BANDITS: Bayesian differential splicing accounting for sample-to-sample variability and mapping uncertainty

Simone Tiberi[1,2*] and Mark D Robinson[1,2]

Abstract

Alternative splicing is a biological process during gene expression that allows a single gene to code for multiple proteins; however splicing patterns can be altered in some conditions or diseases. Here, we present BANDITS, a R/Bioconductor package to perform differential splicing, at both gene and transcript-level, based on RNA-seq data. BANDITS uses a Bayesian hierarchical structure to explicitly model the variability between samples, and treats the transcript allocation of reads as latent variables. We performed an extensive benchmark across both simulated and experimental RNA-seq datasets, where BANDITS has extremely favourable performance with respect to the competitors considered.

Keywords: Alternative Splicing; Differential Splicing; Differential Transcript Usage; RNA-seq; Transcriptomics; Bayesian hierarchical modelling; Markov chain Monte Carlo

Background

Alternative splicing plays a fundamental role in the biodiversity of proteins as it allows a single gene to generate several transcripts and, hence, to code for multiple proteins [1]. However, variations in splicing patterns can be involved in development and disregulated in disease [2–4]. Differential splicing (DS) studies how splicing patterns vary between experimental conditions, and specifically, differential transcript usage (DTU) represents a primary branch to investigate DS [5]. DTU is present when there are changes, between two or more conditions, in the relative abundances of transcripts (i.e., in the transcript proportions), irrespective of the overall output of transcription. Alternative approaches to investigate DS are differential exon usage (DEU) [6], event specific differential splicing based on percent-spliced-in [7–9], and differential transcript expression (DTE) [5], which focuses on changes in the overall abundance of isoforms and, hence, identifies both differential gene expression (DGE) as well as differential splicing.

A significant challenge of DTU, and in general of DS, is that transcript-level counts (i.e., the number of RNA-seq reads originating from each isoform), which are of primary interest, are not observed because most reads map to multiple transcripts (and sometimes, multiple genes). Quantification tools [10] such as Salmon [11] or kallisto [12] allow, via expectation maximization (EM) algorithms, to estimate the expected number of fragments originating from each tran-

[*] Correspondence: simone.tiberi@uzh.ch

[1] Institute of Molecular Life Sciences, University of Zurich, Winterthurerstrasse 190, 8057, Zurich, Switzerland

[2] SIB Swiss Institute of Bioinformatics, 8057, Zurich, Switzerland

Full list of author information is available at the end of the article

script. Most methods for DS (notably, DRIMSeq [13], BayesDRIMSeq [14] and SUPPA2[1] [9]) follow a *plug-in* approach by inputing transcript estimated counts (TECs) and treating them as observed counts, thus neglecting the uncertainty in the estimates. In an attempt to mitigate this issue, rats [15] inputs TECs together with their bootstrap replicates; nevertheless rats is limited by the fact that it uses a G-test based on the Multinomial distribution, which assumes all biological replicates to share the same relative transcript abundance.

Instead of considering TECs, some methods, such as DEXSeq [6] and limma (via *diffSplice* function) [16], perform DEU by testing exon bin counts, which are observed directly; however, reads overlapping multiple exon bins are counted multiple times, once for each exon bin they map to. Furthermore, differential testing is done at the exon level, while transcript-level tests and proportions cannot be computed; for this reason, DEU is widely considered as a surrogate for DTU [5]. An alternative approach, ignoring the quantification step, considers the groups of transcripts that reads are compatible with, usually referred to as equivalence classes (ECs), and the respective counts. Recently, two articles [17, 18] proposed to perform DTU by applying DEXSeq on transcript estimated counts or on equivalence classes counts (ECCs); however, both approaches have limitations. The former, similarly to DRIMSeq, BayesDRIMSeq and SUPPA2, inputs TECs while ignoring their inherent variability. The latter, instead, has limited interpretability because testing cannot be done at the transcript level and transcript-level proportions cannot be computed; moreover, equivalence classes containing transcripts from distinct genes are excluded from the analyses. A further method considering ECs is cjBitSeq [14], which performs a full Bayesian analysis and samples the allocation of each read to its transcripts of origin; however cjBitSeq, similarly to rats, does not allow for sample-specific proportions. Moreover, in the DTU implementation of cjBitSeq[2], the equivalence classes containing transcripts from multiple genes are considered multiple times (once for each gene contained in the EC).

In order to overcome the limitations of current methods for DTU, we present BANDITS (Bayesian ANalysis of DIfferenTial Splicing), a R/Bioconductor package to perform DTU between two or more groups of samples, based on RNA-seq data. BANDITS uses a Bayesian hierarchical model, with a Dirichlet-multinomial structure, to explicitly model the sample-to-sample variability between biological replicates, and inputs the equivalence classes and respective read counts, by treating the transcript allocations of reads as latent variables, i.e., as parameters

[1] SUPPA2 performs both event-specific DS as well as canonical (transcript-level) DTU. Here, we only consider the DTU application of SUPPA2.

[2] cjBitSeq can perform both DTE and DTU analyses. Here, we refer to its DTU method only.

that are sampled, jointly with the model parameters, via Markov chain Monte Carlo (MCMC) techniques. ECCs can be obtained by aligning reads either to a reference transcriptome, with pseudo-aligners Salmon [11] and kallisto [12], or to a reference genome with splice-aware genome aligner STAR [19], and computing the ECCs of the aligned reads via Salmon.

Despite the abundance of DS methods available in the literature, BANDITS introduces some unique features and, in both simulation and experimental data analyses, shows very favourable performance with respect to all the competitors we considered. Supplementary Table S1 summarizes the main features of the most popular methods for DTU based on RNA-seq data. BANDITS is the only DS tool that jointly allows for sample-specific proportions between biological replicates while also sampling the transcript allocation of reads. It is also the only DS method to sample the gene allocation of reads in equivalence classes that contain transcripts from distinct genes (Cmero et al. [18] exclude these ECs, while cjBitSeq considers these classes multiple times, once per gene). Furthermore, BANDITS is the first work to correct for the transcript (effective) lengths when computing the relative abundance of isoforms; hence, it is able to disentangle the probability that reads map to a transcript, from the probability of expressing a transcript (see Results), and uses the latter parameter for statistical testing. BANDITS tests for DTU at both transcript and gene level, allowing scientists to investigate what specific transcripts are differentially used (DU) in selected genes. Furthermore, our tool is not limited to two group comparisons and also allows to test for DTU when samples belong to more than two groups. Finally, despite the computational complexity of full MCMC algorithms, the MCMC sampling is coded in C++, which makes BANDITS highly efficient and feasible to run on a laptop, even for complex model organisms.

Results

The BANDITS hierarchical model

Consider a gene with K transcripts and N samples (i.e., biological replicates) from a given group. We define the latent vector of transcript-level counts for the i-th subject as $X^{(i)} = \left(X_1^{(i)}, \ldots, X_K^{(i)}\right)$, where $X_k^{(i)}$ indicates the number of reads originating from the k-th transcript in the i-th sample, with $i = 1, \ldots, N$ and $k = 1, \ldots, K$. We use a Bayesian hierarchical model [20, 21], which represents a natural approach to gather information from distinct samples, while allowing for sample-specific parameters, in a statistically rigorous way. We assume that $X^{(i)}$ was generated from a multinomial distribution:

$$X^{(i)} \Big| \pi^{(i)} \sim \mathcal{MN}\left(n^{(i)}, \pi^{(i)}\right), i = 1, ..., N, \qquad (1)$$

where $\pi^{(i)} = \left(\pi_1^{(i)}, ..., \pi_K^{(i)}\right)$, with $\pi_k^{(i)}$ indicating the relative abundance of the k-th transcript within the gene in the i-th sample, $n^{(i)}$ represents the total number of counts arising from the gene of interest in the i-th sample, and $\mathcal{MN}(\cdot)$ denotes the multinomial distribution. Assuming independence between genes, the full likelihood for all N samples in a group is defines as:

$$L\left(\underline{\pi} | \underline{x}\right) = \prod_{i=1}^{N} f_{\mathcal{MN}}\left(x^{(i)} \middle| n^{(i)}, \pi^{(i)}\right), \tag{2}$$

where $f_{\mathcal{MN}}(\cdot)$ indicates the density of the Multinomial distribution, $\underline{\pi} = \left(\pi^{(1)}, \ldots, \pi^{(N)}\right)$, and $\underline{x} = \left(x^{(1)}, \ldots, x^{(N)}\right)$, with $x^{(i)} = \left(x_1^{(i)}, \ldots, x_K^{(i)}\right)$ being the realization of the random variable $X^{(i)}$, $i = 1, \ldots, N$.

The transcript proportions for each sample are connected via a common Dirichlet prior distribution:

$$\pi^{(i)} \sim \mathcal{DIR}(\delta), i = 1, ..., N, \tag{3}$$

with $\mathcal{DIR}(\cdot)$ denoting the Dirichlet distribution and $\delta = (\delta_1, ..., \delta_K)$, where $\delta_+ = \sum_{k=1}^K \delta_k$ is the precision parameter, modelling the degree of over-dispersion between samples, and $\bar{\pi} = (\bar{\pi}_1, \ldots, \bar{\pi}_K)$, with $\bar{\pi}_k = \frac{\delta_k}{\delta_+}$ indicating the mean relative abundance of the k-th transcript, for $k = 1, \ldots, K$. The prior distribution for the hierarchical parameters is:

$$P\left(\underline{\pi} | \delta\right) = \prod_{i=1}^{N} f_{\mathcal{DIR}}\left(\pi^{(i)} \middle| \delta\right), \tag{4}$$

where $f_{\mathcal{DIR}}(\cdot)$ indicates the density of the Dirichlet distribution.

In order to exploit the information from other genes, we take advantage of DRIMSeq [13] to infer genewise precision parameters, and use these estimates to formulate an informative prior for δ_+. If precision estimates are not computed, all δ_k parameters follow a vaguely informative prior distribution (see Methods).

Since most reads map to multiple transcripts, transcript-level counts are typically not observed directly. BANDITS inputs, for every gene, the equivalence classes of transcripts and respective counts, while the transcript-level counts are treated as latent variables and are sampled together with the model parameters (see Methods). In ECs with transcripts from more than 1 gene, the gene allocation of reads is also treated as a latent variable and sampled within the MCMC scheme (see Supplementary Section S1.2).

MCMC overview

In order to infer the posterior distribution of the model parameters, we developed a Metropolis-within-Gibbs [22–24] MCMC algorithm where parameters are alternately sampled in three blocks: δ, via a Metropolis algorithm [23, 24] with an adaptive random walk proposal [25], $\underline{\pi}$ and $\underline{X}$, both via a Gibbs sampler [26, 27]. The mathematical details of the sampling scheme are illustrated in Supplementary Section S1.1.

After discarding an initial *burn-in*, the convergence of chains and a potentially wider *burn-in* are assessed via Heidelberger and Welch's stationarity test [28]. To avoid potential false positive results due to poor mixing, if the gene-level test has a p-value below 0.1, a second independent MCMC chain is run and results are recomputed on the aggregation of the two chains (*burn-in* excluded).

Accounting for transcript lengths

We introduce a conceptual distinction between the probability that reads map to a transcript, which depends on the transcript length, and the probability that a gene expresses a transcript. While the former parameter is typically used to test for DTU, we argue that the latter should be employed instead, because it reflects the number of transcripts expressed by a gene, independently of their length. We use the mean relative abundance of transcripts, $\bar{\pi}$, to compute the average probability of expressing transcripts, $\bar{\pi}^T = \left(\bar{\pi}^T_1, \dots, \bar{\pi}^T_K\right)$, where $\bar{\pi}^T_k = \frac{\bar{\pi}_k / l_k}{\sum_{k'=1}^K \bar{\pi}_{k'} / l_{k'}}$, with l_k being the effective length of the k-th transcript, for $k = 1, \dots, K$. In the previous formula, at the numerator we normalize $\bar{\pi}_k$ with respect to the effective length of the k-th isoform, while the denominator term is a scaling factor to ensure that $\sum_{k=1}^K \bar{\pi}^T_k = 1$. In simulation studies, we noticed that testing for DTU via $\bar{\pi}^T$ leads to improved performance compared to using $\bar{\pi}$. Furthermore, unlike other methods for DTU, BANDITS provides users an estimate of the mean transcript relative expression $\bar{\pi}^T$.

DTU testing

After inferring the model parameters, we test for DTU by comparing $\bar{\pi}^T$ between conditions. Given groups A and B, with average transcript relative expression, for the k-th transcript, $\bar{\pi}^{TA}_k$ and $\bar{\pi}^{TB}_k$, respectively, we test the following system of hypotheses:

$$\begin{cases} \mathcal{H}_0 : & \omega_k = 0, \text{ for } k = 1, \dots, K, \\ \mathcal{H}_1 : & \text{otherwise}, \end{cases} \tag{5}$$

where $\omega_k = \bar{\pi}_k^{TA} - \bar{\pi}_k^{TB}$, $k = 1, \ldots, K$. We approximate the posterior distribution of $\omega = (\omega_1, \ldots, \omega_K)$ with a multivariate normal density [29], $\omega | D \dot{\sim} \mathcal{N}\left(\hat{\omega}, \hat{\Sigma}_{\hat{\omega}}\right)$, where $\hat{\omega}$ represents the posterior mode of ω and $\hat{\Sigma}_{\hat{\omega}}$ its covariance matrix, both inferred from the posterior chains, D denotes the input data (i.e., the ECCs) and $\mathcal{N}(\mu, \Sigma)$ indicates the normal density with mean μ and covariance Σ. In order to test for DTU at the gene level, BANDITS performs a multivariate Wald test [30], based on the normal approximation of ω, to test the set of hypotheses (5).

Our method also can unravel the specific transcripts that are DU by testing, for the k-th transcript, the following system of hypotheses: $\mathcal{H}_0 : \omega_k = 0$, vs. $\mathcal{H}_1 : \omega_k \neq 0$. Similarly to the gene-level test, we perform a univariate Wald test based on the normal approximation of the marginal posterior distribution of ω_k: $\omega_k \left| D \dot{\sim} \mathcal{N}\left(\hat{\omega}_k, \hat{\sigma}^2_{\hat{\omega}_k}\right)\right.$, where $\hat{\omega}_k$ and $\hat{\sigma}^2_{\hat{\omega}_k}$ represent the posterior mode and variance of ω_k, respectively, both inferred from the posterior chains. In both gene and transcript-level testing, false discovery rate (FDR) control is obtained by adjusting p-values via Benjamini-Hochberg correction [31].

BANDITS also outputs conservative gene and transcript-level scores, as well as a measure of the strength of DTU (see Methods). Furthermore, our method also allows to test for DTU between 3 or more conditions (see Supplementary Section S1.3).

Simulation studies

We performed three RNA-seq stimulation studies to benchmark BANDITS against nine other DS methods. Details about the simulation and experimental data analyses are reported in Supplementary Section S1.4, while software versions are displayed in Table S2.

First, we considered the human simulation from Soneson et al. [32], where two groups of 3 samples each are compared, and DU genes are simulated by inverting the relative abundance of the two most expressed transcripts across conditions.

We also built a second simulation dataset, from a human genome with two groups of 6 samples each, where DU genes are simulated by randomly permuting the relative abundance of the four most expressed transcripts; if a DU gene has two or three transcripts only, then those are permuted. In our view, this second simulation provides a more varied scenario compared to the first one: the dominant transcript (i.e., the most abundant isoform) does not always change between conditions and some genes will exhibit more changes, but whose magnitude might be smaller. This simulation is made available via FigShare (DOI *10.6084/m9.figshare.9467144*, *10.6084/m9.figshare.9692429* and *10.6084/m9.figshare.9692918*). We will refer to the former and latter datasets as “3 vs. 3” and “6 vs. 6”, respectively.

As a third scenario, we considered the 6 vs. 6 simulation and filtered transcripts, before the differential analyses, based on Salmon estimated counts: we kept transcripts with least 10 counts (across all samples) and an average relative abundance of at least 0.01.

We benchmarked BANDITS against several competitors: BayesDRIMSeq, cjBitSeq, DEXSeq, DEXSeq on ECCs (DEXSeq_ECCs), DEXSeq on TECs (DEXSeq_TECs), DRIMSeq, limma (via *diffSplice* function), rats and SUPPA2. We also consider the conservative gene and transcript-level scores from BANDITS, BANDITS_inv and BANDITS_maxGene (see Methods), as well as the ones from BayesDRIMSeq and cjBitSeq, that we call BayesDRIMSeq_inv and cjBitSeq_inv. Note that SUPPA2 does not perform a global gene-level test: in order to obtain a gene-level score we considered the minimum of the transcript-level adjusted p-values. For cjBitSeq transcript-level test, we used the probability that a transcript is not differentially used; note that this does not guarantee FDR control. Genes and transcripts with less than 20 and 10 estimated counts (across all samples), respectively, are excluded from Figures and Tables.

Figures 1 and 2 report the true positive rate (TPR) vs. FDR curves of all methods for gene and transcript-level tests, respectively. Note that fewer methods are displayed in transcript-level plots, because not all tools perform a transcript-level test. To facilitate graphical interpretation, for each method, we only report three dots corresponding to the observed FDR at 0.01, 0.05 and 0.1 thresholds; the full curves are available in Supplementary Figures S1 and S2. BANDITS exhibits highly favourable performance in all scenarios. In both, unfiltered and filtered, 6 vs. 6 simulation studies, BANDITS and its conservative scores (BANDITS_inv or BANDITS_maxGene) have the highest curves, while they are only second to SUPPA2 in the 3 vs. 3 simulated data. Furthermore, in all cases, BANDITS provides good control of the FDR, particularly for the 0.05 and 0.1 thresholds, while most methods show a significant deviation from these cut-offs. Compared to the original BANDITS tests, the conservative scores, BANDITS_inv and BANDITS_maxGene, provide a better FDR control without lowering the overall curve. Note that in the 3 vs. 3 simulation, BANDITS_inv, BayesDRIMSeq_inv and cjBitSeq_inv scores are favoured by the fact that DTU genes are simulated by inverting the two most expressed transcripts, hence the dominant transcript always changes between conditions in DU genes.

Supplementary Figures S3 and S4 compare results obtained by BANDITS, in both 3 vs. 3 and 6 vs. 6 simulated data, on the original data and when filtering lowly abundant transcripts: in both cases, and particularly in the 3 vs. 3 simulation, transcript pre-filtering leads to an improvement of gene and transcript-level testing.

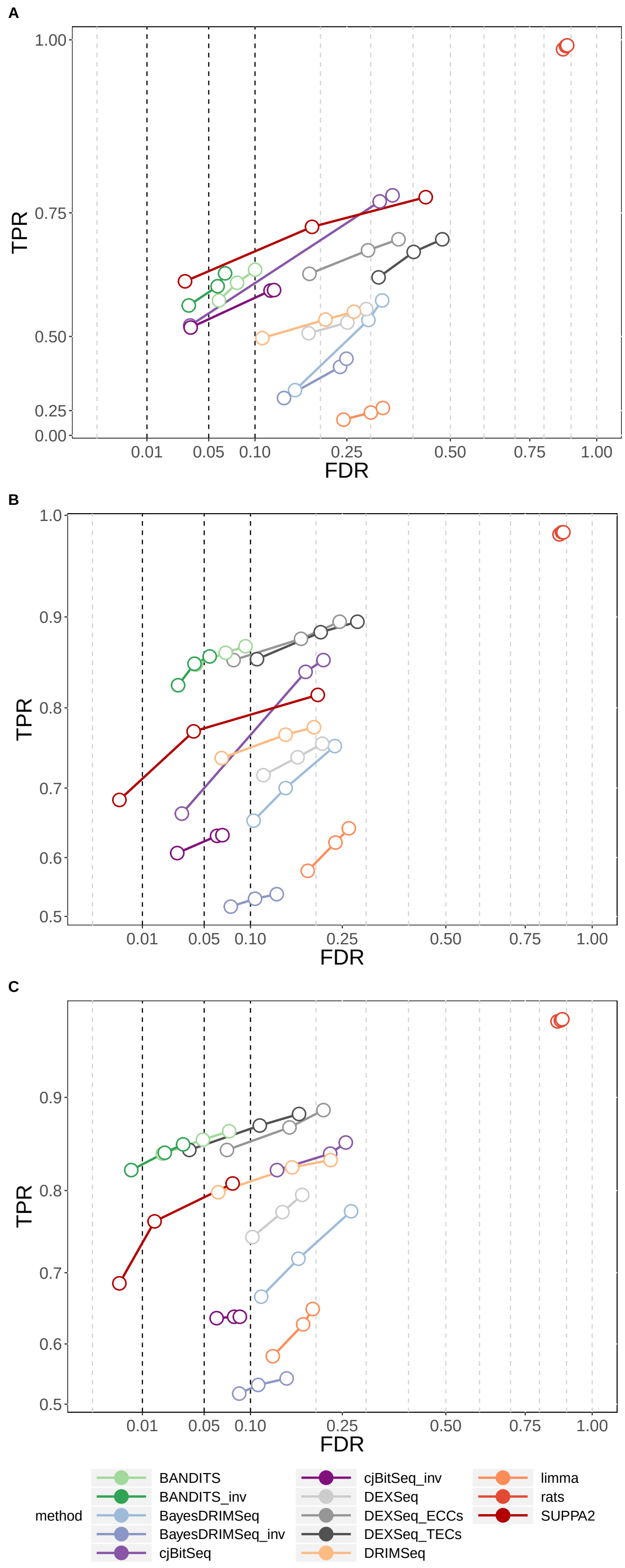

Figure 1 TPR vs. FDR for gene-level testing. A) 3 vs. 3 simulation study; B) 6 vs. 6 simulation study; C) 6 vs. 6 simulation study with transcript pre-filtering. Circles indicate observed FDR for 0.01, 0.05 and 0.1 significance thresholds.

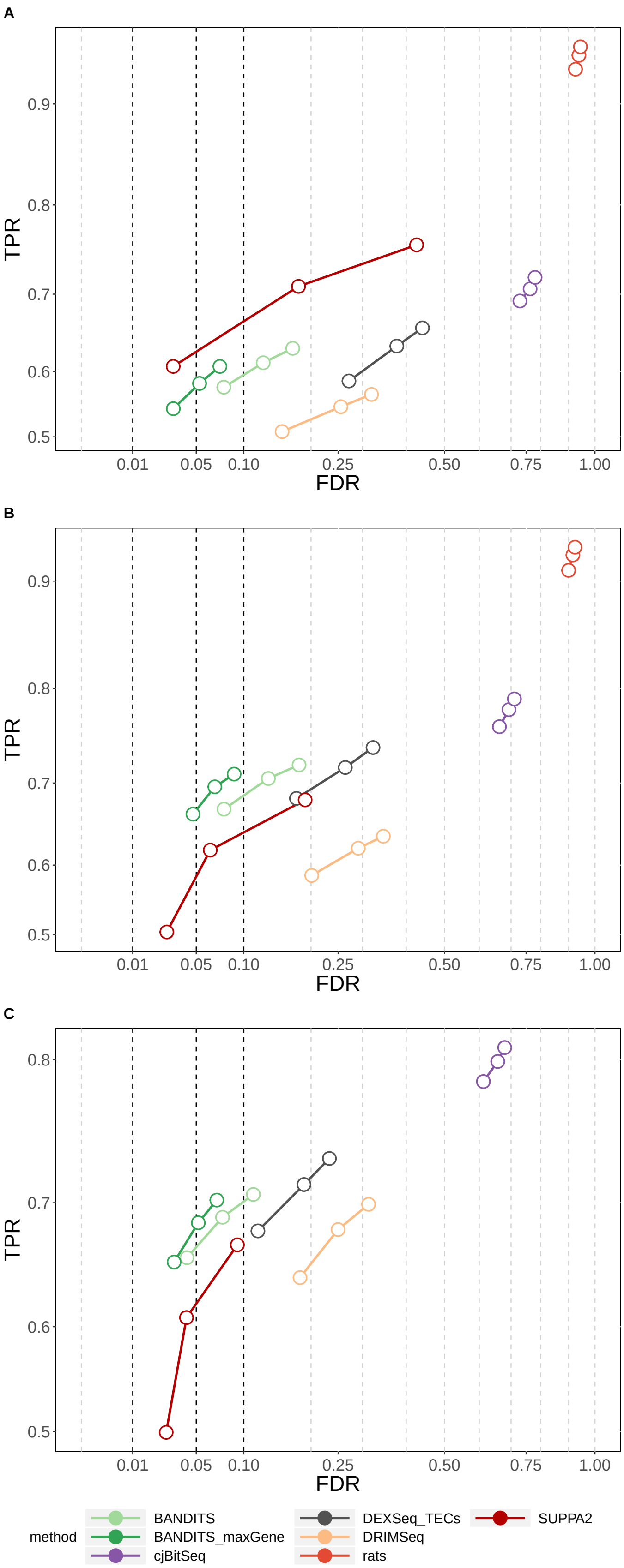

Figure 2 TPR vs. FDR for transcript-level testing. A) 3 vs. 3 simulation study; B) 6 vs. 6 simulation study; C) 6 vs. 6 simulation study with transcript pre-filtering. Circles indicate observed FDR for 0.01, 0.05 and 0.1 significance thresholds.

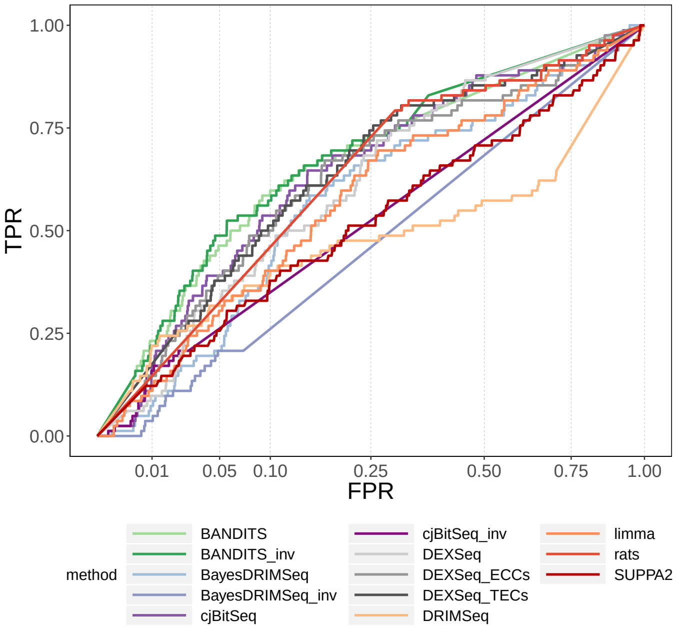

Figure 3 ROC curve (TPR vs. FPR) for gene-level testing in the "Best et al." experimental dataset.

Experimental data analyses

We also applied the previous DTU models to two RNA-seq experimental datasets. First, we studied the human data from Best et al. [9, 33], consisting of a two group comparison with 3 samples in each group, where 83 splicing events, corresponding to 82 genes, were validated via reverse transcriptase polymerase chain reaction (RT-PCR). We restricted our study to the most 10,000 expressed genes (given Salmon estimated counts), which include all 82 validated genes. We will refer to this database as "Best et al.".

Figure 3 shows the receiving operating characteristic (ROC) curves of all methods considered for gene-level testing, while Table 1 reports the area under the curve (AUC), the partial AUC of levels 0.1 and 0.2, and the median position of the 82 validated genes in the raking of 10,000 analyzed genes. BANDITS has again very favourable performance: BANDITS and BANDITS_inv provide the two lowest median rankings for the validated genes, as well as the highest (overall and partial) AUCs, and the highest TPR curves for false positive rate (FPR) between 0 and 0.25.

We further considered a second human experimental dataset [34]. Here, we performed a "null" analysis to investigate FPRs, by comparing two groups of 3 healthy patients each. Again, we only considered genes with at least 20 estimated counts across all samples.

	Median position	AUC	pAUC 0.1	pAUC 0.2
BANDITS_inv	596.00	0.81	0.04	0.11
BANDITS	672.75	0.80	0.04	0.11
cjBitSeq	900.00	0.79	0.04	0.10
rats	942.50	0.80	0.03	0.10
DEXSeq_TECs	968.00	0.79	0.03	0.09
DEXSeq_ECCs	1039.00	0.78	0.03	0.10
BayesDRIMSeq	1231.00	0.74	0.02	0.08
DEXSeq	1348.00	0.78	0.03	0.08
limma	1556.00	0.74	0.03	0.08
SUPPA2	2109.75	0.67	0.02	0.07
DRIMSeq	3248.00	0.59	0.03	0.07
cjBitSeq_inv	5146.50	0.59	0.02	0.05
BayesDRIMSeq_inv	5362.00	0.57	0.02	0.04

Table 1 Results from the "Best et al." experimental dataset; methods are sorted by lowest "Median position". "Median position" indicates the median position of the 83 validated genes in the ranking of 10,000 analyzed genes; AUC refers to the area under the ROC curve; pAUC 0.1 and 0.2 indicate the partial AUC of levels 0.1 and 0.2, respectively.

Figure 4 shows the gene-level test FPR vs. FDR curves of each method. Supplementary Figures S7 and S8 report the same analysis for both gene and transcript-level tests, when considering raw and adjusted p-values, while Supplementary Table S4 displays the FPRs obtained at the 0.05 threshold. Overall limma, BANDITS, BANDITS_inv, DRIMSeq and DEXSeq display the lowest FPRs at the gene level; BANDITS BANDITS_maxGene and DRIMSeq also lead to the lowest FPRs when considering transcript-level tests. Instead, rats, DEXSeq_ECCs and DEXSeq_TECs provide the worst control of FPs in gene-level tests, particularly for 0.01 and 0.05 thresholds, while rats has the highest number of false positives when testing transcript.

Computational benchmark

We performed a computational comparison of all the methods considered in the 6 vs. 6 simulation study, with and without transcript pre-filtering. Analyses were run on 12 cores, when parallelization was allowed, on our Opteron 6100 server.

Figure 5 and Supplementary Tables S6 and S7 illustrate the computational cost of each method. In our benchmark, cjBitSeq stands out as the most computationally intensive tool, both in the alignment (via Bowtie2) and differential components, followed by DEXSeq and limma, mostly due to the python *dexseq_count.py* function which translates the genomic alignments of reads into exon bin counts. On the opposite side DEXSeq_TECs and DRIMSeq, which use transcript estimated counts, are the fastest methods to run. Overall, BANDITS is significantly faster than cjBitSeq, DEXSeq and limma, but slower than DEXSeq_ECCs and than tools using TECs; nonetheless, BANDITS has a 3 time speed-up when pre-filtering transcripts, bringing

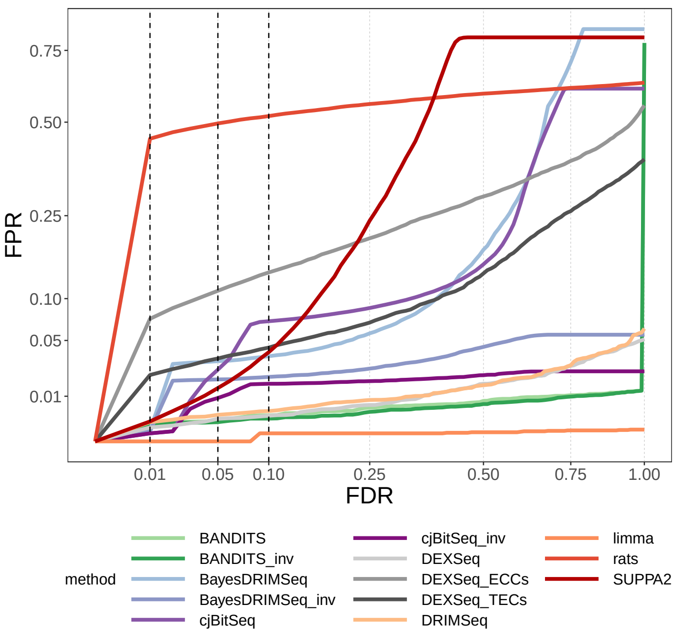

Figure 4 FPR vs. FDR for gene-level testing in the null experimental dataset.

it close to DEXSeq_ECCs. Considering this significant computational gain, and the improved performance obtained when pre-filtering transcripts (Supplementary Figures S3 and S4), we highly encourage users to filter lowly abundant transcripts, which can be done automatically in BANDITS via the *filter_transcripts* function. Furthermore, we found that BANDITS scales well when increasing sample size: it required 43.5 minutes when using 2 samples per group, 50.5 with 3 and 58.8 with 6 (details in Methods).

Note that, except cjBitSeq, DEXSeq and limma, the cost of alignment (via STAR) and quantification (via Salmon) is much higher than the cost of the differential analyses, making the overall cost of the full pipelines of these methods similar.

Stratification by expression level

To investigate how method performance is influenced by gene abundance, we stratified the results of the 6 vs. 6 simulation study, and of both experimental data analyses according to gene expression, by grouping genes into lowly (first tertile), medium (second tertile) and highly expressed (third tertile) .

In the simulation study (Supplementary Figure S5), the ordering of methods is roughly unaltered, while medium and highly expressed genes have a general better FDR control compared to lowly abundant ones. In the Best et al. data analysis (Supplementary Figure S6 and Table

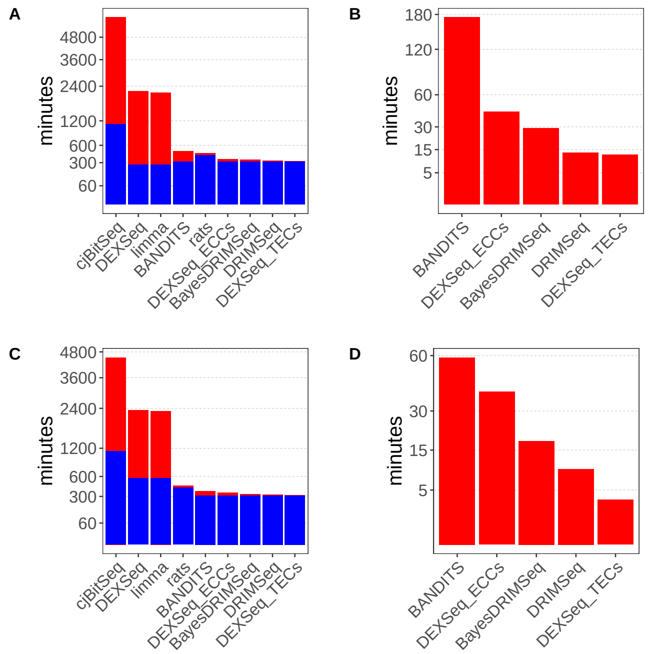

Figure 5 Computational benchmark in the 6 vs 6 simulation study. the (computational) cost of alignment and quantification (when required) is shown in blue; the cost of the differential analyses is shown in red. A) Full pipeline; B) differential analyses after running STAR and Salmon; C) full pipeline, when filtering the transcriptome; D) differential analyses after running STAR and Salmon, when filtering the transcriptome. cjBitSeq, DEXSeq, limma and rats are excluded from B) and D) because they require a distinct alignment pipeline. Details in Supplementary Section S1.4.

S3), medium and highly expressed genes tend to have a better ranking (e.g., median position of validated genes) compared to lowly abundant ones, but no method outperforms the others in all three cases. Finally, the null data analysis (Supplementary Figure S9 and Table S5) shows that more genes are erroneously detected as their expression increases; in particular, rats and DEXSeq_ECCs show worrying FPRs of 82.65% and 29.73%, respectively, for highly expressed genes, given an FDR significance threshold of 0.05. BANDITS and BANDITS_inv, instead, provide among the lowest false detections in any group of genes, with FPRs ranging between 0.05% and 0.42%.

Discussion

In this manuscript, we have introduced a method to perform differential splicing based on RNA-seq data. BANDITS uses a Bayesian hierarchical structure to model the variability between samples, and treats the transcript (and gene) allocations of reads as latent variables; model parameters and latent variables are sampled via MCMC techniques. We designed benchmarks, based on three simulation studies and two experimental data analyses, where we compared BANDITS against the most popular methods for differential splicing. Results highlight BANDITS strong performance, and provide a comprehensive guide for users interested in choosing a tool to investigate DS.

A limitation in common to all methods considered, is to rely on an annotated transcriptome (and genome, for genome alignment), which may lead to inaccurate inference in case of misannotated transcripts and genes [35]; this phenomenon might be particularly present for disease samples, whose condition might lead to the development of unannotated transcripts or genes (e.g., gene fusions). Therefore, all DS methods considered here would benefit from the development of tools that enhance the annotated transcriptome based on the available data, hence accounting for the particular features of the samples considered. Furthermore, BANDITS targets splicing genes and transcripts, but does not identify specific splicing events. Some tools, most notably SUPPA2, target local splicing events (e.g., intron retention or exon skipping), usually based on percent-spliced-in. However, such an approach typically leads to lower power than jointly considering all reads available for a gene. A further limitation of BANDITS is that it does not allow for covariates; to overcome this issue, we introduced a regression structure in our model to incorporate covariates, such as batches. However, when adding batch effects to our simulation studies, even in extreme scenarios, the original version of BANDITS outperformed, in terms of power and FDR, the modified version allowing for covariates (data not shown). Moreover, we noticed that BANDITS was very robust to batch effects, which only marginally altered its performance. This suggests that the misspecification of the model (i.e., ignoring batches when present) might be less deleterious than having a more complex modelling structure, involving more parameters. Therefore, we choose not to include this modification in the final version of BANDITS.

Finally, we note that BANDITS, although developed with a focus on RNA-seq data, can also be applied to long-read sequencing data. Soneson et al. (2019) [35] found that Illumina RNA-seq reads and Oxford Nanopore Technologies long reads generated equivalence classes with almost equivalent average number of transcripts. Hence, one might expect at least the current generation of long read transcriptome data to also benefit from BANDITS transcript latent variable allocation approach.

Conclusions

We presented BANDITS, a novel Bayesian method to investigate differential splicing from RNA-seq data. At present, our tool is the only method that jointly models the variability between biological replicates, by allowing for sample-specific proportions, and the mapping uncertainty of reads, by sampling their transcript (and gene) allocations. BANDITS is also the first DS tool to correct for the transcript effective lengths, allowing it to recover the actual probability of expressing a transcript. Our method tests, both, genes and transcripts for DS, and allows comparisons between more than two groups. We also introduce a measure of the DTU strength, which can be used as an alternative way to rank genes.

In all simulation and experimental datasets analyzed, BANDITS has extremely favourable performance and exhibits good FDR and excellent FPR control. Furthermore, despite requiring full MCMC inference, it is computationally competitive, particularly after applying reasonable expression level filters.

Finally, BANDITS is released as a R/Bioconductor package, which makes it easy to update, distribute and integrate within existing data analysis pipelines.

Methods

Prior distributions

Since the Dirichlet parameters $\delta_1, ..., \delta_K$ are positive, we sample them and formulate their prior in the logarithmic scale, a common choice to improve mixing of positive parameters.

If gene-wise precision parameters are not computed (via *prior_precision* function), we specify a vaguely informative prior distribution for the logarithm of the Dirichlet parameters: $log(\delta_k) \sim \mathcal{N}(\mu = 0, \sigma^2 = 100), k = 1, \dots, K$.

Instead, if gene-wise precision parameters are available, we compute the mean and variance of their logarithm, $\bar{x}_{\delta_+}$ and $s^2_{\delta_+}$, and formulate an informative prior for $log(\delta_+)$ as: $log(\delta_+) \sim \mathcal{N}\left(\mu = \bar{x}_{\delta_+}, \sigma^2 = s^2_{\delta_+}\right)$. The remaining $K-1$ Dirichlet parameters *a priori* are distributed as follows:
$log(\delta_k) \sim \mathcal{N}\left(\mu = \bar{x}_{\delta_+} - log(K), \sigma^2 = 100\right)$, for $k = 1, \dots, K-1$, which corresponds to a vaguely informative prior; setting $\mu = \bar{x}_{\delta_+} - log(K)$ instead of 0, corresponds to assuming that, *a priori*, δ_+ is equally distributed across the K transcripts. In order to obtain the prior distribution for $log(\delta_K)$ we apply the change of variable via the Jacobian transformation ([37]).

Latent variables allocation

We define the set of J equivalence classes available for a given gene as $C = (C_1, \dots, C_J)$, where C_j indicates the list of transcripts present in the j-th equivalence class. Note that ECs not supported by any read are not included in C. The number of reads compatible with C_j in the i-th sample is denoted by $f_j^{(i)}$, $j = 1, \dots, J$ and $i = 1, \dots, N$. For ECs with at least two transcripts, reads in $f_j^{(i)}$ need to be allocated to the transcripts in C_j. We introduce the vector $X_{.j}^{(i)} = \left(X_{1j}^{(i)}, \dots, X_{Kj}^{(i)} \right)$, where $X_{kj}^{(i)}$ indicates the number of reads from the j-th EC that were generated from the k-th transcript in the i-th sample, with $j = 1, \dots, J$, $k = 1, \dots, K$ and $i = 1, \dots, N$. Note that $\sum_{k=1}^{K} X_{kj}^{(i)} = f_j^{(i)}$ and $X_{kj}^{(i)} = 0 \ \forall k \notin C_j$.

Clearly, $X_{.j}^{(i)}$ cannot be observed directly; it is hence treated as a latent variable which, under the assumption of uniform coverage, is sampled from the following density:

$$X_{.j}^{(i)} \Big| \pi^{T(i)} \sim \mathcal{MN}\left(f_j^{(i)}, \pi_{.j}^{T(i)} \right), \tag{6}$$

where $\pi_{.j}^{T(i)} = \left(\pi_{1j}^{T(i)}, \dots, \pi_{Kj}^{T(i)} \right)$, with
$\pi_{kj}^{T(i)} = \dfrac{\mathbb{1}\left(k \in C_j\right) \pi_k^{T(i)}}{\sum_{k'=1}^{K} \mathbb{1}\left(k' \in C_j\right) \pi_{k'j}^{T(i)}}$, where $\mathbb{1}(\mathrm{a})$ is 1 if a is true, and 0 otherwise. Intuitively, $\pi_{.j}^{T(i)}$ modifies $\pi^{T(i)}$ to ensure that reads are only allocated to the transcripts in C_j.

Once EC reads have been allocated to the respective transcripts, we can compute the corresponding counts for the k-th transcript by adding counts across ECs: $X_k^{(i)} = \sum_{j=1}^{J} X_{kj}^{(i)}$, $k = 1, \dots, K$ and $i = 1, \dots, N$.

If an equivalence class has transcripts from more than one gene, the probability vector $\pi_{.j}^{T(i)}$ is modified to include all transcripts from the genes in the EC, and the transcript-level probabilities are weighted by the number of reads associated to each gene (details in Supplementary Section S1.2).

Convergence diagnostic

BANDITS users can specify an initial number of iterations to discard as *burn-in* (minimum $2,000$), as well as the number of iterations the MCMC is run for after the initial *burn-in* (minimum $10,000$).

To ensure the posterior chains have reached convergence, after discarding the pre-specified *burn-in*, BANDITS performs Heidelberger and Welch (HW) stationarity test [28] on the marginal log-posterior of the hyper-parameters, i.e., $log(P(\delta|\pi)) \propto log(P(\pi|\delta)) + log(P(\delta))$; by adding the log-posterior densities from all groups, and performing a global convergence diagnostic test. A wider *burn-in* is removed, if estimated via HW test; moreover, if HW stationarity test is rejected at the 0.01 significance threshold, the full MCMC output is discarded and the algorithm is run again (up to three times).

Furthermore, when a gene-level test has a p-value below 0.1, BANDITS runs a second MCMC chain and, after removing the *burn-in*, recomputes the outputs based on the aggregation of the two chains.

DTU test

For every gene, we test the system of hypothesis (5): since the K equations are linearly dependent, we only need to test $K-1$ parameters; hence, we rewrite the system of hypothesis as:

$$\begin{cases} \mathcal{H}_0: & \omega_k = 0, \text{ for } k \in \{1,\dots,K\} \backslash \{k'\}, \\ \mathcal{H}_1: & \text{otherwise}, \end{cases} \tag{7}$$

where $k' \in \{1,\dots,K\}$ is the transcript that should be removed from the test. The null distribution of $\omega_{-k'} = (\omega_1, \dots, \omega_{k'-1}, \omega_{k'+1}, \dots \omega_K)$ is approximately normal [29], with mean $\hat{\omega}_{-k'}$ and covariance matrix $\hat{\Sigma}_{\hat{\omega}_{-k'}}$, both inferred from the posterior chains. This leads to a multivariate Wald test [30] based on the null distribution of $\hat{\omega}_{-k'} \hat{\Sigma}^{-1}_{\hat{\omega}_{-k'}} \hat{\omega}^T_{-k'} \dot{\sim} \chi^2_{K-1}$, where χ^2_a denotes the chi-square random variable with a degrees of freedom, and b^T and b^{-1} indicate the transpose and inverse of b, respectively. In order to choose the transcript to remove from the test, k', we considered several options: randomly drawing one of the K transcripts, the transcript with the smallest expression, the isoform with the smallest difference between conditions, and averaging the p-values obtained from all K possible choices of k'. After benchmarking all four approaches, we choose the last one, because in our simulation studies it provided the highest sensitivity and best FDR control (data not shown).

Similarly, we test for differential usage in individual isoforms, by considering the system of hypothesis for the k-th transcript: $\mathcal{H}_0 : \omega_k = 0$ vs. $\mathcal{H}_1 : \omega_k \neq 0$. In this case we use a univariate Wald test based on the statistic $\hat{\omega}_k \ \hat{\sigma}^{-2}_{\hat{\omega}_k} \ \hat{\omega}^T_k \dot{\sim} \chi^2_1$, where $\hat{\sigma}^2_{\hat{\omega}_k}$ is the estimated marginal variance of ω_k, inferred from the posterior chains.

Supplementary Section S1.3 shows how to extend this scenario when comparing 3 or more experimental conditions.

Conservative scores and DTU measure

We propose two conservative scores for gene and transcript-level testing. The former is inspired by work from Papastamoulis and Rattray (2017) [14], where the authors propose to filter *a* *posteriori* all genes whose estimated dominant transcript (i.e., the most expressed transcript) is unchanged between conditions, leading to scores BayesDRIMSeq_inv and cjBitSeq_inv. However, excluding all such genes, regardless of their significance, is an excessive filter in our opinion because genes might exhibit DS while preserving their dominant transcript. Here, when testing genes in two group comparisons, we introduce a moderated version of that score, that we call BANDITS_inv: we propose to inflate the adjusted p-value, defined as $\tilde{p}$, by taking its square root when the dominant transcript is unchanged between conditions. If the dominant transcript is estimated to change between conditions (according to the posterior mode of $\bar{\pi}^T$), then BANDITS_inv $= \tilde{p}$, otherwise BANDITS_inv $= \sqrt{\tilde{p}}$. We further propose a conservative transcript score, called BANDITS_maxGene, which takes the maximum between the transcript and gene-level adjusted p-values; in this way, a transcript can only be selected if the corresponding gene is also significant.

Note that, in the Best et al. experimental data analysis, 41% of the validated genes are inferred to have distinct dominant isoforms between conditions, while this value decreases to 17% when considering non-validated genes; this fact seems to empirically justify our intuition of moderating Papastamoulis and Rattray's inversion criterion.

For two group comparisons, we also propose a score, called *DTU_measure*, to measure the intensity of the differential usage change between conditions, similarly to fold changes in differential expression analyses. Given a gene with K transcripts and estimated mean relative transcript abundance $\hat{\bar{\pi}}_1^{TA}, \dots, \hat{\bar{\pi}}_K^{TA}$, for group A, and $\hat{\bar{\pi}}_1^{TB}, \dots, \hat{\bar{\pi}}_K^{TB}$, for group B, *DTU_measure* is defined as the summation of the absolute difference between the two most expressed transcripts: $\sum_{k \in \tilde{K}} \left| \hat{\bar{\pi}}_k^{TA} - \hat{\bar{\pi}}_k^{TB} \right|$, where $\tilde{K}$ indicates the set of two most expressed transcripts across both groups. This measure ranges between 0, when proportions are identical between groups, and 2, when an isoform is always expressed in group A and a different transcript is always chosen in group B.

Scalability

We performed a computational benchmark of BANDITS, based on the 6 vs. 6 simulation, to investigate how computational times scale with respect to the sample size. We selected 2 and 3 samples per group and ran BANDITS on a 2 vs. 2 and 3 vs. 3 group comparison. In all cases, 12 cores from our Opteron 6100 based server were used, and the same transcripts were pre-filtered, based on the transcripts selected from the 6 vs. 6 analysis.

The computational cost scales less than linearly as the sample size increases: BANDITS took 43.5 minutes when using 2 samples per group, 50.5 with 3 and 58.8 with 6.

Author details

[1]Institute of Molecular Life Sciences, University of Zurich, Winterthurerstrasse 190, 8057, Zurich, Switzerland. [2]SIB Swiss Institute of Bioinformatics, 8057, Zurich, Switzerland.

References

1. Gonzàlez-Porta, M., Frankish, A., Rung, J., Harrow, J., Brazma, A.: Transcriptome analysis of human tissues and cell lines reveals one dominant transcript per gene. Genome biology **14**(7), 70 (2013)
2. Lee, Y., Rio, D.C.: Mechanisms and regulation of alternative pre-mRNA splicing. Annual review of biochemistry **84**, 291–323 (2015)
3. Cooper, T.A., Wan, L., Dreyfuss, G.: RNA and disease. Cell **136**(4), 777–793 (2009)
4. Padgett, R.A.: New connections between splicing and human disease. Trends in Genetics **28**(4), 147–154 (2012)
5. Van den Berge, K., Hembach, K.M., Soneson, C., Tiberi, S., Clement, L., Love, M.I., Patro, R., Robinson, M.D.: RNA sequencing data: Hitchhiker's guide to expression analysis. Annual Review of Biomedical Data Science **2**(1), 139–173 (2019)
6. Anders, S., Reyes, A., Huber, W.: Detecting differential usage of exons from RNA-seq data. Genome research **22**(10), 2008–2017 (2012)
7. Wang, E.T., Sandberg, R., Luo, S., Khrebtukova, I., Zhang, L., Mayr, C., Kingsmore, S.F., Schroth, G.P., Burge, C.B.: Alternative isoform regulation in human tissue transcriptomes. Nature **456**(7221), 470 (2008)
8. Venables, J.P., Klinck, R., Bramard, A., Inkel, L., Dufresne-Martin, G., Koh, C., Gervais-Bird, J., Lapointe, E., Froehlich, U., Durand, M., *et al.*: Identification of alternative splicing markers for breast cancer. Cancer research **68**(22), 9525–9531 (2008)
9. Trincado, J.L., Entizne, J.C., Hysenaj, G., Singh, B., Skalic, M., Elliott, D.J., Eyras, E.: SUPPA2: fast, accurate, and uncertainty-aware differential splicing analysis across multiple conditions. Genome biology **19**(1), 40 (2018)
10. Li, B., Dewey, C.N.: RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. BMC bioinformatics **12**(1), 323 (2011)
11. Patro, R., Duggal, G., Love, M.I., Irizarry, R.A., Kingsford, C.: Salmon provides fast and bias-aware quantification of transcript expression. Nature methods **14**(4), 417 (2017)
12. Bray, N.L., Pimentel, H., Melsted, P., Pachter, L.: Near-optimal probabilistic RNA-seq quantification. Nature biotechnology **34**(5), 525 (2016)
13. Nowicka, M., Robinson, M.D.: DRIMSeq: a Dirichlet-multinomial framework for multivariate count outcomes in genomics. F1000Research **5**(1356) (2016)
14. Papastamoulis, P., Rattray, M.: Bayesian estimation of differential transcript usage from RNA-seq data. Statistical applications in genetics and molecular biology **16**(5-6), 387–405 (2017)
15. Froussios, K., Mourão, K., Simpson, G., Barton, G., Schurch, N.: Relative Abundance of Transcripts (RATs): Identifying differential isoform abundance from RNA-seq. F1000Research **8** (2019)
16. Ritchie, M.E., Phipson, B., Wu, D., Hu, Y., Law, C.W., Shi, W., Smyth, G.K.: limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic acids research **43**(7), 47–47 (2015)
17. Love, M.I., Soneson, C., Patro, R.: Swimming downstream: statistical analysis of differential transcript usage following Salmon quantification. F1000Research **7** (2018)
18. Cmero, M., Davidson, N.M., Oshlack, A.: Using equivalence class counts for fast and accurate testing of differential transcript usage. F1000Research **8** (2019)
19. Dobin, A., Davis, C.A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., Gingeras, T.R.: STAR: ultrafast universal RNA-seq aligner. Bioinformatics **29**(1), 15–21 (2013)
20. Gamerman, D., Lopes, H.F.: Markov Chain Monte Carlo Stochastic Simulation for Bayesian Inference, 2nd Ed. Chapman & Hall/CRC, Boca Raton, London, New York (2006)
21. Tiberi, S., Walsh, M., Cavallaro, M., Hebenstreit, D., Finkenstädt, B.: Bayesian inference on stochastic gene transcription from flow cytometry data. Bioinformatics **34**(17), 647–655 (2018)
22. Hastings, W.K.: Monte Carlo sampling methods using Markov chains and their applications. Biometrika **57**, 97–109 (1970)
23. Metropolis, N., Ulam, S.: The Monte Carlo method. Journal of the American Statistical Association **44**, 335–341 (1949)
24. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. The Journal of Chemical Physics **21**, 1087–1092 (1953)
25. Haario, H., Saksman, E., Tamminen, J.: An adaptive Metropolis algorithm. Bernoulli **7**, 223–242 (2001)
26. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence **6**, 721–741 (1984)
27. Gelfand, A.E., Smith, A.F.: Sampling-based approaches to calculating marginal densities. Journal of the American statistical

association **85**(410), 398–409 (1990)

28. Heidelberger, P., Welch, P.D.: Simulation run length control in the presence of an initial transient. Operations Research **31**(6), 1109–1144 (1983)
29. Gelman, A., Stern, H.S., Carlin, J.B., Dunson, D.B., Vehtari, A., Rubin, D.B.: Bayesian Data Analysis. Chapman and Hall/CRC, New York (2013)
30. Li, K.-H., Raghunathan, T.E., Rubin, D.B.: Large-sample significance levels from multiply imputed data using moment-based statistics and an F reference distribution. Journal of the American Statistical Association **86**(416), 1065–1073 (1991)
31. Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society: series B (Methodological) **57**(1), 289–300 (1995)
32. Soneson, C., Matthes, K.L., Nowicka, M., Law, C.W., Robinson, M.D.: Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage. Genome biology **17**(1), 12 (2016)
33. Best, A., James, K., Dalgliesh, C., Hong, E., Kheirolahi-Kouhestani, M., Curk, T., Xu, Y., Danilenko, M., Hussain, R., Keavney, B., *et al.*: Human Tra2 proteins jointly control a CHEK1 splicing switch among alternative and constitutive target exons. Nature communications **5**, 4760 (2014)
34. Kim, S.C., Jung, Y., Park, J., Cho, S., Seo, C., Kim, J., Kim, P., Park, J., Seo, J., Kim, J., *et al.*: A high-dimensional, deep-sequencing study of lung adenocarcinoma in female never-smokers. PloS one **8**(2), 55596 (2013)
35. Soneson, C., Yao, Y., Bratus-Neuenschwander, A., Patrignani, A., Robinson, M.D., Hussain, S.: A comprehensive examination of nanopore native rna sequencing for characterization of complex transcriptomes. Nature communications **10**, 3359 (2019)
36. Crowell, H.L., Soneson, C., Germain, P.-L., Calini, D., Collin, L., Raposo, C., Malhotra, D., Robinson, M.: On the discovery of population-specific state transitions from multi-sample multi-condition single-cell RNA sequencing data. BioRxiv, 713412 (2019)
37. Murphy, K.P.: Machine Learning: a Probabilistic Perspective. MIT press, Cambridge, Massachusetts (2012)

Additional Files

The supplementary file contains further details about the method and analyses, as well as additional Tables and Figures.

Availability of data and material

BANDITS is available on the Bioconductor site (https://bioconductor.org/packages/BANDITS) and on GitHub (https://github.com/SimoneTiberi/BANDITS).

For the 6 vs. 6 simulated data we designed, the fastq files for the 12 samples, TECs, ECCs and truth table are available at FigShare (DOI *10.6084/m9.figshare.9467144*, *10.6084/m9.figshare.9692429* and *10.6084/m9.figshare.9692918*).

The code for simulating the RNA-seq reads and to perform all analyses is made available on GitHub (https://github.com/SimoneTiberi/BANDITS_manuscript).

Competing interests

The authors declare that they have no competing interests.

Author's contributions

ST conceived the method, implemented it and performed the analyses. ST and MDR designed the study and wrote the manuscript. All authors read and approved the final article.

Acknowledgements

The authors would like to thank Charlotte Soneson, Panagiotis Papastamoulis, Magnus Rattray, David Rossell and all members of the Robinson Lab (and particularly Helena Lucia Crowell) for their helpful comments, suggestions and contributions to this work.

Consent for publication

Not applicable.

Funding

This work was supported by the Swiss National Science Foundation (grants 310030_175841, CRSII5_177208) as well as the Chan Zuckerberg Initiative DAF (grant number 2018-182828), an advised fund of Silicon Valley Community Foundation. MDR acknowledges support from the University Research Priority Program Evolution in Action at the University of Zurich.