Computational and Structural Biotechnology Journal 21 (2023) 4044–4055

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Method article

scDemultiplex: An iterative beta-binomial model-based method for accurate demultiplexing with hashtag oligos

Li-Ching Huang [a,b], Lindsey K. Stolze [a,b], Hua-Chang Chen [a,b], Alexander Gelbard [c], Yu Shyr [a,b], Qi Liu [a,b,*], Quanhu Sheng [a,b,*]

[a] *Department of Biostatistics, Vanderbilt University School of Medicine, Nashville, TN 37232, USA*
[b] *Center for Quantitative Sciences, Vanderbilt University Medical Center, Nashville, TN 37232, USA*
[c] *Department of Otolaryngology, Vanderbilt University School of Medicine, Nashville, TN 37232, USA*

ARTICLE INFO

Keywords:
Single-cell sequencing
Hashtag oligo (HTO)
Demultiplexing
Beta-binomial

ABSTRACT

Single-cell sequencing have been widely used to characterize cellular heterogeneity. Sample multiplexing where multiple samples are pooled together for single-cell experiments, attracts wide attention due to its benefits of increasing capacity, reducing costs, and minimizing batch effects. To analyze multiplexed data, the first crucial step is to demultiplex, the process of assigning cells to individual samples. Inaccurate demultiplexing will create false cell types and result in misleading characterization. We propose scDemultiplex, which models hashtag oligo (HTO) counts with beta-binomial distribution and uses an iterative strategy for further refinement. Compared with seven existing demultiplexing approaches, scDemultiplex achieved great performance in both high-quality and low-quality data. Additionally, scDemultiplex can be combined with other approaches to improve their performance.

1. Introduction

Single-cell sequencing provides an unprecedent scale for investigating cellular heterogeneity systematically [1,13,14,20,21,25,28]. Sample multiplexing where multiple samples are pooled and sequenced together is often used in single-cell experiments to increase capacity and reduce costs, and most importantly, minimize batch effects [5]. There are two main methods used for sample multiplexing: barcode-based and single nucleotide polymorphism (SNP)-based. Barcode-based multiplexing labels samples with unique DNA barcodes (also termed hashtag oligos, HTO) using either an antibody tagged with a DNA barcode that targets the cell surface protein [19] or nucleus pore complex [6], or lipid/cholesterol-modified oligonucleotides that tag the cell membrane [15,16]. SNP-based multiplexing distinguishes multiplexed samples based on their natural genetic landscapes [10,27,7,9], but it can only be applied to genetically distinct samples, not those that share the same genetic background, such as samples from the same individuals at different developmental/lineage/experimental stages. Here, we focus solely on barcode-based sample multiplexing.

While barcoded-based multiplexing is extremely useful, it introduces artifacts due to cross- contamination in library construction and sequencing errors, giving rise to the potential that cells are not labelled only by one barcode but varying degrees of other barcodes [15,16,18]. In order to assign cells to each individual sample, demultiplexing approaches are aimed to classify cells into negatives (not true cells), singlets (cells from one sample), and multiplets (cells from two or more samples). Accurate demultiplexing is to recall as many singlets as possible and at the same time not to misrecognize negatives and multiplets as singlets. After cells are hash tagged with barcoded oligos and then pooled and processed for single-cell RNA sequencing, reads are aligned to generate gene-by-cell and HTO-by-cell count matrices. The next step is demultiplexing based on the HTO count matrix, which is critical for the downstream cell type identification and characterization. Misrecognition of negatives and multiplets as singlets will result in false cell types and misleading characterization.

Several demultiplexing approaches have been developed (Table 1). HTODemux uses k-medoids clustering to find negative clusters and then determines thresholds for classifying cells as positives or negatives by fitting a negative binomial distribution to the negative clusters [19]. MULTIseqDemux determines cutoffs based on Gaussian kernel density to classify cells [15,16]. hashedDrops identifies singlets and doublets by the log-fold changes between the most abundant HTO, the second most

[*] Corresponding authors at: Department of Biostatistics, Vanderbilt University School of Medicine, Nashville, TN 37232, USA.
E-mail addresses: qi.liu@vumc.org (Q. Liu), quanhu.sheng.1@vumc.org (Q. Sheng).

https://doi.org/10.1016/j.csbj.2023.08.013
Received 1 June 2023; Received in revised form 15 August 2023; Accepted 15 August 2023
Available online 19 August 2023

Table 1
Summary of eight demultiplexing methods.

Method	Models	Description
HTODemux [19]	• k-medoid clustering • Negative binomial distribution	• Identifies initial clusters by an k-medoids clustering and then fits a negative binomial distribution based on negative cells for further classification.
MULTIseqDemux[15, 16]	• Gaussian kernel density	• Determines the threshold based on Gaussian kernel density and the maximization of the number of singlets.
hashedDrops[12]	• Log-fold change	• classifies based on the two log-fold changes between the most abundant HTO, the second most abundant HTO, and the ambient contamination.
BFF_raw [3]	• Gaussian Kernel density	• Determine the threshold by fitting bimodal distribution based on Gaussian kernel density estimation
BFF_cluster [3]	• Gaussian Kernel density • Bimodal quantile normalization	• BFF_raw followed by bimodal quantile normalization. Then determines the threshold based on the distribution of highest and second-highest counts.
GMM-Demux [26]	• Gaussian mixture model	• Computes the posterior probability of being singlets or multiplets based on Gaussian mixture model
Demuxmix [11]	• k-means clustering • Negative binomial mixture model	• Determines initial clusters by k-means clustering and then calculates the posterior probability for positives and negatives by fitting a negative binomial mixture model.
scDemultiplex	• Gaussian mixture model • Beta-Binomial distribution	• Determines initial clusters based on Gaussian mixture model and then calculates the probability for positives by fitting Beta-Binomial distribution iteratively.

abundant HTO, and ambient contamination[12]. Bimodal Flexible Fitting (BFF), including BFF_raw and BFF_cluster, first determines thresholds based on a simple assumption that count distribution is bimodal. Then BFF_raw and BFF_cluster use determined thresholds to identify negatives, singlets and multiplets [3]. Instead of determining thresholds, GMM-Demux assumes HTO counts come from two separate sources, background and the real sample. GMM-Demux then fits the HTO counts of each sample into a Gaussian mixture model and computes the posterior probability of being singlets or multiplets [26]. Demuxmix uses two-component mixture models like GMM-Demux. Different from GMM-Demux modeling centered log ratio (CLR)-transformed counts by Gaussian distribution, demuxmix models raw HTO counts by negative binomial distribution [11] (Table 1). Overcoming the limitations of threshold-based approaches, GMM-Demux and Demuxmix achieved better performance in identifying singlets and removing multiplets [11, 26,8].

Here, we present scDemultiplex, which models HTO counts of each sample with beta-binomial distribution and calculates the probability of one cell being from the sample. Additionally, scDemultiplex uses an iterative strategy to further refine the model and the classifying result (Table 1). Compared to GMM-Demux using Gaussian mixture distribution, negative-binomial distribution in demuxmix and beta-binomial distribution fit the count nature of the HTO data better. Compared to MULTIseqDemux determining HTO-specific thresholds by an iterative strategy [15,16], scDemultiplex fits HTO counts (specific and non-specific) with beta-binomial distributions (Table 1). Therefore, sample classification in scDemultiplex is determined not only by HTO-specific but also HTO-nonspecific counts. We benchmarked scDemultiplex against seven existing demultiplexing approaches, hashedDrops, HTODemux, GMM-Demux, MULTIseqDemux, BFF_raw, BFF_cluster, and demuxmix using five real HTO datasets. The evaluation demonstrated that scDemultiplex achieved high performance in both high-quality and low-quality data. The iterative strategy makes it easy to combine scDemultiplex with other approaches, which obtained higher performance than using those approaches alone.

2. Materials and methods

2.1. scDemultiplex

Suppose the data have m cells pooled from n samples, i.e, labeled with n HTOs. The HTO count matrix x_{ij} contains the count of the HTO i in the cell j, where $i = 1, 2, \ldots, n$ and $j = 1, 2,., m$. scDemultiplex is designed to classify the cell j as a negative, a singlet, or a multiplet based on the HTO count distributions of the cell, denoted as $X_j = (x_{1j}, x_{2j}, \ldots, x_{nj})$.

2.2. Parameter estimation of beta-binomial distribution

scDemultiplex models HTO counts of each sample with beta-binomial distribution. That is, the corresponding HTO counts X_j from the sample i follow beta-binomial distribution with parameter BetaBin (n_j, α_i, β_i). where n_j is the total counts of the cell j, $n_j = \sum_{i=1}^{n} x_{ij}$. α_i and β_i are estimated from the data.

To estimate the parameters α_i and β_i for the sample i, scDemultiplex uses Gaussian mixture model to perform an initial classifying. First, HTO raw count x_{ij} is normalized by centered log-ratio (CLR) transformation, where the normalized HTO values y_{ij} is the natural log-transformed of the counts divided by the geometric mean of a specified HTO [19]:

$$y_{ij} = \log \frac{x_{ij}}{\left(\prod_{j=1}^{m} x_{ij}\right)^{\frac{1}{m}}}$$

For a given sample i, the normalized HTO values of the cell j, $Y_j = (y_{1j}, y_{2j}, \ldots, y_{nj})$ follow a mixture of two Gaussian distributions, one is a “negative” distribution coming from background and the other is a “positive” distribution deriving from the sample i. Let $Z_{j[k]}$ indicates that the cell j comes from the component k, $Z_{j[k]} \in \{0, 1\}$, $k = 1, 2$. $Z_{j[1]} = 1$ if the cell from the component 1, otherwise 0; and $Z_{j[2]} = 1$ if the cell is from the component 2, otherwise 0. Let π_k be the mixture proportions, the probability density function (*pdf*) for the mixture model is

$$f_{Y_j}(y_j) = \sum_{k=1}^{2} \pi_k f_{Y_j|Z_{j[k]}}(y_j|Z_{j[k]}), 0 < y_j < \infty$$

where the k-th component follows $N(\mu_{ik}, \sigma_{ik})$. To assist in identification of “negative” and “positive” cells, scDemultiplex applied the Expectation–maximization (EM) algorithm to estimate the parameters in the Gaussian mixture model. The fitted mixed model is then used to calculate the threshold T_i that discriminate the cell j into two groups [22]. Given a cell j, if the normalized HTO value y_{ij} greater than the threshold T_i, it is a “positive” cell; otherwise, it is a “negative” cell.

scDemultiplex models each sample with Gaussian mixture model and assign cells to every sample. If *the* cell j is identified as negative in all samples, that is, $\sum_{i=1}^{n} I\left(y_{ij} > T_i\right) = 0$, this cell is negative; if *the* cell j is recognized as positive in only one sample, e.g., y_{kj} greater than T_k, that is, $\sum_{i=1}^{n} I\left(y_{ij} > T_i\right) = 1$, the cell belongs to the sample k (singlet); otherwise the cell j is a multiplet, $\sum_{i=1}^{n} I\left(y_{ij} > T_i\right) \geq 2$. The formula is laid out below:

$$\sum_{i=1}^{n} I\left(y_{ij} > T_i\right) = \begin{cases} 0, j \in \text{negatives} \\ 1, j \in \quad \text{singlets} \\ 2+, j \in \textit{multiplets} \end{cases}$$

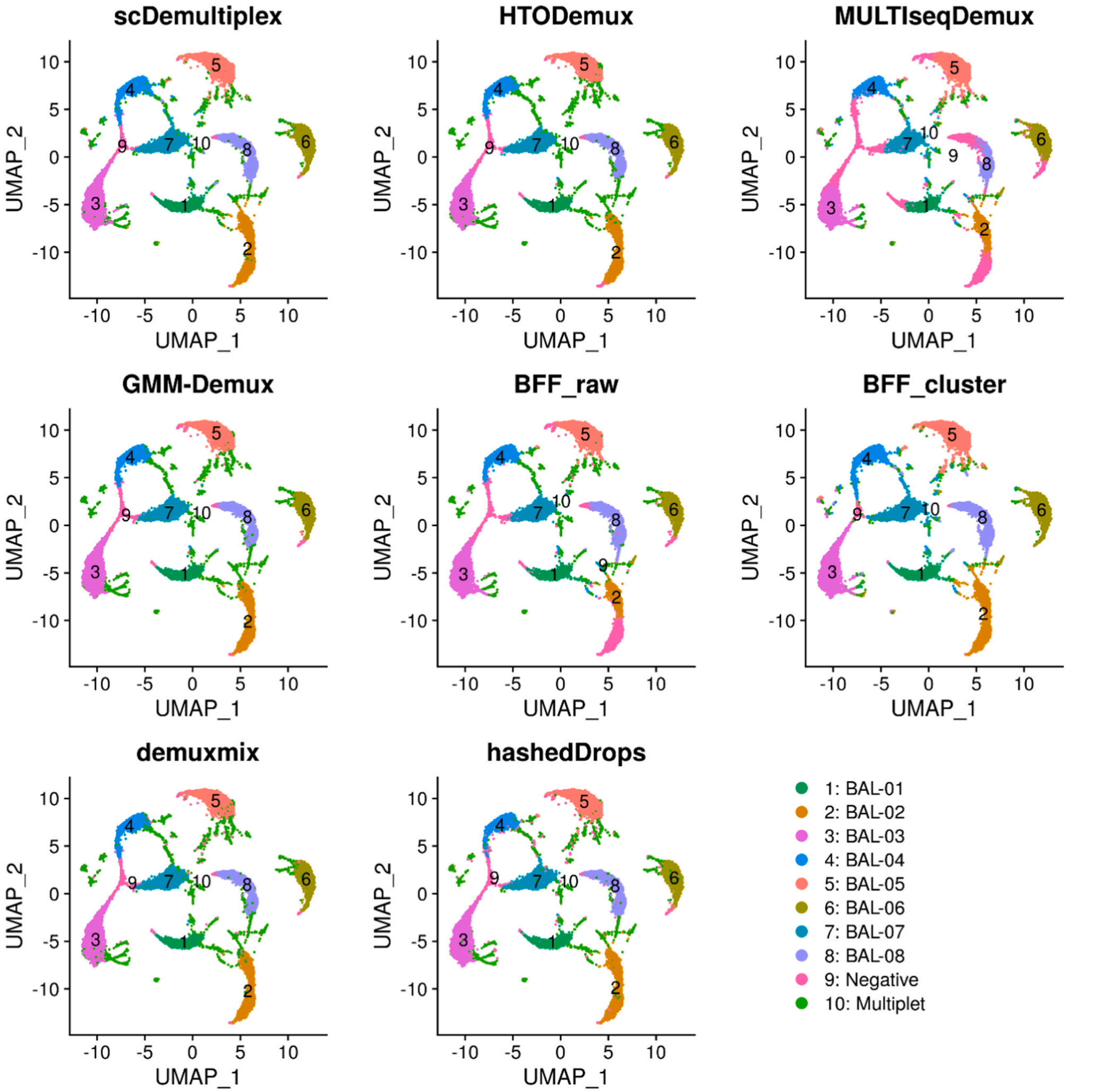


Fig. 1. UMAP plots of the batch1_c1 dataset labeled with demultiplexing results from scDemultiplex, HTODemux, MULTIseqDemux, GMM-Demux, BFF_raw, BFF_cluster, demuxmix, and hashedDrops.

where $I()$ is an indicator function. The cell assignment based on Gaussian mixture model is used to estimate the parameters α_i and β_i in the BetaBin(n_j, α_i, β_i) for the sample *i*. scDemultiplex first fits HTO raw counts with Dirichlet-multinomial distribution DirMult(n_j, $\boldsymbol{\alpha}_i$'). Condition on the total HTO counts, the raw count for each HTO can be modeled with Dirichlet-multinomial distribution, which is justified by the study [12]. The parameters $\boldsymbol{\alpha}_i$'= (α_{i1}', α_{i2}', …, α_{in}') are estimated by the method described in Lun et al. [12] using R dirmult package [23]. To simplify the calculation, scDemultiplex replaces Dirichlet-multinomial distribution with beta-binomial distribution. To fit beta-binomial distribution, scDemultiplex aggregates the n-dimensional vector X_j= *(*x_{1j} *,* x_{2j}*, …,* x_{nj} *)* into a two-dimensional vector with counts coming from the sample *i* (x_{ij}) and sum of counts from other samples $\sum_{l=1,l\neq i}^{n} x_{lj}$. scDemultiplex estimates α_i by taking $\widehat{\alpha}_i = \widehat{\alpha}_{ii}{}'$ and $\widehat{\beta}_i = \sum_{l=1,l\neq i}^{n}\widehat{\alpha}_{il}{}'$.

2.3. Demultiplexing

The probability mass function (*pmf*) of beta-binomial distribution of a given sample *i* for the cell *j* follows

$$f(x) = \binom{n_j}{x}\frac{\Gamma(\alpha_i+\beta_i)}{\Gamma(\alpha_i)\Gamma(\beta_i)}\frac{\Gamma\left(x_{ij}+\alpha_i\right)\Gamma\left(n_j-x+\beta_i\right)}{\Gamma\left(n_j+\alpha_i+\beta_i\right)}, x = 0, 1, 2, \ldots, n_j$$

where n_j is the total counts of *the* cell *j*, $n_j = \sum_{i=1}^{n} x_{ij}$. The probability of the cell *j* coming from a given sample *i*, denoted by P_{ij}, is estimated to be the tail cumulative probability $p(x >= x_{ij})$. The probability is then adjusted by controlling the false discovery rate (FDR) using the Benjamini-Hochberg (BH) method [2]. A specified FDR threshold is set at 0.1%. The cell *j* is identified as negative if the cell is not assigned to any samples, singlet if the cell is only assigned to one sample, multiplets

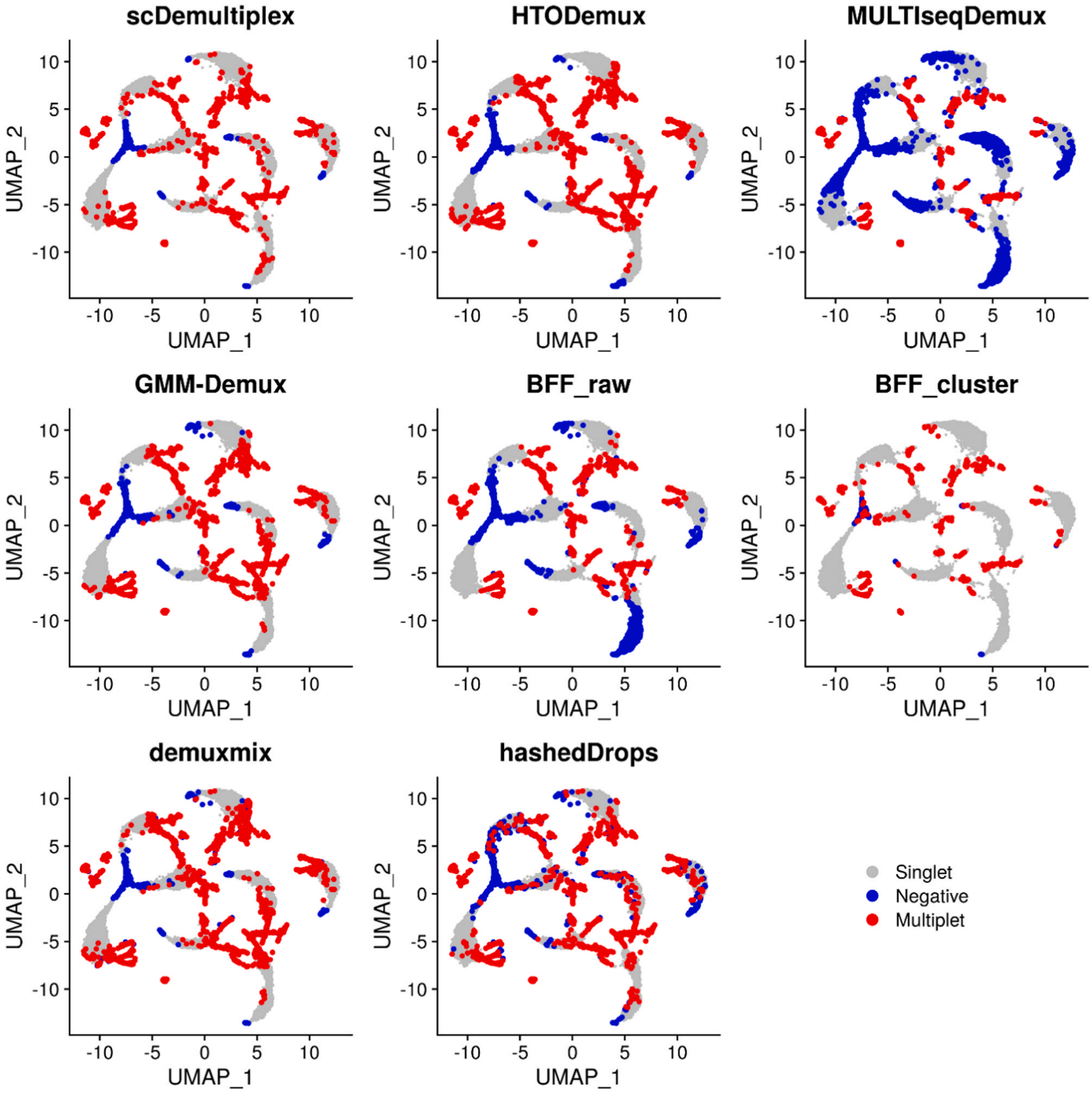


Fig. 2. UMAP plots of the batch1_c1 dataset labeled with singlets, negatives, and multiplets classified by scDemultiplex, HTODemux, MULTIseqDemux, GMM-Demux, BFF_raw, BFF_cluster, demuxmix, and hashedDrops.

if others.

$$\sum_{i=1}^{n} I\left(P_{ij} > 0.001\right) = \begin{cases} 0, j \in negatives \\ 1, j \in singlets \\ 2+, j \in multiplets \end{cases}$$

2.4. Iterative refinement of beta-binomial models and reclassifying

scDemultiplex repeats beta-binomial parameter estimations and the probability calculations n times (default: n = 10) to further refine the model and the classification. Specifically, the classifying result from the previous iteration is used to estimate parameters α and β for every sample and then the estimated parameters are utilized to calculate the probabilities of each cell belonging to each sample and classify each cell to negatives, singlets or multiplets in the next iteration.

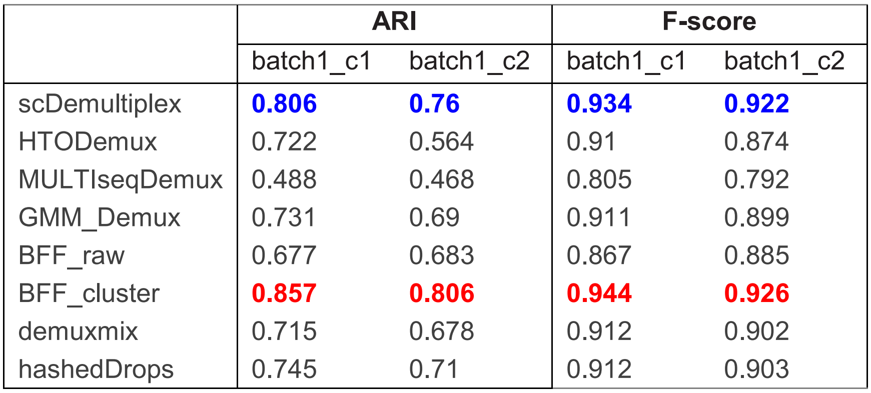
The procedure will terminate earlier if there are no reassignment or multiple types of singlets reclassified to another type of singlet (default: >= 3) during one iteration. Multiple reclassifications suggest something might go awry in the remodeling. This stopping criterion not only increases the speed in high-quality data, but also helps prevent wrong models in poor-quality data, where different types of singlets, negatives, and multiplets are indistinct from each other.

2.5. Real HTO datasets

We used five real HTO datasets to evaluate the performance of scDemultiplex, named as batch1, batch2, batch3, Barnyard, and PBMC8.

The three batches were utilized in a recent study benchmarking single-cell demultiplexing methods [8]. They contain 24 genetically distinct samples of bronchoalveolar lavage fluid. Each batch contains two captures pooled from 8 samples. Each sample was tagged with different Totalseq-A antibody-derived tag (ADT). Batch1 consists of 11,

Table 2
Adjusted rand index and F-score of eight demultiplex methods in the batch1 datasets.



	ARI		F-score	
	batch1_c1	batch1_c2	batch1_c1	batch1_c2
scDemultiplex	**0.806**	**0.76**	**0.934**	**0.922**
HTODemux	0.722	0.564	0.91	0.874
MULTIseqDemux	0.488	0.468	0.805	0.792
GMM_Demux	0.731	0.69	0.911	0.899
BFF_raw	0.677	0.683	0.867	0.885
BFF_cluster	**0.857**	**0.806**	**0.944**	**0.926**
demuxmix	0.715	0.678	0.912	0.902
hashedDrops	0.745	0.71	0.912	0.903

Note: The highest ARI and F-score among the eight approaches is highlighted in red while the second highest ARI and F-score is highlighted in blue.

900 and 12,923 cells, batch2 consists of 24,905 and 25,763 cells, and batch 3 contains 32,886 and 31,956 cells, respectively. All the datasets have been demultiplexed using genetic variants from the RNA by Vireo [8,9], which are used as "ground truth" to evaluate the accuracy of the HTO demultiplexing methods. As indicated in the study [8], cells in the batch1 are well labeled (high-quality), while those in the batch2 and batch3 are hash-tagged poorly (low-quality), highlighting demultiplexing in these two batches is more challenging.

The Barnyard dataset was downloaded from a previous study [15,16]. Briefly, by combining lipid- and cholesterol-modified oligonucleotides (LMOs, CMOs) with three separate cell lines, including Human Embryonic Kidney (HEK) cells, Jurkat T cells, and Mouse Embryonic Fibroblast (MEF) cells, twelve samples consisting of 5877 cells were multiplexed and sequenced. The 'ground truth' was obtained from the original study [15,16], which was generated by marker gene expression analysis.

The PBMC8 dataset was downloaded from a previous study [19]. The dataset comprised 15,113 cells extracted from PBMC in eight different human donors. The "ground truth" was obtained by genetic-based demultiplexing using Vireo [9].

2.6. Performance evaluation

We first evaluated the performance of demultiplexing by visual inspection of low dimensional embeddings of HTO profiles. The low dimensional embeddings was generated by the Seurat RunUMAP function using HTO counts as input [4]. In the low-dimensional embeddings, we expect to find cells from the same sample form a distinct cluster (singlet-cluster), while negative cells generally spread and/or loosely connect to clusters from singlets, and multiplets locate at the edge of singlet-clusters or form a separate cluster between two singlet-clusters. The visual distribution is useful but subjective and lack a quantitative measure for robust comparisons.

Additionally, we used two quantitative metrics to evaluate the performance, the adjusted rand index (ARI) and F-score based on the "ground truth". ARI measures the agreement between the HTO classification and the "ground truth". The higher ARI value means the better agreement, indicating more accuracy of HTO classification. ARI was calculated by the mclust package using adjustedRandIndex function [17]. The F-score is the harmonic mean of precision and recall, defined as TP/(TP+1/2(FP+FN))[8]. The higher F-score suggests better performance.

2.6.1. Implementation and Code availability

We implemented our algorithm in an R package scDemultiplex, which is publicly available at Github (https://github.com/shengqh/scDemultiplex). The code for analyzing the three batches, Barnyard, and PBMC8 datasets by the eight demultiplexing approaches is also provided at the Github repository (https://github.com/shengqh/scDemultiplex_-analysis). A web server running scDemultiplex is available at https://bioinfo.vanderbilt.edu/scdemult/.

3. Results

3.1. Application on the batch1, Barnyard, and PBMC8 datasets with high-quality

We applied scDemultiplex and seven existing demultiplexing approaches on the batch1 datasets with two captures (batch1_c1 and batch1_c2). We obtained similar results for the two captures within the batch1. The UMAP plots labeled with demultiplexing results from each method showed similar patterns (Fig. 1 and Supplementary Fig. S1). There were eight distinct and dense clusters representing eight samples, which were identified as singlets by every method (singlet-cluster). The distinct singlet clusters indicates high-quality data [8]. Negatives spread and/or loosely connected to singlet clusters, and multiplets located at the edge of singlet-clusters or formed a separate cluster between two singlet-clusters.

We further investigated negatives and multiplets identified by each method (Fig. 2 and Supplementary Fig. S2). The visual inspection found most methods obtained reasonable results, which had negatives spreading and loosely connecting to singlet-clusters, and multiplets locating at the edge of singlet-clusters. MULTIseqDemux and BFF_raw, however, misrecognized many singlets as negatives. MULTIseqDemux and BFF_raw had the greatest number of negatives identified (Supplementary Tables S1-S2). The negatives identified by MULTIseqDemux and BFF_raw extended to the center of singlet-clusters (Fig. 2 and Supplementary Fig. S2).

Table 2 listed the ARI and F-score based on the genetic "ground truth". BFF_cluster achieved the highest ARI and F-score in both two captures, followed by scDemultiplex. MULTIseqDemux, in contrast, had the lowest ARI and F-score due to its misrecognizing many singlets as negatives, especially for the BAL_02 and BAL_08 samples (Supplementary Tables S3 and S4). Other methods, such as HTODemux, GMM_Demux, demuxmix, and hashedDrops performed comparably well in the batch1.

In addition, we applied scDemultiplex and seven existing demultiplexing approaches on the Barnyard and the PBMC8 datasets. Both datasets showed distinct singlet clusters, indicating their high quality (Supplementary Figs. S3-S6). Similar to the results from the batch1, most methods performed well when data quality is high. BFF_cluster achieved the highest ARI and F-score, followed by scDemultiplex in the Barnyard dataset (Supplementary Tables S5-S7), while scDemultiplex achieved the highest ARI and F-score, followed by BFF_raw and

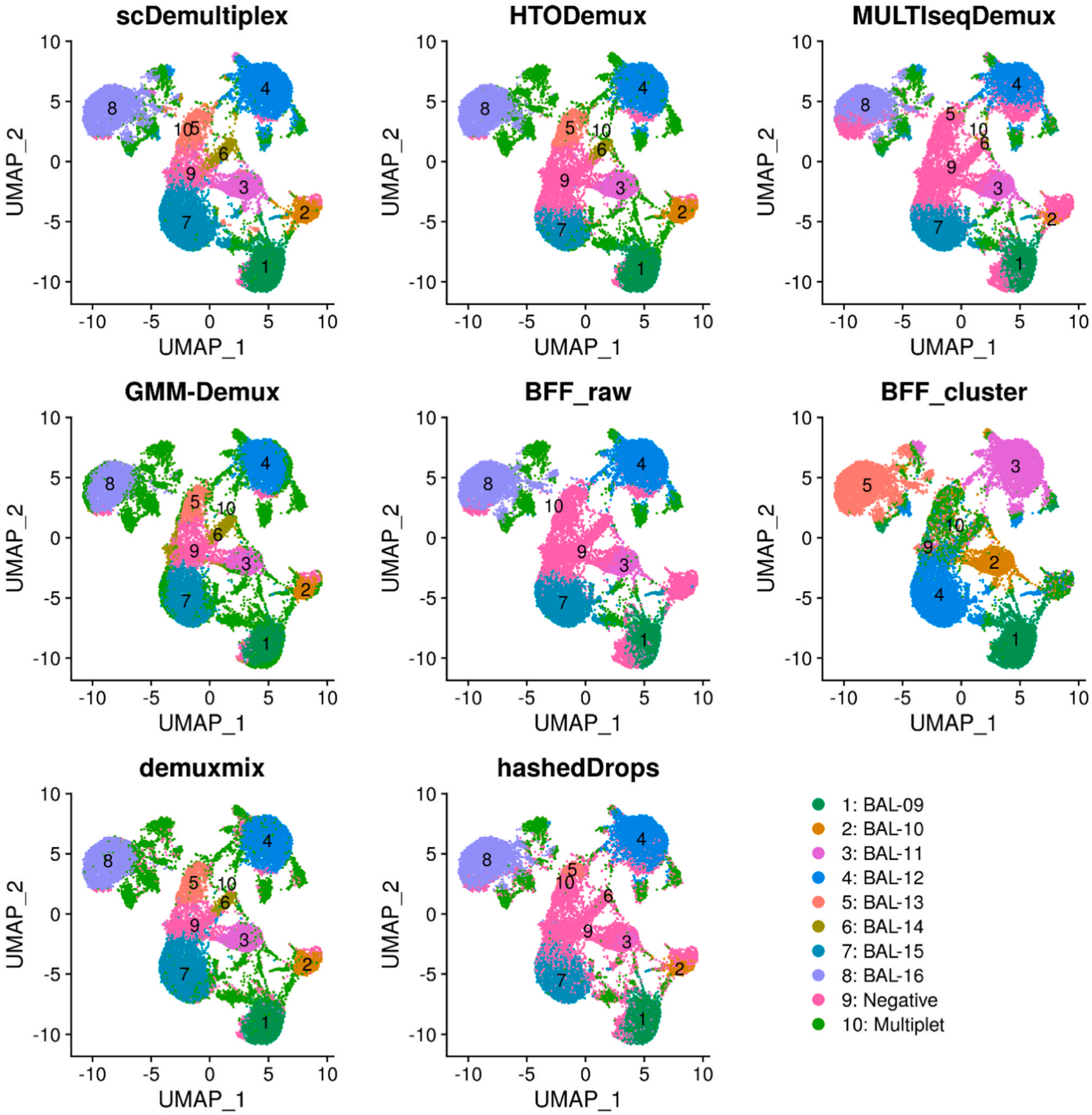


Fig. 3. UMAP plots of the batch2_c1 dataset labeled with demultiplexing results from scDemultiplex, HTODemux, MULTIseqDemux, GMM-Demux, BFF_raw, BFF_cluster, demuxmix and hashedDrops.

demuxmix in the PBMC8 dataset (Supplementary Tables S8-S10). In contrast, hashedDrops and MULTIseqDemux misrecognized many singlets as negatives, and HTODemux misclassified many singlets as multiplets in the Barnyard dataset (Supplementary Fig. S4), resulting in poor performance with low ARI and F-scores (Supplementary Tables S5-S7). In the PBMC8 dataset, MULTIseqDemux also had the lowest ARI and F-score due to its misrecognizing many singlets as negatives (Supplementary Tables S8-S10).

3.2. Application on the batch2 datasets with low-quality

We applied the eight demultiplexing approaches on the batch2 datasets with two captures (batch2_c1 and batch2_c2). We obtained similar results for the two captures within the batch2. The UMAP plots labeled with demultiplexing results from each method showed ambiguous patterns (Fig. 3 and Supplementary Fig. S7). Although most methods identified eight clusters, they were not that distinct from each other compared to the batch1. There was no clear separation between singlet-clusters, negatives, or multiplets, suggesting poor quality of the batch2. Notably, BFF_raw and BFF_cluster only identified five HTO clusters in the batch2_c1 and six in the batch2_c2. BFF_raw failed to recognize BAL-10 and BAL-14, while BFF_cluster missed BAL-15 and BAL-16 in both two captures (Supplementary Tables S11 and S12).

We further investigated negatives and multiplets identified by each method (Fig. 4 and Supplementary Fig. S8). scDemultiplex obtained the most reasonable results, where most negatives either spread in the middle or at the edge of singlet-clusters, and most multiplets located between singlet-clusters or at the edge of singlet-clusters. In comparison, MULTIseqDemux, hashedDrops, and BFF_raw misclassified many singlets as negatives, while HTODemux, GMM_Demux, BFF_cluster, and

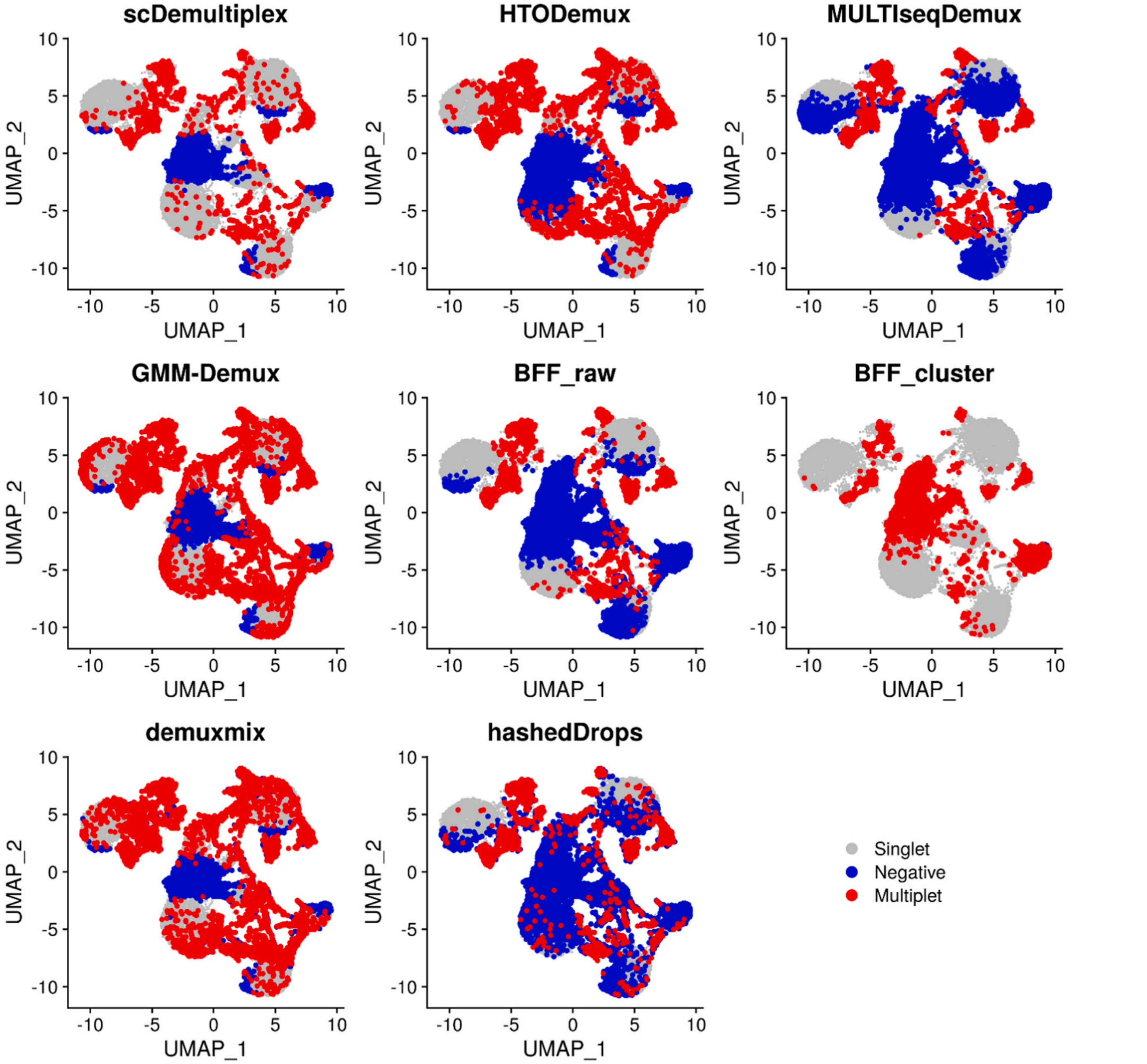


Fig. 4. UMAP plots of the batch2_c1 dataset labeled with singlets, negatives, and multiplets classified by scDemultiplex, HTODemux, MULTIseqDemux, GMM-Demux, BFF_raw, BFF_cluster, demuxmix and hashedDrops.

Table 3
Adjusted rand index and F-score of eight demultiplex methods in the batch2 datasets.

	ARI		F-score	
	batch2_c1	batch2_c2	batch2_c1	batch2_c2
scDemultiplex	**0.671**	**0.633**	**0.842**	**0.836**
HTODemux	0.519	0.437	0.799	0.787
MULTIseqDemux	0.367	0.312	0.576	0.591
GMM_Demux	0.449	0.445	0.759	0.78
BFF_raw	0.477	0.487	0.504	0.568
BFF_cluster	**0.633**	**0.638**	0.121	0.116
demuxmix	0.553	0.52	**0.833**	**0.832**
hashedDrops	0.427	0.408	0.607	0.619

Note: The highest ARI and F-score among the eight approaches is highlighted in red while the second highest ARI and F-score is highlighted in blue.

demuxmix classified numerous cells as multiplets (Supplementary Tables S11 and S12). Those misclassified negatives and multiplets extended to the center of singlet-clusters or even dominated the single-cluster.

Consistent with the UMAP visualization, scDemultiplex achieved the highest/second highest ARI and highest F-score (Table 3) (ARI=0.671 and 0.633; F_score=0.842 and 0.836), demonstrating its high performance in the low-quality data (Supplementary Table S13 and S14). HTODemux, GMM_Demux, and demuxmix showed decent performance. Notably, compared to great performance in the high-quality batch1 datasets, BFF_cluster had the lowest F-score in the batch2, especially for the BAL-15 and BAL-16 samples (Supplementary Table S13 and S14). This result is consistent with a recent benchmark study [8], reporting that methods that assume a bimodal count distribution perform poorly on low-quality data.

3.3. Application on the batch3 datasets with low-quality

We applied the eight demultiplexing approaches on the batch3 datasets with two captures (batch3_c1 and batch3_c2). We obtained similar results for the two captures within the batch3. Similar to the results in the batch2, the UMAP plots labeled with demultiplexing results from each method showed ambiguous patterns (Fig. 5 and Supplementary Fig. S9). Although most methods identified eight clusters, they were not that distinct from each other. Different singlet-clusters were fused into each other, suggesting even worse quality of the batch3 than the batch2. Notably, BFF_raw and BFF_cluster failed to recognize several HTO clusters in both two captures (Supplementary Tables S15 and S16).

We further investigated negatives and multiplets identified by each method (Fig. 6 and Supplementary Fig. S10). scDemultiplex obtained the most reasonable results, where most negatives either spread in the middle or at the edge of singlet-clusters, and most multiplets located between singlet-clusters or at the edge of singlet-clusters. In comparison, MULTIseqDemux, hashedDrops, and BFF_raw misclassified many singlets as negatives, while HTODemux, GMM_Demux, BFF_cluster, and demuxmix classified numerous cells as multiplets (Supplementary Tables S15 and S16). Those misclassified negatives and multiplets extended to the center of singlet-clusters or even dominated singlet-clusters.

Consistently, scDemultiplex achieved the highest ARI and F-score (Table 4) (ARI=0.483 and 0.535; F-score=0.767 and 0.791), demonstrating its high performance in low-quality data. Similar to the results in the batch2, BFF_cluster performed poorly in the low-quality data, while HTODemux, GMM_Demux, and demuxmix had acceptable performance (Table 4 and Supplementary Tables 17 and 18).

In summary, scDemultiplex performs consistently well in both high-quality and low-quality data. In the high-quality data (batch1, Barnyard, and PBMC8), most approaches performed well and scDemultiplex ranked either the best or the second-best. In the low-quality data (batch2 and batch3), performance of demultiplexing approaches differed a lot. Notably, scDemultiplex achieved the highest performance when data quality was poor, demonstrating its power and consistency.

3.4. Integration with other approaches

scDemultiplex uses a two-step strategy, where the first step is to find a starting point and the second step is to fit the beta-binomial model based on the starting point and to refine the model. The two-step strategy makes it very easy to combine other approaches with scDemultiplex by using their results as the starting point.

We found combining approaches with scDemultiplex achieved better performance than using those approaches alone. In all the five datasets, the integrative approaches obtained higher ARI (Table 5) and F-scores (Table 6) except BFF_cluster in the batch1, GMM_Demux in the batch2_c2. We did not combine BFF_cluster and BFF_raw with scDemultiplex in the batch2 and batch3 since they failed to recognize several HTO clusters completely. The improved performance demonstrated that scDemultiplex was able to refine the result from each individual approach, indicating beta-binomial distribution is helpful to model raw HTO counts accurately.

4. Discussion

The experimental design of pooling single cells from multiple samples together with computational demultiplexing is not only cost-effective but also beneficial to the downstream analysis with minimized batch effects. Here, we present scDemultiplex, a novel barcoded-based demultiplexing method. Compared to most methods using Gaussian distribution to model log-transformed count, scDemultiplex directly models raw count by beta-binomial distribution, which fits the nature of the data better. Sample classification based on beta-binomial distribution calculates the probability of cells being from one sample not only by HTO-specific but also HTO-nonspecific counts, which is better than using HTO-specific counts alone. In addition, scDemultiplex uses an iterative strategy to further refine the model and the classification. We evaluated the performance of scDemultiplex by visual inspection, ARI, and F-score based on the “ground truth” obtained from other studies. scDemultiplex not only had great performance in high-quality data, but also performed the best in low-quality data. Since low-quality data is critical to distinguish performance of demultiplexing methods, the best achievement in those challenging datasets demonstrated the power and ability of scDemultiplex.

scDemultiplex uses CLR to normalize the HTO counts. The selection

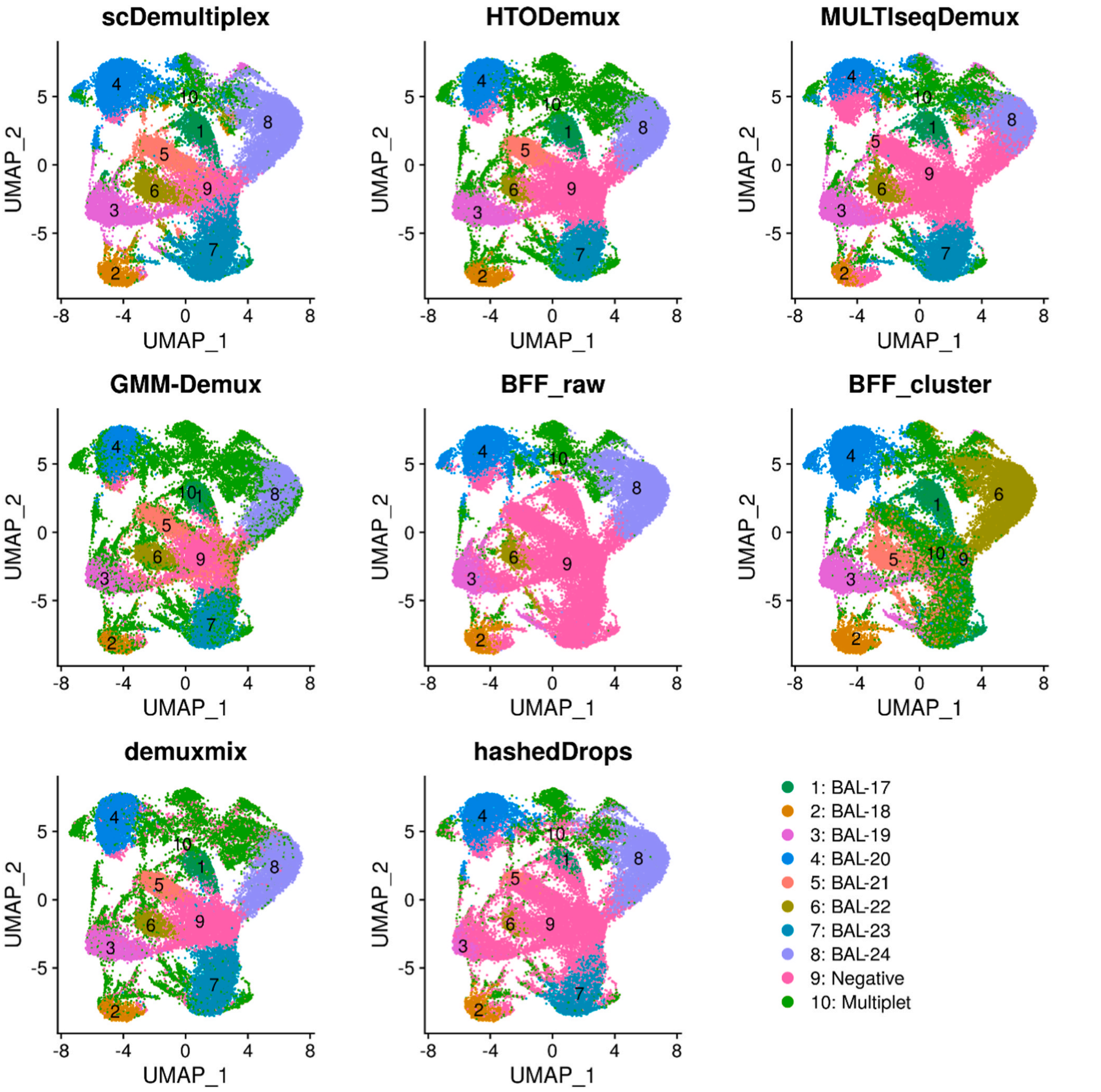


Fig. 5. UMAP plots of the batch3_c1 dataset labeled with demultiplexing results from scDemultiplex, HTODemux, MULTIseqDemux, GMM-Demux, BFF_raw, BFF_cluster, demuxmix and hashedDrops.

of different normalization methods would affect demultiplexing performance. For example, we found the log-transformed library size normalization worsened the performance of scDemultiplex in most cases (Supplementary Table S19), which is possibly because its assumption that each cell has the same number of total HTO counts might not hold true.

scDemultiplex uses an iterative strategy to refine the model. The maximum number of iterations is a tunable parameter to balance between the performance and speed (default=10). The accuracy over the iteration showed a substantial improvement in the first several iterations and slight increase or no change after that in both high-quality and low-quality datasets (Supplementary Fig. S11-S18), which suggested the default 10 times is appropriate. The procedure even stopped earlier since there were no reassignments or multiple misclassifications during one iteration in all the evaluation datasets.

Generally demultiplexing approaches performed comparably on good quality data with a clear bimodal distribution [3]. When the data quality is low and bimodal distribution is ambiguous, demultiplexing approaches have difficulty in determine the thresholds for classifying negatives, singlets, and multiplets and fitting the model. For example, although BFF_cluster achieved the best performance in the high-quality data, it even misrecognized several groups of singlets as multiplets in the low-quality data due to the failure to determine the thresholds when counts did not follow bimodal distribution. With the two-step strategy, scDemultiplex allows users to choose a start point manually, which is very useful for analyzing poor-quality data. Users can start with a stringent cutoff and then fit and refine the model in the second step. Additionally, users can use demultiplexing results of other methods as

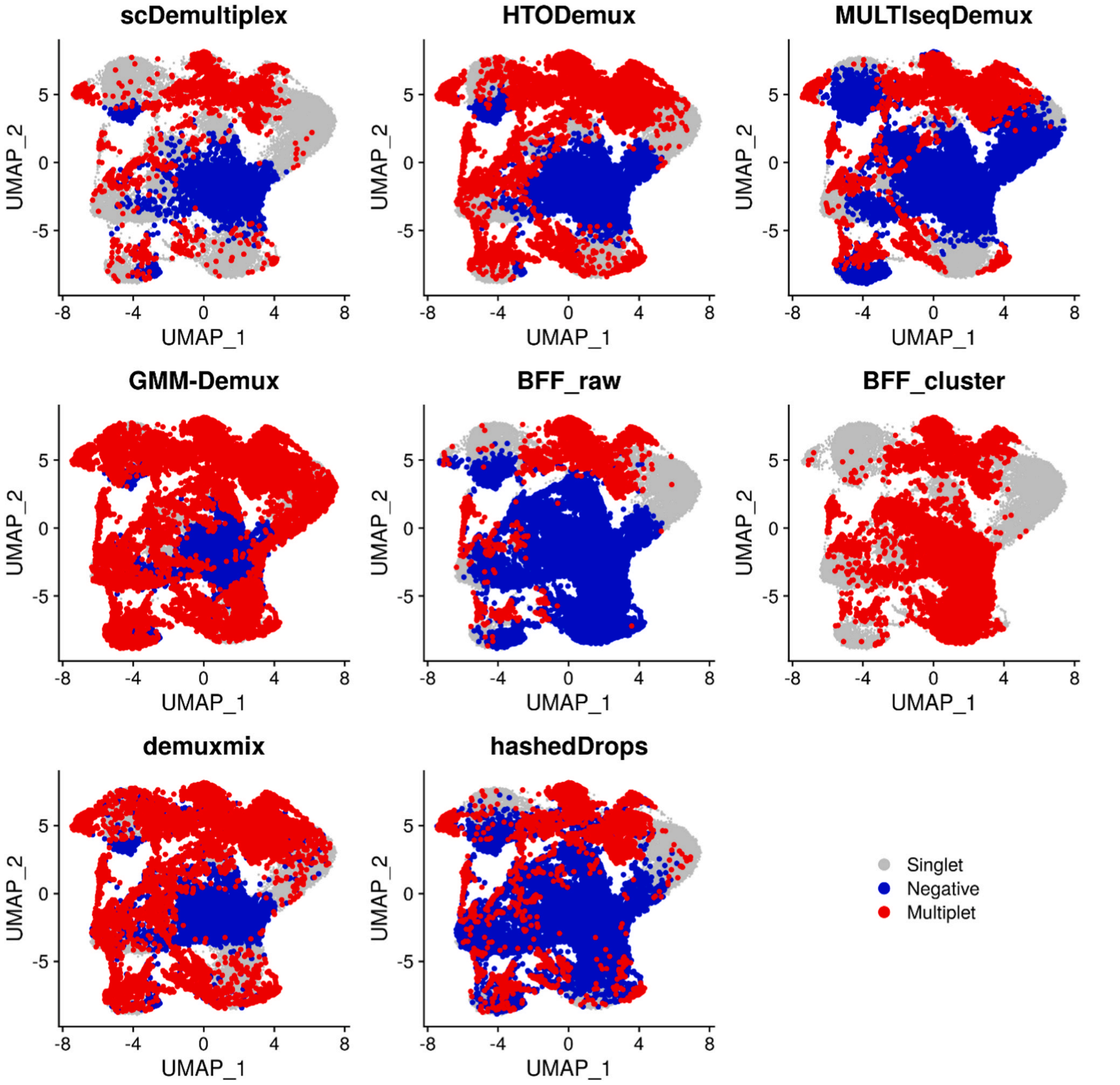


Fig. 6. UMAP plots of the batch3_c1 dataset labeled with singlets, negatives, and multiplets classified by scDemultiplex, HTODemux, MULTIseqDemux, GMM-Demux, BFF_raw, BFF_cluster, demuxmix and hashedDrops.

Table 4
Adjusted rand index and F-score of eight demultiplex methods in the batch3 datasets.

	ARI		F-score	
	batch3_c1	batch3_c2	batch3_c1	batch3_c2
scDemultiplex	**0.483**	**0.535**	**0.767**	**0.791**
HTODemux	0.397	0.412	0.73	0.753
MULTIseqDemux	0.246	0.228	0.604	0.605
GMM_Demux	0.263	0.476	0.644	0.765
BFF_raw	0.249	0.315	0.427	0.541
BFF_cluster	0.338	**0.479**	0.374	0.011
demuxmix	**0.426**	0.46	**0.761**	**0.788**
hashedDrops	0.221	0.265	0.55	0.594

Note: The highest ARI and F-score among the eight approaches is highlighted in red while the second highest ARI and F-score is highlighted in blue.

Table 5
ARI obtained by integrative approaches (each individual approach followed by scDemultiplex) in the three batches.

	batch1_c1	batch1_c2	batch2_c1	batch2_c2	batch3_c1	batch3_c2	Barnyard	PBMC8
HTODemux	0.794(0.722)	0.755(0.564)	0.627(0.519)	0.594(0.437)	0.429(0.397)	0.489(0.412)	0.667(0.46)	0.827(0.79)
MULTIseqDemux	0.681(0.488)	0.641(0.468)	0.533(0.367)	0.508(0.312)	0.389(0.246)	0.449(0.228)	0.698(0.56)	0.797(0.766)
GMM_Demux	0.792(0.731)	0.74(0.69)	0.632(0.449)	0.586(0.445)	0.343(0.263)	0.536(0.476)	0.737(0.631)	0.83(0.814)
BFF_raw	0.727(0.677)	0.712(0.683)	/	/	/	/	0.722(0.639)	0.837(0.819)
BFF_cluster	0.855(0.857)	0.804(0.806)	/	/	/	/	0.802(0.752)	0.825(0.791)
demuxmix	0.806(0.715)	0.756(0.678)	0.678(0.553)	0.643(0.52)	0.493(0.426)	0.544(0.46)	0.722(0.609)	0.84(0.81)
hashedDrops	0.79(0.745)	0.745(0.71)	0.567(0.427)	0.535(0.408)	0.406(0.221)	0.475(0.265)	0.402(0.232)	0.835(0.816)

Note: the number in brackets is the result using the approach alone. Red/blue indicates increased/decreased values after scDemultiplex refinement, and black means no change.

Table 6
F-score obtained by integrative approaches (each individual approach followed by scDemultiplex) in the three batches.

	batch1_c1	batch1_c2	batch2_c1	batch2_c2	batch3_c1	batch3_c2	Barnyard	PBMC8
HTODemux	0.928(0.91)	0.922(0.874)	0.821(0.799)	0.821(0.787)	0.749(0.73)	0.774(0.753)	0.875(0.801)	0.949(0.942)
MULTIseqDemux	0.881(0.805)	0.866(0.792)	0.605(0.576)	0.686(0.591)	0.641(0.604)	0.729(0.605)	0.82(0.722)	0.933(0.923)
GMM_Demux	0.926(0.911)	0.91(0.899)	0.803(0.759)	0.76(0.78)	0.662(0.644)	0.787(0.765)	0.896(0.861)	0.95(0.949)
BFF_raw	0.89(0.867)	0.894(0.885)	/	/	/	/	0.891(0.861)	0.954(0.953)
BFF_cluster	0.943(0.944)	0.925(0.926)	/	/	/	/	0.913(0.899)	0.951(0.946)
demuxmix	0.934(0.912)	0.919(0.902)	0.847(0.833)	0.841(0.832)	0.773(0.761)	0.796(0.788)	0.892(0.856)	0.956(0.953)
hashedDrops	0.923(0.912)	0.91(0.903)	0.682(0.607)	0.714(0.619)	0.693(0.55)	0.766(0.594)	0.819(0.656)	0.953(0.952)

Note: the number in brackets is from the result using the approach alone. Red/blue indicates increased/decreased values after scDemultiplex refinement, and black means no change.

the starting point and then followed by scDemultiplex for refinement, which has been demonstrated to gain improved performance.

Besides barcoded-based demultiplexing, a number of SNP-based tools have been developed to separate samples with different genetic signatures, such as Demuxlet [10], souporcell [7], Vireo [9], and scSplit [27]. In this study, we used SNP-based demultiplexing as the ground truth, which would not be "gold standard" since SNP-based tools have their own limitations, such as high recall and low precision and reduced performance in high proportions of ambient RNA [24]. Those methods would not work when pooled samples have the same genetic background, but they are very useful in the cross-donor or cross-species design. If samples from different genetic background are barcoded and pooled together, an approach to combine scDemultiplex with other SNP-based tools holds great promise to improve the demultiplexing accuracy.

CRediT authorship contribution statement

Li-Ching Huang, Qi Liu and Quanhu Sheng designed the study, implemented the package, analyzed the data, and wrote the manuscript. Hua-Chang Chen implemented web service. Lindsey K Stolze tested the package and drafted the manuscript. Alexander Gelbard and Yu Shyr reviewed and edited the manuscript. All authors read and approved the final manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the National Institutes of Health (NIH) P01AI139449 (Q.L.) and R01HL146401 (A.G.), National Cancer Institute U2CCA233291, U54CA217450, P01CA229123, U54CA274367 (Y.S. and Q.L.) and Cancer Center Support Grant P30CA068485 (Y.S., Q. L. and Q.S.), and Patient-Centered Outcomes Research Institute (PCORI) 1409-22214 (A.G.). We also thank Dr. Zev Gartner for providing the ground truth of the PBMC8 dataset.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.csbj.2023.08.013.

References

[1] Adil A, et al. Single-cell transcriptomics: current methods and challenges in data acquisition and analysis. Front Neurosci 2021;15:591122.
[2] Benjamini Y, Hochberg Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. J R Stat Soc 1995;57:289–300.
[3] Boggy GJ, et al. BFF and cellhashR: analysis tools for accurate demultiplexing of cell hashing data. Bioinformatics 2022;38(10):2791–801.
[4] Butler A, et al. Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nat Biotechnol 2018;36(5):411–20.
[5] Cheng J, et al. Multiplexing methods for simultaneous large-scale transcriptomic profiling of samples at single-cell resolution. Adv Sci (Weinh) 2021;8(17):e2101229.
[6] Gaublomme JT, et al. Nuclei multiplexing with barcoded antibodies for single-nucleus genomics. Nat Commun 2019;10(1):2907.
[7] Heaton H, et al. Souporcell: robust clustering of single-cell RNA-seq data by genotype without reference genotypes. Nat Methods 2020;17(6):615–20.
[8] Howitt, G., et al. Benchmarking single-cell hashtag oligo demultiplexing methods. *bioRxiv [Preprint]* 2022:2022.2012.2020.521313.
[9] Huang Y, McCarthy DJ, Stegle O. Vireo: Bayesian demultiplexing of pooled single-cell RNA-seq data without genotype reference. Genome Biol 2019;20(1):273.

[10] Kang HM, et al. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat Biotechnol 2018;36(1):89–94.

[11] Klein, H.-U. demuxmix: Demultiplexing oligonucleotide-barcoded single-cell RNA sequencing data with regression mixture models. *bioRxiv [Preprint]* 2023: 2023.2001.2027.525961.

[12] Lun ATL, et al. EmptyDrops: distinguishing cells from empty droplets in droplet-based single-cell RNA sequencing data. Genome Biol 2019;20(1):63.

[13] Ma P, et al. Bacterial droplet-based single-cell RNA-seq reveals antibiotic-associated heterogeneous cellular states. Cell 2023;186(4):877–91. e814.

[14] McFarland JM, et al. Multiplexed single-cell transcriptional response profiling to define cancer vulnerabilities and therapeutic mechanism of action. Nat Commun 2020;11(1):4296.

[15] McGinnis CS, Murrow LM, Gartner ZJ. DoubletFinder: doublet detection in single-cell RNA sequencing data using artificial nearest neighbors. Cell Syst 2019;8(4): 329–37. e324.

[16] McGinnis CS, et al. MULTI-seq: sample multiplexing for single-cell RNA sequencing using lipid-tagged indices. Nat Methods 2019;16(7):619–26.

[17] Scrucca L, et al. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. R J 2016;8(1):289–317.

[18] Shin D, et al. Multiplexed single-cell RNA-seq via transient barcoding for simultaneous expression profiling of various drug perturbations. Sci Adv 2019;5 (5):eaav2249.

[19] Stoeckius M, et al. Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome Biol 2018;19(1):224.

[20] Tabula Sapiens C, et al. The Tabula Sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. Science 2022;376(6594). eabl4896.

[21] Tietscher S, et al. A comprehensive single-cell map of T cell exhaustion-associated immune environments in human breast cancer. Nat Commun 2023;14(1):98.

[22] Trang NV, et al. Determination of cut-off cycle threshold values in routine RT-PCR assays to assist differential diagnosis of norovirus in children hospitalized for acute gastroenteritis. Epidemiol Infect 2015;143(15):3292–9.

[23] Tvedebrink T. Overdispersion in allelic counts and theta-correction in forensic genetics. Theor Popul Biol 2010;78(3):200–10.

[24] Weber LM, et al. Genetic demultiplexing of pooled single-cell RNA-sequencing samples in cancer facilitates effective experimental design. GigaScience 2021;10:9.

[25] Winkler EA, et al. A single-cell atlas of the normal and malformed human brain vasculature. Science 2022;375(6584):eabi7377.

[26] Xin H, et al. GMM-Demux: sample demultiplexing, multiplet detection, experiment planning, and novel cell-type verification in single cell sequencing. Genome Biol 2020;21(1):188.

[27] Xu J, et al. Genotype-free demultiplexing of pooled single-cell RNA-seq. Genome Biol 2019;20(1):290.

[28] Ziffra RS, et al. Single-cell epigenomics reveals mechanisms of human cortical development. Nature 2021;598(7879):205–13.