Finding stable clusterings of single-cell RNA-seq data

Victor Klebanoff
vfklebanoff@live.com

Abstract

Run a UMI count matrix through a clustering pipeline to obtain n cell clusters. Suppose that counts from the same experiment for an equal number of additional cells become available. Would including them change the results? Form the matrix containing both sets of counts, process it to obtain n clusters, restrict this (second) clustering to the initial cells and compare it with the initial clustering. If the clusterings are not consistent, conclude that the initial clustering is unstable.

Although this scenario is unrealistic, it is practical to reverse the perspective: given a clustering, process samples of half of the cells. If their clusters are consistent with those of the full set of cells restricted to the samples, conclude that the clustering is stable.

We use divisive hierarchical spectral clustering and describe a possibly novel mapping of the tree it produces to a set of nested clusterings.

Positive affinities are defined for points (representing cells in Euclidean space) that are k-nearest neighbors (k is an input parameter). The affinity equals the inverse of the distance between the points. Ng, Jordan, and Weiss' algorithm divides a set of points into two clusters. The normalized cut measures the clusters' separation. Recursion generates a hierarchy of clusters.

Viewing clusters as nodes of a tree, set the length of the branch between a node and each of its daughters to the normalized cut. Nodes' distances from the root define the mapping of the tree to nested clusterings. For four large data sets, this gave clusterings compatible with published results.

Analysis is performed for all cells and for multiple pairs of complementary samples of cells. For a given number of clusters, each sample's clustering and clusters are compared with those of the full data set (restricted to the sample). This provides measures of the stability of the clustering and its clusters. For two of the large data sets, the clusterings compatible with published results were judged to be stable.

Keywords: Gene expression, Single-cell RNA-seq, UMI counts, Outliers, Clustering, Stability, Replicability, Batch correction

1 Introduction

Although the stability – or replicability – of clusterings of gene expression data has been of interest for at least twenty-five years, there is evidently no consensus on how to find stable clusterings of single-cell RNA-seq data presented as UMI counts.

Early work using sampling to evaluate the effect of data variation on clusterings includes that of Levine and Domany [1], Ben-Hur et al. [2], Dudoit and Fridlyand [3], Lange et al. [4], and Tibshirani and Walther [5]. Hennig [6] discussed evaluating stability by using samples, as well as by replacing points with noise or by jittering. He observed that "stability ... is strongly dependent on the data set. In the same clustering, some clusters may be very stable and others may be extremely unstable."

More recently, Lun [7] discussed bootstrapping scRNA-seq data, Peyvandipour et al. [8] and Tang et al. [9] studied cluster stability for single-cell data with the objective of identifying (novel) cell types, and Patterson-Cross et al. [10] proposed a framework to evaluate the influence on clustering stability of parameters input to an analytical pipeline.

A simple question motivates this work: *If data for twice as many cells were available, would clustering results change?* While this is unknowable, the problem can be approached by reversing the question: *Would using half of the cells give consistent clusterings?* Informally: cluster all of the cells, then randomly divide the set of cells in two, and cluster each subset. Restrict the clustering of all cells to each subset and compare it with the subsets' clusterings. Repeat for multiple samples; if agreement is good for enough samples, the clustering may be considered stable.

We construct a pipeline that takes a UMI count matrix as input and produces clusterings of a range of *sizes* (number of clusters). Next, clusterings are generated for several samples of cells. Comparing the clusterings of the samples with the clusterings of the full set of cells gives stability estimates. Clusterings are compared using what Meilă calls the *misclassification error distance* (MED) [11]; Von Luxburg calls it the *minimal matching distance* [12]. Its distribution across samples characterizes a clustering's stability. For each *cluster* and *sample* we define a membership-based *cluster misclassification error rate* (CMER). Its distribution across samples characterizes the cluster's stability.

We considered, but do not use here, a *misclassification error rate for each cell*. In some cases, exploratory analysis found recalcitrant cells – cells that were often misclassified in clusterings of different sizes. Although we did not find an efficient way to identify these before clustering, the idea may merit further study as a way to identify outlier cells to exclude from downstream analysis.

For gene expression data, differential expression analysis may provide independent evidence of stability. If genes' expression levels between pairs of clusters found to be stable by our method differ consistently across samples, it could corroborate our results. Comparing samples may exempt this from Zhang's critique [13] of performing clustering and differential expression with the same data.

2 Methods

2.1 Overview

Suppose that the UMI count matrix $\boldsymbol{U}$ for a set of cells $\boldsymbol{C}$ is run through a pipeline to produce a clustering of size n (alternatively, an *n-clustering*). Prepare two *complementary samples* of cells: randomly assign half to the subset $\boldsymbol{C_1}$ and the balance to $\boldsymbol{C_2}$. Prepare the corresponding count sub-matrices $\boldsymbol{U_1}$ and $\boldsymbol{U_2}$. Run $\boldsymbol{U_1}$ through the pipeline to obtain an n-clustering. If the clustering of $\boldsymbol{C}$ restricted to $\boldsymbol{C_1}$ is close to the clustering of $\boldsymbol{C_1}$ alone, we might conclude that *if data for twice as many cells were available* – i.e. combining $\boldsymbol{C_2}$ and $\boldsymbol{C_1}$ – *the clustering results would not change*. Since the sample labels are arbitrary, we also require that processing $\boldsymbol{U_2}$ give a clustering consistent with the restriction of the clustering of $\boldsymbol{C}$ to $\boldsymbol{C_2}$.

This is done with multiple samples $\boldsymbol{C_s}$ using the counts $\boldsymbol{U_s}$ to cluster $\boldsymbol{C_s}$. Calculate the MED between the clustering of $\boldsymbol{C_s}$ and the restriction of the clustering of $\boldsymbol{C}$ to $\boldsymbol{C_s}$ for each s. MED is normalized to adjust for the chance grouping of cells. This is analogous to the calculation of the adjusted Rand index. If the values of MED are sufficiently small with enough samples, consider the clustering stable. In this article, we propose calling a clustering stable if the 90th percentile of normalized MED is less than or equal to 0.10. Continue by calculating each cluster's CMER with all samples. Like the MED, CMER is normalized. If the values of CMER are sufficiently small with enough samples, consider the cluster stable. Specifically, we propose calling a cluster stable if the 90th percentile of normalized CMER is less than or equal to 0.50: with at least 90% of the samples, fewer than half of the cells in the cluster are misclassified.

Because – as noted by Hennig (quoted in our Introduction) – a clustering may include clusters that are very stable and others that are extremely unstable, it is necessary to decide when unstable clusters render a clustering unsuitable for downstream analysis. We arbitrarily consider stable clusterings *admissible* for further consideration if their unstable clusters have fewer than 500 cells.

If more than one admissible clustering is found for a data set, we generally review results for the one with the most clusters.

2.2 Data sets

We studied seven publicly available data sets (Table 1). The Zhengmixeq data were prepared for a study that evaluated clustering algorithms' ability of recovering known cell subpopulations [14]. The PBMC data were studied by Zheng et al. in their seminal paper. The other four data sets have recently been used to evaluate proposed analytical methods. Lause et al. [15] studied the retinal data (Fig S3 in Additional file 1). Grabski et al. [16] studied the lung data. Nicol and Miller [17] studied the breast cancer data; they analyzed the monocytes in a preliminary version of their paper.

We performed batch correction at the patient level for the lung and breast cancer data.

Data set	source	cells	genes	reference
Zhengmix4eq	Bioconductor	3,994	15,568	DuoClustering2018 [18]
Zhengmix8eq	(same)	3,994	15,716	(same)
68k PBMC	10x Genomics	68,579	32,738	Zheng et al. [19]
CD14 Monocytes	10x Genomics	2,612	32,738	used by Nicol and Miller [20]
25k retinal	GSE63472	24,769	22,292	Macosko et al. [21]
65k lung	synapse.org	65,662	26,485	Travaglini et al. [22]
100k breast cancer	GSE176078	100,064	29,733	Wu et al. [23]

Table 1 Data sets used

2.3 Filter and transform counts

The first step in the pipeline is preliminary filtering. We follow the model of the Seurat Guided Clustering Tutorial [24] – restricting to genes with nonzero counts on at least 50 cells and, for two of the three data sets that were input through Seurat [25] (PBMC and monocytes), excluding cells with high count contributions from mitochondrial genes.

Samples of cells $\boldsymbol{C_s}$ and the associated count matrices $\boldsymbol{U_s}$ are prepared. Twenty pairs of complementary samples are used, 40 samples in all, to strike a balance between computing time and having sufficient data for meaningful results.

The variability of each gene g is calculated with all counts $\boldsymbol{U}$ – and with each sample's counts $\boldsymbol{U_s}$ – as the *sum of squares (SSQ) of its Pearson residuals (PR) using a Poisson model*, denoted S_g for the full set and $S_g(s)$ for sample s. These computations are efficiently performed with sparse arrays. The Pearson residuals matrices are not required. The calculations apply a formula derived in Appendix A of our 2023 bioRxiv posting [26] which is included for convenience as Appendix A of this article.

Genes are retained for downstream analysis if they are highly variable – arbitrarily defined as being among the top 2,000 – for the full set of cells and every sample. For the data sets studied, this retained between 565 and 1,704 genes, referred to as *analysis genes*.

To compare genes' variability across samples or between data sets, it is convenient to normalize S_g by dividing by the number of cells in the count matrix. This yields the mean SSQ of PR, denoted M_g.

The Pearson residuals matrix is explicitly constructed for the analysis genes. It is viewed as a low rank matrix perturbed by noise. To estimate the rank of the unperturbed matrix, Erichson's optht program [27] is used; it implements Gavish and Donoho's algorithm [28]. For the data sets studied, the ranks of the Pearson residuals range between 11 and 434. The singular value decomposition produces the low rank *Euclidean representation* of the residuals. By *the distance between cells* we mean the distance between the corresponding points in the Euclidean representation. By the *diameter* of the set of cells we mean the maximum distance between points in the Euclidean representation.

2.4 Exclude Euclidean outliers; cluster cells

To cluster points in the Euclidean representation, the Leiden algorithm was initially considered due to its use in Seurat. This presents difficulties because of the need to vary the resolution parameter to obtain all clusterings of a range of sizes. Suppose, for simplicity, that resolution equal to 2 gives a 2-clustering

and resolution equal to 10 gives a 10-clustering. Resolution values between 2 and 10 must be evaluated systematically to obtain all clusterings of sizes 3 through 9. In some cases, the number of clusters failed to increase monotonically with resolution. As a result, some clusterings were not found for some samples. Because computations evaluated resolution values that yielded clusterings of sizes previously found, run times could be large. Worse, few stable clusterings were found.

We explored spectral clustering and developed a divisive hierarchical scheme. Although an exhaustive study was not performed, it appears that the hierarchical approach is more likely to deliver stable clusterings than an alternative that directly outputs clusterings of each size.

Ng, Jordan, and Weiss' [29] algorithm is used, with a modification: the affinity between two points is defined not by a Gaussian function of the distance between them but as the inverse of the distance. (In Section 2.8 of [30], Verma and Meilă define the distance between points as the inverse of their affinity.) Nonzero affinities are defined only for points that are k-nearest neighbors (kNN). In the language of Section 14.5.3 *Spectral Clustering* of The Elements of Statistical Learning [31] the *mutual K-nearest-neighbor graph* describes the linkages between points. The affinity matrix is sparse – keeping computations affordable – at the cost of arbitrarily specifying the number of neighbors to be included. While we cannot offer a principled basis for specifying a value of this parameter, we have obtained credible results by restricting to 64th-nearest neighbors. We have not performed extensive comparisons, but in a few cases using 256th-nearest neighbors did not improve results.

Meilă recommends that outliers be removed before performing spectral clustering [32]. This led us to consider the distribution of kNN distances for points in the Euclidean representation.

These results may be of general interest, with implications for other clustering schemes. Illustrating with the PBMC data set (for details see Section 3.4 - Variation of kNN distance in the Euclidean representation) nearest neighbor distance ranges from 1.3 to 294 (Table 5). If an algorithm relies only on the kNN graph – otherwise ignoring the distance between points – two points 200 units apart are treated no differently than a pair of points separated by 2 units. To avoid this, if one of the points is a k-nearest neighbor of the other for a specified value of k, the affinity is defined as the inverse of the distance between them. For points 200 units apart, the affinity is 0.005; for points two units apart, it equals 0.5.

Returning to Meilă's suggestion, we propose excluding not only nearest neighbor outliers, but also small sets of isolated points. For example, consider a set of points $\{P_1, \ldots, P_{33}\}$ representing 33 cells and suppose that the distance between each pair of points is less than one, but that the distances to points not in the set are much larger. (Hence, the 32nd-nearest neighbor of each point P_i belongs to the set but its 64th-nearest neighbor does not.) The 33 points constitute an isolated set; its members might be excluded as outliers based on criteria proposed below. This poses questions that may require further study. Some of these isolated sets of points may represent clusters of biological interest. The challenge is to find a balance between retaining useful data and excluding outliers. Here, with the objective of identifying stable clusterings, we may err on the side of excluding too many cells.

To identify outliers, the distances between all pairs of cells are calculated. (Admittedly, this is slow for tens of thousands of points in high dimensions.) Ranking the distances from a point to all others yields the distances to its to k-nearest neighbors. We restrict to $k = 1, 2, 4, \ldots, k_{max}$ – arbitrarily setting $k_{max} = 64$. Next, the mean and standard deviation of the kNN distances are calculated for each k. The sum of the mean plus three times the standard deviation is calculated as the cutoff for defining outliers. If the distance from a point to its kNN is larger, the point is excluded from downstream analysis as a *Euclidean outlier*. The points remaining in the Euclidean representation correspond to cells to be clustered. The set's intersections with the samples $\boldsymbol{C_s}$ will also be clustered.

Note that data reduction for each sample of cells $\boldsymbol{C_s}$ uses results found for the full set of cells: the same analysis genes are used to compute Pearson residuals, which are reduced to the rank calculated by optht.

The affinity matrices required for spectral clustering are constructed for the full set of points and for each sample. If two points are 64th-NN or closer, the affinity is set to the inverse of the Euclidean distance between them. Otherwise the affinity equals zero. Although the property of being kNN is not symmetric (if P_2 is the nearest neighbor of P_1, P_1 may not be the nearest neighbor of P_2), affinity matrices are constructed to be symmetric.

For each set of points to be clustered, divisive hierarchical spectral clustering produces a tree. The maximum tree depth and minimum node (cluster) size are specified as stopping conditions. For the three small data sets, maximum tree depth was set to 6 and the minimum cluster size to 50. For the four large data sets, the maximum tree depth was set to 10. For the retinal data set (the smallest of the four), the minimum cluster size was set to 100; for the others we used 200.

The process begins by splitting the set of points into two clusters. Each cluster is split in turn. The process continues until a stopping condition is satisfied: the maximum specified depth is reached or splitting would produce a cluster smaller than the minimum permitted size. This produces a tree for the set of all cells and one for each sample.

2.5 Map trees to sets of nested clusterings

The *normalized cut*, described for example by Meilă in [32] and von Luxburg in [33], is used to prioritize the split at a node. The smaller it is, the better the separation between the node's daughter clusters – and the higher the priority of the split. The length of each branch from a node to its daughters is set to the normalized cut.

After assigning lengths to all branches, each node's distance to the root is calculated and nodes are sorted by their distance from the root. Mapping the tree to a set of nested clusterings is straightforward: the root's daughters correspond to the 2-clustering. Descend to the pair of nodes next closest to the root. They split the points at their parent (one of the root's daughters), yielding the 3-clustering. Repeat by descending the tree to the pair of nodes next closest to the root – and use them to obtain the 4-clustering. Continue until the clustering consisting of all terminal nodes is defined.

2.6 Calculate quantities that characterize stability

Given a value N_{max} we want to evaluate the stability of each clustering of size n for $2 \leq n \leq N_{max}$. We require clustering results for all 40 samples. The stopping conditions may prevent the generation of a clustering of size n for some sample. If this occurs, only smaller clusterings are analyzed. For the three small data sets N_{max} was set to 10. For the four large data sets, values larger than the published number of clusters were used: 25 for PBMC, 70 for the others. Given n, the clustering of $\boldsymbol{C_s}$ is compared with the restriction of the clustering of $\boldsymbol{C}$ to $\boldsymbol{C_s}$ by computing the misclassification error distance (MED). Following Lange et al. [4] (as pointed out by von Luxburg in [12]) the MED is normalized; it is divided by the mean of the MEDs calculated with data randomized by shuffling cluster labels.

We call attention to the fact that using MED to compare clusterings assigns consistent labels to the clusters of the full set of cells and to the clusters of each sample s: each cluster $\boldsymbol{c}$ of cells in $\boldsymbol{C}$ corresponds to a cluster $\boldsymbol{c}$ in each sample $\boldsymbol{C_s}$. Hence the cluster misclassification error rate (CMER) can be calculated easily for each cluster with each sample of cells $\boldsymbol{C_s}$:

- For cluster $\boldsymbol{c}$ of cells in $\boldsymbol{C}$, let $\boldsymbol{c_s}$ denote the intersection of the cells in $\boldsymbol{c}$ with $\boldsymbol{C_s}$
- Consider the cluster label assigned to a cell in $\boldsymbol{c_s}$ in the clustering of $\boldsymbol{C_s}$ (i.e. calculated with its cells alone)
- If it does not equal $\boldsymbol{c}$, the cell is misclassified in $\boldsymbol{C_s}$
- Define the cluster misclassification error rate of $\boldsymbol{c}$ as the fraction of cells in $\boldsymbol{c_s}$ that are misclassified
- Repeat the computations using data with shuffled cluster labels to get denominators for normalizing CMER

The confusion matrix displayed in Table 2 illustrates the calculations of *unnormalized* MED and CMER for a subset of the 68k PBMC data that is reviewed in Section 3.5 - Clustering results for individual data sets. (This is the 9-clustering produced in the third iteration.)

	0	1	2	3	4	5	6	7	8	Total
0	0	145	9	0	0	0	0	0	0	154
1	0	1,825	4	0	0	5	0	0	0	1,834
2	0	4	2,618	0	0	27	0	0	0	2,649
3	0	0	0	9,057	150	0	71	0	0	9,278
4	0	0	0	32	2,907	0	3	0	0	2,942
5	0	1	11	0	1	2,779	146	0	0	2,938
6	0	0	0	74	4	5	8,383	0	0	8,466
7	0	0	0	1	0	0	1	1,440	0	1,442
8	211	0	0	0	0	2	1	0	1,598	1,812
Total	211	1,975	2,642	9,164	3,062	2,818	8,605	1,440	1,598	31,515

Table 2 68k PBMC data (iteration 3): compare the 9-clusterings calculated with all cells and with sample 1

The table compares the cluster assignments calculated with all cells (row labels) with the assignments calculated with a sample containing 31,515 cells (column labels).

- Columns are sorted to maximize the sum of the diagonal. *This assigns the cluster labels in the sample.*
- The (unnormalized) MED equals the fraction of the total count that is off the diagonal, here equal to 0.029.
- In the first row – counts in the sample for cells assigned to cluster 0 in the full set of data – the diagonal entry equals zero. All of these cells are misclassified in the sample; CMER=1.
- In the fifth row – counts in the sample for cells assigned to cluster 4 in the full set of data – the diagonal entry equals 2,907. 35 cells are misclassified in the sample; CMER=0.01.
- In the sixth row – counts in the sample for cells assigned to cluster 5 in the full set of data – the diagonal entry equals 2,779. 159 cells are misclassified in the sample; CMER=0.05.

	0	1	2	3	4	5	6	7	8	Total
0	0	128	5	0	0	0	0	0	0	133
1	0	1,809	11	0	0	1	0	0	0	1,821
2	0	5	2,617	0	0	1	0	0	0	2,623
3	0	0	0	9,195	0	0	183	0	0	9,378
4	0	0	0	2,931	0	0	3	0	0	2,934
5	1,259	1	118	0	0	1,636	5	0	0	3,019
6	115	0	0	26	0	21	8,234	0	0	8,396
7	0	1	0	1	0	0	1	1,438	0	1,441
8	0	0	0	0	793	1	1	0	1,012	1,807
Total	1,374	1,944	2,751	12,153	793	1,660	8,427	1,438	1,012	31,552

Table 3 68k PBMC data (iteration 3): compare the 9-clusterings calculated with all cells and with sample 32

The confusion matrix in Table 3 – for a sample containing 31,552 cells – shows greater differences.

- MED equals 0.178 – six times larger.
- As in Table 2, CMER=1 for cluster 0 (first row).
- For cluster 4, every cell is misclassified; CMER=1 (in Table 2, CMER=0.01).
- For cluster 5, CMER=0.46 (in Table 2, CMER=0.05).

Checking the confusion matrices for the other 38 samples reveals that cluster 0 is merged with cluster 1 in every sample. For emphasis:

- Cluster 0 has no counterpart in any sample
- It is *totally unstable*; CMER=1 with every sample

These patterns in Table 3 appear in the confusion matrices for other samples.

- Cluster 4 is merged with cluster 3 in 14 of the 40 samples.
- A large segment of cluster 5 splits to cluster 0 in 20 samples.

- A large segment of cluster 8 splits to cluster 4 in 6 samples.

In [7] Lun proposed comparing pairs of clusters to evaluate pairwise instability – to identify clusters that are *not stable with respect to each other*, that are likely to be merged in a random sample. This is the behavior seen here for the merging of cluster 0 with 1 and of cluster 4 with 3. The patterns of cluster splits may have similar interest.

Before leaving these examples, we note that four of the six clusterings of the large data sets reviewed in Section 3.5 - Clustering results for individual data sets include at least one totally unstable cluster. They exemplify Hennig's previously referenced observation that a clustering may include clusters that are very stable and others that are extremely unstable.

The assignment of cluster labels in each sample $\boldsymbol{C_s}$ that are consistent with the full set of cells provides the opportunity to use differential expression to compare clusters judged stable based on CMER.

- Suppose that for sample s the differential expression of an analysis gene g between two stable clusters $\boldsymbol{c_1}$ and $\boldsymbol{c_2}$ is measured as the AUC or T-statistic comparing the Pearson residuals of g (calculated from the count sample $\boldsymbol{U_s}$) in the two clusters.
- If g is differentially expressed on the two clusters, then the statistics should have extreme values: AUC should be close to zero or one; the absolute value of T should be large.
- If the gene's statistics are consistent and have extreme values across all samples, we would conclude that it is differentially expressed, suggesting that the clusters may be biologically meaningful.

2.7 Perform single-factor batch correction, if appropriate

For batch correction, the formula for S_g is modified, corresponding to Nicol and Miller's calculating a gene-level intercept for each batch ([17] Supplementary material):

- For each batch b, independently calculate the SSQ of each gene's Pearson residuals. Denote this as S_g^b for the set of all cells and as $S_g^b(s)$ for sample s
- Define $S_g = \sum_b S_g^b$ for the full set of cells and $S_g(s) = \sum_b S_g^b(s)$ for sample s

Calculate Pearson residuals for the cells in each batch. Concatenate the matrices for downstream analysis.

2.8 Identify and exclude cell and gene outliers for iterative analysis

Three sets of analyses were performed for each of the four large data sets. Initially, the count matrices were analyzed as outlined above in Sections 2.3 - Filter and transform counts, 2.4 - Exclude Euclidean outliers; cluster cells, 2.5 - Map trees to sets of nested clusterings, and 2.6 - Calculate quantities that characterize stability.

For the second iteration, outliers were excluded from the count matrix. The algorithms are motivated by a toy example in our 2023 bioRxiv posting [26] which illustrated the impact of a single outlier cell on the value of S_g. In that example, the set of cells is split into two samples. The value of S_g for the subset including the outlier is nine times larger than the value for the complementary sample.

Although we have not explored toy examples for the effect of outliers on clustering, we believe they are impediments to the existence of stable clusterings – that if a count matrix admits a stable clustering, a necessary condition is that the distributions of UMI counts in all sufficiently large samples agree with one another in a sense yet to be discovered. We have not compared the distributions of complementary samples, but have considered properties that a count matrix should have if the distributions of its samples agree. Two, both involving $S_g(s)$, were used to identify outliers:

1. Cells: If a cell's contribution to $S_g(s)$ is large in some sample – i.e. the cell is responsible for a large fraction of $S_g(s)$ – it is an outlier (because it cannot have a comparable impact on the complementary sample). The derivation in Appendix B shows that each cell's fractional contribution to S_g can be easily calculated for the Poisson model. Each cell's maximum fractional contribution to $S_g(s)$ is calculated across all genes and samples.

2. Genes: The toy example suggests that for each gene g, the values of $S_g(s)$ should be consistent across all samples s. To evaluate this, calculate the ratio $I_g = max_s S_g(s)/min_s S_g(s)$ for each gene. If it is close to one, values of $S_g(s)$ are consistent. If it is large, then g is an outlier.

After excluding cell and gene outliers, the second iteration proceeds as outlined in Sections 2.3 through 2.6.

For the third iteration, the counts input to the second iteration were subjected to additional filtering. Cells excluded as Euclidean outliers during that iteration were removed. This was followed by deleting outlier cells and genes as described in the previous paragraph. The resulting counts were analyzed per Sections 2.3 through 2.6.

3 Results

3.1 Impact of data reduction

Table 4 summarizes the impact of data reduction.

- Columns 1 and 2: the numbers of cells and genes retained after preliminary filtering. The Pearson residuals matrix has a column for each cell. Compared to Table 1, the number of genes is smaller because genes with nonzero counts on fewer than 50 cells were excluded. The number of cells is smaller for the Zhengmixeq data because some barcodes have duplicate values (see Section 3.5 - Clustering results for individual data sets). Cells with high count contributions from mitochondrial genes were excluded from the PBMC and monocyte data. Blood cells were excluded from the lung data.
- Column 3: the number of analysis genes – genes that are highly variable in every sample and the full data set. This is the number of rows of the Pearson residuals matrix.
- Column 4: the rank of the Pearson residuals matrix estimated by the optht program. This is the number of rows in the Euclidean representation matrix.
- Column 5: the number of cells retained for clustering after excluding Euclidean outliers.

	preliminary filtering		analysis	estimated	cells
Data set	cells	genes	genes	rank	retained
Zhengmix4eq	3,980	5,837	646	35	3,902
Zhengmix8eq	3,974	6,075	565	37	3,906
68k PBMC	68,286	12,515	652	48	67,792
CD14 monocytes	2,558	3,726	719	11	2,537
25k retinal	24,769	13,552	1,081	51	24,101
65k lung	60,993	17,470	1,659	305	60,114
100k breast cancer	100,064	21,354	1,704	434	98,681

Table 4 Impact of data reduction

3.2 Genes that are highly variable for all cells may not be highly variable for every sample

The plots in Figure 1 show the relation between the mean SSQ of Pearson residuals calculated with all cells (M_g) and the number of cells with nonzero counts for genes retained after preliminary filtering. Black points represent analysis genes retained for downstream analysis: S_g and $S_g(s)$ are among the 2,000 largest when calculated with all cells and with each sample s, respectively.

In particular, some genes that are highly variable in the full set of data (large vertical coordinates) are not highly variable in every sample (red points). They tend to have nonzero counts on fewer cells (small horizontal coordinates).

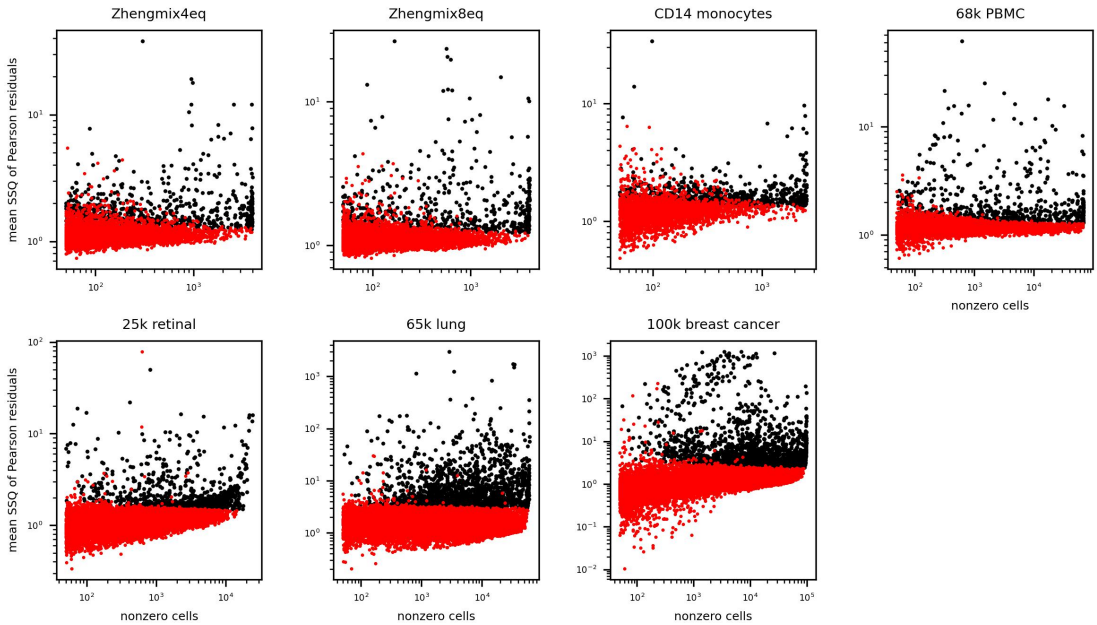

Figure 1 Filtered genes: M_g vs. number of cells with counts >0; analysis genes in black

3.3 Analytic estimates of the rank of the Pearson residuals

The ranks of the Pearson residuals matrices were estimated with the optht program. Four other programs were considered. Although all five gave reasonable results on toy problems (matrices of known low rank with added noise) each of the other four gave problematic results with at least one data set. Issues included:

- Wide variation between results depending on the user-selected algorithm (one program)
- Rank estimates differing by an order for magnitude for very similar input matrices (one program)
- Long run times (two programs)
- Failure to find a solution (one program)

Results are tabulated in column 4 of Table 4 and plotted in Figure 2 using the format of the screeplot program in the Bioconductor PCAtools package [34]. The estimated ranks of the Pearson residuals for the lung and breast cancer data (305 and 434, respectively) are larger than values we have seen in the literature. The maximum value plotted on the horizontal axis is the number of analysis genes – the number of rows in the Pearson residuals matrix (column 3 of Table 4).

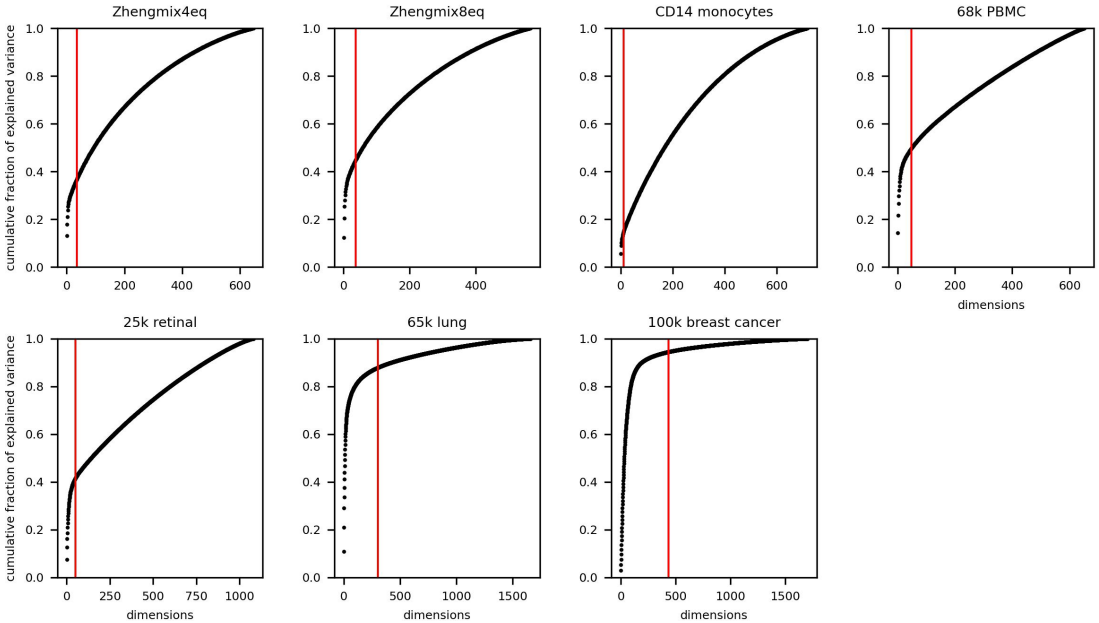

Figure 2 Scree plots: red vertical lines indicate the rank estimated by optht

3.4 Variation of kNN distance in the Euclidean representation

Our interest in the relation between kNN and Euclidean distance was motivated by Meilă's recommendation to exclude outliers before performing spectral clustering and by the work of Cooley et al. [35].

We use the PBMC and breast cancer data for illustration. The variation of kNN distance in the Euclidean representation of the PBMC data is summarized in Table 5. The first column contains statistics for the distance from a point to its nearest neighbor, which ranges from 1.3 to 294, with a mean of 5.0 and standard deviation of 7.4. Subsequent columns list statistics for 2nd-nearest neighbor distance, 4th-nearest, continuing to 64th-nearest, and finally the maximum distance between cells. The diameter of the set equals 823 – the bottom right entry.

	1	2	4	8	16	32	64	maximum
mean	5.0	5.4	5.8	6.1	6.6	7.1	7.7	707.9
stddev	7.4	8.0	8.7	9.7	10.7	11.8	13.4	6.3
min	1.3	1.6	1.8	2.0	2.1	2.3	2.4	507.1
10%	2.3	2.4	2.6	2.7	2.9	3.0	3.2	707.7
50%	3.8	4.0	4.2	4.5	4.7	5.0	5.3	708.2
75%	5.5	5.9	6.3	6.6	7.0	7.4	7.9	708.4
90%	8.2	8.8	9.4	10.0	10.7	11.5	12.4	708.7
95%	10.6	11.4	12.2	13.2	14.3	15.7	17.3	708.9
99%	18.2	20.0	22.0	24.4	27.2	30.6	35.2	711.3
max	294.3	384.4	389.0	433.6	461.9	507.1	551.0	822.6

Table 5 68k PBMC data: distributions of k-nearest neighbor distances

Clearly, kNN neighborhoods need not resemble neighborhoods defined with the Euclidean metric. For half of the cells, the distance to the nearest neighbor is less than 4 units – less than 0.5% of the diameter of the set of cells. However, there is a cell whose nearest neighbor is 75 times further away – 294 units distant, 36% of the diameter of the set of cells.

Cells with exceptionally distant k-nearest neighbors are identified as outliers to be excluded from downstream analysis. Outliers are defined as having values more than three standard deviations larger

than the mean. For nearest neighbors, this threshold equals 27.1; fewer than 1% of cells are outliers based on this criterion. Applying this to 2nd, 4th, ..., and 64th-nearest neighbors as well excludes a total of 494 cells, retaining 67,792. The distributions of k-nearest neighbors distances for the retained cells are summarized in Table 6. Excluding outliers reduces the range of nearest neighbor distances by an order of magnitude and the diameter of the set of cells by 80%.

	1	2	4	8	16	32	64	maximum
mean	4.6	4.9	5.2	5.5	5.9	6.3	6.8	129.3
stddev	2.7	3.0	3.2	3.5	3.8	4.2	4.8	3.5
min	1.3	1.6	1.8	2.0	2.1	2.3	2.4	103.9
10%	2.3	2.4	2.6	2.7	2.9	3.0	3.2	126.5
50%	3.7	4.0	4.2	4.4	4.7	5.0	5.3	130.0
75%	5.5	5.8	6.2	6.6	6.9	7.3	7.8	130.6
90%	8.0	8.6	9.1	9.7	10.4	11.2	12.1	131.2
95%	10.2	10.9	11.7	12.6	13.6	14.8	16.3	131.9
99%	15.1	16.2	17.5	19.1	21.1	23.4	26.7	136.0
max	28.2	31.6	34.2	37.6	40.6	44.6	59.7	163.9

Table 6 68k PBMC data: distributions of k-nearest neighbor distances after excluding Euclidean outliers

For the breast cancer data, the variation is greater. Before excluding Euclidean outliers, nearest neighbor distance varies by a factor of 580 (Table 7). Even after excluding 1.4% of the cells, the maximum is 50 times larger than the minimum (Table 8).

	1	2	4	8	16	32	64	maximum
mean	48.6	54.0	59.4	65.1	71.1	77.2	83.1	6,394.7
stddev	95.6	112.6	129.8	148.7	168.1	184.7	197.7	47.6
min	8.5	9.3	9.6	10.2	10.7	11.5	12.5	6,013.6
10%	18.1	19.6	20.8	21.8	22.9	24.0	25.1	6,390.1
50%	33.4	35.9	38.0	40.0	42.0	44.2	46.5	6,391.0
75%	50.6	54.8	58.6	62.1	65.7	69.7	73.8	6,392.2
90%	79.3	86.9	94.2	101.5	108.9	116.7	124.9	6,394.2
95%	107.9	120.1	133.8	148.2	163.7	180.5	196.9	6,397.2
99%	254.8	316.4	400.1	490.4	598.5	705.0	803.2	6,454.4
max	4,918.7	5,131.6	5,195.4	5,547.0	5,810.0	6,046.1	6,290.6	8,740.9

Table 7 100k breast cancer data: distributions of k-nearest neighbor distances

	1	2	4	8	16	32	64	maximum
mean	42.2	45.8	49.2	52.8	56.6	60.7	65.1	1,189.1
stddev	31.0	34.6	38.9	44.0	50.1	57.1	65.2	18.6
min	8.5	9.3	9.6	10.2	10.7	11.5	12.5	1,050.6
10%	18.0	19.5	20.7	21.8	22.8	23.9	25.1	1,181.6
50%	33.1	35.6	37.6	39.6	41.6	43.7	46.0	1,185.3
75%	49.6	53.5	57.1	60.6	64.1	67.9	71.9	1,189.8
90%	75.7	82.3	89.0	95.6	102.3	109.4	116.8	1,198.2
95%	98.4	108.4	118.4	129.2	141.3	153.6	166.2	1,207.7
99%	168.5	190.1	215.0	246.6	285.3	328.3	385.9	1,266.6
max	427.9	429.6	502.6	535.2	604.7	640.8	892.8	1,566.0

Table 8 100k breast cancer data: distributions of k-nearest neighbor distances after excluding Euclidean outliers

3.5 Clustering results for individual data sets

In Section 2.1 - Overview we propose considering a clustering stable if the 90th percentile of normalized MED is less than or equal to 0.10. A cluster is judged stable if the 90th percentile of normalized CMER is less than or equal to 0.50. We call a cluster *very stable* if the 90th percentile of CMER is less than or equal to 0.10 and *extremely stable* if it is less than or equal to 0.02. Additionally, we consider a stable clustering admissible for downstream analysis if its unstable clusters have fewer than 500 cells.

We begin with the three small data sets. For the Zhengmixeq data we are interested in (1) the relation between the ground truth labels and our method's clusterings and (2) the stability of specific clusterings and clusters. For the Zhengmix4eq data, agreement with ground truth labels is excellent; for Zhengmix8eq data less so, though typical of what we have found in publications. For the monocytes, our results suggest that there are no stable clusterings.

For each of the four large data sets, as outlined in Section 2.8 - Identify and exclude cell and gene outliers for iterative analysis, three sets of analyses were performed, progressively excluding outlier cells and genes. Stable clusterings were identified. We review a total of six clusterings:

- PBMC: two clusterings; an admissible 12-clustering and an unstable 9-clustering
- retinal: an admissible 11-clustering
- lung: two admissible clusterings; one with 19 clusters, the other with 16
- breast cancer: an unstable clustering compatible with published results

Zhengmix4eq

The data set contains counts for 15,568 genes and 3,994 cells. They represent four cell types; ground truth labels are provided with the data.

Seven barcodes appear twice; the corresponding 14 columns were dropped, retaining counts for 3,980 cells. Two EnsemblIDs have the same gene symbol (SRSF10). After excluding the EnsemblID with nonzero counts on fewer cells, and restricting to genes with nonzero counts on at least 50 cells, counts were retained for 5,837 genes and 3,980 cells. Cells were not screened for high mitochondrial DNA levels. 646 genes were found to be highly variable in the full data set and in all 40 samples.

The rank of the Pearson residuals matrix was estimated as 35 by optht. After mapping to a 35-dimensional Euclidean representation and excluding outliers, 3,902 cells were retained and clustered.

Figure 3 displays the distributions of normalized MED for the clusterings of sizes 2-10. There is one line per clustering. The 40 (vertically jittered) black dots in each line show the samples' MED. For each clustering, the blue vertical segment marks the median of the distribution. The 75th percentile marker is green; the 90th percentile marker is red. The clusterings of sizes 2-5 are stable; the 90th percentiles are less than or equal to 0.10. (For the clusterings of sizes 2-4, the plotted percentiles equal 0.00. Only the blue markers are visible because we favor the lower percentile markers.)

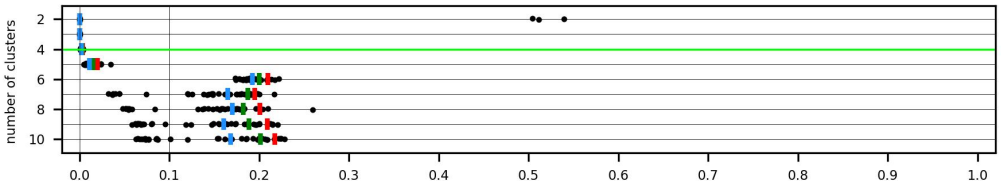

Figure 3 Zhengmix4eq data: distributions of normalized MED for clusterings of sizes 2-10

Because there are 4 ground truth labels, we review the 4-clustering; the green highlighted line displays its MED values.

Figure 4 displays the distributions of normalized CMER for each cluster. All four clusters are extremely stable; the largest value of CMER equals 0.017. The plot shows the median, 75th, and 90th percentiles of the CMER values for each cluster (indistinguishable for clusters 0, 2, and 3). In addition, the colored vertical lines that extend from the bottom to the top of the plot (all very close to 0 on the horizontal axis) indicate the clustering's percentiles – the ones marked in Figure 3.

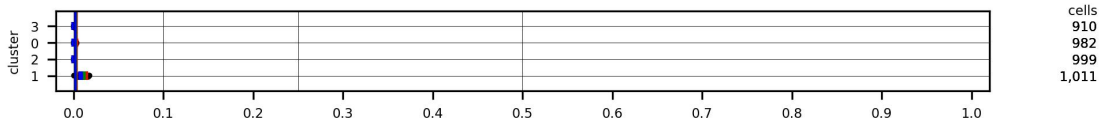

Figure 4 Zhengmix4eq data: distributions of normalized CMER for the 4-clustering

Table 9 compares the clusters with the ground truth labels.

ground truth	hierarchical cluster 0	1	2	3	Total
naive.cytotoxic	977	16	0	0	993
regulatory.t	5	986	0	0	991
b.cells	0	0	997	0	997
cd14.monocytes	0	9	2	910	921
Total	982	1,011	999	910	3,902

Table 9 Zhengmix4eq data: compare ground truth with the 4-clustering

Zhengmix8eq

The data set contains counts for 15,716 genes and 3,994 cells. They represent eight cell types or subtypes; ground truth labels are provided with the data.

Ten barcodes appear twice, the corresponding 20 columns were dropped, retaining counts for 3,974 cells. Cells were not screened for high mitochondrial DNA levels. Two EnsemblIDs have the same gene symbol (SRSF10). After excluding the EnsemblID with nonzero counts on fewer cells and restricting to genes with nonzero counts on at least 50 cells, counts were retained for 6,075 genes and 3,974 cells. 565 genes were found to be highly variable in the full data set and in all 40 samples.

The rank of the Pearson residuals matrix was estimated as 37 by optht. After mapping to a 37-dimensional Euclidean representation and excluding outliers, 3,906 cells were retained and clustered.

Figure 5 displays the distributions of normalized MED for the clusterings of sizes 2-10. Differences with Figure 3 are immediate: values for the clusterings of sizes 2 and 3 are large. The clusterings of sizes 4-8 are stable; the 90th percentiles (red) are less than or equal to 0.10. Because there are 8 ground truth labels, the clusterings of sizes 7 and 8 are reviewed (green lines).

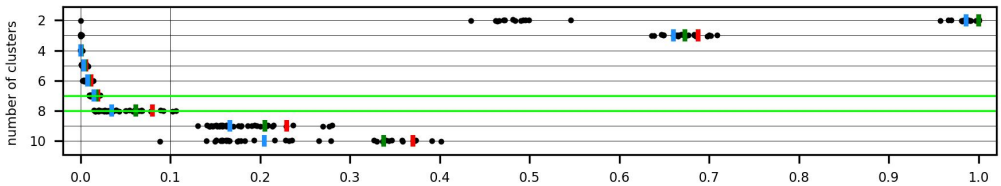

Figure 5 Zhengmix8eq data: distributions of normalized MED for clusterings of sizes 2-10

Figure 6 displays the distributions of normalized CMER for the 7 clusters. All clusters are very stable; the largest value of CMER equals 0.10. Three clusters are extremely stable (4, 5, and 6).

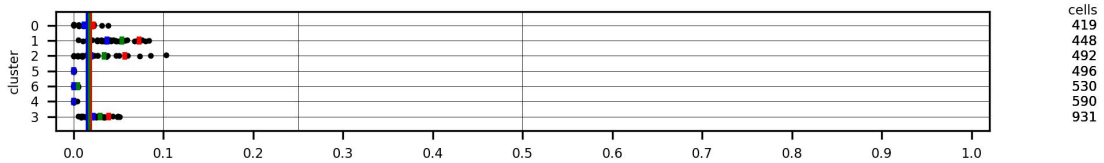

Figure 6 Zhengmix8eq data: distributions of normalized CMER for the 7-clustering

Table 10 compares the clusters with the ground truth labels. Results are very accurate for cd56.nk, b.cells, and cd14.monocytes; less accurate for naive.cytotoxic and memory.t cells. Three subtypes – regulatory.t, cd4.t.helper, and naive.t – are commingled in clusters 2 and 3. The adjusted Rand index equals 0.74.

	hierarchical cluster							
ground truth	0	1	2	3	4	5	6	Total
naive.cytotoxic	389	5	1	1	0	1	0	397
memory.t	26	435	33	1	0	0	0	495
regulatory.t	0	1	308	186	0	0	0	495
cd4.t.helper	0	2	138	258	0	0	0	398
naive.t	2	0	7	485	0	0	0	494
cd56.nk	2	5	0	0	589	1	0	597
b.cells	0	0	0	0	0	493	0	493
cd14.monocytes	0	0	5	0	1	1	530	537
Total	419	448	492	931	590	496	530	3,906

Table 10 Zhengmix8eq data: compare ground truth with the 7-clustering

The 8-clustering is formed from the 7-clustering by splitting its cluster 4 (590 cells) into the two clusters 4 (339 cells) and 5 (251 cells). Figure 7 displays the distributions of normalized CMER for the 8 clusters. All clusters are stable: the 90th percentile is less than or equal to 0.50. However, the two new clusters (top two lines) are much less stable than the extremely stable cluster from which they were formed.

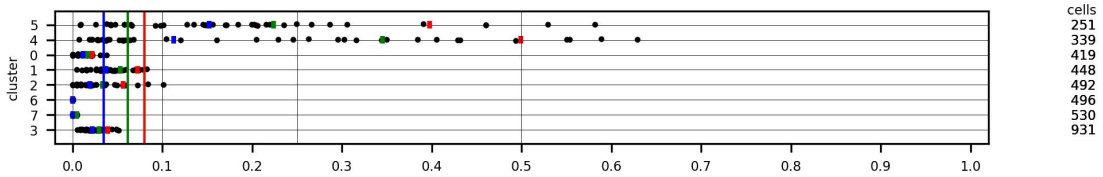

Figure 7 Zhengmix8eq data: distributions of normalized CMER for the 8-clustering

Table 11 compares the clusters with the ground truth labels. The adjusted Rand index equals 0.68.

	hierarchical cluster								
ground truth	0	1	2	3	4	5	6	7	Total
naive.cytotoxic	389	5	1	1	0	0	1	0	397
memory.t	26	435	33	1	0	0	0	0	495
regulatory.t	0	1	308	186	0	0	0	0	495
cd4.t.helper	0	2	138	258	0	0	0	0	398
naive.t	2	0	7	485	0	0	0	0	494
cd56.nk	2	5	0	0	339	250	1	0	597
b.cells	0	0	0	0	0	0	493	0	493
cd14.monocytes	0	0	5	0	0	1	1	530	537
Total	419	448	492	931	339	251	496	530	3,906

Table 11 Zhengmix8eq data: compare ground truth with the 8-clustering

CD14 Monocytes

The data set contains counts for 32,738 genes and 2,612 cells. Filtering to exclude genes with nonzero counts on fewer than 50 cells and to exclude cells with more than 5% of counts due to mitochondrial DNA retained 3,726 genes and 2,558 cells. 719 genes were found to be highly variable in the full data set and in all 40 samples.

The rank of the Pearson residuals matrix was estimated as 11 by optht. After mapping to a 11-dimensional Euclidean representation and excluding Euclidean outliers, 2,537 cells were retained and clustered.

Figure 8 displays the distributions of normalized MED for the clusterings of sizes 2-10. No other data set considered in this article has so many large values for small clusterings (10 clusters or fewer). The median is greater than 0.50 for all clusterings. The large values are consistent with the supposition that all of the cells in the data set are of the same type, so that any clustering would be spurious.

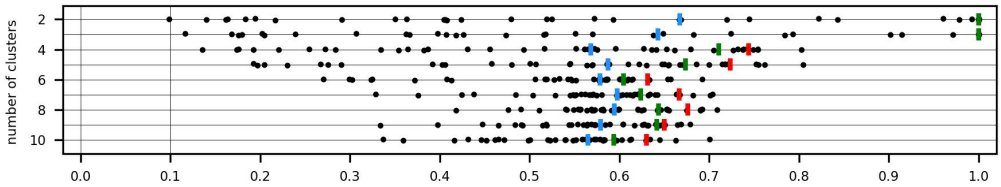

Figure 8 CD14 Monocytes: distributions of normalized MED for clusterings of sizes 2-10

68k PBMC

The data set contains counts for 32,738 genes and 68,579 cells. Filtering to exclude genes with nonzero counts on fewer than 50 cells and to exclude cells with more than 5% of counts due to mitochondrial DNA retained 12,515 genes and 68,286 cells. 652 genes were found to be highly variable in the full data set and in all 40 samples.

The rank of the Pearson residuals matrix was estimated as 48 by optht. After mapping to a 48-dimensional Euclidean representation and excluding Euclidean outliers, 67,792 cells were retained and clustered.

Our objective was to evaluate clusterings of sizes up to 25 – larger than the 10 discussed in the paper. Clusterings of all sizes in the range 2-25 were found in each of the three sets of analyses. For the first iteration, the clusterings of sizes 2, 3, 6, 11, and 12 are stable. The largest (12 clusters) is admissible. The second iteration, using fewer cells, also yields a stable 12-clustering, but it is not admissible; it has a totally unstable cluster of 1,716 cells. The third iteration yields two stable clusterings, but they are small – with 2 and 3 clusters.

Figure 9 displays the distributions of normalized MED for the clusterings of sizes 2-25 found with the first iteration. The green line indicates the 12-clustering.

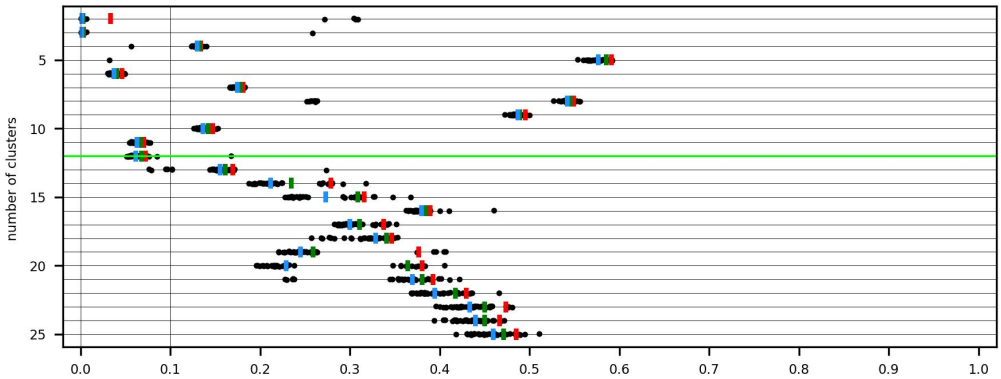

Figure 9 68k PBMC data (iteration 1): distributions of normalized MED for clusterings of sizes 2-25

Figure 10 displays the distributions of normalized CMER for the 12-clustering. The two smallest clusters, 8 and 1, are totally unstable; their data lines are highlighted red. Three clusters are extremely stable (0, 2, and 7). Four more are very stable. We judge the clustering admissible for downstream analysis because the unstable clusters have fewer than 500 cells. The blue, green, and red lines extending from the bottom to the top of the plot indicate the median, 75th, and 90th percentiles of MED for the clustering.

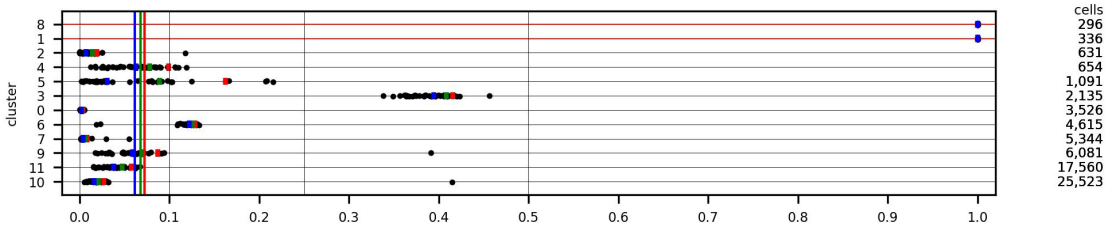

Figure 10 68k PBMC data (iteration 1): distributions of normalized CMER for the 12-clustering

To compare clusterings found by our method with published results, the input data (68,579 cells) were clustered (k-means) using code from a 10x Genomics Github repository (see Data availability below). The sizes of the 10 clusters we obtained do not agree precisely with the percentages in Figure 3b of the paper [19]. In particular, we obtained one small cluster – of 176 cells. The remaining 9 clusters range in size from 3,000 to 18,000 cells. When the clusters obtained with our process were matched to these, none of the 176 cells in the smallest cluster were included – filtering discarded all of them.

Table 12 compares the 9 k-means clusters with the 12 hierarchical clusters. The adjusted Rand index equals 0.55.

	hierarchical cluster												
k-means cluster	0	1	2	3	4	5	6	7	8	9	10	11	Total
10	3,435	331	19	7	0	0	3	0	291	0	0	1	4,087
9	0	0	609	1,980	654	1,091	2	0	0	0	0	0	4,336
1	2	0	0	1	0	0	3,871	32	0	1	0	0	3,907
6	12	0	0	4	0	0	274	5,211	0	827	0	2	6,330
5	17	0	0	9	0	0	454	89	1	5,098	8	2,577	8,253
4	19	0	0	10	0	0	1	0	1	2	17,193	1,010	18,236
7	6	2	0	11	0	0	3	0	0	24	6,724	484	7,254
2	31	0	0	88	0	0	3	2	1	69	1,598	10,638	12,430
3	4	3	3	25	0	0	4	10	2	60	0	2,848	2,959
Total	3,526	336	631	2,135	654	1,091	4,615	5,344	296	6,081	25,523	17,560	67,792

Table 12 68k PBMC data (iteration 1): compare 9 k-means clusters with the 12-clustering

Note that

- The majority of cells in k-means cluster 10 are split three ways: most are in hierarchical cluster 0, which is extremely stable, approximately 300 cells each are in clusters 1 and 8, which are totally unstable
- k-means cluster 9 is split four ways: cells are divided among hierarchical clusters 2, 3, 4, and 5; differential expression may help to determine if these clusters are meaningful or spurious
- k-means clusters 1 and 6 correspond to hierarchical clusters 6 and 7, respectively
- 60% of the cells of k-means cluster 5 account for 80% of hierarchical cluster 9
- 95% of cells in k-means cluster 4 are grouped with 90% of the cells in k-means cluster 7 in hierarchical cluster 10, which is very stable
- 90% of cells in k-means cluster 2 are combined with nearly all of k-means cluster 3 and cells from other clusters into the very stable hierarchical cluster 11

We review a second analysis of the 68k PBMC data set because it yields a clustering more compatible with the k-means clusters. It is the result of the third iteration described in Section 2.8 - Identify and exclude cell and gene outliers for iterative analysis: counts for gene and cell outliers were deleted, retaining 63,281 cells. Figure 11 displays the distributions of normalized MED. As noted above, only the 2 and 3-clusterings are stable.

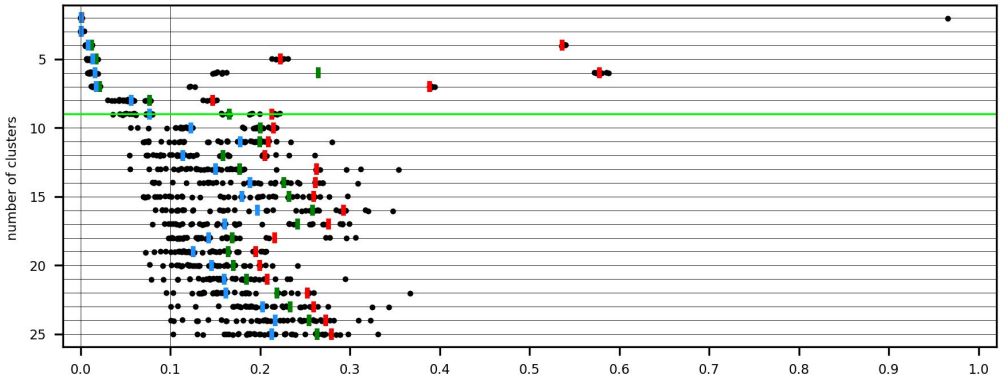

Figure 11 68k PBMC data (iteration 3): distributions of normalized MED for clusterings of sizes 2-25

Preliminary exploratory analysis using the *median* of MED instead of the 90th percentile to define stable clusterings led to consideration of the 9-clustering (green highlighted line) because it is the largest with median MED (blue marker) less than or equal to 0.10. It fails to satisfy the (90th percentile) criterion we propose to define a stable clustering: the 90th percentile of MED equals 0.21, double the cutoff we are using.

The confusion matrices in Tables 2 and 3 are for two of the samples represented in Figure 12, which displays the distributions of normalized CMER for the 9 clusters.

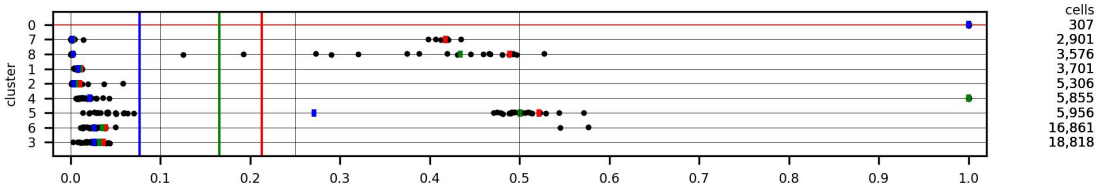

Figure 12 68k PBMC data (iteration 3): distributions of normalized CMER for the 9-clustering

As stated in the discussion of Table 3, cluster 0 (top line) is totally unstable. Clusters 4 and 5 are unstable and cluster 8 is barely stable (the 90th percentile of CMER equals 0.49). Two clusters are extremely stable (1 and 2) and two others are very stable.

Table 13 compares the 9 k-means clusters with the 9 hierarchical clusters. The adjusted Rand index equals 0.66.

	hierarchical cluster									
k-means cluster	0	1	2	3	4	5	6	7	8	Total
1	286	3,086	28	0	0	1	0	0	0	3,401
6	20	173	5,167	0	0	591	2	0	5	5,958
4	0	1	0	16,371	298	3	1,042	10	19	17,744
7	0	1	0	1,058	5,499	25	481	13	5	7,082
5	1	432	104	1	7	5,222	2,134	5	14	7,920
2	0	3	0	1,388	51	65	10,477	86	23	12,093
3	0	5	7	0	0	49	2,725	15	3	2,804
9	0	0	0	0	0	0	0	2,772	0	2,772
10	0	0	0	0	0	0	0	0	3,507	3,507
Total	307	3,701	5,306	18,818	5,855	5,956	16,861	2,901	3,576	63,281

Table 13 68k PBMC data (iteration 3): compare 9 k-means clusters with the 9-clustering

Note that

- The totally unstable cluster 0 is split from k-means cluster 1; the two totally unstable clusters in the 12-clustering are split from k-means cluster 10
- Each k-means cluster except 3 is unambiguously associated with a hierarchical cluster

25k retinal

Although the counts downloaded from the Gene Expression Omnibus contain data for 49,300 cells, we followed Lause et al. [15] by restricting to replicates p1 and r4-r6, netting counts for 22,292 genes and 24,769 cells. Thirty-nine cell clusters were reported (Figure 5D of [21]). Restricted to the cells we analyzed, cluster sizes range from 14 to 15,709 cells.

Filtering to exclude genes with nonzero counts on fewer than 50 cells retained 13,552 genes. Because data were not input through Seurat, cells were not screened for high mitochondrial DNA levels. 1,081 genes were found to be highly variable in the full data set and in all 40 samples.

The rank of the Pearson residuals matrix was estimated as 51 by optht. After mapping to a 51-dimensional Euclidean representation and excluding Euclidean outliers, 24,101 cells were retained and clustered. These cells belong to 37 of the 39 reported clusters.

Our objective was to evaluate clusterings of sizes 2-70 – the largest being greater than the number of reported cell clusters (39). However, the stopping conditions limited the largest clustering for at least one sample to a smaller size: 61 for iteration 1; 62 for iteration 2; 58 for iteration 3.

The largest stable clusterings found in the first and second iterations have 5 clusters. The largest stable clustering found in the third iteration has 11; we show below that it is admissible. The third iteration retained 22,416 cells, which belong to 36 of the 39 published cell clusters.

Figure 13 displays the distributions of normalized MED for the clusterings of sizes 2-58 found with the third iteration.

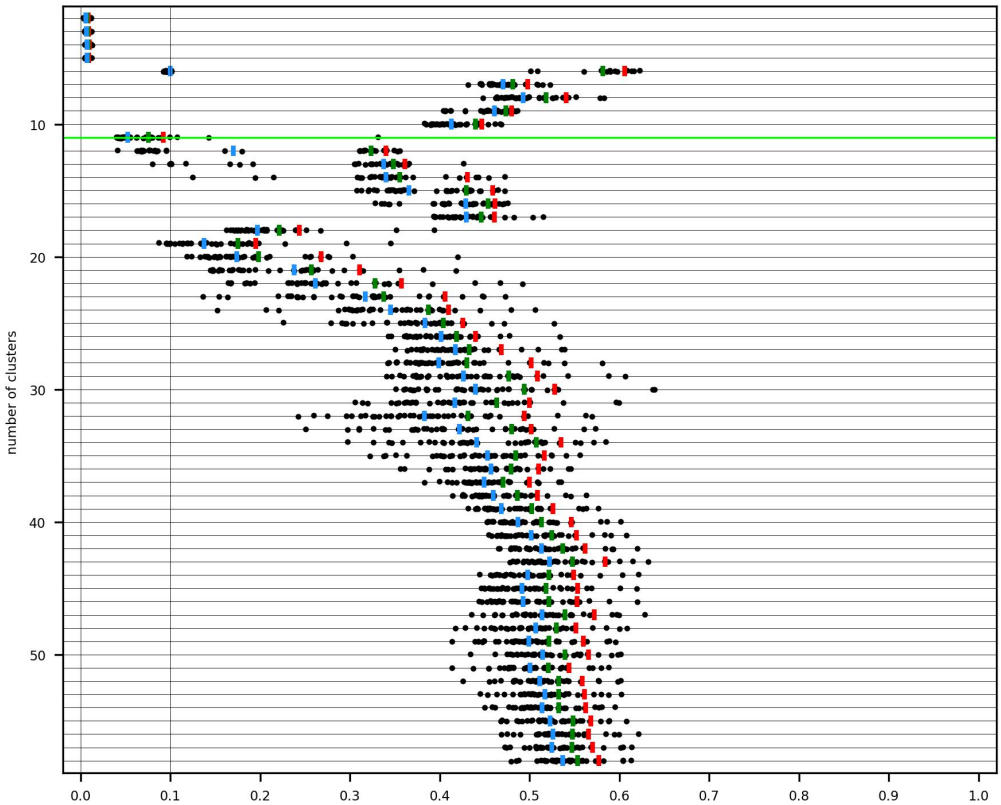

Figure 13 25k retinal data (iteration 3): distributions of normalized MED for clusterings of sizes 2-58

Figure 14 displays the distributions of normalized CMER for the 11 clusters. Cluster 4 is totally unstable and cluster 3 is unstable (top two lines). One cluster is extremely stable (10). Four others are very stable. The clustering is admissible for downstream analysis because the unstable clusters have fewer than 500 cells.

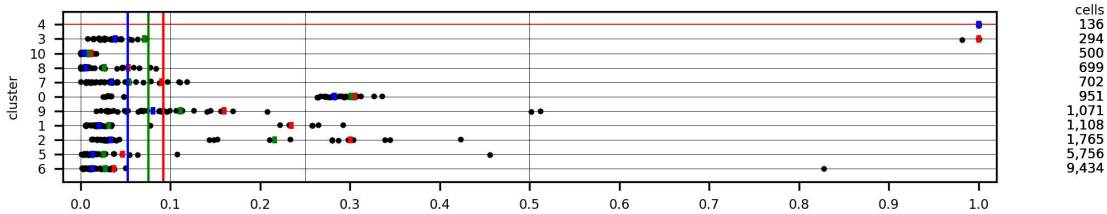

Figure 14 25k retinal data (iteration 3): distributions of normalized CMER for the 11-clustering

Table 14 shows the compatibility of the 11 hierarchical clusters with the reported cell clusters. The adjusted Rand index equals 0.49.

cell cluster	hierarchical cluster 0	1	2	3	4	5	6	7	8	9	10	Total
1	1	1	0	0	0	0	0	0	0	0	0	2
16	152	0	3	1	0	1	0	0	1	1	0	159
17	220	0	1	0	0	0	0	1	0	0	0	222
18	41	0	0	0	0	1	0	0	0	0	0	42
19	74	0	0	0	0	1	0	0	0	0	0	75
20	178	0	2	0	0	11	17	2	0	0	0	210
21	116	2	3	0	0	7	8	1	0	0	0	137
22	116	5	2	1	0	4	12	0	0	1	1	142
39	1	1	0	0	0	0	0	0	0	0	0	2
2	0	77	0	0	0	2	2	0	0	0	0	81
4	1	10	0	0	0	0	1	0	0	0	0	12
5	0	13	0	0	0	0	1	0	0	0	0	14
6	1	99	0	0	0	0	1	0	0	0	0	101
7	0	160	2	0	0	1	0	0	0	1	0	164
8	0	8	0	0	0	1	1	0	0	0	0	10
9	0	189	0	0	0	0	2	0	0	0	0	191
10	1	54	1	0	0	1	1	0	0	0	0	58
11	1	118	1	0	0	0	1	0	0	0	0	121
12	3	136	0	0	0	0	11	1	0	1	0	152
13	0	24	0	0	0	0	0	0	0	0	0	24
14	0	1	0	0	0	0	0	0	0	0	0	1
23	0	140	2	0	0	2	9	1	0	1	0	155
28	0	0	230	16	0	4	10	10	0	0	0	270
29	2	0	249	11	0	9	36	2	3	3	1	316
30	1	1	315	0	0	1	10	5	2	0	1	336
31	3	2	255	3	0	2	7	3	4	0	0	279
32	0	0	179	0	0	4	1	2	1	0	0	187
33	2	4	410	0	0	12	23	27	6	4	1	489
27	0	1	20	259	0	17	28	14	4	0	0	343
3	0	2	0	0	135	1	0	0	0	0	0	138
36	0	0	0	0	0	1	0	0	0	0	1	2
24	36	59	84	3	1	5,646	9,221	14	1	6	60	15,131
38	0	0	0	0	0	0	0	1	0	0	0	1
26	0	1	1	0	0	2	1	614	676	0	0	1,295
25	1	0	5	0	0	21	29	3	0	1,053	0	1,112
34	0	0	0	0	0	4	1	1	1	0	435	442
Total	951	1,108	1,765	294	136	5,756	9,434	702	699	1,071	500	22,416

Table 14 25k retinal data (iteration 3): compare 36 of 39 published clusters with the 11-clustering

Note that

- Most members of cell cluster 24 (rods) belong to hierarchical clusters 5 and 6, both of which are very stable; we anticipate using differential expression analysis to evaluate whether this split is appropriate or spurious
- Similarly, cell cluster 26 is split between hierarchical clusters 7 and 8, which are very stable
- Cell cluster 25 (cones) agrees closely with hierarchical cluster 9
- Cell cluster 3 agrees almost exactly with the totally unstable hierarchical cluster 4
- 80% of the cells in cluster 27 belong to the unstable hierarchical cluster 3
- Cell cluster 34 agrees closely with the extremely stable hierarchical cluster 10

65k lung

The data set contains counts for 26,485 genes and 65,662 cells. The accompanying metadata file provides 57 cell type annotations.

Excluding data for blood cells retained 60,993 cells. Filtering to exclude genes with nonzero counts on fewer than 50 cells retained 17,470 genes. Because data were not input through Seurat, cells were not screened for high mitochondrial DNA levels. 1,659 genes were found to be highly variable in the full set of data and all 40 samples.

The rank of the Pearson residuals matrix was estimated as 305 by optht. After mapping to a 305-dimensional Euclidean representation and excluding Euclidean outliers, 60,114 cells were retained and clustered.

The paper [22] describes separately clustering the data for each patient. Our analysis included batch correction; each of the 3 patients' data was treated as an independent batch.

Our objective was to evaluate clusterings of sizes 2-70 – the largest being greater than the number of reported cell types (57). However, the stopping conditions limited the largest clustering for at least one sample to a smaller size: 69 for iteration 1; 48 for iteration 2; 41 for iteration 3. The largest stable clustering found in the first iteration has 19 clusters; we show below that it is admissible. The largest stable clusterings found in the second and third iterations have 13 clusters.

Figure 15 displays the distributions of normalized MED for the clusterings of sizes 2-69 found with the first iteration. The green lines highlight data for the clusterings of sizes 16 and 19, which we review.

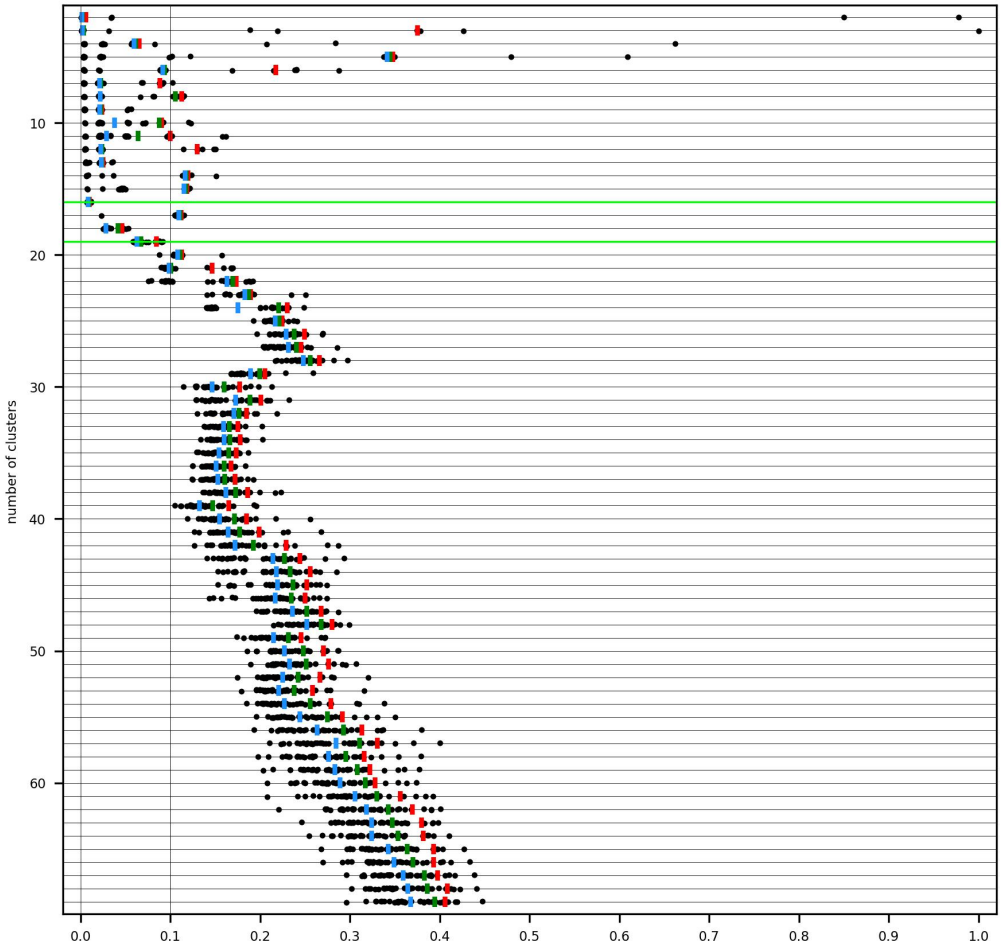

Figure 15 65k lung data (iteration 1): distributions of normalized MED for clusterings of sizes 2-69

Figure 16 displays the distributions of normalized CMER for the 19 clusters. Clusters 10 and 2 (first and third lines) are totally unstable. Eight clusters are extremely stable (3, 6, 8, 12, 13, 16, 17, 18). Six more are very stable. The clustering is admissible for downstream analysis because the unstable clusters have fewer than 500 cells.

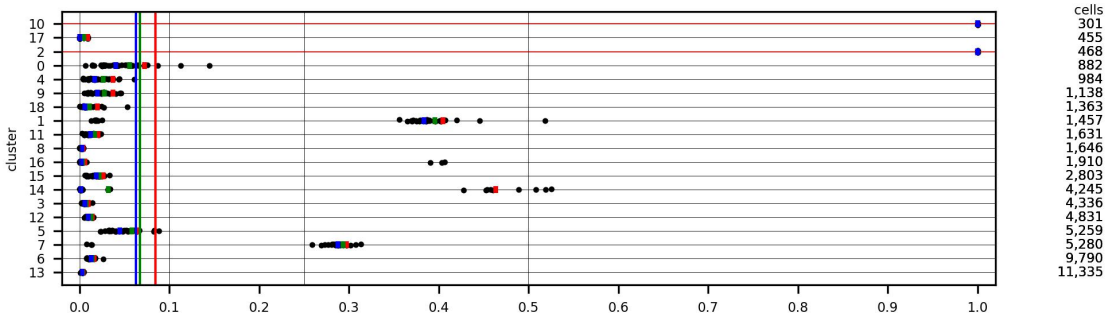

Figure 16 65k lung data (iteration 1): distributions of normalized CMER for the 19-clustering

Because blood cells were excluded from our analysis, data for one cell type were eliminated. Table 15 shows the compatibility of the 56 retained cell types with the 19 hierarchical clusters. The adjusted Rand index equals 0.65.

Macrophages are divided between clusters 5 (very stable) and 6 (extremely stable). Differential expression may help determine if this split is valid or spurious.

	hierarchical cluster																			
cell type	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Total
Basal	368	19	0	3	0	0	7	0	0	0	0	0	1	3	1	5	0	0	0	407
Differentiating Basal	184	44	0	0	0	0	6	0	0	0	0	0	0	1	1	7	0	0	0	243
Goblet	114	3	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	1	122
Proliferating Basal	36	0	0	0	0	0	1	0	0	0	0	0	0	0	2	1	0	0	0	40
Proximal Basal	149	1	0	0	0	0	4	0	0	0	0	0	0	0	0	1	0	0	0	155
Club	6	816	0	1	2	0	1	0	0	0	0	0	1	1	36	0	1	0	0	865
Mucous	16	332	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	351
B	0	0	239	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	242
Plasma	0	0	93	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	94
Plasmacytoid Dendritic	0	0	114	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	116
CD4+ Memory/Effector T	0	2	6	2,385	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	2,414
CD4+ Naive T	0	0	1	287	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	288
CD8+ Memory/Effector T	0	0	0	693	0	0	0	77	0	0	0	0	0	0	0	0	0	0	0	770
Mesothelial	2	0	0	7	0	1	0	0	0	1	4	1	0	4	0	0	0	0	0	20
Natural Killer T	0	0	0	252	0	0	0	83	0	0	0	0	0	0	0	0	0	0	0	335
Alveolar Epithelial Type 1	1	3	0	0	943	0	0	0	0	0	0	0	9	3	2	0	1	0	0	962
TREM2+ Dendritic	0	0	1	0	0	84	46	0	0	0	0	0	1	1	0	16	0	0	0	149
Ionocyte	2	0	0	2	0	0	14	0	0	0	0	0	0	1	1	2	0	0	0	22
Macrophage	0	3	0	8	0	5,132	9,468	1	1	1	0	0	3	6	1	66	11	0	0	14,701
Proliferating Macrophage	0	0	0	0	0	38	187	0	0	0	0	0	0	0	0	0	1	0	0	226
CD8+ Naive T	0	0	3	625	0	0	0	776	0	0	0	0	0	0	0	0	2	0	0	1,406
Natural Killer	0	0	0	15	0	0	0	4,232	0	0	0	0	0	0	0	0	0	0	0	4,247
Proliferating NK/T	0	0	0	22	0	0	0	78	0	0	0	0	0	0	0	0	1	0	0	101
Neuroendocrine	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	3
Pericyte	0	0	0	1	1	0	2	0	1,585	15	0	1	0	0	0	0	0	0	0	1,605
Airway Smooth Muscle	0	0	0	0	0	0	0	0	6	679	1	0	0	0	0	1	0	0	0	687
Vascular Smooth Muscle	0	0	0	2	0	0	0	0	45	410	1	4	0	0	0	0	0	0	0	462
Fibromyocyte	0	0	0	0	0	0	0	0	0	30	62	0	0	1	0	0	0	0	0	93
Myofibroblast	0	0	0	0	0	0	0	1	2	0	207	11	0	3	0	0	0	0	0	224
Adventitial Fibroblast	0	0	0	0	0	0	0	0	0	0	5	354	0	9	0	0	0	0	0	368
Alveolar Fibroblast	0	0	0	0	0	0	1	2	1	0	7	1,240	0	7	0	3	0	0	0	1,261
Lipofibroblast	0	0	0	0	0	0	0	0	0	0	13	15	0	0	0	0	0	0	0	28
Capillary Aerocyte	0	0	0	0	0	0	0	0	0	0	1	0	4,250	98	1	0	1	0	0	4,351
Capillary Intermediate 1	0	0	0	0	0	0	0	0	0	0	0	0	475	152	0	0	1	0	0	628
Artery	0	0	0	0	0	0	0	1	1	0	0	1	7	1,470	0	0	0	0	0	1,480
Bronchial Vessel 1	0	0	1	0	0	0	0	0	1	0	0	1	1	431	0	0	0	2	0	437
Bronchial Vessel 2	0	0	0	0	0	0	0	0	0	0	0	2	8	217	0	1	0	0	0	228
Capillary	0	0	1	3	0	0	3	2	0	1	0	0	54	7,321	0	0	1	0	0	7,386
Capillary Intermediate 2	0	0	0	0	0	0	0	0	0	1	0	0	7	456	0	0	0	0	0	464
Vein	0	0	0	0	0	0	0	0	1	0	0	0	7	1,141	1	0	0	0	0	1,150
Alveolar Epithelial Type 2	3	175	0	1	36	0	0	2	0	0	0	0	2	5	3,564	1	2	0	0	3,791
Signaling Alveolar Epithelial Type 2	0	31	0	0	1	0	0	0	0	0	0	0	2	0	634	1	0	0	0	669
Classical Monocyte	0	0	0	2	0	0	1	0	0	0	0	0	0	0	0	1,010	0	0	0	1,013
EREG+ Dendritic	0	0	0	0	0	0	11	0	0	0	0	0	0	1	0	129	0	0	0	141
IGSF21+ Dendritic	0	0	0	7	0	0	6	0	0	0	0	0	0	0	0	263	1	0	0	277
Intermediate Monocyte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	187	0	0	0	187
Myeloid Dendritic Type 1	0	0	8	9	0	0	1	1	0	0	0	0	0	0	0	94	0	0	0	113
Myeloid Dendritic Type 2	0	0	0	6	0	2	7	2	0	0	0	0	0	0	0	213	0	0	0	230
Nonclassical Monocyte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	603	0	1	0	604
OLR1+ Classical Monocyte	0	0	1	1	0	0	2	0	0	0	0	0	0	1	0	196	3	0	0	204
Basophil/Mast 1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	1	1,336	0	0	1,339
Basophil/Mast 2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	547	0	0	548
Lymphatic	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	452	0	455
Ciliated	0	27	0	0	0	0	13	1	0	0	0	0	0	1	1	0	0	0	1,270	1,313
Platelet/Megakaryocyte	1	1	0	0	1	2	2	0	0	0	0	0	0	0	0	0	0	0	4	11
Proximal Ciliated	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	88	88
Total	882	1,457	468	4,336	984	5,259	9,790	5,280	1,646	1,138	301	1,631	4,831	11,335	4,245	2,803	1,910	455	1,363	60,114

Table 15 65k lung data (iteration 1): compare 56 of 57 reported cell types with the 19-clustering

We review the 16-clustering of the same data because the results – for both the clustering and its clusters – are the most stable we found in the four large data sets. Referring to Figure 15, the *maximum* value of normalized MED equals 0.01. We cannot explain this why it is so extraordinarily small.

Figure 17 displays the distributions of normalized CMER for the 16 clusters. Ten clusters are extremely stable; all of the others are very stable.

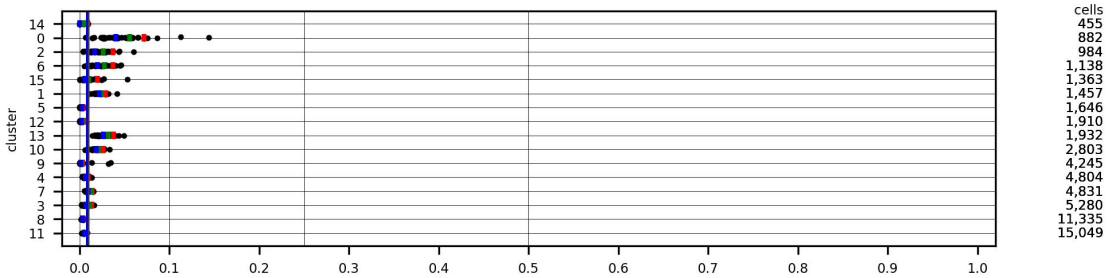

Figure 17 65k lung data (iteration 1): distributions of normalized CMER for the 16-clustering

Table 16 compares the published cell types with the 16 hierarchical clusters. The adjusted Rand index equals 0.81. Nearly all macrophages belong to cluster 11 (extremely stable).

	hierarchical cluster																
cell type	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Basal	368	19	0	0	3	0	0	1	3	1	5	7	0	0	0	0	407
Differentiating Basal	184	44	0	0	0	0	0	0	1	1	7	6	0	0	0	0	243
Goblet	114	3	0	0	0	0	0	0	0	0	0	4	0	0	0	1	122
Proliferating Basal	36	0	0	0	0	0	0	0	0	2	1	1	0	0	0	0	40
Proximal Basal	149	1	0	0	0	0	0	0	0	0	1	4	0	0	0	0	155
Club	6	816	2	0	1	0	0	1	1	36	0	1	1	0	0	0	865
Mucous	16	332	0	0	0	0	0	2	0	0	0	1	0	0	0	0	351
Alveolar Epithelial Type 1	1	3	943	0	0	0	0	9	3	2	0	0	1	0	0	0	962
CD8+ Naive T	0	0	0	776	628	0	0	0	0	0	0	0	2	0	0	0	1,406
Natural Killer	0	0	0	4,232	15	0	0	0	0	0	0	0	0	0	0	0	4,247
Proliferating NK/T	0	0	0	78	22	0	0	0	0	0	0	0	1	0	0	0	101
B	0	0	0	0	241	0	0	0	0	0	0	0	1	0	0	0	242
CD4+ Memory/Effector T	0	2	0	21	2,391	0	0	0	0	0	0	0	0	0	0	0	2,414
CD4+ Naive T	0	0	0	0	288	0	0	0	0	0	0	0	0	0	0	0	288
CD8+ Memory/Effector T	0	0	0	77	693	0	0	0	0	0	0	0	0	0	0	0	770
Mesothelial	2	0	0	0	7	0	1	0	4	0	0	1	0	5	0	0	20
Natural Killer T	0	0	0	83	252	0	0	0	0	0	0	0	0	0	0	0	335
Plasma	0	0	0	0	93	0	0	0	0	0	1	0	0	0	0	0	94
Plasmacytoid Dendritic	0	0	0	0	114	0	0	0	0	0	1	1	0	0	0	0	116
Neuroendocrine	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	3
Pericyte	0	0	1	0	1	1,585	15	0	0	0	0	2	0	1	0	0	1,605
Airway Smooth Muscle	0	0	0	0	0	6	679	0	0	0	1	0	0	1	0	0	687
Vascular Smooth Muscle	0	0	0	0	2	45	410	0	0	0	0	0	0	5	0	0	462
Capillary Aerocyte	0	0	0	0	0	0	0	4,250	98	1	0	0	1	1	0	0	4,351
Capillary Intermediate 1	0	0	0	0	0	0	0	475	152	0	0	0	1	0	0	0	628
Artery	0	0	0	1	0	1	0	7	1,470	0	0	0	0	1	0	0	1,480
Bronchial Vessel 1	0	0	0	0	1	1	0	1	431	0	0	0	0	1	2	0	437
Bronchial Vessel 2	0	0	0	0	0	0	0	8	217	0	1	0	0	2	0	0	228
Capillary	0	0	0	2	4	0	1	54	7,321	0	0	3	1	0	0	0	7,386
Capillary Intermediate 2	0	0	0	0	0	0	1	7	456	0	0	0	0	0	0	0	464
Vein	0	0	0	0	0	1	0	7	1,141	1	0	0	0	0	0	0	1,150
Alveolar Epithelial Type 2	3	175	36	2	1	0	0	2	5	3,564	1	0	2	0	0	0	3,791
Signaling Alveolar Epithelial Type 2	0	31	1	0	0	0	0	2	0	634	1	0	0	0	0	0	669
Classical Monocyte	0	0	0	0	2	0	0	0	0	0	1,010	1	0	0	0	0	1,013
EREG+ Dendritic	0	0	0	0	0	0	0	0	1	0	129	11	0	0	0	0	141
IGSF21+ Dendritic	0	0	0	0	7	0	0	0	0	0	263	6	1	0	0	0	277
Intermediate Monocyte	0	0	0	0	0	0	0	0	0	0	187	0	0	0	0	0	187
Myeloid Dendritic Type 1	0	0	0	1	17	0	0	0	0	0	94	1	0	0	0	0	113
Myeloid Dendritic Type 2	0	0	0	2	6	0	0	0	0	0	213	9	0	0	0	0	230
Nonclassical Monocyte	0	0	0	0	0	0	0	0	0	0	603	0	0	0	1	0	604
OLR1+ Classical Monocyte	0	0	0	0	2	0	0	0	1	0	196	2	3	0	0	0	204
Ionocyte	2	0	0	0	2	0	0	0	1	1	2	14	0	0	0	0	22
Macrophage	0	3	0	1	8	1	1	3	6	1	66	14,600	11	0	0	0	14,701
Platelet/Megakaryocyte	1	1	1	0	0	0	0	0	0	0	0	4	0	0	0	4	11
Proliferating Macrophage	0	0	0	0	0	0	0	0	0	0	0	225	1	0	0	0	226
TREM2+ Dendritic	0	0	0	0	1	0	0	1	1	0	16	130	0	0	0	0	149
Basophil/Mast 1	0	0	0	0	2	0	0	0	0	0	1	0	1,336	0	0	0	1,339
Basophil/Mast 2	0	0	0	0	0	0	0	0	0	0	0	1	547	0	0	0	548
Adventitial Fibroblast	0	0	0	0	0	0	0	0	9	0	0	0	0	359	0	0	368
Alveolar Fibroblast	0	0	0	2	0	1	0	0	7	0	3	1	0	1,247	0	0	1,261
Fibromyocyte	0	0	0	0	0	0	30	0	1	0	0	0	0	62	0	0	93
Lipofibroblast	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	28
Myofibroblast	0	0	0	1	0	2	0	0	3	0	0	0	0	218	0	0	224
Lymphatic	0	0	0	0	0	0	0	1	1	0	0	0	0	1	452	0	455
Ciliated	0	27	0	1	0	0	0	0	1	1	0	13	0	0	0	1,270	1,313
Proximal Ciliated	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	88	88
Total	882	1,457	984	5,280	4,804	1,646	1,138	4,831	11,335	4,245	2,803	15,049	1,910	1,932	455	1,363	60,114

Table 16 65k lung data (iteration 1): compare 56 of 57 reported cell types with the 16-clustering

100k breast cancer

The data set contains counts for 29,733 genes and 100,064 cells. Nine major cell types were reported, as well as 29 minor cell types and 49 cell type subsets.

Filtering to exclude genes with nonzero counts on fewer than 50 cells retained 21,354 genes. The data include cells for which up to 20% of counts are due to mitochondrial DNA. (The percentage exceeds 5% for half of the cells.) 1,704 genes were found to be highly variable in the set of counts for all cells and in all 40 samples.

The rank of the Pearson residuals matrix was estimated as 434 by optht. After mapping to a 434-dimensional Euclidean representation, 98,681 cells were retained and clustered.

Discussion in [23] and programs posted at https://github.com/Swarbricklab-code/BrCa_cell suggest that batch correction was performed for each patient's data for some, if not all, analyses. Our analysis included batch correction; each of the 26 patients' data was treated as an independent batch.

Our objective was to evaluate clusterings of sizes 2-70 – the largest being greater than the number of reported cell type subsets (49). However, the stopping conditions limited the largest clustering for at least one sample to a smaller size: 58 for iteration 1; 39 for iteration 2; 51 for iteration 3. No clustering found in any of the iterations is stable. The smallest values of the 90th percentile of MED found are 0.15 (first iteration), 0.16 (second), and 0.107 (third) – all exceed 0.10.

Figure 18 displays the distributions of normalized MED for the clusterings of sizes 2-51 found with the third iteration. The smallest value of the 90th percentile (0.107) was found for the 9-clustering, which we review – while recognizing the hazard of applying hard thresholds.

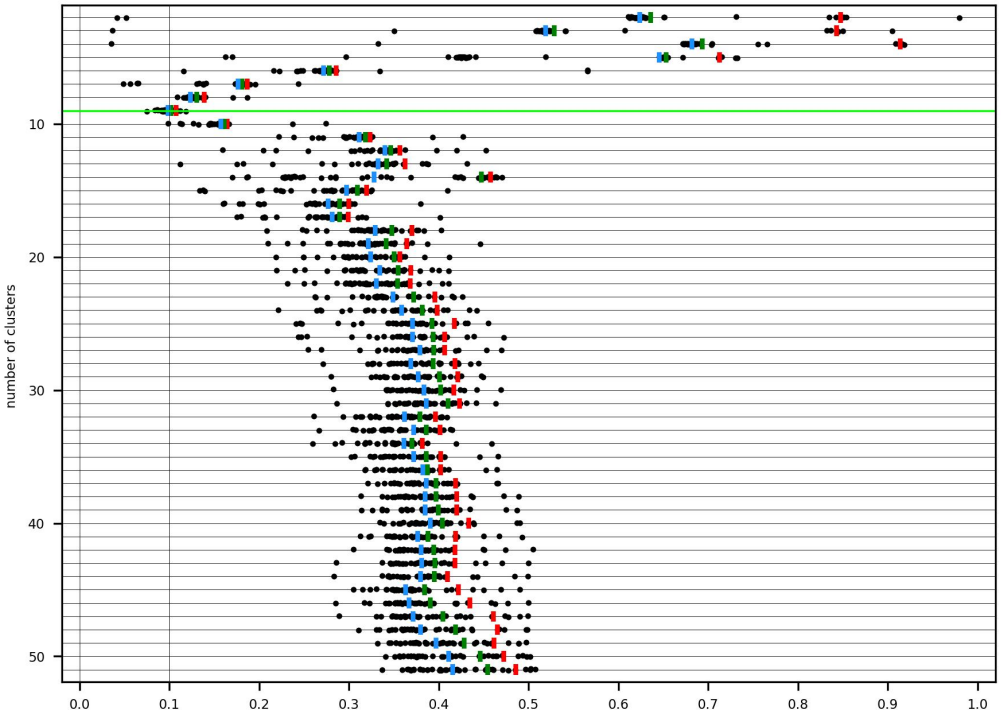

Figure 18 100k breast cancer data (iteration 3): distributions of normalized MED for clusterings of sizes 2-51

Figure 19 displays the distributions of normalized CMER for the 9 clusters. Cluster 4 (top line) is extremely unstable; CMER equals 1 with 39 of the 40 samples. Cluster 1 (fifth line) is unstable: the 90th percentile of CMER equals 0.55, larger than the cutoff of 0.50 that we propose to define stable clusters; this is a second reminder of the risk of imposing hard selection criteria. Although no cluster is extremely stable, three are very stable (0, 5, and 7).

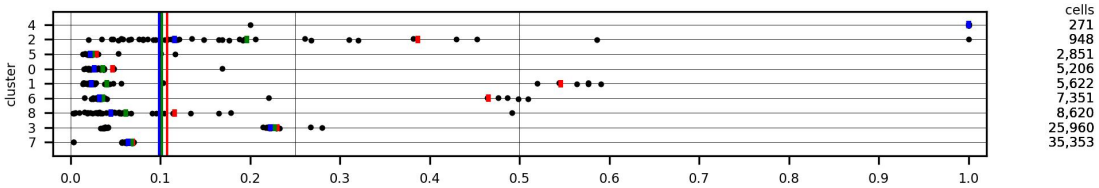

Figure 19 100k breast cancer data (iteration 3): distributions of normalized CMER for the 9-clustering

.

The third iteration retained 92,182 cells, which include all 49 published cell type subsets. Filtering had a disproportionate impact on plasmablasts, one of the 9 major cell types. The data downloaded from the GEO included counts for 100,064 cells (Table 1) of which 3,524 are plasmablasts. Excluding Euclidean outliers before clustering retained 98,681 cells (Table 4) of which 2,290 are plasmablasts. (They account for 1,234 of the 1,383 cells excluded as Euclidean outliers.) Only 801 plasmablasts are among the cells retained for clustering in the third iteration.

Table 17 compares the 9 hierarchical clusters with the major cell types reported in the paper. The adjusted Rand index equals 0.86. Note that

- The hierarchical clusters do not separate normal from cancer epithelial cells, though approximately a quarter of the normal cells are assigned to cluster 2, accounting for more than 95% of its membership
- The extremely unstable cluster 4 represents a segment of myeloid cells
- 90% of the plasmablasts that survived the filtering iterations are split between clusters 3 and 5

	hierarchical cluster									
major cell type	0	1	2	3	4	5	6	7	8	Total
PVL	4,915	39	4	46	0	0	26	2	4	5,036
CAFs	157	5,364	2	73	0	1	4	8	13	5,622
Cancer Epithelial	61	165	19	21,868	0	23	18	1,402	144	23,700
Normal Epithelial	3	1	909	2,910	0	2	1	22	10	3,858
Plasmablasts	9	6	3	431	2	300	43	7	0	801
B-cells	14	3	8	254	3	2,425	98	25	45	2,875
Endothelial	22	12	0	59	0	2	7,109	3	4	7,211
T-cells	7	4	2	205	0	85	12	33,831	13	34,159
Myeloid	18	28	1	114	266	13	40	53	8,387	8,920
Total	5,206	5,622	948	25,960	271	2,851	7,351	35,353	8,620	92,182

Table 17 100k breast cancer data (iteration 3): compare major cell types with the 9-clustering

Table 18 compares the 9 hierarchical clusters with the 49 cell type subsets identified in the paper. The adjusted Rand index equals 0.20. Note that

- Cluster 2, mostly normal epithelial cells, corresponds to myoepithelial cells
- Cluster 4 (extremely unstable) corresponds to myeloid c4 DCs pDC IRF7 cells

cell type subset	hierarchical cluster 0	1	2	3	4	5	6	7	8	Total
Cycling PVL	37	5	0	0	0	0	2	0	0	44
PVL Differentiated s3	3,235	1	3	26	0	0	2	2	3	3,272
PVL Immature s1	997	27	1	16	0	0	14	0	1	1,056
PVL Immature s2	646	6	0	4	0	0	8	0	0	664
CAFs MSC iCAF-like s1	22	1,893	1	19	0	0	0	1	0	1,936
CAFs MSC iCAF-like s2	102	594	1	34	0	1	3	4	6	745
CAFs Transitioning s3	2	474	0	6	0	0	0	0	1	483
CAFs myCAF like s4	27	500	0	12	0	0	0	3	5	547
CAFs myCAF like s5	4	1,903	0	2	0	0	1	0	1	1,911
Myoepithelial	2	1	899	28	0	0	1	0	1	932
Cancer Basal SC	30	108	8	3,609	0	7	2	258	49	4,071
Cancer Cycling	11	28	7	4,826	0	6	6	249	48	5,181
Cancer Her2 SC	8	3	2	3,464	0	4	2	47	29	3,559
Cancer LumA SC	5	19	1	7,405	0	4	7	143	15	7,599
Cancer LumB SC	7	7	1	2,564	0	2	1	705	3	3,290
Luminal Progenitors	0	0	10	1,807	0	1	0	8	8	1,834
Mature Luminal	1	0	0	1,075	0	1	0	14	1	1,092
Plasmablasts	9	6	3	431	2	300	43	7	0	801
Myeloid c4 DCs pDC IRF7	0	23	0	5	266	1	0	3	10	308
B cells Memory	14	3	5	231	3	1,930	78	25	45	2,334
B cells Naive	0	0	3	23	0	495	20	0	0	541
Endothelial ACKR1	2	8	0	22	0	1	4,395	1	4	4,433
Endothelial CXCL12	1	1	0	7	0	1	1,537	0	0	1,547
Endothelial Lymphatic LYVE1	0	1	0	24	0	0	138	2	0	165
Endothelial RGS5	19	2	0	6	0	0	1,039	0	0	1,066
T cells c0 CD4+ CCR7	0	0	0	2	0	12	0	4,858	0	4,872
T cells c1 CD4+ IL7R	2	1	0	113	0	18	3	7,449	3	7,589
T cells c10 NKT cells FCGR3A	0	0	0	3	0	1	1	1,038	0	1,043
T cells c11 MKI67	2	2	0	57	0	18	2	1,346	3	1,430
T cells c2 CD4+ T-regs FOXP3	2	0	0	3	0	10	1	4,081	1	4,098
T cells c3 CD4+ Tfh CXCL13	0	0	1	1	0	3	0	2,161	0	2,166
T cells c4 CD8+ ZFP36	0	0	1	7	0	5	2	4,967	1	4,983
T cells c5 CD8+ GZMK	0	0	0	0	0	0	0	273	0	273
T cells c6 IFIT1	0	1	0	7	0	3	2	970	2	985
T cells c7 CD8+ IFNG	0	0	0	5	0	8	0	3,067	1	3,081
T cells c8 CD8+ LAG3	0	0	0	4	0	2	1	1,873	2	1,882
T cells c9 NK cells AREG	1	0	0	3	0	5	0	1,748	0	1,757
Cycling Myeloid	0	0	0	18	0	8	3	15	384	428
Myeloid c0 DC LAMP3	12	0	0	29	0	2	4	3	59	109
Myeloid c1 LAM1 FABP5	1	0	0	9	0	0	10	13	1,859	1,892
Myeloid c10 Macrophage 1 EGR1	1	0	1	11	0	0	1	4	1,976	1,994
Myeloid c11 cDC2 CD1C	0	1	0	4	0	1	0	3	310	319
Myeloid c12 Monocyte 1 IL1B	0	0	0	14	0	0	3	7	1,254	1,278
Myeloid c2 LAM2 APOE	4	2	0	19	0	0	18	1	1,010	1,054
Myeloid c3 cDC1 CLEC9A	0	0	0	0	0	0	1	2	150	153
Myeloid c5 Macrophage 3 SIGLEC1	0	1	0	0	0	0	0	0	24	25
Myeloid c7 Monocyte 3 FCGR3A	0	0	0	0	0	0	0	0	53	53
Myeloid c8 Monocyte 2 S100A9	0	0	0	5	0	0	0	1	859	865
Myeloid c9 Macrophage 2 CXCL10	0	1	0	0	0	1	0	1	439	442
Total	5,206	5,622	948	25,960	271	2,851	7,351	35,353	8,620	92,182

Table 18 100k breast cancer data (iteration 3): compare cell type subsets with the 9-clustering

4 Discussion and conclusions

Our approach has yielded results compatible with published work. The four clusters of the Zhengmix4eq data set agree very closely with the ground truth labels.

For the Zhengmix8eq data, the failure of our method to match all of the ground truth labels, specifically for T-cells, seems to be typical. Lun [7] mentions "various flavours of T cells" as examples of "cell types that are difficult to separate." The recent paper by Mullan et al. [36] is concerned with the failure of (unsupervised) clustering to correctly identify T-cell subtypes.

Results for the monocytes suggest that our approach will not find clusters in homogeneous data.

The comparison of our results for the PBMC data with those in [19] suffers because the ten k-means clusters summarized in (the paper's) Figure 3 are evidently not publicly available. In a November 2018 exchange posted on a 10x Genomics Github page [37], it was asked "... whether the celltype data in the last column of this file '68k_pbmc_barcodes_annotation' are the 'true' cell populations that can serve as the benchmark to evaluate the clustering accuracy of certain clustering method like kmeans?" Zheng replied that the "68k_pbmc_barcodes_annotation represents our best attempt to annotate cell clusters at the time of the publication. I'd advise you to use the purified populations (check the rds files posted here) to check clustering accuracy." The referenced annotation file lists eleven cell types, consistent with Figure 3j. As noted in Section 3.5 - Clustering results for individual data sets, our effort to use the 10x Genomics k-means clustering code gave clusters of apparent different sizes than implied by Figure 3b. (The adjusted Rand index comparing the k-means clusters with the correlation-based cell type assignments equals 0.25.)

For the retinal data, the compatibility of the 11-cluster segmentation with the published cell clusters is encouraging, as is the opportunity for future exploration of the hierarchical clusters that split cell clusters 24 (rods) and 26.

Understanding the reason for the exceptional stability of the 16-clustering of the lung data might suggest better algorithms. Even without this, it would be insightful to know why our approach essentially stops at 16 clusters while 56 cell types were reported. Although the 19-clustering is stable, two of its clusters are totally unstable and a third is formed by dividing macrophages between two clusters.

Results for the breast cancer data may point to ways to improve our process. As discussed in Section 3.5, iterating to remove UMI count outliers had a disproportionate impact on plasmablasts. We note here that two of the 26 patients account for 70% of the plasmablasts and that no plasmablasts were reported for twelve patients. The distribution of normal epithelial cells is also very uneven: one patient accounts for 45%; none were reported for thirteen patients. It may be worthwhile to explore why our approach – despite the compatibility with published results summarized in Tables 17 and 18 – does not separate normal from cancer epithelial cells, as well as why the two smallest clusters (2 and 4, the latter extremely unstable) align well with two cell type subsets.

Several methodological issues call for further study. These include

- Considering alternative stability criteria: our proposals are preliminary – to illustrate the methodology and prompt suggestions for improvement
- Having better criteria to decide when unstable clusters render a stable clustering inadmissible for downstream analysis. Requiring that unstable clusters have fewer than 500 cells is an expedient.
- Calculating the distances between points in the Euclidean representation (to identify outliers). We use the scikit-learn euclidean_distances function, which is very slow for the lung and breast cancer data, with many cells in high dimensions. A faster method – perhaps an approximation using sampling – is desirable
- Exploring the variation across samples of the differential expression of analysis genes between stable clusters
- Better understanding the iterative process to remove count outliers, because of its inconsistent impact: for the retinal and breast cancer data, the third iteration gave the largest stable (or nearly-stable, for the cancer data) clusterings; for the PBMC and lung data, the first iteration was best. It is not obvious how many iterations are appropriate, nor if a point of diminishing returns can be found – to stop the process because subsequent iterations would not add value. Three iterations were used for illustration; more were not performed because of the cost of calculating the distance between points

- Exploring whether – as speculated in Section 2.8 - Identify and exclude cell and gene outliers for iterative analysis – for a UMI count matrix to admit a stable clustering, it is necessary that the distributions of UMI counts (or of lower-dimension derived data such as the Euclidean representation) in all sufficiently large samples agree

Our work complements that of Levine and Domany [1], who proposed evaluating a clustering's stability by comparing it with clusterings of random samples.

The requirement that results be replicable is fundamental to this study. We propose that a clustering appropriate for downstream analysis should be consistent with those obtained for random samples of half of the cells. Our implementation is not necessary; any clustering pipeline can be used. It is a matter of preparing random half-samples of the input counts, processing each, and comparing their clusterings, and clusters, with those found for the full set of cells.

Data availability

Zhengmix4eq/8eq

Loaded from the Bioconductor DuoClustering2018 package [18]

68k PBMC

Downloaded from the 10x Genomics website
https://www.10xgenomics.com/datasets/fresh-68-k-pbm-cs-donor-a-1-standard-1-1-0

Select 'Explore Data' then 'Output and supplemental files.' Specifying the gene/cell matrix (filtered) 124 MB file yields the fresh_68k_pbmc_donor_a_filtered_gene_bc_matrices.tar file, which contains the files matrix.mtx, barcodes.tsv, and genes.tsv with data for 32,738 genes and 68,579 cells

K-means clustering of the 68,579 cells was performed using an excerpt from the program main_process_68k_pbmc.R posted in
https://github.com/10XGenomics/single-cell-3prime-paper/tree/master/pbmc68k_analysis

This program, along with the util.R utility program and two RDS files accessible with links in the README file have been placed in our Github repository (see Code availability below) in the sub-folder 10x_Genomics_material of the 68k_PBMC folder

CD14 Monocytes

Downloaded from the 10x Genomics website

25k retinal

These instructions are from https://github.com/berenslab/umi-normalization, associated with [15]

Counts
- Visit https://www.ncbi.nlm.nih.gov/geo/
- Search for GSE63472
- Download GSE63472_P14Retina_merged_digital_expression.txt.gz (50.7 MB)

- Extract to GSE63472_P14Retina_merged_digital_expression.txt

Cluster annotations
- Download from http://mccarrolllab.org/wp-content/uploads/2015/05/retina_clusteridentities.txt

65k lung

Metadata krasnow_hlca_10x_metadata.csv and counts krasnow_hlca_10x_UMIs.csv were downloaded from https://www.synapse.org/#!Synapse:syn21041850. The count file is dense: it explicitly contains all zero counts

100k breast cancer

As documented in [23], data are available through the GEO Series accession number GSE176078. GSE176078_Wu_etal_2021_BRCA_scRNASeq.tar.gz contains the files count_matrix_sparse.mtx, count_matrix_barcodes.tsv, count_matrix_genes.tsv, and metadata.csv

Code availability

The Github repository https://github.com/victorkleb/scRNA-seq_stable_clust contains 7 folders, one for each data set studied. Each folder contains the pipeline for analyses described in this article.

Appendix A Closed-form expression for the mean sum of squares of Pearson residuals with a Poisson model

For convenience, this is copied – with minor changes – from [26]. The formula is sparse – involving only entries of the UMI count matrix and row and column sums. Hence, the SSQ of genes' Pearson residuals can be computed sparsely; the dense Pearson residuals matrix is not needed.

We use the notations

X_{gc} = the UMI count for gene g in cell c
$G_g = \sum_c X_{gc}$ the total count for gene g
$\pi_{c|g} = X_{gc}/G_g$ the fraction of the total count of gene g that is in cell c
$D_c = \sum_g X_{gc}$ the total count for cell c – its sequencing depth
$T = \sum_{gc} X_{gc}$ the total of all counts in the matrix
C = the total number of cells in the matrix
$D = T/C$ the mean sequencing depth of the matrix
$\overline{D_c} = D_c/D$ the normalized sequencing depth of cell c

The Pearson residuals are

$$R_{gc} = (X_{gc} - \widehat{\mu}_{gc}) \Big/ \sqrt{\widehat{\mu}_{gc}}$$

with the maximum likelihood estimate

$$\widehat{\mu}_{gc} = \frac{G_g \cdot D_c}{T}$$

Squaring gives

$$R_{gc}^2 = \frac{(X_{gc} - \widehat{\mu}_{gc})^2}{\widehat{\mu}_{gc}}$$

$$= \frac{X_{gc}^2}{\widehat{\mu}_{gc}} - 2 \cdot X_{gc} + \widehat{\mu}_{gc}$$

Summing over all cells c gives $\boldsymbol{S}_g$, the sum of squares of Pearson residuals for gene g:

$$\boldsymbol{S}_g = \sum_c \left(\frac{X_{gc}^2}{\widehat{\mu}_{gc}} - 2 \cdot X_{gc} + \widehat{\mu}_{gc} \right)$$

$$= \sum_c \left(\frac{X_{gc}^2}{\widehat{\mu}_{gc}} \right) - 2 \cdot G_g + G_g \qquad \text{because } \sum_c X_{gc} = \sum_c \widehat{\mu}_{gc} = G_g$$

$$= \sum_c \left(\frac{X_{gc}^2}{\widehat{\mu}_{gc}} \right) - G_g$$

$$= \sum_c \left(\frac{X_{gc}^2}{\frac{G_g \cdot D_c}{T}} \right) - G_g \qquad \text{using } \widehat{\mu}_{gc} = \frac{G_g \cdot D_c}{T} \tag{*}$$

$$= \left(\sum_c \frac{\pi_{c|g} \cdot X_{gc}}{D_c} \right) \cdot T - G_g \qquad \text{since } \pi_{c|g} = X_{gc}/G_g$$

Dividing by the number of cells in the UMI matrix, C, gives the mean sum of squares of Pearson residuals for gene g:

$$\boldsymbol{M}_g = \left(\sum_c \frac{\pi_{c|g} \cdot X_{gc}}{D_c} \right) \cdot \frac{T}{C} - \frac{G_g}{C}$$

$$= \left(\sum_c \frac{\pi_{c|g} \cdot X_{gc}}{D_c} \right) \cdot D - \frac{G_g}{C} \qquad \text{because } T/C = D$$

$$= \left(\sum_c \pi_{c|g} \cdot \frac{X_{gc}}{D_c/D} \right) - \frac{G_g}{C}$$

$$= \left(\sum_c \pi_{c|g} \cdot \frac{X_{gc}}{\overline{D_c}} \right) - \frac{G_g}{C} \quad \text{since } \overline{D_c} = D_c/D \text{ – the normalized sequencing depth of cell } c$$

Observations:

- The summation includes *only* cells on which gene g has nonzero counts
- The mean value of the denominators $\overline{D_c}$ *over all cells in the UMI count matrix* is 1
- The subtracted term $\frac{G_g}{C}$ is negligible for lowly expressed genes
- Since $\sum \pi_{c|g} = 1$, the summation is the convex combination of the ratios $X_{gc}/\overline{D_c}$. This provides intuition for the fact that many values of $\boldsymbol{M}_g$ are near 1. They are convex combinations of terms that are often close to 1: most nonzero counts X_{gc} equal 1 and the mean of the $\overline{D_c}$ equals 1

Appendix B Closed-form expression for a cell's contribution to the sum of squares of a gene's Pearson residuals

Continuing the above derivation from equation (*) identifies each cell's contribution to S_g

$$\boldsymbol{S}_g = \sum_c \left(\frac{X_{gc}^2}{\frac{G_g \cdot D_c}{T}} \right) - G_g$$

$$= \sum_c \left(\frac{X_{gc}^2 \cdot T}{G_g \cdot D_c} - X_{gc} \right) \qquad \text{because } G_g = \sum_c X_{gc}$$

$$= \sum_c \left(\frac{X_{gc}}{G_c} \cdot \frac{T}{D_c} - 1 \right) \cdot X_{gc}$$

Denoting the contribution of cell c to $\boldsymbol{S}_g$ as K_{gc}

$$K_{gc} = \left(\frac{X_{gc}}{G_g} \cdot \frac{T}{D_c} - 1 \right) \cdot X_{gc}$$

$$= \left(\left(\frac{X_{gc}}{G_g} \right) \Big/ \left(\frac{D_c}{T} \right) - 1 \right) \cdot X_{gc}$$

Observations:

- The numerator X_{gc}/G_g is the fraction of the count of gene g in cell c
- The denominator D_c/T is the fraction of the total UMI count in cell c
- Hence, the ratio compares the contribution of cell c to the count of g with the cell's contribution to the matrix's total count
- If this ratio equals 1, there is *no* contribution to the SSQ of the PR of g
- If it is less than 1, the cell's contribution to the SSQ of the PR is *negative*

References

[1] Levine, E., Domany, E.: Resampling method for unsupervised estimation of cluster validity. Neural computation **13**(11), 2573–2593 (2001)

[2] Ben-Hur, A., Elisseeff, A., Guyon, I.: A stability based method for discovering structure in clustered data. In: Biocomputing 2002, 6-17. World Scientific (2001)

[3] Dudoit, S., Fridlyand, J.: A prediction-based resampling method for estimating the number of clusters in a dataset. Genome biology **3**, 1–21 (2002)

[4] Lange, T., Roth, V., Braun, M.L., Buhmann, J.M.: Stability-based validation of clustering solutions. Neural computation **16**(6), 1299–1323 (2004)

[5] Tibshirani, R., Walther, G.: Cluster validation by prediction strength. Journal of Computational and Graphical Statistics **14**(3), 511–528 (2005)

[6] Hennig, C.: Cluster-wise assessment of cluster stability. Computational Statistics & Data Analysis **52**(1), 258–271 (2007)

[7] Lun, A.: Bootstrapping for cluster stability (2019)

[8] Peyvandipour, A., Shafi, A., Saberian, N., Draghici, S.: Identification of cell types from single cell data using stable clustering. Scientific reports **10**(1), 12349 (2020)

[9] Tang, M., Kaymaz, Y., Logeman, B.L., Eichhorn, S., Liang, Z.S., Dulac, C., Sackton, T.B.: Evaluating single-cell cluster stability using the jaccard similarity index. Bioinformatics **37**(15), 2212–2214 (2021)

[10] Patterson-Cross, R.B., Levine, A.J., Menon, V.: Selecting single cell clustering parameter values using subsampling-based robustness metrics. BMC bioinformatics **22**, 1–13 (2021)

[11] Meilă, M.: Good (k-means) clusterings are unique (up to small perturbations). Journal of Multivariate Analysis **173**, 1–17 (2019)

[12] Von Luxburg, U., *et al.*: Clustering stability: an overview. Foundations and Trends® in Machine Learning **2**(3), 235–274 (2010)

[13] Zhang, J.M., Kamath, G.M., David, N.T.: Valid post-clustering differential analysis for single-cell rna-seq. Cell systems **9**(4), 383–392 (2019)

[14] Duò, A., Robinson, M.D., Soneson, C.: A systematic performance evaluation of clustering methods for single-cell rna-seq data. F1000Research **7**, 1141 (2020)

[15] Lause, J., Berens, P., Kobak, D.: Analytic pearson residuals for normalization of single-cell rna-seq umi data. Genome biology **22**(1), 1–20 (2021)

[16] Grabski, I.N., Street, K., Irizarry, R.A.: Significance analysis for clustering with single-cell rna-sequencing data. Nature Methods **20**(8), 1196–1202 (2023)

[17] Nicol, P.B., Miller, J.W.: Model-based dimensionality reduction for single-cell rna-seq using generalized bilinear models. Biostatistics **26**(1), 024 (2025)

[18] Duò, A., Soneson, C.: DuoClustering2018: Data, Clustering Results and Visualization Functions From Duò et al (2018). (2024). R package version 1.7.1

[19] Zheng, G.X., Terry, J.M., Belgrader, P., Ryvkin, P., Bent, Z.W., Wilson, R., Ziraldo, S.B., Wheeler, T.D., McDermott, G.P., Zhu, J., *et al.*: Massively parallel digital transcriptional profiling of single

cells. Nature communications **8**(1), 1–12 (2017)

[20] Nicol, P.B., Miller, J.W.: Model-based dimensionality reduction for single-cell rna-seq using generalized bilinear models. bioRxiv, 2023–04 (2024)

[21] Macosko, E.Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., Tirosh, I., Bialas, A.R., Kamitaki, N., Martersteck, E.M., *et al.*: Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. Cell **161**(5), 1202–1214 (2015)

[22] Travaglini, K.J., Nabhan, A.N., Penland, L., Sinha, R., Gillich, A., Sit, R.V., Chang, S., Conley, S.D., Mori, Y., Seita, J., *et al.*: A molecular cell atlas of the human lung from single-cell rna sequencing. Nature **587**(7835), 619–625 (2020)

[23] Wu, S.Z., Al-Eryani, G., Roden, D.L., Junankar, S., Harvey, K., Andersson, A., Thennavan, A., Wang, C., Torpy, J.R., Bartonicek, N., *et al.*: A single-cell and spatially resolved atlas of human breast cancers. Nature genetics **53**(9), 1334–1347 (2021)

[24] Seurat Guided Clustering Tutorial. https://satijalab.org/seurat/articles/pbmc3k_tutorial.html. [Accessed 07-Nov-2024]

[25] Hao, Y., Stuart, T., Kowalski, M.H., Choudhary, S., Hoffman, P., Hartman, A., Srivastava, A., Molla, G., Madad, S., Fernandez-Granda, C., Satija, R.: Dictionary learning for integrative, multimodal and scalable single-cell analysis. Nature Biotechnology (2023) https://doi.org/10.1038/s41587-023-01767-y

[26] Klebanoff, V.: Normalization and gene selection for single-cell rna-seq umi data using sampling-adjusted sums of squares of pearson residuals with a poisson model. bioRxiv, 2023–12 (2023)

[27] Erichson, B.: Optimal Hard Threshold for Matrix Denoising. https://github.com/erichson/optht. [Accessed 04-Sep-2025]

[28] Gavish, M., Donoho, D.L.: The optimal hard threshold for singular values is 4/sqrt(3). IEEE Transactions on Information Theory **60**(8), 5040–5053 (2014)

[29] Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: Analysis and an algorithm. Advances in neural information processing systems **14** (2001)

[30] Verma, D., Meila, M.: A comparison of spectral clustering algorithms. University of Washington Tech Rep UWCSE030501 **1**, 1–18 (2003)

[31] Hastie, T., Tibshirani, R., Friedman, J., et al.: The elements of statistical learning, 2nd Edition. Springer. [Online; Corrected 12th printing - Jan 13, 2017; Accessed 01-Aug-2020] (2009)

[32] Hennig, C., Meila, M.: Cluster analysis: an overview. Handbook of cluster analysis **1**, 20 (2015)

[33] Von Luxburg, U.: A tutorial on spectral clustering. Statistics and computing **17**(4), 395–416 (2007)

[34] Blighe, K., Lun, A.: PCAtools: PCAtools: Everything Principal Components Analysis. (2025). https://doi.org/10.18129/B9.bioc.PCAtools . R package version 2.20.0. https://bioconductor.org/packages/PCAtools

[35] Cooley, S.M., Hamilton, T., Aragones, S.D., Ray, J.C.J., Deeds, E.J.: A novel metric reveals previously unrecognized distortion in dimensionality reduction of scrna-seq data. Biorxiv, 689851 (2019)

[36] Mullan, K.A., Valkiers, S., Vrij, N., Li, C., Verbandt, S., Pu, T., Meysman, P.: Where single-cell transcriptomics fails t cells: The misuse of unsupervised clustering for t-cell annotation.

ImmunoInformatics, 100063 (2025)

[37] https://github.com/10XGenomics/single-cell-3prime-paper/commit/989aeed58745e01fe13acc439bdc19c2c185a1aa. [Accessed 05-Jan-2025]