MarkerCount: A stable, count-based cell type identifier for single cell RNA-Seq experiments

HanByeol Kim
Dankook University

Joongho Lee
Dankook University

Keunsoo Kang
Dankook University

Seokhyun Yoon (syoon@dku.edu)
Dankook University https://orcid.org/0000-0002-0464-2233

Research Article

Keywords: Downstream Analysis, Gene Expression, Batch Effect, Cluster-basis

Posted Date: April 20th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-418249/v2

MarkerCount: A stable, count-based cell type identifier for single cell RNA-Seq experiments

HanByeol Kim[1], Joongho Lee[1], Keunsoo Kang[2] and and Seokhyun Yoon[3,*]

[1] Dept. of Computer Science, College of SW Convergence, Dankook University, Yongin-si, Korea
[2] Dept. of Microbiology, College of Natural Sciences, Dankook University, Cheonan-si, Korea
[3] Dept. of Electronics and Electrical Eng., College of Engineering, Dankook University, Yongin-si Korea
* Corresponding: syoon@dku.edu

Abstract: Cell type identification is a key step to downstream analysis of single cell RNA-seq experiments. Indispensible information for this is gene expression, which is used to cluster cells, train the model and set rejection thresholds. Problem is they are subject to batch effect arising from different platforms and preprocessing. We present MarkerCount, which uses the number of markers expressed regardless of their expression level to initially identify cell types and, then, reassign cell type in cluster-basis. MarkerCount works both in reference and marker-based mode, where the latter utilizes only the existing lists of markers, while the former required pre-annotated dataset to train the model. The performance was evaluated and compared with the existing identifiers, both marker and reference-based, that can be customized with publicly available datasets and marker DB. The results show that MarkerCount provides a stable performance when comparing with other reference-based and marker-based cell type identifiers.

Keyword: cell type identification, single-cell RNA-Seq., marker-based identification, reference-based identification, cell type markers

Background

Single-cell RNA-Seq technology [1-3] enabled transcriptomic analysis of micro environment in heterogeneous tissues [4], such as tumor micro environment, where many cells of different types co-exist. A key step to its analysis is the cell type identification, for which there are many manual annotation tools, such as Seurat[5], SC3[6], SCANPY[7] and Alona[8], by which cell type annotations were made publicly available for further research. Automatic annotation pipelines were also developed, one of which is marker-based approach. This approach utilizes lists of known markers to identify cell types using gene expression profiles obtained from single cell RNA-Seq experiments. Garnett [9], SCINA[10], scSorter[11] and CellAssign[12] fall into this class, for which, there are several utilizable databases for cell type markers such as Panglao DB[13] and CellMarker DB[14]. Another approach is reference-based methods that utilize the existing annotation to obtain cell-type profiles to be used for identification, e.g., SingleR[15], scPred[16], scmap[17], CaSTLe[18] and CHETAH[19]. Various machine

learning techniques, such as classification and clustering, were used as key enabling tools for such types of cell type identification. They are mostly work well, if all the cell types in the test data is already known, i.e., they are in the references or their markers are available. As also noticed in [20], however, one of the important functions of cell type identification is to precisely identify unknown cell clusters too, which can be implemented by suitably setting the rejection thresholds. Although most of the cell type identifiers provide such functions, they look not working so well as shown in [19]. Other things to note are the measurement noise due to insufficient number of RNA molecules per cell and the batch effect caused by different platforms and preprocessing. Most of the cell type identifiers, especially the reference-based ones, utilizes directly the genes expression profiles to train the identification models. Gene expression profiles, however, is subject to noise and batch effect, which may limit the performance.

To overcome such problems, we present MarkerCount, a count-base cell type identifier that support both marker-based and reference-based identification. The overall procedure is shown in Figure 1 and the detailed description can be found in the method section. Briefly speaking, the data processing pipeline consists of two steps, (1) selecting markers for given reference and (2) identification of cell type utilizing "marker counts" and cluster-wise cell type correction, where the second step can be slightly modified to make it marker-based identification utilizing the existing cell type markers, e.g., those in [13] and [14]. The key is to suitably set rejection threshold to determine 'unknown' cell type, for example, unknown tumor cells, that is not in the set of target cell types. Most of the identifiers, either marker based or reference based, showed reasonable performance when there are no unknowns. However, a good identifier must be able to successfully identify such cell clusters for further study on their genomic characteristics not identified so far.

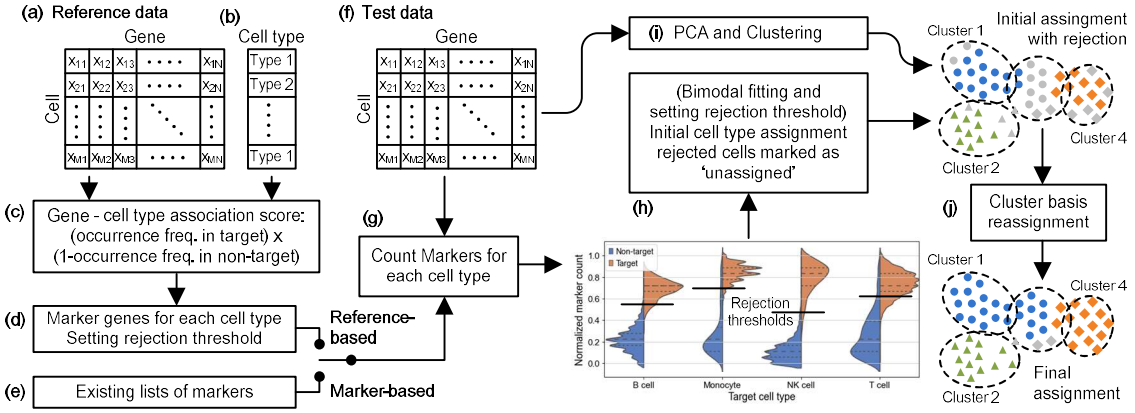

Figure 1 An overview of MarkerCount data processing. MarkerCount operates in both reference-based and marker-based mode. In the former, it requires reference data (a) along with cell type annotation (b) to find makers of each cell type (d), for which gene-cell type association score (c) is used. In marker-based mode, it uses the existing markers (e). In test phase, it take cell type markers and the test data (f) as input. Follwing data processing includes marker count (g), initial cell type assignment and rejection of unclear cells (h), PCA and clustering (i) and the cluster basis reassignment for final identification (j).

Results

For evaluation, we ran MarkerCount in both reference-based and marker-based mode and compared separately with the existing reference-based identifiers [15, 17-19] and the marker-based ones [9-11] with the cell type markers in [14], respectively. To this end, we used 8 single cell RNA-Seq data for which manual annotation is available. Among them, two are the Pancreas data reported in [21] (Pancreas 2K) and [22] (Pancreas 8K), two are peripheral blood [23] (CBMC 8K) and [24] (PBMC 68K) consisting of various immune related cells, two are lung data used in [25] and [26], which can be downloaded from the human cell atlas (https://data.humancellatlas.org/) and the remaining two are from the tissue with tumor [27] (Melanoma 5K) and [28] (Head and Neck 6K). A summary on these datasets is shown in Supplemental Table 1. There are other software packages for cell type identification, such as Moana [29], DigitalCellSorter[30], Cell-BLAST[31], scMatch[32], ACTINN[33], SCSA[34] and mores in [20]. However, some of them do not provide functions to build custom models either utilizing list of markers or reference data with annotation. Therefore, we consider only those with which one can build customized model with gene expression patterns and cell type annotation for reference-based methods and with list of existing markers for marker-based methods.

In the reference-based identification, we considered four pairs of data sets (Pancreas 2K and Pancreas 8K), (CBMC 8K and PBMC 68K), (Lung 29K and Lung 39K) (Melanoma 5K, HeadNeck 6K) separately. For each of these pairs, we performed cross-validation, i.e., circularly used one as test data and others as reference to identify marker genes and to set rejection threshold. To show the stable performance, we also tested tumor data using CBMC 8K or PBMC 68K as reference to find markers for various immune related cells including T cells, B cells, monocytes, dendritic cells and so on. Note that no tumor cells were found in CBMC 8K and PBMC 68K, while there are many tumor cells in Melanoma and HeadNeck data. Ideally, those tumor cells and the cell types not annotated in the reference must be determined as 'unknown' or 'unassigned'.

Figure 2 summarizes the results, where, similar to those in [19], we considered 5 criteria, that is, correct(C), error(E), erroneously assigned (EA), erroneously unassigned (EUA) and correctly unassigned (CUA). These are defined according to (1) whether any valid label assigned by the identifier or not (i.e., marked as unknown, unclear or unassigned), (2) whether the assigned label is in the reference or not and (3) if they are in the reference cell type, whether the predicted label is the same as the assigned label or not. The detailed definitions of the five measures are described in the method section.

Ideally, only C and CUA should exist, which was depicted in the first bar in all figures. Imperfect identification causes E, EA and/or EUA, each of which may have different impact according to the purpose of the analysis. Although too much EUA is undesired, E and EA might have worse impact than EUA in most cases as one can manually characterize clusters predicted as unknown or unassigned. Roughly speaking, it is desirable that the predicted results is as close to the ideal case (the first bar) as

possible. In pancreas, blood and lung data, there is very small portion of cells of which their type is not in the references while, the melanoma and head-neck data contain a large portion of tumor cells that must be identified ideally as unknown (or unassigned).

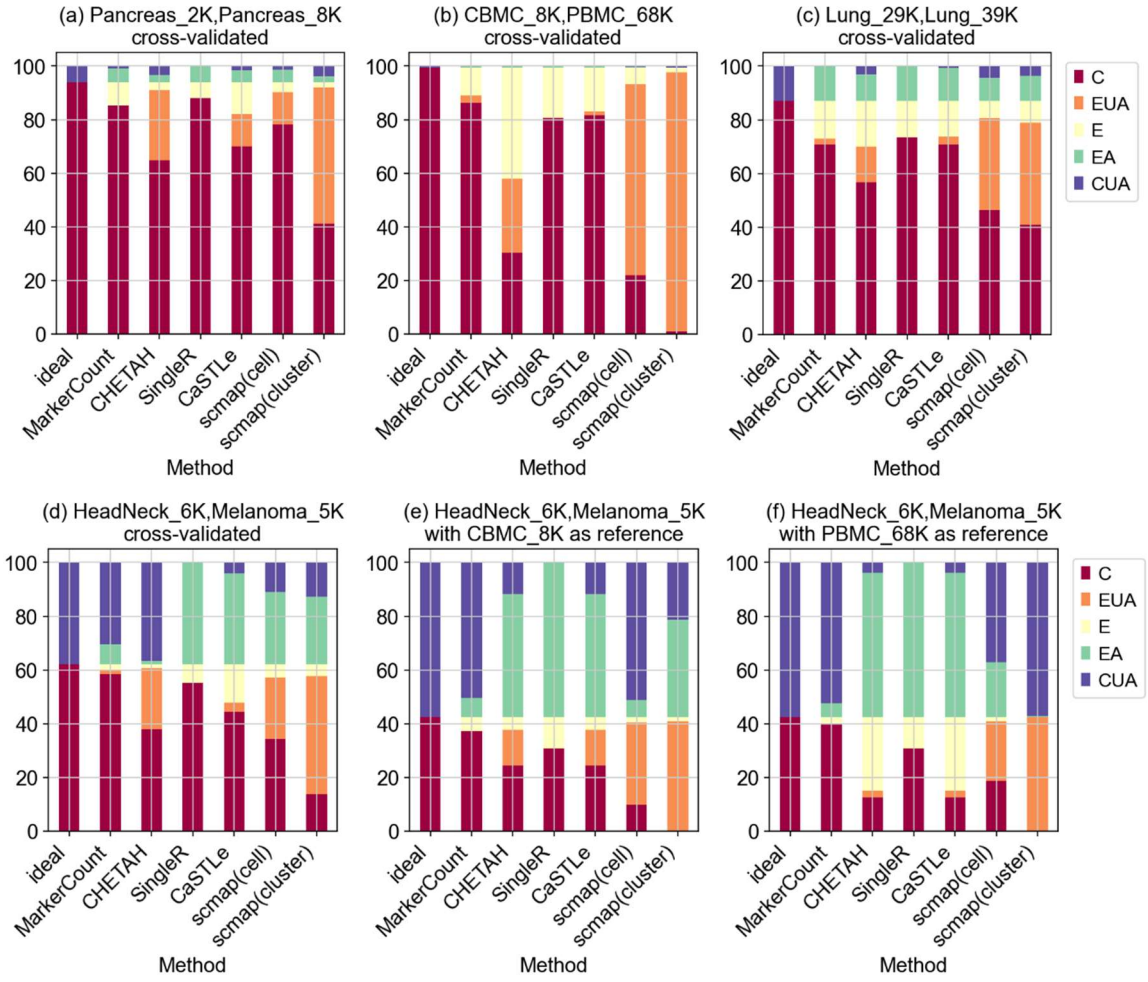

Figure 2 Comparisons of the reference-based cell type identification performances of MarkerCount, CHETAH, SingleR, CaSTLe, scmap (cell) and scmap (cluster). (a) Pancreas data, (b) Peripheral Blood (immune system) and (c) Tissues with tumor. C: Correct, EUA: Erroneously unassigned, E: Error, EA: Erroneously assigned, CUA: Correctly unassigned. The first bar in each figure shows the ideal performance, where CUA account for tumor cells and those cell types not in the reference. Ideally, they must be predicted as unknown (or unassigned).

For the pancreas and lung data, SingleR has shown to be the best in terms of the correct prediction and MarkerCount, scmap(cell), CaSTLe, CHETAH, scmap (cluster) follow. However, if E and EA are desired as small as possible, scmap(cell), scmap(cluster) or CHETAH look better than SingleR, MarkerCount and CaSTLe, even though scmap(cluster) and CHETAH unassigned cell type for large portion of cells. In blood data, MarkerCount showed the best performance in terms of correct prediction. Although scamp(cell) and scmap(cluster) showed very small error, their EUAs are too large and the

correct precision were very small. The prediction accuracy in terms of CUA is highlighted in Figure 1 (d) to (f) for tumor datasets with different references, where, in (e) and (f), we used CBMC 8K and PBMC 68K as references, respectively. In all 3 results, MarkerCount showed the closest pattern to ideal case with small portion of E and EA. Other cell type identifiers, such as CHETAH, scmap(cell) and scmap(cluster), showed smaller E than MarkerCount in (e) and (f). But, they erroneously assigned most of tumor cells to other types of normal cell type. Overall, the MarkerCount showed better performance than others for the blood and tumor datasets. Although its performance for pancreas and lung was not the best, it performed reasonably good.

Unlike other reference-based methods, MarkerCount provides also marker-based identification with a slight modification. Therefore, we also compared MarkerCount with the existing marker-based methods, including Garnett [9], SCINA [10] and scSorter [11] (We couldn't run CellAssign [12] due to installation error). Marker-based approach requires only lists of cell markers for which we used CellMarker DB [14]. We extracted cell markers for pancreas, lung, blood and peripheral blood, where the last two were used for CBMC 8K, PBMC 68K and the two tumor tissues, Melanoma 5K and HeadNeck 6K. The results are summarized in Figure 2 for the four pairs of datasets, pancreas, peripheral blood (CBMC and PBMC), lung and tumor tissues. Similar to the comparison of reference-based cell type identifiers, we measured 5 criteria, C, EUA, E, EA and CUA. For pancreas dataset, scSorter performed slightly better than MarkerCount. However, for all other datasets, MarkerCount showed the best performance.

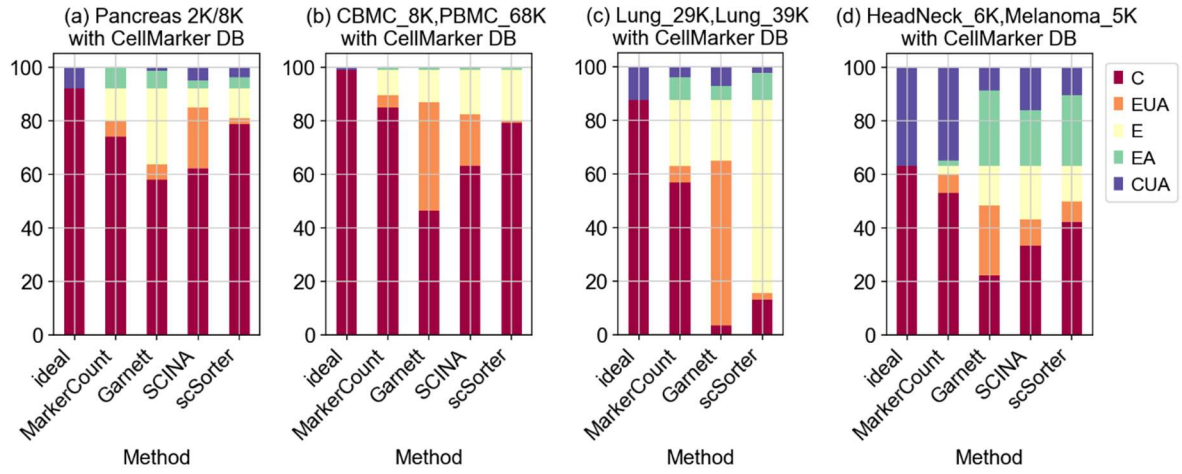

Figure 3 Comparisons of the marker-based cell type identification performances of MarkerCount, scSorter, Garnett, and SCINA. (a) Pancreas data, (b) Peripheral Blood, (c) Lung and (d) Tissues with tumor, using CellMarker DB. C: Correct, EUA: Erroneously unassigned, E: Error, EA: Erroneously assigned, CUA: Correctly unassigned. The first bar in each figure shows the ideal performance, where CUA account for tumor cells and those cell types not in the reference cell type. Ideally, they must be predicted as unknown (or unassigned).

Discussion

Marker-based approach and reference-based approach have their own advantage and disadvantage. In both approaches, one of key components is clustering and, for a good identification, each cell types should be well clustered, which is not always the case. Although many clustering algorithms, such as partitioning-based, distribution-based or graph-based, are available, performance difference with different clustering seems not critical. Rather, handling unclear clusters of highly differentiable cells, as in monocytes, macrophage and dendritic cells, seems to be more important issue.

Reference-based approaches select their own markers internally and look more robust if the reference annotation is sufficiently reliable. However, they can be subject to batch effect when working with datasets from different platforms and preprocessing. Therefore, using the binary indication of gene expression, as in MarkerCount, makes it more robust to the batch effect.

Another point to consider in reference-based approach is the uneven number of cell type population. In Supplemental Figure 1, we plotted Sankey diagram between the manual annotation and the predicted one for blood and tumor datasets, respectively. The figure shows that the cell types having sufficient number of cells shows better performance than those with smaller number of cells. Having large number of cells, they can be well characterized to obtain the best set of markers so that one can better identify those cell types while the identification model for those cell types with small number of cells tends to be overfitted to the data. Doubtlessly, we need more cells to improve the identification performance in most of the reference-based identifiers. The problem is that uneven cell population is inherent. One may combine two or more dataset into one reference data to train the model to improve the performance. However, it may not be always true because of the batch effect. Most of the cell type identifiers utilizes clustering, anyhow, and the identification model is fitted in cluster basis. Problem arises if there are two or more clusters far apart for one cell type due to batch or tissue specific effect. Another issue to consider is the cell types that are highly differentiable, such as monocytes, macrophages and dendritic cells. As shown in Supplemental Figure 1 and 2, some of these cell types were hardly recovered and interchangeably detected. In many cases, they are not well clustered with sufficient separation as shown in Supplemental Figure 2.

Compared to reference-based approach, the marker-based approach does not require reference annotation, which is a big advantage compared to reference-based approach. However, it depends highly on the selection of markers and the best set of markers should be selected taking the specific identification procedures into account. One reason to high dependency on the set of markers seems to be the uneven number of cell markers. In CellMarker DB, several subtypes of dendritic cells have hundreds of marker genes while others, e.g., natural killer cells and B cells, have only around ten markers. Such uneven number of markers may degrade the identification performance and this is why MarkerCount (marker-based mode) reselect markers if too many markers is provided in marker DB (see method section). Depending on the specific procedures and algorithm of cell type identifiers, the

best set of markers will also be different. Literally, 'cell marker' means that it is expressed only in a specific cell type and not expressed at all elsewhere. However, it seems not be always hold since the definition of cell type can be quite different according to the analytical purposes and applications. For example, one can required only identify T cells as a whole in some applications, while one may require identification of specific subtypes, such as cytotoxic T cells, memory T cells, helper T cells and regulatory T cells. Depending on what specific set of cell types are required, different approach and/or different tools must be used. Supplemental Figure 3 and 4 show the Sankey diagram and the UMAP plots similar to those in the reference-based identification experiments. In Supplemental Figure 3 and 4, we see that specific subtypes are interchangeably identified to other subtypes within the same broader cell types. Having markers of both broad and specific cell type, simultaneous identification of broad cell types and specific cell types may cause unexpected results as shown in Supplemental Figure 3. As it is also risky to identify specific cell type directly using only a few markers, a better option is to use hierarchical identification, i.e., identify broad cell type first using many markers and then determine their subtypes using a few specific markers. What specific hierarchy should be used depends on specific tissues to analyze.

Conclusion

Single-cell RNA-Sequencing is certainly a useful tool to analyze heterogeneous tissues, for example, to study tumor micro-environment and cell-differentiation and proliferation, for which cell type identification is a first step to the downstream analysis and automatic identification of cell type may speed up those researches by replacing time-consuming manual annotations with fast and reliable annotation techniques. For medical application, identifying different tumor types and immune cells that co-exist in a tissue may give us a new insight into the personalized cancer therapy. Although the application of single-cell RNA-Seq technology to these studies is still in its early stage, many of the clinical factors and measurements can be obtained by using one single-cell RNA-Seq experiment on the target tissue. And, to this end, more data should be filed with appropriate annotations and cross-checked with different annotation tools.

Methods

The operation of MarkerCount can be divided into two steps, (1) Finding markers for each cell types using reference data and (2) Use markers to identify cell types in cluster basis.

Finding markers

Using the prior annotation provided with the gene expression matrix, reference cell types are identified first. Let G be the set of genes and x_{ij} be the genes expression of the ith cell and jth gene. We first convert them to binary indicator $b_{ij} \in \{0,1\}$ representing whether the gene is expressed in that cell or

not, i.e., $b_{ij} = 1$ if $x_{ij} > 0$ or 0 otherwise. We then compute marker score $s_{m,j}$ for a cell type m as

$$s_{m,j} = \left(\frac{1}{|C_m|}\sum_{i\in C_m} b_{ij}\right)\left(1 - \frac{1}{|\overline{C}_m|}\sum_{i\in \overline{C}_m} b_{ij}\right)^n \quad (1)$$

where C_m is the set of cells annotated as of type m and $\overline{C}_m$ is the complementary set of C_m. $|C_m|$ is the cardinality of C_m. $\frac{1}{|C_m|}\sum_{i\in C_m} b_{ij}$ is simply the occurrence frequency of the jth gene in the mth cell type. n is an hyperparameter relatively weighing the second term. We set it 2 in all the experiments performed in this work. Then, for each cell type m, we sort $s_{m,j}$ in descending order and select first N genes as its markers. N is also an hyper-parameter and we fixed it to 18 for all the cell types in this work. Note that the selection of marker genes is done separately for each cell type and it is possible for two or more cell types to share some common markers. In addition, before comparing the score in (1), one can narrow down the candidate genes by enforcing the condition, $\frac{1}{|C_m|}\sum_{i\in C_m} b_{ij} \geq f_{th}$ for some occurrence frequency threshold f_{th} which we set 0.9.

Marker count

In test phase, for given binary expression profile of the ith cell, b_{ij}, we obtain the normalized marker count $y_{i,m} = \frac{1}{N}\sum_{j\in M_m} b_{ij}$ where M_m is the set of markers of the mth cell type. The cell type is initially determined by taking the maximum of $y_{i,m}$, i.e., $m_i^* = \underset{m}{\operatorname{argmax}}\, y_{i,m}$, which is accepted if $\max_m y_m$ is greater than or equal to the cell type specific rejection threshold, t_m. Otherwise, the cell is marked as 'unassigned'.

Obtaining the rejection threshold

The count threshold t_m can be obtain in various ways. In this work, we considered two approach, i.e., parametric and non-parametric. In non-parametric approach, we directly obtain from y_m's. Consider two set of cells for given rejection threshold t, $C_m(t)$ and $C_m^*(t)$. The former is the set of cells (in the reference data) of which the true (manually annotated) cell type is m and its normalized marker count $y_{i,m} \geq t$, while the latter is the set of cells that are decided to be the mth cell type according to $y_{i,m}$ satisfying (1) $y_{i,m} > y_{i,m'}$ for all other cell types m' and (2) $y_{i,m} \geq t$. Then, the false positive rate can be defined as

$$FPR(t) = 1 - \frac{|C_m(t) \cap C_m^*(t)|}{|C_m(t)|} \quad (2)$$

where the second term is the true positive rate. The objective is to find t such that $FPR(t) \approx p$ for given target *FPR*, p. In non-parametric approach, one can find t by sorting $y_{i,m}$ in descending order and find minimum $y_{i,m}$ such that $FPR(t) \leq p$. In parametric approach, we use univariate Gaussian mixture model with two components, i.e., $y_{i,m} \sim N(\mu_1, \sigma_1^2)$ for $m \in C_m$ and $y_{i,m'} \sim N(\mu_0, \sigma_0^2)$ for $m' \in \overline{C}_m$. Denoting the relative size of C_m and $\overline{C}_m$ as π_0 and π_1, respectively, $FPR(t)$ can be defined as

$$FPR(t) = \frac{F_0(t)}{F_1(t)+F_0(t)} \text{ with } F_k(t) \equiv \int_t^{\infty} \pi_k N(y; \mu_k, \sigma_k^2)\, dy \quad (3)$$

The parametric approach especially useful for marker-based operation of MarkerCount, where reference data is not available. Given test data only, one can resort to the bimodal fitting, for which expectation maximization (EM) algorithm can be used to find π_k, μ_k, σ_k^2 for $k = 0,1$. Note that SCINA also used bimodal fitting. But, it applied to gene expression level, while MarkerCount applies it to the normalized marker counts $y_{i,m}$ to determine the rejection threshold. In reference-based mode, the rejection thresholds for all the cell types are computed in the training phase using the reference data using either (2) or (3), while, in marker-based mode, it can be obtained by applying bimodal fitting to the test data. Supplemental Figure 5 shows comparisons of the two distributions of C_m and $\overline{C}_m$ for some cell types in pancreas, CBMC 8K and PBMC68K.

Cluster basis cell type reassignment

Although marker count based cell type identification works reasonably well, one can improve the identification accuracy, specifically the recall, by employing cluster basis reassignment. The procedure is as follows. We first perform clustering. While some clusters have all its cells assigned by a cell type, others may not be fully covered or not covered at all. The latter might be unknown cells that does not exist in the reference cell type. However, the former may be supposed to be a specific cell type that was not fully covered due to high rejection threshold. We can reassign cell types to those unassigned cells in a partially covered region in cluster by cluster fashion. To this end, we first identify cell types that are partially occupying the cluster. With their centroids, covariance matrices and size, we reassign a cell type to those unassigned by comparing their distances from the centroids as follows.

$$d(\mathbf{z}, \boldsymbol{\mu}_m; \boldsymbol{\Sigma}_m, \pi_m) = \frac{1}{\pi_m}(\mathbf{z} - \boldsymbol{\mu}_m)^T(\boldsymbol{\Sigma}_m + \rho \boldsymbol{I})^{-1}(\mathbf{z} - \boldsymbol{\mu}_m) \quad (4)$$

where $\mathbf{z}$ is a dimension-reduced gene expression profile of a cell, $\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m, \pi_m$ are the centroid (mean), covariance matrix and the relative size of the mth cell type that reside in the cluster and ρ is a regularization constant, which we set 0.1 of the average variances. By comparing the weighted Mahalanobis distance in (4) of an unassigned cell from the centroids for the cell types in that cluster, we decide its cell type to the closest one. For the dimension reduction, we applied principal component analysis (PCA) as typically used in the preprocessing for clustering. The reassignment is performed if the portion of the unassigned cells is below a certain threshold, say 0.8. Although more sophisticated methods can be devised, we used this rather simple heuristic approach.

Reference-based and marker-based operation of MarkerCount

Using the functions described above, the operation of MarkerCount can now be described more succinctly. In the reference-based operation, MarkerCount uses the reference data and its annotation to find the best markers for each cell type in the reference. Using these markers, it computes the

normalized marker counts and decide initial prediction of cell types for all the cells in the reference data. Using the initial prediction and the annotated cell type, the rejection thresholds are obtained. The outputs of the training phase consist of the sets of markers and the rejection threshold for each cell type, which are then used for cell type identification for the test data. In the test phase, we first assign cell types to those cells whose normalized marker count is above the rejection threshold. Then clustering is performed for the test data and the distance-based reassignment is applied in cluster by cluster basis. In marker-based mode, on the other hand, we first compute the normalized marker counts for all the cells and all the cell types utilizing the existing markers. For each cell type, bimodal fitting is applied to determine the rejection threshold for each cell type. One possible problem in the marker-based mode is the uneven number of markers. In this case, cell types with a few markers tend to get higher priority than those with much larger number of markers. This does not happen in the reference-based mode as we select the same or at least similar number of markers for each cell type. To handle this problem, we applied the following tricks.

Resolving uneven number of markers

One possible solution is to reselect markers for those cell types with large number of markers. That is, in the first round of cell type assignment, we assign cell types with higher rejection threshold and use the prediction in the first around to obtain the score in (1) for all the markers and finally reselect markers that have higher scores. This is done only for those cell types that has the number of markers larger than the desired numbers. This approach, however, solve the problem only partially since there exist cell types having only a few or sometimes only one specific markers. To resolve this problem, we applied penalty weight per cell type according to their number of markers as $w = \left(1 + 3e^{-(n-1)/2}\right)^{-1}$, where n is the number of markers. The penalty weight w is multiplied to the normalized marker count before obtaining the rejection threshold and making cell type assignment. Although these tricks were devised for the case when the existing marker DB is used, the best way to avoid such problem in MarkerCount is to use the same or similar number of markers for each cell type, in which case, the penalty weight will be the same for all the cell types so that it does not affect the MarkerCount operation.

Dataset and cell type renaming

To properly evaluate and compare performances, we collected 6 single cell RNA-Seq data with manual annotation available. Those data were used in [19] and partially in other works. The description of the data is summarized in Supplemental Table 1. In all data, we did not take into count those cells annotated as ‘unknown’ or ‘unclear’ in the performance evaluation even though we performed identification anyway.

Two tumor data and CBMC 8K used rather broad cell types, such as T cells, NK cells and dendritic cells, while PBMC 68K used specific types, such as CD8+/CD45RA+ Naive Cytotoxic, CD4+ T Helper2,

CD4+/CD45RA+/CD25- Naive T, CD8+ Cytotoxic T, CD4+/CD45RO+ Memory and CD4+/CD25 T Reg. Therefore, in the comparison of reference-based identifiers, we renamed PBMC cell types to their corresponding broad cell type, e.g., Naive Cytotoxic, T Helper2, Naive T, Cytotoxic T, Memory T and T Reg were all mapped to one broad type, T cell, to train the models for all the reference-based identifier. In marker-based, the two marker DB provide markers for specific cell type along with those for broad cell type. For example, CellMarker DB provides 87 T cell markers along with several specific markers for CD4+, CD8+, CD4+ memory T cells, and so on. Therefore, we used the markers as is for identification of specific cell type. The problem was that, when identifying specific cell types, the performances were unacceptable for all the identifiers we examined, as shown in Supplemental Figure 3. Hence, in the comparison of marker-based identifiers, we renamed after identification, using the predicted specific cell types. As a matter of fact, immune related cells are highly differentiable so that they do not form clearly separable clusters as shown in Supplemental Figure 2 and 4. Moreover, it is questionable that the manual annotation can be used as ground truth since it is typically done in cluster basis, which limits the resolution of cell type clustering.

Performance criteria

Five performance measures, C, E, EA, CUA and EUA, are defined as follows:

A. C is the portion of cells that has an assigned label, one from the reference cell type, and the predicted label is the same as the original assignment.
B. E is the portion of cells that has an assigned label existing in the reference, while the predicted label is different from the original one.
C. EA is those cells that has an assigned label but that is not in the reference and the identifier predicted as one of the reference cell type
D. CUA is those that has an assigned label but that is not in the reference, including tumor cells, and the identifier predicted as unknown (or unassigned).
E. EUA is those that has an assigned label in the reference and the identifier predicted as unknown (or unassigned).

In general, there exists tradeoff between the two. Although it is also informative to show precision versus recall, we used these more specific measures to give better insight into the performance of cell type identifiers.

Cross validation for the reference-based identification

To show the effectiveness of the reference-based cell type identification, we used cross validation, where, with M datasets, one is selected for test while others are used as reference to find cell type markers and this is repeated by circularly shifting their role in the experiment. This was done for the results in Figure 1 (a) to (d). For the results in Figure 1 (e) and (f), the reference and the test data were fixed.

Ethics approval and consent to participate

Not applicable

Consent for publication

Not applicable

Availability of data and materials

All datasets are publicly available and their download site are summarized in Supplemental Table 1. The python code and the example in Jupyter notebook were deposited to Github and can be obtained at https://github.com/combio-dku/MarkerCount/tree/master (Project name: MarkerCount, license: GPL 3.0, Operating system(s): Platform independent, Programming language: python 3, other requirement: None).

Competing Interest

The authors declare that they have no competing interest.

Funding

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2020R1F1A1066320).

Authors' contribution

HK and SY devised the concept and key idea. SY and HK developed the python code. HK and JL performed experiments. SY guided experiments and data analysis. KK gave interpretation and feedback on the results. All authors wrote, read, and approved the final manuscript.

References

1. Tang, F. C., Barbacioru, C., Wang, Y. Z., Nordman, E., Lee, C., Xu, N. L., et al. (2009). mRNA-Seq whole-transcriptome analysis of a single cell. *Nat. Methods* 6, 377–386. doi: 10.1038/nmeth.1315
2. Picelli, S., Bjorklund, A. K., Faridani, O. R., Sagasser, S., Winberg, G., and Sandberg, R. (2013). Smart-seq2 for sensitive full-length transcriptome profiling in single cells. *Nat. Methods* 10, 1096–1098. doi: 10.1038/nmeth.2639
3. Kolodziejczyk, A. A., Kim, J. K., Svensson, V., Marioni, J. C., and Teichmann, S. A. (2015). The technology and biology of single-cell RNA sequencing. *Mol. Cell* 58, 610–620. doi: 10.1016/j.molcel.2015.04.005

4. Chung, W., Eum, H. H., Lee, H. O., Lee, K. M., Lee, H. B., Kim, K. T., et al. (2017). Single-cell RNA-seq enables comprehensive tumour and immune cell profiling in primary breast cancer. *Nat. Commun.* 8:15081. doi: 10.1038/ncomms15081
5. Satija, Rahul, et al. Spatial reconstruction of single-cell gene expression data. Nature biotechnology 2015;33:495-502.
6. Kiselev, Vladimir Yu, et al. SC3: consensus clustering of single-cell RNA-seq data. Nature methods 2017;14:483-486.
7. Wolf, F. Alexander, Philipp Angerer, and Fabian J. Theis. SCANPY: large-scale single-cell gene expression data analysis. Genome biology 2018;19:1-5.
8. Franzén, Oscar, and Johan LM Björkegren. alona: a web server for single-cell RNA-seq analysis. Bioinformatics 2020;36:3910-3912.
9. Pliner, Hannah A., Jay Shendure, and Cole Trapnell. "Supervised classification enables rapid annotation of cell atlases." Nature methods 2019;16:983-986.
10. Zhang, Ze, et al. SCINA: a semi-supervised subtyping algorithm of single cells and bulk samples. Genes 2019;10:531.
11. Guo, Hongyu, and Jun Li. scSorter: assigning cells to known cell types according to marker genes. Genome biology 2021;22: 1-18.
12. Campbell, K. R., S. P. Shah, and A. W. Zhang. Assigning scRNA-seq data to known and de novo cell types using CellAssign. 2019;2019
13. Franzén, Oscar, Li-Ming Gan, and Johan LM Björkegren. PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data. Database 2019;2019
14. Zhang, Xinxin, et al. CellMarker: a manually curated resource of cell markers in human and mouse. Nucleic acids research 2019;47:D721-D728.
15. Aran, Dvir, et al. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. Nature immunology 2019;20:163-172.
16. Alquicira-Hernandez, Jose, et al. scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. Genome biology 2019;20:1-17.
17. Kiselev, Vladimir Yu, Andrew Yiu, and Martin Hemberg. scmap: projection of single-cell RNA-seq data across data sets. Nature methods 2018;15:359-362.
18. Lieberman, Yuval, Lior Rokach, and Tal Shay. CaSTLe–classification of single cells by transfer learning: harnessing the power of publicly available single cell RNA sequencing experiments to annotate new experiments. PloS one 2018;13:e0205499.
19. de Kanter, Jurrian K., et al. CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. Nucleic acids research 2019;47:e95-e95.
20. Abdelaal, T., Michielsen, L., Cats, D. *et al.* A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biol* **20,** 194 (2019)
21. Muraro, Mauro J., et al. A single-cell transcriptome atlas of the human pancreas. Cell systems 2016;3:385-394.

22. Baron, Maayan, et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. Cell systems 2016;3:346-360.
23. Stoeckius, Marlon, et al. Simultaneous epitope and transcriptome measurement in single cells. Nature methods 2017;14:865-868.
24. Zheng, Grace XY, et al. Massively parallel digital transcriptional profiling of single cells. Nature communications 2017;8:1-12.
25. Madissoon, E., Wilbrey-Clark, A., Miragaia, R.J. *et al.* scRNA-seq assessment of the human lung, spleen, and esophagus tissue stability after cold preservation. *Genome Biol* **21,** 1 (2020)
26. Reyfman PA, Walter JM, Joshi N, et al. Single-Cell Transcriptomic Analysis of Human Lung Provides Insights into the Pathobiology of Pulmonary Fibrosis. *Am J Respir Crit Care Med*. 2019;199(12):1517–1536.
27. Tirosh, Itay, et al. Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. Science 2016;352:189-196.
28. Puram, Sidharth V., et al. Single-cell transcriptomic analysis of primary and metastatic tumor ecosystems in head and neck cancer. Cell 2017;171:1611-1624.
29. Wagner, Florian, and Itai Yanai. Moana: A robust and scalable cell type classification framework for single-cell RNA-Seq data. BioRxiv 2018;456129.
30. Domanskyi, Sergii, et al. Polled Digital Cell Sorter (p-DCS): Automatic identification of hematological cell types from single cell RNA-sequencing clusters. BMC bioinformatics 2019;20:1-16.
31. Cao, Zhi-Jie, et al. Searching large-scale scRNA-seq databases via unbiased cell embedding with Cell BLAST. Nature communications 2020; 11:1-13.
32. Hou R, Denisenko E and Forrest ARR, scMatch: a single-cell gene expression profile annotation tool using reference datasets, *Bioinformatics 2019;* 35(22): 4688–4695
33. Ma F and Pellegrini M, ACTINN: automated identification of cell types in single cell RNA sequencing, *Bioinformatics 2020;* 36(2):533–538
34. Cao Y, Wnag X and Peng G, SCSA: A Cell Type Annotation Tool for Single-Cell RNA-seq Data, Front. Genet. 2020; 11:490

Figures

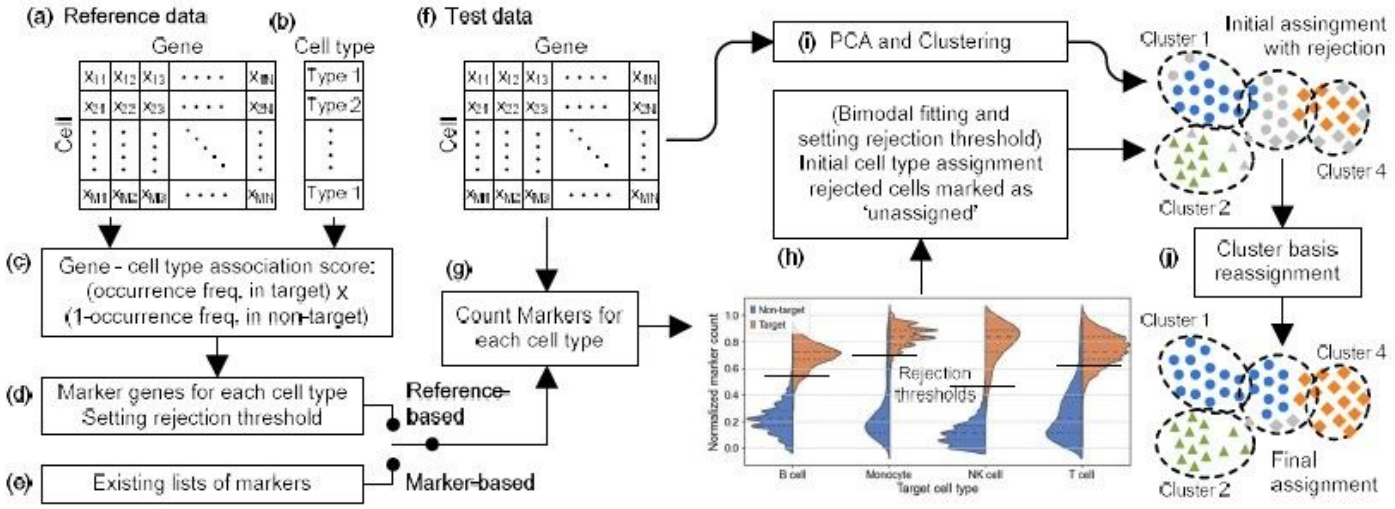

Figure 1

An overview of MarkerCount data processing. MarkerCount operates in both reference-based and marker-based mode. In the former, it requires reference data (a) along with cell type annotation (b) to find makers of each cell type (d), for which gene-cell type association score (c) is used. In marker-based mode, it uses the existing markers (e). In test phase, it take cell type markers and the test data (f) as input. Following data processing includes marker count (g), initial cell type assignment and rejection of unclear cells (h), PCA and clustering (i) and the cluster basis reassignment for final identification (j).

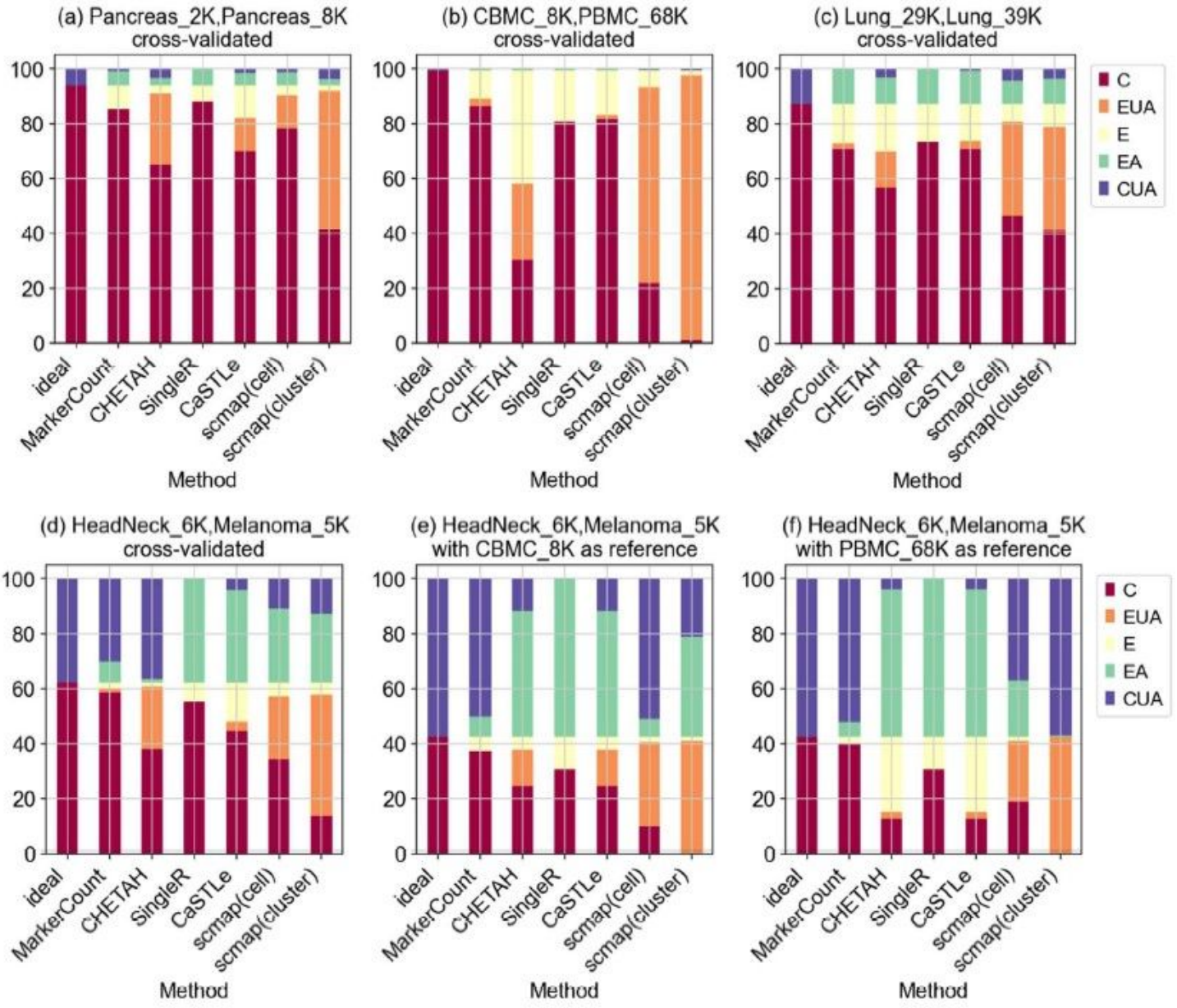

Figure 2

Comparisons of the reference-based cell type identification performances of MarkerCount, CHETAH, SingleR, CaSTLe, scmap (cell) and scmap (cluster). (a) Pancreas data, (b) Peripheral Blood (immune system) and (c) Tissues with tumor. C: Correct, EUA: Erroneously unassigned, E: Error, EA: Erroneously assigned, CUA: Correctly unassigned. The first bar in each figure shows the ideal performance, where CUA account for tumor cells and those cell types not in the reference. Ideally, they must be predicted as unknown (or unassigned).

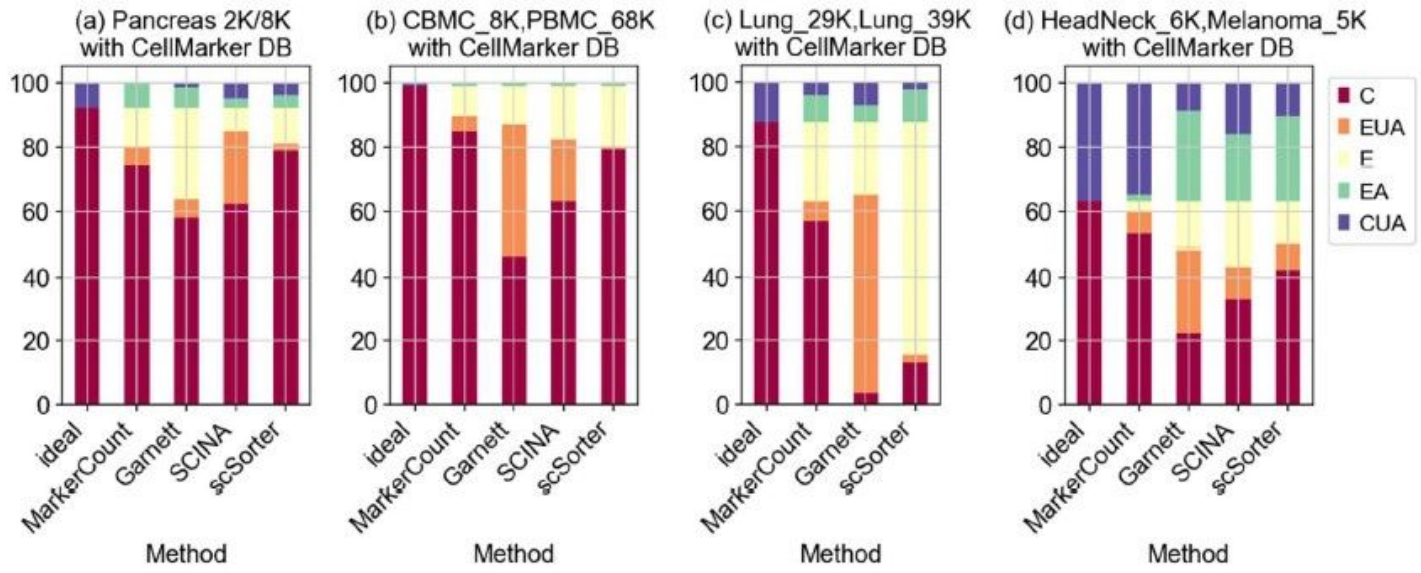

Figure 3

Comparisons of the marker-based cell type identification performances of MarkerCount, scSorter, Garnett, and SCINA. (a) Pancreas data, (b) Peripheral Blood, (c) Lung and (d) Tissues with tumor, using CellMarker DB. C: Correct, EUA: Erroneously unassigned, E: Error, EA: Erroneously assigned, CUA: Correctly unassigned. The first bar in each figure shows the ideal performance, where CUA account for tumor cells and those cell types not in the reference cell type. Ideally, they must be predicted as unknown (or unassigned).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementalMaterials.pdf