METHOD ARTICLE

*satuRn*: Scalable analysis of differential transcript usage for bulk and single-cell RNA-sequencing applications [version 1; peer review: 2 approved with reservations]

Jeroen Gilis¹⁻³, Kristoffer Vitting-Seerup⁴⁻⁵, Koen Van den Berge¹⁻³⁻⁶, Lieven Clement¹⁻³

¹Applied Mathematics, Computer science and Statistics, Ghent University, Ghent, 9000, Belgium
²Data Mining and Modeling for Biomedicine, VIB Flemish Institute for Biotechnology, Ghent, 9000, Belgium
³Bioinformatics Institute, Ghent University, Ghent, 9000, Belgium
⁴Department of Biology, Kobenhavns Universitet, Copenhagen, 2200, Denmark
⁵Biotech Research and Innovation Centre (BRIC), Kobenhavns Universitet, Copenhagen, 2200, Denmark
⁶Department of Statistics, University of California, Berkeley, Berkeley, California, USA

Abstract

Alternative splicing produces multiple functional transcripts from a single gene. Dysregulation of splicing is known to be associated with disease and as a hallmark of cancer. Existing tools for differential transcript usage (DTU) analysis either lack in performance, cannot account for complex experimental designs or do not scale to massive scRNA-seq data. We introduce *satuRn*, a fast and flexible quasi-binomial generalized linear modelling framework that is on par with the best performing DTU methods from the bulk RNA-seq realm, while providing good false discovery rate control, addressing complex experimental designs and scaling to scRNA-seq applications.

Keywords

RNA-seq, single-cell transcriptomics, splicing, differential transcript usage, statistical framework, satuRn

This article is included in the Bioconductor gateway.

Open Peer Review

Reviewer Status

Invited Reviewers

version 1

11 May 2021

Report

Report

1. Marek Cmero, Peter MacCallum Cancer Centre, Melbourne, Australia
   The University of Melbourne, Parkville, Australia

2. Alejandro Reyes, Novartis Institutes for BioMedical Research (NIBR), Basel, Switzerland

Any reports and responses or comments on the article can be found at the end of the article.
Corresponding authors: Koen Van den Berge (Koen.VanDenBerge@UGent.be), Lieven Clement (lieven.clement@ugent.be)

Author roles: Gilis J: Formal Analysis, Funding Acquisition, Investigation, Methodology, Software, Validation, Visualization, Writing – Original Draft Preparation; Vitting-Seerup K: Methodology, Validation, Writing – Review & Editing; Van den Berge K: Conceptualization, Funding Acquisition, Methodology, Supervision, Validation, Writing – Review & Editing; Clement L: Conceptualization, Funding Acquisition, Methodology, Supervision, Validation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: Research Foundation Flanders
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2021 Gilis J et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Gilis J, Vitting-Seerup K, Van den Berge K and Clement L. satuRn: Scalable analysis of differential transcript usage for bulk and single-cell RNA-sequencing applications [version 1; peer review: 2 approved with reservations] F1000Research 2021, 10:374 https://doi.org/10.12688/f1000research.51749.1

First published: 11 May 2021, 10:374 https://doi.org/10.12688/f1000research.51749.1
Introduction

Studying differential expression (DE) is one of the key tasks in the downstream analysis of RNA-seq data. Typically, DE analyses identify expression changes on the gene level. However, the widespread adoption of expression quantification through pseudo-alignment, which enables fast and accurate quantification of expression at the transcript level, has effectively paved the way for transcript-level analyses. Here, we specifically address differential transcript usage (DTU) analysis, one type of transcript-level analysis that studies the change in relative usage of transcripts/isoforms within the same gene. DTU analysis holds great potential: previous research has shown that most multi-exon human genes are subject to alternative splicing and can thus produce a variety of functionally different isoforms from the same genomic locus. The dysregulation of this splicing process has been reported extensively as a cause for disease, including several neurological diseases such as frontotemporal dementia, Parkinsonism and spinal muscular atrophy, and is a well-known hallmark of cancer.

In this context, full-length single-cell RNA-Seq (scRNA-seq) technologies such as Smart-Seq2 and Smart-Seq3 hold the promise to further increase the resolution of DTU analysis from bulk RNA-seq data towards the single-cell level, where differences in transcript usage are expected to occur naturally between cell types. However, only a few bespoke DTU methods have been developed for scRNA-seq data and they lack biological interpretation. Indeed, methods specifically developed for scRNA-seq data are either restricted to exon/event level analysis (e.g. pinpointing exons involved in splicing events), or they can only pinpoint DTU genes without unveiling the actual transcripts that are involved. Interestingly, many DTU methods for bulk RNA-seq do provide inference at the transcript level and their performance has already been extensively profiled in benchmark studies. Based on a subset of the simulated RNA-seq dataset from Love et al. (see Methods), we show the performance of six DTU tools: DEXSeq, DoubleExpSeq, DRIMSeq, edgeR diffSplice, limma diffSplice and NBSplice (Figure 1A). DEXSeq and DoubleExpSeq have a higher performance than the other methods. In addition, we observe that most methods, and DRIMSeq in particular, fail to control the false discovery rate (FDR) at its nominal level, which is in line with previous reports.

In order to assess DTU in single-cell applications, however, these bulk RNA-seq DTU tools should scale to the large data volumes generated by full-length scRNA-seq platforms, which can profile the transcriptome of several thousands of cells in a single experiment. In Figure 1B, we evaluate the required computational time in function of the number of sequenced libraries for a two-group DTU analysis on a subset of the scRNA-seq dataset from Chen et al. In spite of its good performance, the popular tool DEXSeq already required more than five hours to analyze two groups of 32 cells and clearly does not scale to large bulk nor scRNA-seq datasets.

In addition, DTU methods should allow for the analysis of datasets with large numbers of (unique) transcripts. The number of transcripts that are typically assessed depends on the coverage of the RNA-seq experiment and the adopted filtering criteria in the data analysis workflow. As the coverage of RNA-seq experiments has increased rapidly over the past few years and can be expected to continue expanding, scalability towards large numbers of transcripts will be essential to enable a transcriptome-wide view on the isoform usage changes. In Figure 1C, we perform a DTU analysis across a range of transcripts in a two-group comparison with 16 cells each, using the dataset from Chen et al. Here, we observed that the DTU tool BANDITS scales particularly poorly to large numbers of transcripts. More specifically, BANDITS did not complete the DTU analysis on the dataset with 7,500 transcripts within 137 hours on our system (see Methods); therefore, larger analyses were omitted. As such, BANDITS had to be omitted from the analyses shown in Figures 1A and 1B. For a performance and scalability evaluation of BANDITS on datasets with an (artificial) lower number of transcripts, we refer to Extended data figures S1 and S3.

Besides scalability, several other issues arise when porting bulk RNA-seq DTU tools towards scRNA-seq applications. Indeed, modelling scRNA-seq data often requires multifactorial designs, for instance when comparing expression levels across multiple cell types between multiple treatment groups. Accounting for multiple covariates, however, is not implemented in BANDITS, NBSplice and DoubleExpSeq, jeopardizing their utility for (sc-)RNA-seq DTU analysis. Another issue arises with the large numbers of zero counts in scRNA-seq data, which seems to be particularly problematic for NBSplice that fails to converge if the gene-level count of any of the samples or cells is zero. As such, NBSplice could not be evaluated in Figures 1B and 1C.

Altogether, many of the existing DTU analysis tools are not well suited to analyze large bulk RNA-seq and full-length scRNA-seq datasets, leaving the great potential of these data largely unexploited. In light of these shortcomings we developed satuRN, which is an acronym for Scalable Analysis of differential Transcript Usage for RNA-seq data, a novel method for DTU analysis that (i) is highly performant, (ii) provides a good control of the false discovery rate (FDR) (iii) scales seamlessly to the large data volumes of contemporary (sc-)RNA-seq datasets, (iv) allows for modelling complex experimental designs, (v) can deal with realistic proportions of zero counts and (vi) provides direct inference on
the biologically relevant transcript level. In brief, satuRn adopts a quasi-binomial (QB) generalized linear model (GLM) framework. satuRn provides direct inference on DTU by modelling the relative usage of a transcript, in comparison to other transcripts from the same gene, between groups of interest. To stabilize the estimation of the overdispersion parameter of the QB model, we borrow strength across transcripts by building upon the empirical Bayes methodology as introduced by Smyth et al.\textsuperscript{23} In order to control the number of false positive findings, an empirical null distribution is used to obtain the p-values, which are corrected for multiple testing with the FDR method of Benjamini and Hochberg.\textsuperscript{31} Our method is implemented in an R package available at https://github.com/statOmics/satuRn and will be submitted to the Bioconductor project.

Figure 1. Performance and scalability evaluation of six differential transcript usage (DTU) methods. A: Performance evaluation on the simulated bulk RNA-Seq dataset from Love et al.\textsuperscript{18} Each curve displays the performance of each method by evaluating the sensitivity (true positive rate, TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. DEXSeq and DoubleExpSeq clearly have the highest performances. Note that most methods, and DRIMSeq in particular, fail to control the FDR at its nominal level. B: Scalability with respect to the number of cells in a scRNA-Seq dataset. While all other methods scale linearly with an increasing number of cells, DEXSeq scales quadratically. As such, DEXSeq cannot be used for the analysis of large bulk and scRNA-Seq datasets. For all sample sizes, the number of transcripts in the datasets were set at 30,000. Note that NBSplice needed to be omitted from this analysis as it fails to converge on datasets with a large proportion of zero counts (see below). C: Scalability with respect to the number of transcripts in a scRNA-Seq dataset. While all other methods scale linearly with an increasing number of cells, BANDITS scales quadratically. Moreover, BANDITS failed to run on our system for datasets with 7,500 transcripts or more. As such, it had to be omitted from panels A and B. A performance and scalability evaluation of BANDITS on datasets with an (artificial) lower number of transcripts is provided as Extended data figures S1 and S3.\textsuperscript{25}
Methods

SatuRn model

As input, satuRn requires a matrix of transcript-level expression counts, which may be obtained either through pseudo-alignment using kallisto\(^1\) or salmon\(^2\) or by classical alignment-based tools followed by transcript-level quantification (e.g. STAR\(^3,33\) and RSEM\(^34\)). Let \(Y_{gi}\) denote the observed expression value for a given transcript \(t = 1, \ldots, T_g\) of gene \(g = 1, \ldots, G\) in cell or sample \(i = 1, \ldots, n\). The total expression of gene \(g\) in sample \(i\) can then be expressed as

\[
Y_{gi} = \sum_{t \in T_g} Y_{gti},
\]

i.e. by taking the sum of expression values for all \(T_g\) transcripts belonging to gene \(g\) in sample \(i\). The usage of transcript \(t\) in sample or cell \(i\) can then be estimated as

\[
U_{gti} = \frac{Y_{gti}}{Y_{gi}}.
\]

Next, we adopt a quasi-binomial (QB) generalized linear modelling (GLM) strategy to model DTU. As opposed to canonical maximum likelihood models, this quasi-likelihood modelling strategy only requires the specification of the first two moments of the response distribution, i.e. the mean and the variance. We define the mean of the QB model as

\[
\left\{
\begin{array}{l}
E[U_{gti}|X_i, Y_{gi}] = \pi_{gti} \\
\log\left(\frac{\pi_{gti}}{1 - \pi_{gti}}\right) = \eta_{gti} \\
\eta_{gti} = X_i^T \beta_{gt}
\end{array}
\right.
\]

(3)

In this notation, \(\pi_{gti}\) is the expected probability of observing transcript \(t\) within the pool of transcripts \(1, \ldots, T_g\) belonging to gene \(g\) in sample \(i\) and, as such, corresponds to its expected usage for that sample. We model \(\pi_{gti}\) using a logit link function, where \(\beta_{gt}\) is a \(p \times 1\) column vector of regression parameters modelling the association between the average usage and the covariates for transcript \(t\). Finally, \(X_i\) is a row in the \(n \times p\) design matrix \(X\) that corresponds with the covariate pattern of sample \(i\), with \(p\) the number of parameters of the mean model, i.e. the length of vector \(\beta_{gt}\).

The variance of the QB model can be described as

\[
\text{Var}[U_{gti}|X_i, Y_{gi}] = \frac{\pi_{gti}(1 - \pi_{gti})}{Y_{gi}} \phi_{gt}
\]

(4)

with \(Y_{gi}, \pi_{gti}(1 - \pi_{gti})\) the canonical variance of the binomial distribution and \(\pi_{gti}\) a transcript-specific overdispersion parameter to describe additional variance in the data with respect to the binomial variance. We adopt the empirical Bayes procedure from Smyth et al.\(^{23}\), as implemented in the \textit{squeezVar}\ function of the \textit{limma} Bioconductor R package, to stabilize the estimates of \(\pi_{gti}\) by borrowing information across transcripts, which is adopted in the default edgeR quasi-likelihood workflow for bulk RNA-seq data.\(^{22,35}\) Note that stabilizing the dispersion estimation is particularly useful in datasets with a small sample size.

Taken together, the quasi-binomial thus allows us to model the log-odds of drawing a particular transcript \(t\) from the pool of transcripts in the corresponding gene \(g\) across samples. The intercept also has an interpretation of a log-odds and the remaining mean model parameters are log-odds ratios, which may thus be interpreted in terms of differential transcript usage. We adopt t-tests that are computed based on the log-odds ratio estimates of the QB model and the posterior variance, as obtained from the empirical Bayes procedure. P-values are computed assuming a t-distribution under the null hypothesis with posterior degrees of freedom calculated as the sum of the residual degrees of freedom and the prior degrees of freedom from the empirical Bayes procedure.

For bulk analyses, the implementation of satuRn as described above provides a high performance and a good control of the FDR. However, for single-cell datasets we observed that our inference is too liberal (see Extended data\(^{25}\) figure S10), which could suggest that the theoretical null, the t-distribution, is no longer valid. Indeed, in large-scale inference settings, failure of the theoretical null distribution is often observed. Efron\(^{36}\) (Chapter 6) describes four reasons why the theoretical null distribution may fail; failed mathematical assumptions, correlation across features (transcript expression), correlation across subjects (samples or cells), and unobserved confounders in observational studies. To avoid these issues, Efron proposes to exploit the massive parallel data structure of omics datasets to empirically estimate the null distribution of the
To this end, Efron converts the test statistic to z-scores, which should follow a standard normal distribution under the theoretical null, and then proposes to approximate the empirical null distribution with a normal distribution with unknown mean ($\mu^*$) and standard deviation ($\sigma^*$), which can be estimated by maximum likelihood on a subset of the test statistics near zero.

As such, we first convert the two-sided p-values to z-scores according to

$$z_{gt} = \Phi^{-1}\left(\frac{p_{gt}}{2}\right) \cdot \text{sign}(S),$$

with $\Phi$ the cumulative distribution function for the standard normal distribution, $p_{gt}$ the original two-sided p-value indicating the statistical significance of differential usage of transcript $t$ from gene $g$ between the conditions of interest, $\text{sign}(S)$ the sign of the t-test statistic $S$ and $z_{gt}$ the resulting z-score. Next, we adopt the maximum likelihood procedure, implemented in the `locfdr` function of the `locfdr` R package from CRAN, to estimate the mean $\mu^*$ and standard deviation $\sigma^*$ of the empirical null distribution. Based on these estimates, we recompute the z-scores and corresponding p-values as follows

$$z^*_{gt} = \frac{z_{gt} - \mu^*}{\sigma^*},$$

$$p^*_{gt} = 2\Phi\left(-\text{abs}\left(z^*_{gt}\right)\right).$$

Finally, the resulting (empirical) p-values are corrected for multiple testing with the FDR method of Benjamini and Hochberg. As opposed to the original p-values that were calculated based on the theoretical null distribution for the t-statistics, we found that this procedure allows for a better FDR control in single-cell applications.

**DTU tools literature**

Table 1 provides a brief description of the DTU methods that were included in the performance benchmarks of this paper. For more details, we refer to the Extended data and the respective original publications. Note that all methods were run in R 3.6.1 using default settings.

**Filtering**

We adopted two different strategies for filtering transcripts in each of the RNA-seq datasets in the performance benchmarks.

The first filtering strategy uses the `filterByExpr` function implemented in edgeR. This filtering strategy only retains transcripts that have at least an expression level of `min.count` counts-per-million (CPM, calculated as the number of read counts divided by the total number of reads in the dataset and multiplied by one million) in at least $n$ samples or cells. In addition, the sum of the CPM of the transcript across all cells or samples must be at least `min.total.count`. For the bulk RNA-seq datasets, we use the default settings (`min.count = 10`, $n = \min(10, 0.7*\text{sample size of the smallest group in the comparison})$ and `min.total.count = 10`). For the scRNA-seq datasets, the settings are adjusted to; `min.count = 1` (as requiring a transcript to be expressed in all single-cells is a stringent criterium), $n = 0.5*\text{sample size of the smallest group in the comparison}$ and `min.total.count = 0`. In addition, if only one transcript of a gene passes this filtering criterion, it is omitted from the analysis, as DTU analysis is meaningless when only one transcript is retained. As such, we specifically set the parameters to generate a very lenient filtering criterium.

The second filtering strategy uses the `dmFilter` function implemented in DRIMSeq. This filter is more stringent and specifically designed for DTU analysis. The filtering process can be thought of as proceeding in three steps. Let $n_i$ be the number of samples or cells in the smallest group. The first step requires the transcripts to have a count of at least 10 in at least $n_i$ samples. The second filtering step requires the transcript to make up at least 10% of the total count of its corresponding gene in at least $n_i$ samples. The third filtering step removes all transcripts for which the corresponding gene has a count below 10 in any of the samples or cells in the dataset. Again, if only one transcript of a gene passes this filtering criterion, it is omitted from the analysis.

**Bulk simulation study**

To evaluate the performance of the different DTU analysis methods, we first adopt three simulated bulk RNA-seq datasets from previous publications: the simulated dataset from Love et al. (dataset 1) and both the *Drosophila melanogaster*
Notably, there is a subtle difference in how DTU is introduced between the two simulation frameworks. For dataset 1, the origin of DTU is twofold: On the one hand, DTU was specifically introduced by swapping the transcript-per-million (TPM) abundances between two expressed isoforms. On the other hand, DTU was also obtained as a consequence of introducing DTE, where a single expressed isoform was induced to be differentially expressed at a certain log fold change, which leads to DTU if this transcript belongs to a gene expressing multiple isoforms. For datasets 2 and 3, there is only one source of DTU. The number of differentially used transcripts within a gene was sampled ranging from a minimum of two up to a random number drawn from a binomial distribution with size equal to the number of transcripts and success probability 1/3. DTU was introduced by swapping the TPM abundances between the differentially used transcripts. As such, the latter framework allows for differential usage of multiple transcripts of the same gene, which is not possible in the former framework.

Table 1. Brief description of different DTU tools that were included in our performance benchmarks. Columns 1 to 5 respectively display the name of each method, the package version used, a brief description of each method, the test statistic used for inference on differential transcript usage, and whether the method can handle complex designs, e.g., to incorporate additional covariates such as batch effects. All packages are available from Bioconductor 3.10. Acronyms: NB: negative binomial, GLM: generalized linear model, LRT: likelihood ratio test, DB: double binomial, DM: Dirichlet-multinomial.

<table>
<thead>
<tr>
<th>Method</th>
<th>Version</th>
<th>Brief description of the modelling procedure</th>
<th>Test</th>
<th>Complex designs</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEXSeq</td>
<td>1.32.0</td>
<td>First a matrix ( C = [C_{gti}] ) is calculated, which defines how many reads ( C_{gti} ) map to any of the other transcripts of the same gene ( g ) as respective transcript ( t ) in cell ( i ). Next, matrix ( C ) and the original expression matrix ( Y ) are concatenated as ( Y' = [Y \ C] ). Then, a NB GLM is fitted to each transcript of ( Y' ). The design matrix ( X ) of the GLM defines a covariate pattern with (i) sample-level intercepts that account for the fact that ( Y_{gti} ) and ( C_{gti} ) originate from the same sample ( i ), and (ii) other covariates associated with the design of the experiment. DTU is parameterized as an interaction effect that evaluates if the log fold change between transcript ( t ) and all other transcripts in its corresponding gene differs between conditions of interest.</td>
<td>LRT</td>
<td>Yes</td>
</tr>
<tr>
<td>Double-ExpSeq</td>
<td>1.1</td>
<td>Assumes a DB model for each transcript that models the log-odds of drawing a particular transcript ( t ) from the pool of transcripts in the corresponding gene ( g ) across samples. The intercept has an interpretation of a log-odds and the remaining mean model parameter(s) are log-odds ratios, which may thus be interpreted in terms of DTU.</td>
<td>LRT</td>
<td>No</td>
</tr>
<tr>
<td>DRIMSeq</td>
<td>1.14.0</td>
<td>Assumes that the transcript-level expression counts follow a DM distribution. The quantity of interest is the change in proportion of each transcript within a gene between groups of samples or cells.</td>
<td>LRT</td>
<td>Yes</td>
</tr>
<tr>
<td>Limma diffsplice</td>
<td>3.42.2</td>
<td>Assumes a linear model to model the log-transformed transcript-level counts, using weights to account for heteroskedasticity. Assesses DTU by comparing the log-fold change in expression of transcript ( t ) within gene ( g ) with the average log-fold change of all other transcripts of gene ( g ).</td>
<td>T-test</td>
<td>Yes</td>
</tr>
<tr>
<td>edgeR diffsplice</td>
<td>3.28.1</td>
<td>Assumes a NB GLM for each transcript and tests for DTU by comparing the obtained log-fold changes for each transcript within a gene with the log-fold change of the entire gene.</td>
<td>LRT</td>
<td>Yes</td>
</tr>
<tr>
<td>NBSplice</td>
<td>1.4.0</td>
<td>Assumes a NB GLM for each gene. DTU between groups of interest can be tested using a LRT, where the full model contains an isoform-condition interaction term that is omitted in the null model.</td>
<td>LRT</td>
<td>Yes</td>
</tr>
<tr>
<td>BANDITS</td>
<td>1.2.3</td>
<td>Adopts a Bayesian DM hierarchical model. Equivalence class counts are used as additional input to account for the quantification uncertainty arising from reads mapping to multiple transcripts. BANDITS tests the mean relative usage of each transcript within its corresponding gene across conditions.</td>
<td>Wald</td>
<td>No</td>
</tr>
</tbody>
</table>

(dataset 2) and Homo sapiens (dataset 3) simulation studies from Van den Berge et al. [40]. All three datasets were generated based on parameter values obtained from real RNA-seq samples, to mimic real RNA-seq data as close as possible.
not possible with the framework used for generating dataset 1. Additionally, dataset 1 uses salmon\textsuperscript{7} (version 1.1.0) for estimating transcript-level abundances, whereas datasets 2 and 3 were quantified with kallisto\textsuperscript{1} (version 0.46.0).

**Real bulk study**

We evaluate the performance of the different DTU methods on real bulk RNA-seq data, by subsampling a homogeneous set of samples from the large bulk RNA-seq dataset available from the Genotype-Tissue Expression (GTEx) consortium\textsuperscript{41} release version 8. Nine datasets were generated non-parametrically. More specifically, we first selected samples from adrenal gland tissue that were extracted with the RNA extraction method “RNA Extraction from Paxgene-derived Lysate Plate Based”. From the remaining samples we subsampled nine datasets, comprising three repeats for each of three sample sizes: 5 versus 5, 20 versus 20 and 50 versus 50 samples. Next, DTU is artificially introduced with the swapping strategy that is described in the bulk simulation study paragraph of the Methods section of this paper. The GTEx data was quantified with RSEM\textsuperscript{34} version 1.3.0.

**Real single-cell study**

We evaluate the performance of the different DTU methods on real scRNA-seq datasets. These scRNA-seq datasets were generated non-parametrically by subsampling a homogeneous set of cells from three real scRNA-seq datasets,\textsuperscript{26,29,42} after which DTU is artificially introduced by the swapping strategy that is described in the bulk simulation study paragraph of the Methods section of this paper.

For the dataset of Chen \textit{et al.}\textsuperscript{29} which was used to construct Figures 4 and the Extended data figure S7,\textsuperscript{25} we selected a homogeneous population of cells by considering only the EpiStem cells of female mice, resulting in a dataset of 120 cells. From this homogeneous population of cells, we then subsampled six datasets, comprising three repeats for each of two sample sizes: 20 versus 20 and 50 versus 50 cells. Next, DTU was artificially introduced with the swapping strategy that is described in the bulk simulation study paragraph of the Methods section of this paper. Finally, we adopted either edgeR or DRIMSeq for filtering.

The other two scRNA-seq datasets were generated analogously. For the dataset of Tasic \textit{et al.}\textsuperscript{42} which was used to construct Figure S8 (Extended data\textsuperscript{25}), we selected a homogeneous population of cells by considering only the Lamp5 cells in the anterior lateral motor cortex of mice without any eye conditions, resulting in a dataset of 897 cells. After introducing DTU, we randomly subsampled 20, 75 or 200 cells from each group. For the dataset of Darmanis \textit{et al.},\textsuperscript{26} which was used to construct Extended data figure S9, we selected the immune cells that clustered together in tSNE cluster 8 of the original publication, resulting in a dataset of 248 cells. After introducing DTU, we randomly subsampled 20, 50 or 100 cells from each group.

**Case study DGE analysis**

We perform a DGE analysis on a subset of the Tasic single-cell dataset,\textsuperscript{42} i.e. between different cell types originating from the ALM and VISp regions of the glutamatergic L5 IT subclass. We use the quasi-likelihood method of edgeR\textsuperscript{43} to model the gene expression profiles and additionally adopt the edgeR \textit{glmTreat} function to test differential expression against a log2-fold change threshold (log2-fold change = 1). Statistical significance was evaluated at the 5% FDR level.

**Performance assessment**

We assess the performance of different DTU methods on a bulk simulation dataset with scatterplots of the true positive rate (TPR) versus the false discovery rate (FDR), according to the following definitions:

$$TPR = \frac{TP}{TP + FN}$$

$$FDP = \frac{FP}{FP + TP}$$

$$FDR = E[FDP]$$

where FN, FP and TP denote the numbers of false negatives, false positives and true positives, respectively. The FDR-TPR curves are constructed using the Bioconductor R package ICOBRA version 1.14.0.\textsuperscript{44}

**Scalability benchmark**

The scalability benchmark was run on subsets of the Chen scRNA-seq dataset,\textsuperscript{29} which contains 617 cells in total. For the scalability benchmark with respect to the number of cells in the dataset, we randomly subsample a certain number of cells.
different strategies as implemented by edgeR and DRIMSeq, which correspond to a lenient and more stringent approach to the read generation and the simulation of DTU signal (see Methods). In terms of transcript filtering, we adopt two approaches: some methodological differences between the simulation framework of dataset 1 and that of datasets 2 and 3 with respect to the read generation and the simulation of DTU signal (see Methods). We further find the performance of satuRN is on par with the best performing tools from the literature, DEXSeq and DoubleExpSeq. In addition, both satuRN and DoubleExpSeq provide a stringent control of the FDR, while DEXSeq and DRIMSeq are often too liberal, as reported previously.

While simulation studies are common for evaluating the performance of DE analysis methods, there is currently no consensus on the simulation strategy that best mimics real (sc)RNA-seq data. In addition, simulation frameworks generally provide raw count estimates as input data, except for Dataset 1 from Love et al., which both contain two groups of five samples each. In brief, all datasets were constructed by generating sequencing reads based on parameters that are estimated from real bulk RNA-seq data. DTU between groups of samples was artificially introduced in the data, prior to the quantification of expression using either Salmon (dataset 1) or kallisto (dataset 2 and 3). Notably, there are some methodological differences between the simulation framework of dataset 1 and that of datasets 2 and 3 with respect to the read generation and the simulation of DTU signal (see Methods). In terms of transcript filtering, we adopt two different strategies as implemented by edgeR and DRIMSeq, which correspond to a lenient and more stringent filtering, respectively (see Methods).

The result of the performance evaluation of satuRN with respect to other DTU methods on the three simulated bulk datasets is displayed in Figure 2A. Figures 2B and 2C display the performance on datasets 2 and 3 after edgeR filtering, respectively (see Methods). In all three datasets, satuRN outperforms NBSplice, edgeR diffsplice and limma diffsplice. Intriguingly, the performance of DRIMSeq varies strongly between the three datasets. This discrepancy may be explained by the different strategies for generating reads and introducing DTU between dataset 1 and dataset 2. Notably, there are some methodological differences between the simulation framework of dataset 1 and that of datasets 2 and 3 with respect to the read generation and the simulation of DTU signal (see Methods). In terms of transcript filtering, we adopt two different strategies as implemented by edgeR and DRIMSeq, which correspond to a lenient and more stringent filtering, respectively (see Methods).

We also evaluated the effects of sample size and different filtering criteria on the performance of the different DTU methods (see Extended data figures S2, S3, S4 and S5). Neither sample size nor filtering criterion had a profound impact on the ranking of the performances of the different DTU methods; satuRN, DEXSeq and DoubleExpSeq remain the best performing methods overall. In addition, we studied the impact of using either raw count estimates or normalized abundance estimates (scaledTPM, see Methods) as input data for the DTU algorithms. We observed a slightly higher performance in all datasets when providing raw abundance estimates, except for Dataset 1 from Love et al. All performance evaluations in the body of this publication therefore were generated with raw count estimates as input data, except for Figure 2, panel A. For a full overview on the effects of sample size, filtering criteria and data input type, we refer to Figures S2 and S9 of the Extended data.
typically generate data according to parametric assumptions on the data-generating mechanism, thus potentially favoring DE methods that adopt similar distributional assumptions in their statistical model. An alternative procedure is to non-parametrically modify a real dataset. Here, we obtained different subsamples from the large bulk RNA-seq dataset available from the Genotype-Tissue Expression (GTEx) consortium, generating nine datasets in total, i.e. three repeats for each of three sample sizes; 5 versus 5, 20 versus 20 and 50 versus 50 samples. We then artificially introduced DTU in these data by swapping transcript usages between groups of samples (see Methods for details). Again, we adopt two different filtering strategies as implemented by edgeR and DRIMSeq, for all datasets. In addition, our method consistently controls the FDR close to its imposed nominal FDR threshold.

The results of the performance evaluation of satuRn on the real bulk datasets upon edgeR filtering is displayed in Figure 3. In agreement with the results obtained from the simulated bulk RNA-seq study, we observe that the performance of satuRn is on par with the best tools from the literature, DEXSeq and DoubleExpSeq. In this evaluation, DRIMSeq performs poorly, in contrast to the performance evaluation on the simulated bulk RNA-seq dataset 1. Note that DEXSeq, DRIMSeq and NBSplice were omitted from the

---

**Figure 2. Performance evaluation of satuRn on three simulated bulk RNA-Seq datasets.** Each curve visualizes the performance of each method by displaying the sensitivity of the method (true positive rate, TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. The performance of satuRn is on par with the best tools from the literature, DEXSeq and DoubleExpSeq, for all datasets. In addition, our method consistently controls the FDR close to its imposed nominal FDR threshold.
analysis of the largest dataset (50 versus 50 samples), as these methods do not scale to such large datasets (Figure 1).

Adopting the DRIMSeq-based filtering did not have a qualitative impact on the performance (Extended data figure S6).

Performance on real single-cell data

Finally, we evaluate the performance of satuRn on single-cell RNA-seq data. As with the real bulk analysis, the single-cell datasets were generated by subsetting from three different real scRNA-seq datasets (see Methods). Again, we subsampled three repeats of different sample sizes, artificially introduced DTU with the swapping strategy and applied either the edgeR- or DRIMSeq-based filtering criterium (see Methods for details).

By subsampling the Chen et al. dataset, we generated three repeats of two sample sizes, i.e. 20 versus 20 and 50 versus 50 cells. The results of the performance evaluation of satuRn on this dataset upon edgeR filtering is displayed in Figure 4. The performance of satuRn is slightly better than that of the best tool from the literature, DoubleExpSeq. As compared to the evaluations on bulk data, we observe a performance drop for DEXSeq relative to satuRn and DoubleExpSeq. This, in combination with its poor scalability (Figure 1), greatly compromises the use of DEXSeq for the analysis of scRNA-seq data. satuRn again provides a stringent control of the FDR, while the inference of DoubleExpSeq is too liberal, again becoming more problematic for larger sample sizes. Adopting the DRIMSeq filter did not have a qualitative impact on the performances (Extended data figure S7). The results of the performance evaluations on the other two scRNA-seq datasets are in strong agreement with the results displayed here, with satuRn performing at least on par with DoubleExpSeq and satuRn additionally controlling the FDR around the nominal level (Extended data figures S8 and S9).

Notably, we found that the theoretical null distribution of the test statistics from satuRn failed to provide good FDR control in single-cell analyses (Extended data figure S10). To obtain proper p-values with satuRn in single-cell applications, we therefore estimate the null distribution of the test statistic empirically (see Methods, satuRn paragraph). Note, that the use of the empirical null distribution in our bulk RNA-seq benchmarks does not affect the results because no deviations of the theoretical null distribution occur. However, the empirical null resulted in much improved FDR control in scRNA-seq datasets (Extended data figure S10). We therefore adopt the empirical null estimation as the default setting in satuRn. As such, all satuRn results in this publication are relying on the empirical null strategy. As a final remark, we likewise attempted to improve the FDR control of DoubleExpSeq. However, in all analyses with DoubleExpSeq we observed a large spike of p-values equal to 1, which poses a problem when estimating the empirical null distribution (Extended data figure S11). Therefore, this strategy could not be used to improve the FDR control of DoubleExpSeq.
We performed a computational benchmark of satuRn to investigate its scalability with respect to the number of samples/cells and the number of transcripts in an RNA-seq dataset. All scalability benchmarks were run on a single core of a Linux machine with an Intel(R) Xeon(R) CPU E5-2420 v2 (2.20GHz, Speed: 2200 MHz) processor and 30GB RAM. The results are displayed in Figure 5. Figure 5A displays the scalability with respect to the number of cells in the dataset, while keeping the number of transcripts in the dataset fixed at 30,000. From the left panel, it is clear that DRIMSeq and especially DEXSeq scale very poorly with the number of cells in the dataset, which was already shown in Figure 1B. In the right panel, we focus on the four remaining methods. satuRn scales linearly with increasing numbers of cells, with a slope comparable to limma diffsplice. As such, satuRn is able to perform a DTU analysis on a dataset with two groups of 256 cells each and 30,000 transcripts in less than three minutes. Note that BANDITS30 was not included in this benchmark, as it does not scale to datasets with this many transcripts. For a performance and scalability evaluation of BANDITS on datasets with a lower number of transcripts, we refer to Extended data figure S1.25 NBSplice was also omitted as it fails to converge on datasets with a large proportion of zero counts.

Figure 5B shows the scalability with respect to the number of transcripts in the dataset, while keeping the number of cells in the dataset fixed to two groups of 16 cells. As shown in Figure 1C, BANDITS, DEXSeq and DRIMSeq scale poorly to datasets with many transcripts. From the right panel, satuRn scales linearly with increasing numbers of transcripts, albeit with a steeper slope than edgeR diffsplice, DoubleExpSeq and limma diffsplice. Note, that the scalability of DTU analyses can be improved through parallelization, if this is allowed by the underlying algorithm. Parallel execution is implemented in satuRn and in all methods from the literature that were discussed in this manuscript, except for DoubleExpSeq and NBSplice.

Altogether, we find that while several methods for DTU analysis exist, none are optimally suited for analyzing single-cell datasets. DRIMSeq, NBSplice, edgeR diffsplice and limma diffsplice have a much lower overall performance in our benchmarks. DEXSeq does not scale to large datasets. Finally, DoubleExpSeq does not support experimental designs that require an analysis with multiple additive effects, e.g. randomized complete block designs and designs where batch-effect
correction is required, which are essential for many practical scRNA-Seq analysis settings. In addition, it fails to control the FDR at the desired level, especially with increasing sample sizes.

Case study
We use satuRn to perform a DTU analysis on a subset of the single-cell (SMART-seq2) RNA-seq dataset from Tasic et al. In addition, we analyze the same dataset with DoubleExpSeq and limma diffsplice, which are the only other DTU methods that scale to large scRNA-seq datasets and have a reasonable performance in our benchmarks. In the original

Figure 5. Scalability evaluation of satuRn on scRNA-Seq data. A: Runtime with respect to the number of cells in a scRNA-Seq dataset. Left panel: it is clear that DRIMSeq and especially DEXSeq scale very poorly with the number of cells in the dataset. Right panel: Detailed plot of the remaining methods. satuRn scales linearly with increasing numbers of cells, with a slope that is comparable to that of limma diffsplice. As such, satuRn is able to perform a DTU analysis on a dataset with two groups of 256 cells each and 30,000 transcripts in less than three minutes. For all sample sizes, the number of transcripts in the datasets were set at 30,000. Note that NBSplice was not included in this analysis as it fails to converge on datasets with a large proportion of zero counts. B: Runtime with respect to the number of transcripts in a scRNA-Seq dataset. Left panel: DEXSeq, DRIMSeq and especially BANDITS scale poorly to the number of transcripts in the dataset. Right panel: Detailed plot of the remaining methods. satuRn scales linearly with increasing numbers of transcripts, but with a steeper slope than edgeR diffsplice, DoubleExpSeq and limma diffsplice. The number of cells in the dataset was set fixed to two groups of 16 cells. All scalability benchmarks were run on a single core.
publication, the authors studied differential gene expression between cell types originating from two areas at distant poles of the mouse neocortex; the primary visual cortical area (VISp), which processes sensory information with millisecond timescale dynamics \(^{46-48}\) and the anterior lateral motor cortex (ALM), which displays slower dynamics related to short-term memory, deliberation, decision-making and planning. \(^{49,50}\) Based on marker genes, Tasic et al.\(^ {42}\) assigned all of the 23,822 cells from the scRNA-seq dataset to one of three cell classes: glutamatergic (excitatory) neurons, GABAergic (inhibitory) neurons or non-neuronal cells. The authors then further classified the neuronal cells into several subclasses based on their dominant layer of dissection and projection patterns (through a retrograde labelling experiment). Finally, these subclasses are further classified into cell types based on the expression of specific marker genes.

**DGE analysis with edgeR**

In their original DGE analysis, Tasic et al.\(^ {42}\) obtained the largest number of differentially expressed genes between the cell types originating from the ALM and VISp regions of the glutamatergic L5 IT subclass (2,739 cells in total), where L5 refers to layer-of-dissection 5 and IT refers to the intratelencephalic projection type. Here, we first perform a DGE analysis with an edgeR-based workflow (see Methods) on the same comparisons between L5 IT cell types that were assessed by Tasic et al. Table 2 shows the number of differentially expressed genes between the groups of interest in column 4.

**DTU analysis with satuRn**

Next, we perform a DTU analysis for the same cell types using satuRn. In column 5 of Table 2, we display the number of differentially used transcripts for each comparison. We also show the number of unique genes in which we find evidence for changes in usage of at least one transcript (column 6). While the number of differentially used transcripts is lower than the number of differentially expressed genes in each of the contrasts, we did identify differentially used transcripts in all contrasts of interest. Most interestingly, we observe that the overlap between the differentially expressed genes and the genes in which we found evidence for DTU is very limited (Table 2, column 7). This shows that the information obtained from our DTU analyses are orthogonal to the results from the canonical DGE analyses, which has been reported previously for simulated bulk data.\(^ {18}\)

**Gene set enrichment analysis**

We perform a gene set enrichment analysis (GSEA\(^ {51}\)) on the three comparisons with most DE genes and most genes with evidence for DTU (comparisons 5, 6 and 7). Similar gene ontology categories are returned for the set of DGE genes and the set of DTU genes, with many of the enriched processes being biologically relevant in the context of this case study. Enriched gene sets include the gene ontology classes, synapse, neuron projection, synaptic signaling and cell projection organization. This shows that the complementary information brought by the DTU analysis is indeed biologically relevant. For an extensive overview of the GSEA of the set of DGE genes and genes with evidence of DTU in comparisons 5, 6 and 7, we refer to the Extended data.\(^ {25}\)

**satuRn identifies biologically relevant DTU transcripts**

To display the utility of satuRn for identifying and visualizing DTU transcripts in scRNA-seq datasets, we focus on comparison #6 of the DTU analysis and discuss the gene P2X Purinoceptor 4 or P2rx4 (Ensembl ID ENSMUSG00000029470), alsb

---

**Table 2. Number of differentially expressed genes and differentially used transcripts in eight comparisons between cell types.** The first three columns indicate the comparisons between ALM (column 2) and VISp (column 3) cell types, respectively. Column 4 indicates the number of differentially expressed genes as identified with an edgeR analysis. Column 5 displays the number of transcripts that satuRn flagged as differentially used. Column 6 shows the number of unique genes in which satuRn finds evidence of differential usage of at least one transcript. Column 7 displays the absolute number of genes that overlap between columns 4 and 6.

<table>
<thead>
<tr>
<th>Comparison</th>
<th>Cell type 1 (ALM)</th>
<th>Cell type 2 (VISp)</th>
<th>DGE</th>
<th>DTU Tx</th>
<th>DTU Gene</th>
<th>Overlap</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cpa6 Gpr88</td>
<td>Batf3</td>
<td>203</td>
<td>24</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Cbln4 Fezf2</td>
<td>Col27a1</td>
<td>281</td>
<td>92</td>
<td>53</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Cpa6 Gpr88</td>
<td>Col6a1 Fezf2</td>
<td>154</td>
<td>7</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Gkn1 Pcdh19</td>
<td>Col6a1 Fezf2</td>
<td>231</td>
<td>33</td>
<td>22</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Lypd1 Gpr88</td>
<td>Hsd11b1 Endou</td>
<td>331</td>
<td>118</td>
<td>69</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Tnc</td>
<td>Hsd11b1 Endou</td>
<td>595</td>
<td>193</td>
<td>112</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>Tmem163 Dmrtb1</td>
<td>Hsd11b1 Endou</td>
<td>471</td>
<td>90</td>
<td>53</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>Tmem163 Arhgap25</td>
<td>Whrn Tox2</td>
<td>197</td>
<td>63</td>
<td>40</td>
<td>1</td>
</tr>
</tbody>
</table>
gene which is part of a family of purinergic receptors that have been implicated in functions such as learning, memory and sleep. In the DGE analysis, no evidence for differential expression of P2rx4 was found at the gene level (FDR-adjusted p-value = 1). By contrast, in our DTU analysis the transcripts of P2rx4 displayed the highest statistical evidence for differential usage within the set of transcripts that could be assigned to the ontology class ‘neuron projection’.52 The mean usage of transcript ENSMUST00000081554 is estimated to be 28.9% in Tnc cells and 75.9% in Hsd11b1 Endou cells (FDR-adjusted p-value = 1.22E-13). For transcript ENSMUST00000195963, the transcript usage changes from 58.2% in Tnc cells and 16.6% in Hsd11b1 Endou cells (FDR-adjusted p-value = 1.79E-10). For the third transcript of P2rx4 that was assessed in our DTU analysis, ENSMUST00000132062, we found no statistical evidence for DTU (FDR-adjusted p-value = 0.534). In Figure 6, we

Figure 6. Differential transcript usage in the P2rx4 gene. Each panel shows transcript usage or gene expression across cells of the Tnc and Hsd11b1 cell types. For the transcript-level figures, the size of each datapoint is weighted according to the total expression of the gene in that cell, i.e. the gene counts per cell. The yellow diamonds indicate the estimated mean usage of a transcript for each cell type, as estimated by satuRn. The cyan and dark green diamonds indicate mean and median gene expression levels per cell type, respectively. The two top panels display the transcript usage across cells of the Tnc and Hsd11b1 Endou cell types of transcripts ENSMUST00000081554 and ENSMUST00000195963, respectively. The proportion of usage of the former transcript is clearly higher in Hsd11b1 Endou cells, while the latter transcripts is most abundant in Tnc cells. For the third transcript ENSMUST00000132062 (bottom left panel) there is no evidence for differential usage between both cell types. In addition, there is no evidence for differential expression of P2rx4 on the gene level (bottom right panel). DTU and DGE significance levels are indicated in the figure headers.
show the output for the visualization of the transcript usages for P2rx4 as obtained from satuRn. Interestingly, the majority transcript in the Tnc cell type, ENSMUST00000195963, is not protein coding.\(^5\) By contrast, the majority transcript in the Hsd11b1 Endou cell type, ENSMUST00000081554, is coding for the P2X purinoceptor protein (UniProt ID Q9Z256). As such, the changes in transcript usage between both cell types represent actual biological differences in the functionality of the gene products, which may be relevant to the process of neuron projection. This functional difference would have remained obscured when only performing a canonical DGE analysis.

**Comparison to limma diffsplice**

We also analyzed the case study dataset with limma diffsplice.\(^2\) When running limma diffsplice with default settings, a large number of DTU transcripts was returned (Extended data figure S12\(^2\)) and we observe that the p-values were shifted towards smaller values (Extended data figures S13 and S14). Therefore, we adopted the same empirical null strategy as implemented in satuRn to post-process the results. While this dramatically decreased the number of significant DTU transcripts, limma diffsplice still identified more transcripts (i.e. true or false positives) than our method. However, when we inspected the transcripts that were highly ranked in the top DTU list of limma diffsplice but lowly ranked in our top list, we found that most of these transcripts either originate from genes that are lowly expressed, or they are transcripts with a large fraction of zero counts (i.e. zero expression in a large percentage of cells). Limma diffsplice thus claims differential usage more often for transcripts that only contain little information for assessing DTU. This is depicted in Figure 7.

This behavior can be expected. Limma diffsplice tests for DTU by comparing the log-fold change in expression of transcript \(t\) with the average log-fold change in the expression of all transcripts belonging to the same gene as transcript \(t\). As such, limma diffsplice does not incorporate any information on the absolute gene expression levels. In contrast, our quasi-binomial GLM framework models the log-odds of drawing a particular transcript \(t\) from the pool of transcripts in the corresponding gene. As a consequence, transcripts belonging to lowly expressed genes are correctly considered less informative in satuRn and are thus less likely to be picked up. For example, in Figure 8A, we show that while our method estimates a mean usage of 7% in Tnc cells and 26% in Hsd11b1 Endou cells (indicated by the gold diamond), the transcript is not identified as differentially used, given the low abundance of the corresponding gene and the highly variable single-cell level observations.

Conversely, by looking at the transcripts that were highly ranked in our DTU list but lowly ranked in the top list of limma, we observe that our model is more likely to capture small changes in transcript usage that are stable across all cells and

![Figure 7](https://example.com/fig7.png)

**Figure 7.** Global comparison between differential transcript usage (DTU) transcripts uniquely identified by satuRn, uniquely identified by limma diffsplice or by both methods. **Left panel:** Boxplots on the average gene-level count for the DTU genes identified by the respective methods. Transcripts uniquely identified by satuRn originate from genes that have a much higher gene-level count (averaged over cells) as compared to transcripts uniquely identified by limma diffsplice. Note that the y-axis is displayed on a log10 scale. **Right panel:** Violin plots indicating the fraction of cells in which the transcripts are expressed. Transcripts uniquely identified by satuRn are expressed, on average, in a much larger fraction of the cells. Conversely, transcripts identified as DTU uniquely by limma diffsplice often have no expression in a large fraction of the cells. The dark green diamond indicates the median fraction of cells in which the DTU transcripts are expressed.
belong to genes that are highly expressed. An example of such a transcript is shown in Figure 8B. satuRn estimates a mean usage of 3% in Tnc cells and 6% in Hsd11b1 Endou cells. While this is only a minor change in transcript usage, satuRn still identifies this transcript as differentially used because the gene is highly expressed and the small change in usage is

Figure 8. Three examples displaying differential transcript usage (DTU) transcripts that are uniquely identified by satuRn or limma diffsplice. Each panel shows transcript usage across cells of the Tnc and Hsd11b1 cell types. The size of each datapoint is weighted according to the total expression of the corresponding gene in that cell, i.e. the total gene count per cell. The yellow diamonds indicate the estimated mean usage of a transcript for each cell type, as estimated by satuRn. The cyan diamonds indicate the mean transcript expression levels per cell type. The header of each panel indicates the FDR-adjusted p-value and the rank of the DTU finding in the top lists by limma diffsplice and satuRn analyses. Panel A: Transcript uniquely identified as differentially used by limma diffsplice. The DTU claim by limma is driven by the difference in mean transcript usage between cell types. Given the low abundance of the corresponding gene and the highly dispersed single-cell level observations, satuRn does not identify the transcript as differentially used. Panel B: Transcript uniquely identified as differentially used by satuRn. Even though the mean difference in transcript usage between cell types is estimated to be 3%, satuRn claims significance given that the difference is stably supported by many cells with high gene-level expression levels. Panel C: Transcript uniquely identified as differentially used by limma diffsplice. The DTU claim by limma is driven by the raw mean difference in transcript usage between cell types. In contrast, satuRn takes into account that the Hsd11b1 Endou cells expressing the transcript at 0% usage have low gene-level count. The size of the dots (which represent individual cells) is weighted according to the total expression of the gene in that cell, i.e. the total gene count per cell. The yellow diamonds indicate the estimated mean usage of a transcript for each cell type, as estimated by satuRn. The cyan diamonds indicate the raw mean transcript usage levels per cell type.
supported by a large number of cells. In case such small differences in usage are not considered biologically meaningful, it is possible to set a threshold on the minimal desired difference. Finally, by not taking into account gene abundances, limma is more influenced by outlying observations that have a low gene-level abundance (Figure 8C). Indeed, DTU claims by limma are driven by differences in raw mean usages of transcripts. In Figure 8C, the raw mean usage of the transcript is 77% in Tnc cells and 45% in Hsd11b1 Endou cells, as indicated by the cyan diamonds. By contrast, the mean usage estimate by satuRn, which takes into account that the Hsd11b1 Endou cells expressing the transcript at 0% usage have low gene-level count, is 83% for Tnc cells and 75% for Hsd11b1 Endou cells, as indicated by the gold diamonds.

We therefore argue that, given the above observations, the transcripts identified by satuRn should be considered more reliable, as they generally originate from genes containing more information for assessing DTU.

Comparison to DoubleExpSeq
We additionally analyzed the dataset by Tasic et al. with DoubleExpSeq. DoubleExpSeq identified a large number of DTU transcripts in all eight comparisons between cell types, ranging from 335 to 4580 DTU transcripts (Extended data figure S12). This is consistent with our performance benchmarks, which already suggested that DoubleExpSeq becomes overly liberal in single-cell datasets with a large number of cells (Figures 4, Extended data figures S7, S8 and S9). We therefore expect many of these transcripts to correspond to false positives. Furthermore, this is reflected in the pathological distribution of p-values obtained by DoubleExpSeq, where p-values have a tendency to be small and therefore the analysis too liberal (Extended data figure S15). Furthermore, as discussed in the benchmark studies, we could not adopt the empirical null strategy to improve the FDR control of DoubleExpSeq. Again, a large number of p-values equal 1 poses a problem for estimating the empirical null distribution (Extended data figure S16).

While the results of DoubleExpSeq are likely to be overly liberal, the ranking of the transcripts (based on the p-values of the DTU analysis) might still be reasonable. In Figure 9 we observe a large overlap between the top 200 transcripts identified by satuRn in comparison #6 of the case study and the top 200 transcripts of DoubleExpSeq in that comparison. This overlap is considerably smaller with a limma diffsplice analysis.

Finally, we note that while DoubleExpSeq could still be used in this case study given the simple factorial design (using a single factor to assign each cell to a cell type), DoubleExpSeq cannot be used in multifactorial designs, for instance to compare expression levels across multiple cell types between multiple samples or treatment groups.

Discussion
In this manuscript, we have proposed satuRn, a new software tool for DTU analysis. satuRn adopts a quasi-binomial GLM framework and obtains direct inference on DTU by modelling the relative usage of a transcript, in comparison to other transcripts from the same gene, between conditions of interest. We evaluated the performance of satuRn with respect to seven other DTU methods on three simulated bulk RNA-seq datasets, a real bulk RNA-seq dataset and three real scRNA-seq datasets. These benchmarks underscored the strong performance of satuRn, as well as its ability to
control the FDR close to the nominal level. In addition, we showed that satuRn scales seamlessly to the large data volumes that are produced in contemporary (sc-)RNA-seq experiments. Furthermore, given the underlying GLM framework, our method can handle complex experimental designs that are commonplace in scRNA-seq experiments. Finally, satuRn can extract biologically relevant information from a large scRNA-seq dataset that would have remained obscured in a canonical DGE analysis.

Since most sequencing reads map to multiple transcripts, quantification tools such as Salmon or kallisto only provide an estimate of the expected number of fragments originating from each transcript. Incorporating quantification uncertainty has recently been shown to improve results in differential expression analysis of single-cell RNA-seq datasets.54 Currently, satuRn and all other DTU methods discussed in this manuscript, except for BANDITS,30 neglect the uncertainty on this abundance estimate. BANDITS models the abundance uncertainty; however, it had a markedly lower performance than our method in our benchmark evaluation (Extended data figure S125).

One challenge common to all DTU methods is that the power to detect differentially used transcripts depends strongly on the quality of the scRNA-seq dataset. This becomes clear when comparing the performances for the three different scRNA-seq benchmarks in this manuscript. The performances on the Darmanis26 dataset (Extended data figure S9) are markedly lower than the performances on the other two datasets (Figure 4 and Extended data figure S8). A closer inspection of the Darmanis dataset showed that, after filtering, the transcript-level counts matrix contains a much larger percentage of zero counts than the other datasets. We also more frequently observed the scenario where the expression level of a gene could be attributed to a single isoform. This effectively causes the transcript usage to appear binary, with either 0% or 100% usages of a certain transcript. We argue that while this may reflect the true underlying biology, for instance through the process of transcriptional bursting,30 it is more likely to be a technical artefact as a consequence of more shallow sequencing, given the lower percentage of binary usage profiles in the Chen and Tasic datasets. The supposedly binary expression of transcripts due to coverage-dependent bias and the use of more stringent filtering criteria to reduce this bias has already been comprehensively reported by Najar et al.57

We conclude with the following recommendations for DTU analysis from an applied perspective. In case of small bulk RNA-seq datasets, satuRn, DEXSeq and DoubleExpSeq can be used interchangeably. In case of datasets with more complex designs that require the DTU model to incorporate additional covariates, e.g. batch effects, DoubleExpSeq cannot be used. For single-cell datasets, using DEXSeq will become infeasible in terms of scalability and DoubleExpSeq may give overly liberal results. As such, we recommend satuRn for performing DTU analyses in large bulk and single-cell RNA-seq datasets.

Data availability
Underlying data
Zenodo: Datasets associated with this publication https://doi.org/10.5281/zenodo.4439415.58

This project contains the following underlying data:

- **Case_study.zip** (Transcript-level expression count matrix Tasic_caseStudy_transcript_counts.Rds and corresponding metadata files Tasic-metadata_1.xlsx and Tasic-metadata_2.csv of a subset of the dataset by Tasic et al.12)

- **Performance_Chen.zip** (Transcript-level expression matrices Chen_counts.Rds and Chen_scaledTPM.Rds, as well as the corresponding metadata file Chen_metadata.csv of the dataset by Chen et al.25)

- **Performance_Darmanis.zip** (Transcript-level expression count matrix Darmanis_counts.Rds and the corresponding metadata file Darmanis-metadata.Rdata of the dataset by Darmanis et al.26)

- **Performance_Dmelanogaster.zip** (Dmelanogaster_kallisto is a folder containing the full output of the quantification of the Dmelanogaster dataset40 as quantified with kallisto1 version 0.46.0. The corresponding metadata can be retrieved from Dmelanogaster-metadata_1.txt and Dmelanogaster-metadata_2.txt.)

- **Performance_GTEx.zip** (Transcript-level expression matrices GTEx_counts.gz and GTEx_scaledTPM.gz, as well as the corresponding metadata file GTEx-metadata.txt of the GTEx dataset.41)

- **Performance_Hsapiens.zip** (Hsapiens_kallisto is a folder containing the full output of the quantification of the Hsapiens dataset30 as quantified with kallisto1 version 0.46.0. The corresponding metadata can be retrieved from Hsapiens-metadata_1.txt and Hsapiens-metadata_2.txt.)
- **Performance_Love.zip** (*Love_kallisto*) is a folder containing the full output of the quantification of the dataset by Love *et al.* as quantified with salmon' version 0.1.0. The corresponding metadata can be retrieved from *Love_metadata.rda*. Effective transcript length estimates for BANDITS are available from *Love_eff_len.rds*.

- **Performance_Tasic.zip** (Transcript-level expression matrices *Tasic_counts.Rds* and *Tasic_scaledTPM.Rds*, as well as the corresponding metadata files *Tasic_metadata_1.xlsx* and *Tasic_metadata_2.csv* of a subset of the dataset by Tasic *et al.*

- **Scalability_analysis.zip** (Several .Rdata files containing the scalability results of the different DTU tools on datasets of different sizes.)

In addition, all folders except *Scalability_analysis.zip* contain intermediate DTU analysis results that are available as .Rdata files or, in the case of *Case_study.zip*, .Rds files. Data are available under the terms of the Creative Commons Attribution 4.0 International license (CC-BY 4.0).

**Extended data**


This project contains the following extended data:

- **Supplementary_Figures.pdf**: The supplementary figures to the satuRn publication, including figure captions.

- **DTU_Methods_Detail.pdf**: A .pdf text file that describes the different DTU tools that were included in our benchmarks in greater detail as compared to the description in our main publication. For even higher detail, we refer to the respective original publications.

- **GSEA_MSigDB.xlsx**: The output of the Gene Set Enrichment Analyses (GSEA) for the case study of our publication as generated by the online MSigDB platform.

License: Data are available under the terms of the CC-BY 4.0 license.

**Software availability**

Source code available from: [https://github.com/statOmics/satuRn](https://github.com/statOmics/satuRn)

Archived source code at time of publication: [https://doi.org/10.5281/zenodo.4656084](https://doi.org/10.5281/zenodo.4656084).

License: Data are available under the terms of the CC-BY 4.0 license.

Code to reproduce analyses and figures available from: [https://github.com/statOmics/satuRnPapers](https://github.com/statOmics/satuRnPapers)

Archived analysis code at time of publication: [https://doi.org/10.5281/zenodo.4655310](https://doi.org/10.5281/zenodo.4655310).

License: Data are available under the terms of the CC-BY 4.0 license.

**Acknowledgements**

The authors would like to thank Milan Malfait for his suggestions and comments throughout this project. We also note that an earlier version of this article can be found in bioRxiv.
References


Publisher Full Text


Open Peer Review

Current Peer Review Status: 🟢 🟡

Version 1

Reviewer Report 21 June 2021

https://doi.org/10.5256/f1000research.54945.r85196

© 2021 Reyes A. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Alejandro Reyes
Novartis Institutes for BioMedical Research (NIBR), Basel, Switzerland

The authors propose a new method to test for differences in transcript usage for both bulk RNA-seq and single-cell RNA-seq data. The manuscript is clear and the method is sound and well explained. The authors provide an excellent implementation of their method as a Bioconductor package whose vignette is easy to follow. The main advantage and novelty of satuRn is that it scales well with very large samples, which is a limitation of some of the best performing methods in literature. Below are a couple of suggestions to provide more context to the DTU events on the benchmarks, assess to what extent DTU for single cells might be a limitation and a couple of requests for clarifications in the manuscript.

○ The transcript swapping strategy seems quite powerful to benchmark DTU methods. This strategy is probably introducing DTU of different magnitudes. It would be highly informative if the authors can assess the performance of the different methods as a function of the strength of the induced DTU. How is the performance of satuRn and the other tools across different ranges of DTU induction?

○ The authors describe a strategy to calculate p-values based on t-test on the estimated log-odds transcript ratios. They also describe that this strategy is too liberal for scRNA-seq analyses and provide an alternative approach that converts these p-values to z-scores and use locfdr to estimate an empirical null distribution. Based on the paper and the vignette, it was unclear to me which strategy is followed for bulk RNA-seq analyses? Does the user have the possibility of choosing which strategy to use?

○ The filtering strategy for scRNA-seq datasets is very stringent (at least 1 count in 50% of the cells) and might result in mostly highly expressed genes/transcripts. How many genes can be tested for DTU after such thresholds are applied in real scRNA-seq data? It would also be very informative to show benchmarks with less stringent criteria to assess how the presence of many zeros affect each method.

○ Given that most scRNA-seq protocols only sequence the 3' end of transcripts, it might be unfeasible to reliably estimate transcript expression (this is discussed in Ntranos et al. (2019
Examples would be when the transcripts are not distinguishable from the 3’ends alone. The authors should consider testing their approach and benchmarks at the level of transcript compatibility counts.

- I found unclear whether stageR applied by default (as I understand from the vignette)? If so, I think it is important to apply stageR to all the methods tested in the benchmarks to distinguish the performance of saturn vs satuRn + stageR.

- The authors describe the applicability of their method for differential transcript usage. However, as I understand the model, it could also be used for differential exon usage, or groups of reads supporting an alternative splicing event (counts from alternative exon usage, or groups of reads supporting an alternative splicing event A counts from event B, or intron clusters as in leafcutter, etc). I think the authors could showcase these examples to expand the applications of their model.

- In most ROC curves it’s a bit hard to distinguish each different line due to over plotting.

References

Is the rationale for developing the new method (or application) clearly explained?
Yes

Is the description of the method technically sound?
Yes

Are sufficient details provided to allow replication of the method development and its use by others?
Yes

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bioinformatics, Computational Biology, Drug Discovery

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have
significant reservations, as outlined above.

Reviewer Report 10 June 2021

https://doi.org/10.5256/f1000research.54945.r85874

© 2021 Cmero M. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Marek Cmero

1 Peter MacCallum Cancer Centre, Melbourne, Vic, Australia
2 Sir Peter MacCallum Department of Oncology, The University of Melbourne, Parkville, Vic, Australia

Gilis et al. present satuRn, a novel method for performing differential transcript usage analysis on bulk and single-cell RNA-seq data. The approach improves upon existing methods through computational scalability and support for multifactorial experimental designs, while also tightly controlling FDR rates for bulk and single-cell experiments. The authors demonstrate performance using simulated, as well as real bulk and single-cell RNA-seq data, showing that satuRn performs on par with the best available approaches, while controlling the FDR and scaling favourably to large numbers of cells and transcripts.

The paper is well written, logical and easy to follow. The experimental methodology and results are sound and well presented. The authors have also done a great job with the software. I was able to install satuRn and run the vignette without any difficulty.

The following points would help improve the manuscript:

- The manuscript cites support for multifactor experimental designs as a key advantage of satuRn, however, this is not demonstrated. A small case study experiment with a multifactor experimental design, perhaps using the Tasic et al. data, would be helpful in illustrating this feature.

- The authors benchmark satuRn's scalability on scRNA data, however, not on bulk RNA-seq (although competing methods are benchmarked on bulk data in Figure 1). While this is computationally less challenging, it would be helpful for the authors to also show satuRn's scalability on bulk RNA-seq, for instance, using the Love et al. data as in Figure 1.

- The Discussion notes that performance on different scRNA datasets is affected by the percentage of zero counts: It would be helpful if the authors could show the zero count distributions in the three different scRNA data sets to illustrate this.

- It can be quite difficult to compare across the FDR-TPR curves when assessing performance using the different filters and abundance estimation methods. Perhaps for a subset of the comparisons and only showing the best performing tools, the authors could consider adding a set of simplified figures that compares the different filtering/counting approaches...
The authors state that performance of raw abundance estimates was better in all data sets, except for the Love et al. data, which is presumably due to the simulation methodology. Do the authors have any insights into why this dataset performed better using pseudo-alignments?

The quantification methodology of the scRNA data is not described in the methods and should be included.

In the two sections describing the real bulk and real single-cell study data, the authors refer to the ‘swapping strategy’ described in the Methods. However, two different swapping strategies are detailed. Please clarify.

In the ‘Performance assessment’ section, the acronym FDP is not spelled out.

Panels A-C in Figure 8 are missing labels.

It would be helpful if the authors could explain the null distribution histograms a bit more, i.e. what the green and blue lines, red triangles and highlighted distributions show.

**Is the rationale for developing the new method (or application) clearly explained?**
Yes

**Is the description of the method technically sound?**
Yes

**Are sufficient details provided to allow replication of the method development and its use by others?**
Partly

**If any results are presented, are all the source data underlying the results available to ensure full reproducibility?**
Yes

**Are the conclusions about the method and its performance adequately supported by the findings presented in the article?**
Yes

**Competing Interests:** No competing interests were disclosed.

**Reviewer Expertise:** Bioinformatics, method development, transcriptomics

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.
The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com