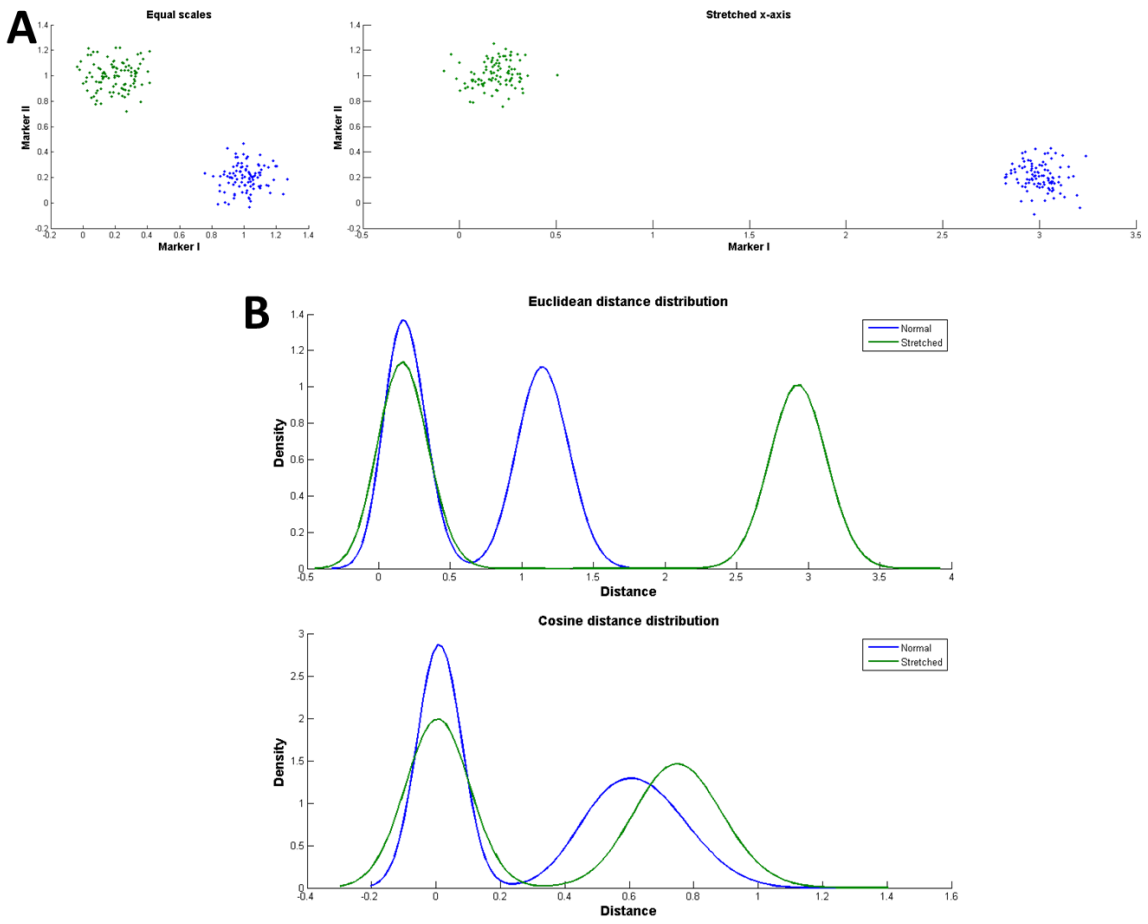


Figure M4. Cosine distances are scale-independent.
(a) Left: In this toy example two separate clusters have been randomly sampled from a two-dimensional Gaussian with center (0.2, 1) (green) or (1, 0.2) (blue). The two dimensions have equal scales. Right: The X-axis has been stretched by a factor of three, shifting the blue cluster to the right. (b) Top: The L2 norm (Euclidean distance) distribution in the equal scale data (blue) and the stretch X-axis data (green). We see that intra-cluster distances remain the same, while inter-cluster distances are shifted by the same factor as the stretching of the X-axis. Bottom: The cosine distance distribution in each dataset. Intra-cluster distances still remain the same and the inter-cluster distances are much closer to each other.

Wanderlust is robust when applied to mass cytometry data

Wanderlust is robust over multiple runs and different individuals

The Wanderlust algorithm initialization includes two stages that require random choices: the l-k-NNG ensemble generation (choosing of l out of the k-nearest neighbors for each node) and the landmark selection. These random processes could influence the output trajectory and lead to different results. To evaluate the robustness of the algorithm and reject this possibility, we re-ran the Wanderlust algorithm five times using the same healthy bone marrow sample data that appears in figure 2 (main text). The five runs were executed independently and each started from a different seed for the random number generator. The cell orderings over the five runs were almost identical (figure M5) with Pearson correlation greater than 0.99. While the second half of the trajectory seems to have more variation between pairs of runs, this was a visual artifact caused by the higher number of cells in that region of the trajectory (which was composed of more mature cells). In summary, Wanderlust is a robust algorithm that provides a consistent trajectory over multiple runs on the data.

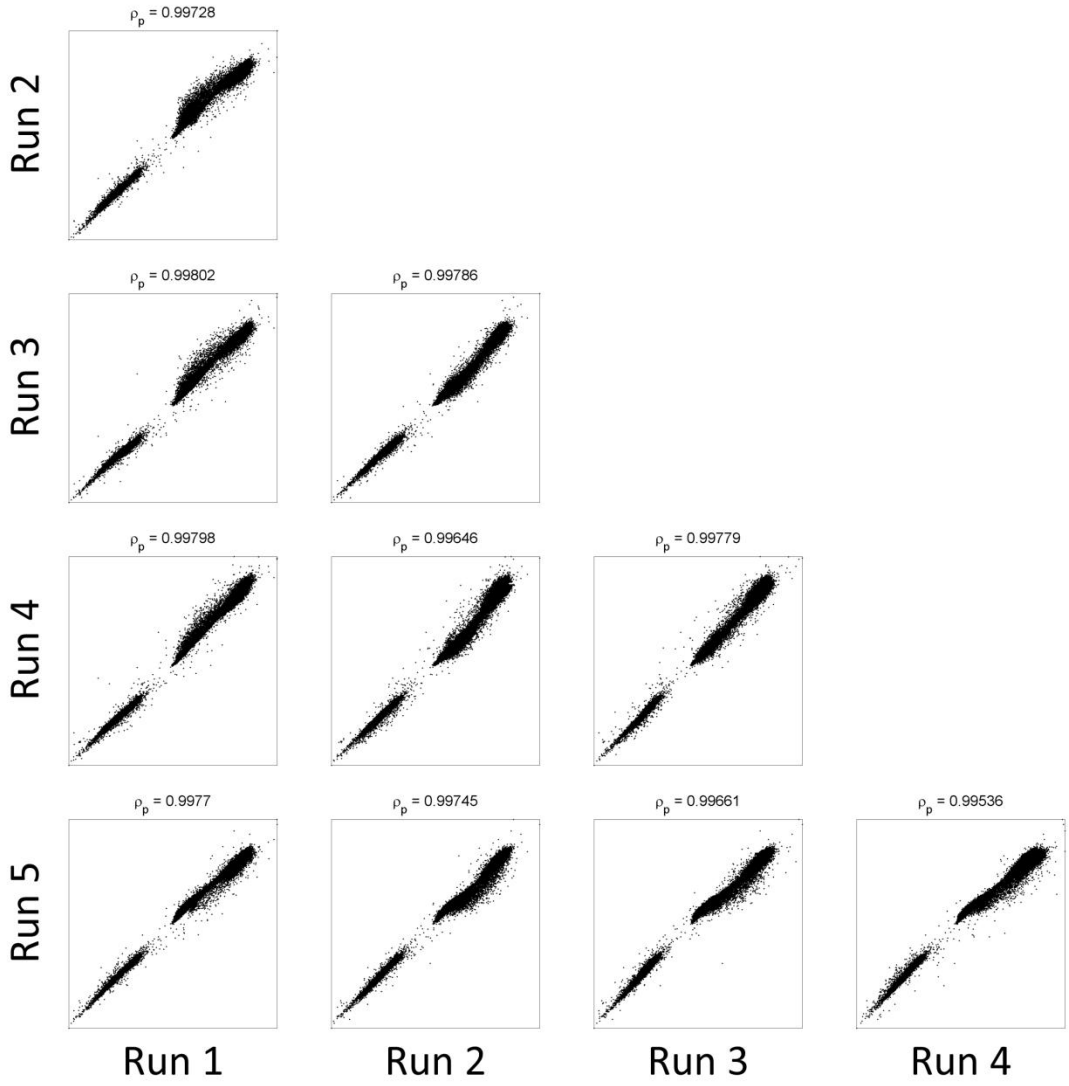


Figure M5. Wanderlust outputs a consistent trajectory over multiple runs and different samples.
Repeated Wanderlust runs for the same sample, using different random seeds. The X-axes and the Y-axes are the algorithm's output for the respective runs. Pearson's correlation is given in each plot's title. The correlation never drops below 0.99.

Next, we verified that the Wanderlust trajectory is consistent over multiple individuals, thus representing human B-cell development, rather than that of a specific individual. We applied the algorithm to data from four healthy bone marrow samples that were acquired from different healthy individuals. However, a direct comparison between the trajectories is misleading. If a sample includes many cells of a given subtype, these cells will occupy a larger

region of the Wanderlust output than other, less-represented subtypes. When examining multiple bone marrow samples, the proportions of different cell subtypes vary according to many factors (such as genetics, exposure to pathogens, and others). The altered proportions will lead to scaling discrepancies between the output trajectories, where less-populated areas shrink and higher-populated areas expand (figure M6a).

To synchronize the trajectory across individuals we calculated the cross-correlation between each sample and an arbitrarily chosen base sample. The cross-correlation was calculated as the mean of all marker cross-correlations. Then, the trajectory score of each cell in each sample was shifted by the value that maximized the cross-correlation (figure M6b).

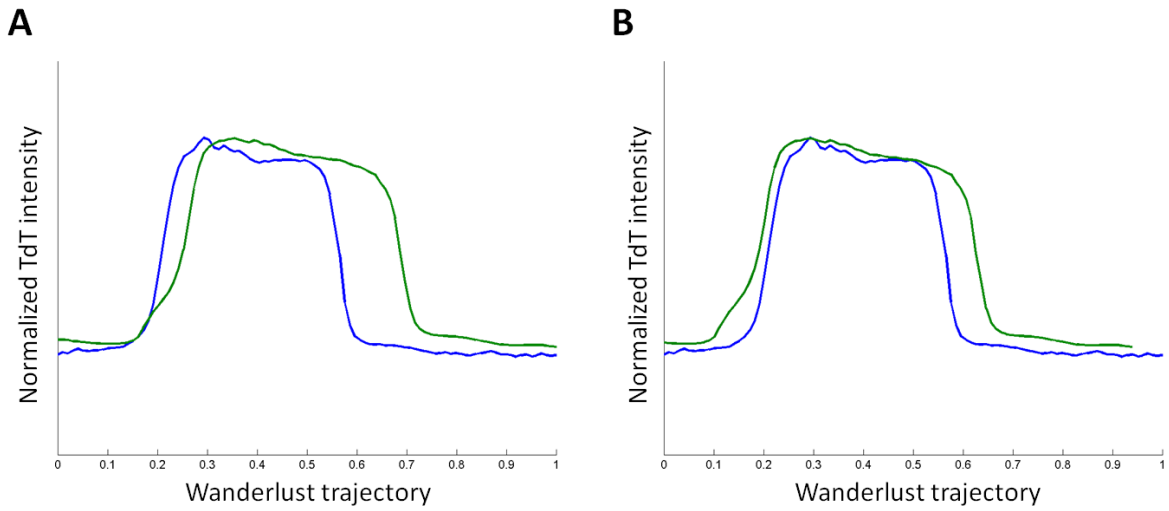


Figure M6. Cross-correlation allows comparison of trajectories between samples. Wanderlust has been applied to two healthy bone marrow samples, in blue and green, respectively. (a) The TdT trace over the trajectory in each sample, before cross-correlation shift. The green sample has more TdT^+ cells, leading to a wider TdT^+ region in the trajectory. Pearson's ρ=0.8. (b) After shifting by the maximal cross-correlation between the two samples, the TdT^+ section has a higher overlap. Pearson's ρ=0.89.

We examined the cross-correlation corrected traces of several markers across the four trajectories (figure M7). Even after shifting by the aggregate cross-correlation, the mean Pearson's correlation between the marker traces over the four samples remained high ($\rho > 0.9$ for all markers). The traces for TdT, which was not used in the Wanderlust analysis, were

especially correlated (ρ = 0.94). Apart from that, no one marker seemed to be better coordinated across the samples. Likewise, no members of sample pairs seemed to be better correlated with each other: for example, while the IgM and TdT traces for samples b and c neatly overlap, their CD24 and IgD traces are dissimilar. Combining figures M5 and M7, we see that the Wanderlust trajectory remains consistent within and between healthy bone-marrow samples.

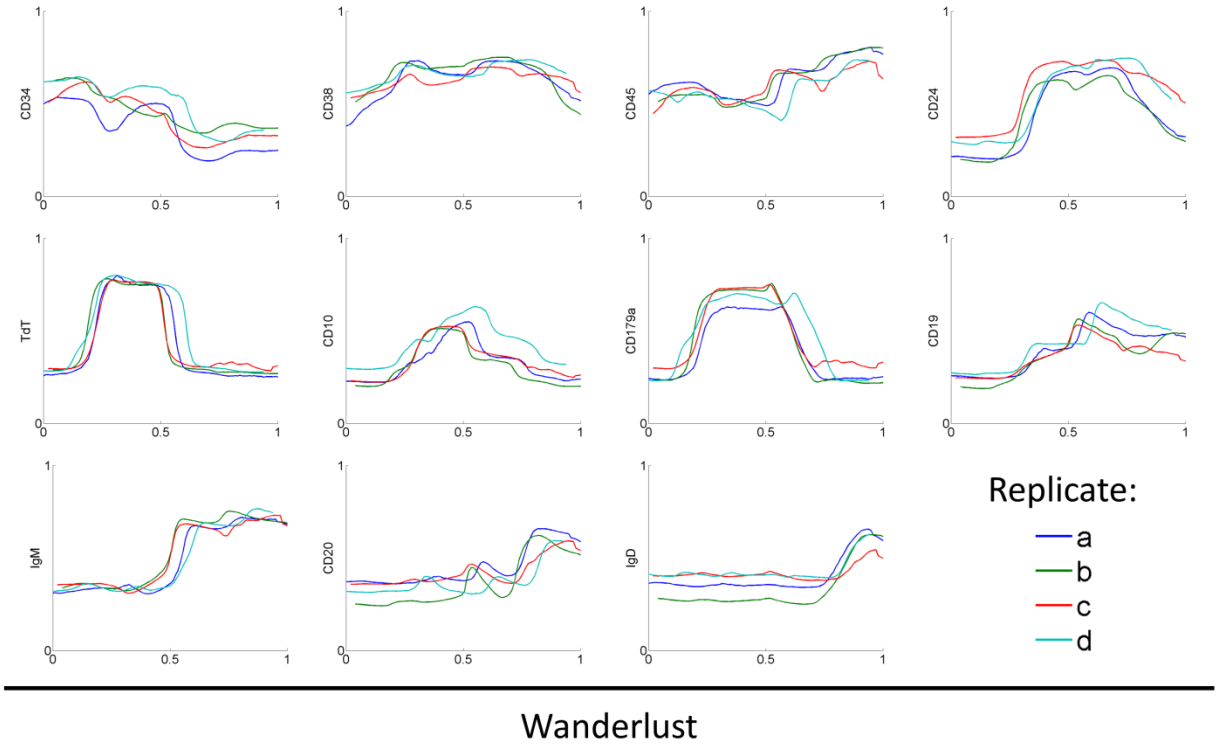


Figure M7. Wanderlust outputs a consistent trajectory over different samples.
Marker traces across the Wanderlust trajectory for 4 different healthy human bone marrow samples (denoted a to d). The trajectories were aligned using cross-correlation. The traces are almost identical between the samples (Pearson's ρ never drops below 0.9).).

Wanderlust is robust over a wide range of parameter choices

Wanderlust requires the user to supply several parameters along with the input data. Key among these is the initiator cell that serves as the start of the initial orientation trajectory. However, locating an absolute starting point for B-cell development or other biological processes is challenging. First, the definition of the initiator cell might be inaccurate or

unavailable; for example, only a subset of the stem-cell markers might be known. Noise in the data might result in a later cell having higher values in the relevant markers, obscuring the true starting cell. Alternatively, the measured panel might not even include the markers needed due to technical reasons. When applying Wanderlust to the healthy B-cell data, we supplied the algorithm with a $CD34^{+}Lin^{-}$ stem cell. This is an approximate initiator cell and the data might have other cells preceding it in the developmental chronology.

We used the output from the Wanderlust run described in figure 2 as a baseline trajectory for testing the influence of the initiator-cell parameter on the algorithm's output. We re-ran Wanderlust ten times. Each time, the initiator cell (s) was shifted by 0.1 across the baseline trajectory and the output was compared to our initial run (figure M8). As long as s remained within the first third of the baseline trajectory the two trajectories overlapped (Pearson's ρ=0.99, 0.99 and 0.98 for s=0.1, 0.2 and 0.3, respectively). When s is set to the midsection of the baseline trajectory (s=0.4, 0.5, 0.6 or 0.7), we see that Wanderlust breaks the output trajectory in half: the algorithm models the second half of the trajectory well, then backtracks and models the second half in reverse. The latest cells are connected to the earliest cells, with the middle becoming a starting point. Finally, when s is a later cell (s=0.8, 0.9, 1.0), Wanderlust detects the reverse trajectory, starting from the latest cells and going back to the earliest cells (Pearson's ρ=-0.98, -0.96 and -0.95, respectively). The algorithm has a meaningful output for a wide range of early cell choices, and only an approximate early cell is needed to detect the trajectory. Moreover, Wanderlust can recover an accurate reverse trajectory when starting from a late cell and going towards an early cell.

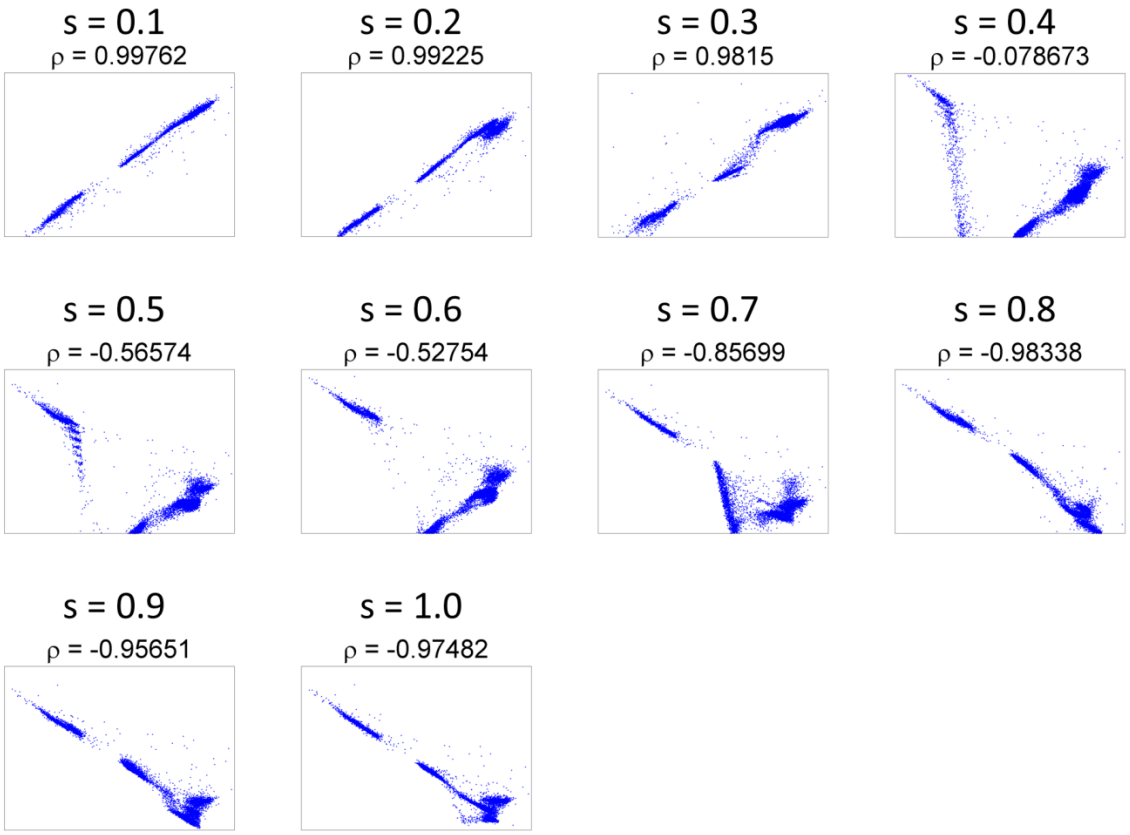


Figure M8. Wanderlust is robust to early-cell parameter choice.
The Wanderlust algorithm has been rerun ten times with the initiator-cell parameter advancing across the baseline trajectory from figure 2. X-axes are baseline trajectory, Y-axes are Wanderlust's output for given s. Each dot is a cell along the trajectory. Pearson's correlation and "starting point" location is given in title. The Wanderlust trajectory is well correlated with the baseline for s=0.1, 0.2 and 0.3, and is inversely correlated (the algorithm detects the reverse trajectory) for s=0.8, 0.9 and 1.0. For other values of s, Wanderlust broke the trajectory in half and modeled each half correctly.

The generation of the l-k-NNG ensemble involves two parameters: k, the number of nearest neighbors in the initial k-NNG, and l, the neighbor subset size for each node in each graph. These parameters have several ramifications on the algorithm's performance. If k is too low, the k-NNG might be disconnected. Likewise, if l is too low, too many edges will be removed and the l-k-NNG will not be connected. In both cases some of the cells will be unreachable by the graph walk and will not be ordered at all by the algorithm. On the other hand, a high k will

increase the number of short-circuits in the k-NNG. Similarly, if l is too close to k, more short circuits will overlap across the l-k-NNG ensemble. From a complexity perspective, increasing l reduces the scarcity factor of the graph, leading to slower run times. In light of the above, k/l choice is crucial for accurate results.

All prior Wanderlust runs used the values k=30 and l=5 (each cell had 30 neighbors in the initial k-NNG, 5 of which were chosen in each l-k-NNG in the ensemble). We tested all combinations of k and l values over a set of values (k=20, 30, 50, 100, l=5, 10, 20, 30). The Wanderlust trajectory generated in figure 2 was again used as the comparison baseline. The correlation between the figure 2 trajectory and each k/l combination trajectory was high (figure M9, Pearson's ρ greater than 0.99), showing that the algorithm is generally consistent over choices of these two parameters. However, when examining the scatter plot for k=20, l=20 (figure M9, bottom left), we see the effect of a short circuit: a cloud of cells diverges from the baseline toward the end of the trajectory, showing that that region is not modeled well. Since l was identical to k, there is no l-k-NNG ensemble and the algorithm is susceptible to this short circuit. As long as l is lower than k, the l-k-NNG ensemble is used, leading to accurate trajectories.

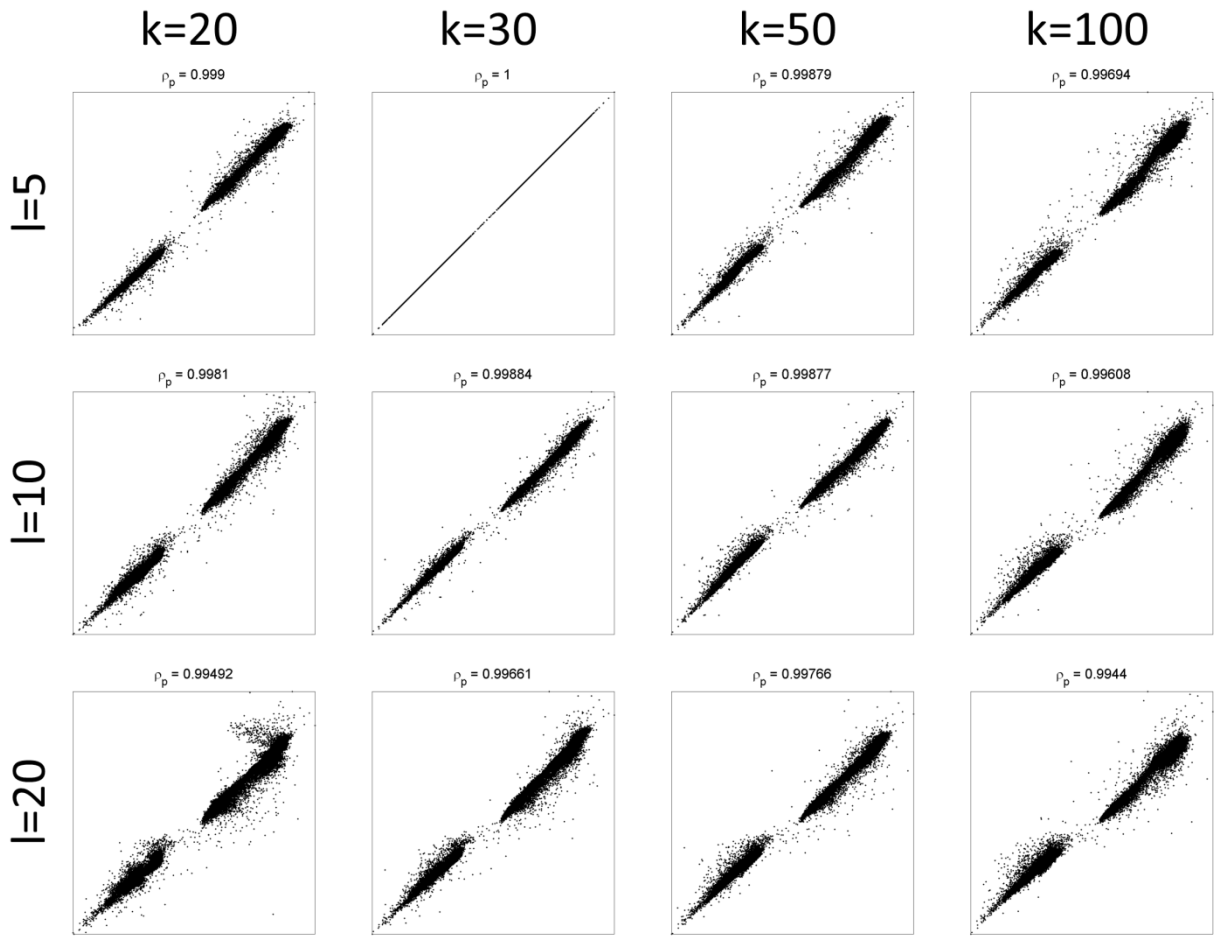


Figure M9. Wanderlust is robust to k/l parameter choice.
The Wanderlust algorithm has been rerun twelve times with different k/l parameter values. The figure 2 trajectory used k=30 and l=5 as parameters (top row, second from left). X-axes are the baseline trajectory, Y-axes are the output trajectory for the k/l parameter choices denoted above and to the left. Pearson's correlation values are given in the titles. The correlation is greater than 0.99 for all parameter combinations. A short-circuit can be seen when k=20, l=20 (bottom left plot).

The last set of parameters is Ng, the number of graphs in the l-k-NNG ensemble, and Nl, the number of waypoints. The number of graphs is again tied to Wanderlust's resistance to short circuits: a certain minimum number of graphs are needed or a short circuit might randomly appear in enough of them to skew the output trajectory. The number of waypoints is related to the algorithm's ability to reduce the variability caused by using the shortest-path distance. Since short distances are more reliable than long distances, enough waypoints are needed to guarantee that each cell has a waypoint nearby. For purposes of optimizing accuracy, there is

no downside to increasing Ng and Nl, as there is no threshold above which these parameters will diminish the quality of the output trajectory. For purposes of optimizing runtime, however, there can be a downside to increasing Ng and Nl, because both are factors in Wanderlust's complexity.

To evaluate robustness, we tested multiple values for Ng and Nl (figure M10). The rest of the parameters were identical to the run in figure 2, which was used as the baseline, and in which Ng=20 and Nl=20. When Ng=1, Wanderlust performs poorly irrespectively of the number of waypoints utilized (figure M10, left). The algorithm follows the set of short circuits randomly chosen in that l-k-NNG, leading to a distorted view of the second half of the trajectory. The layered structure of the scatter plot allows us to follow the number of short circuits in each graph (for example, for Ng=1, Nl=5, there are seven short circuits in the graph). The trajectory improves when Ng=10, although the variability of its second half is very high when Nl=5. This trend of high noise between later cells continues as long as Nl=5, irrespective of Ng. Finally, when using at least ten graphs and at least twenty waypoints, Wanderlust outputs a consistent, high-quality trajectory, marking these values as the threshold for robust trajectory detection in the healthy B-cell dataset.

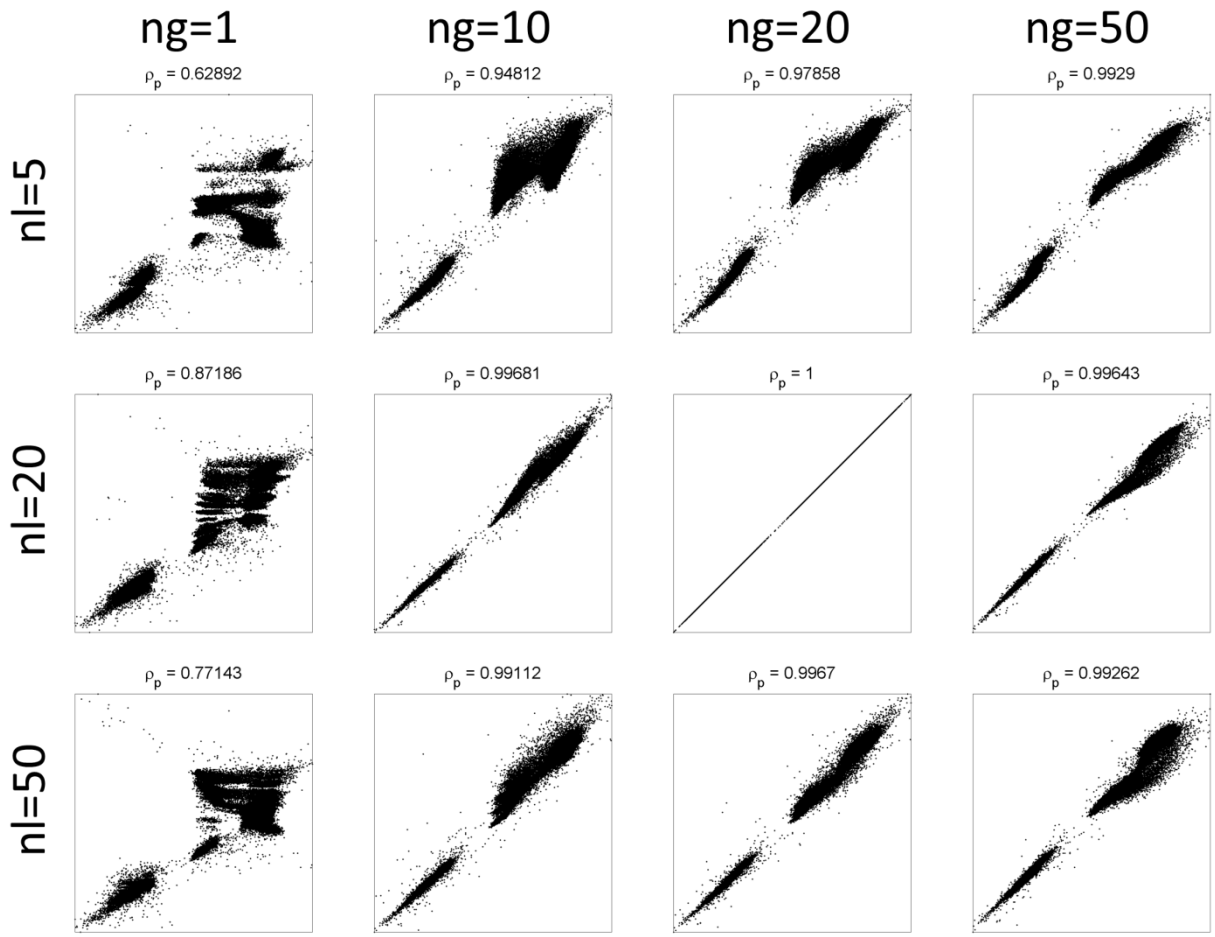
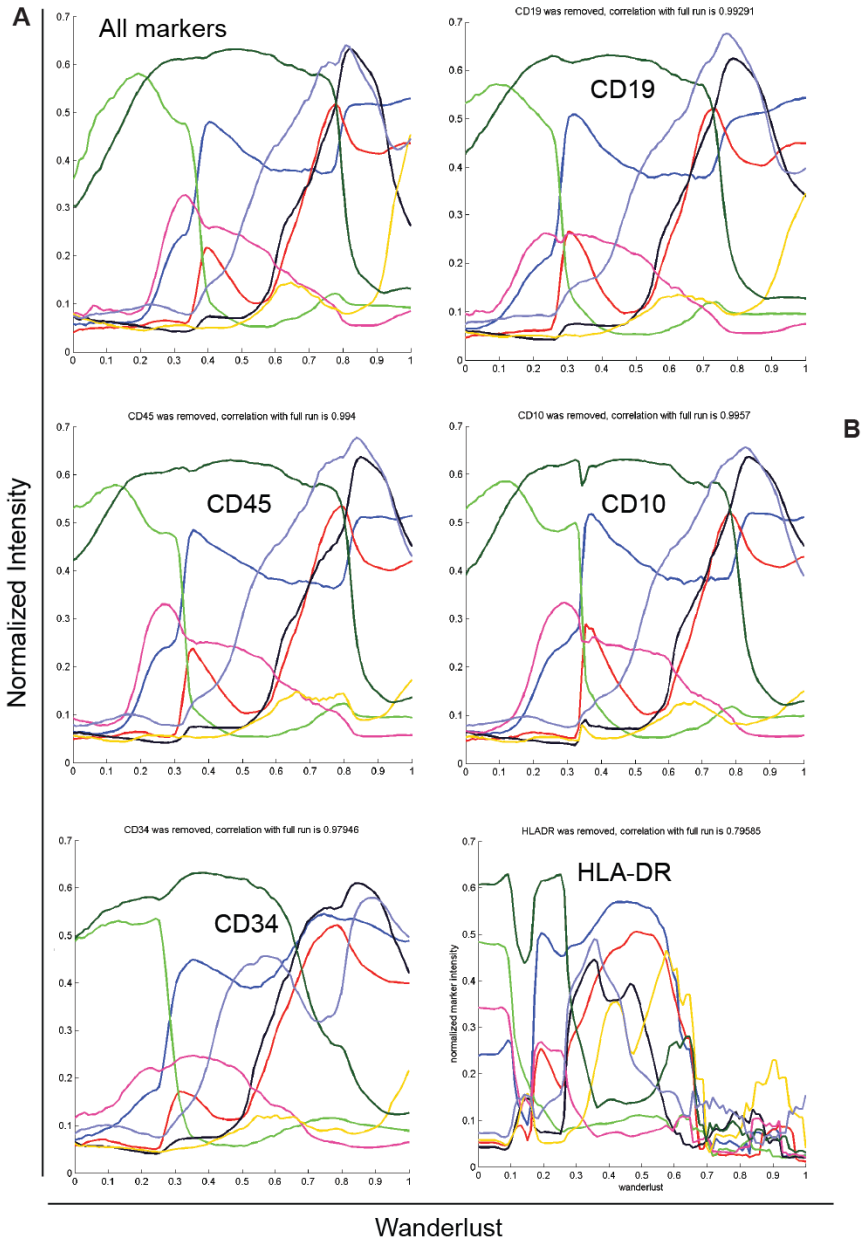


Figure M10. Wanderlust is robust to Ng/Nl parameter values past a certain threshold. Wanderlust has been rerun a dozen times with different Ng/Nl parameter values. The comparison baseline is the trajectory from figure 2 (where Ng=20, Nl=20). X-axes are the baseline trajectory and Y-axes are the Wanderlust trajectory for that Ng/Nl parameter combination. Pearson's correlation values are given in the title. The algorithm fails the model the trajectory when Ng=1. The second half of the trajectory has high variation when Nl=5. In all other cases Wanderlust outputs a faithful representation of the trajectory.

Wanderlust is robust to marker selection

Marker selection is a central part of experiment design. Canonical markers are considered crucial in the identification of certain cell stages. Additionally, while mass cytometry offers a substantial increase in panel size, we cannot comprehensively include all of the surface markers that are expressed by the variety of immune system cell types, or even by just B-lineage cells. Since Wanderlust is only given a subset of possible markers, this raises the possibility that a missing linchpin marker will skew the algorithm's result.

To test the robustness of the trajectory to our selection of phenotypic features we independently ran Wanderlust multiple times. Each time, we removed one marker and calculated both marker traces and the correlation between the Wanderlust output and the original trajectory (Figure M11). Exclusion of any one individual maker had little effect on the results of the overall trajectory as evidenced by the strong correlation with the original model (Pearson's $\rho > 0.97$). Qualitatively, marker traces are identical between the different runs and faithfully follow the continuum of B-cell development. The only exception is HLA-DR, a component of all antigen-presenting cells, which had the greatest influence on the algorithm output with its exclusion dropping the correlation to 0.796 (Figure M11aB, red box). Notably, HLA-DR did not appear to affect the relative ordering of B-cell stages, only their position across the Wanderlust trajectory, suggesting it has a role in partitioning unrelated cell-types. Overall, for the representative Wanderlust trajectory discussed here, no single cellular marker served as a linchpin in the analysis.



B

Phenotypic Marker	Correlation to Original
CD10	0.996
CD19	0.993
CD20	0.994
CD22	0.994
CD24	0.986
CD34	0.979
CD38	0.936
CD40	0.986
CD45	0.994
CD49d	0.994
CD72	0.995
CD79b	0.994
CD117	0.991
CD127	0.995
CD179a	0.988
CD179b	0.993
HLA-DR	0.796
IgD	0.987
IgMi	0.990
IgMs	0.992
kappa	0.987
lambda	0.978

Figure M11a. Wanderlust does not rely on any individual marker.
(a) Resulting Wanderlust traces from leave one out testing in which Wanderlust was independently rerun by individually omitting the from the construction of the trajectory each of CD19, CD45, CD10, CD34 and HLA-DR. The resulting order of events remained consistent. (b) Correlation to original trajectory (Figure 2) when indicated marker is omitted from the construction of the trajectory. Omission of HLA-DR (red box) in the construction of the trajectory has the largest effect on the overall trajectory (correlation 0.796).

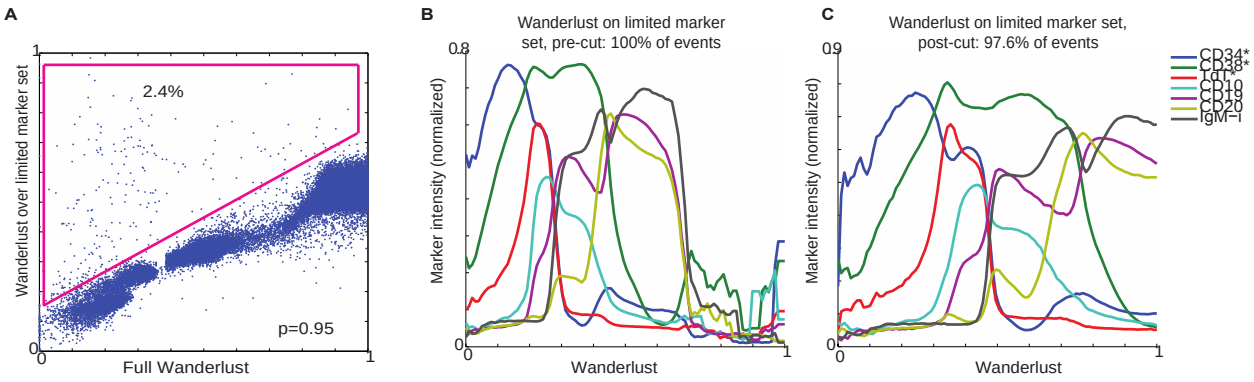


Figure M11b. Wanderlust is robust to loss of 6 canonical B cell markers. A) Scatterplot representing correlation between original trajectory (X axis) and trajectory constructed using only 11 markers, when 6 classical B cell markers were omitted (CD10, CD19, CD20, CD79b, IgMi, IgMs). Data was uniformly downsampled to 50,000 cells to make plot clearer. Correlation remains high (ρ=0.95). Only 2.4% of the total cells reside in the pink box representing uncorrelated off-diagonal cells. **B**) Wanderlust traces after omission of 6 key markers demonstrates that general trends are maintained when compared to the original trajectory (Figure2A) however, trajectory becomes noisy toward the end. Such noisy "trendless" behavior of the trace is a telltale sign for poor reconstruction of that segment. These cells at the end are exactly those represented in the pink box in panel B. **C**) By omitting these cells (Figure S2B, pink box), the resulting Wanderlust trace was now closer to the original trace in Figure 2A and correctly place the remaining 97.6% of cells. Note that only 3 markers included in this trace were used to build the trajectory (denoted by "*"). Taken together, Wanderlust is able to capture the progression of the system even without the inclusion of definitional markers.

We further tested the robustness of the trajectory to the selection of phenotypic features by running Wanderlust without the inclusion of six canonical B cell markers (CD10, CD19, CD20, CD79b, IgMi, and IgMs. The correlation to the original trajectory remained high (ρ=0.95) (Figure M11bA). The Wanderlust trace for this "leave 6 out" test captured major trends but towards the end of the trajectory (>0.6) the marker levels are noisy (M11bB). Such noisy "trendless" behavior of the trace is a telltale sign for poor reconstruction of that segment. These cells at the end are exactly those represented in the pink box in panel A, representing only 2.4% of cells distant from the regression line. By omitting these cells (Figure M11bA, pink box), the resulting Wanderlust trace was now closer to the original trace and correctly place the remaining 97.6% of cells (Figure M11bC). Note that only 3 markers included in this trace were used to

build the trajectory. Taken together, Wanderlust is able to capture the progression of the system even without the inclusion of definitional markers.

Comparison of Wanderlust to other methods

Wanderlust is unique in its ability to order single cells along a one-dimensional trajectory that can track a non-linear developmental chronology. Hence, it is not obvious where Wanderlust should be placed in relation to existing algorithm taxonomy and it might be considered as a sole member of the new algorithm family of trajectory detection. Nevertheless, there are a number of methods that can inform us of the developmental signal, which is dominantly strong in our data. We apply a number of existing methods to our data and indirectly compare these to Wanderlust. We note that these methods were all designed to organize the data so as to visualize key trends in the data, the developmental progression being one such trend, but none of these methods were explicitly designed to infer the developmental trajectory and hence it is not surprising that these do not perform as well as Wanderlust.

Principle Component Analysis (PCA)

PCA is a linear transformation of the data into the principal eigenvectors (called components) of the covariance matrix of the data. PCA provides a linear mapping of the data, the basic assumption being that the relationship between variables in the data is linear. However, we have clearly shown that the developmental progression in a non-linear (Figure 1A).

We previously applied PCA to B-cell data collected with mass cytometry and found that the first principle component resembled the developmental trajectory as in Bendall et al., *Science*, 2011 (Bendall et al 2011), though in that application, we inferred a short segment of the trajectory, from pre-B cells to immature B-cells. Here we test what happens when we apply PCA to the entire spectrum, from hematopoietic stem cell to immature B-cell. While PCA maps

individual cells in a very robust manner, the accuracy and resolution of the chronology is lost to a large degree due to the linear mapping (Figure M12). For example, CD10 and TdT peak at the very beginning of the trajectory, suggestion recombination begins in stem cells. Additionally, the resolution of the rare and early cell subsets which vary in expression of these markers is lost. Therefore, while the developmental trend is clearly a strong signal in the data, as indicated by its dominance in the first principle component, a non-linear approach is needed to capture the correct chronology.

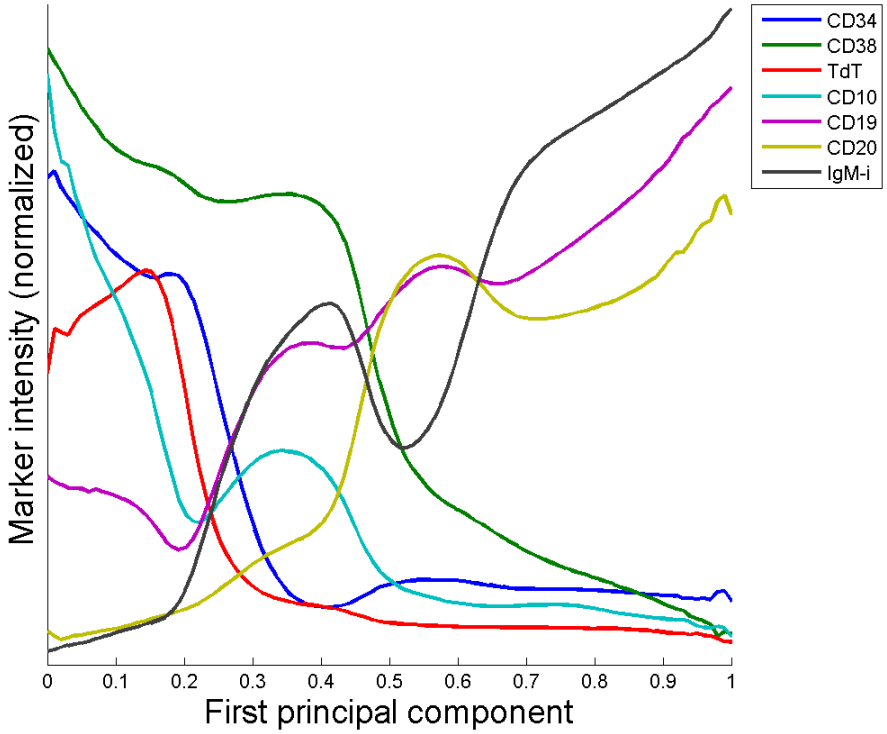


Figure M12. Applying PCA to the B-cell single-cell data.
PCA was applied to the same B-cell data from Figure 2. Marker traces were calculated using the same method as in the main text. X-axis, value of PCA (normalized to 0-1 range). Y-axis, represents marker intensities. The same markers from figure 2 are plotted.

<u>viSNE, non-linear dimensionality reduction</u>

viSNE is a non-linear dimensionality reduction algorithm (based on tSNE), that transforms high-dimensional data into a two-dimensional map, making it accessible to visual exploration as in Amir et al., *Nature Biotechnology*, 2013 (Amir et.al.2013). It was designed with two- or three- dimensional mapping in mind, and as such is less appropriate for a one-dimensional mapping. Indeed, using tSNE to reduce to one dimension produces a trajectory with fluctuating marker means and a very high degree of variance that has no relationship to known trends in development.

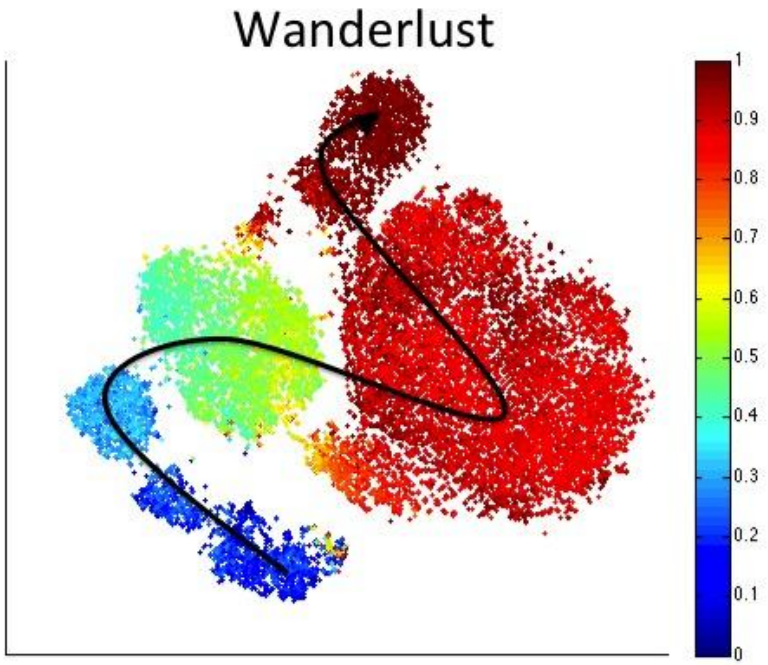


Figure M13a. Applying viSNE to the B-cell single-cell data.
viSNE was applied to the same B-cell data from Figure 2. Each individual cell is colored on a gradient based on their order in the Wanderlust trajectory. The black line roughly tracks the trajectory.

Given that the developmental trend is the strongest signal in the data, the two-dimensional viSNE map leads to an informative visual that roughly recapitulates the trajectory (figure M13a). Nevertheless, this visual does not define an explicit ordering on the cells that can

be used to generate a marker trace or derivative analysis. This figure further highlights the non-linear nature of the developmental progression, even though the viSNE mapping is a non-linear projection in itself, the trend of the Wanderlust trajectory proceeds through this mapping in a curved, non-linear fashion (black line in figure M13). Had we not known the correct ordering of the trajectory, it would have been hard to infer it from this visual. Moreover, the temporal resolution is missing from the viSNE map resulting in a loss of the ability to infer the correct order of events or identify coordination points (figure M13b).

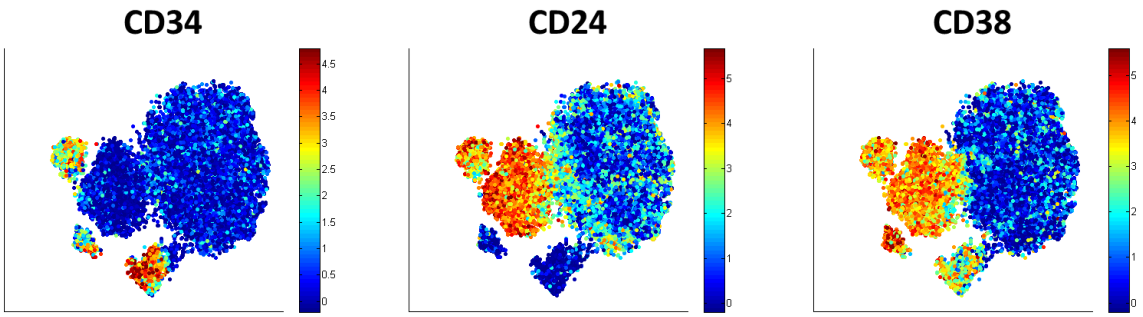


Figure M13b. Applying viSNE to the B-cell single-cell data.
Same mapping as in M13a, here each cell is colored based on the marker level as indicated in the image title. The resolution of marker behavior along the trajectory is coarse and not as accurate as achieved in Wanderlust.

SPADE, a Minimum Spanning Tree (MST) based approach

SPADE is a clustering-based method, based on minimum spanning tree (MST), for extraction of cellular hierarchy from single-cell data(Qiu et al., 2011). Similarly to viSNE, the SPADE map roughly visualizes some of the developmental trends in the data, in a piecewise manner (Figure M14A). However, SPADE is not a robust mapping and as a result multiple runs on the same data performed with identical parameters lead to different outputs that define different orderings, which do not necessarily globally match the correct chronology (Figure M14A). For example, in the middle panel the most mature cells come in between the

hematopoietic stem cell and the progenitors. Without prior knowledge, there is no way to judge which output leads to the correct interpretation of the system. Additionally SPADE clusters the data, losing the single-cell resolution.

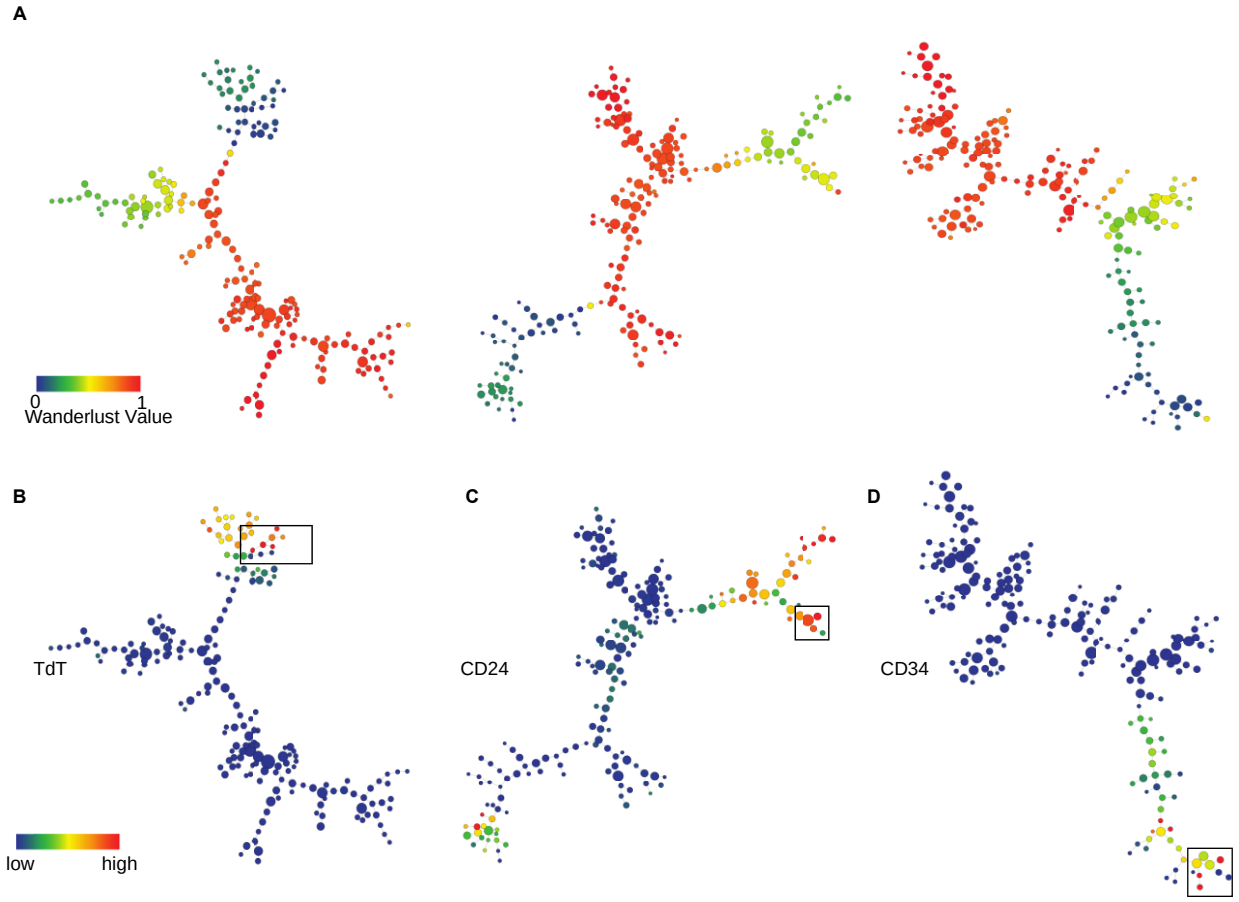


Figure M14. SPADE output for B-cell data.
(A) Three SPADE runs are presented, each with a different minimum spanning tree (MST). Nodes are colored based on the Wanderlust trajectory. While one of these runs has a structure that loosely corresponds to the trajectory, the other two do not capture the proper progression. (B) MSTs (same as those in A) nodes are colored based on mean marker intensity of the indicated marker.

Examining individual clusters for local trends, it is notable that these are not ordered according to the correct chronology. The problem being that the MST algorithm, underlying SPADE, is not explicitly designed to infer a trajectory and but rather suitable to capture rough

trends in the data. While most development is a branching process, there is no explicit drive in the MST algorithm to match it's branching choices with developmental branch points. A case in point is this non-branching B-cell progression, where the resulting MST branches are not robust across repeated runs and are not biologically supported. Moreover, the marker levels across the backbone of the tree do not follow clear trends but rather fluctuate along the backbone resulting in clusters both high and low for a given marker occupying the same region of the backbone (Figure M14B, boxes). Therefore, although the MST approach may provide an estimation of the developmental ordering, single cell resolution is lost as is the ability to infer the correct order of events and identify coordination points across the progression.

Supplemental References

Amir, E.-A. D., Davis, K. L., Tadmor, M. D., Simonds, E. F., Levine, J. H., Bendall, S. C., et al. (2013). viSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. *Nature Biotechnology*. doi:10.1038/nbt.2594

Bendall, S. C., Simonds, E. F., Qiu, P., Amir, E.-A. D., Krutzik, P. O., Finck, R., et al. (2011). Single-Cell Mass Cytometry of Differential Immune and Drug Responses Across a Human Hematopoietic Continuum. *Science (New York, NY)*, *332*(6030), 687–696. doi:10.1126/science.1198704

Fienberg, H. G., Simonds, E. F., Fantl, W. J., Nolan, G. P., & Bodenmiller, B. (2012). A platinum-based covalent viability reagent for single-cell mass cytometry. *Cytometry. Part a : the Journal of the International Society for Analytical Cytology*, *81*(6), 467–475. doi:10.1002/cyto.a.22067

Finck, R., Simonds, E. F., Jager, A., Krishnaswamy, S., Sachs, K., Fantl, W., et al. (2013). Normalization of mass cytometry data with bead standards. *Cytometry. Part a : the Journal of the International Society for Analytical Cytology*, *83*(5), 483–494. doi:10.1002/cyto.a.22271

Kotecha, N., Krutzik, P. O., & Irish, J. M. (2010). Web-based analysis and publication of flow cytometry experiments. *Curr Protoc Cytom*.

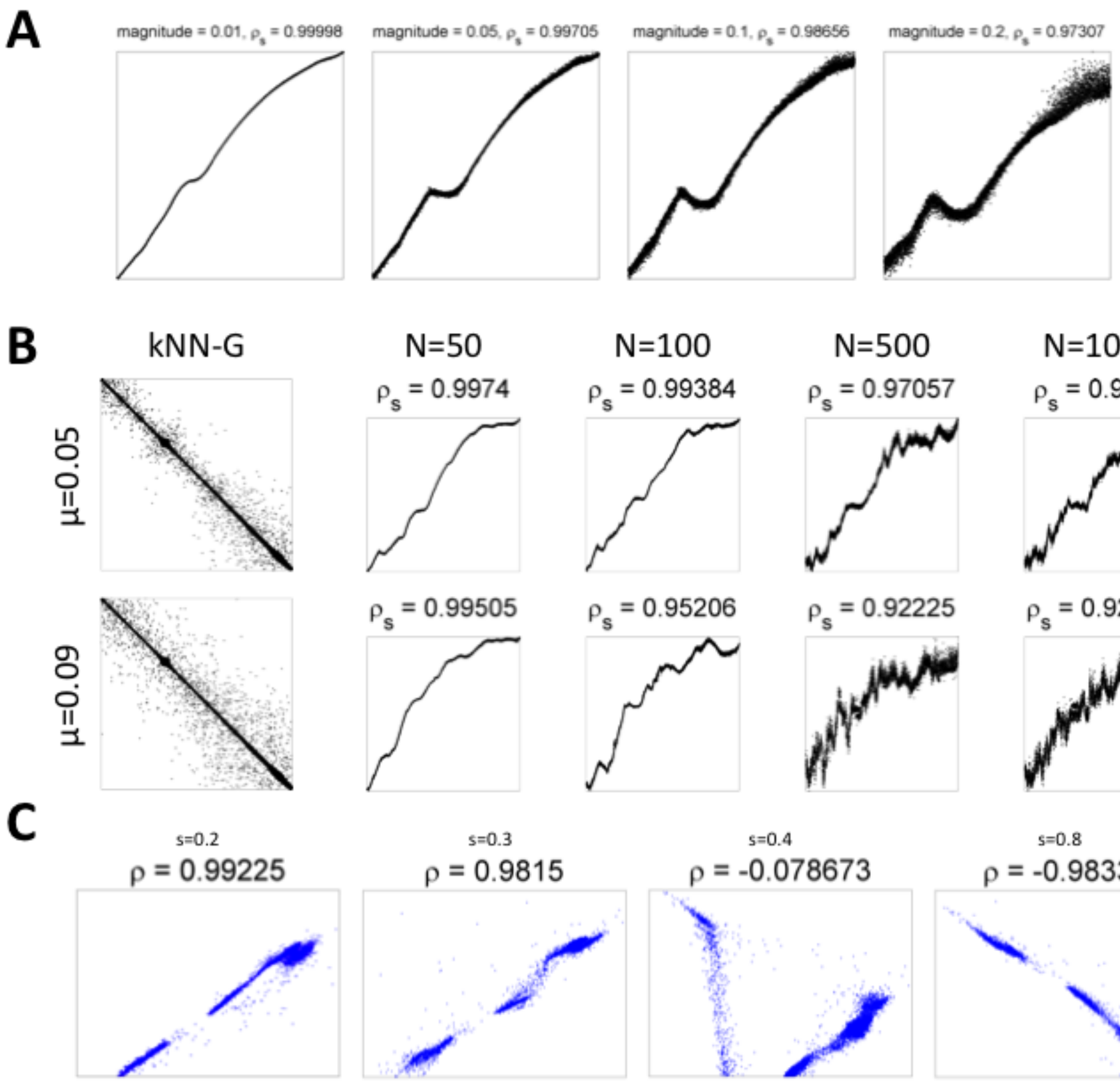
Qiu, P., Simonds, E. F., Bendall, S. C., Gibbs, K. D., Bruggner, R. V., Linderman, M. D., et al. (2011). Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nature Biotechnology*, *29*(10), 886–891. doi:10.1038/nbt.1991

Sanz, E., Munoz-A, N., Monserrat, J., Van-Den-Rym, A., Escoll, P., Ranz, I., et al. (2010). Ordering human CD34+CD10-CD19+ pre/pro-B-cell and CD19- common lymphoid progenitor stages in two pro-B-cell development pathways. *Proceedings of the National Academy of Sciences of the United States of America*, *107*(13), 5925–5930. doi:10.1073/pnas.0907942107

van Dongen, J. J. M., Langerak, A. W., Brüggemann, M., Evans, P. A. S., Hummel, M., Lavender, F. L., et al. (2003). Design and standardization of PCR primers and protocols for detection of clonal immunoglobulin and T-cell receptor gene recombinations in suspect lymphoproliferations: report of the BIOMED-2 Concerted Action BMH4-CT98-3936. *Leukemia*, *17*(12), 2257–2317. doi:10.1038/sj.leu.2403202

Supplemental Figure 1

Nolan & Pe`er Figure S1



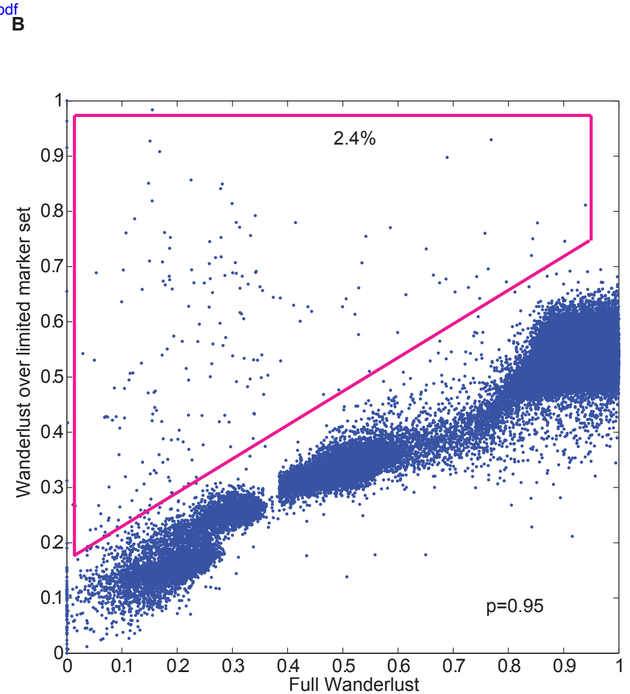
Supplemental Figure 2

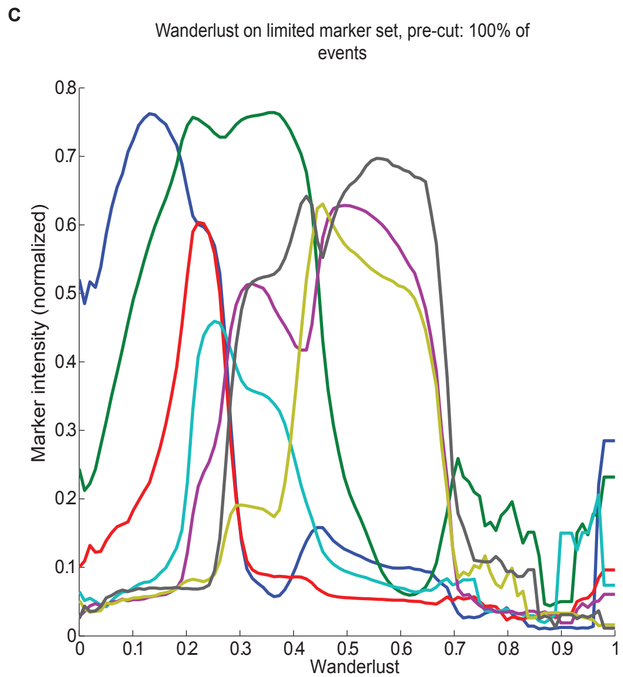
Nolan & Pe'er Figure S2

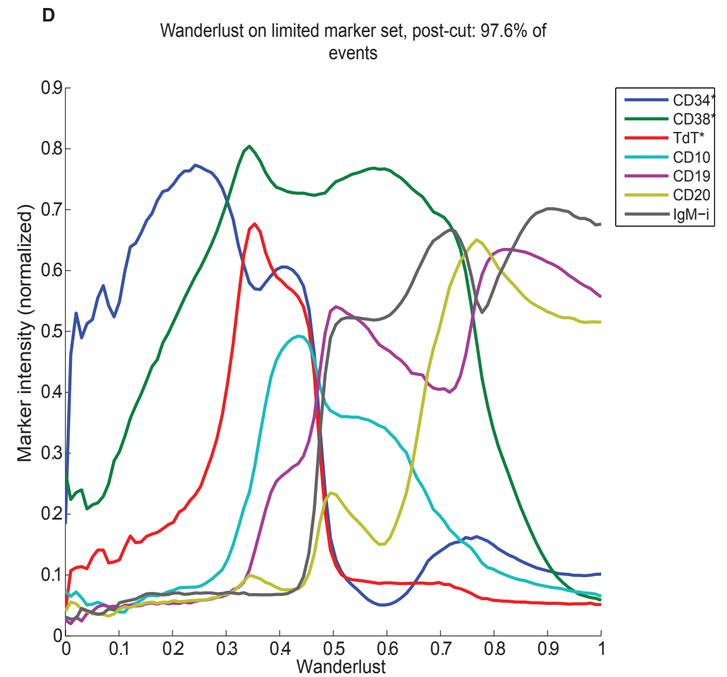
Click here to download Supplemental Figure: MABEE-SupFig2-V6.pdf

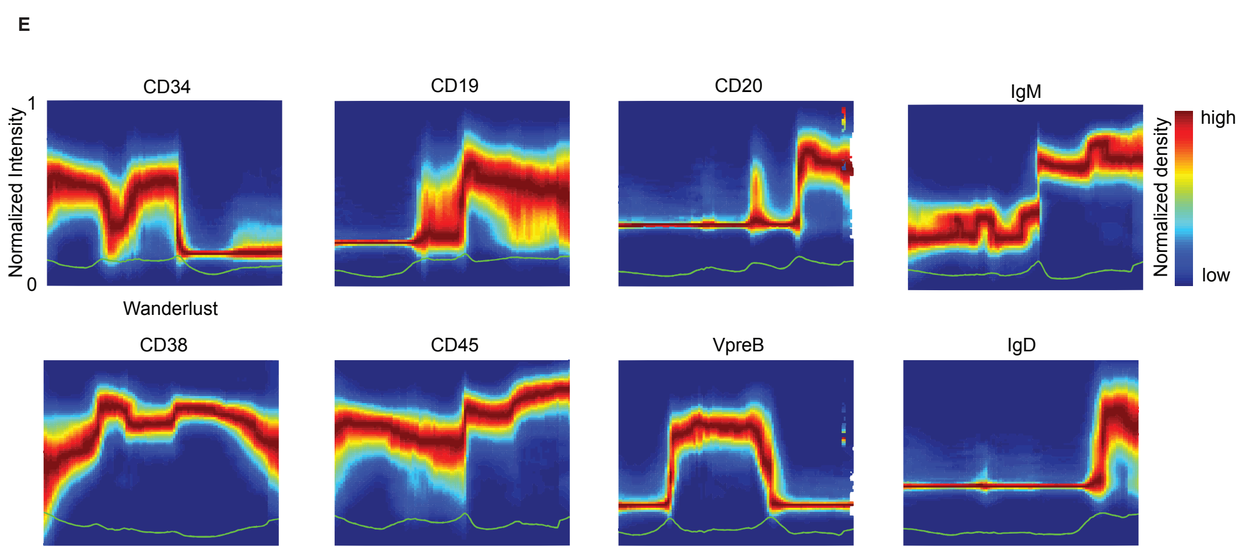
A

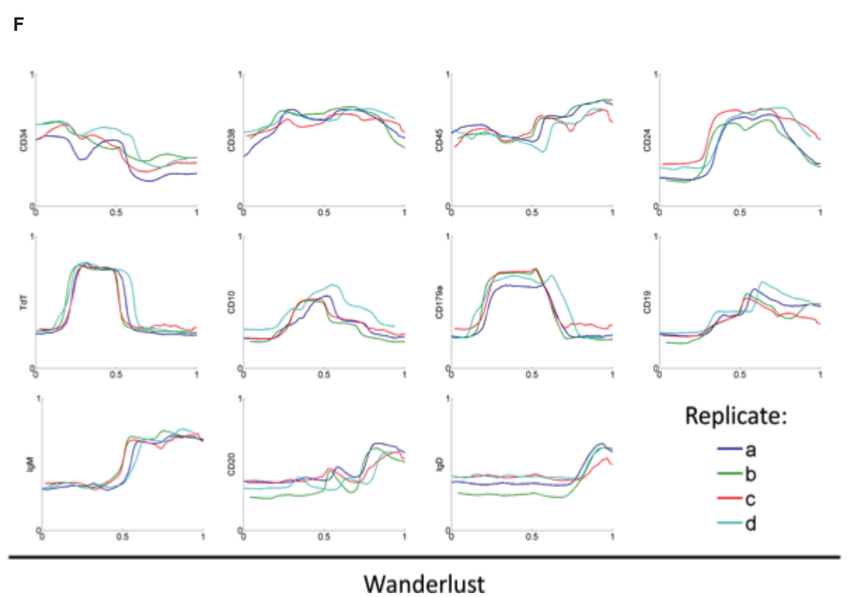
Phenotypic Marker	Correlation to Original
CD10	0.996
CD19	0.993
CD20	0.994
CD22	0.994
CD24	0.986
CD34	0.979
CD38	0.936
CD40	0.986
CD45	0.994
CD49d	0.994
CD72	0.995
CD79b	0.994
CD117	0.991
CD127	0.995
CD179a	0.988
CD179b	0.993
HLA-DR	0.796
IgD	0.987
IgMi	0.990
IgMs	0.992
kappa	0.987
lambda	0.978

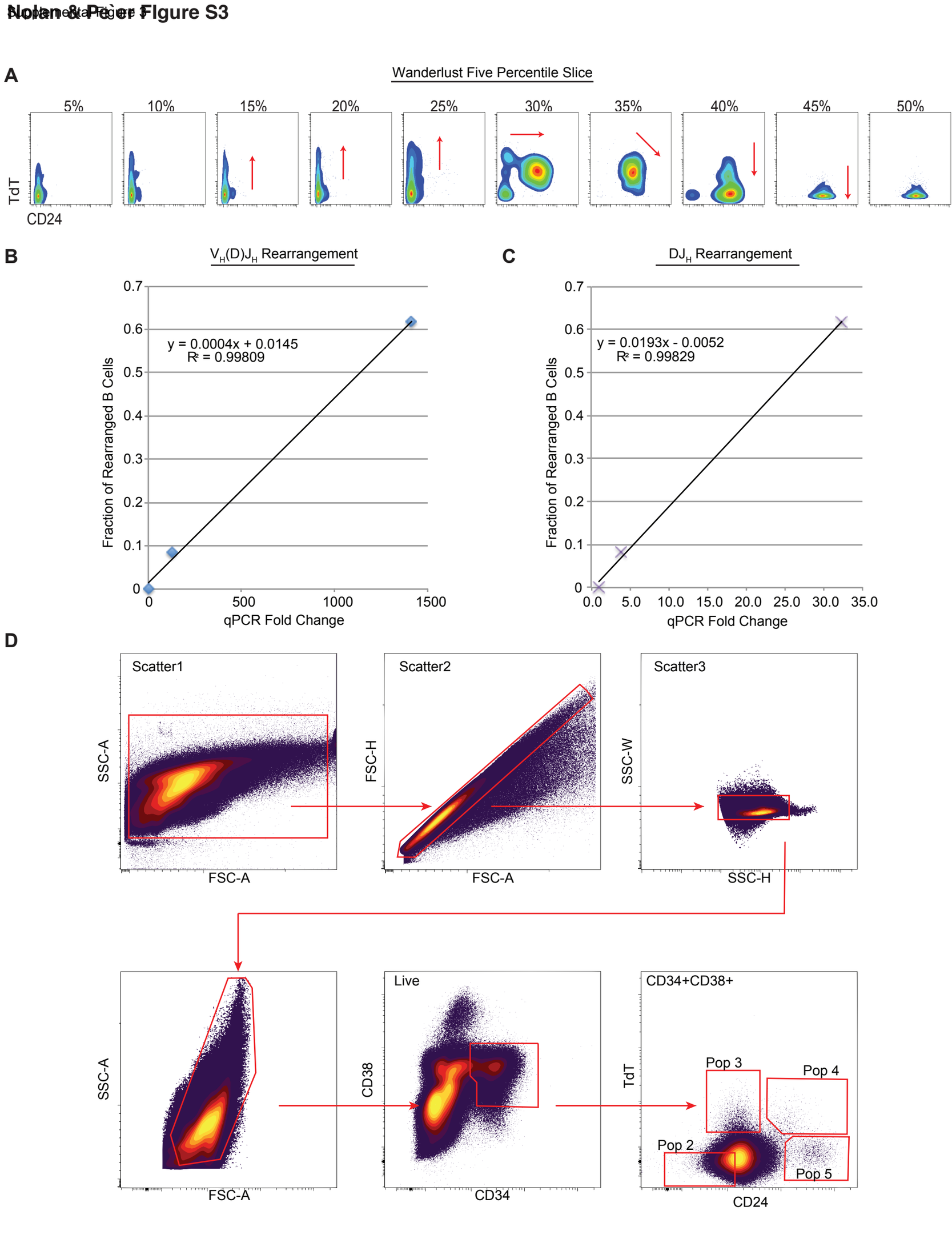


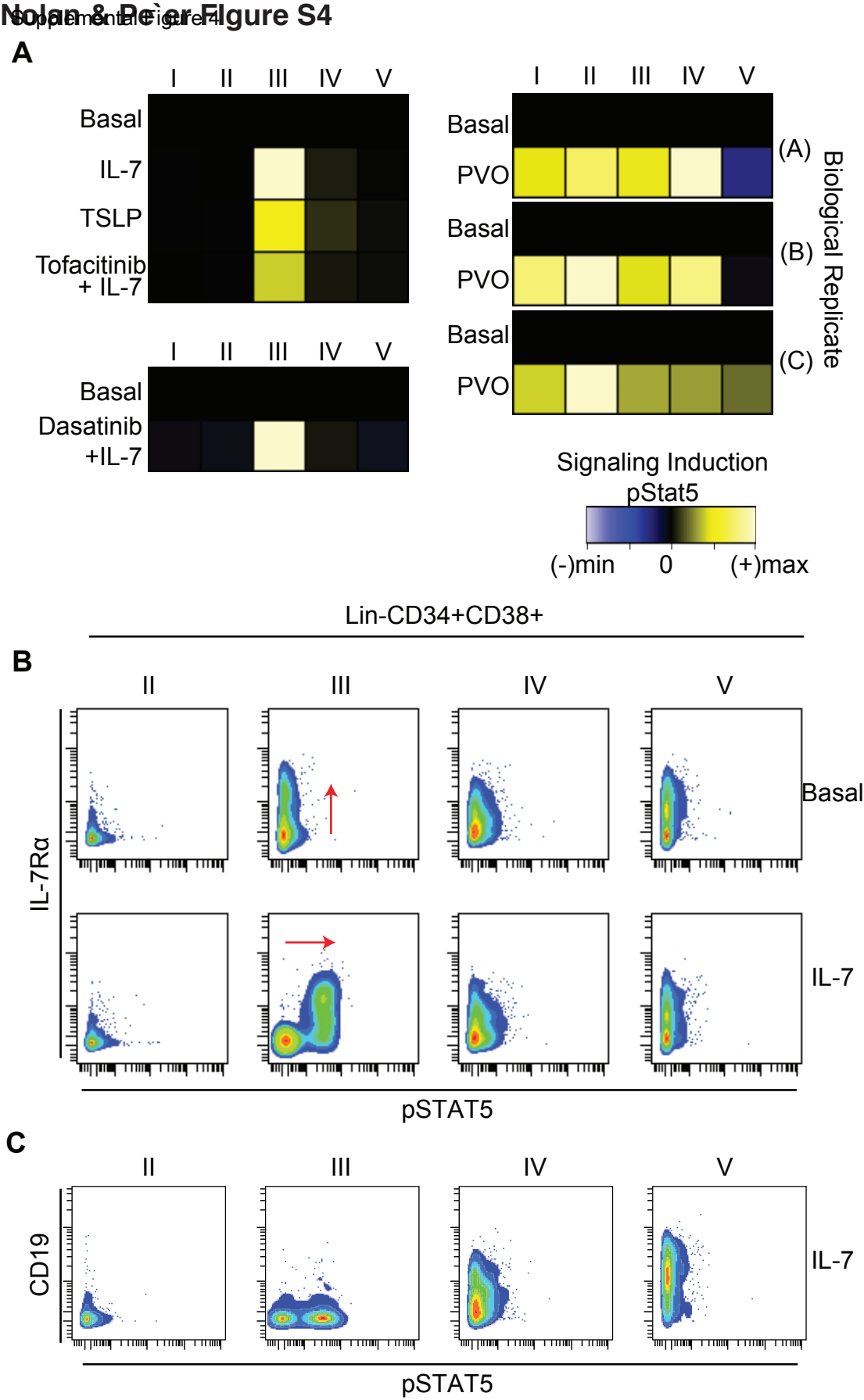









A
Wanderlust Five Percentile Slice
5%
10%
15%
20%
25%
30%
35%
40%
45%
50%
TdT
CD24
B
V_H(D)J_H Rearrangement
y = 0.0004x + 0.0145
R² = 0.99809
Fraction of Rearranged B Cells
qPCR Fold Change
C
DJ_H Rearrangement
y = 0.0193x - 0.0052
R² = 0.99829
Fraction of Rearranged B Cells
qPCR Fold Change
D
Scatter1
SSC-A
FSC-A
Scatter2
FSC-H
FSC-A
Scatter3
SSC-W
SSC-H
SSC-A
FSC-A
Live
CD38
CD34
CD34+CD38+
TdT
CD24
Pop 2
Pop 3
Pop 4
Pop 5


A
I II III IV V
Basal
IL-7
TSLP
Tofacitinib + IL-7
I II III IV V
Basal
Dasatinib +IL-7
I II III IV V
Basal
PVO
(A)
Basal
PVO
(B)
Basal
PVO
(C)
Biological Replicate
Signaling Induction
pStat5
(-)min 0 (+)max
Lin-CD34+CD38+
B
II III IV V
IL-7Rα
Basal
IL-7
pSTAT5
C
II III IV V
CD19
IL-7
pSTAT5

Supplemental Figure 5

Nolan & Pe'er Figure S5

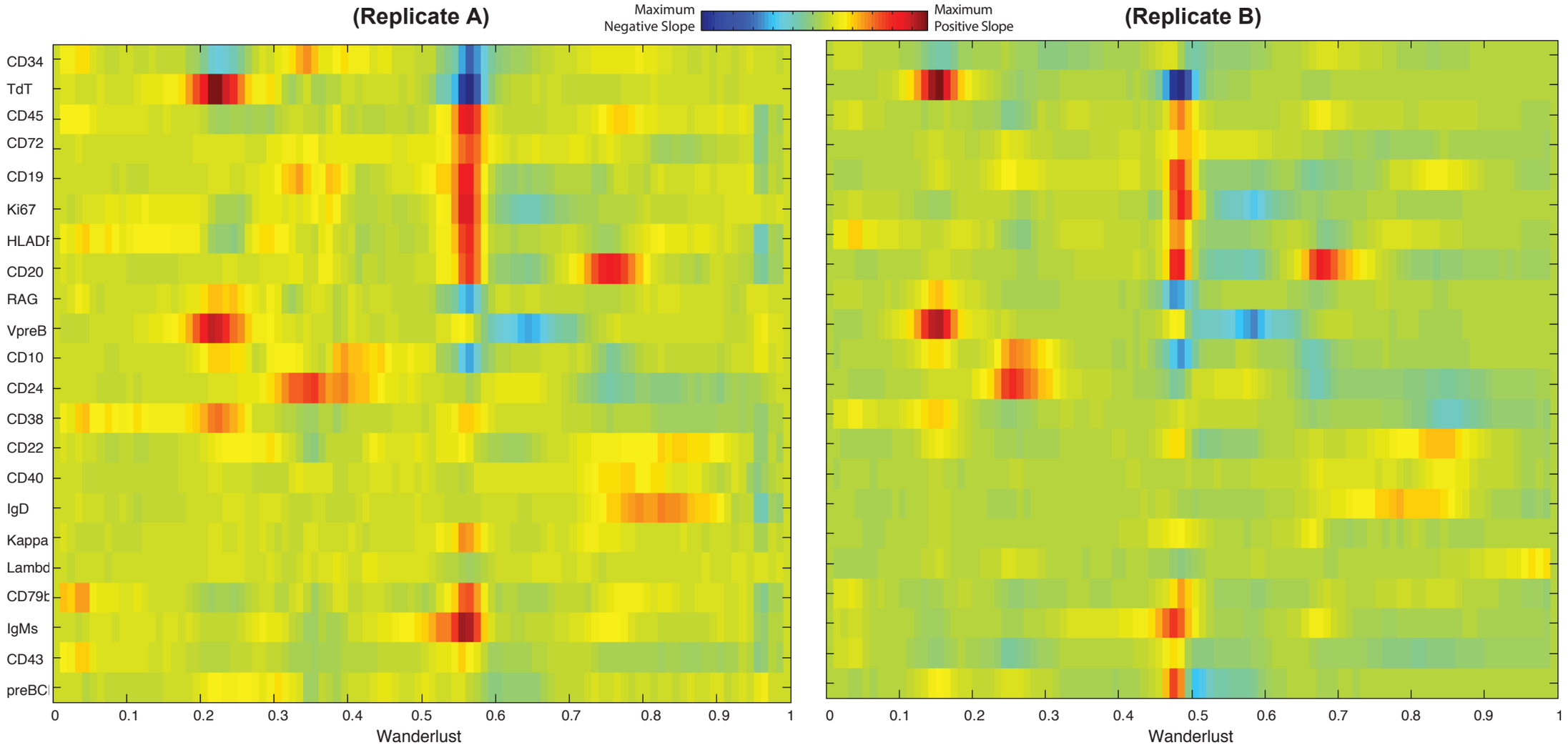
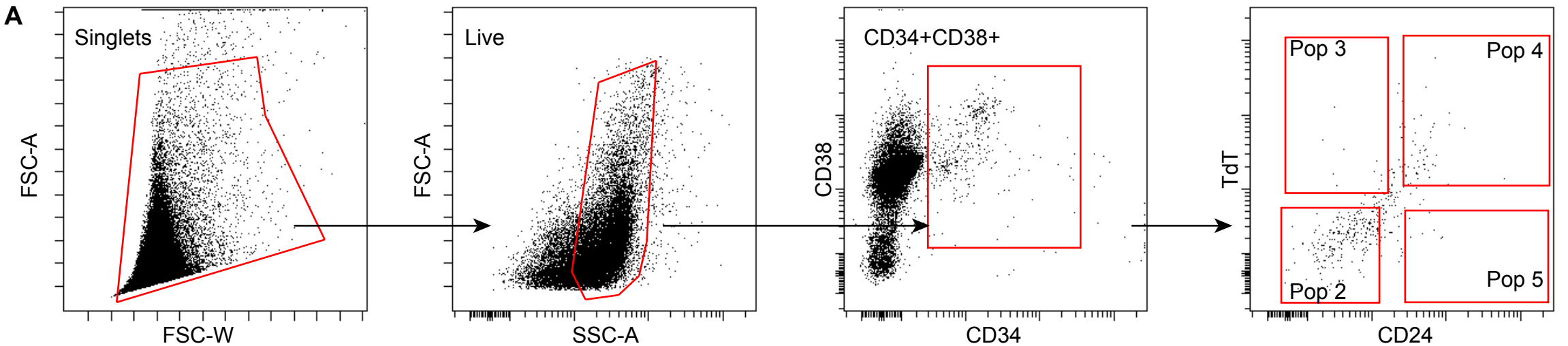


Figure S6


A
Singlets
FSC-A
FSC-W
Live
FSC-A
SSC-A
CD34+CD38+
CD38
CD34
TdT
CD24
Pop 3
Pop 4
Pop 2
Pop 5

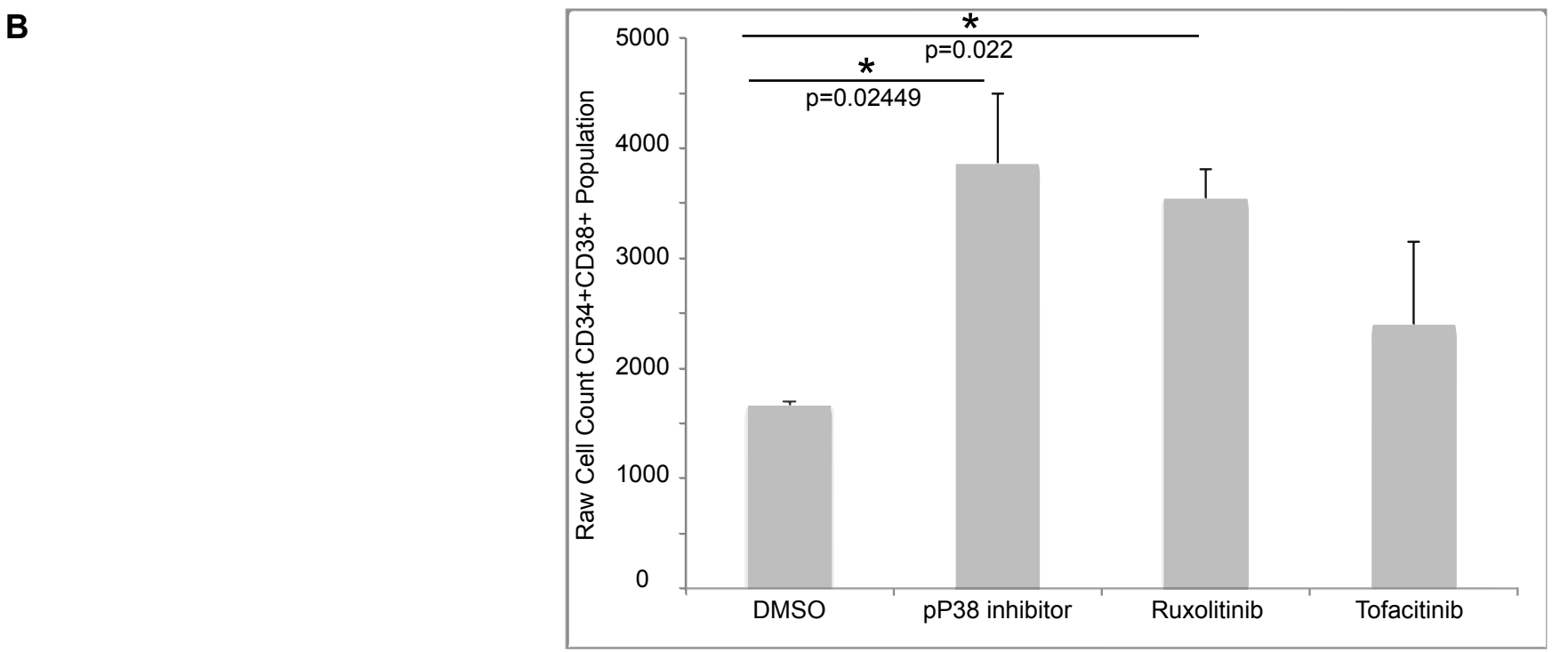
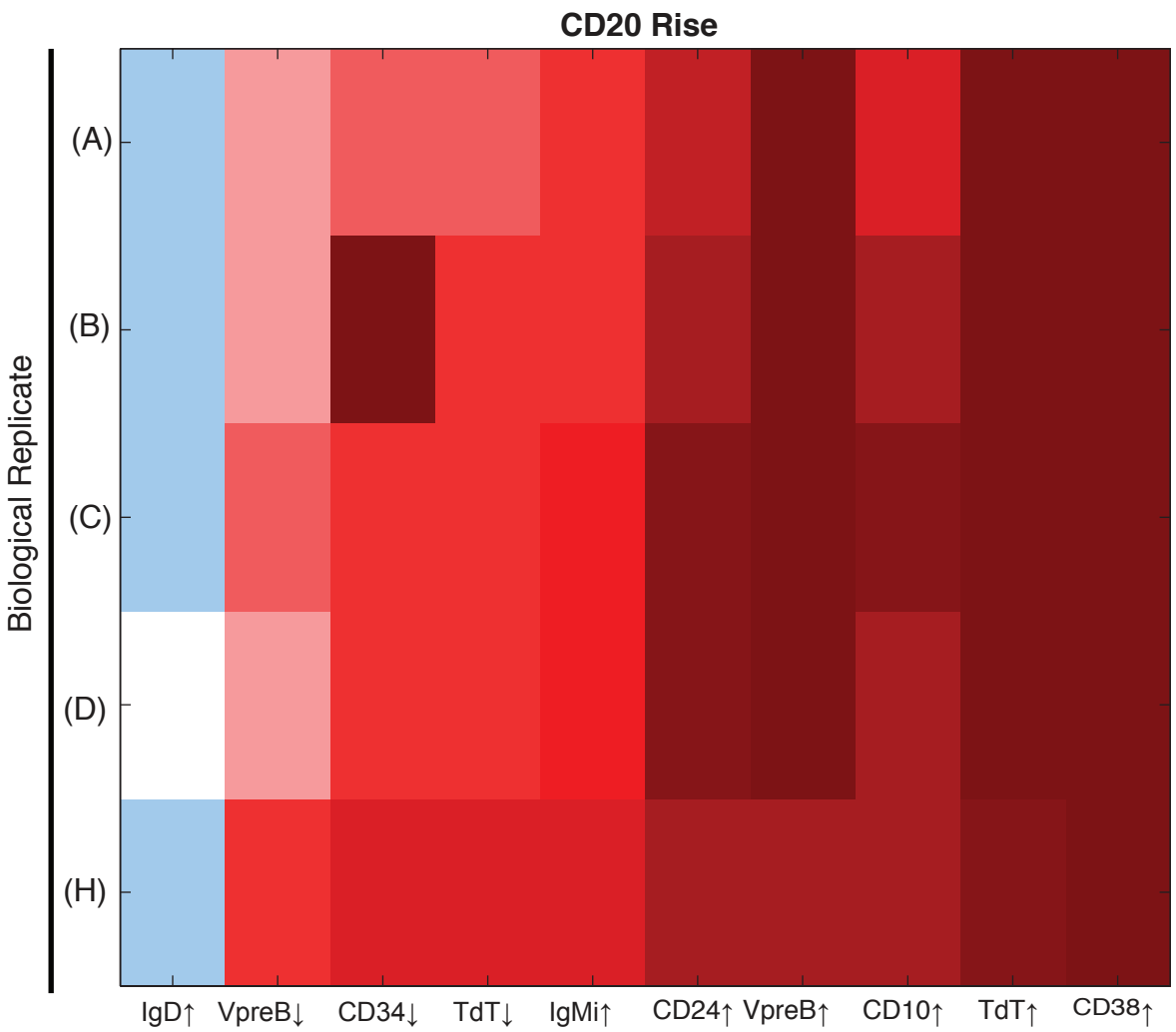
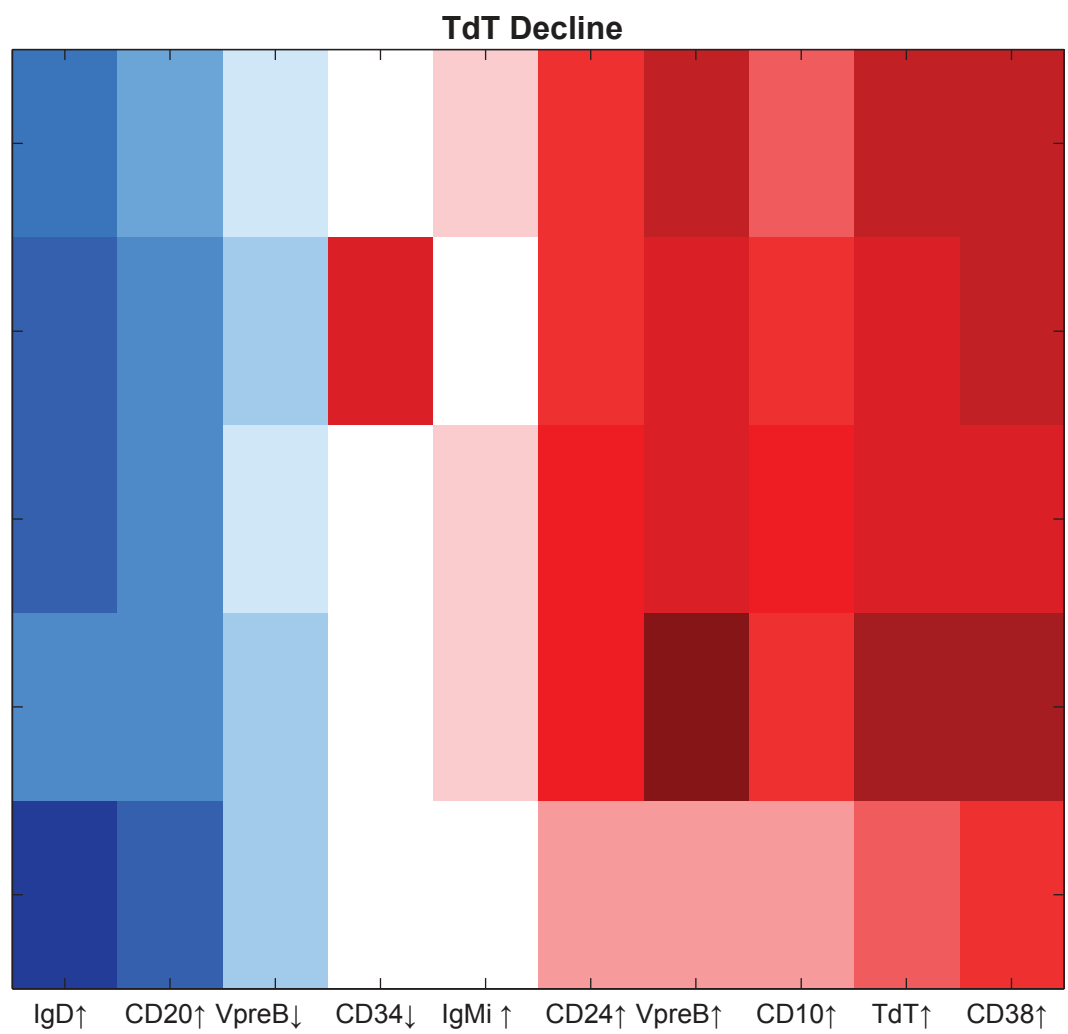
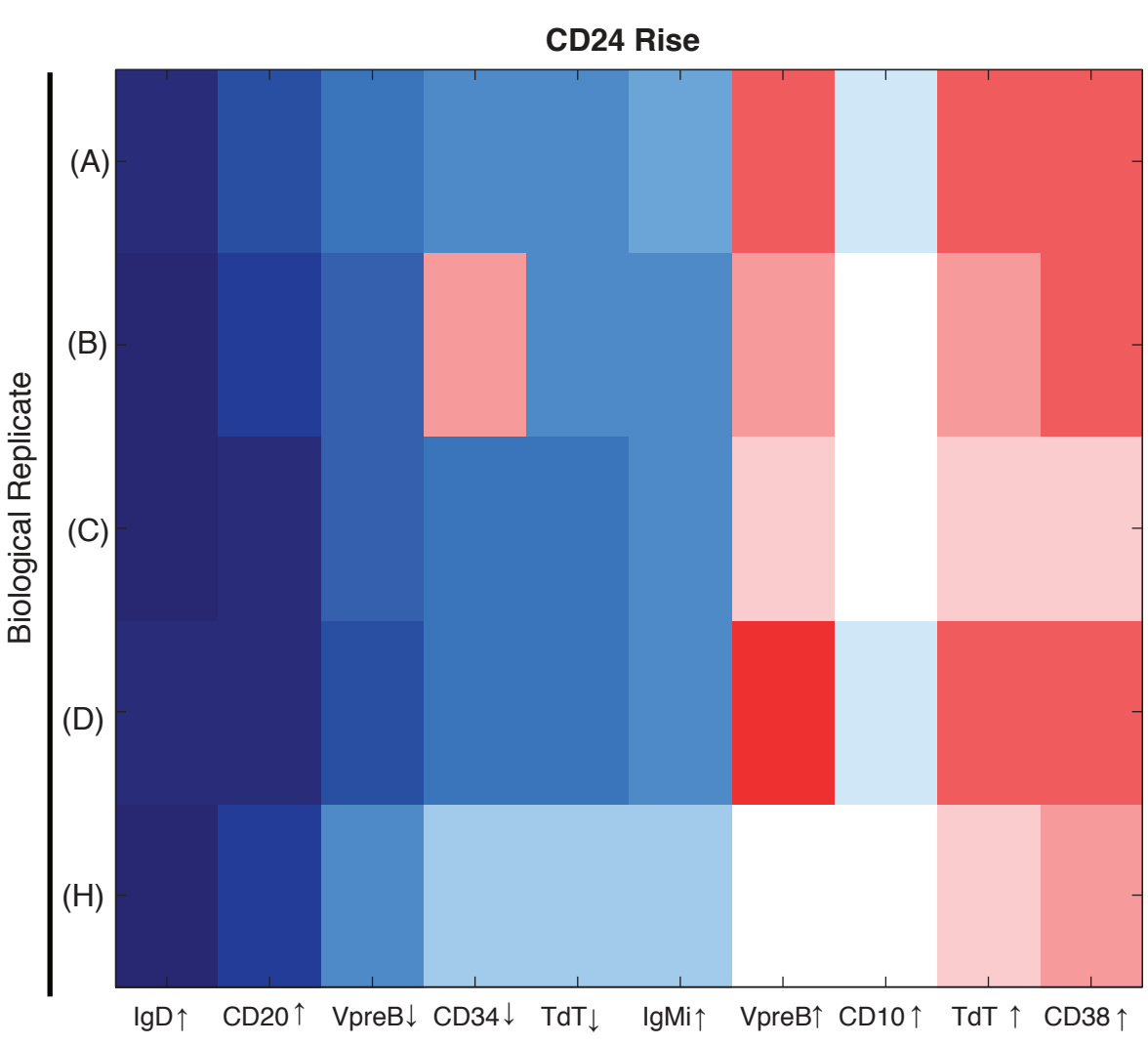
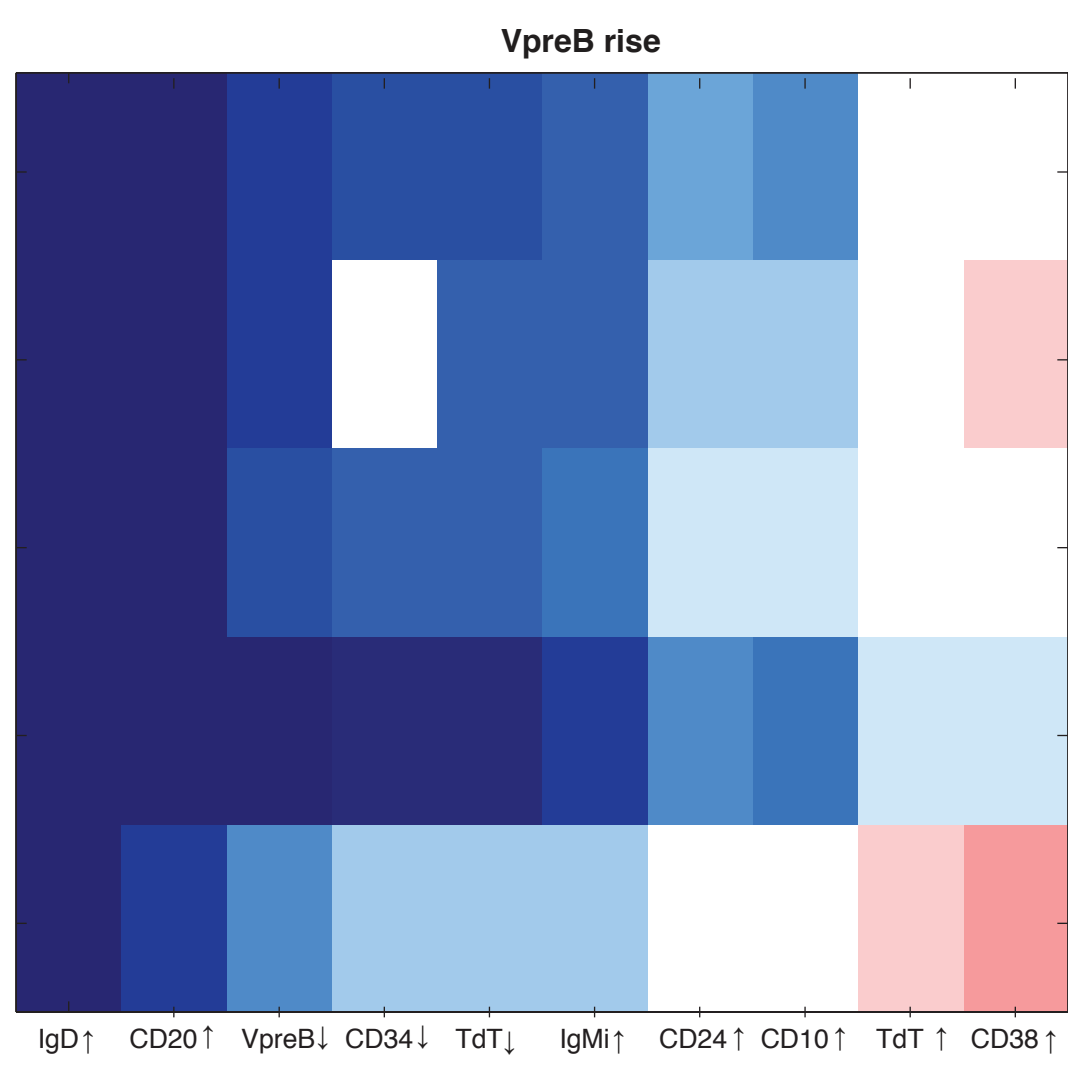

B
Raw Cell Count CD34+CD38+ Population
5000
4000
3000
2000
1000
0
*
p=0.022
*
p=0.02449
DMSO
pP38 inhibitor
Ruxolitinib
Tofacitinib

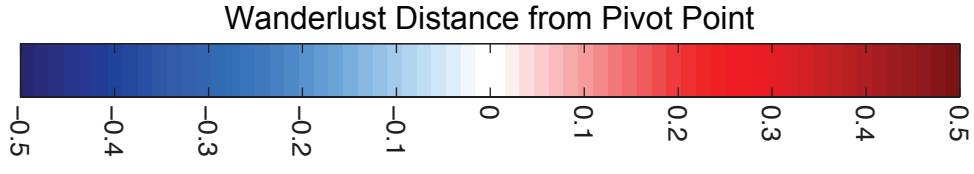
Figure S7


CD20 Rise
Biological Replicate
(A)
(B)
(C)
(D)
(H)
IgD↑ VpreB↓ CD34↓ TdT↓ IgMi↑ CD24↑ VpreB↑ CD10↑ TdT↑ CD38↑


TdT Decline
IgD↑ CD20↑ VpreB↓ CD34↓ IgMi ↑ CD24↑ VpreB↑ CD10↑ TdT↑ CD38↑


CD24 Rise
Biological Replicate
(A)
(B)
(C)
(D)
(H)
IgD↑ CD20↑ VpreB↓ CD34↓ TdT↓ IgMi↑ VpreB↑ CD10↑ TdT ↑ CD38↑


VpreB rise
IgD↑ CD20↑ VpreB↓ CD34↓ TdT↓ IgMi↑ CD24↑ CD10↑ TdT ↑ CD38↑


Wanderlust Distance from Pivot Point
−0.5 −0.4 −0.3 −0.2 −0.1 0 0.1 0.2 0.3 0.4 0.5