Finding stable clusterings of single-cell RNA-seq data

Victor Klebanoff
vfklebanoff@live.com

Abstract

A sampling-based method that can identify stable (replicable) clusterings of cells for data presented as UMI counts is described. The structure of the processing pipeline is conventional: filter and transform counts, restrict to data for highly variable genes, reduce dimensionality, and cluster cells.

Divisive (binary) hierarchical spectral clustering is used. We propose what may be a novel method to map a clustering tree to a set of nested clusterings.

For spectral clustering, non-zero affinities are defined for points that are k-nearest neighbors (k is an input parameter). The affinity equals the inverse of the distance between the points. This led to exploration of the variation of the distance between points (that represent cells in low-dimensional Euclidean space) that are k-nearest neighbors. Variation can be large – ranging over three orders of magnitude for one data set studied. This may have implications for other clustering schemes.

Given a set of points, Ng, Jordan, and Weiss' algorithm is used to divide it into two clusters. Repeating for each daughter cluster – and its descendants – generates a clustering tree. Because the algorithm splits a set of points into two subsets, the points are mapped to two-dimensional Euclidean space for clustering. The clusters' separation is measured by a quantity, H, calculated in two dimensions, which is formally identical to the F-statistic, equal to the between-cluster sum of squares divided by the within-cluster sum of squares, scaled by degrees of freedom. The larger H is, the greater the separation between the clusters.

Each cluster corresponds to a node of the clustering tree. Dividing a set of points into two subsets corresponds to defining two daughter nodes. Assign the length of the branch between a node and each of its daughters to equal 1/H. That is, the larger the separation between the daughter clusters, the closer they are (viewed as nodes) to their parent node in the tree. Nodes' distances from the root define the mapping of the tree to a set of nested clusterings.

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 to those of the full data set (restricted to the sample). If differences are small for all samples, the clustering may be considered stable.

The method supports single-factor batch correction.

Preliminary analysis not discussed here suggests that differential expression can contribute to evaluating stability.

Keywords: Gene expression, Single-cell RNA-seq, UMI counts, Data preparation, Batch correction, Outliers, Clustering, Differential expression

1 Introduction

Although the stability – or replicability – of clusterings of gene expression data has been of interest for over twenty years, there is no consensus on how to find stable clusterings of single-cell RNA-seq data.

Early work using sampling to evaluate the effect of data variation on clusterings includes Ben-Hur et al. [1], Dudoit and Fridlyand [2], and Tibshirani and Walther [3].

In the general setting (i.e. not limited to gene expression data) Levine and Domany [4] proposed clustering a data set, then multiple samples, and comparing each sample's clustering to the clustering of the full data set (restricted to the sample). Lange et al. [5] proposed a stability measure to quantify the reproducibility of clusterings on samples and Hennig [6] discussed evaluating stability by using samples, as well as by replacing points with noise or by jittering. Hennig 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 motivated the work presented here: *If data for twice as many cells were available, would clustering results change and, if so, how?* While this is unknowable, the problem can be approached by reversing the question: *Does using only half of the cells give very different results?* 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 to the clusterings found with the subsets alone. Repeat for multiple samples; if agreement is good for enough samples, the clustering may tentatively 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 to the clusterings of the full set of cells gives stability estimates. Following Meilă [11] and von Luxburg [12], clusterings are compared using the *misclassification error distance* (MED); von Luxburg refers to it as the minimal matching distance. Its distribution across samples characterizes a clustering's stability. Additionally, each *cluster's misclassification error rate* (CMER) is computed with each sample. Its distribution characterizes a cluster's stability.

We considered, but do not use here, the *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 may provide independent criteria to assess stability. If genes' expression levels differ consistently between pairs of clusters across all (or most) samples, this may be additional evidence that a clustering is stable. Comparing samples may exempt this approach from Zhang's critique [13] of performing clustering and differential expression with the same data. Such an analysis is out of scope for the current discussion, but may provide the framework for future work.

2 Process outline

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 *no: adding more cells* ($\boldsymbol{C_2}$) *does not change the result enough to matter.* Since the sample labels are arbitrary, we also require that processing $\boldsymbol{U_2}$ give a clustering that agrees with the restriction of the clustering of $\boldsymbol{C}$ to $\boldsymbol{C_2}$.

We propose doing this with multiple samples $\boldsymbol{C_s}$. Use 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. If the values of MED are small, then each cluster's misclassification error rate (CMER) should be calculated for all samples. We suggest that if these are small for all (or enough) clusters, the clustering should be deemed stable – or subjected to further scrutiny.

2.1 Prepare data: from UMI counts to clustering input

The first step in the pipeline is rudimentary filtering. We follow the model of the Seurat Guided Clustering Tutorial [14] to exclude cells with high count contributions from mitochondrial genes and restrict to genes with nonzero counts on at least 50 cells.

Samples of cells $\boldsymbol{C_s}$ and the associated count matrices $\boldsymbol{U_s}$ are prepared. We use 20 pairs of complementary samples, 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 earlier paper [15] which is included as Appendix A of this paper for convenience.

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 all samples. For the data sets studied, this retained between 565 and 1,704 genes, which we refer 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, we used Erichson's optht program [16], which implements Gavish and Donoho's algorithm [17]. The singular value decomposition (SVD) produces the low rank *Euclidean representation* of the residuals matrix. For the data sets studied, the rank of the Pearson residuals ranges between 11 and 434. 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.

2.2 Cluster the Euclidean representation: algorithmic considerations

To cluster points in the Euclidean representation, we initially considered the Leiden algorithm due to its use in Seurat[18]. This presents difficulties because of the need to vary the resolution parameter to obtain all clusterings of a range of sizes. Suppose, for example, 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 size 3 through 9. Although programming this is manageable, the number of clusters can fail to increase monotonically with resolution. As a result, some clusterings were not generated for some samples, or even for the set of all cells. Because computations included 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 that has given attractive results. Although we did not conduct an exhaustive study, it appears that the hierarchical approach is more likely to deliver stable clusterings than an alternative that outputs clusterings of each size.

We use Ng, Jordan, and Weiss' algorithm [19] 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, restricted to k-nearest neighbors (kNN). This restriction keeps the affinity matrix sparse – and 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.

In [20] Meilă recommends that outliers be identified and removed before performing spectral clustering. 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 68k PBMC data set (details are in Section 3.5): nearest neighbor distance ranges from 1.3 to 294 (Table 3). 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.

Given two points such that one is a k-nearest neighbor of the other for a specified value of k (we used $k = 64$) we define the affinity as the inverse of the distance between them. The affinity of the distant pair is 0.005; for the points closer to each other, it equals 0.5.

Returning to Meilă's suggestion, we propose excluding not only nearest neighbor outliers, but also small groups of isolated points. For example, suppose that points P_1 and P_2 are one unit apart, but that the distance to their 2nd-nearest neighbor is 50. P_1 and P_2 constitute an isolated pair of points, which might be excluded as outliers (depending on analytic criteria described 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 time consuming for tens of thousands of points in high dimensions.) Ranking the distance 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 independently 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*.

Having identified highly variable genes, calculated their Pearson residuals, reduced dimensionality, and excluded Euclidean outliers, divisive hierarchical clustering is performed to produce a clustering tree. The maximum tree depth and minimum node (cluster) size are specified as termination criteria.

The process begins by using spectral clustering to split the set of points (in the Euclidean representation) into two clusters. Each cluster is split in turn. The process continues until stopping criteria are satisfied: the maximum specified depth is reached or splitting would produce a cluster smaller than the minimum permitted size.

2.3 Assign lengths to the branches of a clustering tree; map the tree to a set of nested clusterings

At each non-terminal node, the (identical) lengths of the branches to its two daughter nodes are specified as a quantity that prioritizes the split at the node. The higher the priority, the shorter the branch. 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 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 split the set of points at their parent. Continue until the clusters corresponding to all non-terminal nodes are split.

To prioritize the split at a node, we define a quantity, H, which is formally identical to the F-statistic. The larger the value of H, the higher the priority that should be assigned to the split, hence the shorter the length of the branches from the node to its daughters. The length of each branch is set to 1/H.

H is calculated in two-dimensional Euclidean space, unlike the F-statistic, defined in one dimension. To split a set of points $\{P\}$ into two clusters, Ng, Jordan, and Weiss' algorithm first computes the top two eigenvectors of the normalized graph Laplacian matrix. Denoting these as $\boldsymbol{e_i}$, $(i = 1, 2)$, each point P is mapped to a coordinate e_i^P of the eigenvector $\boldsymbol{e_i}$. Each coordinate pair (e_1^P, e_2^P) is viewed as a point in two-dimensional Euclidean space and normalized to length 1 as a point (u_1^P, u_2^P) on the unit circle. K-means clustering splits the points on the circle into two clusters. The cluster label of (u_1^P, u_2^P) is assigned to P.

Between-cluster and within-cluster sums of squares are calculated (in one dimension) for each set of numbers $\{e_i^P\}$, $(i = 1, 2)$. These are added together and H is defined as the ratio of the sums, scaled for degrees of freedom. The larger H is, the greater the separation between the clusters. One motivation for this approach is the use of Ward's linkage with hierarchical clustering by Grabski et al.[21].

2.4 Characterize the stability of a clustering and its clusters

Clustering has been described for the set of all cells. A similar process is applied to each sample, using results found for the full set of cells: the same analysis genes are used to compute Pearson residuals, which are reduced by SVD to the same rank, and the same Euclidean outlier cells are excluded.

The stability of each clustering of size n is evaluated for, typically, $2 \leq n \leq 25$. Given n, the clustering of $\boldsymbol{C_s}$ is compared to the restriction of the clustering of $\boldsymbol{C}$ to $\boldsymbol{C_s}$ by computing the misclassification error distance (MED). Following Lange et al.[5] – as pointed out by von Luxburg [12] – the MED is normalized against 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: 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 and every sample of cells $\boldsymbol{C_s}$.

- For cluster $\boldsymbol{c}$ of cells in $\boldsymbol{C}$, let $\boldsymbol{c_s}$ denote its restriction to cells in $\boldsymbol{C_s}$
- Consider the cluster labels assigned to the cells in $\boldsymbol{c_s}$ in the clustering of $\boldsymbol{C_s}$ (calculated with its cells alone)
- If they are not equal to $\boldsymbol{c}$, the cells are misclassified in $\boldsymbol{C_s}$
- Define the cluster misclassification error rate (CMER) of $\boldsymbol{c}$ as the fraction of cells in $\boldsymbol{c_s}$ that are misclassified

If the values of a clustering's MED are small for all samples and if the values of each cluster's CMER are also small for all samples, we believe that it is appropriate to judge the clustering, and clusters, as stable.

Because consistent cluster labels are assigned to all samples of cells $\boldsymbol{C_s}$ for clusterings of a specified size n, differential expression (of analysis genes) may help evaluate stability.

- Suppose that for sample s the differential expression of gene g between clusters $\boldsymbol{c_1}$ and $\boldsymbol{c_2}$ is measured as the T-statistic comparing the Pearson residuals of g calculated with the count sample $\boldsymbol{U_s}$ in the two clusters
- The T-statistics can be compared for each gene across all samples
- If they agree for all (or, perhaps, enough) genes, samples, and pairs of clusters, this supports a decision that the clustering is stable

2.5 Identify cell and gene outliers in the UMI count matrix: iterative analyses

We recognize this section might be expected at the start of the process outline, not the end. It is placed here because we proceeded iteratively: (1) analyzing the input data as described above, (2) then excluding outliers from the count matrix, analyzing and comparing to initial results, and (3) excluding additional outliers and analyzing again. Three iterations – listed later in this section – were performed for each of the four large data sets (PBMC, retinal, lung, and breast cancer).

Our earlier paper [15] includes a toy example showing the impact of an outlier cell on the value of S_g. The set of cells is split into two samples. The value of S_g calculated for the subset containing the outlier is nine times larger than the value for the complementary sample.

Although we have not prepared examples to explore the effect on clustering of outlier cells such as in the toy example, we expect that they are impediments to stable clusterings – and 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 that may not yet be known.

This led us to consider properties that a count matrix should have if the distributions of its samples agree. We describe two. Both involve $S_g(s)$.

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 g. A value near one corresponds to consistent values of $S_g(s)$. A large value, due to $S_g(s_1) \gg S_g(s_2)$ for samples s_1 and

s_2, indicates a difference between the distributions of the counts of g in the samples. We suggest that such g should be excluded as outliers.

The second property suggested by the toy example involves 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 should be excluded as an outlier. 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 over all genes and samples. We propose that cells for which this is large be excluded as outliers.

The three iterations performed for the large data sets were:

1. Data preparation and analysis as described in Sections 2.1 through 2.4
2. Excluding outlier cells and genes from the count matrix as described in this section and analyzing as described in Sections 2.1 through 2.4
3. Starting with the count matrix prepared for the second iteration, excluding cells dropped as Euclidean outliers during the iteration, then excluding cell and gene outliers as described in this section, and analyzing the resulting count matrix as described in Sections 2.1 through 2.4

3 Results and discussion

3.1 Data sets

Seven publicly available data sets were analyzed (Table 1). The Zhengmixeq data were prepared to evaluate clustering algorithms [22]. The PBMC data were studied by Zheng et al. in their seminal paper. Lause et al. studied the retinal data [23] (Supp Fig 2, in Additional file 1). Grabski et al. [21] studied the lung data. Nicol and Miller [24] compared a clustering of the breast cancer data produced by their algorithm to results reported by Wu et al. They used the monocytes (all cells of the same type) to illustrate features of their method. 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 [25]
Zhengmix8eq	(same)	3,994	15,716	(same)
68k PBMC	10x Genomics	68,579	32,738	Zheng et al. [26]
CD14+ Monocytes	10x Genomics	2,612	32,738	used by Nicol and Miller [24]
25k retinal	GSE63472	24,769	22,292	Macosko et al. [27]
65k lung	synapse.org	65,662	26,485	Travaglini et al. [28]
100k breast cancer	GSE176078	100,064	29,733	Wu et al. [29]

Table 1 Data sets used

3.2 Impact of data filtering and reduction

Table 2 summarizes the impact of data filtering and reduction.

- The first two columns list the number of cells and genes retained after preliminary filtering. The Pearson residuals matrix has a column for each cell. Compared to the counts in Table 1, the number of genes is smaller because genes with nonzero counts on fewer than 50 cells were deleted
- Column 3 lists the number of analysis genes – genes that are highly variable in all samples and the full data set. This is the number of rows of the Pearson residuals matrix
- Column 4 lists the rank of the Pearson residuals matrix calculated by the optht program
- Column 5 lists the number of cells retained for clustering after excluding Euclidean outliers

Data set	preliminary filtering cells	genes	analysis genes	estimated rank	cells 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 cancer	100,064	21,354	1,704	434	98,681

Table 2 Impact of data reduction

3.3 A gene with large S_g may not have large $S_g(s)$ for all samples s

The plots in Figure 1 show the relation between the mean SSQ of Pearson residuals (M_g) and the number of cells with nonzero counts for genes retained after preliminary filtering. Black points represent genes retained for downstream analysis: S_g and $S_g(s)$ are among the 2,000 largest when calculated with all cells and with all samples, respectively. In particular, some genes that are highly variable in the full set of data (large M_g) are not highly variable in all samples (red points with large vertical coordinates).

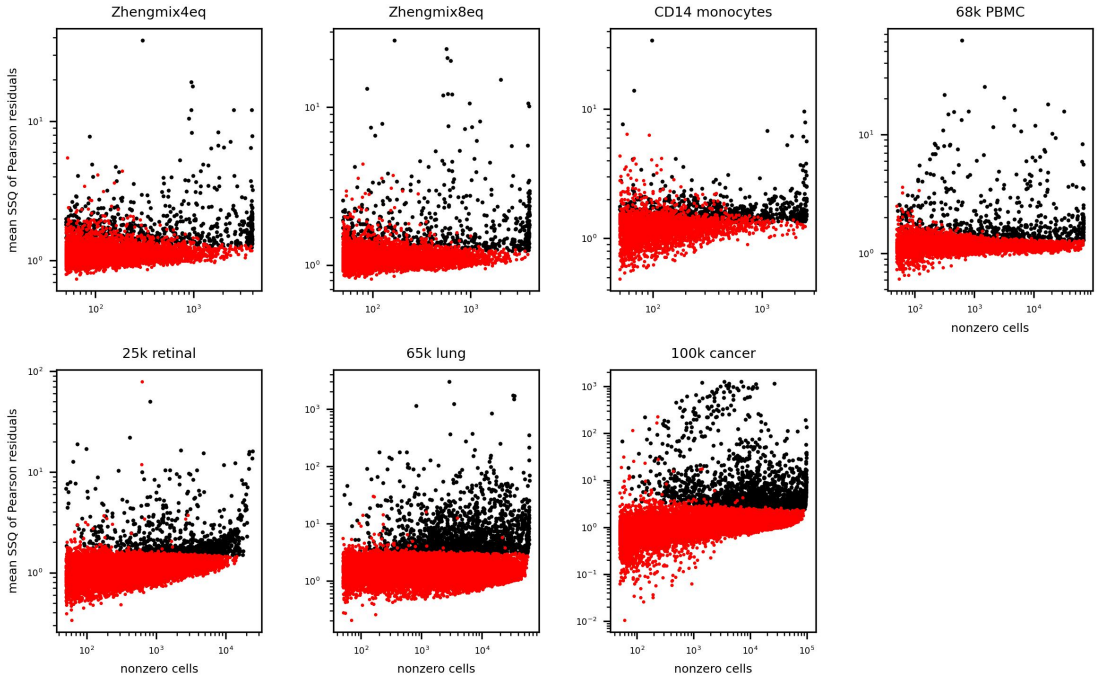

Figure 1 Genes satisfying preliminary filtering: mean SSQ of Pearson residuals (M_g) vs. number of cells with nonzero counts; black points identify analysis genes retained for downstream analysis

3.4 Analytic estimates of the rank of the Pearson residuals

The ranks of the Pearson residuals matrices were estimated analytically with the optht program. Four other programs were also 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-specified 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 for the seven data sets are displayed in Figure 2. We use the format of the screeplot program in the Bioconductor PCAtools package [30]. Red vertical lines indicate the program's estimates.

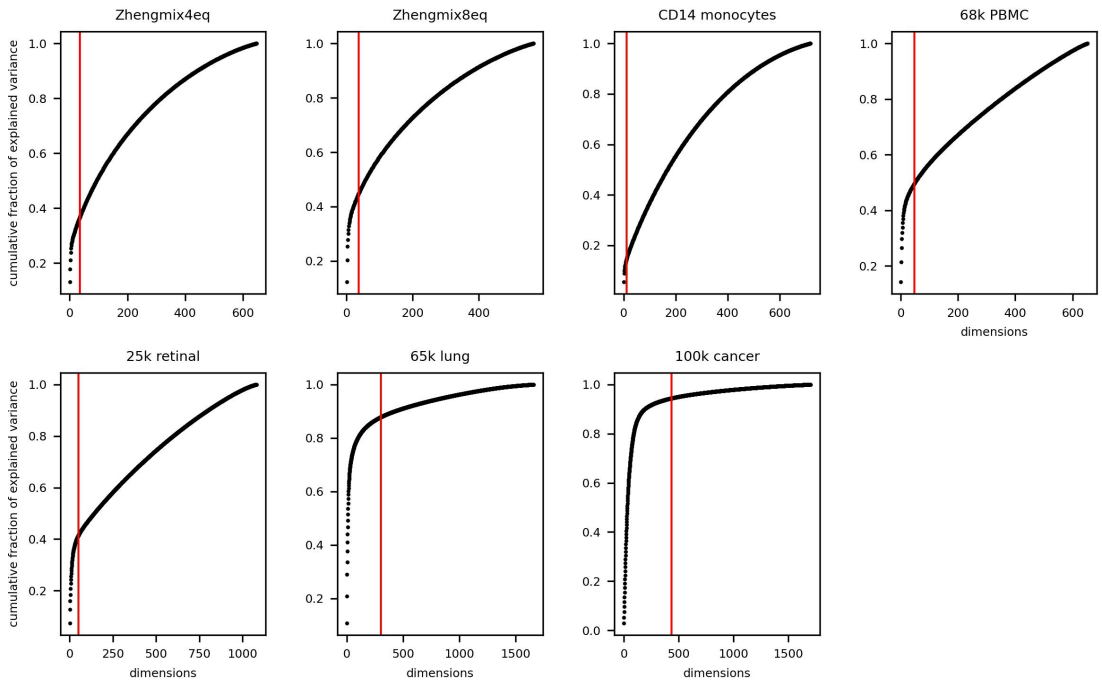

Figure 2 Scree plots: red vertical lines indicate the rank calculated by optht

3.5 Variation of kNN distance

Our interest in the relation between kNN and Euclidean distance was motivated by Meilă's recommendation to exclude outliers before performing spectral clustering and the work of Cooley et al. [31]

This section includes results for the 68k PBMC and 100k breast cancer data sets. Supplement 1 contains tables for the other five data sets. The variation of kNN distance in the Euclidean representation of the 68k PBMC data is summarized in Table 3. The first column contains statistics for nearest neighbor distance, 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 3 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 4. 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 4 68k PBMC data: distributions of k-nearest neighbor distances after excluding Euclidean outliers

For the 100k breast cancer data, the variation is greater. Before excluding Euclidean outliers, nearest neighbor distance varies by a factor of 580 (Table 5). Even after excluding 1.4% of the cells, variation is by a factor of 50 (Table 6).

	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 5 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 6 100k breast cancer data: distributions of k-nearest neighbor distances after excluding Euclidean outliers

3.6 Results for individual data sets

Our approach to reviewing results is exploratory. Though patterns emerge from the survey in this section, we believe that more work is necessary to define an algorithm to identify clusterings that will generally be accepted as stable.

As described in Sections 2.4 and 4.4, the distributions of MED and CMER characterize clustering and cluster stability. For each data set, the distributions of MED are tabulated for a range of clusterings – of size 2-10 for the three small data sets; of size 2-25 for the four large ones. We informally review each table and select a clustering for exploration, using criteria such as "this is the largest clustering for which all, or most, MED are smaller than a specified threshold."

After selecting a clustering, CMER values are tabulated for each cluster. In some cases, all of these are small; these situations seem unambiguous – the clusters (hence, the clustering) would be considered stable by most analysts. The clusterings explored for the two Zhengmixeq data sets are like this: MED < 0.05 with all samples and CMER < 0.10 with all but one sample-cluster instance (the exception is one cluster of the Zhengmix8eq data having CMER ≥ 0.10 with one sample). However, for some of the large data sets, there are instances of CMER ≥ 0.90 for clusterings that might reasonably be considered stable. It is not obvious how many large CMER values can be allowed in a stable cluster.

Zhengmix4eq

The data set contains counts for 15,568 genes and 3,994 cells. They represent four cell types; ground truth assignments were 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.

Table 7 summarizes the distribution of normalized MED for the clusterings of size 2-10. Columns list the number of samples with MED in the specified ranges. Since 40 samples were used, each row's total equals 40. For the 4-clustering, normalized MED < 0.01 with all samples; values this small for all samples were rarely found in these analyses. All other clusterings have 3 or more samples with MED ≥ 0.10.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	37	0	0	0	0	0	3	0
3	37	0	0	0	0	0	3	0
4	40	0	0	0	0	0	0	0
5	2	9	0	0	29	0	0	0
6	0	10	6	0	24	0	0	0
7	0	0	14	2	24	0	0	0
8	0	0	3	12	24	1	0	0
9	0	0	0	6	17	17	0	0
10	0	0	0	0	40	0	0	0

Table 7 Zhengmix4eq data: distribution of normalized MED for clusterings of size 2-10

Table 8 summarizes the distribution of normalized CMER for the 4-clustering. Columns list the number of samples with CMER in the specified ranges. CMER < 0.02 for all clusters and samples. For the data sets discussed in this paper, these are exceptionally small values.

		range: normalized Cluster Misclassification Error Rate							
cluster	cells	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	982	40	0	0	0	0	0	0	0
1	1,011	25	15	0	0	0	0	0	0
2	999	40	0	0	0	0	0	0	0
3	910	40	0	0	0	0	0	0	0

Table 8 Zhengmix4eq data: distribution of normalized CMER for 4 clusters

Table 9 compares our method's 4-clustering to the ground truth cell types. Like many other methods, ours gives good results for these data.

	hierarchical cluster				
ground truth	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 cell types to the 4-clustering

Zhengmix8eq

The data set contains counts for 15,716 genes and 3,994 cells. They represent eight cell types; ground truth assignments were 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.

Table 10 summarizes the distribution of normalized MED for the clusterings of size 2-10. The 7-clustering is the largest with MED < 0.05 with all samples. Of the others, only the 4 and 5-clusterings have smaller MED with all samples. We explore the 7-clustering because MED are small and its size is close to 8 – the number of ground truth cell types.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	1	0	0	0	0	12	1	26
3	2	0	0	0	0	0	38	0
4	40	0	0	0	0	0	0	0
5	40	0	0	0	0	0	0	0
6	10	3	0	0	0	27	0	0
7	3	34	3	0	0	0	0	0
8	0	5	14	9	12	0	0	0
9	0	0	14	12	13	1	0	0
10	0	0	2	6	31	1	0	0

Table 10 Zhengmix8eq data: distribution of normalized MED for clusterings of size 2-10

Table 11 summarizes the distribution of normalized CMER.

		range: normalized Cluster Misclassification Error Rate							
cluster	cells	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	419	14	18	8	0	0	0	0	0
1	448	1	6	21	12	0	0	0	0
2	492	15	7	12	5	1	0	0	0
3	931	5	13	21	1	0	0	0	0
4	590	40	0	0	0	0	0	0	0
5	496	40	0	0	0	0	0	0	0
6	530	40	0	0	0	0	0	0	0

Table 11 Zhengmix8eq data: distribution of normalized CMER for 7 clusters

Table 12 compares the 7-clustering to the 8 ground truth cell types. Results are very accurate for cd56.nk, b.cells, and cd14.monocytes; less accurate for naive.cytotoxic and memory.t cells. The three other cell types – regulatory.t, cd4.t.helper, and naive.t – are commingled in clusters 2 and 3.

ground truth	hierarchical cluster 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 12 Zhengmix8eq data: compare ground truth cell types to the 7-clustering

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.

Table 13 summarizes the distribution of normalized MED for the clusterings of size 2-25.

clustering	range: normalized Misclassification Error Distance 0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	36	0	0	0	0	4	0	0
3	12	19	5	0	0	0	4	0
4	1	24	11	0	0	4	0	0
5	1	30	9	0	0	0	0	0
6	0	17	22	0	1	0	0	0
7	0	3	36	0	1	0	0	0
8	0	0	1	1	38	0	0	0
9	0	0	0	2	38	0	0	0
10	0	0	0	14	26	0	0	0
11	0	0	0	25	15	0	0	0
12	0	0	1	25	14	0	0	0
13	0	0	0	18	22	0	0	0
14	0	0	0	1	39	0	0	0
15	0	0	0	0	39	1	0	0
16	0	0	0	0	38	2	0	0
17	0	0	0	1	39	0	0	0
18	0	0	0	1	39	0	0	0
19	0	0	0	0	40	0	0	0
20	0	0	0	0	36	4	0	0
21	0	0	0	0	33	7	0	0
22	0	0	0	0	32	8	0	0
23	0	0	0	0	35	5	0	0
24	0	0	0	0	32	8	0	0
25	0	0	0	0	24	16	0	0

Table 13 68k PBMC data: distribution of normalized MED for clusterings of size 2-25

The 7-clustering is the largest with MED < 0.05 with all but one sample; we explore it further. (For all larger clusterings, MED ≥ 0.05 with all samples – with two exceptions: the 8 and 12-clusterings, each having MED < 0.05 with one sample.)

Normalized CMER are summarized in Table 14. With two exceptions, CMER < 0.10 for all clusters and samples: over 90% of cells in cluster 0 are misclassified with one sample and 10-25% of cells in cluster 6 are misclassified with one sample.

We are not ready to judge the 7-clustering as unstable because CMER ≥ 0.90 with one sample in cluster 0. However, there is no ambiguity for the clusterings of size 8-25. The distribution of CMER was reviewed for all of them. Each clustering includes a 296-cell cluster having CMER ≥ 0.90 with all samples. We believe that all of these clusterings should be considered unstable.

		range: normalized Cluster Misclassification Error Rate							
cluster	cells	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	5,846	7	20	12	0	0	0	0	1
1	19,677	0	3	25	12	0	0	0	0
2	6,081	0	3	12	25	0	0	0	0
3	9,959	38	1	1	0	0	0	0	0
4	17,560	0	5	25	10	0	0	0	0
5	4,158	40	0	0	0	0	0	0	0
6	4,511	36	2	1	0	1	0	0	0

Table 14 68k PBMC data: distribution of normalized CMER for 7 clusters

To attempt to compare the clusterings found by our method to published results, the input data were clustered (K-means) using code from a 10xGenomics 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 Zheng et al. paper [26]. 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 (using fewer cells, as described above) were matched to these, none of the 176 cells in the smallest cluster were included – filtering discarded all of them.

Table 15 compares the 9 K-means clusters to the 7 clusters. The Adjusted Rand Index equals 0.63.

	hierarchical cluster							
K-means cluster	0	1	2	3	4	5	6	Total
7	5,536	1,188	24	3	484	8	11	7,254
4	259	16,934	2	1	1,010	20	10	18,236
5	6	2	5,098	543	2,577	18	9	8,253
1	0	0	1	3,903	0	2	1	3,907
6	0	0	827	5,485	2	12	4	6,330
2	45	1,553	69	5	10,638	32	88	12,430
3	0	0	60	14	2,848	9	28	2,959
10	0	0	0	3	1	4,057	26	4,087
9	0	0	0	2	0	0	4,334	4,336
Total	5,846	19,677	6,081	9,959	17,560	4,158	4,511	67,792

Table 15 68k PBMC data: compare 9 K-means clusters produced with 10x Genomics code to the 7-clustering

Two iterations to exclude cell and gene outliers were performed as described in Section 2.5. The second iteration yielded 66,421 cells for clustering. Each clustering of size 8-25 includes a 267-cell cluster having CMER $\geq$ 0.90 with all samples. Hence, we believe that all of these clusterings should be judged unstable.

The third iteration yielded 63,281 cells for clustering. Table 16 summarizes the distribution of normalized MED for the clusterings of size 2-25.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	39	0	0	0	0	0	0	1
3	40	0	0	0	0	0	0	0
4	30	10	0	0	0	0	0	0
5	5	34	1	0	0	0	0	0
6	0	32	7	0	0	1	0	0
7	0	18	20	0	2	0	0	0
8	0	18	17	0	5	0	0	0
9	0	0	39	0	1	0	0	0
10	0	0	35	4	1	0	0	0
11	0	0	26	2	12	0	0	0
12	0	0	23	10	7	0	0	0
13	0	0	16	12	12	0	0	0
14	0	0	2	11	27	0	0	0
15	0	0	0	2	37	1	0	0
16	0	0	0	3	37	0	0	0
17	0	0	0	0	40	0	0	0
18	0	0	0	0	40	0	0	0
19	0	0	0	0	40	0	0	0
20	0	0	0	1	39	0	0	0
21	0	0	0	1	38	1	0	0
22	0	0	0	2	36	2	0	0
23	0	0	0	2	36	2	0	0
24	0	0	0	0	38	2	0	0
25	0	0	0	0	40	0	0	0

Table 16 68k PBMC data – third iteration – after excluding outlier cells and genes from the UMI count matrix: distribution of normalized MED for clusterings of size 2-25

We believe that the clusterings of size 14-25 should be judged unstable. While we have no reason to consider any clustering of size 10-13 unstable, we explore the 9-clustering.

- The 9-clustering is the largest having MED $<$ 0.05 with all but one sample
- The 14-clustering includes a 426-cell cluster having CMER $\geq$ 0.90 with 37 samples
- Each clustering of size 15-25 includes a 307-cell cluster having CMER $\geq$ 0.90 with all samples

The distribution of normalized CMER for the 9-clustering is displayed in Table 17. For cluster 4, CMER $\geq$ 0.90 with one sample. There are more large CMER values than for the 7-clustering (Table 14). Here, there are eleven sample-cluster instances with 0.10 $\leq$ CMER $<$ 0.25; the 7-clustering has one. Also, here there is one instance with 0.25 $\leq$ CMER $<$ 0.50. Supplement 2 includes CMER tables for the clusterings of size 10-13.

cluster	cells	range: normalized Cluster Misclassification Error Rate							
		0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	4,008	23	17	0	0	0	0	0	0
1	5,306	35	3	1	1	0	0	0	0
2	18,818	1	4	32	3	0	0	0	0
3	5,855	11	17	11	1	0	0	0	0
4	6,058	9	16	12	2	0	0	0	1
5	10,803	0	0	6	23	11	0	0	0
6	5,956	0	2	28	10	0	0	0	0
7	2,901	38	1	0	0	0	1	0	0
8	3,576	40	0	0	0	0	0	0	0

Table 17 68k PBMC data – third iteration – after excluding outlier cells and genes from the UMI count matrix: distribution of normalized CMER for 9 clusters

Table 18 compares the 9 K-means clusters to the 9 hierarchical clusters. The Adjusted Rand Index equals 0.67.

K-means cluster	hierarchical cluster									Total
	0	1	2	3	4	5	6	7	8	
1	3,372	28	0	0	0	0	1	0	0	3,401
6	193	5,167	0	0	2	0	591	0	5	5,958
4	1	0	16,371	298	345	697	3	10	19	17,744
7	1	0	1,058	5,499	39	442	25	13	5	7,082
3	5	7	0	0	2,693	32	49	15	3	2,804
2	3	0	1,388	51	2,752	7,725	65	86	23	12,093
5	433	104	1	7	227	1,907	5,222	5	14	7,920
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	4,008	5,306	18,818	5,855	6,058	10,803	5,956	2,901	3,576	63,281

Table 18 68k PBMC data – third iteration – after excluding outlier cells and genes from the UMI count matrix: compare 9 K-means clusters produced with 10x Genomics code to the 9-clustering

Comparing this to the 7-clustering (Table 15)

- Filtering outliers reduced the number of cells in K-means cluster 9 from 4,336 to 2,772
- K-means clusters 4,5,7, 9 and 10 continue to correspond to individual hierarchical clusters
- K-means clusters 1 and 6 are combined in the 7-clustering but are mapped to separate hierarchical clusters in the 9-clustering
- Discrimination between K-means clusters 2 and 3 is improved: in the 7-clustering, most of the cells of K-means cluster 2 are in hierarchical cluster 4, along with nearly all cells of K-means cluster 3 and smaller shares of other K-means clusters. In the 9-clustering, K-means cluster 3 continues to be commingled – primarily with cells of K-means cluster 2, but the majority of cells of K-means cluster 2 correspond to hierarchical cluster 5

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.

Table 19 summarizes the distribution of normalized MED for the clusterings of size 2-10. None of the other data sets have so many large values for clusterings of size 10 or smaller. 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.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	0	0	0	1	6	8	10	15
3	0	0	0	0	7	11	22	0
4	0	0	0	0	0	35	5	0
5	0	0	0	0	0	17	23	0
6	0	0	0	0	0	9	31	0
7	0	0	0	0	0	11	29	0
8	0	0	0	0	0	15	25	0
9	0	0	0	0	0	3	37	0
10	0	0	0	0	0	9	31	0

Table 19 CD14+ Monocytes: distribution of normalized MED for clusterings of size 2-10

25k retinal

The data set contains counts for 22,292 genes and 24,769 cells. Thirty-nine cell clusters were reported (Figure 5D of [27]). 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. 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.

Table 20 summarizes the distribution of normalized MED for the clusterings of size 2-25.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	35	5	0	0	0	0	0	0
3	37	3	0	0	0	0	0	0
4	0	1	0	0	39	0	0	0
5	3	1	0	0	3	33	0	0
6	0	18	6	0	14	2	0	0
7	0	19	19	2	0	0	0	0
8	0	0	33	5	1	1	0	0
9	0	0	29	8	3	0	0	0
10	0	0	8	14	17	1	0	0
11	0	0	0	24	14	2	0	0
12	0	0	0	13	24	3	0	0
13	0	0	0	1	36	3	0	0
14	0	0	0	14	26	0	0	0
15	0	0	0	3	35	2	0	0
16	0	0	0	5	31	4	0	0
17	0	0	0	0	35	5	0	0
18	0	0	0	0	24	16	0	0
19	0	0	0	0	16	24	0	0
20	0	0	0	0	17	23	0	0
21	0	0	0	0	12	28	0	0
22	0	0	0	0	6	34	0	0
23	0	0	0	0	0	40	0	0
24	0	0	0	0	1	39	0	0
25	0	0	0	0	0	40	0	0

Table 20 25k retinal data: distribution of normalized MED for clusterings of size 2-25

The 7-clustering is the largest having normalized MED < 0.10 with all samples; we explore it further. Normalized CMER are summarized in Table 21.

		range: normalized Cluster Misclassification Error Rate							
cluster	cells	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	9,718	26	5	9	0	0	0	0	0
1	6,017	7	10	16	4	3	0	0	0
2	2,169	0	9	30	1	0	0	0	0
3	2,864	12	25	3	0	0	0	0	0
4	1,119	2	7	12	10	8	1	0	0
5	1,453	29	11	0	0	0	0	0	0
6	761	40	0	0	0	0	0	0	0

Table 21 25k retinal data: distribution of normalized CMER for 7 clusters

Table 22 displays the relation between the 7 clusters and the reported cell clusters: hierarchical clusters 0 and 1 include the majority of rods (cell cluster 24), while cones (cell cluster 25) are concentrated in

cluster 4. To evaluate whether dividing rods between two clusters is appropriate or spurious, we anticipate using differential expression analysis. Note that

- The 8-clustering is formed by subdividing cluster 0 – yielding a 3-way split of rod cells
- Each clustering of size 13-17 includes at least one cluster (of 400 or more cells) having CMER $\geq$ 0.90 with 16 or more samples
- Each clustering of size 18-25 includes a 238-cell cluster having CMER $\geq$ 0.90 with all 40 samples

We believe that the clusterings of size 13-25 should not be considered stable because each includes a cluster having large values of CMER with many samples. Differential expression may help to evaluate the clusterings of size 7-12.

cell cluster	hierarchical cluster 0	1	2	3	4	5	6	Total
24	9,476	5,879	91	123	9	18	62	15,658
27	33	19	281	3	0	25	0	361
28	10	5	258	2	1	15	0	291
29	36	13	277	1	3	5	2	337
30	11	1	325	2	0	9	1	349
31	7	2	261	4	0	9	0	283
32	1	4	185	1	0	3	0	194
33	24	13	414	8	4	40	2	505
37	5	2	39	0	0	0	0	46
1	11	3	0	83	0	0	1	98
2	1	2	0	183	0	0	0	186
3	0	1	0	183	0	0	0	184
4	1	1	0	38	0	0	0	40
5	1	0	0	47	0	0	0	48
6	1	0	0	115	1	0	0	117
7	0	1	3	190	1	0	0	195
8	1	1	1	75	0	0	0	78
9	4	0	0	226	0	1	0	231
10	1	1	2	85	0	0	0	89
11	2	0	2	127	0	0	0	131
12	12	2	0	152	2	1	0	169
13	0	0	0	29	0	0	0	29
14	0	2	0	4	0	0	0	6
15	1	0	0	33	0	0	0	34
16	0	1	5	165	3	2	0	176
17	3	3	1	227	0	1	0	235
18	0	2	0	41	0	0	0	43
19	0	2	0	84	1	0	0	87
20	20	13	3	217	0	2	0	255
21	6	8	4	122	1	1	0	142
22	11	5	4	137	1	0	1	159
23	9	2	2	153	3	1	0	170
39	0	0	0	2	0	0	0	2
25	27	21	10	1	1,089	3	0	1,151
26	1	3	1	1	0	1,314	0	1,320
38	0	0	0	0	0	1	0	1
34	2	5	0	0	0	2	692	701
Total	9,718	6,017	2,169	2,864	1,119	1,453	761	24,101

Table 22 25k retinal data: compare 37 of 39 reported cell clusters to the 7-clustering

The iterations described in Section 2.5 were performed to remove gene and cell outliers. Results are consistent with the ones reported here.

65k lung

The data set contains counts for 26,485 genes and 65,662 cells. Fifty-seven cell types were reported.

Excluding data for blood cells retained 60,993 cells. Seurat was not used for filtering. Consequently, cells were not screened for high mitochondrial DNA levels. Excluding genes with nonzero counts on fewer than 50 cells retained 17,470 genes. 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 [28] describes separately clustering the lung data for each patient. Our analysis included batch correction; each of the 3 patients' data was treated as an independent batch.

Table 23 summarizes the distribution of normalized MED for the clusterings of size 2-25. The 13-clustering is the largest having MED < 0.05 with a majority of samples; we explore it further.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	36	0	2	0	0	0	1	1
3	3	0	4	27	0	5	1	0
4	5	0	4	23	4	3	1	0
5	2	0	3	15	3	15	2	0
6	3	1	6	17	5	8	0	0
7	1	0	3	28	3	5	0	0
8	2	6	20	0	10	2	0	0
9	2	1	5	2	30	0	0	0
10	5	1	27	0	7	0	0	0
11	3	1	25	0	9	2	0	0
12	0	2	13	0	22	3	0	0
13	1	4	27	0	8	0	0	0
14	1	2	17	15	5	0	0	0
15	0	2	0	9	28	1	0	0
16	0	1	0	23	14	2	0	0
17	0	1	3	29	7	0	0	0
18	0	0	17	16	7	0	0	0
19	0	0	9	27	4	0	0	0
20	0	0	10	28	2	0	0	0
21	0	0	3	32	5	0	0	0
22	0	0	4	29	7	0	0	0
23	0	0	1	17	22	0	0	0
24	0	0	0	36	4	0	0	0
25	0	0	0	35	5	0	0	0

Table 23 65k lung data: distribution of normalized MED for clusterings of size 2-25

Normalized CMER are summarized in Table 24. Three clusters have CMER $\geq$. 0.90, for a total of 11 sample-cluster instances: cluster 0 with 7 samples, cluster 4 with one sample, and cluster 9 with three samples. In cluster 12, CMER $\geq$ 0.10 with 35 of 40 samples.

This is the first case where we have seen this many instances of high CMER values. For each of the two clusterings of the 68k PBMC data that were reviewed, one cluster has CMER $\geq$ 0.90 with only one sample. This example raises the issue more forcefully: can this clustering be considered stable?

		range: normalized Cluster Misclassification Error Rate							
cluster	cells	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	5,259	0	0	18	15	0	0	0	7
1	9,790	11	29	0	0	0	0	0	0
2	5,280	27	13	0	0	0	0	0	0
3	4,804	37	3	0	0	0	0	0	0
4	4,831	20	19	0	0	0	0	0	1
5	11,335	39	0	0	0	0	1	0	0
6	2,803	12	25	3	0	0	0	0	0
7	1,910	40	0	0	0	0	0	0	0
8	2,784	40	0	0	0	0	0	0	0
9	1,932	0	9	28	0	0	0	0	3
10	455	40	0	0	0	0	0	0	0
11	1,363	28	8	3	1	0	0	0	0
12	7,568	2	3	0	0	28	4	3	0

Table 24 65k lung data: distribution of normalized CMER for 13 clusters

Because blood cells were excluded from our analysis, one cell type was eliminated. Table 25 compares the 56 retained cell types to the hierarchical clusters. Just as rods are divided between two hierarchical clusters in the retinal data (Table 22, first row), the majority of Macrophages are split between clusters 0 and 1 (third row). Differential expression may help determine if this division is appropriate or spurious.

The division of Macrophages between clusters 0 and 1 provides an insight into the instances of CMER $\geq$ 0.90 in cluster 0 with 7 samples. Examination of the confusion matrices that compare the clusters defined with all cells to the clusters defined for the 40 samples shows that the 7 high CMER values are indeed due to the misclassification of cells in cluster 0 to cluster 1 in 7 samples. The 7 samples are among the 8 with normalized MED $\geq$ 0.10. In Supplement 2, the 14-clustering is explored; it is formed by splitting cluster 12 into clusters containing 3,323 and 4,245 cells. The 4,245-cell daughter cluster is stable: CMER $<$ 0.05 with all samples. The 3,323-cell daughter is not: CMER $\geq$ 0.25 with 37 samples – in fact CMER $\geq$ 0.50 with 19. It illustrates Lun's observations on merging clusters [7].

We do not believe that any clustering of size 14-25 should be considered stable.

- As noted above, the 14-clustering includes a 3,323-cell cluster having CMER $\geq$ 0.50 with 19 samples
- Each clustering of size 15-17 includes a cluster containing more than 1,000 cells having CMER $\geq$ 0.90 with 19 or more samples
- Each clustering of size 18-25 includes a 468-cell cluster having CMER $\geq$ 0.90 with 36 or more samples

cell type	hierarchical cluster 0	1	2	3	4	5	6	7	8	9	10	11	12	Total
TREM2+ Dendritic	84	46	0	1	1	1	16	0	0	0	0	0	0	149
Ionocyte	0	14	0	2	0	1	2	0	0	0	0	0	3	22
Macrophage	5,132	9,468	1	8	3	6	66	11	2	0	0	0	4	14,701
Proliferating Macrophage	38	187	0	0	0	0	0	1	0	0	0	0	0	226
CD8+ Naive T	0	0	776	628	0	0	0	2	0	0	0	0	0	1,406
Natural Killer	0	0	4,232	15	0	0	0	0	0	0	0	0	0	4,247
Proliferating NK/T	0	0	78	22	0	0	0	1	0	0	0	0	0	101
B	0	0	0	241	0	0	0	1	0	0	0	0	0	242
CD4+ Memory/Effector T	0	0	21	2,391	0	0	0	0	0	0	0	0	2	2,414
CD4+ Naive T	0	0	0	288	0	0	0	0	0	0	0	0	0	288
CD8+ Memory/Effector T	0	0	77	693	0	0	0	0	0	0	0	0	0	770
Mesothelial	1	0	0	7	0	4	0	0	1	5	0	0	2	20
Natural Killer T	0	0	83	252	0	0	0	0	0	0	0	0	0	335
Plasma	0	0	0	93	0	0	1	0	0	0	0	0	0	94
Plasmacytoid Dendritic	0	1	0	114	0	0	1	0	0	0	0	0	0	116
Capillary Aerocyte	0	0	0	0	4,250	98	0	1	0	1	0	0	1	4,351
Capillary Intermediate 1	0	0	0	0	475	152	0	1	0	0	0	0	0	628
Artery	0	0	1	0	7	1,470	0	0	1	1	0	0	0	1,480
Bronchial Vessel 1	0	0	0	1	1	431	0	0	1	1	2	0	0	437
Bronchial Vessel 2	0	0	0	0	8	217	1	0	0	2	0	0	0	228
Capillary	0	3	2	4	54	7,321	0	1	1	0	0	0	0	7,386
Capillary Intermediate 2	0	0	0	0	7	456	0	0	1	0	0	0	0	464
Vein	0	0	0	0	7	1,141	0	0	1	0	0	0	1	1,150
Classical Monocyte	0	1	0	2	0	0	1,010	0	0	0	0	0	0	1,013
EREG+ Dendritic	0	11	0	0	0	1	129	0	0	0	0	0	0	141
IGSF21+ Dendritic	0	6	0	7	0	0	263	1	0	0	0	0	0	277
Intermediate Monocyte	0	0	0	0	0	0	187	0	0	0	0	0	0	187
Myeloid Dendritic Type 1	0	1	1	17	0	0	94	0	0	0	0	0	0	113
Myeloid Dendritic Type 2	2	7	2	6	0	0	213	0	0	0	0	0	0	230
Nonclassical Monocyte	0	0	0	0	0	0	603	0	0	0	1	0	0	604
OLR1+ Classical Monocyte	0	2	0	2	0	1	196	3	0	0	0	0	0	204
Basophil/Mast 1	0	0	0	2	0	0	1	1,336	0	0	0	0	0	1,339
Basophil/Mast 2	0	1	0	0	0	0	0	547	0	0	0	0	0	548
Airway Smooth Muscle	0	0	0	0	0	0	1	0	685	1	0	0	0	687
Neuroendocrine	0	0	0	0	0	0	0	0	3	0	0	0	0	3
Pericyte	0	2	0	1	0	0	0	0	1,600	1	0	0	1	1,605
Vascular Smooth Muscle	0	0	0	2	0	0	0	0	455	5	0	0	0	462
Adventitial Fibroblast	0	0	0	0	0	9	0	0	0	359	0	0	0	368
Alveolar Fibroblast	0	1	2	0	0	7	3	0	1	1,247	0	0	0	1,261
Fibromyocyte	0	0	0	0	0	1	0	0	30	62	0	0	0	93
Lipofibroblast	0	0	0	0	0	0	0	0	0	28	0	0	0	28
Myofibroblast	0	0	1	0	0	3	0	0	2	218	0	0	0	224
Lymphatic	0	0	0	0	1	1	0	0	0	1	452	0	0	455
Ciliated	0	13	1	0	0	1	0	0	0	0	0	1,270	28	1,313
Platelet/Megakaryocyte	2	2	0	0	0	0	0	0	0	0	0	4	3	11
Proximal Ciliated	0	0	0	0	0	0	0	0	0	0	0	88	0	88
Alveolar Epithelial Type 1	0	0	0	0	9	3	0	1	0	0	0	0	949	962
Alveolar Epithelial Type 2	0	0	2	1	2	5	1	2	0	0	0	0	3,778	3,791
Basal	0	7	0	3	1	3	5	0	0	0	0	0	388	407
Club	0	1	0	1	1	1	0	1	0	0	0	0	860	865
Differentiating Basal	0	6	0	0	0	1	7	0	0	0	0	0	229	243
Goblet	0	4	0	0	0	0	0	0	0	0	0	1	117	122
Mucous	0	1	0	0	2	0	0	0	0	0	0	0	348	351
Proliferating Basal	0	1	0	0	0	0	1	0	0	0	0	0	38	40
Proximal Basal	0	4	0	0	0	0	1	0	0	0	0	0	150	155
Signaling Alveolar Epithelial Type 2	0	0	0	0	2	0	1	0	0	0	0	0	666	669
Total	5,259	9,790	5,280	4,804	4,831	11,335	2,803	1,910	2,784	1,932	455	1,363	7,568	60,114

Table 25 65k lung data: compare 56 of 57 reported cell types to the 13-clustering

Two iterations to exclude cell and gene outliers were performed as described in Section 2.5. The results of the third iteration show that excluding outliers may not yield larger stable clusterings. Here, results are worse. Table 26 summarizes the distribution of normalized MED for the clusterings of size 2-25.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	30	0	9	0	0	0	1	0
3	0	27	9	1	0	3	0	0
4	0	7	3	0	0	30	0	0
5	0	25	5	0	5	5	0	0
6	0	25	3	7	3	2	0	0
7	0	3	3	31	3	0	0	0
8	0	3	11	25	1	0	0	0
9	0	3	12	2	23	0	0	0
10	0	3	0	0	27	10	0	0
11	0	3	0	0	14	23	0	0
12	0	0	3	0	37	0	0	0
13	0	0	3	0	37	0	0	0
14	0	0	3	0	35	2	0	0
15	0	0	2	0	33	5	0	0
16	0	0	2	1	37	0	0	0
17	0	0	0	3	36	1	0	0
18	0	0	0	3	9	28	0	0
19	0	0	1	2	0	37	0	0
20	0	0	2	1	0	37	0	0
21	0	0	1	2	0	37	0	0
22	0	0	0	1	2	37	0	0
23	0	0	0	0	3	37	0	0
24	0	0	0	0	3	37	0	0
25	0	0	0	0	3	37	0	0

Table 26 65k lung data – third iteration – after excluding outlier cells and genes from the UMI count matrix: distribution of normalized MED for clusterings of size 2-25

CMER were reviewed for the clusterings of size 6-25. Each clustering of size 7 and above includes a 369-cell cluster having CMER ≥ 0.90 with 37 samples; we believe that none should be considered stable.

100k breast cancer

The data set contains counts for 29,733 genes and 100,064 cells. Nine major cell types were reported.

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.

References in [29] and programs posted online 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.

Table 27 summarizes the distribution of normalized MED for the clusterings of size 2-25.

	range: normalized Misclassification Error Distance							
clustering	0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
2	0	4	8	5	15	8	0	0
3	0	0	2	1	5	13	19	0
4	0	0	2	3	7	25	3	0
5	0	0	14	10	8	8	0	0
6	0	0	12	8	16	4	0	0
7	0	0	11	10	16	3	0	0
8	0	0	2	4	30	4	0	0
9	0	0	0	12	25	3	0	0
10	0	0	0	5	30	5	0	0
11	0	0	0	2	37	1	0	0
12	0	0	0	1	39	0	0	0
13	0	0	0	1	34	5	0	0
14	0	0	0	2	33	5	0	0
15	0	0	0	1	19	20	0	0
16	0	0	0	0	19	21	0	0
17	0	0	0	0	20	20	0	0
18	0	0	0	0	13	27	0	0
19	0	0	0	0	8	32	0	0
20	0	0	0	0	13	27	0	0
21	0	0	0	0	15	25	0	0
22	0	0	0	0	16	24	0	0
23	0	0	0	0	10	30	0	0
24	0	0	0	0	4	36	0	0
25	0	0	0	0	3	37	0	0

Table 27 100k breast cancer data: distribution of normalized MED for clusterings of size 2-25

We explore the 7-clustering, the largest having MED < 0.10 with a majority of samples. Table 28 summarizes the distribution of normalized CMER. There are more large values than in any of the cases discussed above.

- Cluster 1: CMER ≥ 0.90 with 10 samples
- Cluster 5: CMER ≥ 0.50 with 17 samples; we believe that this cluster (and therefore the clustering) – with so many large values – should not be judged stable

All larger clusterings include a cluster having CMER ≥ 0.90 with a large number of samples. We believe that none should be considered stable.

- The 8-clustering includes an 8,640-cell cluster having CMER ≥ 0.90 with 29 samples
- Each clustering of size 9-25 includes a 289-cell cluster having CMER ≥ 0.90 with all 40 samples

cluster	cells	range: normalized Cluster Misclassification Error Rate							
		0 - 0.01	0.01 - 0.02	0.02 - 0.05	0.05 - 0.10	0.10 - 0.25	0.25 - 0.50	0.50 - 0.90	0.90 +
0	22,693	0	10	22	3	5	0	0	0
1	5,378	0	5	22	2	1	0	0	10
2	9,614	1	2	12	14	5	6	0	0
3	5,334	0	10	18	5	1	3	1	2
4	35,479	8	7	22	1	0	2	0	0
5	12,542	9	2	3	3	1	5	15	2
6	7,641	12	19	9	0	0	0	0	0

Table 28 100k breast cancer data: distribution of normalized CMER for 7 clusters

Table 29 compares the 7 clusters to the nine major cell types reported in the paper [29]. The Adjusted Rand Index equals 0.81. These differences are immediate

- Our results do not discriminate Normal from Cancer Epithelial cells, though 21% of the cancer cells are separated into cluster 1
- Our results do not discriminate between CAFs and PVL, assigning nearly all to cluster 5

major cell type	hierarchical cluster							Total
	0	1	2	3	4	5	6	
Cancer Epithelial	18,352	5,110	100	23	647	205	33	24,470
Normal Epithelial	3,632	6	7	2	17	670	1	4,335
Myeloid	60	23	9,090	303	22	50	100	9,648
B-cells	48	39	102	2,979	26	1	1	3,196
Plasmablasts	113	43	261	1,859	11	2	1	2,290
T-cells	195	117	42	149	34,673	4	10	35,190
CAFs	186	10	9	12	44	6,268	4	6,533
PVL	95	5	2	3	30	5,273	11	5,419
Endothelial	12	25	1	4	9	69	7,480	7,600
Total	22,693	5,378	9,614	5,334	35,479	12,542	7,641	98,681

Table 29 100k breast cancer data: compare 9 major cell types to the 7-clustering

Two iterations to exclude cell and gene outliers were performed as described in Section 2.5. No clusterings of size 7-25 have CMER values that suggest they are more stable than the 7-clustering reviewed above.

- Second iteration
  - The 7-clustering includes an 880-cell cluster having CMER ≥ 0.90 with 37 samples
  - In the 8-clustering, the same cluster has CMER ≥ 0.90 with 20 samples
  - The 9-clustering includes an 8,703-cell cluster having CMER ≥ 0.90 with 12 samples and a 11,598-cell cluster having CMER ≥ 0.50 with 17
  - Each clustering of size 10-25 includes a 289-cell cluster having CMER ≥ 0.90 with 39 or 40 samples
- Third iteration
  - The 7-clustering includes a 9,473-cell cluster having CMER ≥ 0.90 with 30 samples
  - Each clustering of size 8-25 includes a 271-cell cluster having CMER ≥ 0.90 with 39 or 40 samples

3.7 Next steps

If MED and CMER are found to be useful measures, applying them may require a decision rule of the form: a clustering is considered stable if MED $\leq \boldsymbol{m}$ with at least $\boldsymbol{s_1}$ samples and CMER $\leq \boldsymbol{c}$ with at least $\boldsymbol{s_2}$ samples in each cluster, for parameters to be determined.

Preliminary results comparing differential expression across samples were mentioned. This should be explored further.

One algorithmic issue should receive attention. A faster alternative to the scikit-learn euclidean_distances function for points in the Euclidean representation would be welcome. Sampling might improve efficiency without compromising accuracy, but we have not explored this.

Results of the iterative process to exclude count outliers (Section 2.5) are unsatisfying. Filtering the UMI count matrix does not give the expected results. For the 68k PBMC data, the third iteration indeed yields a larger stable clustering; for the 65k lung data, it does not. Perhaps conditions can be identified which guarantee that a UMI count matrix admits stable clusterings,

4 Methods

4.1 Batch correction

For batch correction, the basic method is modified.

Calculate S_g

- 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 $S_g(s, b)$ for sample s
- Define $S_g = \sum_b S_g(b)$ for the full set of cells and $S_g(s) = \sum_b S_g(s, b)$ for sample s
- This corresponds to Nicol and Miller's calculating a gene-level intercept for each batch

Calculate Pearson residuals

- For each batch, calculate residuals matrices for all cells and for each sample
- Concatenate them to construct matrices for downstream analysis

4.2 Divisive hierarchical clustering

This produces a set of clustering trees – one for all cells and one for each sample.

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 respective values are 10 and 200.

Sparse affinity matrices are constructed for the full set of cells and each sample. If two points are 64th-NN or closer, the affinity is set to the inverse of the distance between them in the Euclidean representation. Otherwise the affinity equals zero. Since the property of being kNN is not symmetric (i.e. if P_2 is the nearest neighbor of P_1, P_1 may not be the nearest neighbor of P_2) the affinity matrix is constructed to be symmetric. Exploratory analysis with 256th-NN did not yield more attractive clusterings – i.e. clusterings of the same size or larger with smaller values of MED and CMER.

The clustering tree is stored as a python dictionary. Each item represents a node of the tree. The key is a tuple that indexes nodes by their depth, parent, and whether it is its parent's left or right daughter. Examples

- The key for the root node is (0,0): depth 0; no parent. Keys would be simpler if python allowed tuples with a single entry, with root node's key being (0). Since this is not possible, the second element of each key tuple is the dummy value zero
- The keys for the root's daughters are (1,0,0) and (1,0,1): depth 1; parent (0,0); the third entries distinguish the daughters
- If the node with key (1,0,1) is not terminal, its daughters' keys are (2,0,1,0) and (2,0,1,1): depth 2; parent (1,0,1)

The value of a node's dictionary item is itself a dictionary containing 6 items, except for the root, which contains 5

- A list of cells (the cluster) at the node
- The distance to its parent (absent for the root)
- The distance to the root node (zero, trivially, if it is the root)
- A flag indicating that the node is terminal because splitting would produce a child containing fewer cells than the stopping criterion allows
- For non-terminal nodes: the within-cluster and between-cluster SSQ for its daughter nodes (stored as a tuple) – SSQ are calculated in the two-dimensional space (per Ng, Jordan, and Weiss' algorithm) in which clusters are defined
- A flag indicating that the node was analyzed for splitting because it does not satisfy the maximum-depth stopping criterion and is not terminal

Ng, Jordan, and Weiss' algorithm splits the set of points $\{P\}$ at a node into two clusters as follows
- Select the rows and columns of the affinity matrix corresponding to the cells at the node; calculate the normalized graph Laplacian matrix
- Compute its top two eigenvectors $\boldsymbol{e_1}$ and $\boldsymbol{e_2}$
- Map each point P to the corresponding coordinate e_i^P of each eigenvector $\boldsymbol{e_i}$
- Treat each pair (e_1^P, e_2^P) as a point in two-dimensional Euclidean space and normalize it to length 1 – mapping it to (u_1^P, u_2^P) on the unit circle
- Cluster the points (u_1^P, u_2^P) with K-means to obtain two clusters and assign the cluster label of (u_1^P, u_2^P) to P

To update the clustering tree when the set of points at a node is split
- Calculate the between-cluster and within-cluster sums of squares for each set of numbers $\{e_1^P\}$ and $\{e_2^P\}$ with the pingouin ANOVA program
- Calculate the heuristic quantity H by adding the SSQ for the two sets together and computing their ratio, scaled for degrees of freedom
- Add the between-cluster SSQ and within-cluster SSQ to the dictionary that is the node's value
- Add daughter nodes to the tree; include the distance (1/H) to their parent and to the tree's root in their dictionaries

4.3 Map a clustering tree to a set of nested clusterings

Sort nodes by their distance from the root. The root's daughters correspond to the 2-clustering. Descend to the pair of nodes that are the next closest to the root. They split the cells 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 split the set of cells at their parent node, defining the 4-clustering. Continue until the clusters corresponding to all non-terminal nodes are split.

4.4 Evaluate cluster stability

Select a range of clusterings for review. For this article, sizes $2 \leq n \leq 25$ were used for the large data sets and $2 \leq n \leq 10$ for the small ones. Given n, compare the clustering of all cells to the clusterings of the samples. Calculate the misclassification error distance (MED) between the clustering of each sample and the restriction to the sample of the clustering of all cells. As noted in Section 2.4, we follow Lange et al. [5] and normalize MED against randomized data: each sample's cluster assignments are shuffled, the resulting "clusterings" are compared to the clustering of all cells, and the mean MED is calculated. Dividing the MED values for the true data by the mean of the MEDs calculated with the randomized data yields the normalized MEDs.

Similarly, given a clustering, each cluster's misclassification error rate (CMER) is calculated for each sample. The calculations are then repeated using data with shuffled cluster assignments to obtain normalized values of CMER. The distributions of normalized MED and CMER are tabulated for review.

Supplementary information

Supplement 1 contains tables of the distributions of kNN distance between cells for 5 data sets: Zhengmix4eq/8eq, CD14 monocytes, 25k retinal, and 65k lung.

Supplement 2 contains tables of the distributions of CMER for the clusterings of size 10-13 of the 68k PBMC data. They are for data produced by the third iterative analysis – after two rounds of excluding outliers from the UMI count matrix. The supplement also includes results for the 14-clustering of the 65k lung data: tables of (1) distributions of CMER and (2) the relation between the 56 reported cell types and the 14 hierarchical clusters.

Data availability

Zhengmix4eq/8eq

Loaded from the Bioconductor DuoClustering2018 package [25]

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 – 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 util.R utility program and two RDS files accessible with links in the README file have been placed in our Github repository (see below in Code availability) in the subfolder 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 [23]

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 [29], 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 [15]. 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] Ben-Hur, A., Elisseeff, A., Guyon, I.: A stability based method for discovering structure in clustered data. In: Biocomputing 2002, pp. 6–17. World Scientific, ??? (2001)

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

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

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

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

[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] NYGC, S.L.: Single Cell RNA-seq Secondary Analysis of 68k PBMCs. https://satijalab.org/seurat/articles/pbmc3k_tutorial.html Accessed 2023-10-31

[15] 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)

[16] Erichson, B.: Optimal Hard Threshold for Matrix Denoising. https://github.com/erichson/optht Accessed 2025-09-04

[17] 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)

[18] 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

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

[20] Hennig, C., Meila, M.: Cluster analysis: an overview. Handbook of cluster analysis, 1–19 (2015)

[21] 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)

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

[23] 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)

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

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

[26] 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)

[27] 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)

[28] 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)

[29] 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)

[30] 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

[31] 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)