Decontamination of ambient RNA in single-cell RNA-seq with DecontX

Shiyi Yang[1], Sean E. Corbett[1], Yusuke Koga[1], Zhe Wang[1], W. Evan Johnson[1], Masanao Yajima[2], and Joshua D. Campbell[1]

[1]Division of Computational Biomedicine, Department of Medicine, Boston University School of Medicine, Boston, MA, USA.

[2]Department of Mathematics & Statistics, Boston University, Boston, MA, USA.

ABSTRACT

Droplet-based microfluidic devices have become widely used to perform single-cell RNA sequencing (scRNA-seq) and discover novel cellular heterogeneity in complex biological systems. However, ambient RNA present in the cell suspension can be incorporated into these droplets and aberrantly counted along with a cell's native mRNA. This results in cross-contamination of transcripts between different cell populations and can potentially decrease the precision of downstream analyses. We developed a novel hierarchical Bayesian method called DecontX to estimate and remove contamination in individual cells from scRNA-seq data. DecontX accurately predicted the proportion of contaminated counts in a mixture of mouse and human cells. Decontamination of PBMC datasets removed aberrant expression of cell type specific marker genes from other cell types and improved overall separation of cell clusters. In general, DecontX can be incorporated into scRNA-seq workflows to assess quality of dissociation protocols and improve downstream analyses.

INTRODUCTION

Single-cell RNA sequencing (scRNA-seq) has emerged as a powerful technique to study complex biological systems at single-cell resolution (Wang et al,2015). Droplet-based scRNA-seq platforms have been widely adopted because of their ability to profile a large number of cells at relatively low cost (Ziegenhain et al. 2017). These devices work by using droplets to partition cells into nanoliter reaction chambers along with beads harboring oligonucleotide primers with unique barcodes. Within each droplet, cells are lysed and the mRNAs will be tagged with the oligonucleortide primers to create barcoded cDNA after reverse transcription (Macosko et al,2015, Zilionis et al,2016, Zheng et al,2017).

Despite their many advantages, droplet-based single-cell technologies can suffer from the presence of cross-contamination from ambient RNA in each droplet. Ambient RNA is the pool of mRNA molecules that have been released in the cell suspension, likely from cells that are stressed or have undergone apoptosis. Cross-contamination occurs when the ambient RNA gets incorporated into the droplets and is barcoded and amplified along with a cell's native mRNA (**Figure 1A**). Contamination from ambient RNA is evident when highly-expressed cell-type specific genes are observed at low levels in other cell populations. Different levels of contamination can be found in different droplets depending on the amount of ambient and native mRNA present. Two major goals of many scRNA-seq studies are to cluster cells into subpopulations and identify unique combinations of marker genes that define each cell population (Trapnell,2015). Ambient RNA can hinder these tasks by causing different cell populations to blend together and the expression of true marker genes to be detected across multiple cell populations. We developed a computational method called DecontX to estimate and remove ambient RNA for scRNA-seq data. We applied DecontX to three datasets to demonstrate its ability to accurately quantify and remove contamination within each cell from other populations and to improve downstream clustering.

RESULTS

To address the issue of contamination, we developed a novel Bayesian method called DecontX that identifies and removes contamination in individual cells. We assume the observed expression of a cell is a mixture of counts from two multinomial distributions: 1) a distribution of native transcript counts from the cell's actual population and 2) a distribution of contaminating transcript counts from all other cell populations

captured in the assay (**Online Methods**, **Figure 1B**). The native expression distribution for each cell population is characterized by a multinomial parameter ϕ_k, where ϕ_{kg} is the probability of gene g being expressed in population k. Likewise, the contamination distribution for each cell population k is characterized by a multinomial parameter η_k, where η_{kg} is the probability of gene g contaminating population k. Each individual cell j has a parameter θ_j, which follows a beta distribution and represents the proportion of counts derived from the native expression distribution. Each transcript count has a hidden state, y_{jt}, which follows a bernoulli distribution parameterized by θ_j and denotes the transcript's membership to the native expression distribution ($y_{jt} = 1$) or contamination distribution ($y_{jt} = 0$). This framework is similar to a discrete Bayesian hierarchical model called latent Dirichlet allocation (LDA) (Blei et al,2003) where documents are mixtures of K topics and each topic is a mixture of words from a predefined vocabulary. However, rather than having K different distributions to model the mixtures of counts from different cell populations within each cell, we explicitly define the contamination distribution to be a weighted combination of all other cell population distributions. We use variational inference (Jordan et al,1999) to approximate posterior distributions to allow fast and scalable inference in large datasets (Blei et al,2017). Ultimately, DecontX will deconvolute a gene-by-cell count matrix and a vector of cell population labels into a matrix of contamination counts and a matrix of native counts which can be used in downstream analyses (**Figure 1C**).

To demonstrate the accuracy of DecontX, we utilized a public datase containing a mixture of fresh frozen human embryonic cells (HEK293T) and mouse embryonic fibroblasts (NIH3T3) cells from 10X Genomics. Using CellRanger (Zheng et al,2017), reads were uniquely aligned to a combined human-mouse reference genome (hg19 and mm10) to ensure that only reads specific to each organism will be counted while those that align to the genome of both organisms will be excluded. Cells were classified as human, mouse, or multiplets based on the levels of the organism-specific transcript counts (**Supplementary Figure 1**). The cells predicted to be either mouse or human still exhibited low levels of expression of counts aligning specifically to the other organism (**Figure 2A**). The proportion of mouse-specific genes in human cells was highly correlated to the distribution of expression in an average mouse cell (R=0.96; **Figure 2B**). Conversely, the proportion of human-specific genes in mouse cells was highly correlated to the distribution of expression in an average human cell (R=0.99; **Figure 2C**). These results also show that highly expressed genes in one cell subpopulation are more likely to contribute to contamination in other cell populations. Furthermore, while the median contamination was relatively low (1.09% in human cells and 2.75% in

mouse cells), the percentage of contamination varied substantially from cell to cell (0.43%-45.09% in human; 1.25%-44.43% in mouse; **Figure 2D**) and demonstrates the need to have individual estimates of contamination for each cell.

We applied DecontX to 12,079 non-multiplet cells in the human-mouse mixture dataset. Most of the exogenous transcripts were identified and removed by DecontX (**Figure 3A**). The estimated proportion of contamination in individual human cells was highly correlated to the proportion of mouse-specific transcripts in those cells (R= 0.99; RMSE=0.002; **Figure 3B**). A high correlation was also observed in mouse cells (R=0.99; RMSE=0.005; **Figure 3C**), demonstrating the ability of DecontX to accurately detect contamination from other cell populations. The estimated gene-level contamination distributions for human or mouse cell populations were also highly correlated to the expression of an average mouse or human cell, respectively (**Supplementary Figure 2**).

We next sought to understand the effect and extent of contamination in publicly-available scRNA-seq datasets of peripheral mononuclear cells (PBMCs). To establish baseline expression of cell-type specific marker genes in a settting with limited possibility for contamination, we examined 4 different immune populations (sorted PBMCs) isolated by flow cytometry and profiled with the 10X Genomics Chromium in separate channels (Zheng et al,2017). As each population was isolated and profiled in a different channel, gene markers for a specific immune population were detected at relatively low levels in other populations. For example, the mRNA expression of T-cell specific genes such as CD3E and CD3D were only found 0.07% in the B-cells sorted on CD20. Conversely, B-cell specific markers such as CD79A, CD79B and MS4A1 were only detected in 9.09% of T-cells sorted on CD8A or CD4 (**Figure 4A**, **4B**). Similarly, low percentages of marker genes of other cell types could be found for B-cells and monocytes, monocytes and T-cells, T-cells and NK-cells, NK-cells and B-cells, and NK-cells and monoctyes (**Figure 4B**, **Supplementary Figure 3**).

In the second dataset, over four thousand PBMCs (4K PBMC) were isolated and profiled in a single channel of the 10X Genomics Chromium. Since cluster labels were not available from flow cytometry, we utilized Celda (Corbett et al,2019) to identify 19 cell populations where each population was a unique combination of 150 gene modules (**Supplementary Figures 5**, **6**). In contrast to the previous dataset, higher levels of cell-type specific marker genes could be detected in other cell types including CD3E and CD3D in 21.12% B-cell population; CD79A, CD79B and MS4A1 in 25.32% T-cell population (**Figure 4**); Likewise,

higher level of a marker gene (GNLY) for NK-cells was found in monocytes and B-cells, marker genes (LYZ, S100A8 and S100A9) for monocytes in NK-cells, B-cells and T-cells (**Figure 4B**, **Supplementary Figure 3**). After we applied DecontX to remove contamination, the expression of T-cell specific marker genes was eliminated in B-cells and expression of B-cell specific marker genes was eliminated in T-cells (**Figure 4A**). The percentage of cells within each subpopulation that had expression of marker genes from other cell types markedly decreased (**Figure 4B**, **Supplementary Figure 3**). While the overall levels of the NK-cell marker GNLY was substantially reduced in T-cells and the T-cell markers CD3D and CD3E were reduced in NK-cells, some expression of these markers still remained in both populations. Decontaminated counts resulted in improved separation in two dimensions when applying tSNE (Maaten et al,2008) (**Figure 5A**, **5B**). Additionally, the mean silhouette width, a measure of cluster stability and separation, improved from 0.04 on original normalized expression to 0.06 on normalized expression after decontamination (**Figure 5C**). Specifically, all clusters have improved mean silhouette width except for the cluster 17, which shows a decrease of average silhouette width from -0.002 to -0.042 (**Figure 5C**). Interestingly, cells from cluster 17 were predicted to be doublets by a doublet prediction method Scrublet (Wolock et al,2019) (**Figure 6A**). Cells predicted to be doublets by Scrublet are associated with higher contamination estimated by DecontX (p-value $<$ 2e-16, **Figure 6**). In fact, all cells estimated to have high levels of contamination ($>$ 70%) were predicted to be doublets by Scrublet suggesting that DecontX contamination estimates can be used as orthogonal information for doublet detection (**Figure 6C**).

DISCUSSION

We developed a method called DecontX to estimate the percentage of cross contamination within each cell due to ambient RNA in droplet-based single-cell RNA sequencing experiments. In human-mouse mixture data, DecontX was able to accurately estimate the percentage of exogenous transcripts. After estimating and removing contaminated transcripts in 4K PBMC data, the profiles of key marker genes for each subpopulation better resembled those from sorted PBMCs. Furthermore, decontamination resulted in improved downstream clustering and visualization. The cells estimated to be highly contaminated by DecontX in 4K PBMC were also estimated to be doublets by Scrublet. Therefore, high contamination levels may also be useful as a quality control criterion for excluding cells. Additionally, estimating the levels of background RNA contributing to the contamination will be important for quality assessment of

cell dissociation protocols.

By utilizing raw counts for estimation of the multinomial distributions, DecontX eliminates the potential variability that could be introduced by different normalization methods. One limitation is that cell cluster labels are needed a *priori*. While we automatically use Celda to identify cell clusters if none are supplied, any fast cell clustering approach can be substituted. As the contamination distribution for each cell population is derived from all other populations present in the dataset, it may sometimes better to use broader cell population labels. For example, including all T-cells in one cluster rather than treating individual T-cell subpopulation as a separate subcluster may help alleviate T-cell specific counts in the calculation of the contamination distribution. Overall, computational decontamination of single-cell counts with DecontX will aide in down-stream clustering and visualization and can be systematically included in analysis workflows.

ONLINE METHODS

Statistical model

We assume there are K known distinct cell populations among the M cell samples, where cell j has N_j observed transcripts. We denote native expression distribution for cell population k as a G-length vector ϕ_k. For the notational convenience, we will use $\boldsymbol{\phi}_{-k} = \{\phi_{k'} : k' \neq k, k' \in \{1, 2, ..., K\}\}$ to represent gene expressions from all other cell populations other than k. Each cell j has a parameter θ_j to represent the proportion of counts that are derived from native expression distribution. θ_j is assumed to from a global beta distribution which leverages the variation of contamination level cross all the cells in the dataset, with hyperparameters a_1 and a_2 *a-priori*. The t^{th} transcript x_{jt} in cell j has a hidden state, y_{jt}, which follows a bernoulli distribution parameterized by θ_j and denotes the transcript's membership to be native expression distribution ($y_{jt} = 1$) or contamination distribution ($y_{jt} = 0$). Assuming that transcripts are conditionally independent given hidden state y_{jt} and cell's population z_j, x_{jt} follows a multinomial distribution either parameterized by ϕ_{z_j} denoting native expression or $\boldsymbol{\phi}_{-z_j}$ denoting contamination. The joint posterior distribution can be expressed as:

$$P(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{\theta}|\boldsymbol{\phi}, a_1, a_2) = \prod_{j=1}^{M} p(\theta_j|a_1, a_2) \prod_{t=1}^{N_j} \Bigg(\Big[p(y_{jt} = 1|\theta_j) \cdot p(x_{jt} = g|\phi_{z_j})\Big]^{I(y_{jt}=1)} \Big[p(y_{jt} = 0|\theta_j) \cdot p(x_{jt} = g|\boldsymbol{\phi}_{-z_j})\Big]^{I(y_{jt}=0)} \Bigg) \tag{1}$$

To simplify conputation work and notation, we assume the contamination distribution η_k is a simple linear combination of $\boldsymbol{\phi}_{-k}$.

$$\eta_k = \sum_{k':k\neq k} w_{k'}\phi_{k'} \tag{2}$$

where the weight $w_{k'}$ is the proportion of native transcripts from cluster k' and is calculated using expected values, of which the full definition is given later in inference.

Variational inference

We use variational inference to approximate the probability densities for our model. The following variational distributions are introduced to break down the coupling of $\boldsymbol{\theta}$ and $\boldsymbol{Y}$ for variational inference:

$$q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi}) = \prod_{j=1}^{M} q(\theta_j|\gamma_j) \prod_{t=1}^{N_j} q(y_{jt}|\pi_{jt}) \tag{3}$$

where the Beta parameter $\gamma_j = \{\gamma_{j1}, \gamma_{j2}\}$, and Bernoulli parameter $\pi_{jt} = \{\pi_{jt1}, \pi_{jt2}\}$ are the free variational pameters. π_{jt} satisfies $\pi_{jt1} + \pi_{jt2} = 1$, and $q(y_{jt}) = \pi_{jt1}^{I(y_{jt}=1)}\pi_{jt2}^{I(y_{jt}=0)}$. The variational Beta distribution for θ_j is $q(\theta_j) = \frac{\Gamma(\gamma_{j1}+\gamma_{j2})}{\Gamma(\gamma_{j1})\Gamma(\gamma_{j2})}\theta_{j1}^{\gamma_{j1}-1}\theta_{j2}^{\gamma_{j2}-1}$

The need to compute the expectation of the θ_j arises in deriving the variational inference. Using the general fact for exponential family, that the derivative of the log normalization factor with respect to the natural parameter is equal to the expectation of the sufficient statistic ($\log\theta_{ji}$, $i \in \{1, 2\}$ in our Beta distribution) , we have:

$$E[\log\theta_{ji}|\gamma_{j1}, \gamma_{j2}] = \Psi(\gamma_{ji}) - \Psi(\gamma_{j1} + \gamma_{j2}), i \in \{1, 2\} \tag{4}$$

where Ψ is the digamma function, the first derivative of the log Gamma function.

For simplicity in notation, let us use $Q = \{\boldsymbol{\theta}, \boldsymbol{Y}\}$ and $a = \{a_1, a_2\}$. We begin variational inference by bounding the log-likelihood using Jensens inequality.

$$\begin{aligned}
\log p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{a}, \boldsymbol{\phi}) &= \log \int_Q p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{Y}|a, \boldsymbol{\phi}) dQ \\
&= \log \int_Q \frac{p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{Y}|a, \boldsymbol{\phi})}{q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})} q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})) dQ \\
&\geq \int_Q \log \frac{p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{Y}|a, \boldsymbol{\phi})}{q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})} q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})) dQ \\
&= E_Q[\log p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{Y}|a, \boldsymbol{\phi})] - E_Q[\log q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})]
\end{aligned} \tag{5}$$

Jensens inequality provides us with a lower bound on the log likelihood for an arbitrary variational distribution $q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})$.

We then expand the lower bound:

$$\begin{aligned}
L(\boldsymbol{\gamma}, \boldsymbol{\pi}; a, \boldsymbol{\phi}) &= E_Q \left[\log p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{Y}|a, \boldsymbol{\phi})\right] - E_Q \left[\log q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})\right] \\
&= E_Q \left[\log p(\boldsymbol{\theta}|a) + \log p(\boldsymbol{Y}|\boldsymbol{\theta}) + \log p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\phi})\right] \\
&\quad - E_Q \left[\log q(\boldsymbol{\theta}|\boldsymbol{\gamma}) + \log q(\boldsymbol{Y}|\boldsymbol{\pi})\right]
\end{aligned} \tag{6}$$

Expanding each term in the lower bound by taking expectation with respect to $q(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{\gamma}, \boldsymbol{\pi})$:

$$\begin{aligned}
E_Q \left[\log p(\boldsymbol{\theta}|a)\right] &= E_Q \left[\log \prod_{j=1}^{M} p(\theta_j|a)\right] = E_Q \left[\sum_{j=1}^{M} \log p(\theta_j|a)\right] = \sum_{j=1}^{M} E_Q \left[\log p(\theta_j|a)\right] \\
&= \sum_{j=1}^{M} E_Q \left[\log \Gamma(a_1 + a_2) - \log \Gamma(a_1) - \log \Gamma(a_2) + (a_1 - 1) \log \theta_{j1} + (a_2 - 1) \log \theta_{j2}\right] \\
&= \sum_{j=1}^{M} \left[\log \Gamma(a_1 + a_2) - \left(\sum_{i=1}^{2} \log \Gamma(a_i)\right) + \sum_{i=1}^{2} (a_i - 1) \left(\Psi\left(\gamma_{ji}\right) - \Psi\left(\gamma_{j1} + \gamma_{j2}\right)\right)\right]
\end{aligned} \tag{7}$$

$$
\begin{aligned}
E_Q\left[\log p(\boldsymbol{Y}|\boldsymbol{\theta})\right] &= E_Q\left[\log \prod_{j=1}^{M}\prod_{t=1}^{N_j} p(y_{jt}|\theta_j)\right] = E_Q\left[\sum_{j=1}^{M}\sum_{t=1}^{N_j} \log p(y_{jt}|\theta_j)\right] = \sum_{j=1}^{M}\sum_{t=1}^{N_j} E_Q\left[\log p(y_{jt}|\theta_j)\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j} E_Q\left[y_{jt}\log\theta_{j1} + (1-y_{jt})\log\theta_{j2}\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j} \left[\pi_{jt1}\left(\Psi(\gamma_{j1}) - \Psi(\gamma_{j1}+\gamma_{j2})\right) + \pi_{jt2}\left(\Psi(\gamma_{j2}) - \Psi(\gamma_{j1}+\gamma_{j2})\right)\right]
\end{aligned}
\tag{8}
$$

$$
\begin{aligned}
E_Q\left[\log p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{Y},\boldsymbol{\phi})\right] &= E_Q\left[\log \prod_{j=1}^{M}\prod_{t=1}^{N_j} p(x_{jt},z_j|y_{jt},\phi_{z_j},\eta_{z_j})\right] = \sum_{j=1}^{M}\sum_{t=1}^{N_j} E_Q\left[\log p(x_{jt},z_j|y_{jt},\phi_{z_j},\eta_{z_j})\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j} E_Q\left[\sum_{g=1}^{G} x_{jt}^g y_{jt}\log\phi_{z_j,g} + x_{jt}^g(1-y_{jt})\log\eta_{z_j,g}\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j}\sum_{g=1}^{G} E_Q\left[x_{jt}^g y_{jt}\log\phi_{z_j,g} + x_{jt}^g(1-y_{jt})\log\eta_{z_j,g}\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j}\sum_{g=1}^{G} \left[x_{jt}^g \pi_{jt1}\log\phi_{z_j,g} + x_{jt}^g\pi_{jt2}\log\eta_{z_j,g}\right]
\end{aligned}
\tag{9}
$$

$$
\begin{aligned}
E_Q\left[\log q(\boldsymbol{\theta}|\boldsymbol{\gamma})\right] &= E_Q\left[\log \prod_{j=1}^{M} q(\theta_j|\gamma_j)\right] = E_Q\left[\sum_{j=1}^{M} \log q(\theta_j|\gamma_j)\right] = \sum_{j=1}^{M} E_Q\left[\log q(\theta_j|\gamma_j)\right] \\
&= \sum_{j=1}^{M} E_Q\left[\log\Gamma(\gamma_{j1}+\gamma_{j2}) - \log\Gamma(\gamma_{j1}) - \log\Gamma(\gamma_{j2}) + (\gamma_{j1}-1)\log\theta_{j1} + (\gamma_{j2}-1)\log\theta_{j2}\right] \\
&= \sum_{j=1}^{M}\left[\log\Gamma(\gamma_{j1}+\gamma_{j2}) - \left(\sum_{i=1}^{2}\log\Gamma(\gamma_{ji})\right) + \sum_{i=1}^{2}(\gamma_{ji}-1)\left(\Psi(\gamma_{ji}) - \Psi(\gamma_{j1}+\gamma_{j2})\right)\right]
\end{aligned}
\tag{10}
$$

$$
\begin{aligned}
E_Q\left[\log q(\boldsymbol{Y}|\boldsymbol{\pi})\right] &= E_Q\left[\log \prod_{j=1}^{M}\prod_{t=1}^{N_j} q(y_{jt}|\pi_{jt})\right] = E_Q\left[\sum_{j=1}^{M}\sum_{t=1}^{N_j} \log q(y_{jt}|\pi_{jt})\right] = \sum_{j=1}^{M}\sum_{t=1}^{N_j} E_Q\left[\log q(y_{jt}|\pi_{jt})\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j} E_Q\left[y_{jt}\log \pi_{jt1} + (1-y_{jt})\log \pi_{jt2}\right] \\
&= \sum_{j=1}^{M}\sum_{t=1}^{N_j} \left[\pi_{jt1}\log \pi_{jt1} + \pi_{jt2}\log \pi_{jt2}\right]
\end{aligned}
\tag{11}
$$

We then maximize the lower bound with respect to the variational parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\pi}$.

First to maximize the lower bound with respect to $\boldsymbol{\pi}$. Since π_{jt} are independent, for $t \in \{1, 2, ..., N_j\}$, we isolate the terms that contains π_{jt}. Lagrangian multiplier is added due to the constraint $\pi_{jt1} + \pi_{jt2} = 1$. We substituted $x_{jt}^g \pi_{jt1} \log \phi_{z_j,g}$ and $x_{jt}^g \pi_{jt2} \log \eta_{z_j,g}$ from equation (9) with $\pi_{jt1} \log \phi_{z_j,g}$ and $\pi_{jt2} \log \eta_{z_j,g}$ respectively, since $x_{jt}^g = I(x_{jt} = g)$ and is observed

$$
\begin{aligned}
L_{[\pi_{jt}]} = &\left[\pi_{jt1}\left(\Psi(\gamma_{j1}) - \Psi(\gamma_{j1}+\gamma_{j2})\right) + \pi_{jt2}\left(\Psi(\gamma_{j2}) - \Psi(\gamma_{j1}+\gamma_{j2})\right)\right] \\
&+ \left[\pi_{jt1}\log \phi_{z_j,g} + \pi_{jt2}\log \eta_{z_j,g}\right] \\
&- \left[\pi_{jt1}\log \pi_{jt1} + \pi_{jt2}\log \pi_{jt2}\right] \\
&- \lambda(\pi_{jt1} + \pi_{jt2} - 1)
\end{aligned}
\tag{12}
$$

Taking derivative with respect to π_{jt1}, we obtain:

$$
\frac{\partial L}{\partial \pi_{jt1}} = \left(\Psi(\gamma_{j1}) - \Psi(\gamma_{j1}+\gamma_{j2})\right) + \log \phi_{z_j,g} - \log \pi_{jt1} - \lambda - 1 \tag{13}
$$

Setting this derivative to zero yields the maximizing value of the variational parameter π_{jt1}:

$$
\pi_{jt1} \propto \phi_{z_j,g} \exp\left(\Psi(\gamma_{j1}) - \Psi(\gamma_{j1}+\gamma_{j2})\right) \tag{14}
$$

Similarly we could have π_{jt2}:

$$
\pi_{jt2} \propto \eta_{z_j,g} \exp\left(\Psi(\gamma_{j2}) - \Psi(\gamma_{j1}+\gamma_{j2})\right) \tag{15}
$$

Next, we maximize the lower bound with respect to $\boldsymbol{\gamma}$. Since γ_j are independent for $j \in 1, 2, ..., M$, each

γ_j can be estimated separately. We isolate the terms that contain γ_j.

$$L_{[\gamma_j]} = \sum_{i=1}^{2}(a_i - 1)\left(\Psi\left(\gamma_{ji}\right) - \Psi\left(\gamma_{j1} + \gamma_{j2}\right)\right) + \sum_{t=1}^{N_j}\left[\pi_{jt1}\left(\Psi(\gamma_{j1}) - \Psi(\gamma_{j1} + \gamma_{j2})\right) + \pi_{jt2}\left(\Psi(\gamma_{j2}) - \Psi(\gamma_{j1} + \gamma_{j2})\right)\right] - \left[\log\Gamma\left(\gamma_{j1} + \gamma_{j2}\right) - \left(\sum_{i=1}^{2}\log\Gamma(\gamma_{ji})\right) + \sum_{i=1}^{2}\left(\gamma_{ji} - 1\right)\left(\Psi(\gamma_{ji}) - \Psi(\gamma_{j1} + \gamma_{j2})\right)\right] \tag{16}$$

Taking derivative with respect to γ_{ji}, we obtain:

$$\frac{\partial L}{\partial \gamma_{ji}} = \Psi'(\gamma_{ji})\left(a_i + \sum_{t=1}^{N_j}\pi_{jt1} - \gamma_{j1}\right) - \Psi'(\gamma_{j1} + \gamma_{j2})\left(a_1 + \sum_{t=1}^{N_j}\pi_{jt1} - \gamma_{j1} + a_2 + \sum_{t=1}^{N_j}\pi_{jt2} - \gamma_{j2}\right) \tag{17}$$

where Ψ' is the derivative of the digamma function. Setting this derivative to zero yields a maximum at:

$$\gamma_{ji} = a_i + \sum_{t=1}^{N_j}\pi_{jt1}, i \in \{1, 2\} \tag{18}$$

Finally, we move forward to estimating $\boldsymbol{\phi}$, a, and to update $\boldsymbol{\eta}$.

To maximize with respect to ϕ_k, we isolate terms and add Lagrangian multiplier due to the constraint $\sum_{g=1}^{G}\phi_{kg} = 1$:

$$L_{[\phi_k]} = \sum_{j:z_j=k}\sum_{t=1}^{N_j}\sum_{g=1}^{G} x_{jt}^{g}\pi_{jt1}\log\phi_{kg} - \lambda(\sum_{g=1}^{G}\phi_{kg} - 1) \tag{19}$$

Taking the derivative with respect to ϕ_{kg} and set it to zero, we get:

$$\phi_{kg} \propto \sum_{j:z_j=k}\sum_{t=1}^{N_j} x_{jt}^{g}\pi_{jt1} \tag{20}$$

The weight $w_{k'}$ is the proportion of native transcripts from cluster k' and is calculated using expected

values:

$$w_{k'} = \frac{\sum_{k':k'\neq k}\left(\sum_{j:z_j=k'}\sum_{t=1}^{N_j}\pi_{jt1}\right)}{\sum_{j:z_j\neq k}\sum_{t=1}^{N_j}\pi_{jt1}} \tag{21}$$

Hence we have our updated $\eta_{k'g}$ as:

$$\eta_{kg} = \frac{\sum_{k':k'\neq k}\left(\sum_{j:z_j=k'}\sum_{t=1}^{N_j}\pi_{jt1}\right)\phi_{k'g}}{\sum_{j:z_j\neq k}\sum_{t=1}^{N_j}\pi_{jt1}} \tag{22}$$

To maximize with respect to a, we isolate terms and get:

$$L_{[a]} = \sum_{j=1}^{M}\left[\log\Gamma(a_1+a_2) - \left(\sum_{i=1}^{2}\log\Gamma(a_1)\right) + \sum_{i=1}^{2}(a_i-1)\left(\Psi\left(\gamma_{ji}\right) - \Psi\left(\gamma_{j1}+\gamma_{j2}\right)\right)\right] \tag{23}$$

A Newton iteration can be used to find the maximal point a (Minka,2000), which requires both the first and second derivatives of $L_{[a]}$. The first derivative, gradient g, and the second derivative, Hessian matrix H are:

$$g_i = \frac{\partial L_{[a]}}{\partial a_i} = \sum_{j=1}^{M}\left(\Psi(a_1+a_2) - \Psi(a_i) + \Psi(\gamma_{ji}) - \Psi\left(\gamma_{j1}+\gamma_{j2}\right)\right) \tag{24}$$

$$\begin{aligned} H_{ii} &= \frac{\partial^2 L_{[a]}}{\partial a_i^2} = M\left(\Psi'(a_1+a_2) - \Psi(a_i)\right), i\in\{1,2\} \\ H_{ij} &= \frac{\partial^2 L_{[a]}}{\partial a_i \partial a_j} = M\Psi'(a_1+a_2), j\neq i \end{aligned} \tag{25}$$

One Newton step is then

$$a^{new} = a^{old} - H^{-1}g \tag{26}$$

Analysis of sorted human-mouse mixture single-cell dataset

A mixture of fresh frozen human (HEK293T) and mouse (NIH3T3) cells were sequenced together in 10X Genomics Chromium. This data is available at `https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/hgmm_12k`. 6,164 human cells, 5,915 mouse cells and 741 multiplets were

detected by CellRanger. Excluding multiplets, 12,079 cells with CellRanger predicted cell type were used to estimate contamination using DecontX.

Analysis of sorted PBMC single-cell datasets

9 publicly-available PBMC datasets totalling of 84,432 cells were obtained from 10X Genomics. Each dataset consisted of a population of cells that were isolated with flow cytometry based on expression of a predefined protein marker. Cell populations included progenitor cells(CD34+), monocytes (CD14+), B cells (CD19+), Natural Killer cells (CD56+), helper T-cells (CD4+), regulatory T-cells (CD4+/CD25+), native T-cells (CD4+/CD45RA+/CD25-), naive cytotoxic T-cells (CD8+/CD45RA+) and cytotoxic T-cells (CD8+). A total of 7,363 genes which contained at least 3 counts across 3 cells were included in the analysis. DecontX used cell label by flow cytometry to estiamte contamination. Celda was used to identify and 76 gene modules and 21 cell clusters, including 8 clusters predominantly expressing T-cell markers, 2 clusters predominantly expressing Natural Killer cell markers, 2 clusters predominantly expressing B-cell markers, 2 clusters predominantly expressing monocyte markers, and 7 clusters predominantly expressing CD34 progenitor cell markers. These computationally inferred cell type labels were used in downstream analyses that examined the percentage of cells that express various marker genes. Using computationally derived cell clustered mitigated instances where a cell was improperly sorted and labeled by flow cytometry as belonging to one population when in fact it transcriptionally similar to another population.

Analysis of the 4K PBMC single-cell dataset

4,340 PBMCs from a healthy donor were sequenced in a single channel of the 10X Genomics Chromium. Data is available at `https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/pbmc4k`. A total of 4,529 genes which contained at least 3 counts across 3 cells were included in the analysis. 19 cell clusters and 150 gene modules were identified with Celda. Cell clusters 2 and 3 were classified as B-cells (MS4A1+); cell clusters 5, 6, 7, 8, 9 and 11 were classified as T-cells (CD3D+/CD3E+); cell clusters 13 and 14 were identified as LYZ+ monocyte group; cell cluster 15 was identified as FCGR3A+

monocytes group; cell cluster 10 was identified as NKG7+ and GNLY+ NK-cell group; cell clusters 18 and 19 were identified as FCER1A+ dendritic cell group; cell cluster 4 was identified as IRF7+ and IRF8+ plasmacytoid dendritic cell group; cell cluster 16 was identified as PPBP+ Megakaryocytes; cell cluster 1 was identified as IGHG1+ and IGHG2+ plasma cell group; cell cluster 12 was identified as CD34+ cell group; cell cluster 17 is likely to be multiplets for it has shown IL7R, CD3D and CD14 markers. DecontX used Celda estimated cluster label to estimate contamination.

ACKNOWLEDGMENTS

This work was funded by LUNGevity Career Development Award (J.D.C.) and Informatics Technology for Cancer Research (ITCR) 1U01 CA220413-01 (W.E.J.). We thank Carter Merenstein, Ke Xu and Xinyi Shi for helpful suggestions during the analysis.

AUTHOR CONTRIBUTIONS

JDC coneived the project; JDC, SY, MY developped the model; SY, YK performed the analysis; SY, JDC, MY wrote the manuscript; SC, ZW assisted in software development; SY, JDC, MY, YK, SC, ZW, ZW, EJ reviewed the manuscript.

REFERENCES

Wang, Yong, and Nicholas E. Navin. "Advances and applications of single-cell sequencing technologies." Molecular cell 58.4 (2015): 598-609.

Ziegenhain, Christoph, Beate Vieth, Swati Parekh, Bjrn Reinius, Amy Guillaumet-Adkins, Martha Smets, Heinrich Leonhardt, Holger Heyn, Ines Hellmann, and Wolfgang Enard. Comparative Analysis of Single-Cell RNA Sequencing Methods. Molecular Cell 65, no. 4 (February 16, 2017): 631-643.e4. https://doi.org/10.1016/j.molcel.2017.01.023.

Macosko, Evan Z., et al. "Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets." Cell 161.5 (2015): 1202-1214.

Zilionis, Rapolas, et al. "Single-cell barcoding and sequencing using droplet microfluidics." Nature protocols 12.1 (2017): 44.

Zheng, Grace XY, et al. "Massively parallel digital transcriptional profiling of single cells." Nature communications 8 (2017): 14049.

Trapnell, Cole. "Defining cell types and states with single-cell genomics." Genome research 25.10 (2015): 1491-1498.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." Journal of machine Learning research 3.Jan (2003): 993-1022.

Jordan, Michael I., et al. "An introduction to variational methods for graphical models." Machine learning 37.2 (1999): 183-233.

Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe. "Variational inference: A review for statisticians." Journal of the American Statistical Association 112.518 (2017): 859-877.

Corbett S, Campbell J, Koga Y, Wang Z (2019). celda: CEllular Latent Dirichlet Allocation. R package version 1.0.4. `http://bioconductor.org/packages/celda/`

Maaten, Laurens van der, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9.Nov (2008): 2579-2605.

Wolock, Samuel L., Romain Lopez, and Allon M. Klein. "Scrublet: computational identification of cell doublets in single-cell transcriptomic data." Cell systems 8.4 (2019): 281-291.

Minka, Thomas. Estimating a Dirichlet distribution. Technical report, M.I.T., 2000.

FIGURE LEGENDS

Figure 1: **Overview of decontamination with DecontX**. **(A)** In droplet-based microfluidic devices, ambient RNA can be incorporated into droplets along with oligonucleotide-barcoded beads and cells. Both native mRNA from the cell and contaminating ambient RNA will be barcoded and counted within a droplet. **(B) Left**: DecontX assumes that each cell is a mixture of two multinomial distributions: 1) a distribution of native transcripts from the cells true population and 2) a distribution of contaminating

transcripts from all other cell populations captured in the assay. **Right**: Simulation of an example cell with 20% contamination. The 800 native transcripts are from the multinomial distribution for cell population 1 while the 200 contaminating transcripts are derived from a probability distribution that is a weighted combination of the two other populations. **(C)** DecontX will take an expression count matrix and cell cluster labels and estimate matrices of native expression and contamination from ambient RNA.

Figure 2: **Contamination in a human-mouse cell mixture dataset**. **(A)** The total number of UMIs aligned specifically to the mouse or human genome is plotted for each droplet. **(B)** The proportion of counts for mouse genes in human cells is highly correlated to the average expression of these genes across all mouse cells indicating that the amount of contamination for each gene is proportional to how highly that gene is expressed in the contaminating cell population. **(C)** Similarly, the proportion of counts for human genes in the mouse cells is highly correlated to the average expression of those genes across all human cells. **(D)** While each droplet is predicted to contain a single cell, the median percentage of contamination for human and mouse cells is 1.09% and 2.75%, respectively. The range of contamination is 0.43% - 45.09%indicating the need for contamination estimation for each individual cell.

Figure 3: **Decontamination of the human-mouse cell mixture dataset**. **(A)** The number of human UMIs is again plotted against the number of mouse UMIs for each droplet before and after decontamination with DecontX. After DecontX, the median percentage of contaminating counts for each droplet is 0.26% (0.12% - 0.73%). **(B**, **C)** The DecontX-estimated contamination proportion is highly correlated to the known proportion of exogenous transcripts for each droplet predicted to have a human or mouse cell.

Figure 4: **Expression of cell-type specific marker genes before and after decontamination in PBMCs**. **(A)** For each gene, the average expression in the B-cell clusters is plotted against the average expression in T-cell clusters for three different datasets: data from sorted PMBCs profiled in different channels **(left)**; data from the PBMC 4K before decontamination **(middle)**; and the PBMC 4K data after decontamination with DecontX **(right)**. **(B)** Percentage of cells expressing specific marker genes for different cell-types for three different datasets. Markers included CD79A, CD79B and MS4A1 for B-cells, CD3E and CD3D for T-cells, GNLY for NK-cells; and LYZ, S100A8 and S100A9 for monocytes.

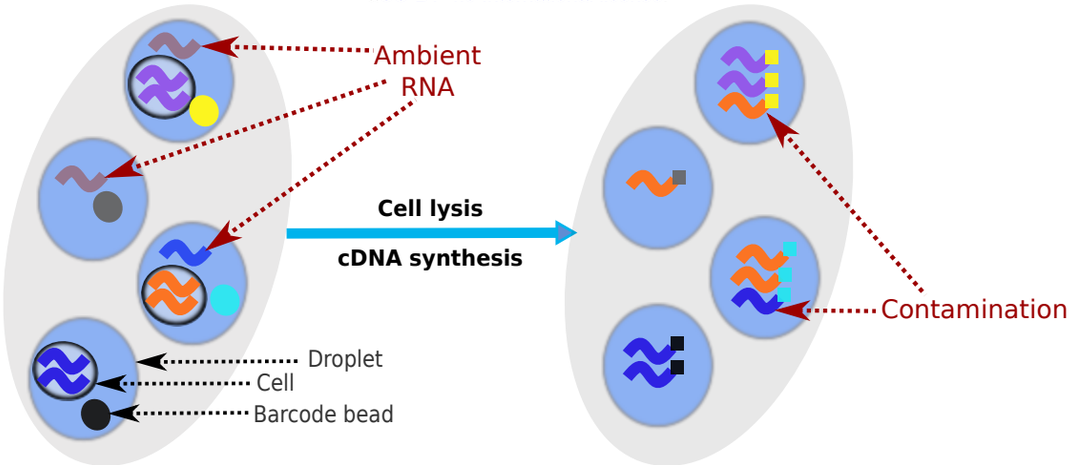
Figure 5: **Cluster similarity before and after decontamination**. **(A)** tSNE of 19 cell clusters from the PBMC 4K dataset before decontamination. **(B)** Decontamination with DecontX improved separation on tSNE between different cell clusters. **(C)** The mean silhouette width was derived for each cluster before

and after decontamination with DecontX. Each point represents the difference in the mean silhouette width for each cluster. All clusters except 17 showed an increase in silhouette width after decontamination. Cluster 17 was predicted to contain mostly doublets by Scrublet. Cluster 1 had only one cell and was not included in the analysis.

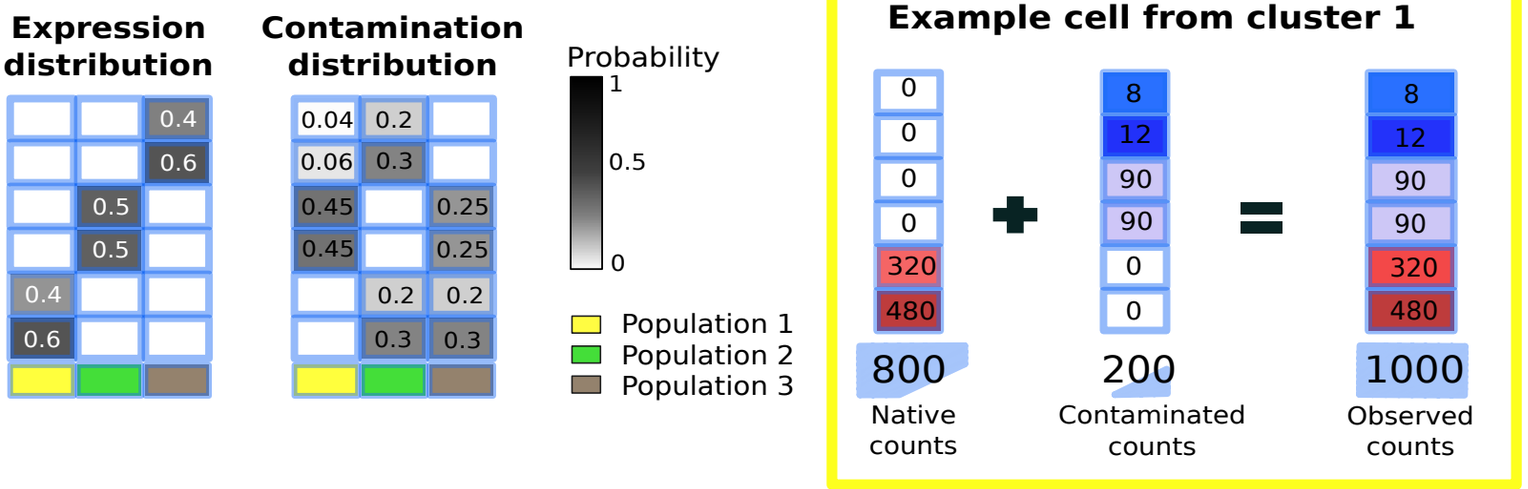
Figure 6: **Comparison of contamination levels with predicted doublets**. **(A)** tSNE of DecontX decontaminated PBMC 4K data. Red cells are predicted doublets by Scrublet. **(B)** Each cell is colored by contamination level estimated by DecontX. **(C)** Predicted doublets had significantly higher levels of estimated contamination compared to singlets. The median contamination for doublets was 45.95% (7.18% - 99.39%) while the median for singlets was 6.81% (0.02% - 73.18%).

Figure 1

A



B



C

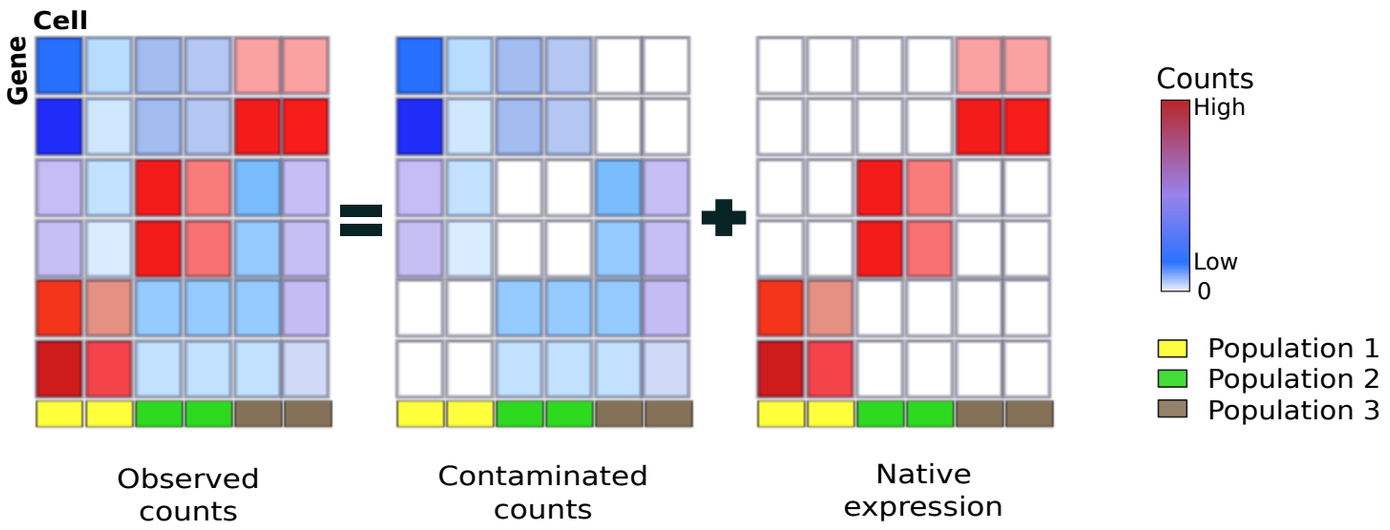
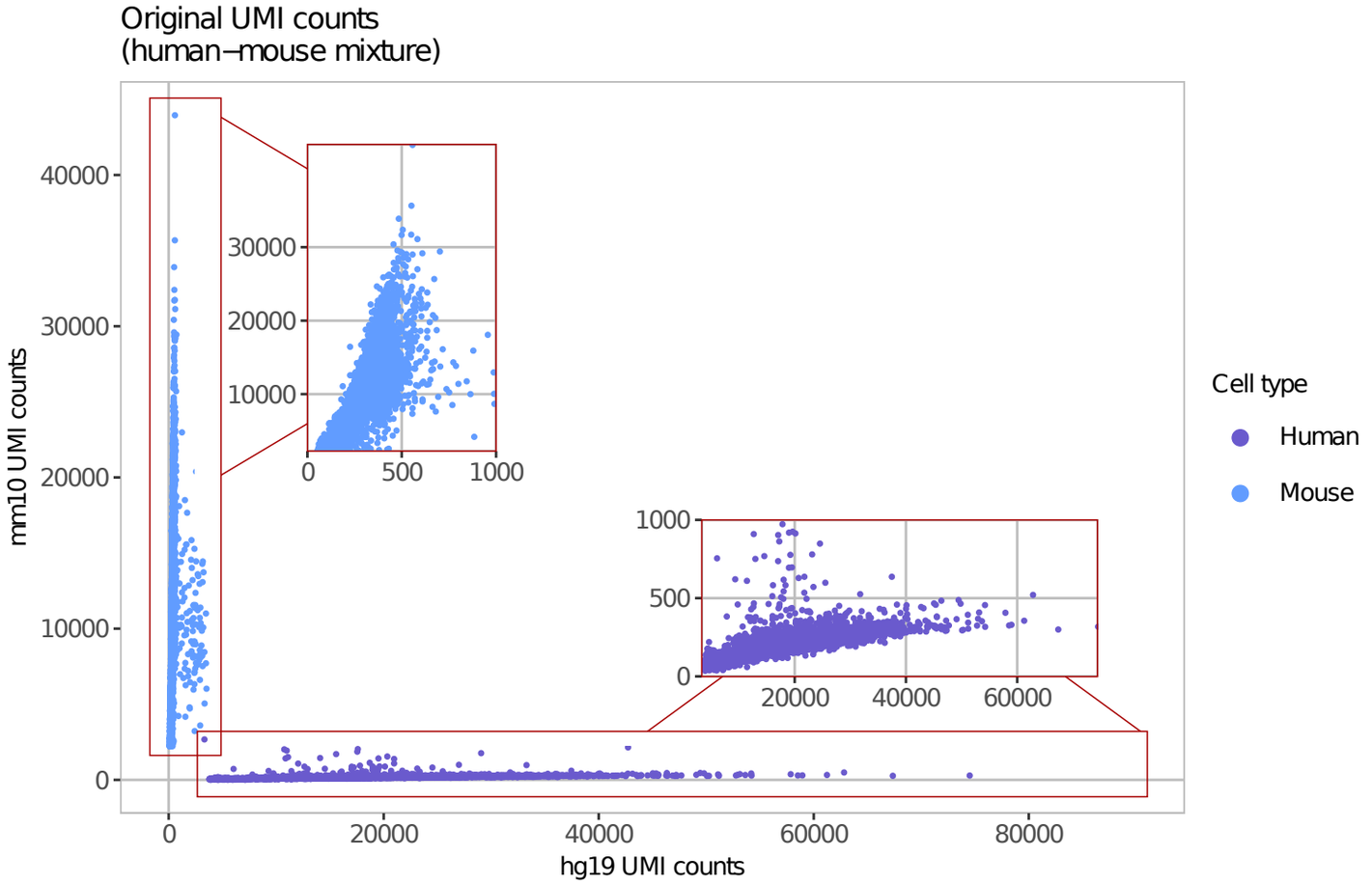
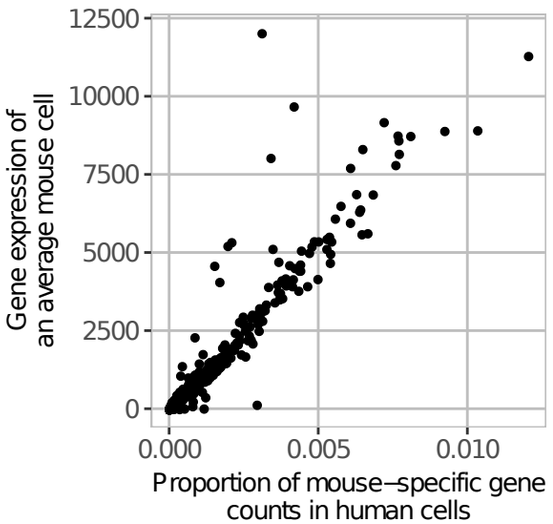


Figure 2

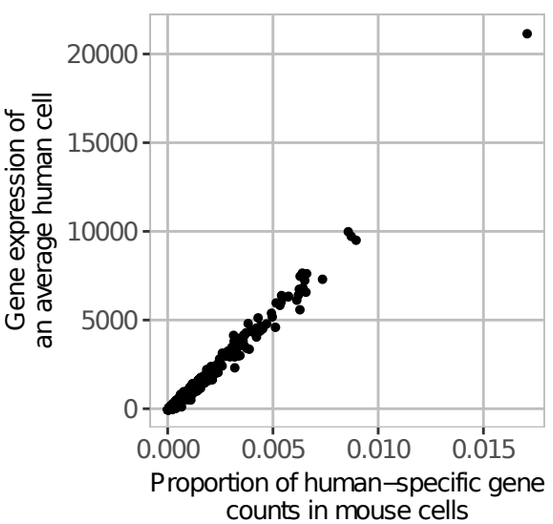
A



B



C



D

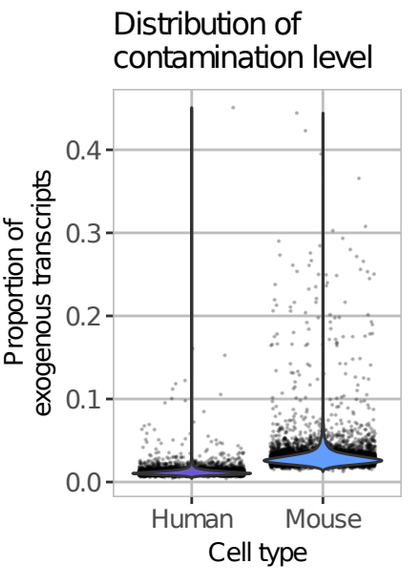
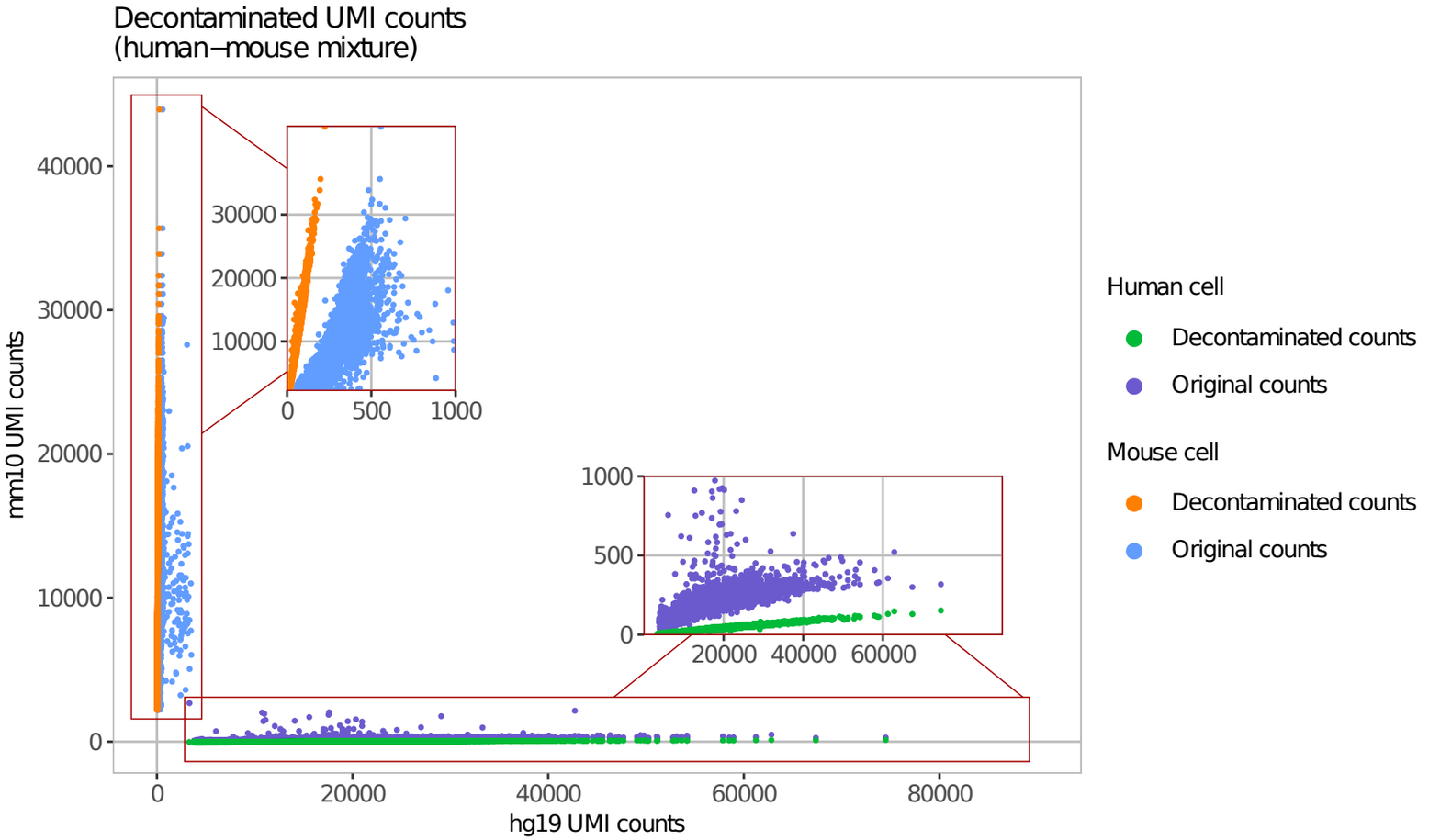
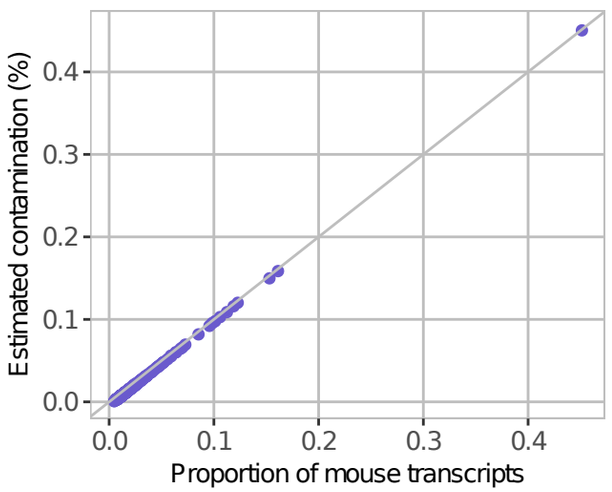


Figure 3

A



B



C

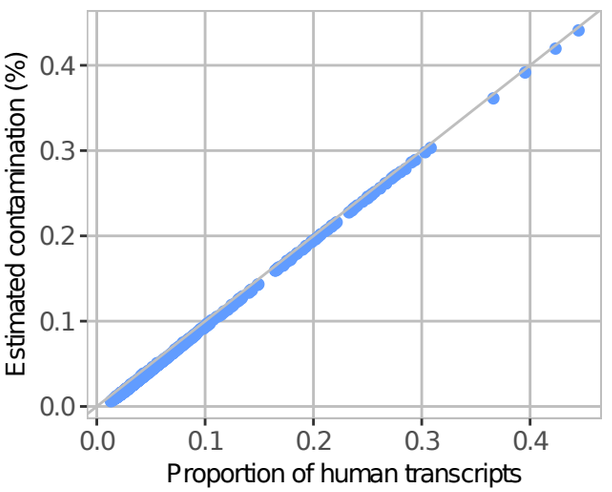
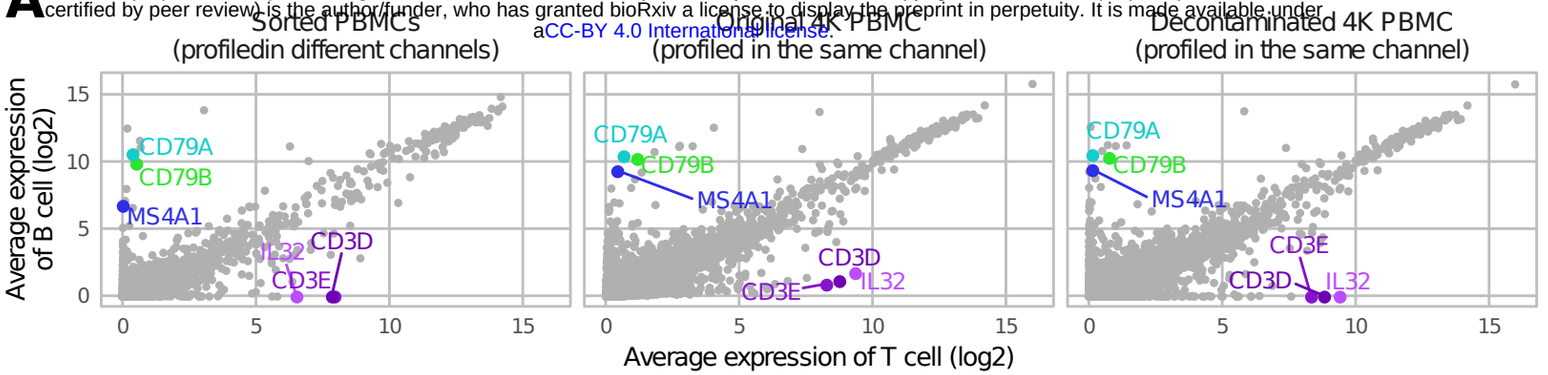


Figure 4

A



B

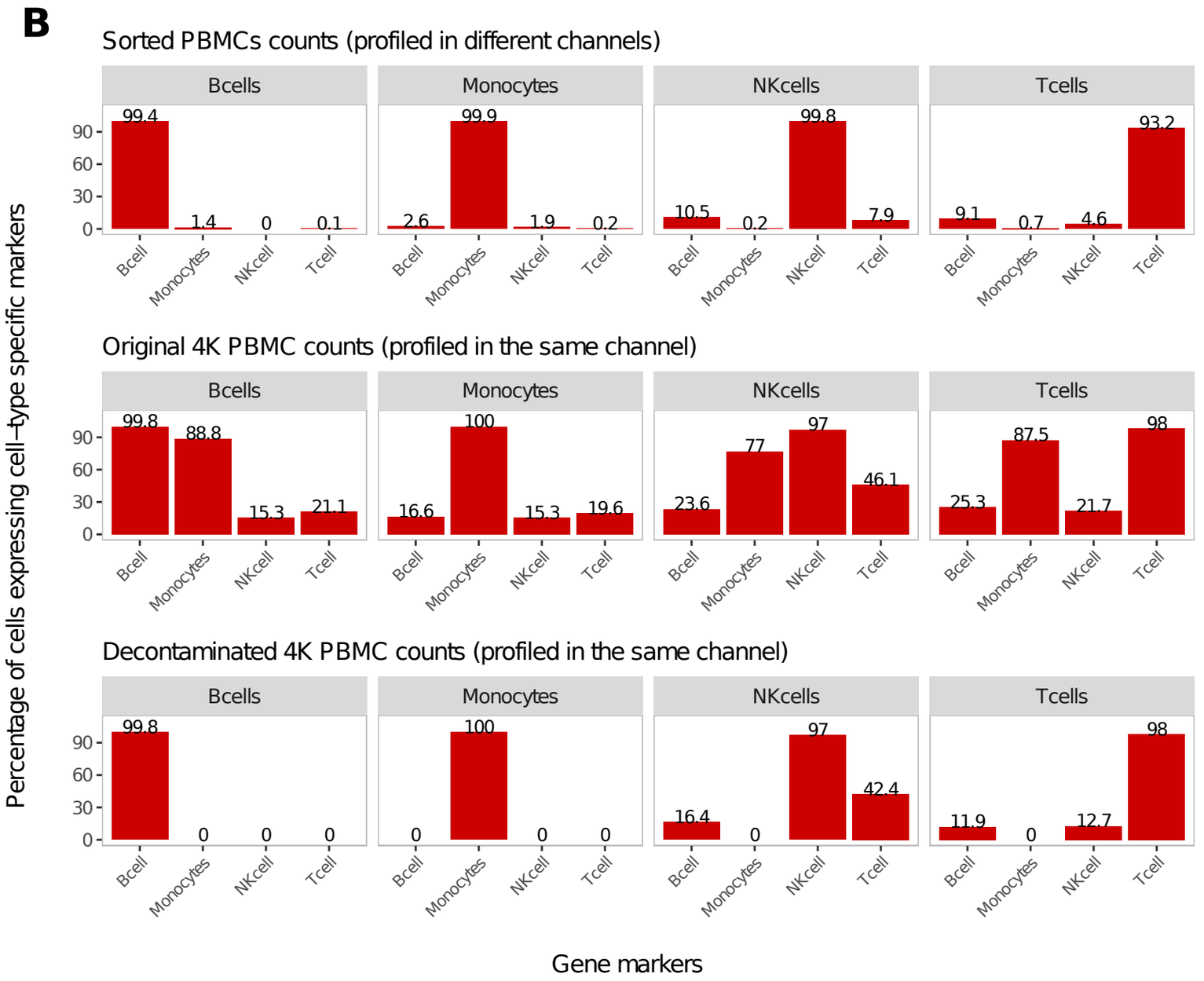
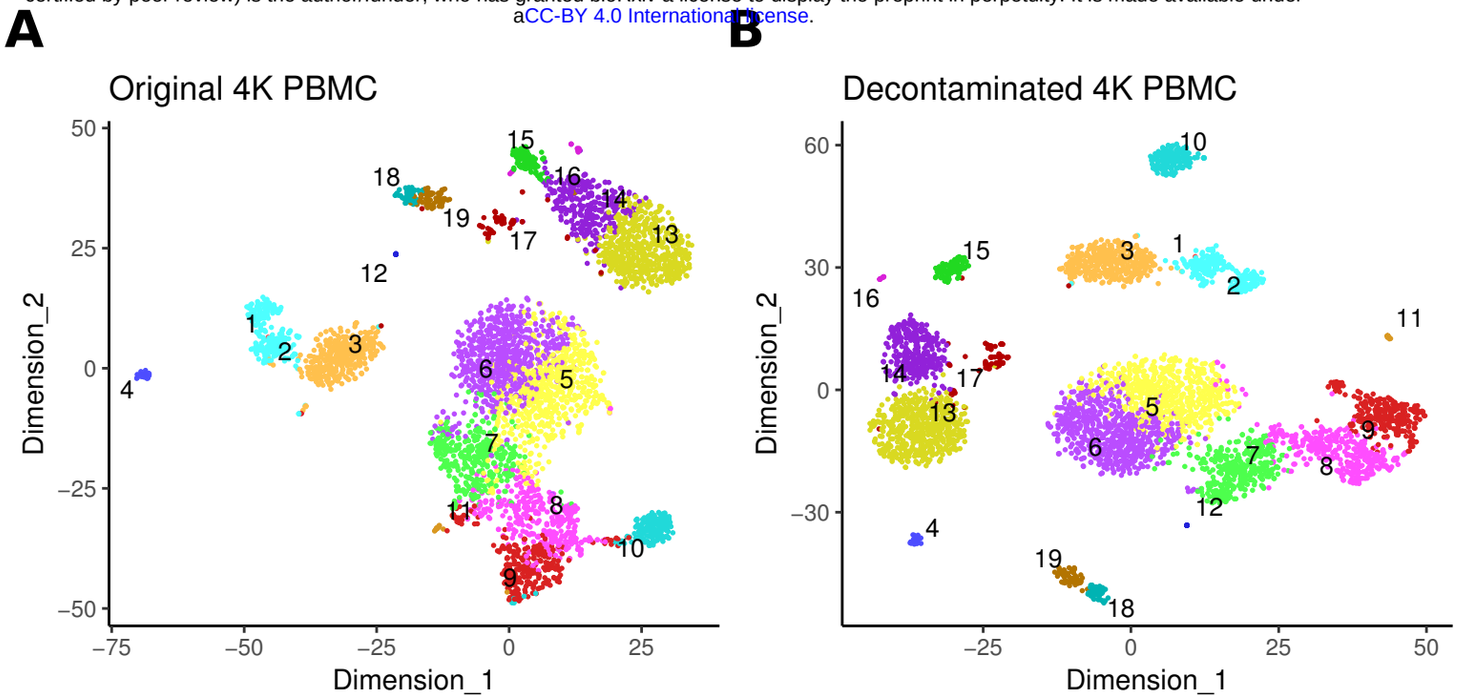


Figure 5



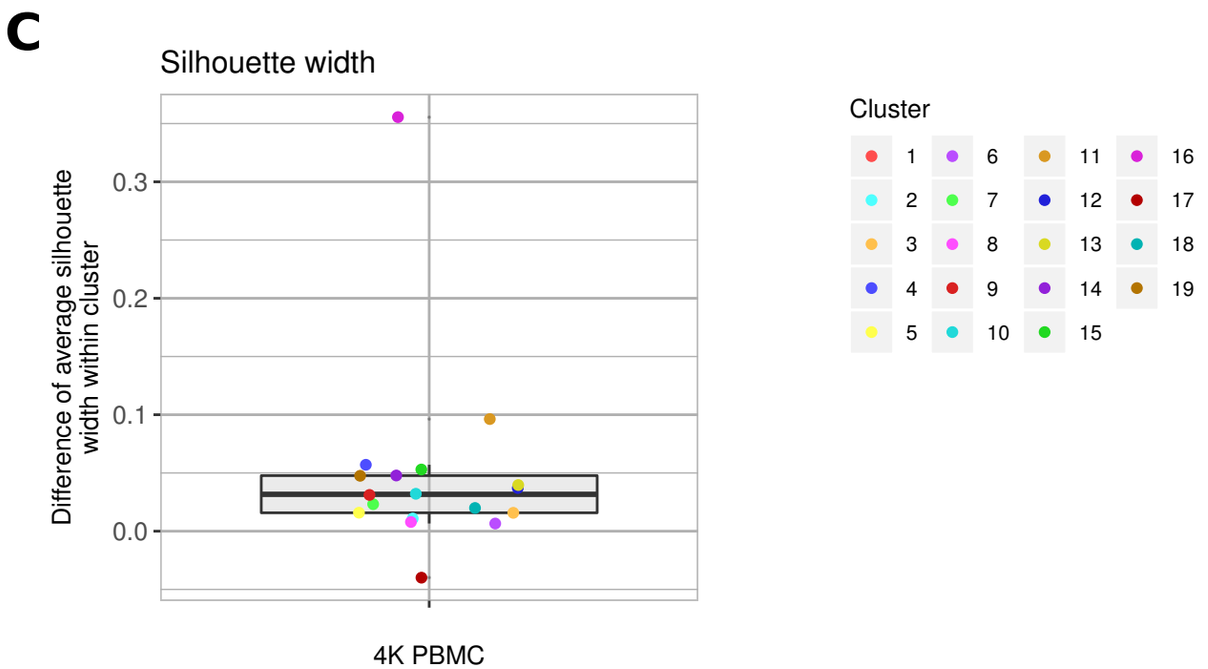


Figure 6