Bioinformatics, 38(5), 2022, 1287–1294
https://doi.org/10.1093/bioinformatics/btab807
Advance Access Publication Date: 2 December 2021
Original Paper

Gene expression

Isoform-level quantification for single-cell RNA sequencing

Lu Pan[1], Huy Q. Dinh[2,3], Yudi Pawitan[1] and Trung Nghia Vu [1,*]

[1]Department of Medical Epidemiology and Biostatistics, Karolinska Institutet, 17177 Stockholm, Sweden, [2]McArdle Laboratory for Cancer Research, Department of Oncology, School of Medicine and Public Health, University of Wisconsin—Madison, Madison, WI 53705-227, USA and [3]Department of Biostatistics and Medical Informatics, School of Medicine and Public Health, University of Wisconsin—Madison, Madison, WI 53726, USA

*To whom correspondence should be addressed.

Associate Editor: Anthony Mathelier

Received on July 26, 2021; revised on November 19, 2021; editorial decision on November 24, 2021; accepted on November 25, 2021

Abstract

Motivation: RNA expression at isoform level is biologically more informative than at gene level and can potentially reveal cellular subsets and corresponding biomarkers that are not visible at gene level. However, due to the strong 3′ bias sequencing protocol, mRNA quantification for high-throughput single-cell RNA sequencing such as Chromium Single Cell 3′ 10× Genomics is currently performed at the gene level.

Results: We have developed an isoform-level quantification method for high-throughput single-cell RNA sequencing by exploiting the concepts of transcription clusters and isoform paralogs. The method, called Scasa, compares well in simulations against competing approaches including Alevin, Cellranger, Kallisto, Salmon, Terminus and STARsolo at both isoform- and gene-level expression. The reanalysis of a CITE-Seq dataset with isoform-based Scasa reveals a subgroup of CD14 monocytes missed by gene-based methods.

Availability and implementation: Implementation of Scasa including source code, documentation, tutorials and test data supporting this study is available at Github: https://github.com/eudoraleer/scasa and Zenodo: https://doi.org/10.5281/zenodo.5712503.

Contact: trungnghia.vu@ki.se

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

RNA expression measurement based on sequencing technology (RNA-Seq) has progressed from bulk sequencing to highly multiplexed single-cell (sc) RNA-Seq (Svensson *et al.*, 2018). Computational methods have been developed to analyze single-cell RNA sequencing (scRNA-Seq) data from high-throughput protocols such as Chromium Single Cell 3′ 10× Genomics (Melsted *et al.*, 2021; Srivastava *et al.*, 2019; Zheng *et al.*, 2017), but because of the strong 3′ bias due to polyA-based mRNA capture in library preparation, these methods only attempt gene-level quantification.

Estimation of isoform expression is still a challenging problem for bulk RNA-Seq, and the problem is even harder for scRNA-Seq (Westoby *et al.*, 2020). We provide evidence that the use of methods originally developed to estimate transcript abundance for bulk RNA-Seq data (Bray *et al.*, 2016; Patro *et al.*, 2014, 2017) indeed leads to substantial quantification errors. The 3′ bias from high-throughput scRNA-Seq such as Chromium Single Cell 3′ 10× Genomics generates substantial similarities between the read statistics from different isoforms, making isoform-level quantification highly challenging. The 3′ bias reduces the number of reads from the 5′ end of the isoforms, thus reduces our ability to separate the isoforms. Furthermore, the bias can also break the symmetry in read sharing between isoforms which is often presumed by bulk RNA-Seq quantification methods. In the symmetric case, if isoform A generates reads that are shared with (map to) isoform B, then isoform B will likewise generate reads that are shared with isoform A. But in biased protocol, we observe asymmetric cases where, e.g. all shared reads between several isoforms originate from a single isoform A, but none of the other isoforms contribute reads shared with isoform A. The asymmetry happens when the 3′ exons of one isoform are close to the 5′ end of the other isoforms. Applying a bulk-based method, which ignores asymmetry, will lead to poor quantification. Several groups have made a great effort to estimate isoform-level expression from full-length scRNA-Seq data (Hu *et al.*, 2020; Huang and Sanguinetti, 2017; Song *et al.*, 2017). Recently, STARsolo (Kaminow *et al.*, 2021) has attempted to quantify splicing events occurring at 3′-end of transcripts in droplet-based scRNA-Seq data. However, this method does not perform isoform expression estimation. To our best knowledge, to date no method has been developed for isoform expression quantification from 3′ bias high-throughput scRNA-Seq data. To address these issues, we have developed

Scasa to estimate isoform expression from scRNA-Seq data by relying on the concepts of transcription clusters and isoform paralogs.

Conceptually, the most similar strategy is the transcript compatibility count (TCC) method (Ntranos *et al.*, 2016), providing group rather than individual transcript counts by calculating read counts in groups of highly similar transcripts, also known as equivalence classes of transcripts (Patro *et al.*, 2014). In principle, the equivalence classes obtained from TCCs provide a basis for single-cell-level transcript quantification by applying the expectation maximization (EM) algorithm using bulk RNA-Seq methods such as Kallisto (Bray *et al.*, 2016) or Salmon (Patro *et al.*, 2017). Another attempt is introduced in Terminus (Sarkar *et al.*, 2020) to use transcript grouping for quantification of isoform expression from bulk RNA-Seq data. This method utilizes the output of Salmon to group together transcripts in an experiment based on their inferential uncertainty. We have implemented these bulk-based approaches as comparative methods and refer to them with their original names. [We note that the authors of these methods have developed new tools (Melsted *et al.*, 2021; Srivastava *et al.*, 2019) for scRNA-Seq data that are gene-based, so the use of the original bulk RNA-Seq tools here is performed only for the purpose of comparing isoform-level quantification results.]

2 Materials and methods

2.1 Overview of Scasa

An overview of the Scasa procedure for isoform expression estimation is presented in Figure 1a. The Scasa protocol consists of three main components: (i) estimation of transcript abundance (β) using an alternating expectation maximization (AEM) algorithm, (ii) processing of scRNA-Seq data to produce read-count data (Y) and (iii) *in silico* identification of transcription clusters (TCs) and isoform paralogs and construction of an initial design matrix (X) for each TC. Scasa is tailored for use with high-throughput scRNA-Seq technologies with a unique molecular identifier (UMI) barcoding procedure. Scasa takes advantage of the efficient preprocessing already provided by existing pseudoaligners such as Kallisto-bustools (Melsted *et al.*, 2021) or Alevin (Srivastava *et al.*, 2019) to produce a read-count equivalent-class matrix. Using the unique barcode sequences, Scasa splits the equivalence class output by cell (Fig. 1a) and applies the AEM algorithm to multiple cells together. These procedures are described in detail below.

2.2 Statistical model and AEM algorithm

Each read r_i from an RNA-Seq dataset maps to a set of k isoforms $(T_{i1}, \ldots, T_{ik})$; this set defines an equivalence class (eqClass) of all reads that map to the set. Therefore, conceptually, all sequence reads from a cell can be summarized in a read-count vector y that represents all eqClasses from the cell. This is performed separately for each cell. In theory, selecting a read from an annotated transcript sequence, which is similar to the protocol to generate a read of RNA sequencing, can be considered as a Poisson process. Here, the process is extended to approximate the selection from the sequences of the transcript set of an equivalence class. So, the underlying statistical model in Scasa assumes that y is Poisson with mean μ that follows the bilinear model

$$\mu = X\beta, \tag{1}$$

where μ is the vector of the expected number of reads mapped to all eqClasses, β is the vector of isoform expression values and X is the design matrix summarizing exon sharing between isoforms. The elements of X transfer the transcript abundance to eqClass counts. Standard isoform-level quantification tools such as Kallisto (Bray *et al.*, 2016) or Salmon (Patro *et al.*, 2017) use the same linear model, but the X matrix is assumed to be known and appears only implicitly in their algorithm (computed based on exon sharing between isoforms). In contrast, in Scasa, the matrix X is explicit, and both X and β are treated as unknown parameters.

Hg38 is used as the transcriptome reference, β is of length ~71 000 isoforms, and y of length ~136 000, so X is a matrix of size ~136 000 × 71 000. The estimation of X is clearly not feasible without exploiting the fact that isoforms are naturally organized into independent TCs, so X can be broken down into many small Xs that can be analyzed separately. In principle, once an initial X is available for each TC, given $Y \equiv (y_1, \ldots, y_n)$ the matrix of collated read-count data from n cells, the estimation proceeds as follows:

0. Start with an initial X.
1. Given Y and X, use the EM algorithm to estimate isoform abundance (β) (EM Step 1 in Fig. 1a) based on the linear model $\mu_{jc} = \sum_t x_{jt}\beta_{tc}$, where the subscript j refers to the equivalent class, t refers to the transcript and c refers to the cell. The algorithm applies to each cell.
2. Given Y and β, use the EM algorithm to update design matrix X (EM Step 2 in Fig. 1a) based on the linear model $\mu_{jc} = \sum_t \beta_{tc}x_{jt}$, where data from all the cells are now combined.
3. Iterate between 2 and 3 until convergence.

The joint estimation procedure is called an AEM algorithm, for which the exact formulas are given in Deng *et al.* (2020). At convergence, the output β represents the estimated transcript abundances for individual cells.

2.3 Processing of scRNA-Seq to produce read-count data

Scasa allows the use of scRNA-Seq data from high-throughput scRNA-Seq protocols, such as the Chromium Single Cell 3′ 10× Genomics protocol, and includes read mapping to a reference transcriptome and counting of the supporting reads of eqClasses from each cell. Read mapping and read counting for eqClasses are implemented using available external tools such as Alevin (Srivastava *et al.*, 2019) or Kallisto-bustools (Melsted *et al.*, 2021). Computational details are shown in the Scasa Wiki page on GitHub repository: https://github.com/eudoraleer/Scasa_Paper/wiki.

2.4 Construction of initial design matrix *X*

Droplet-based protocols, such as the Chromium Single Cell 3′ 10× Genomics approach, typically produce strongly 3′-biased sequences due to the polyA tail capture of mRNAs. This violates the standard assumption applied in isoform quantification methods for bulk RNA-Seq, which presumes that the RNA sequences are relatively uniformly distributed. The key methodological innovation of Scasa is the *in silico* construction of the TCs, each with a corresponding initial design matrix X that adapts to the actual sequencing protocol used. As discussed previously (Deng *et al.*, 2020), X also automatically accounts for unknown biases in a sequencing protocol. Moreover, an explicitly available X makes the statistical processing of the paralogs (isoforms with highly similar sequences) tractable.

We constructed the initial X matrix *in silico* for all isoforms in the transcriptome. However, instead of generating standard bulk RNA-Seq paired-end reads with a uniform read distribution, we simulated scRNA-Seq mimicking the settings of the Chromium Single Cell 3′ 10× Genomics method (see Section 2.6). Briefly, each isoform was simulated with the number of reads set to twice the transcript length but with a minimum of 1000. The steps are as follows:

1. Map each read to the transcriptome reference; hg38 is used throughout.
2. Identify the eqClasses associated with each isoform.
3. Group the isoforms into TCs such that all isoforms that belong to the overlapping eqClasses are included in one cluster.
4. Summarize all the reads from the eqClasses and isoforms that belong to one TC in a matrix; see Supplementary Table S1a for illustration.

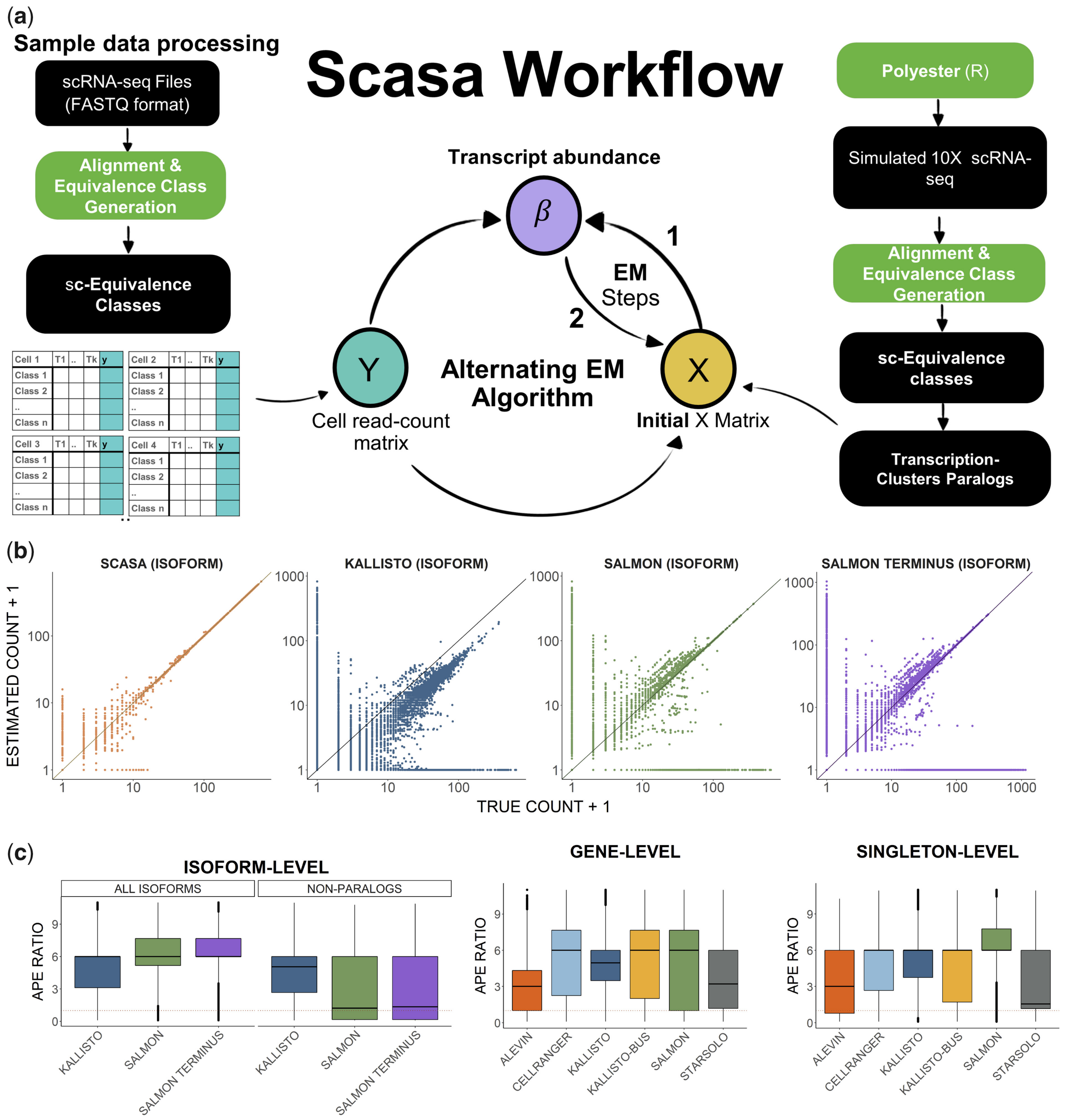

Fig. 1. Scasa workflow and its performance against existing quantification tools. (**a**) The Scasa workflow consists of three main parts: (i) fitting the statistical model $Y \sim X\beta$ using an AEM algorithm, based on (ii) mapping of the scRNA-Seq data to produce count matrix Y and (iii) the *in silico* construction of the transcription clusters and isoform paralogs to obtain the initial X matrices. (**b**) A simulation study ($n = 3955$ cells) indicates that Scasa performs well against existing methods in terms of isoform quantification. Isoform-level estimates are plotted against the true values. (**c**) From left to right: boxplots of the APE ratios of Kallisto and Salmon against Scasa for all isoforms ($n = 12\,203$ truly expressed isoforms) and for non-paralog isoforms ($n = 1342$); boxplots of APE ratios for gene-level quantification methods against Scasa ($n = 8052$ truly expressed genes); boxplots of APE ratios for singleton genes ($n = 2318$)

5. Normalize the columns of the matrix to 1. This is the initial X matrix associated with a TC.
6. Identify the paralogs and merge the appropriate columns of X; see Supplementary Table S1b. This is described in more detail below.

We used the approach of Deng *et al.* (2020) to merge highly similar isoforms into isoform paralogs. Theoretically, the number of non-zero singular values of X determines the number of estimable paralogs; in practice, we set a low non-zero threshold (1/30). The paralogs are then constructed using k-means clustering. For illustration purposes, we use the TC associated with the RPL13A gene, which contains five isoforms. The original X matrix is the normalized version of the matrix in Supplementary Table S1a. Its singular values are 1.632, 0.868, 0.692, 0.004 and 0.001, clearly indicating the need to reduce the parameters to three estimable paralogs. k-means clustering produces one paralog of size three and preserves two original isoforms. The merged X matrix is shown in Supplementary Table S1b; the paralog members show highly similar columns in Supplementary Table S1a. Not using the paralogs, e.g. the standard tool Salmon, results in poor estimation of the expression of the isoforms (see Supplementary Fig. S2b and c). In contrast, Scasa estimates paralog expression well.

2.5 Accuracy metrics

For comparisons between different methods applied to simulated data, we computed a symmetric absolute proportion error (APE) for each transcript i in each cell as

$$\mathrm{APE}_i = \left|\frac{t_i - e_i}{t_i + e_i + 1}\right|, \quad (2)$$

where t_i is the true abundance of individual gene i or transcript i, and e_i is the corresponding estimated abundance. A pseudocount of 1 is added to the denominator to avoid division by zero. In the real sample of human peripheral blood mononuclear cells (PBMCs) upon which we base the simulation, there is a large proportion of isoforms/genes (~93–94%) with zero expression across cells, and the estimation methods produce concordant zero APE values for them, so they are not informative toward the comparisons. Theoretically, in a paired comparison between two methods, the comparison of performance should be more sensitive in situations where the two methods have discordant APE values than in situations where they have the same APE. We thus limit the comparisons to transcripts (in any cell) that produce discordant APE values, and we can meaningfully compute their ratios. The APE ratio of transcript i (in any cell) between Scasa and a competing method is computed as

$$R_i(\mathrm{Method}) = \frac{\mathrm{APE}_i(\mathrm{Method}) + 0.1}{\mathrm{APE}_i(\mathrm{Scasa}) + 0.1}, \quad (3)$$

where the constant 0.1 is added to reduce the variability in the ratio because the APE values are often close to zero. The effect is also to attenuate the ratio. For example, if $\mathrm{APE}_i(\mathrm{Method}) = 0.2$ and $\mathrm{APE}_i(\mathrm{Scasa}) = 0.01$, the raw ratio is 20, but the attenuated ratio is $0.3/0.11 = 2.73$. A ratio >1 indicates that Scasa performs better. Ratios are summarized and reported in boxplots. For a simpler comparison, we also compute the proportion of $\mathrm{APE}_i(\mathrm{Scasa}) < \mathrm{APE}_i(\mathrm{Method})$ whenever the two APE values are discordant. To make the paralogs comparable, they are split into individual members, each carrying the APE of the source paralog.

2.6 Simulated dataset

The purpose of generating simulated data was to ensure that accuracy could be measured against the ground truth. We also sought to compare our software with other existing quantification software using simulated data. We first created simulated 3′-biased paired-end scRNA-Seq data using the Polyester RNA-Seq read simulator (Frazee *et al.*, 2015) with human reference genome hg38 to model the simulation as close as to the end-biased real data from 10× Genomics as possible by mimicking the read length and fragment length distribution (i.e. average and standard deviation). We revised the biased model of Polyester so that its bias was similar to the 3′ bias of the Chromium Single Cell 3′ 10× Genomics method. Specifically, for a transcript read pair, we fixed the 3′ read at the 3′ end of the transcript and varied the 5′ read. All parameters of the simulation are listed in Scasa Wiki on GitHub.

Since we were measuring the accuracy of counts at the single-cell transcript level, we also modeled the transcript-count ratio of a given gene in the simulated data by using the single-cell transcript-ratio reference from full-length Smart-Seq2 data (Ding *et al.*, 2020). Since Smart-Seq2 is a full-length RNA-Sequencing method for single cells, isoform quantification could be used without strong end bias effect to infer transcript ratios for our simulation. To mimic actual cell-wise and gene-wise read depth from droplet-based scRNA-Seq data, we referenced the read depth for each gene from one healthy human PBMC donor [Single Cell Gene Expression Datasets (Single Cell 3′ v3), Chromium Connect Channel 1, 10× Genomics] (February 28, 2020).

We post-processed the simulated output from Polyester by replacing complementary read 1 of the paired-end reads with cell barcode and UMI sequences that were representative of the actual sample data from 10× single-cell gene expression data (see Supplementary Fig. S9). The 16-base-pair (bp) cell barcodes used in these cases were taken from the most recent barcode whitelist of 10× Genomics, which includes a total of 3 million cell barcodes that can be sampled. The 12-bp UMI sequences that were used to identify unique RNA molecules were generated at random to ensure that all unique generated RNA molecules were covered. Thus, no correction was needed for cell barcodes or UMIs in these simulated data. The final paired-end single-cell-level simulated data consisted of read 1 as the read sequence containing cell barcode and UMI information, while each of complementary read 2 contained information of the mapped sequences corresponding to the unique cell and unique RNA molecule, as indicated in read 1. The detailed code for producing the simulated data is provided on the Scasa GitHub website.

2.7 Comparative methods

We ran the simulated data with (i) Scasa (single-cell isoform/gene expression quantification), (ii) Cellranger (single-cell gene expression quantification), (iii) Kallisto-bustools (single-cell gene expression quantification), (iv) Alevin (single-cell gene expression quantification), (v) STARsolo (Kaminow *et al.*, 2021) (single-cell gene expression quantification), (vi) Kallisto (bulk RNA-Seq isoform/gene expression quantification), (vii) Salmon (bulk RNA-Seq isoform/gene expression quantification) and (viii) Terminus (Sarkar *et al.*, 2020) (bulk RNA-Seq isoform expression quantification, which uses the output of Salmon for transcript grouping) and compared their quantification results with the ground truth from simulated data. It is worth noting that in the simulated dataset, we focused on the performance of gene/isoform quantification by individual methods. Barcode and UMI correction were not assessed in the simulation setting and would not have any effect on the results. Both Kallisto and Salmon were run on the RNA reads in the single-end mode of the individual tools to quantify isoform- and gene-level expression. Gene expression levels were calculated with Scasa by summing up the isoform counts of each gene; paralog expression was calculated similarly by summing up the member isoforms.

2.8 Real datasets

2.8.1 CITE-Seq data

Bone marrow mononuclear cell CITE-Seq single-cell data from a recently published paper (Stuart *et al.*, 2019) (GEO accession number: GSE128639) were used as a real data benchmark. RNA-Seq measurements from this dataset were used for comparison and validation. For comparison, we ran this dataset with Scasa to obtain gene-level and isoform-level gene expression, we used Alevin to obtain gene-level gene expression values following the Alevin tutorial (https://combine-lab.github.io/alevin-tutorial/2020/alevin-features/), and we collected the gene-level gene expression data of Cellranger from the original publication downloaded from https://github.com/satijalab/seurat-data. For fair comparison with Cellranger, only single cells with the same barcode IDs among Cellranger, Alevin and Scasa were used ($n = 20\,840$ cells).

Dimension reduction for abundance estimates was carried out for each of the quantification methods used (i.e. Scasa, Alevin and Cellranger) for the CITE-Seq data, and clustering was performed with homogenous settings for each method. Specifically, we followed the description from the original study (Stuart *et al.*, 2019) and the tutorial from the vignettes of Seurat package version 4.0.0. Normalization, feature selection and standardization for isoform/gene expression were performed with the default settings by using the NormalizeData, FindVariableFeatures and ScaleData functions of the Seurat package (Stuart *et al.*, 2019), followed by the RunPCA and RunUMAP functions for dimension reduction. Clusters were identified based on the first 30 PCA components using the FindNeighbors and FindClusters functions. The pathway analysis for the DE gene set was performed by using Reactome (Jassal *et al.*, 2020).

Details of the scripts used for implementation are provided in the Scasa Wiki and on the Scasa website. To obtain differentially expressed markers between the TY32.25 Mono and CD14 Mono groups, we used the logistic regression differential expression (DE) test (Ntranos *et al.*, 2019) implemented with the FindMarkers function of Seurat. Only genes/isoforms detected in 25% of either of the two groups were considered. To correct for the occurrence of false positives brought about by multiple testing, we used the false discovery rate (FDR) (Pawitan *et al.*, 2005). Significant markers were chosen according to a threshold of an FDR < 0.05.

2.8.2 Smart-Seq2 bone marrow data

Smart-Seq2 scRNA-Seq data (ArrayExpress: E-MTAB-9067) for 3055 bone marrow cells from the femora and hips of 15 human fetuses were collected for the validation of TY32.25. The annotations of the cells were collected from the original study (Ranzoni *et al.*, 2021). We applied Salmon (Patro *et al.*, 2017) to the Smart-Seq2 scRNA-Seq dataset using the hg38 annotation to perform isoform quantification. We then used the same procedure applied to the CITE-Seq data for cluster analysis to identify clusters from the isoform expression data of the bone marrow cells.

3 Results

3.1 Strong 3′ bias challenges isoform quantification in scRNA-Seq

The key difference between bulk RNA-Seq and scRNA-Seq data is highlighted in Supplementary Figure S1 in terms of the paralog structure. Among the 70 865 isoforms in the hg38 transcriptome reference, 52 046 (73.3%) are separately quantifiable (paralogs of size 1) according to bulk RNA-Seq data; the rest (26.7%) belonged to paralogs of size 2 or more. In contrast, the 3′ bias in the scRNA-Seq data reduces the number of separately quantifiable isoforms to 21 287 (30.0%). Among this last group, 13 753 isoforms belong to singleton genes, so there are still 7534 non-paralog isoforms belonging to multi-isoform genes that are separately quantifiable in the scRNA-Seq data.

Isoform paralogs with high similarity in their sequences result in high similarity in the corresponding columns in the X matrix. Full similarity causes the X matrix to be singular and the β parameter to be non-identifiable; near similarity causes the X matrix to be poorly conditioned and creates estimation problems. The strong 3′ bias in 10× Genomics sequencing also increases the similarity between isoforms in the X matrix. This is illustrated in Supplementary Figure S2a, where the isoforms of the RPL13A gene are not distinguishable at the 3′ end. Ignoring these paralogs would lead to poor estimation (Supplementary Fig. S2b); in contrast, by properly identifying the paralog Scasa is able to yield much more precise quantification (Supplementary Fig. S2c).

The bias also breaks the symmetry in read sharing between isoforms, an important condition presumed by bulk RNA-Seq methods. For example, in Supplementary Table S1a, the binary code 01011 of the eqClass at row 3 conveys that all reads of this eqClass are mapped to three isoforms NM_012423, NR_026712 and NR_073024. However, following the values in row 3, only NR_026712 contributes 22 reads to this eqClass, while both NM_012423 and NR_073024 contribute no reads to this set. The reason is that the 3′ exons of NR_026712 are close to the 5′ end of the other isoforms such that the reads of these latter isoforms cannot map to NR_026712. A bulk RNA-based method that ignores this problem performs poorly (Supplementary Fig. S3a). The problem does not appear in bulk RNA-Seq data, so the method performs much better (Supplementary Fig. S3b and Supplementary Table S2).

3.2 Scasa outperforms other methods in simulated data

We used simulated data generated with Polyester (Frazee *et al.*, 2015) to compare Scasa against Cellranger (Zheng *et al.*, 2017), Kallisto-bustools (Melsted *et al.*, 2021), Alevin (Srivastava *et al.*, 2019), STARsolo (Kaminow *et al.*, 2021), Kallisto (Bray *et al.*, 2016), Salmon (Patro *et al.*, 2017) and Terminus (Sarkar *et al.*, 2020). The first four methods are gene-based quantification methods designed for scRNA-Seq data, while the last three are designed for bulk RNA-Seq data. For the isoform-based quantification methods, a comparative gene-level value was computed as the sum of component isoforms according to the transcriptome reference. In the simulation (see Section 2.6), since the RNA reads of individual cells could be separate and independent from cell barcodes and UMIs, these bulk RNA-Seq methods could be run on the data in single-end read mode to obtain both isoform-level and gene-level expression.

At the isoform level, Scasa was compared against Kallisto, Salmon and Terminus for (i) all isoforms and (ii) non-paralog isoforms, which are isoforms that belong to multi-isoform genes but not to any paralog. Singleton isoforms, which belong to single-isoform genes, were compared at the gene level. At the gene level, Scasa was compared against all of the other methods. Gene-level comparisons were performed for (i) all genes and (ii) singleton genes. The quality of quantification by each method was expressed in terms of the absolute proportion of error (APE) in the estimated abundance versus the true counts (see Sections 2 and 2.5 for details). The methods are compared in two ways: (i) according to the ratio of the APE of the competing method to the APE of Scasa, where an APE ratio >1 indicated that the performance of Scasa was better (the APE ratios are summarized in boxplots); and (ii) according to the proportion of Scasa APE that is less than the APE of the competing methods, a proportion >0.5 indicates that the performance of Scasa is better than the competing method (Fig. 1c). Scasa also outperforms Terminus, a method that allows transcript grouping from the output of Salmon. The distribution of paralog size of Terminus (Supplementary Fig. S1) is similar to the pattern for bulk RNA-Seq data, indicating that Terminus seems to treat the scRNA-Seq data similarly to the bulk RNA-Seq data.

Figure 1b shows the true versus estimated abundances at the isoform level for Scasa, Kallisto and Salmon, clearly demonstrating the high noise levels of the quantification methods developed for the bulk RNA-Seq data. In addition to exhibiting high variability, Kallisto has a bias problem that we cannot explain. Similar plots at the gene level are presented in Supplementary Figure S4a. The boxplots of the APE ratios in Figure 1c show that Scasa performs well overall against the existing methods in all categories of isoform- and gene-level quantification. The most similar performance is observed between Scasa and Salmon for non-paralog isoforms, which are isoforms that belong to multi-isoform genes but not to any paralogs. Among the existing methods that provide gene-level quantification, Alevin and STARsolo perform relatively well. These results are corroborated in Supplementary Table S3, which shows that Scasa produces a higher proportion of lower APE values than any of the other methods.

3.3 Scasa reveals a novel subgroup of CD14 monocytes

The simulation results led us to ask how isoform quantification would improve cell-type identification through scRNA-Seq. To address this question, we used publicly available 10× Chromium CITE-Seq data for bone marrow mononuclear cells (Stuart *et al.*, 2019), as the combined use of RNA and antibodies significantly improved cell-type identification and annotation. Through the isoform-level quantification of the same data, Scasa (Scasa isoform) identified a distinct subgroup of CD14 monocytes (Fig. 2a, Supplementary Fig. S5). This monocyte subset was largely annotated as CD14 monocytes (420/454 cells, 92.5%) and was missed by gene-level quantification using Scasa-gene, Cellranger and Alevin (Supplementary Fig. S4b). DE analysis identified eight isoforms (Fig. 2b) from five genes, TYROBP, HLA-DPA1, RNASE6, FCGRT and LGALS2, as statistically significant DE isoforms between this monocyte subset and the rest of the CD14+ monocytes (Pawitan *et al.*, 2005; FDR < 0.05; Supplementary Table S4).

Interestingly, the top four DE isoforms all came from the transmembrane immune signaling adaptor gene TYROBP (Fig. 2d, Supplementary Fig. S6a). Statistically, these isoforms showed substantially higher significance (FDR $\leq$ 4E−139) than the next most significant isoform paralog from the HLA-DPA1 gene (FDR $=$ 0.0019); see Supplementary Table S4 for detailed results. However, the expression at gene level for TYROBP shows no difference across different quantification methods as compared to its isoform-level quantification (Fig. 2d, Supplementary Fig. S4c). The

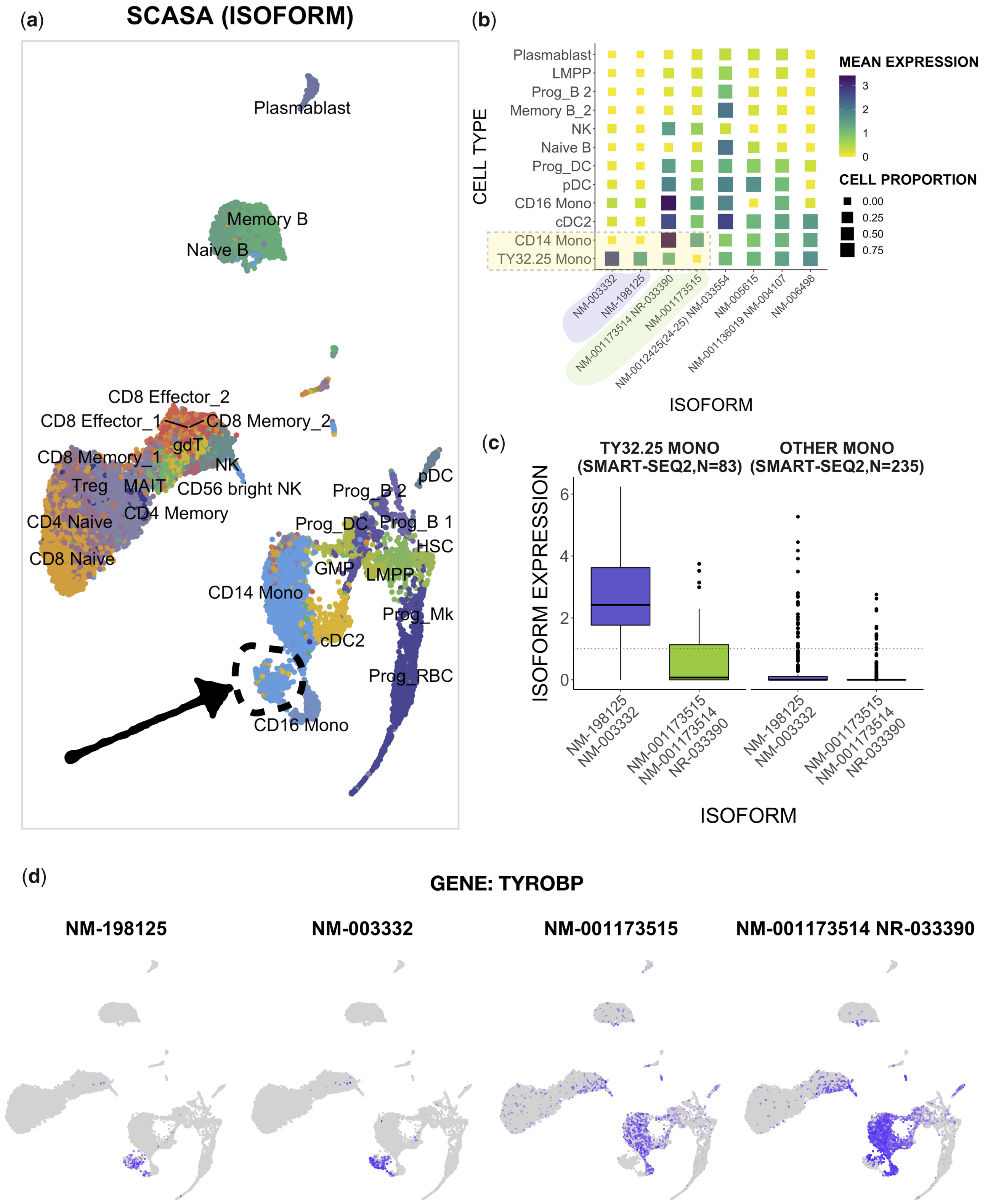

Fig. 2. Isoform-level quantification from Scasa unveil new cell-type cluster and differential expressed isoforms. (**a**) UMAP of isoform quantification from Scasa using a bone marrow dataset and cell annotations queried from the original study 12. Arrow points to a distinct subgroup of CD14 monocytes (which we will call it TY32.25 Mono here) discovered via Scasa isoform quantification. (**b**) Heatmap of median expression of isoforms in each cluster, including TY32.25 Mono. Each cluster (row) is annotated by the most dominant cell types of single cells in that cluster. Mutually exclusive expression pattern of TYROBP isoforms (highlighted in purple and green in the *x* axis labels) are observed in the TY32.25 Mono group. Size of each square is proportional to the percentage of cells expressing the isoform in their corresponding cell-type group. Color gradient represents the mean expression of the isoforms in each cell-type group. For convenience, the clusters with low total expression of the isoforms are excluded from the plot. NM-0012425 (24–25) refers to two isoforms, NM-001242524 and NM-001242525. (**c**) Boxplot of the isoforms of TYROBP gene in the TY32.25 Mono group and the other mono groups from the Smart-Seq2 dataset. Full-length-transcript Smart-Seq2 data from the bone marrow cells of 15 human fetuses 14 is used in this validation. (**d**) Patterns of the differential expressed isoforms of TYROBP gene observed in all cells, with distinct observation in the TY32.25 Mono group as indicated in (a)

significant biological pathways enriched with DE genes are provided in Supplementary Table S5, where the interferon gamma signaling pathway is ranked at the top. We found isoform-specific expression of TYROBP in CD14 monocytes, in which NM_198125 and NM_003332 were highly expressed in the new subset, whereas the majority of CD14 monocytes expressed the paralog isoform NM001173514-NR_033390 (Supplementary Fig. S6b).

To validate these findings, we used independent full-length scRNA-Seq data for human bone marrow (Ranzoni *et al.*, 2021) generated via the Smart-Seq2 protocol[15]. The Smart-Seq2 scRNA-Seq data allowed isoform quantification to be conducted, since full-length transcript sequencing of single cells is performed in this method (Picelli *et al.*, 2014). We used Salmon7 to obtain isoform expression levels, and we performed clustering and cell-type identification (Supplementary Fig. S7). A cluster of 83 monocytes (out of 318 cells) expressing the two dominant TYROBP isoforms, NM_198125 and NM_003332, was identified (Fig. 2c). The results suggested that the monocyte subset identified from 10× 3′ data indeed expressed TYROBP in an isoform-specific manner. We did not, however, observe clear mutually exclusive expression of the two isoform groups, as found in the 10× data (Supplementary Fig. S8), potentially due to biological differences between the two datasets (e.g. adult cells in the CITE-Seq data versus fetal cells in the Smart-Seq2 data).

4 Discussion

Isoforms are the result of alternative splicing, a regulatory process of inclusion and exclusion of exons from the same gene. Even though they have highly similar sequences, the different isoforms of the same gene can have different biological functions. Estimation of isoform expression is still a challenging problem for both bulk RNA-Seq and scRNA-Seq (Westoby *et al.*, 2020). In principle, many isoform quantification software designed for bulk RNA-Seq can be utilized for scRNA-Seq datasets. However, we showed that the bulk RNA-Seq methods did not perform well for the high-throughput scRNA-Seq with a strong 3′ bias such as 10× Genomics. The bias increases substantially the similarities between the read statistics from different isoforms, and breaks the symmetry in read sharing between them, making the quantification highly challenging for the bulk RNA-Seq methods. We have developed Scasa to deal with this challenge and showed that it was able to accurately quantify isoform expression for 10× 3′ scRNA-Seq data and it performed well against competing methods.

Scasa relies on the concepts of transcription clusters and isoform paralogs, which are inherited from XAEM (Deng *et al.*, 2020). As an extension of XAEM, Scasa formalizes and clarifies the issues caused by the 3′-end bias in the droplet-based scRNA-Seq data. The novel contributions include (i) processing the sample data to produce the count matrix Y from highly multiplexed cells and (ii) the *in silico* construction of transcription cluster to take into account the asymmetry issue and the high proportion of isoform paralogs, specifically occurring in the strong 3′ bias scRNA-Seq data. In an application to a 10× Chromium CITE-Seq dataset, we identified an isoform-specific cellular subset that was only detectable in full-length scRNA-Seq data. Further applications of Scasa to various types of biological and clinical data could potentially reveal more cellular subsets and corresponding biomarkers that are not visible at the gene level.

In the simulation study, the initial X matrix in Scasa and the simulated data are based on the same simulator (Polyester). We investigate whether this confers a special advantage to Scasa. To do that, we utilize RNASeqReadSimulator (https://github.com/davidliwei/RNASeqReadSimulator) to generate a new 10× Genomics simulated dataset of 3955 single cells, using the same setting as we did for the simulated data generated from Polyester. The same analysis procedures as before for Scasa, Kallisto, Kallisto-Bustools, Cellranger, Salmon and Alevin are performed on the new simulated dataset. For Scasa, we use the initial design matrix X constructed *in silico* using Polyester. The results reported in Supplementary Figure S10 show that Scasa generally still performs well against the competing methods at both isoform level and gene level; Scasa's performance is comparable to Salmon for non-paralogs, and to Alevin for gene-level estimation. Thus, the use of the same simulator for initialization and data generation does not give a special advantage to Scasa over other methods.

In this study, we have evaluated Scasa for Chromium Single Cell 3′ 10× Genomics, the most commonly used high-throughput scRNA-Seq. However, Scasa is not limited to run only on the droplet-based method. Conceptually the method could be applicable to any high-throughput scRNA-Seq method with either 3′ or 5′ bias. The key step was to build an appropriate initial X matrix that corresponds to the scRNA-Seq protocol; specific instructions are given in the Scasa webpage. Scasa can also run independently as long as the equivalent classes of single cell are provided. Thus, Scasa can be used as a plug-in module for the current tools developed for gene-level quantification such as Alevin and Kallisto-bustools where the steps of quality of mapping, barcode correction and read processing are already well developed.

For its limitation, Scasa cannot estimate all the isoforms available in the annotation reference. This is because isoforms with high similarity in their sequences are statistically unidentifiable from each other. Scasa resolves the problem by combining them into paralogs to improve the estimation accuracy, but compensates for the number of identifiable isoforms. Further characterization of the isoform members of an interesting paralog requires extra information and perhaps experiments, but this is out of the scope of the current study.

Acknowledgement

The authors acknowledge Xuexin Li for naming the Scasa software.

Funding

This work was partially supported by research grants from the Swedish Research Council (VR) and the Swedish Foundation for Strategic Research (SSF). The computations were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC) in Uppsala, which is partially funded by the Swedish Research Council under Grant 2018-05973.

Conflict of Interest: none declared.

References

Bray,N.L. *et al.* (2016) Near-optimal probabilistic RNA-seq quantification. *Nat. Biotechnol.*, **34**, 525–527.

Deng,W. *et al.* (2020) Alternating EM algorithm for a bilinear model in isoform quantification from RNA-seq data. *Bioinformatics*, **36**, 805–812.

Ding,J. *et al.* (2020) Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nat. Biotechnol.*, **38**, 737–746.

Frazee,A.C. *et al.* (2015) Polyester: simulating RNA-seq datasets with differential transcript expression. *Bioinformatics*, **31**, 2778–2784.

Hu,Y. *et al.* (2020) Detecting differential alternative splicing events in scRNA-seq with or without unique molecular identifiers. *PLoS Comput. Biol.*, **16**, e1007925.

Huang,Y. and Sanguinetti,G. (2017) BRIE: transcriptome-wide splicing quantification in single cells. *Genome Biol.*, **18**, 123.

Jassal,B. *et al.* (2020) The reactome pathway knowledgebase. *Nucleic Acids Res.*, **48**, D498–D503.

Kaminow,B. *et al.* (2021) *STARsolo: Accurate, Fast and Versatile Mapping/Quantification of Single-Cell and Single-Nucleus RNA-seq Data. Technical Report.* Cold Spring Harbor Laboratory, Cold Spring Harbor, NY.

Melsted,P. *et al.* (2021) Modular, efficient and constant-memory single-cell RNA-seq preprocessing. *Nat. Biotechnol.*, **39**, 813–816.

Ntranos,V. *et al.* (2016) Fast and accurate single-cell RNA-seq analysis by clustering of transcript-compatibility counts. *Genome Biol.*, **17**, 112.

Ntranos,V. *et al.* (2019) A discriminative learning approach to differential expression analysis for single-cell RNA-seq. *Nat. Methods*, **16**, 163–166.

Patro,R. *et al.* (2014) Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nat. Biotechnol.*, **32**, 462–464.

Patro,R. *et al.* (2017) Salmon provides fast and bias-aware quantification of transcript expression. *Nat. Methods*, **14**, 417–419.

Pawitan,Y. *et al.* (2005) Bias in the estimation of false discovery rate in microarray studies. *Bioinformatics*, **21**, 3865–3872.

Picelli,S. *et al.* (2014) Full-length RNA-seq from single cells using Smart-seq2. *Nat. Protoc.*, **9**, 171–181.

Ranzoni,A.M. *et al.* (2021) Integrative single-cell RNA-seq and ATAC-seq analysis of human developmental hematopoiesis. *Cell Stem Cell*, **28**, 472–487.e7.

Sarkar,H. *et al.* (2020) Terminus enables the discovery of data-driven, robust transcript groups from RNA-seq data. *Bioinformatics*, **36**, i102–i110.

Song,Y. *et al.* (2017) Single-cell alternative splicing analysis with expedition reveals splicing dynamics during neuron differentiation. *Mol. Cell*, **67**, 148–161.e5.

Srivastava,A. *et al.* (2019) Alevin efficiently estimates accurate gene abundances from dscRNA-seq data. *Genome Biol.*, **20**, 65.

Stuart,T. *et al.* (2019) Comprehensive integration of single-cell data. *Cell*, **177**, 1888–1902.e21.

Svensson,V. *et al.* (2018) Exponential scaling of single-cell RNA-seq in the past decade. *Nat. Protoc.*, **13**, 599–604.

Westoby,J. *et al.* (2020) Obstacles to detecting isoforms using full-length scRNA-seq data. *Genome Biol.*, **21**, 74.

Zheng,G.X.Y. *et al.* (2017) Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.*, **8**, 14049.