Doublet detection in Pegasus

Bo Li

1 Overview

Doublets consist of transcriptomes from two different cells. Doublets may be mistakenly considered as new biology (e.g. rare cell types) due to their distinction from singlets. In addition, doublets introduces noise in downstream analysis and can worsen the clustering and visualization quality. Thus, identifying and removing doublets becomes a critical data cleaning step in single-cell and single-nucleus RNA-seq (sc/snRNA-seq) data analysis. Based on Scrublet [5] paper's definition, we can classify doublets into two categories: embedded doublets and neotypic doublets. Embedded doublets are composed of highly similar cells and thus are hardly distinguishable from singlets. Neotypic doublets are composed of cells with dissimilar transcriptomes. They are the doublets that cause most trouble. Fortunately, they are also distinguishable from singlets.

Our goal is to identify and remove neotypic doublets. In this manuscript, we will describe a three-step stragety used in Pegasus to identify and remove neotypic doublets. First, Pegasus calculates doublet scores per sample using a slightly modified Scrublet [5] method. Second, Pegasus infers a doublet score cutoff between neotypic and embedded doublets per sample automatically using a method combining Gaussian mixture model and signed curvature scores. Lastly, Pegasus tests if any cluster consists of more neotypic doublet than expected using Fisher's exact test. Clustering should be performed on all samples after batch correction. Users can determine if they want to mark any statistically significant cluster as a neotypic cluster and all cells in a neotypic cluster would be marked as neotypic doublets. This last step is inspired by Pijuan-Sala et al. [3].

In the following sections, we will describe each of the three steps in details.

2 Doublet score calculation

Pegasus reimplements Scrublet [5] with slightly modifications. We reimplemented Scrublet for two reasons: 1) Scrublet source code was not maintained since July 2019; 2) a re-implementation allows us to cut many unnecessary dependencies and gives us more flexibility on future improvements.

Scrublet has three major steps: preprocessing, doublet simulation and doublet score calculation using a KNN classifier. The preprocessing step consists of 4 sub-steps (see Default Preprocessing section of the Scrublet paper): a) data normalization, b) highly variable gene selection, c) data standardization and d) PCA. In our reimplementation, we replac a) and b) with Pegasus data normalization and log transformation [log(TP100K+1)], followed by Pegasus-style highly variable gene selection [1]. It is also worth noting that we directly work on the TP100K matrix in c), instead of log(TP100K+1) matrix.

For the doublet simulation and doublet score calculation steps, we exactly follow the Scrublet method, except that we built kNN graph using Pegasus' kNN building functions [1], which utilizes the Hierarchical Navigable Small World algorithm [2]. For users' convenience, we also provide a brief derivation of how the doublet score is calculated below. More details can be found in the Scrublet paper [5].

Let r be the ratio between simulated doublets and observed doublets, $P_D^{'}(x)$ be the approximated density function of doublets and $P_{obs}(x)$ be the density function of observed cells, which can be used as an approximation of density function of singlets (assuming doublet rate is low). The probability of a simulated doublet appeared in the neighborhood of cell x becomes

$$q(x) = \frac{P_D^{'}(x)r}{P_D^{'}(x)r + P_{obs}}. \tag{1}$$

Let $\hat{\rho}$ be the expected doublet rate, the probability of x is a doublet becomes

$$\mathscr{L}(x) \approx \frac{P_D^{'}(x)\hat{\rho}}{P_D^{'}(x)\hat{\rho} + P_{obs}(x)(1-\hat{\rho})}. \tag{2}$$

Reorganize equation (1), we get

$$P_{obs}(x) = P_D^{'}(x) \cdot \frac{r(1-q)}{q} \cdot (1-\hat{\rho}). \tag{3}$$

Plug equation (3) into equation (2), we get

$$\mathscr{L}(x) = \frac{q(x)\hat{\rho}/r}{(1-\hat{\rho}) - q(x)(1-\hat{\rho}-\hat{\rho}/r)}. \tag{4}$$

Following Scrublet notations, we denote k as the average number of observed cell neighbors and k_{adj} as the total number of neighbors. By default, we have

$$\begin{aligned} k &= \lfloor 0.5 \cdot \sqrt{\text{number of cells}} \rceil, \\ k_{adj} &= \lfloor k \cdot (1+r) \rceil. \end{aligned}$$

If we put a non-informative prior $Beta(1,1)$ on $q(x)$, the expectation of $q(x)$ becomes

$$\langle q(x) \rangle = \frac{k_d(x)+1}{k_{adj}+2}, \tag{5}$$

where $k_d(x)$ is the number of simulated doublets in cell x's neighborhood. Note that the neighborhood here does not include x itself.

Plug equation (5) into equation (4), we get the formula for doublet score as

$$\langle \mathscr{L}(x) \rangle \approx \frac{\langle q(x) \rangle \hat{\rho}/r}{(1-\hat{\rho}) - \langle q(x) \rangle (1-\hat{\rho}-\hat{\rho}/r)}. \tag{6}$$

2.1 Estimate doublet rate prior automatically for 10x Genomics data

In Scrublet, users need to set a doublet rate prior parameter manually, which might be challenging. In Pegasus, we have developed a method to **automatically** estimate this prior based on total number of cells.

We assume the number of cells n entering a droplet or microwell follow a Poisson distribution parameterized by λ, i.e. $n \sim Pois(\lambda)$. Then we can estimate the doublet rate ρ as

$$\rho = \frac{P(n>1)}{P(n>0)} = \frac{(1.0 - e^{-\lambda} - \lambda e^{-\lambda})}{1.0 - e^{-\lambda}}. \tag{7}$$

λ can be interpreted as the rate of an event happening in an interval of time, where the event is a cell entering the droplet or microwell. If we denote N as the total number of cells, it is intuitive to assume that $\lambda(N)$, the rate parameter for capturing N cell in one channel, is proportional to N, or

$$\lambda(N) = c \cdot N. \tag{8}$$

Based on equations (7) and (8), we can estimate $\lambda(N)$ for 10x Genomics data from the multiplet rate table available at 10x Genomics website. Based on the table, we estimated

$$\hat{\lambda}(N) = \frac{0.00785}{500} \cdot N,$$

where 0.00785 is the estimated λ for 500 cells.

If other protocols also provide multiplet rate tables similar to 10x Genomics, we can easily estimate $\lambda(N)$ using equations (7) and (8).

In Pegasus, if users do not provide a doublet rate prior value, we automatically set $\hat{\rho}$ as

$$\hat{\rho} = \frac{(1.0 - e^{-\hat{\lambda}(N)} - \hat{\lambda}(N)e^{-\hat{\lambda}(N)})}{1.0 - e^{-\hat{\lambda}(N)}}.$$

Note that if the data are not 10x Genomics, users may consider to provide a prior value instead of using this automatic feature.

3 Doublet cutoff inference

Scrublet provides a method to determine doublet score cutoff between embedded and neotypic doublets based on simulated doublets. However, this method is far from ideal. Figure 1 showed the Scrublet histograms generated for bone marrow donor 3, channel 1 from the Immune Cell Atlas dataset. We ran Scrublet using default parameters except setting $\hat{rho} = 0.0031$. We can observe that the "ideal" cutoff should be around 0.2, while Scrublet set the threshold in the middle of the "neotypic" doublet peak.

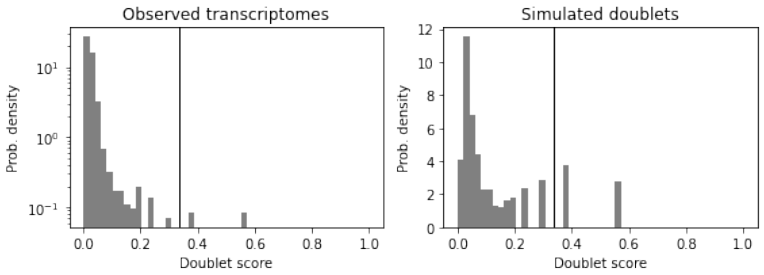

Figure 1: **Scrublet histograms for observed cells (left) and simulated doublets (right).** The vertical line indicates the cutoff.

Thus, we developed a novel method to automatically determine the cutoff in Pegasus. Our method is based on several observations from real data, which we will describe in the following.

First, we observed that log transform the doublet score helps us to group neotypic doublets together. For example, we performed Kernel density estimation (KDE) on both doublet scores and log-transformed doublet scores for simulated doublets (Figure 2). We can clearly observe two peaks on the KDE plot generated from log-transformed scores. Thus, we will work on log-transformed ($\log x$) doublet scores for determining the cutoff.

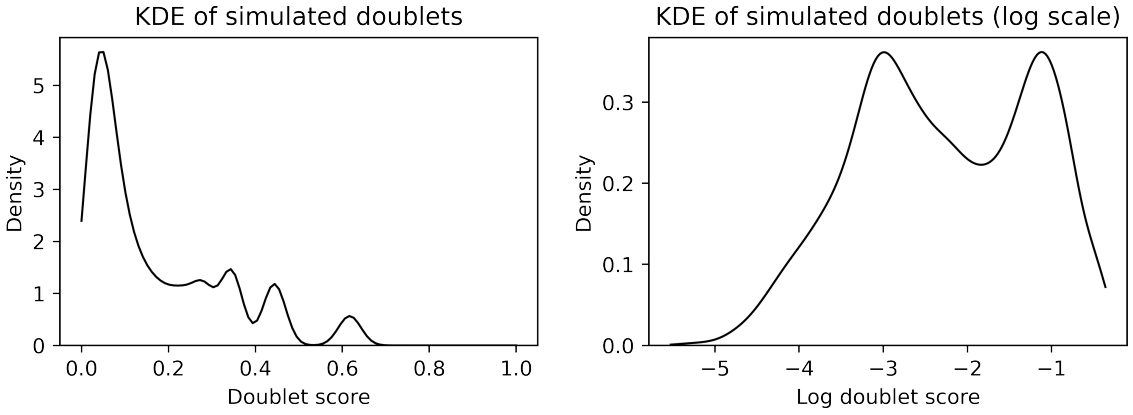

Figure 2: **KDE plots on doublet scores (left) and log-transformed doublet scores for simulated doublets.**

Secondly, we observed that the left peak (embedded doublets, Figure 2, right panel) has a long tail on the left side. This suggests that we'd better to model the left peak with two Gaussian distributions and model the right peak with one Gaussian. Thus, as a first attempt to partition embedded and neotypic doublets, we apply a Guassian Mixture Model (GMM) with three components to fit the log-transformed doublet scores. The Gaussian component with the largest mean corresponds to "neotypic" doublets and the other two components correspond to "embedded" doublets. If we denote s_{embed} as the maximum log doublet score in "embedded" doublets and $s_{neotypic}$ as the minimum log doublet score in "neotypic" doublets, the cutoff based on GMM is

$$s_{gm} = e^{\frac{s_{embed}+s_{neotypic}}{2}}.$$

Figure 3 shows the histograms of observed cells and simulated doublets with the cutoff s_{gm} indicated. We can observe that s_{gm} cutoff makes more sense than the Scrublet cutoff.

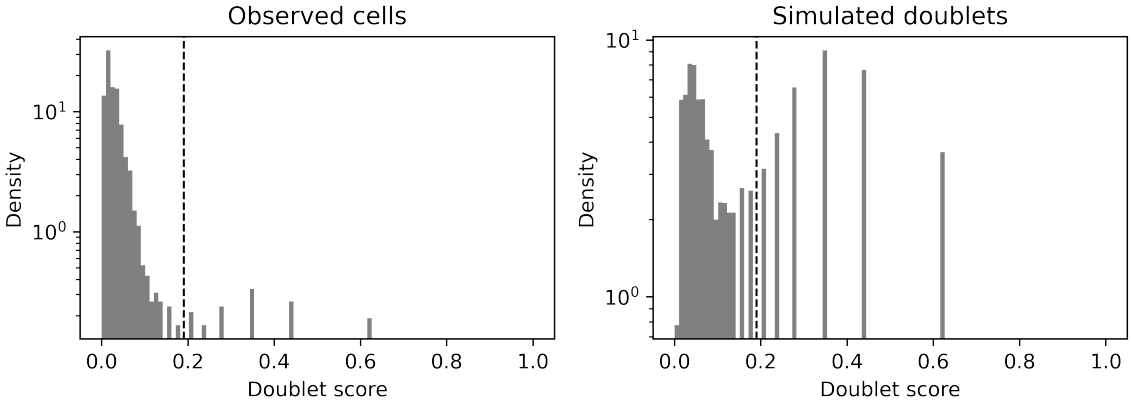

Figure 3: **Histogram of doublet scores for bone marrow donor 3, channel 1 with s_{gm} cutoff.** Histogram density is plotted against doublet score for observed cells (left) and simulated doublets (right). Density is plotted in $\log_{10}$ scale. The black dashed line indicates the cutoff and cells on the right side of the cutoff are neotypic doublets.

Thirdly, there are cases where we can only observe one major peak, such as cord blood donor 3, channel 1 (Figure 4, left) from the Immune Cell Atlas dataset. In this case, the major peak locates in the "embedded" group and the cutoff obtained from GMM seems to be arbitrary.

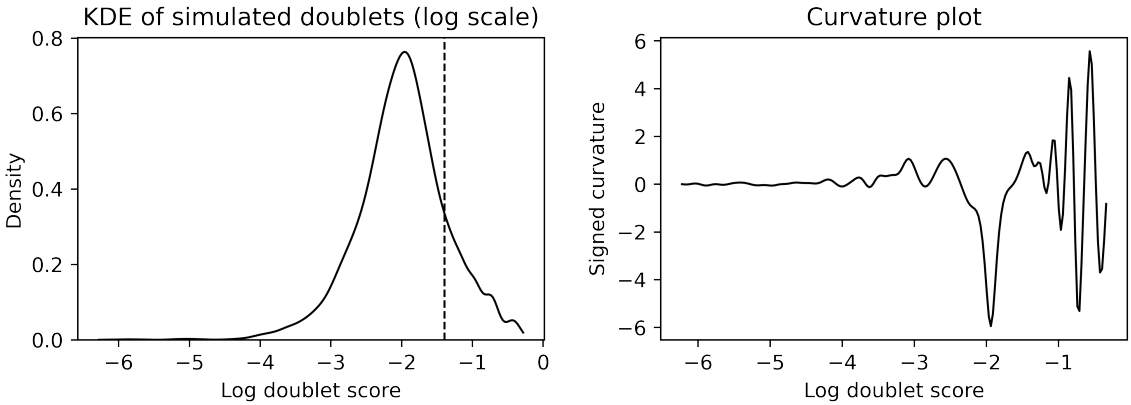

Figure 4: **KDE and signed curvature plots on log-transformed doublet scores for cord blood donor 3, channel 1.** Left is the KDE plot with cutoff s_{gm}, right is the signed curvature plot.

In order to distinguish cases like Figure 2 from cases like Figure 4, we first need to define major peaks. We define a major peak as a local maxima in the KDE that is no smaller than 0.2 of the global maxima, where

0.2 is a threshold determined emperically. We find major peaks by first partition the x axis into no less than 200 bins such as each bin contains at most one distinct log doublet score, and then scan each bin boundary to test for local maxima.

Once we have all major peaks, we consider three cases. 1) We have both major peaks in "embedded" and "neotypic" doublets as determined by GMM. Figure 2 belongs to this case. In this case, the cutoff is the value that results in the minimal density in KDE between the right most major peak in "embedded" doublets and the left most major peak in "neotypic" doublets. 2) We only have major peaks in "embedded" doublets. Figure 4 belongs to this case. In this case, we need to find the cutoff based on signed curvature value [4] (finding knee point or elbow point). Signed curvature $K_f(x)$ can be calculated as

$$K_f(x) = \frac{f^{''}(x)}{(1 + f^{'}(x)^2)^{1.5}}.$$

Based on Figure 4 (right), we can see that local minima with negative signed curvatures in the curvature plot represent peak-like features in the KDE plot. To be robust, we only consider local minima with its curvature value ≤ -1.0. We find the point with first positive curvature value on rightside of the rightmost local curvature minima in "embedded" doublets and denote it as start. We find the point with first positive curvature value on leftside of the leftmost local curvature minima in "neotypic" doublets and denote it as end. We then consider all points in the interval $(\text{start}, \text{end})$. Among the points considered, the one with the largest absolute curvature value is picked as the cutoff. 3) We only have major peaks in "neotypic" doublets. In this case, it is highly likely we only have one major peak, which is for "embedded" doublets, but our GMM wrongly classified this peak to "neotypic". We denote as start the point with first positive curvature value on rightside of the leftmost local maxima and denote as end the point with first positive curvature value on lefside of the first local curvature minima next to the local maxima. The one with the largest absolute curvature value in $(\text{start}, \text{end})$ is selected as the cutoff.

In summary, we have the following method:

1. Log transform doublet scores.
2. Apply a Gaussian Mixture Model with 3 components to the log-transformed scores. Group data into "embedded" and "neotypical" groups.
3. Perform Kernel Density Estimation (KDE) and calculate major peaks and signed curvature scores. Determine cutoff based on three cases: 1) both groups contain major peaks; 2) only "embedded" group contains major peaks; and 3) only "neotypic" group contain major peaks. Assume the cutoff is at x, transform it back to non-log space and get the final cutoff $s_{cutoff} = e^x$.
4. apply cutoff s_{cutoff} to observed cells.

Figure 5 compared the cutoff inferred using GMM and our final algorithm.

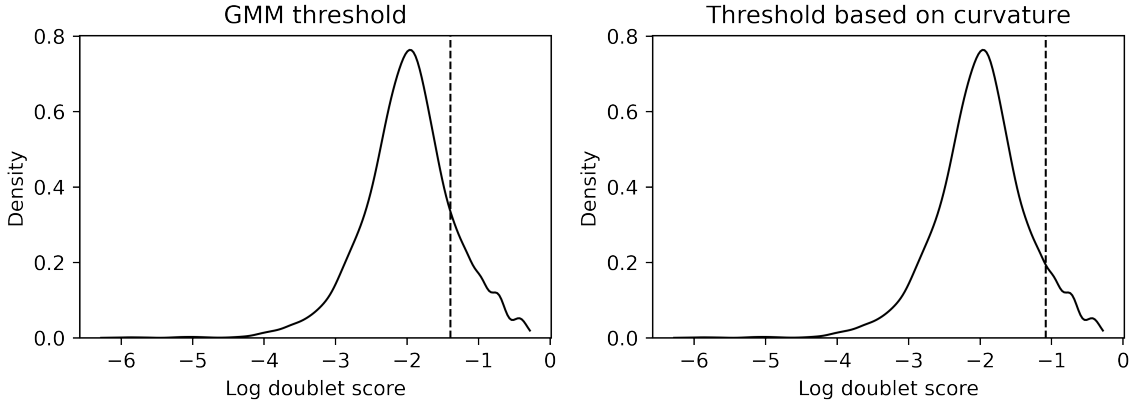

Figure 5: **Cutoff inferred from GMM (left) and the final algorithm (right).**

4 Doublet cluster identification

Once we have identified neotypic doublets, we can assess if a cluster is significantly enriched for doublets using Fisher's exact test by constructing the follow data table. We conduct Fisher's exact test for all clusters and control the False Discover Rate at $\alpha = 0.05$. Among clusters that are significantly enriched for doublets, users can determine if they want to mark some clusters in whole as doublets.

	Within cluster	**Outside cluster**	*Row total*
Singlets	**a**	**b**	a + b
Doublets	**c**	**d**	c + d
Column total	a + c	b + d	a + b + c + d

Table 1: **Data table for Fisher's exact test.** $c + d$ is the total number of identified (neotypic) doublets.

References

[1] B. Li, J. Gould, Y. Yang, S. Sarkizova, M. Tabaka, O. Ashenberg, Y. Rosen, M. Slyper, M. Kowalczyk, A.-C. Villani, T. Tickle, N. Hacohen, O. Rozenblatt-Rosen, and A. Regev. Cumulus provides cloud-based data analysis for large-scale single-cell and single-nucleus RNA-seq. *Nature Methods*, 17(8):793–798, 2020.

[2] Y. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836, 2020.

[3] B. Pijuan-Sala, J. A. Griffiths, C. Guibentif, T. W. Hiscock, W. Jawaid, F. J. Calero-Nieto, C. Mulas, X. Ibarra-Soria, R. C. V. Tyser, D. L. L. Ho, W. Reik, S. Srinivas, B. D. Simons, J. Nichols, J. C. Marioni, and B. Göttgens. A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature*, 566(7745):490–495, 2019.

[4] V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan. Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pages 166–171, 2011.

[5] S. L. Wolock, R. Lopez, and A. M. Klein. Scrublet: Computational identification of cell doublets in single-cell transcriptomic data. *Cell Systems*, 8(4):281–291, 2019.