

**Comparison of Clustering Methods for
High-Dimensional Single-Cell Flow and Mass
Cytometry Data:
Supporting Information**

Lukas M. Weber^{1,2}, Mark D. Robinson^{1,2}

¹Institute of Molecular Life Sciences, University of Zurich, Zurich, Switzerland

²SIB Swiss Institute of Bioinformatics, University of Zurich, Zurich, Switzerland

November 18, 2016

Contents

Supporting Information Methods	2
Parameters spreadsheet	2
Clustering methods	2
Data sets	2
Data pre-processing	4
Number of clusters	4
Optimal number of clusters for FlowSOM	5
Subsampling and multiple processor cores	6
Ensemble clustering	7
Stability of clustering results	7
Supporting Information Results	8
Results for FlowCAP data sets	8
Supporting Information Figures	9
Main results: all data sets	9
Additional results for individual populations	15
Ensemble clustering	19
Stability of clustering results	24
Optimal number of clusters for FlowSOM	26
Cluster median expression profiles	27
Gating diagrams	42

Supporting Information Methods

Parameters spreadsheet

Supporting Information Table S1 is provided as a separate spreadsheet file (`Supp_Table_S1_clustering_methods_parameters.xlsx`). This spreadsheet includes additional details on all clustering methods, including software package versions, numbers of clusters, all parameter settings used for the final results, and additional explanatory notes.

Clustering methods

Unsuccessful methods

In addition to the 18 clustering methods listed in Table 1 (main paper), we also attempted to include the following methods, but were unsuccessful:

- FLAME: This method no longer appears to be available as an R package. It has instead been included in the “GenePattern” analysis platform, which is available both online and for local download. However, our attempts to run FLAME through GenePattern did not work, possibly due to the large size and high dimensionality of our data sets. Troubleshooting was also difficult due to the custom platform.
- HDPGMM: This method is available as part of the Python library “fcm”. However, we were unable to install this library, due to an error during installation of a dependency package, “dpmix”. We attempted this on both a Linux server and Mac laptop, but the installation was unsuccessful on both machines. Due to the installation problems, we were unable to run the method.
- ASPIRE: We were unable to run ASPIRE due to a requirement for input files to be in a specialized binary format, which we were unfamiliar with. No documentation or examples could be found with instructions on how to generate these input data files. In addition, pre-compiled program files were only available for Windows environments, which caused additional difficulties.

Data sets

Levine_32dim and Levine_13dim

Note that these two data sets are referred to as “benchmark data set 2” (`Levine_32dim`) and “benchmark data set 1” (`Levine_13dim`) in the original publication (Levine et al., 2015).

Samusik_01 and Samusik_all

The original publication for these data sets (Samusik et al., 2016) states that they are 38-dimensional. This was a typing error; they are in fact 39-dimensional, as shown in Table 2 (main paper).

FlowCAP data sets

Supporting Information Table S2 provides additional details on the two FlowCAP data sets included in this study, **FlowCAP_ND** and **FlowCAP_WNV**. In addition to the main reference (Aghaeepour et al., 2013), further details on these data sets are also available from the FlowCAP website (<http://flowcap.flowsite.org/>).

For these data sets, we ran clustering methods individually for each sample, calculated mean F1 scores on a per-sample basis, and subsequently averaged across samples to calculate an overall mean F1 score. This strategy was required because the manually gated population indices differed across samples (i.e. population 1 in one sample may be labeled as population 4 in the next sample); and additional meta-data required to harmonize the sample indices across samples (e.g. cell population descriptions) was not available. For this reason, methods accessible via graphical interfaces were also excluded from the analysis for the FlowCAP data sets; since the mean F1 scores needed to be calculated separately for each sample, a scripting approach was required.

By contrast, for the six main data sets (Table 2 in main paper), we ran clustering algorithms on the combined data from all samples (for the data sets with more than one sample). This approach makes it easier to compare detected clusters between different samples during subsequent downstream analysis.

Supporting Information Table S2. Summary of FlowCAP data sets included in this study.

Data set	FlowCAP-I name	Clustering task	No. of cells	No. of samples	No. of dimensions	No. of manually gated populations	Sample description
FlowCAP_ND	Normal Donors (ND)	Detect multiple populations	Approx. 60,000 per sample	30	10	7	Cells exposed to various stimuli for a set of healthy donors
FlowCAP_WNV	Symptomatic West Nile Virus (WNV)	Detect multiple populations	Approx. 90,000 per sample	13	6	4	Stimulated peripheral blood cells from a set of patients with symptomatic West Nile virus

Data pre-processing

Data sets with multiple FCS files

Data sets that were provided as multiple FCS files (e.g. one file per sample, or one file per manually gated population) were concatenated into a single file. Clustering algorithms were then run once on the combined FCS file (except for the FlowCAP data sets; see above).

No additional normalization between samples was required, since the authors of the data sets containing multiple samples (`Levine_32dim` and `Samusik_all`) already performed sample normalization, and the remaining data sets each consisted of only one sample.

Gating for data sets Nilsson_rare and Mosmann_rare

For the data sets with a single rare cell population of interest (`Nilsson_rare` and `Mosmann_rare`), manually gated population labels were not available with the original data files. However, gating diagrams were included in the original publications, so we reproduced these gating schemes instead (Figure 2 in Nilsson et al. 2013; and Figure 4A in Mosmann et al. 2014).

We performed the gating in Cytobank (online analysis platform). First, we excluded doublets, debris, and dead cells, following the gating hierarchies in the published figures. Next, we generated population labels for the rare populations of interest by following the remainder of the gating hierarchies from the figures.

Gating-ML 2.0 files exported from Cytobank are included together with all the other data files in FlowRepository (repository FR-FCM-ZZPH), allowing our gating schemes to be easily reproduced. In addition, copies of the gating diagrams are included in Supporting Information Figures S37–S38.

Number of clusters

Selection options and final number of clusters for each method

The number of clusters was a key parameter for many methods. Some methods provided an automatic option, others controlled it via an indirect parameter, and others left it as a direct parameter input. Supporting Information Table S3 summarizes the available options for each method; the selected option for the final results (for methods where more than one option was available); and the final number of clusters given by the selected option for each method. More details are provided in Supporting Information Table S1.

Supporting Information Table S3. Number of clusters for each method.

Method	Data set						Selection options		
	Lev_32	Lev_13	Sam_01	Sam_all	Nil_rare	Mos_rare	Automatic	Indirect parameters	Manual
ACCENSE	33	42	51	52	54	40	✓	✓*	
ClusterX	23	39	28	28	21	16	✓		
DensVM	14	25	7	15	11	9	✓		
FLOCK	32	20	23	31	22	9	✓		
flowClust	NA	40	40	40	40	40	✓		✓*
flowMeans	40	40	40	40	40	40	✓		✓*
flowMerge	NA	19	21	28	34	15		✓	
flowPeaks	4	7	2	10	1	1	✓	✓*	
FlowSOM	40	40	40	40	40	40	✓		✓*
FlowSOM_pre	100	100	100	100	100	400			✓
immunoClust	88	160	50	71	36	72	✓*	✓	
k-means	40	40	40	40	40	40			✓
PhenoGraph	32	26	24	30	29	25	✓*	✓	
Rclusterpp	40	40	40	40	40	40			✓
SamSPECTRAL	13	12	11	6	12	7	✓*	✓	
SPADE	NA	40	40	40	40	40			✓
SWIFT	762	673	614	701	95	144	✓		
X-shift	31	153	46	74	34	115	✓*	✓	

Options available for selecting the number of clusters for each method are indicated with check marks (✓). Stars indicate the option used when multiple options were available (✓*). Additional details are provided in Supporting Information Table S1 (spreadsheet file); including the number of clusters for the other options and the FlowCAP data sets, and additional explanatory notes.

Optimal number of clusters for FlowSOM

We also performed further analysis to investigate the optimal number of clusters for FlowSOM. We focused on FlowSOM for this analysis due to its good overall performance and fast runtime. For each data set, we ran FlowSOM over a sequence of values for k (the number of clusters), ranging from 5 to 80 in steps of 5. Results are presented in Supporting Information Figure S21. The results show that the mean F1 scores and F1 scores are maximized near 40 clusters for all six data sets, confirming that 40 clusters is an appropriate choice for these data sets.

Subsampling and multiple processor cores

Several methods required subsampling, due to the large size and dimensionality of the data sets. In addition, some methods were able to make use of multiple processor cores. The amount of subsampling and number of cores used for each method are summarized in Supporting Information Table S4 (also shown in Supporting Information Table S1). Subsampling was used when runtime was greater than 12 hours for methods on the Linux server, or greater than 6 hours for methods on the laptop (see Table 3 in main paper for final runtimes; and Supporting Information Table S1 for the environments used for each method).

Supporting Information Table S4. Subsampling and number of processor cores.

Method	No. of data points						No. of cores
	Lev_32	Lev_13	Sam_01	Sam_all	Nil_rare	Mos_rare	
ACCENSE	20,000	20,000	20,000	20,000	20,000	20,000	1
ClusterX	100,000	all	all	100,000	all	100,000	1
DensVM	100,000	100,000	all	100,000	all	100,000	1
FLOCK	all	all	all	all	all	all	1
flowClust	10,000	100,000	20,000	20,000	all	100,000	1
flowMeans	all	all	all	100,000	all	all	1
flowMerge	10,000	100,000	20,000	10,000	20,000	100,000	1
flowPeaks	all	all	all	all	all	all	1
FlowSOM	all	all	all	all	all	all	1
FlowSOM_pre	all	all	all	all	all	all	1
immunoClust	100,000	all	all	100,000	all	all	1
k-means	all	all	all	all	all	all	1
PhenoGraph	all	all	all	all	all	all	1
Rclusterpp	all	all	all	100,000	all	100,000	8
SamSPECTRAL	all	all	all	100,000	all	all	1
SPADE	all	all	all	all	all	all	up to 64*
SWIFT	100,000	all	all	100,000	all	all	2
X-shift	all	all	all	300,000	all	all	4

No subsampling was required for entries labeled “all”. (*) SPADE did not allow the number of cores to be specified, so all available cores on the Linux server were used. Additional details are provided in Supporting Information Table S1 (spreadsheet file); including numbers for the FlowCAP data sets, hardware specifications, and additional explanatory notes.

Ensemble clustering

For the ensemble clustering (consensus clustering), we included all methods for each data set, excluding the following:

- Methods that required subsampled data, since consensus clustering requires cluster labels for the same data points for each method (see Supporting Information Table S4).
- Methods that removed outliers, for the same reason. This excluded **SamSPECTRAL** for all data sets; **flowClust** for the two data sets with a single rare population of interest; **SWIFT** for the four data sets with multiple populations of interest; and **X-shift** for all data sets except **Samusik_01** and **Nilsson_rare**.
- Methods that generated a large number of small clusters (**FlowSOM_pre**; and **SWIFT** for the data sets with multiple populations), since this greatly slowed runtime.

Stability of clustering results

For the stability analysis, we ran clustering methods 30 times with varying random starts or bootstrap resampling. In order to keep total computational requirements manageable, we restricted the analysis as follows:

- We included only one data set with multiple populations of interest (**Levine_32dim**), and one data set with a single rare population of interest (**Mosmann_rare**).
- Methods that could not be run via a scripting approach were excluded, since it was not possible to automate the multiple runs (i.e. methods running via graphical interfaces were excluded; see Table 1 in main paper).
- Methods that required more than 5 hours of runtime for each individual run were excluded (see Table 3 in main paper).
- Methods that required subsampling were excluded (see Supporting Information Table S4), since this caused problems during parallelized evaluation of the F1 scores.
- Methods that required multiple processor cores for each instance were excluded (see Supporting Information Table S4), since this also caused problems during parallelization.

We ran the methods in parallel on the Linux server where possible (**FLOCK** was run in series, since it saves external results files that are automatically overwritten). The final sets of included methods can be seen in Supporting Information Figures S17–S20.

Supporting Information Results

Results for FlowCAP data sets

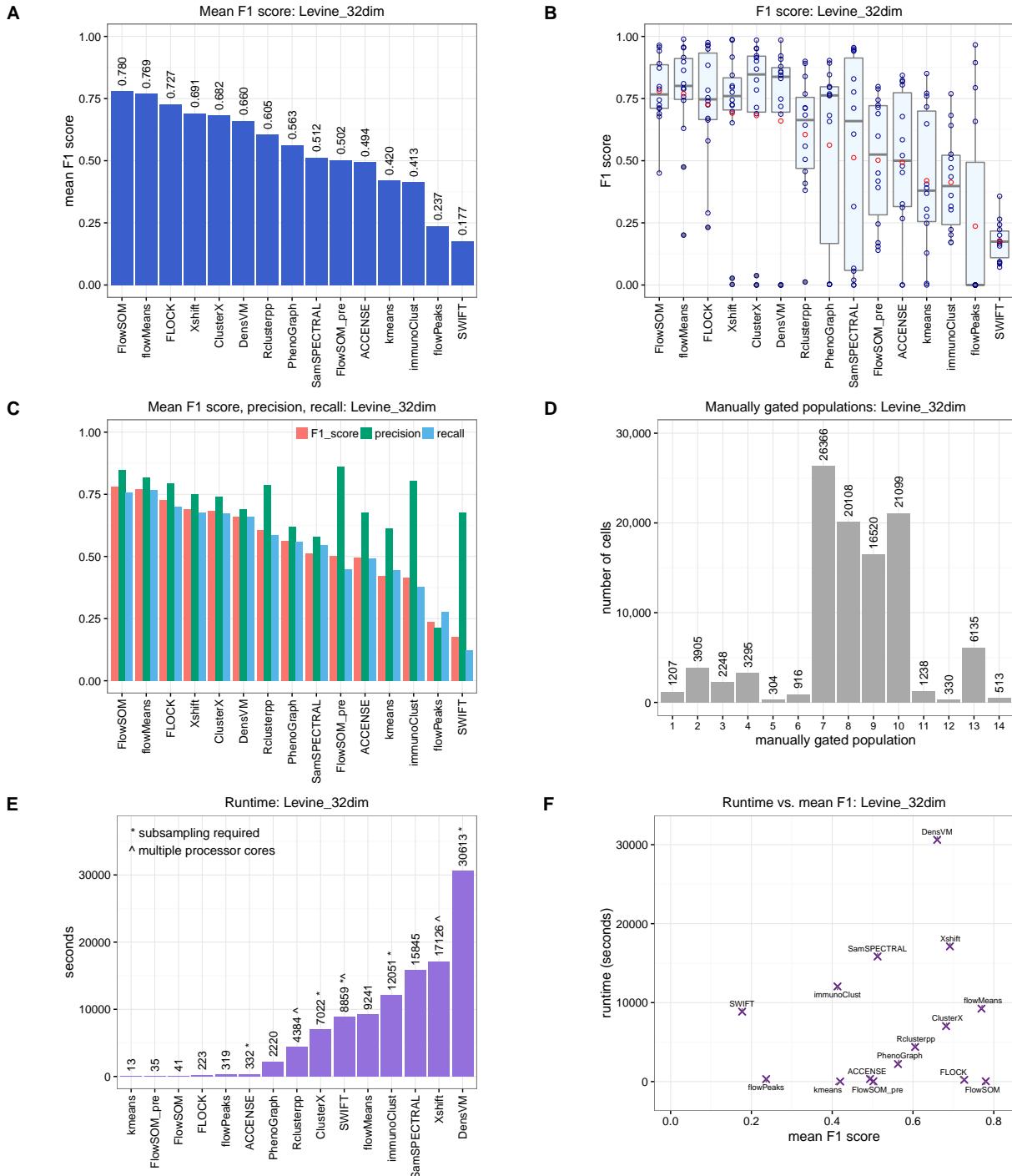
Supporting Information Table S5. Results for FlowCAP data sets.

Method	Updated evaluation methodology		FlowCAP-I evaluation methodology	
	FlowCAP_ND	FlowCAP_WNV	FlowCAP_ND	FlowCAP_WNV
			mean F1	mean F1
ACCENSE	NA	NA	NA	NA
ClusterX	0.393	0.550	0.439	0.398
DensVM	0.401	0.532	0.515	0.457
FLOCK	0.423	0.732	0.897	0.871
flowClust	0.391	0.604	0.739	0.837
flowMeans	0.182	0.685	0.667	0.891
flowMerge	0.485	0.678	0.900	0.864
flowPeaks	0.129	0.314	0.764	0.733
FlowSOM	0.306	0.486	0.879	0.732
FlowSOM_pre	0.248	0.273	0.103	0.096
immunoClust	0.246	0.591	0.783	0.463
k-means	0.381	0.586	0.659	0.816
PhenoGraph	NA	NA	NA	NA
Rclusterpp	0.453	0.552	0.874	0.799
SamSPECTRAL	0.355	0.632	0.879	0.851
SPADE	0.249	NA	0.632	NA
SWIFT	NA	NA	NA	NA
X-shift	NA	NA	NA	NA

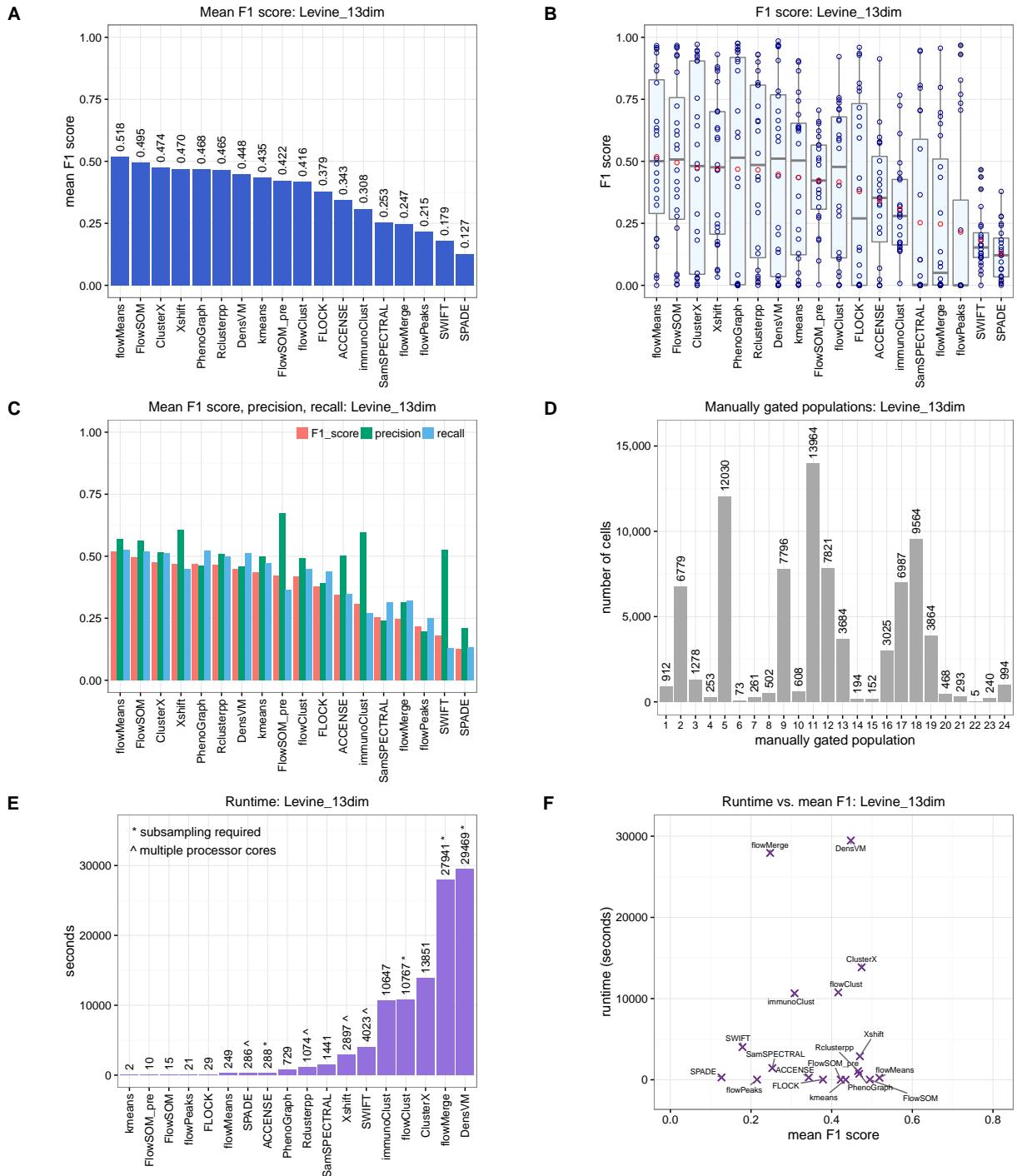
Results show the mean F1 score for each data set. Differences in evaluation methodologies are explained in Materials and Methods (main paper). Methods accessible via graphical interfaces were excluded, since differences in population indices between samples necessitated evaluation on a per-sample basis, which required a scripting approach (see Supporting Information Methods; Data sets). NA = not available, due to graphical interface or errors. See Supporting Information Table S1 (spreadsheet file) for further details.

Supporting Information Figures

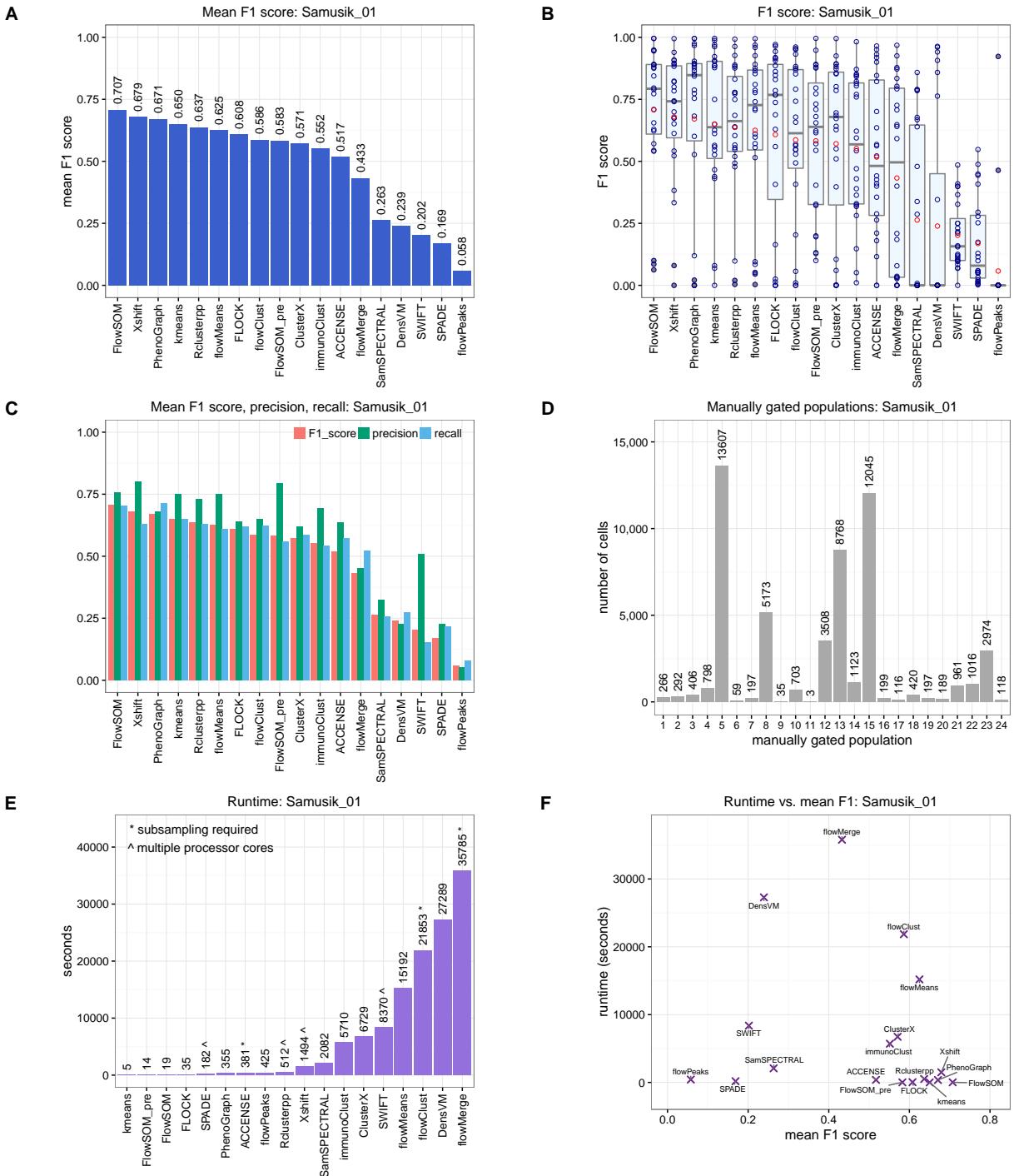
Main results: all data sets



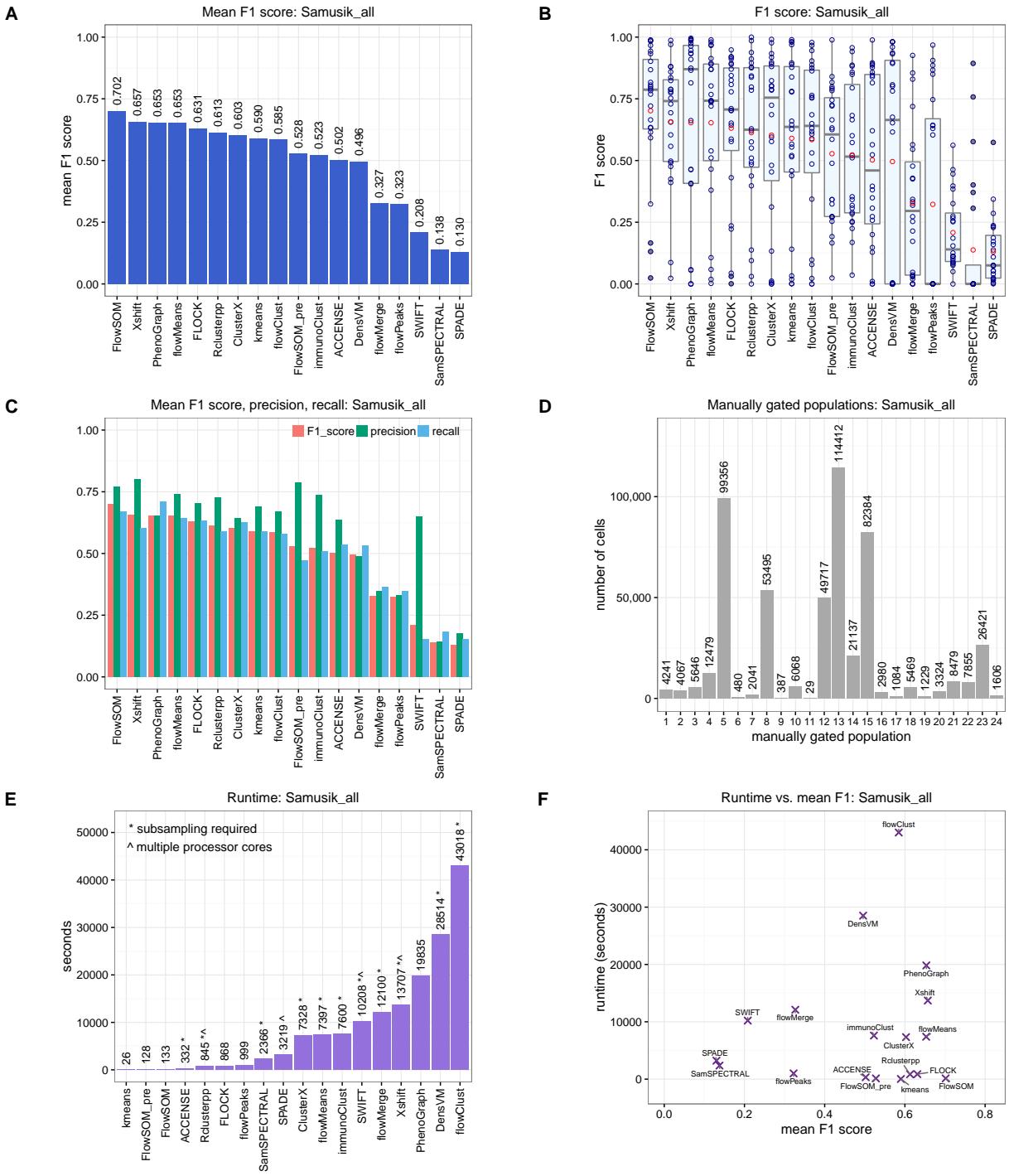
Supporting Information Figure S1. Results of comparison of clustering methods for data set Levine_32dim. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall. (D) Number of cells per reference population. (E) Runtimes. (F) Runtime vs. mean F1 score; methods combining high mean F1 scores with fast runtimes are seen toward the bottom-right. (This figure is included as Figure 1 in the main paper.)



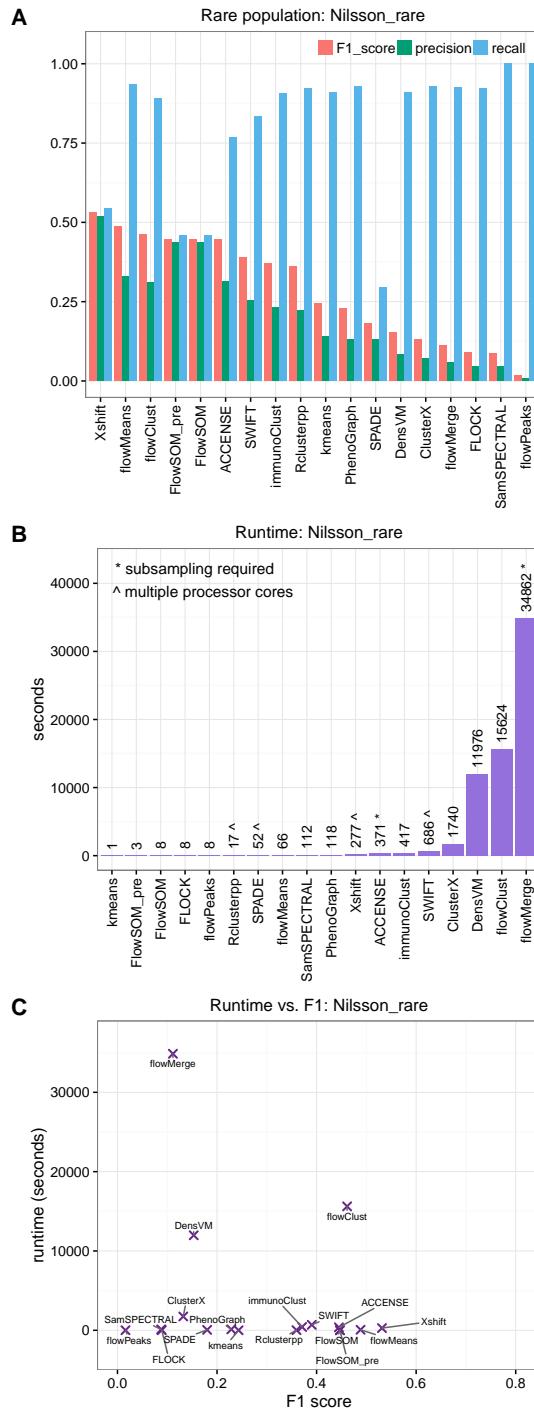
Supporting Information Figure S2. Results of comparison of clustering methods for data set Levine_13dim. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall. (D) Number of cells per reference population. (E) Runtimes. (F) Runtime vs. mean F1 score; methods combining high mean F1 scores with fast runtimes are seen toward the bottom-right.



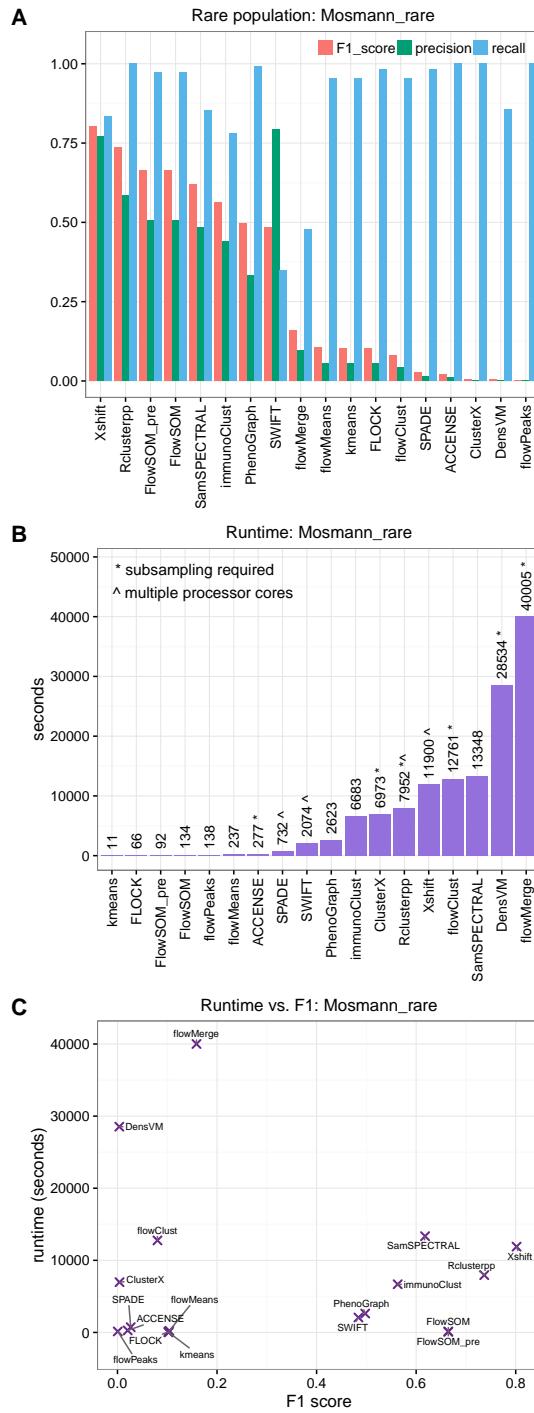
Supporting Information Figure S3. Results of comparison of clustering methods for data set Samusik_01. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall. (D) Number of cells per reference population. (E) Runtimes. (F) Runtime vs. mean F1 score; methods combining high mean F1 scores with fast runtimes are seen toward the bottom-right.



Supporting Information Figure S4. Results of comparison of clustering methods for data set Samusik_all. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall. (D) Number of cells per reference population. (E) Runtimes. (F) Runtime vs. mean F1 score; methods combining high mean F1 scores with fast runtimes are seen toward the bottom-right.

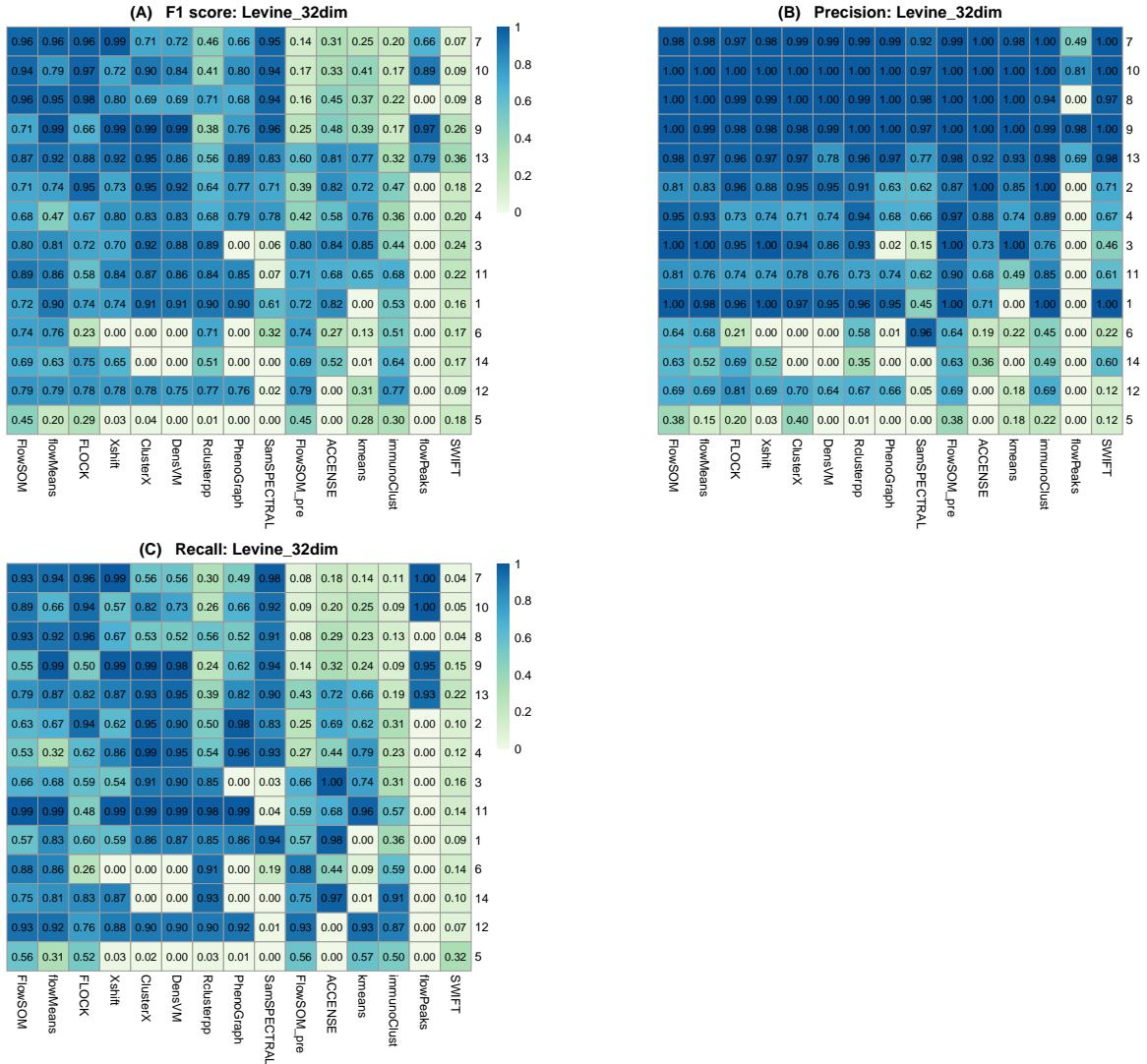


Supporting Information Figure S5. Results of comparison of clustering methods for data set Nilsson_rare. (A) F1 score, precision, and recall for the rare cell population of interest. The rare population contains approximately 0.8% of total cells. (B) Runtimes. (C) Runtime vs. F1 score; methods combining high F1 scores with fast runtimes are seen toward the bottom-right.

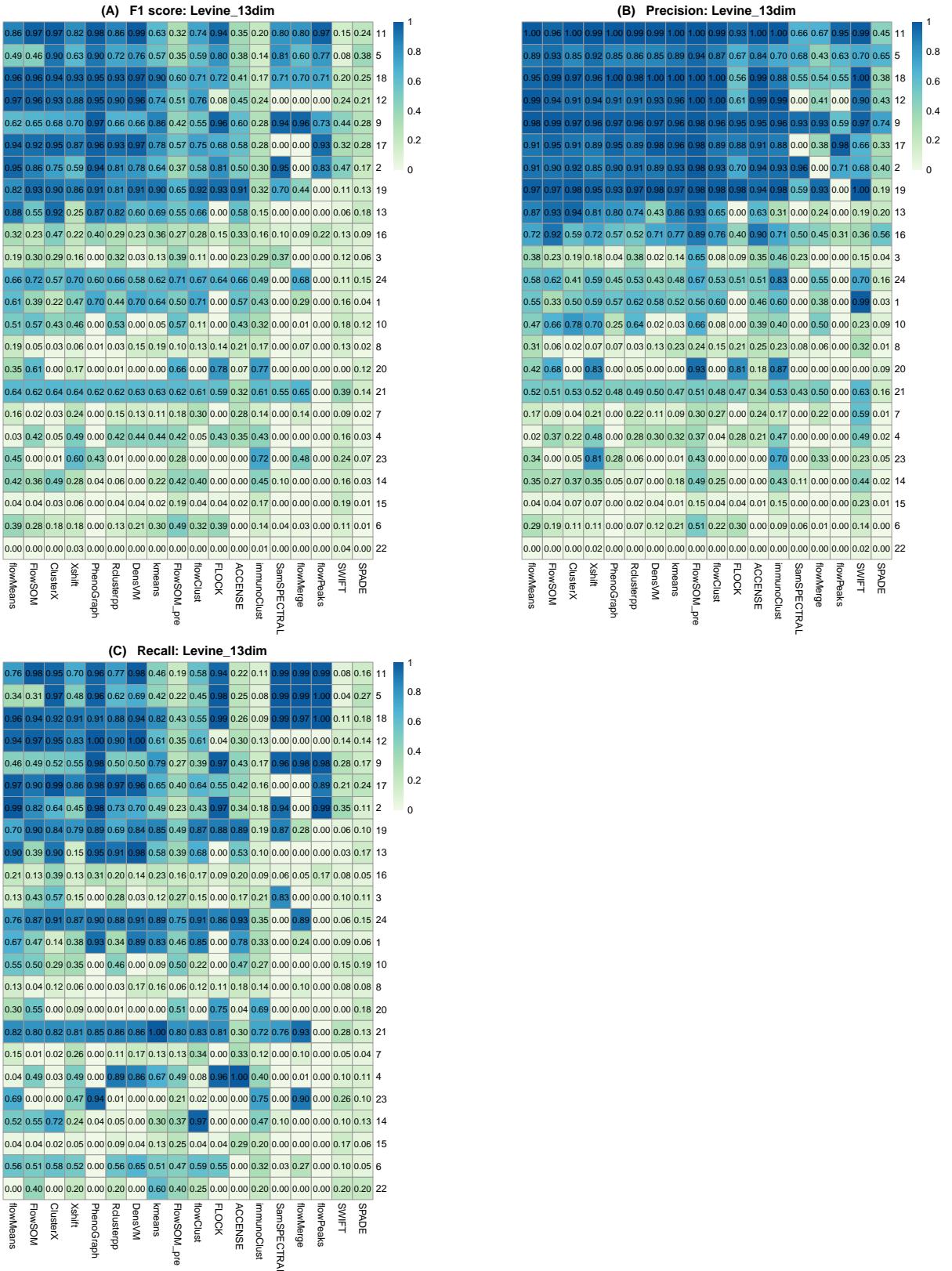


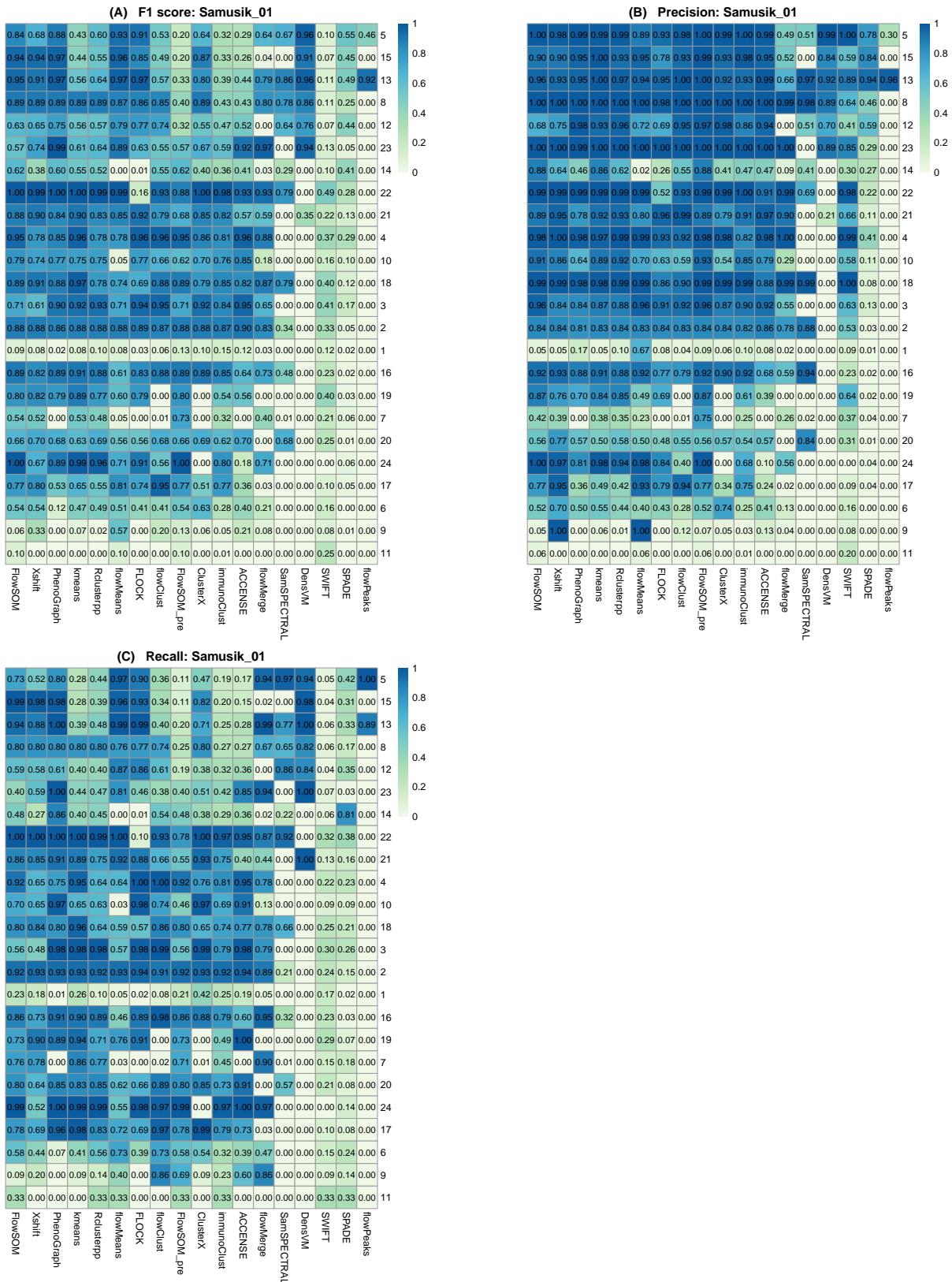
Supporting Information Figure S6. Results of comparison of clustering methods for data set Mosmann_rare. (A) F1 score, precision, and recall for the rare cell population of interest. The rare population contains approximately 0.03% of total cells. (B) Runtimes. (C) Runtime vs. F1 score; methods combining high F1 scores with fast runtimes are seen toward the bottom-right. (This figure is included as Figure 3 in the main paper.)

Additional results for individual populations

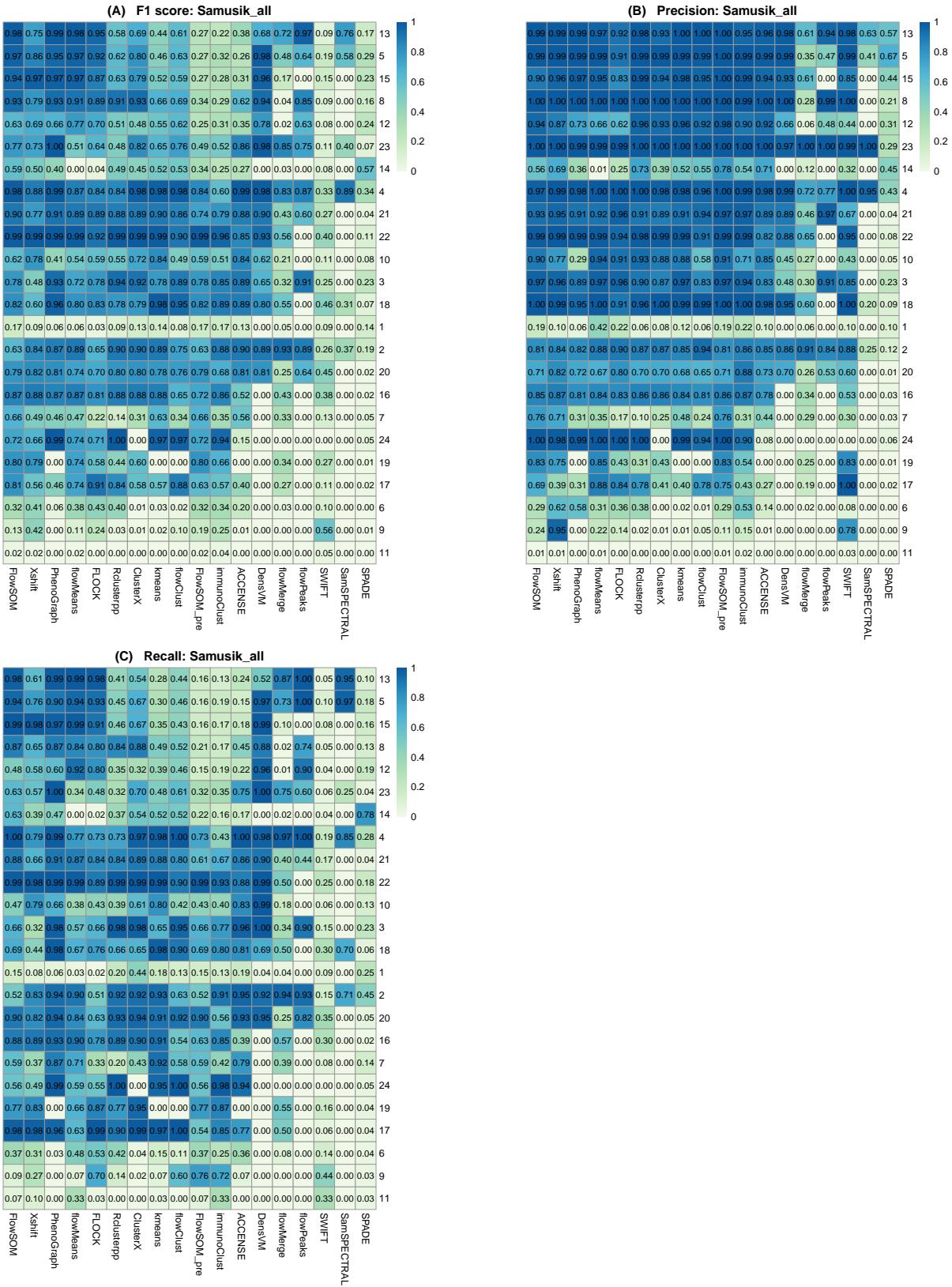


Supporting Information Figure S7. Additional results for individual cell populations (ordered by size) for data set Levine_32dim. Heatmaps show F1 scores (A), precision (B), and recall (C) for each reference population (manually gated population). Rows (populations) are ordered by decreasing population size, with the largest population at the top. Columns (clustering methods) are ordered by mean F1 score, as in the main results.



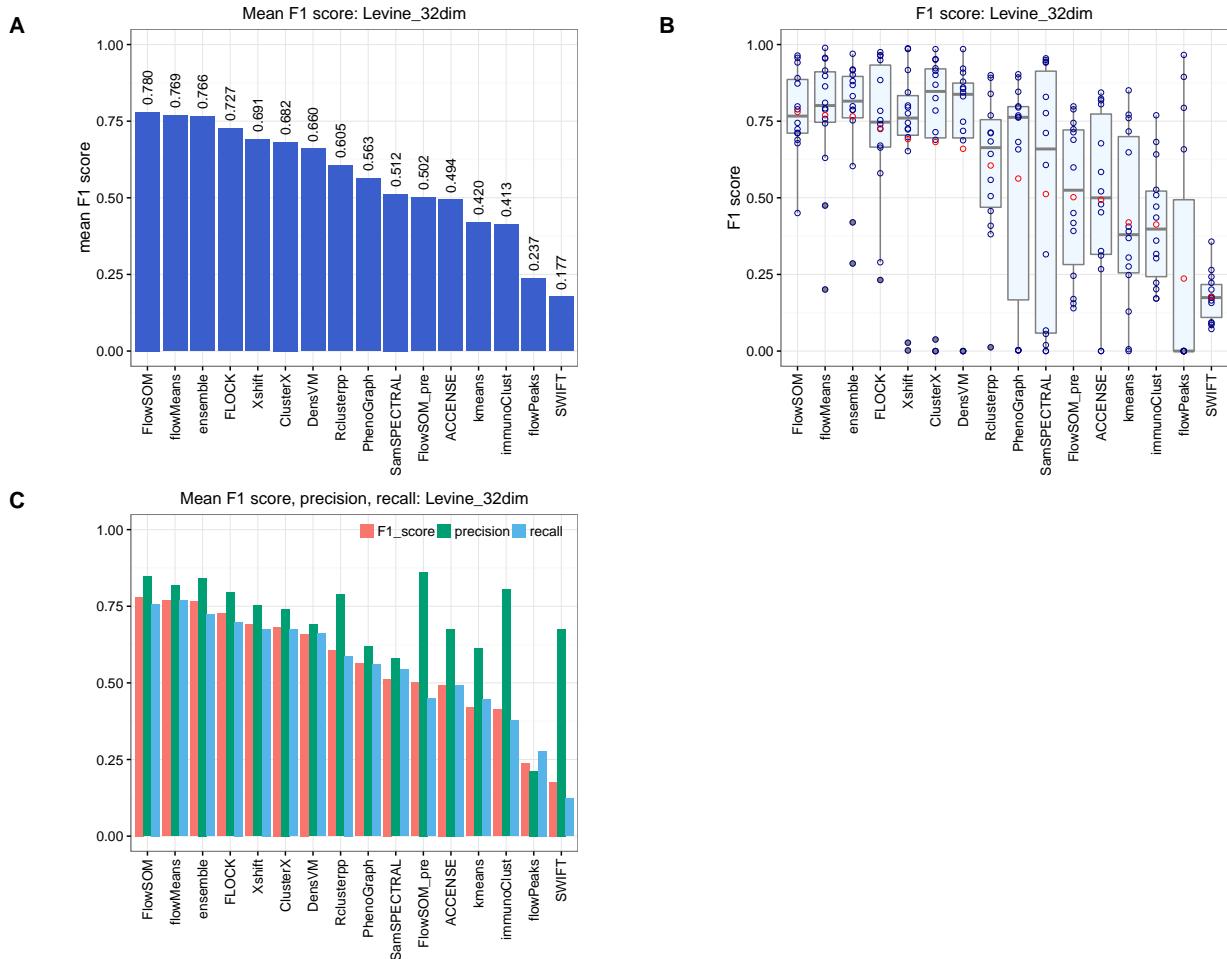


Supporting Information Figure S9. Additional results for individual cell populations (ordered by size) for data set Samusik.01. Heatmaps show F1 scores (A), precision (B), and recall (C) for each reference population (manually gated population). Rows (populations) are ordered by decreasing population size, with the largest population at the top. Columns (clustering methods) are ordered by mean F1 score, as in the main results.

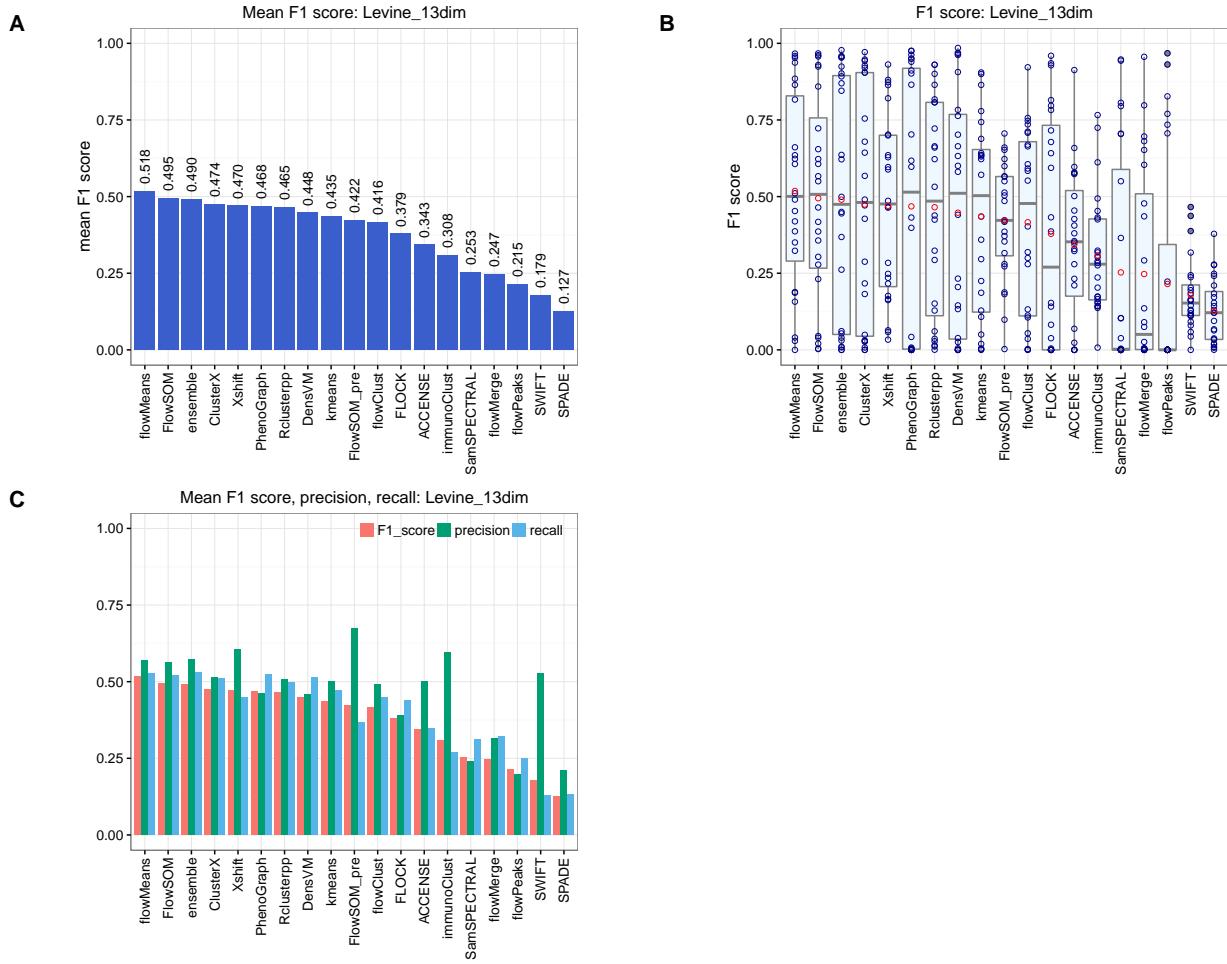


Supporting Information Figure S10. Additional results for individual cell populations (ordered by size) for data set Samusik.all. Heatmaps show F1 scores (A), precision (B), and recall (C) for each reference population (manually gated population). Rows (populations) are ordered by decreasing population size, with the largest population at the top. Columns (clustering methods) are ordered by mean F1 score, as in the main results.

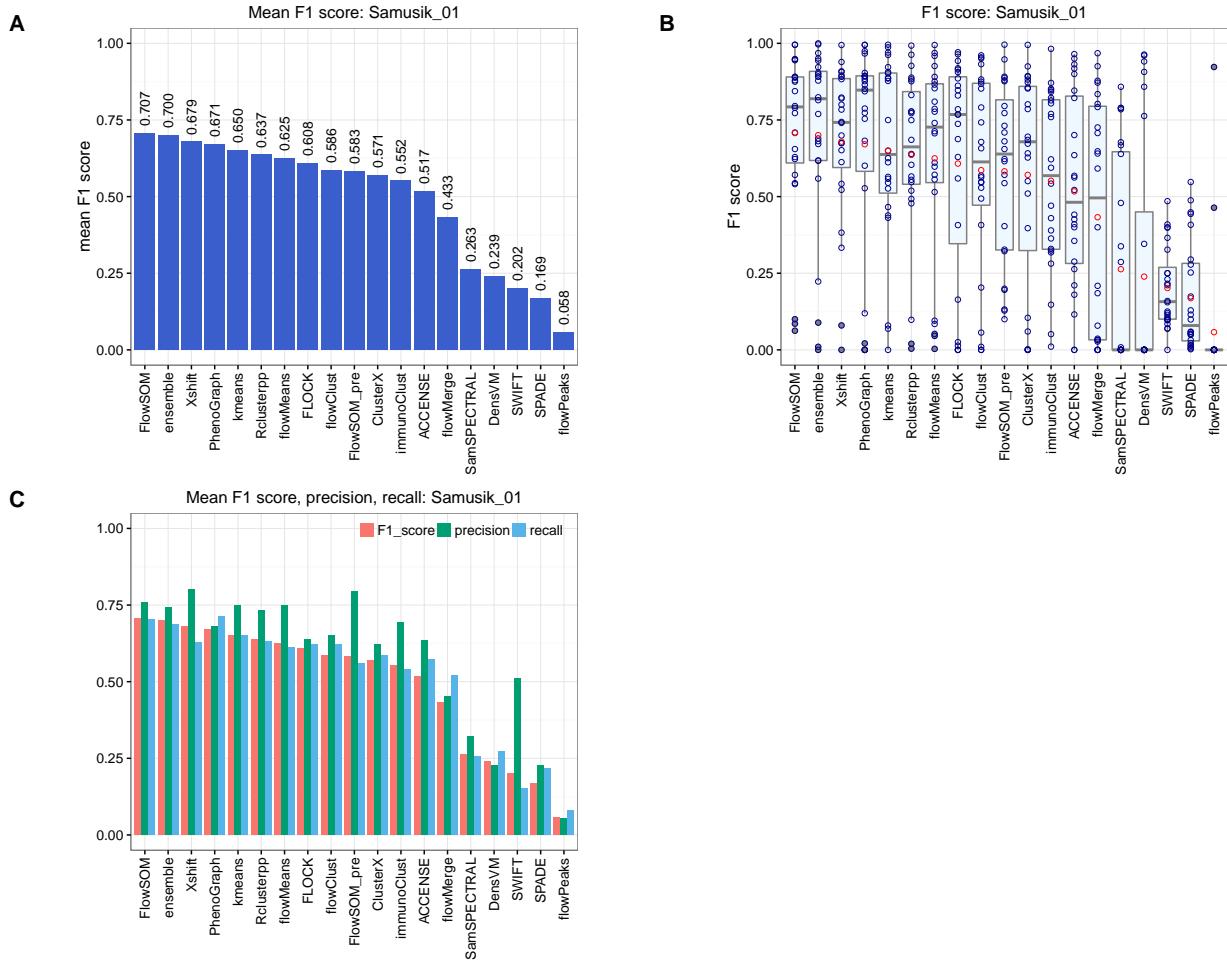
Ensemble clustering



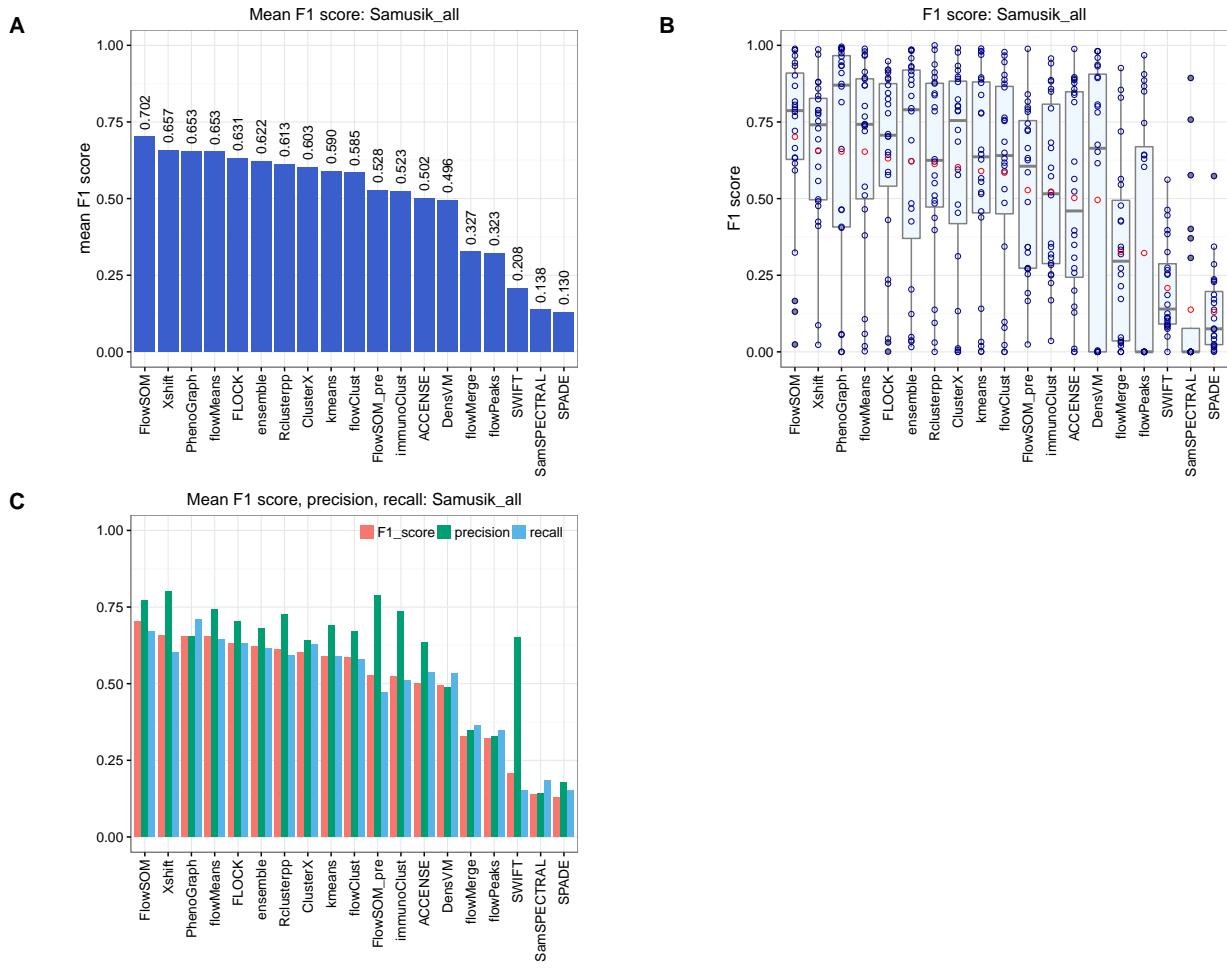
Supporting Information Figure S11. Ensemble clustering results for data set Levine_32dim. Results of comparison of clustering methods, including ensemble clustering. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall.



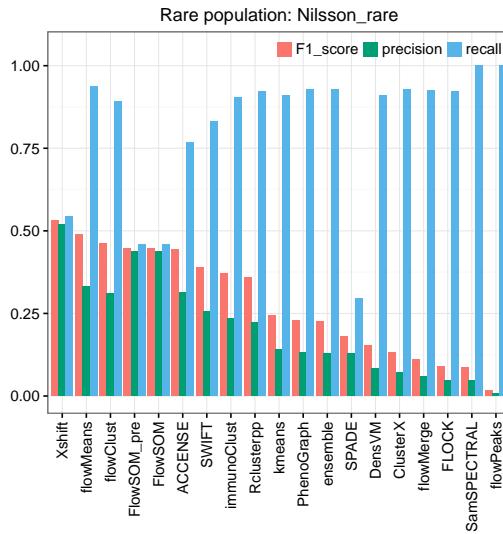
Supporting Information Figure S12. Ensemble clustering results for data set Levine_13dim. Results of comparison of clustering methods, including ensemble clustering. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall.



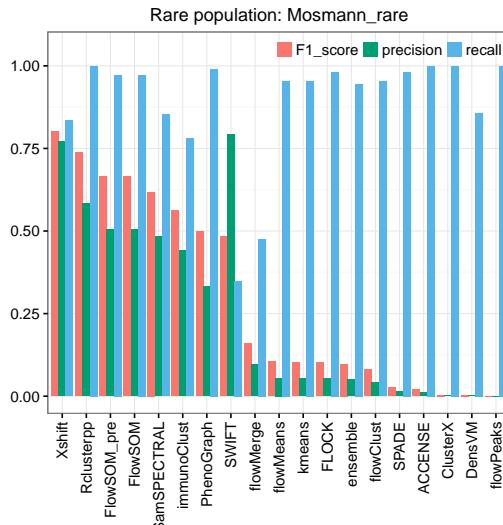
Supporting Information Figure S13. Ensemble clustering results for data set Samusik_01. Results of comparison of clustering methods, including ensemble clustering. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall.



Supporting Information Figure S14. Ensemble clustering results for data set Samusik_all. Results of comparison of clustering methods, including ensemble clustering. (A) Mean F1 score across cell populations. (B) Distributions of F1 scores across cell populations. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers, with means shown additionally in red. (C) Mean F1 scores, mean precision, and mean recall.

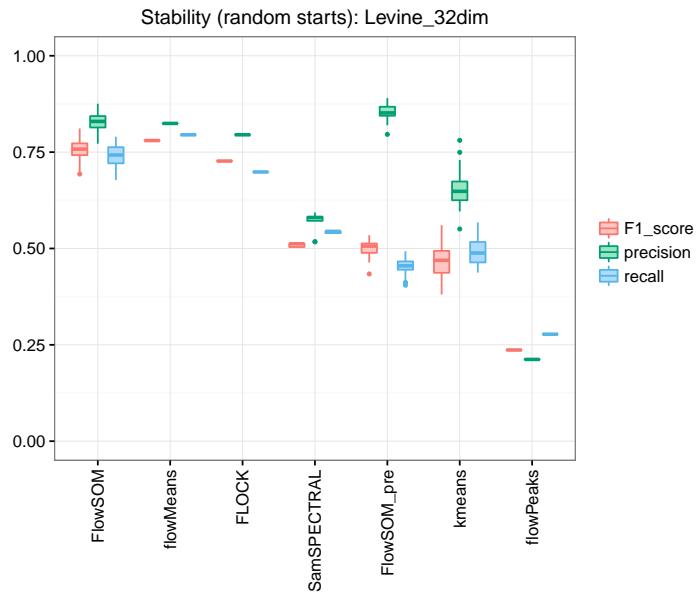


Supporting Information Figure S15. Ensemble clustering results for data set Nilsson_rare.
Results of comparison of clustering methods, including ensemble clustering. F1 score, precision, and recall for the rare cell population of interest. The rare population contains approximately 0.8% of total cells.

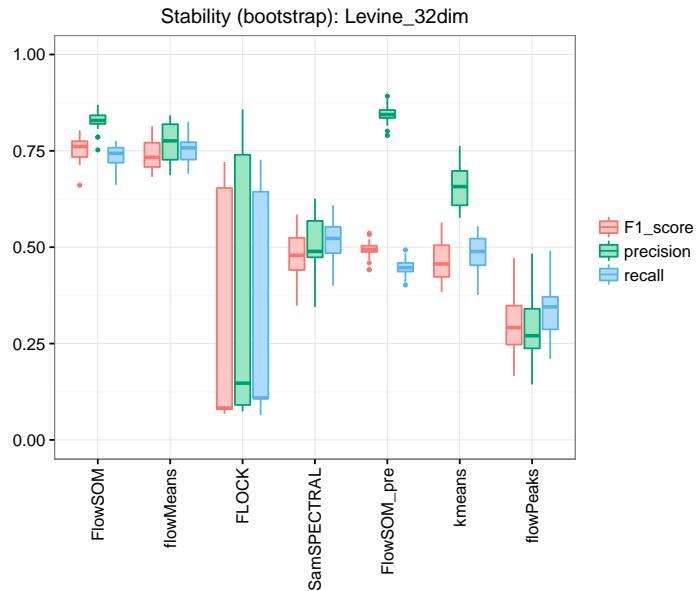


Supporting Information Figure S16. Ensemble clustering results for data set Mosmann_rare.
Results of comparison of clustering methods, including ensemble clustering. F1 score, precision, and recall for the rare cell population of interest. The rare population contains approximately 0.03% of total cells.

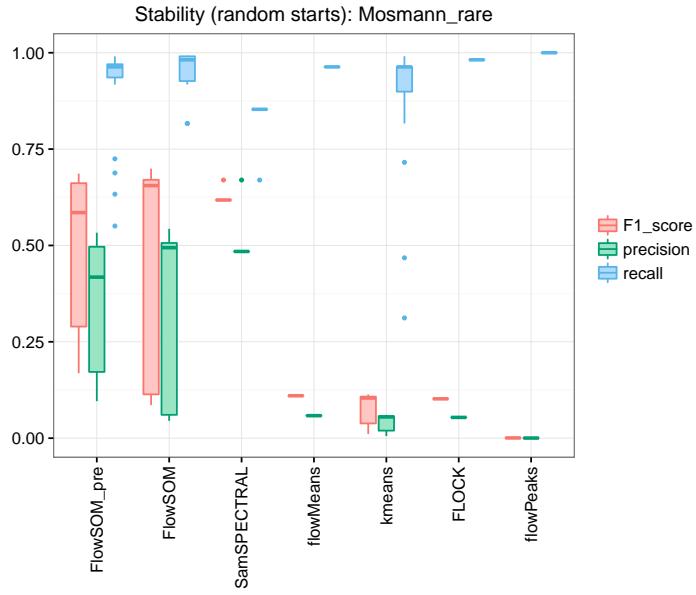
Stability of clustering results



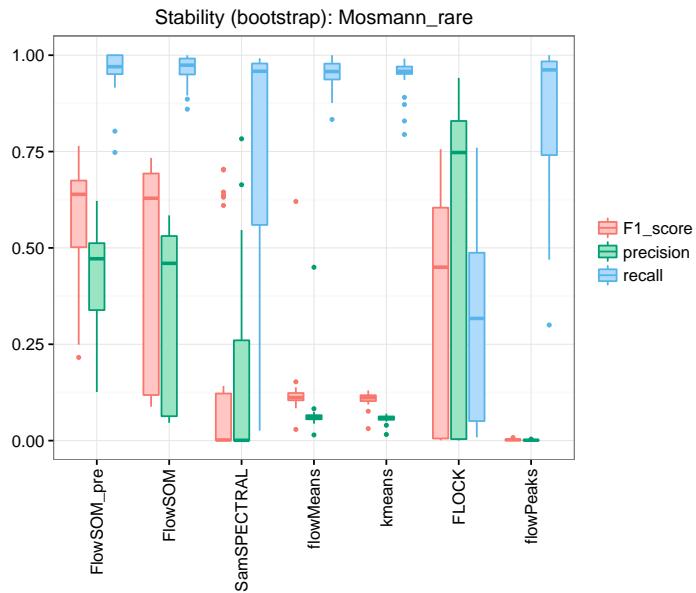
Supporting Information Figure S17. Stability of clustering results across random starts for data set Levine_32dim. Methods were run 30 times with different random starts. Figure displays distributions of mean F1 scores, mean precision, and mean recall across the replicates. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers. Note that some methods did not allow access to random seeds during parallelization; for these, the bootstrap results (next figure) are more informative.



Supporting Information Figure S18. Stability of clustering results across bootstrap resamples for data set Levine_32dim. Methods were run 30 times for different bootstrap resamples. Figure displays distributions of F1 scores, precision, and recall across the bootstrap resamples. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers.

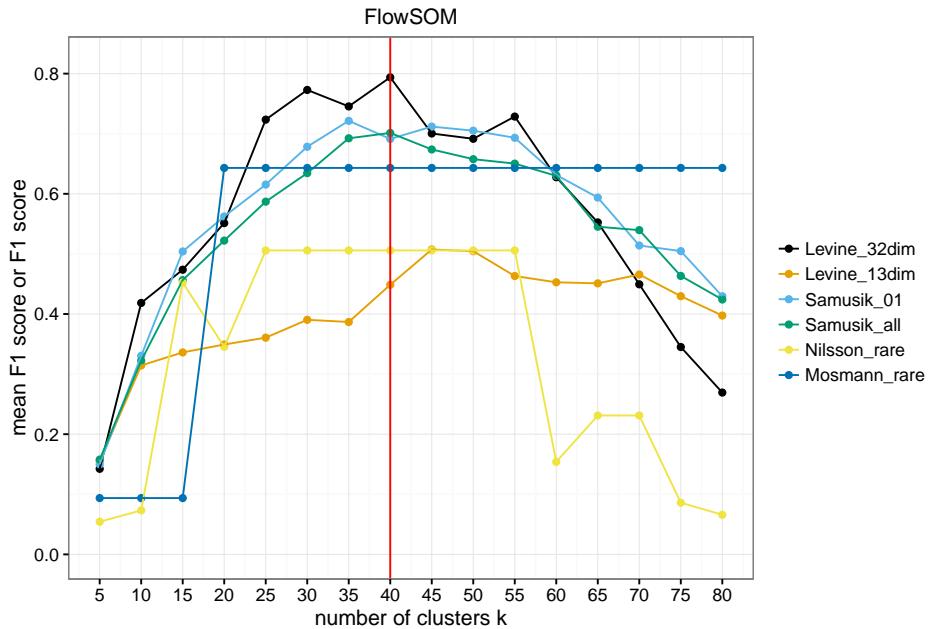


Supporting Information Figure S19. Stability of clustering results across random starts for data set Mosmann_rare. Methods were run 30 times with different random starts. Figure displays distributions of mean F1 scores, mean precision, and mean recall across the replicates. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers. Note that some methods did not allow access to random seeds during parallelization; for these, the bootstrap results (next figure) are more informative.



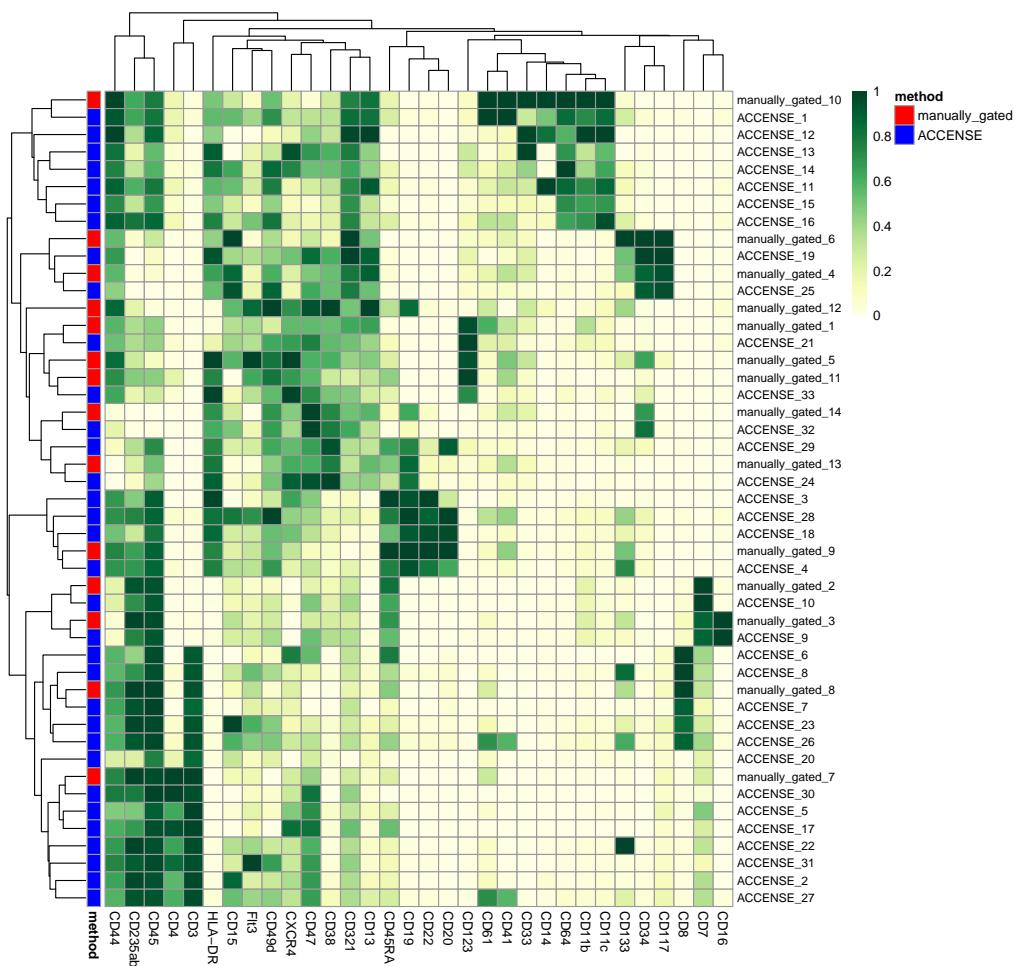
Supporting Information Figure S20. Stability of clustering results across bootstrap resamples for data set Mosmann_rare. Methods were run 30 times for different bootstrap resamples. Figure displays distributions of F1 scores, precision, and recall across the bootstrap resamples. The box plots show medians, upper and lower quartiles, whiskers extending to 1.5 times the interquartile range, and outliers.

Optimal number of clusters for FlowSOM

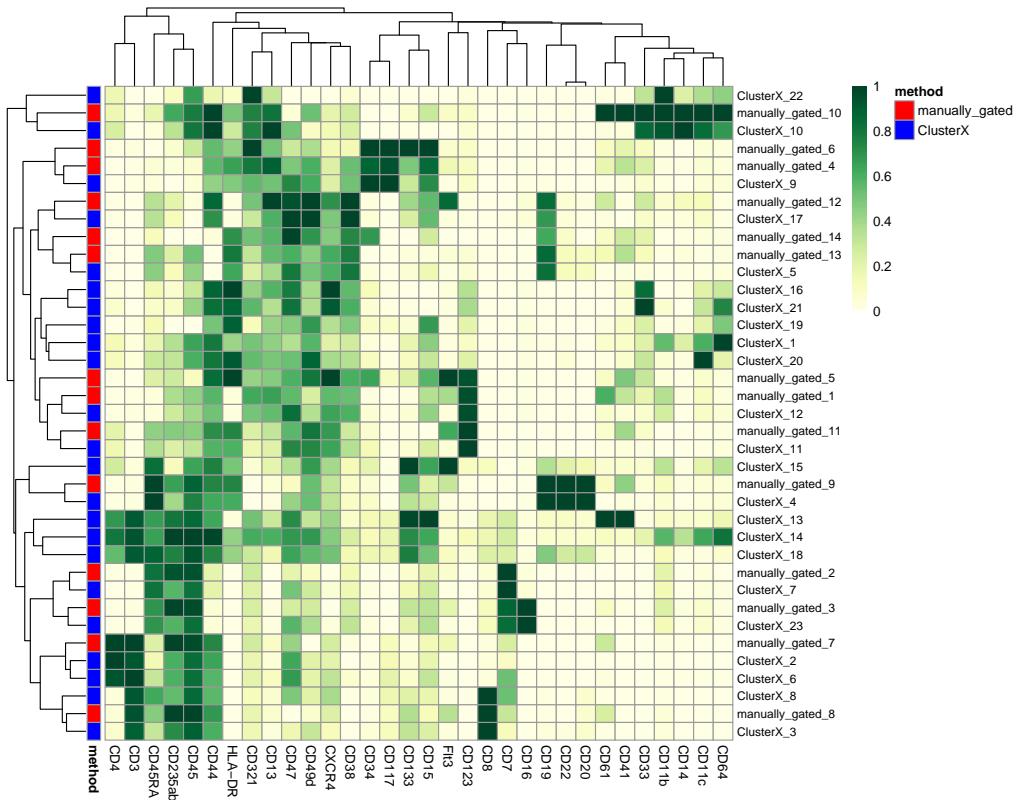


Supporting Information Figure S21. Optimal number of clusters for FlowSOM. FlowSOM was run over a sequence of values for k (number of clusters) between 5 and 80, in steps of 5. Figure displays mean F1 scores (data sets with multiple cell populations of interest: Levine_32dim, Levine_13dim, Samusik_01, and Samusik_all), and F1 scores (data sets with a single rare cell population of interest: Nilsson_rare and Mosmann_rare). The main results used 40 clusters, indicated with a red vertical line.

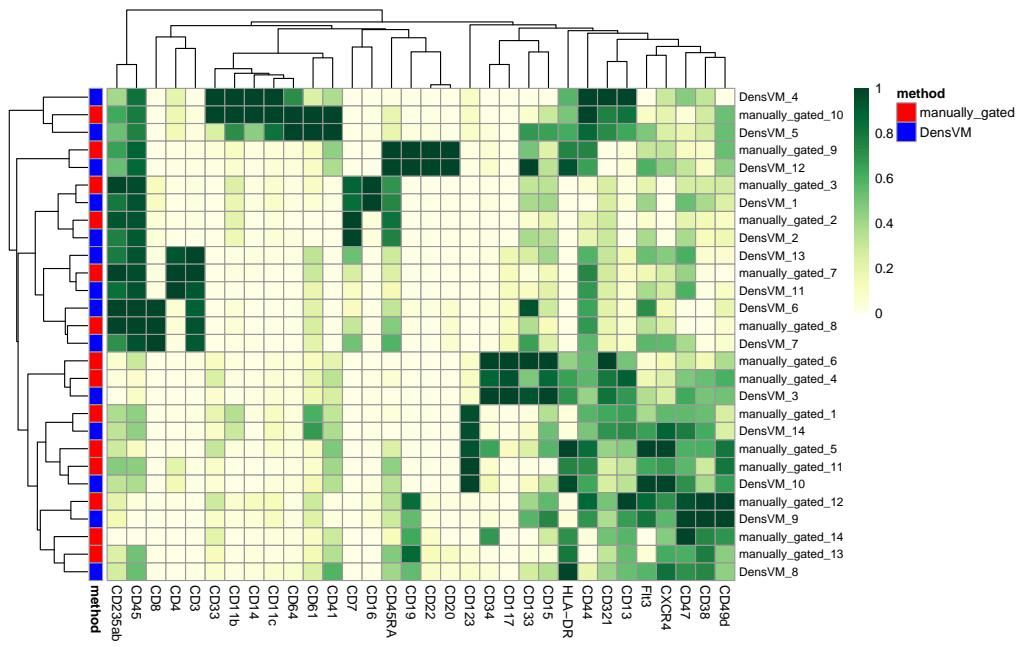
Cluster median expression profiles



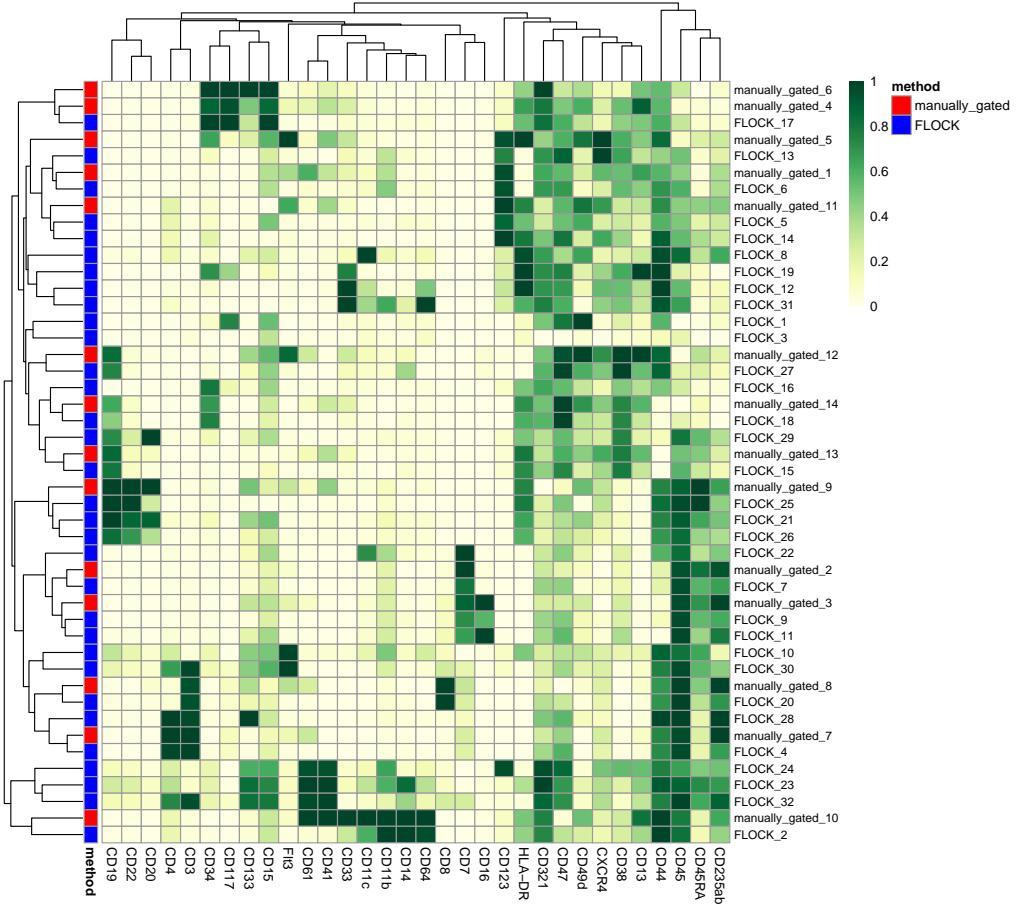
Supporting Information Figure S22. Median expression profiles of detected clusters and reference populations, ACCENSE, data set Levine.32dim. See Figure 2 (main paper) for details.



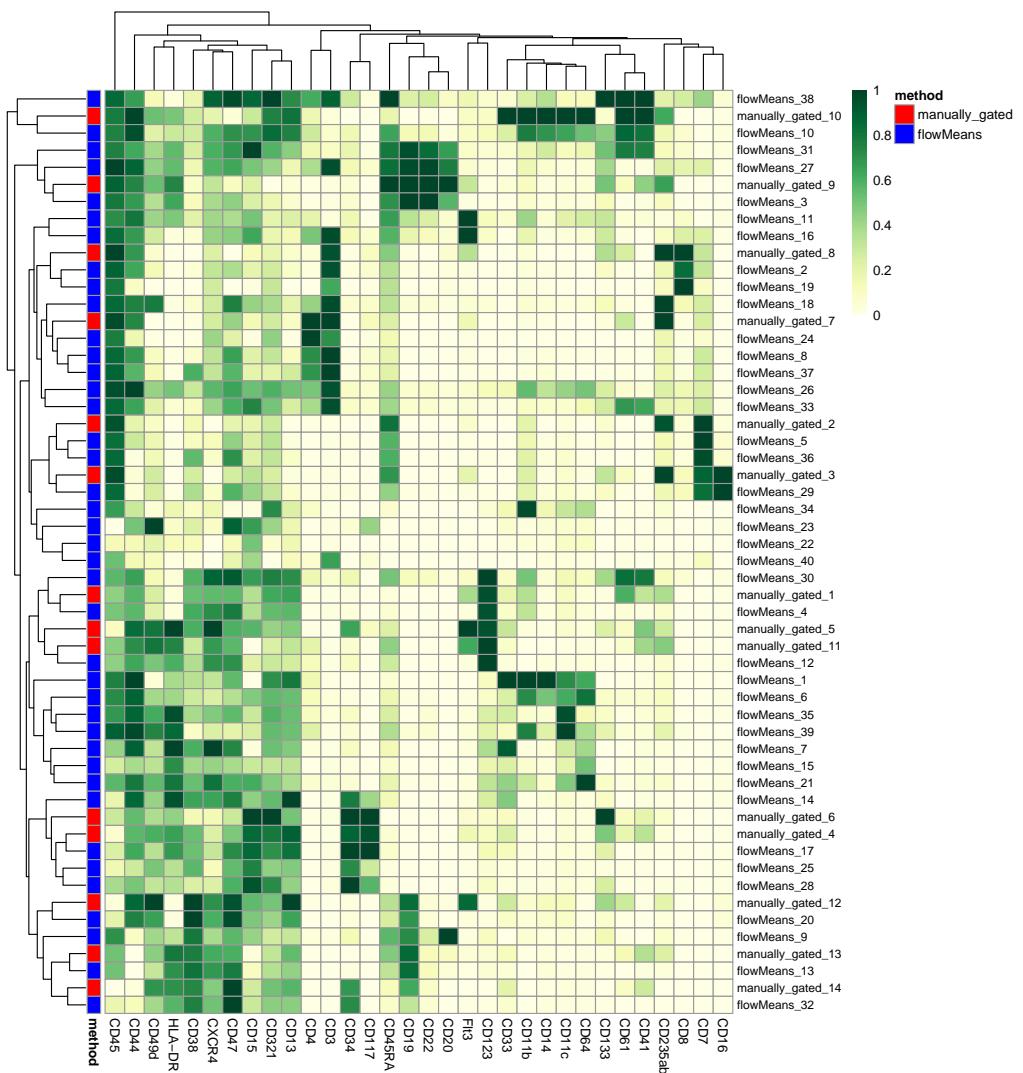
Supporting Information Figure S23. Median expression profiles of detected clusters and reference populations, ClusterX, data set Levine_32dim. See Figure 2 (main paper) for details.



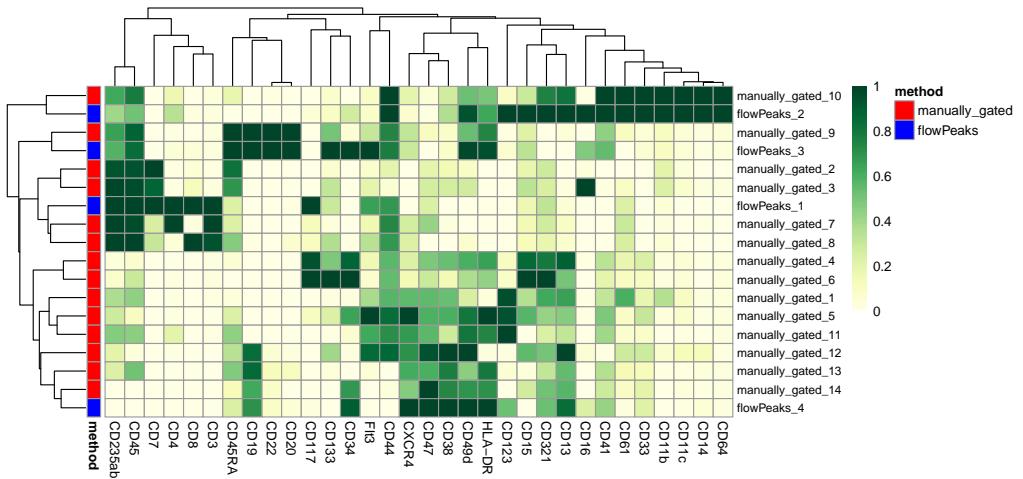
Supporting Information Figure S24. Median expression profiles of detected clusters and reference populations, DensVM, data set Levine_32dim. See Figure 2 (main paper) for details.



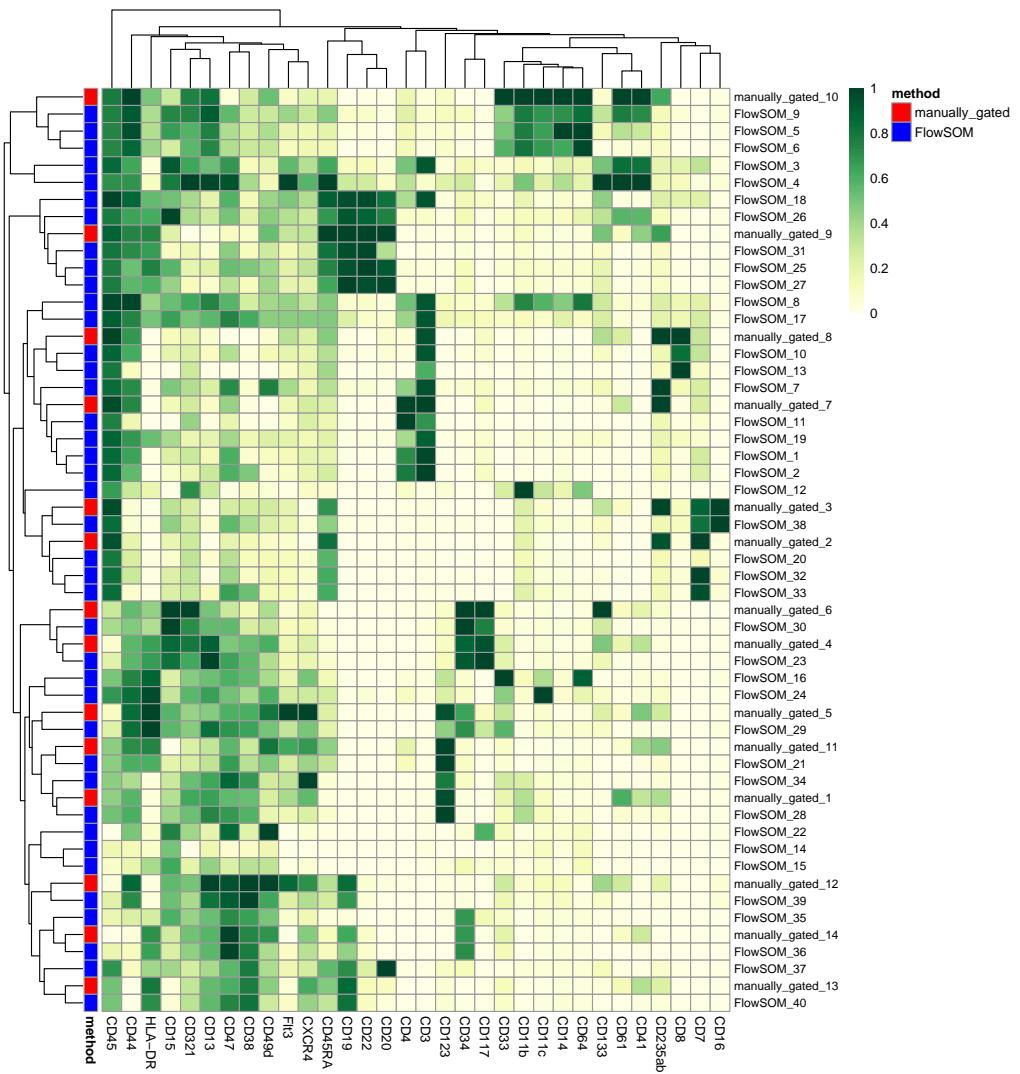
Supporting Information Figure S25. Median expression profiles of detected clusters and reference populations, FLOCK, data set Levine_32dim. See Figure 2 (main paper) for details.



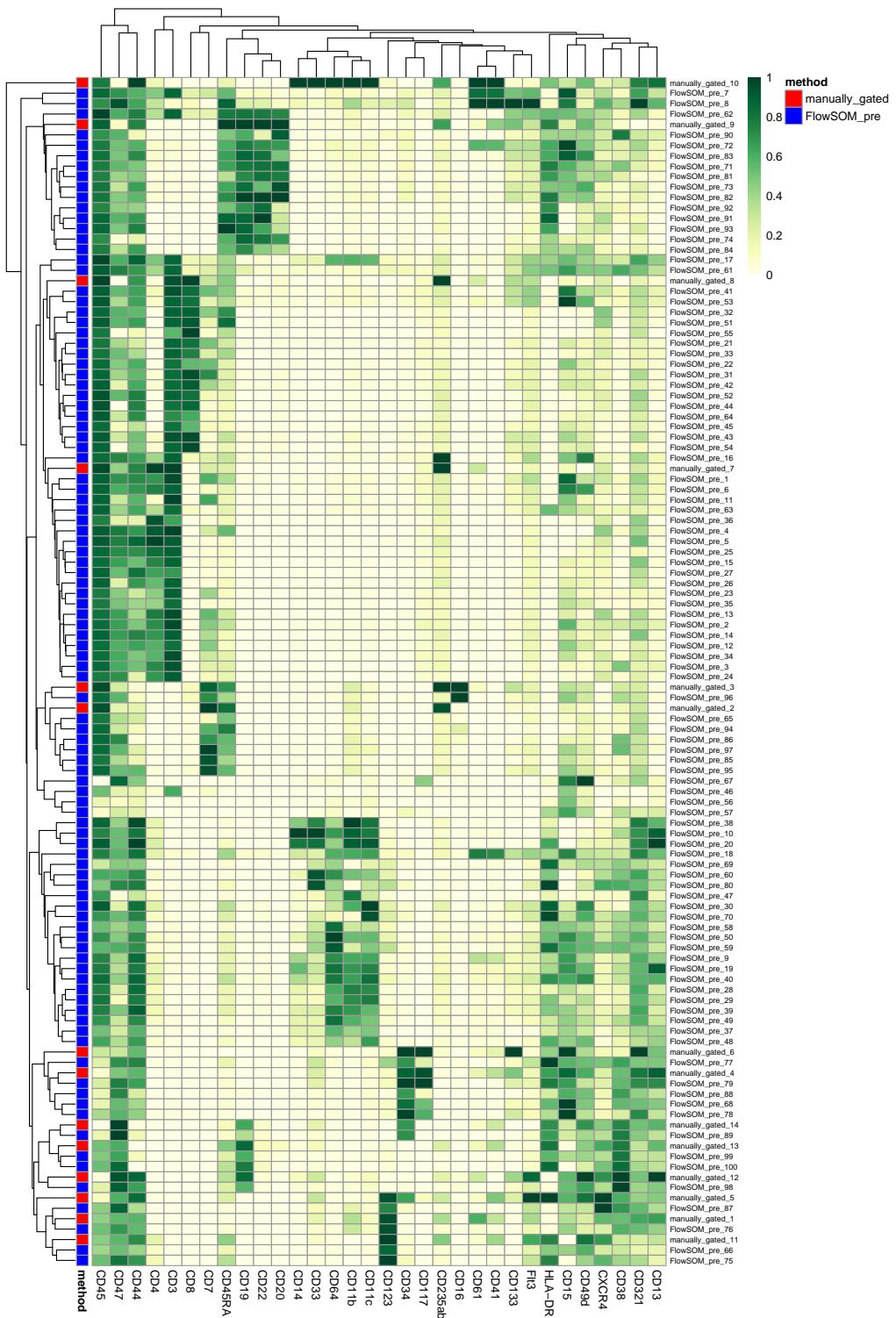
Supporting Information Figure S26. Median expression profiles of detected clusters and reference populations, **flowMeans**, data set Levine_32dim. See Figure 2 (main paper) for details.



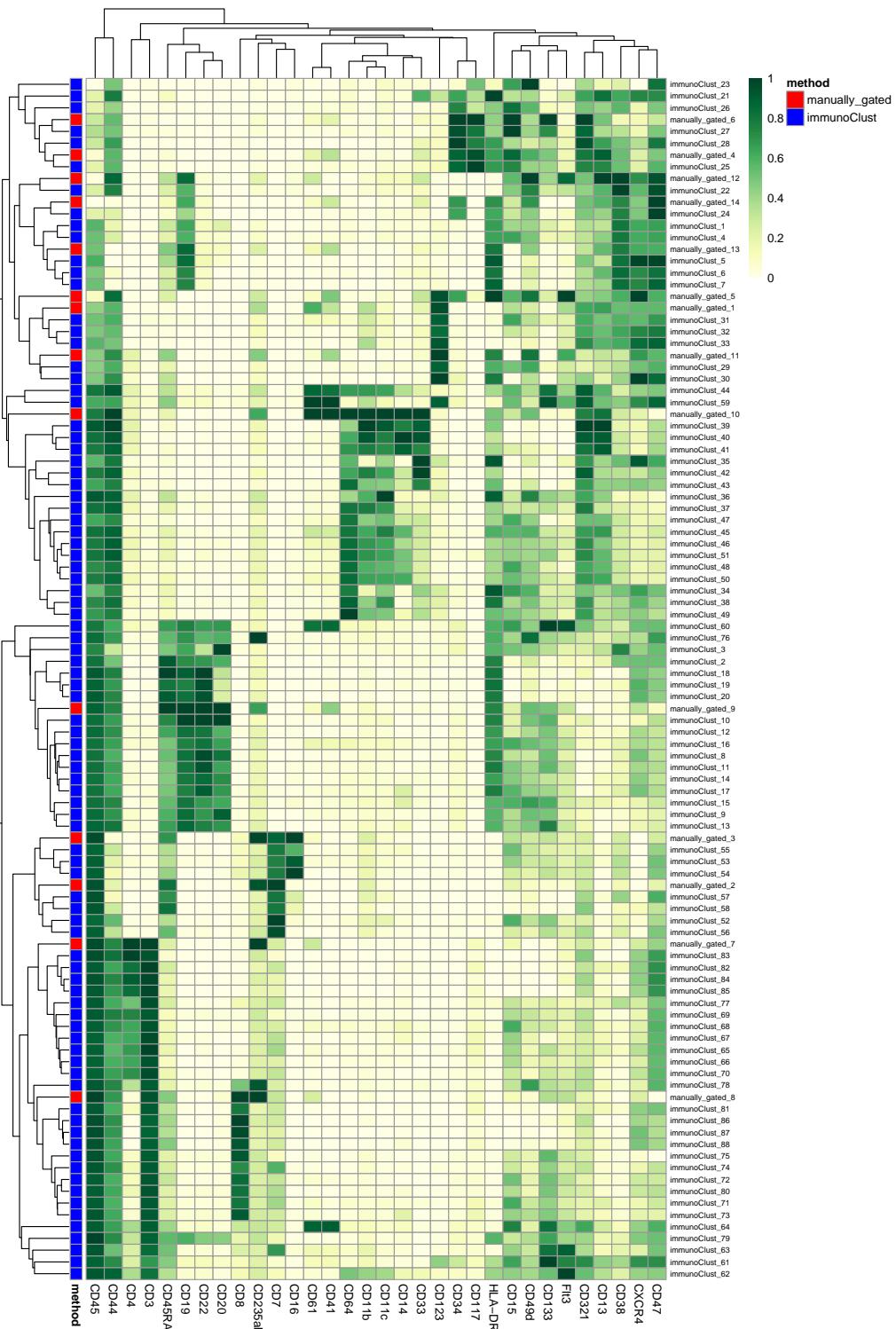
Supporting Information Figure S27. Median expression profiles of detected clusters and reference populations, **flowPeaks**, data set Levine_32dim. See Figure 2 (main paper) for details.



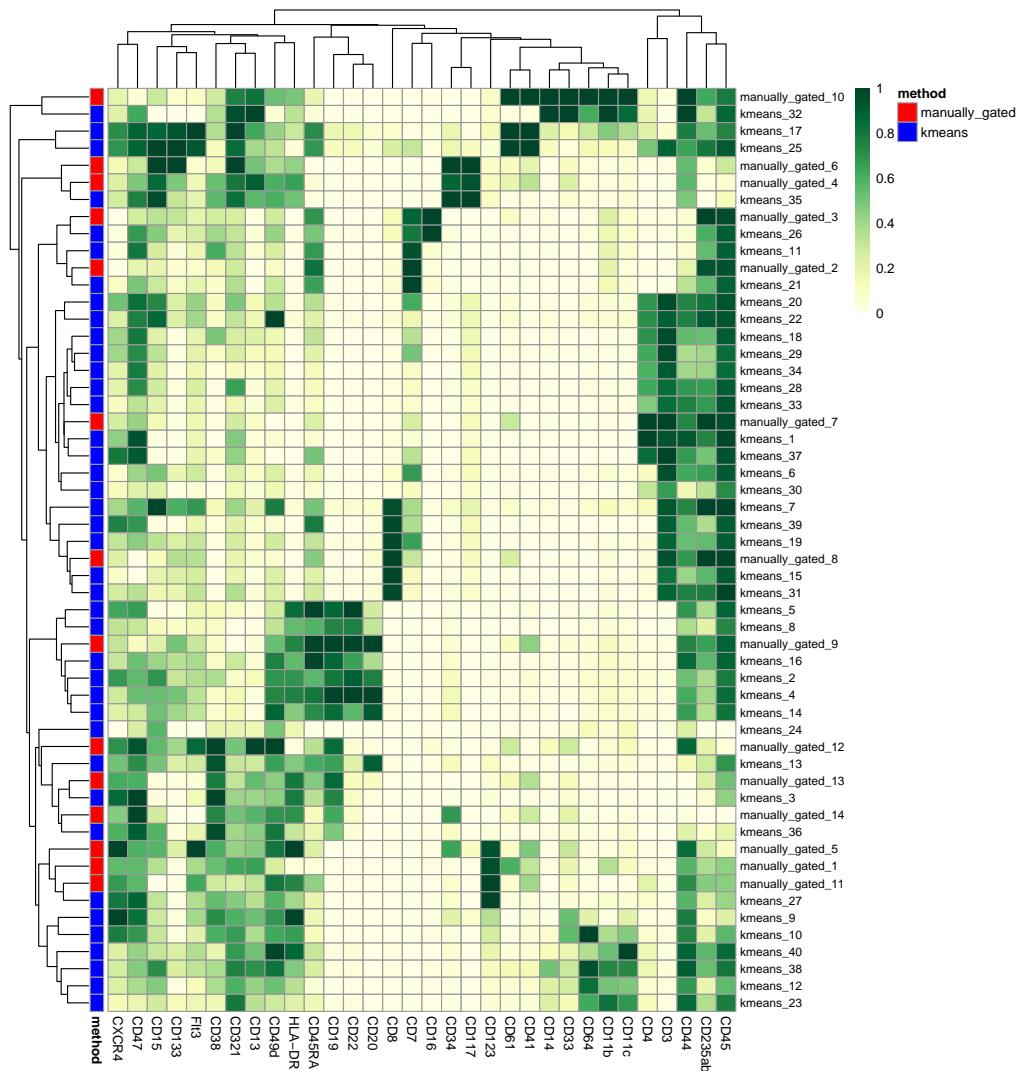
Supporting Information Figure S28. Median expression profiles of detected clusters and reference populations, FlowsOM, data set Levine_32dim. See Figure 2 (main paper) for details. (This figure is included as Figure 2 in the main paper.)



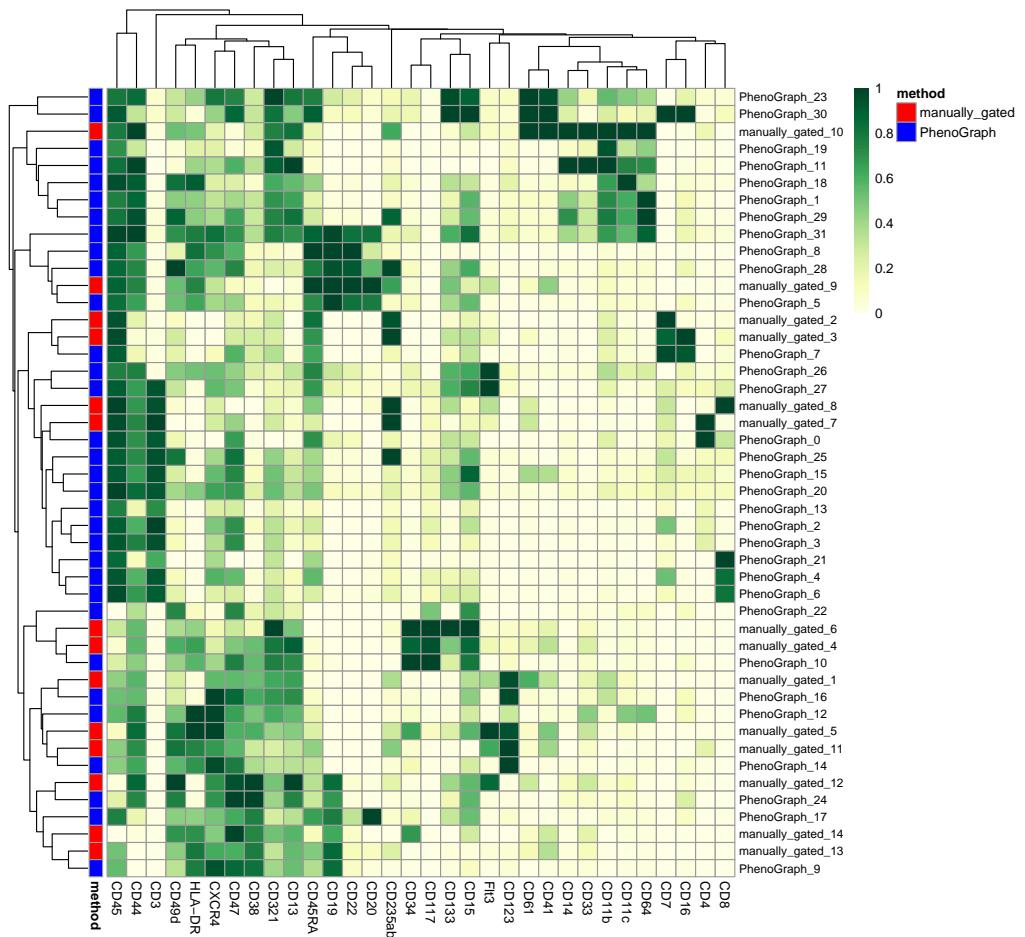
Supporting Information Figure S29. Median expression profiles of detected clusters and reference populations, FlowSOM_pre, data set Levine_32dim. See Figure 2 (main paper) for details.



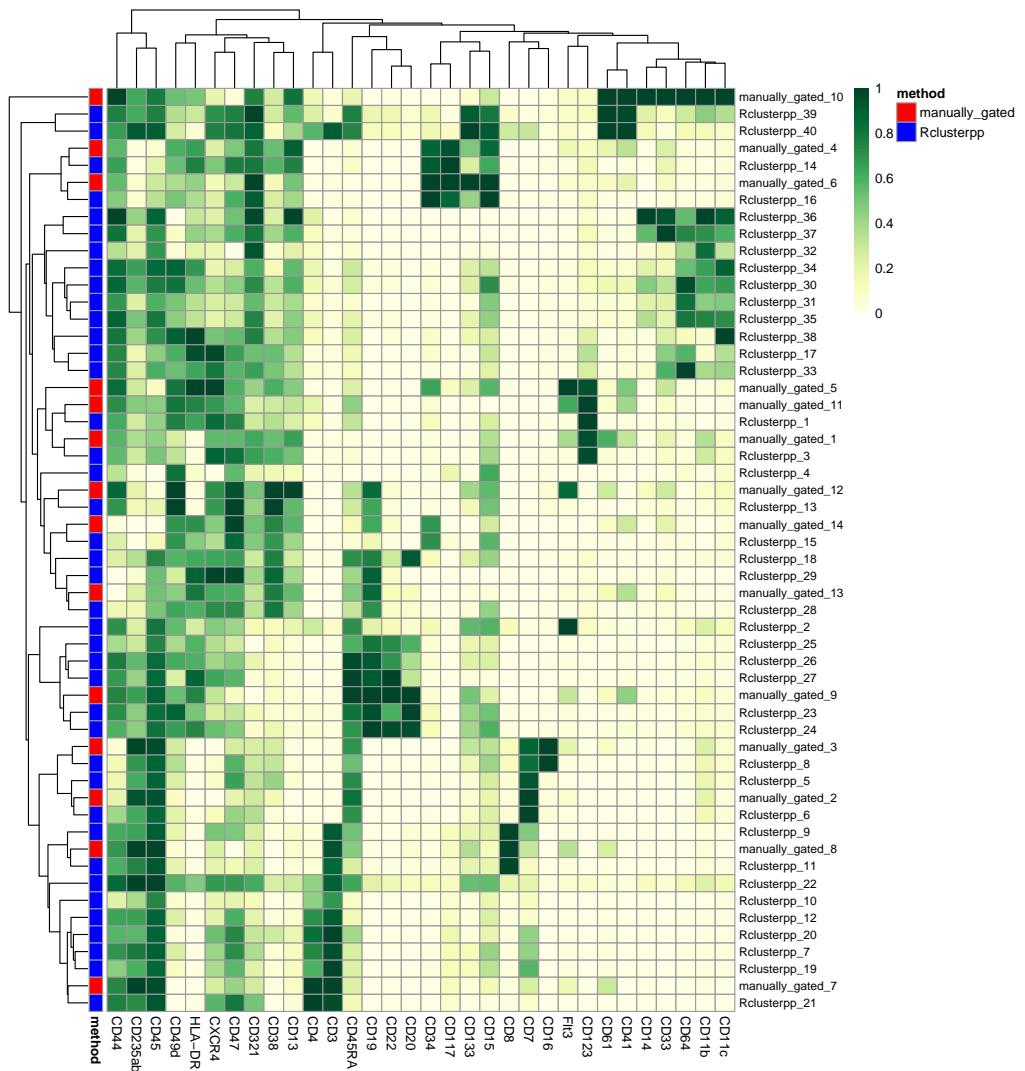
Supporting Information Figure S30. Median expression profiles of detected clusters and reference populations, immunoClust, data set Levine_32dim. See Figure 2 (main paper) for details.



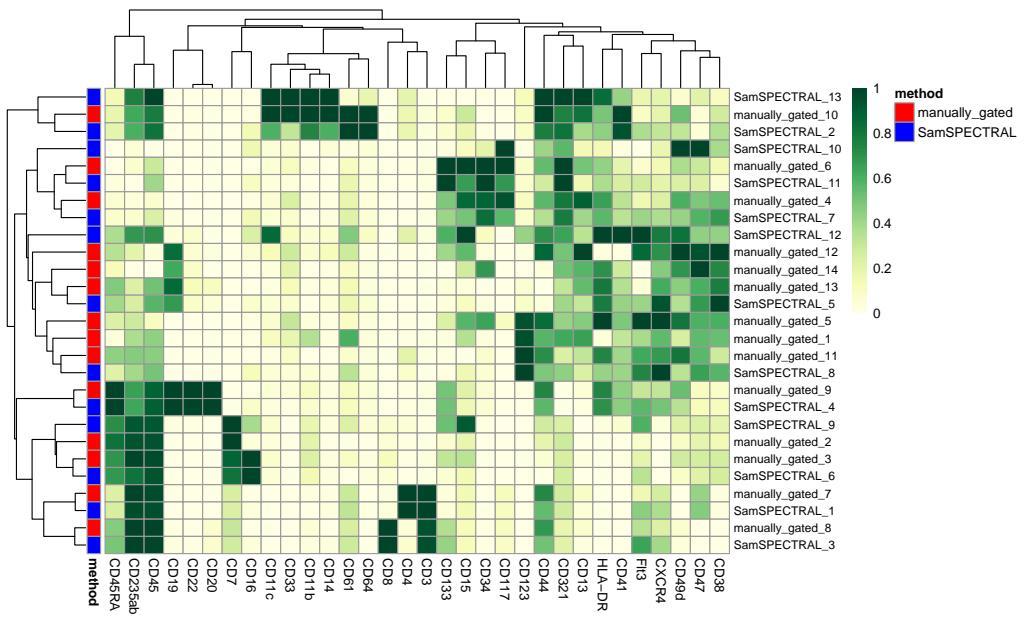
Supporting Information Figure S31. Median expression profiles of detected clusters and reference populations, k-means, data set Levine_32dim. See Figure 2 (main paper) for details.



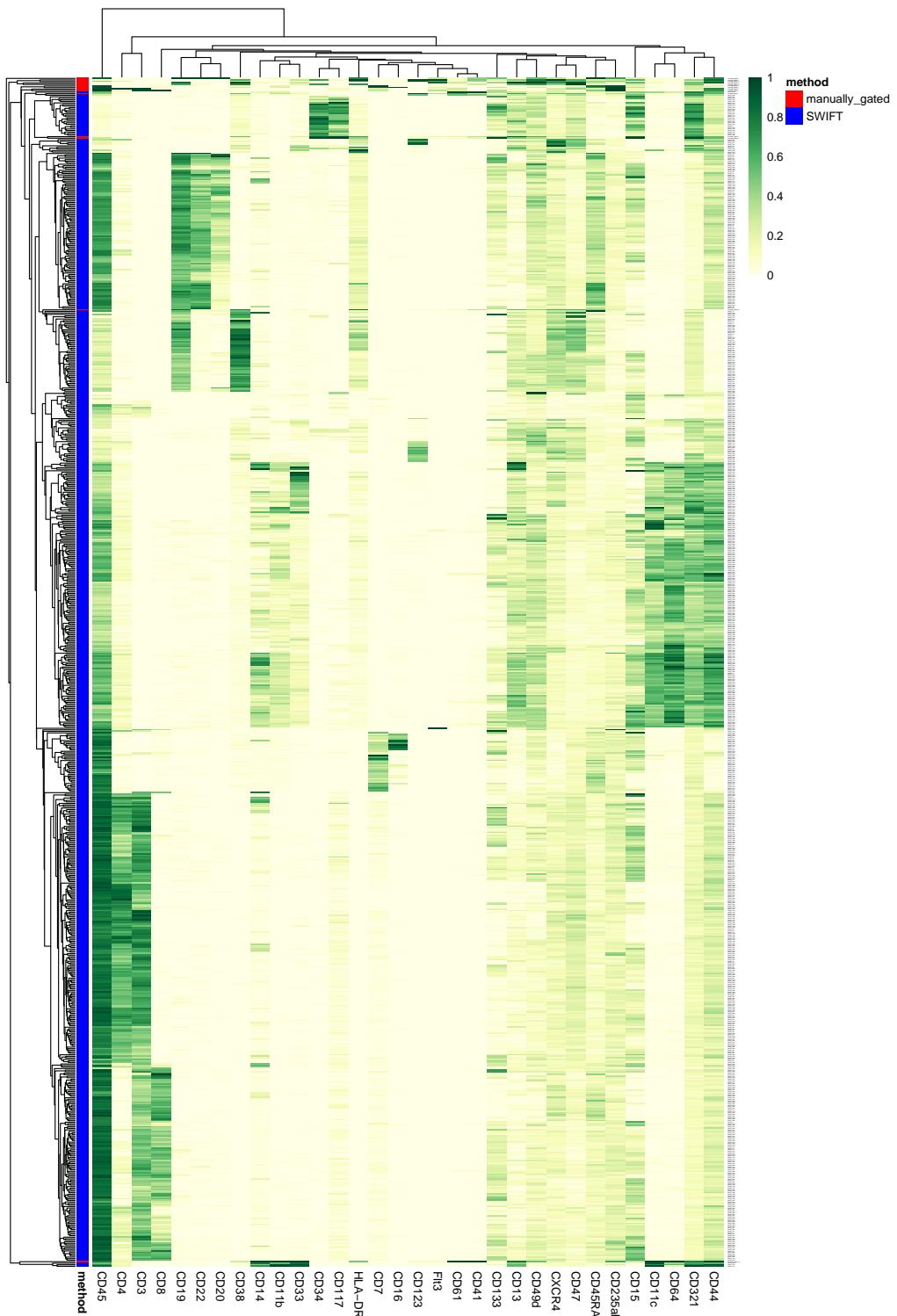
Supporting Information Figure S32. Median expression profiles of detected clusters and reference populations, PhenoGraph, data set Levine_32dim. See Figure 2 (main paper) for details.



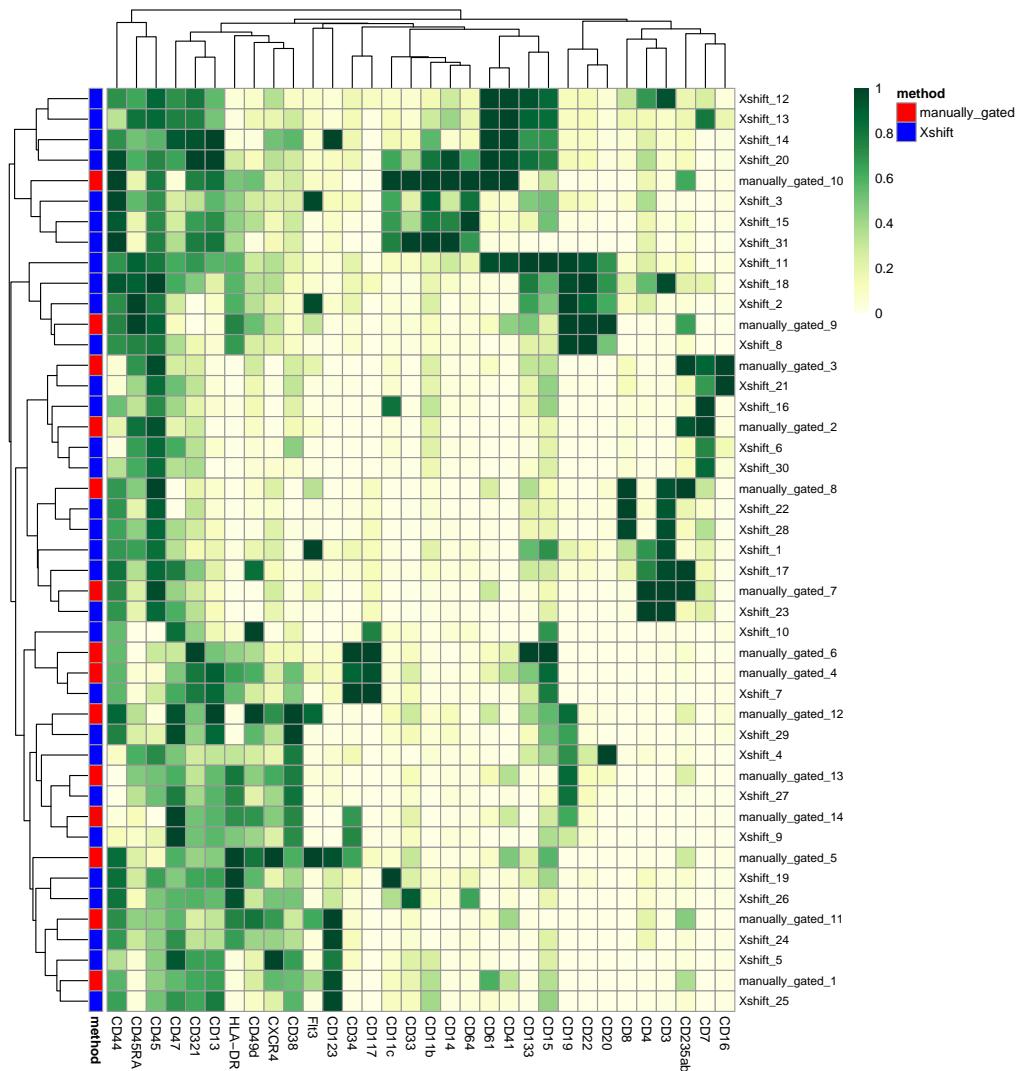
Supporting Information Figure S33. Median expression profiles of detected clusters and reference populations, Rclusterpp, data set Levine_32dim. See Figure 2 (main paper) for details.



Supporting Information Figure S34. Median expression profiles of detected clusters and reference populations, SamSPECTRAL, data set Levine_32dim. See Figure 2 (main paper) for details.



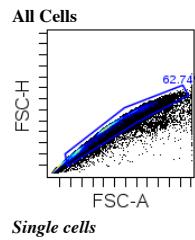
Supporting Information Figure S35. Median expression profiles of detected clusters and reference populations, SWIFT, data set Levine_32dim. See Figure 2 (main paper) for details.



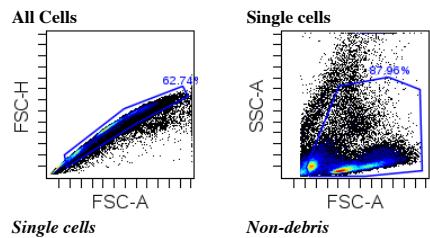
Supporting Information Figure S36. Median expression profiles of detected clusters and reference populations, X-shift, data set Levine.32dim. See Figure 2 (main paper) for details.

Supp. Fig. S37: Gating diagrams for data set Nilsson_rare

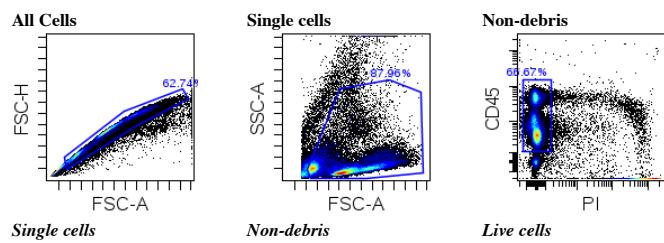
Single cells



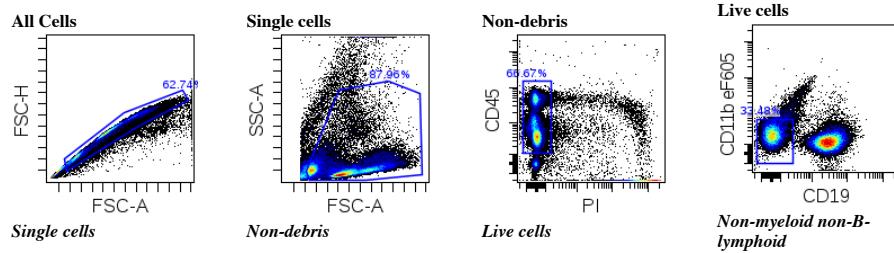
Non-debris



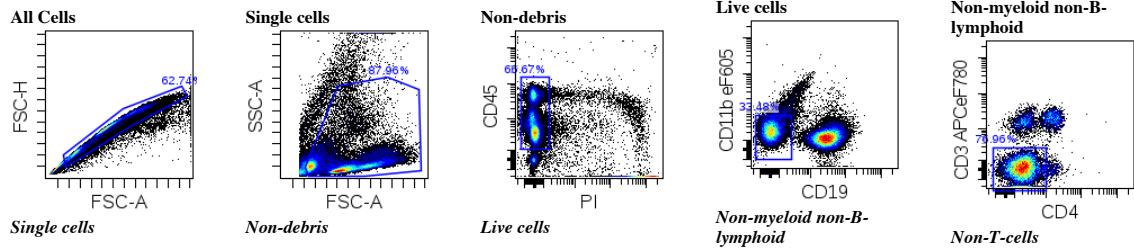
Live cells



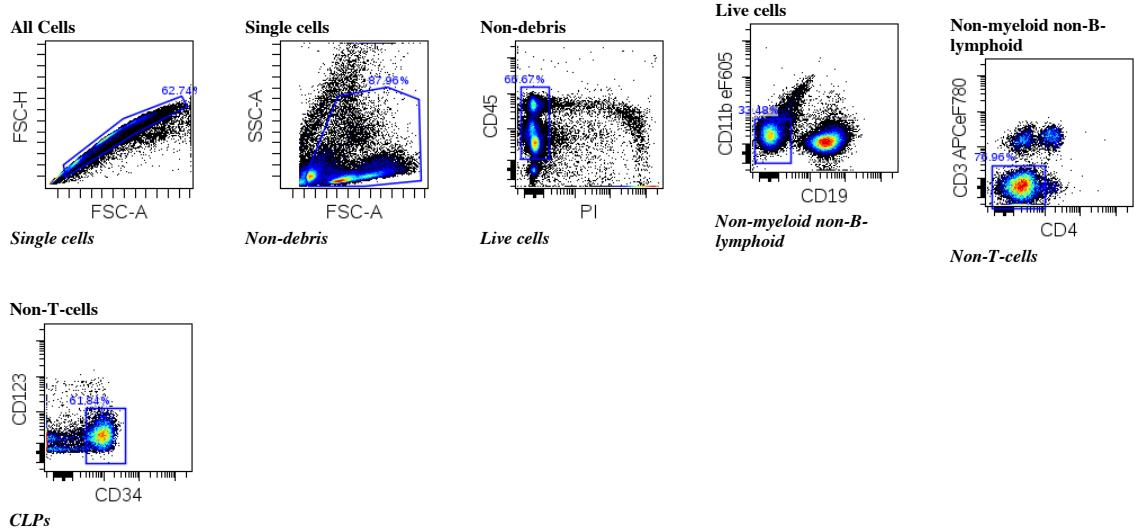
Non-myeloid non-B-lymphoid



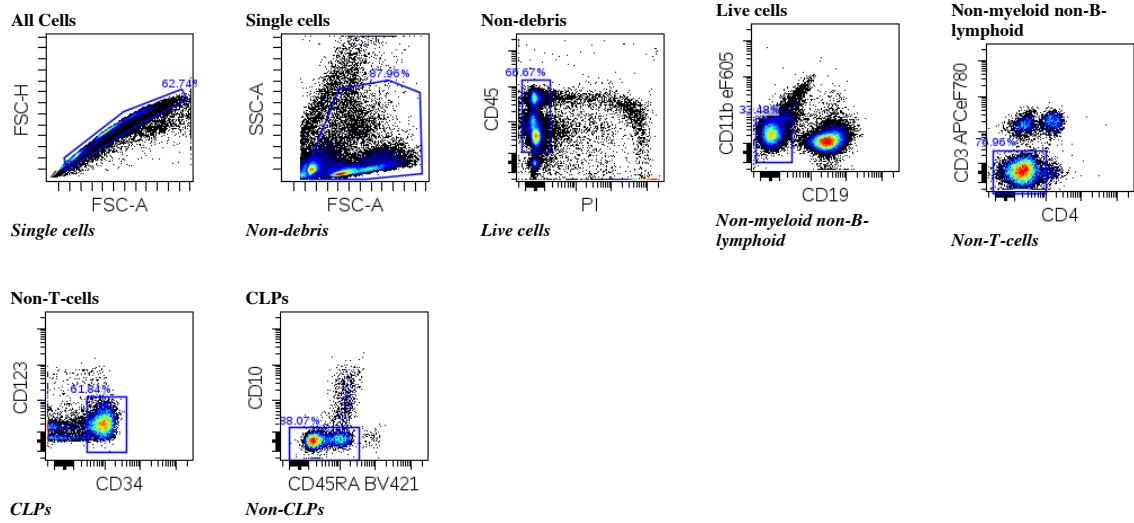
Non-T-cells



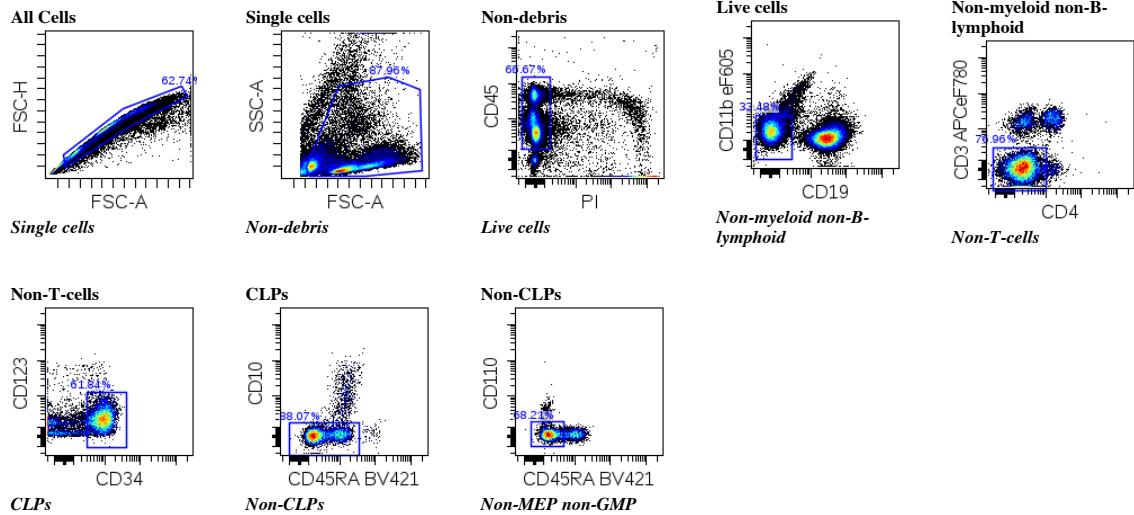
CLPs



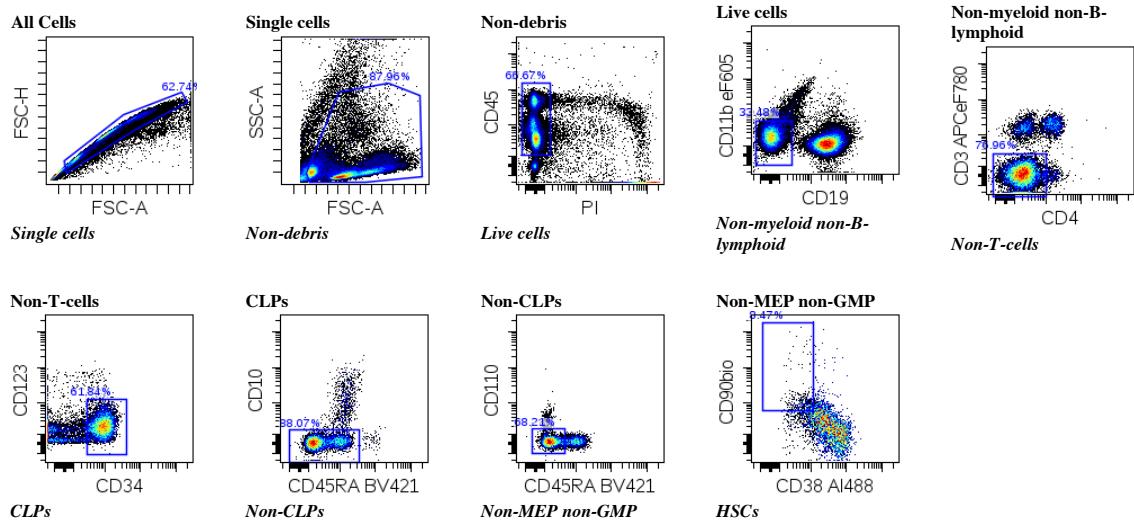
Non-CLPs



Non-MEP non-GMP

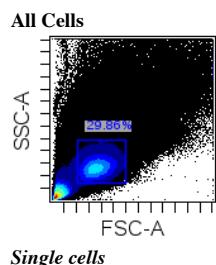


HSCs

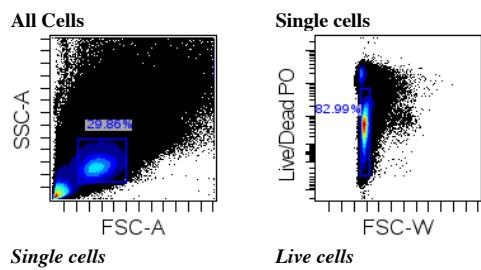


Supp. Fig. S38: Gating diagrams for data set Mosmann_rare

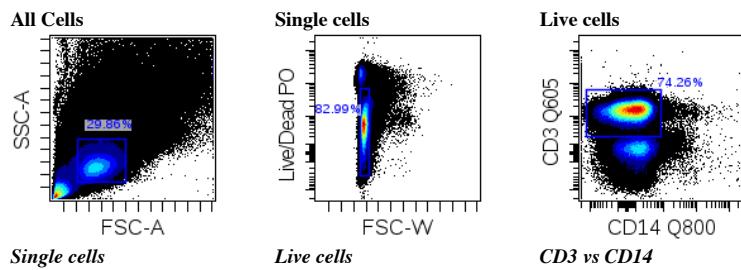
Single cells



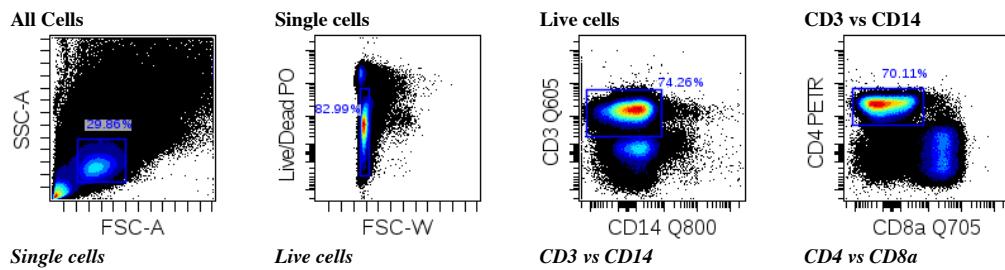
Live cells



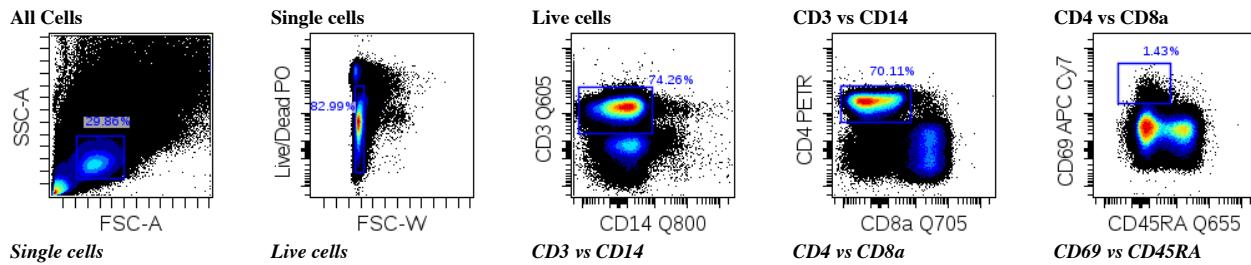
CD3 vs CD14



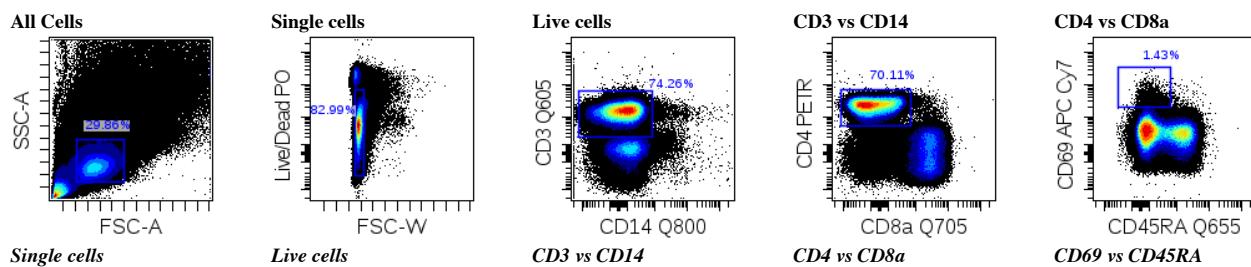
CD4 vs CD8a



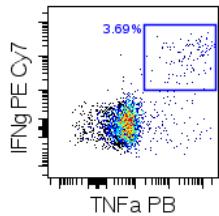
CD69 vs CD45RA



IFNg vs TNFa



CD69 vs CD45RA



IFNg vs TNFa