Journal Title Here, 2021, pp. 1–9

doi: DOI HERE
Advance Access Publication Date: Day Month Year
Paper

Subject Section

PAPER

Baltica: integrated splice junction usage analysis

Thiago Britto-Borges,[1,2,*] Volker Boehm,[3,4] Niels H. Gehring[3,4] and Christoph Dieterich[1,2,*]

[1]Section of Bioinformatics and Systems Cardiology, Klaus Tschira Institute for Integrative Computational Cardiology, University Hospital Heidelberg, 69120, Heildeberg, Germany, [2]Department of Internal Medicine III (Cardiology, Angiology, and Pneumology), University Hospital Heidelberg, 69120, State, Heidelberg, Germany, [3]Institute for Genetics, University of Cologne, 50674, Cologne, Germany and [4]Center of Molecular Medicine Cologne (CMMC)), University of Cologne, 50937, Cologne, Germany
[*]Corresponding author.
FOR PUBLISHER ONLY Received on Date Month Year; revised on Date Month Year; accepted on Date Month Year

Abstract

Alternative splicing is a tightly regulated co- and post-transcriptional process contributing to the transcriptome diversity observed in eukaryotes. Several methods for detecting differential junction usage (DJU) from RNA sequencing (RNA-seq) datasets exist. Yet, efforts to integrate the results from DJU methods are lacking. Here, we present Baltica, a framework that provides workflows for quality control, *de novo* transcriptome assembly with `StringTie2`, and currently 4 DJU methods: `rMATS`, `JunctionSeq`, `Majiq`, and `LeafCutter`. Baltica puts the results from different DJU methods into context by integrating the results at the junction level. We present Baltica using 2 datasets, one containing known artificial transcripts (SIRVs) and the second dataset of paired Illumina and Oxford Nanopore Technologies RNA-seq. The data integration allows the user to compare the performance of the tools and reveals that `JunctionSeq` outperforms the other methods, in terms of F1 score, for both datasets. Finally, we demonstrate for the first time that meta-classifiers trained on scores of multiple methods outperform classifiers trained on scores of a single method, emphasizing the application of our data integration approach for differential splicing identification. Baltica is available at https://github.com/dieterich-lab/Baltica under MIT license.

Key words: RNA-Seq, differential splicing, workflows, data integration, reproducibility

Introduction

Alternative promoters, splice sites, and polyadenylation sites define the transcriptome complexity by producing different transcript isoforms. Alternative splicing (AS), defined as the differential removal of introns by alternative splice site usage instead of canonical splice sites, is widespread in eukaryotic genomes. AS regulation is central to physiological processes, such as tissue remodeling[1], and defective splicing has been linked to human disease[2]. However, most of the cataloged AS events are yet to be associated with their functional consequence[3]. Furthermore, there are increasing numbers of genomic single nucleotide variants associated with mis-splicing events[4], pointing to a latent link between multi-factorial diseases and mis-splicing due to genetic variation. AS is regulated by context-dependent proteins named splicing regulatory factors, which define splice sites and lead transcript isoforms changes. Altogether, AS and its regulation are crucial to studying human health and disease. Computational methods for AS identification from RNA sequencing (RNA-seq) have helped to scale up these discoveries.

There are different approaches to identifying splicing events from RNA-seq. Methods that model intron usage are popular methods, as shown in Supplementary Figure S1. In addition, these methods have been applied to a broad range of studies, for example, the effects of genetic variation in

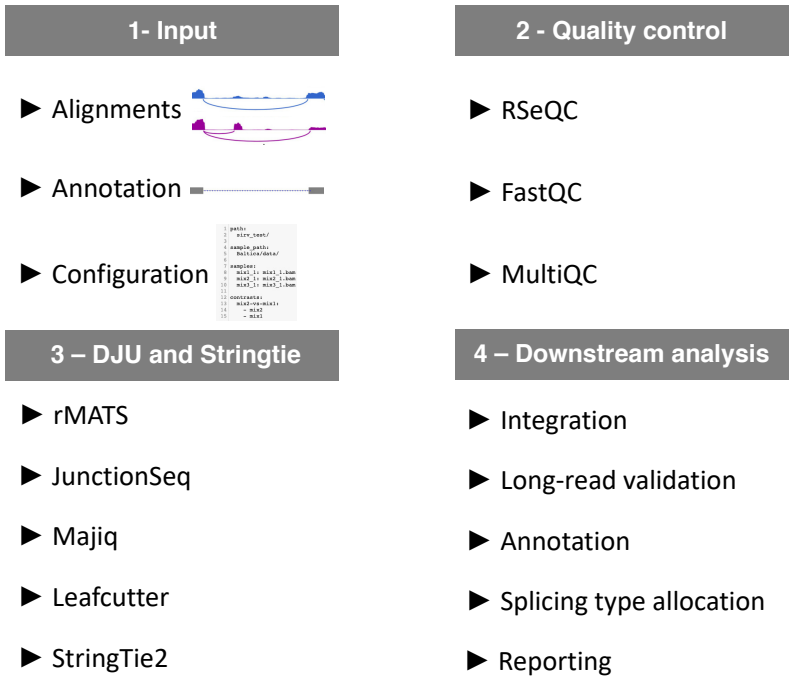


Fig. 1. Baltica framework overview. Baltica is a framework to execute and integrate differential junction usage (DJU) analysis. **1 – Input:** Baltica takes as input RNA-seq alignments, reference annotation, and a configuration file. **2 – Quality control:** First, Baltica performs quality control of alignments with `RSeQC` and `FastQC`, which is reported by `MultiQC`. **3 – DJU and StringTie**: Next, Baltica computes DJU with `rMATS`, `JunctionSeq`, `Majiq`, and `LeafCutter`, and uses `StringTie` to detected transcripts and exons, which deviate from the reference annotation. **4 – Downstream analysis:** Finally, we integrate the results from the DJU method. Optionally, Baltica can include an extra piece of evidence for DJU (hereafter the orthogonal dataset), such as DJU obtained from Oxford Nanopore Technologies (ONT) RNA-seq. The set of introns is re-annotated using information from *de novo* transcriptome annotation, and splice types between SJ and exons are assigned. Finally, Baltica compiles a report with the most relevant information.

splicing[5], identification of splicing factor-mediated AS events [6], associate AS to nonsense-mediated decay[7] and testing for splicing therapeutical intervention in animal models[8]. We here name these methods as different junction usage (DJU) methods. As suggested by Mehmood and collaborators[9], comparing results from multiple methods could improve AS event prioritization. Despite the popularity and critical application to human health, individual DJU methods have limitations.

DJU methods differ in software granularity. While some methods implement multiple functionality steps, from sequencing read filtering to results reporting, others focus solely on statistical modeling of RNA-seq split reads. These differences in implementation and poorly defined concepts are barriers to data integration from DJU method results. Specifically, DJU methods results are not comparable, as not all methods output standard file formats. Second, differences in AS event definition limits the comparison of event-specific metrics. The PSI (percent spliced in; Ψ) represents the proportion of splice site usage within an AS event per experimental group and indicates effect size[10]. In general, methods do not adopt a standardized definition for AS event or Ψ, thus complicating the comparison of effect sizes. Third, methods do not share common steps to facilitate result integration and benchmark. For example, it is not trivial to input the same matrix of splice junctions (SJ) read counts to all DJU methods. Collectively, these points are obstacles to data integration.

In this paper, we present Baltica, a framework that facilitates the execution and enables the integration of DJU methods results. Baltica comprises of a command-line interface, `snakemake`[11] workflows, containers[12], and scripts that provided reports on the integrated results. We propose a protocol to integrate results from DJU methods and further prioritize introns that undergo AS based on the decision of such methods. Optionally, Baltica integrates of results obtained with orthogonal experiments, such as AS evidence from Oxford Nanopore Technologies (ONT) RNA-seq. To our knowledge, there are no others solutions for integrating DJU results. We apply Baltica to 2 datasets. The first uses spike-ins with known experimental group concentration and transcriptome structure, the so-called Spike-in RNA Variant Control Mixes (SIRVs). The second ones uses paired Illumina and ONT RNA-seq datasets. In addition, Baltica integration allows us to compare the performance of different DJU methods and test the usability of a meta-classifier trained on the decision of the methods.

Material and Methods

Baltica method overview

Figure 1 shows an overview of the features included in the Baltica framework. Baltica comprises a command-line interface, workflow implementations, and scripts that handle DJU methods' result parsing, integration, annotation, and reporting. The framework requires `snakemake`[11], and

`singularity`[12]. Singularity containers and Bioconda[13] handle the dependencies for workflows and scripts. The containers allow the execution of software dependencies in isolation and provide reproducible workflows that don't require direct user instalation.

Baltica works as a standard Python package, and its command-line interface facilitates the execution of the workflows by, for example, automatically handling `singularity` arguments. The configuration file centralizes the required information for workflows. Specifically, it contains file paths, file to group assignments, method parameters, and pairwise comparisons between experimental groups to be tested. The required inputs are RNA-seq alignment files in BAM format, a reference transcriptome annotation (GTF/GFF format), and its sequence (FASTA format). Users can also input results from other evidence sources prepared in BED or GFF formats if available.

Baltica implements workflows for quality control methods, DJU methods, *de novo* transcriptome assembly, and downstream analysis. The included methods for quality control are `RSeQC`[14] and `FastQC`[15]. We use `MultiQC`[16] to summarize the output from both tools. In addition to the quality control of reads and alignments, this step helps to identify systematic differences among the analyzed samples or conditions. `RSeQC` implements an SJ saturation diagnostic, which quantifies the abundance of known and novel SJ. The tool also provides the proportion of reads per feature in the input annotation, which may indicate splicing changes due to, for example, an increase of reads mapping to introns.

Currently, the frameworks supports 4 DJU methods: `rMATS`[17], `JunctionSeq`[18], `Majiq`[19], and `LeafCutter`[20]. We detail the method inclusion criteria and workflows at Section 2.2. Finally, the analysis workflow proceeds with DJU integration, annotation, and reporting. Scripts for the analysis workflow were developed with R[21] and based on Bioconductor's infrastructure to handle genomic coordinates[22, 23] as well as tools from the Tidyverse[24].

Differential junction usage algorithms

Due to the high number of DJU methods available in the literature, we have established a set of rules for method inclusion into Baltica. We may include a method if it fits the following criteria:

- supports as input RNA-seq read alignment in the BAM format and transcriptome annotation in the GTF/GFF format
- provides test statistics, such as p-value, at the event or SJ level for pairwise comparisons
- outputs effect size estimates, such as the Ψ
- detects SJ independent of the reference annotation

We present an initial set of 4 DJU methods. These methods fulfill the criteria and are among the most popular methods for differential splicing identification, as shown in Supplementary Figure S1. But are we are aware that other DJU software packages exist, such as SUPPA2[25] and PSI-Sigma[26]. Therefore, we hope to include more of these packages into Baltica, especially with the help of the user community.

`rMATS-turbo`

`rMATS-turbo` (v4.1.1), or simply, `rMATS`, estimates the splicing-type specific isoform proportion from RNA-seq reads. First, `rMATS` uses the reference annotation to determine the splicing events grouped by splicing types: skipped exon, mutually exclusive exons, alternative 3' splice site, alternative 5' splice site, retained intron. More recently, `rMATS`' developers released experimental support for unannotated introns with the '–novelSS' argument. `rMATS` uses the effective length-scaled junction read counts and, optionally, exon read counts to estimate Ψ. Then, it applies the likelihood-ratio to test whether $\Delta\Psi$ ($\Delta\Psi = \Psi_{i1} - \Psi_{i2}$, for the intron i, and groups 1 and 2) surpasses the 0.05 threshold.

`JunctionSeq`

`JunctionSeq` (v1.16) takes as input a read count matrix obtained with `QoRTs`[27] (v1.1.8), for annotated SJ, novel SJ, and exons, so in fact, `JunctionSeq` falls into the differential exon usage and DJU classes. Based on `DEXSeq`, `JunctionSeq` uses disjoint genomic bins as features, and applies a generalized linear model[28] to model the feature expression. Beyond modeling the modeling aspect, `JunctionSeq` also invests in the visualization of the exon and intron usage and builds tracks for genome browsers. `JunctionSeq` does not identify splicing events, so the results are associated with intron coordinates.

`Majiq`

`Majiq` (v2.2-e25c4ac) generates splice graphs for genes present on the RNA-seq dataset and the reference annotation. Next, it detects splicing events, quantifies the SJ usage from normalized SJ read counts, and computes the PSI value for the sample groups. `Majiq` uses a Bayesian framework to assess which $\Delta\Psi$ changes threshold among groups are significant by a user-defined probability. The local splicing variations implementation includes more than 2 SJ per event. So it supports complex AS event types, which is more realistic than modeling splicing events by SJ pairs.

`LeafCutter`

`LeafCutter` (v0.2.7) uses `regtools`[29] to extract and select reads SJ from RNA-seq alignments. Next, it uses an iterative clustering procedure to eliminate SJ with low usage. Finally, the `LeafCutter` fits a Dirichlet-multinomial generalized linear model on SJ usage proportion within intron-clusters.

A more detailed description of the workflow implementation is available at Baltica manual online[30].

Baltica integration and reporting

To parse, integrate and annotate the results from the DJU methods, we use the Bioconductor infrastructure. While parsing the results files from the methods, Baltica pivots the results tables, so each row in the data table corresponds to a single SJ. Because `rMATS` outputs one result file for each AS event type, Baltica selects the SJ representing feature inclusion and exclusion events from each file. Because `LeafCutter` and `rMATS` assign the test statistics to the event instead of the SJ, we assign the same test statistics to multiple SJs contained in the AS event.

One challenge to integrating results from DJU methods is correcting for different coordinate systems. For example, methods can use 0-indexed (BED format) or 1-indexed (GTF format) files and use exon or intron splice site coordinates to represent the SJ genomic position. We make no assumptions regarding the method choice for the coordinate system, and this flexibility allows us to support many methods. To overcome the issue without fixing the coordinates adjusted for each method, we first compute the genomic overlap between introns in the

reference annotation (subject) and a set of SJ output from a method (query). Then, we compute coordinate offsets (in nucleotides) between subject and query, determine the most frequent difference in start and end coordinates between the 2 sets, and finally, apply corrections to the coordinates in a strand-specific manner for each method. This procedure allows Baltica to report groups of SJ that represent splicing events in different genomic coordinate systems.

Next, Baltica uses a *de novo* and guided transcriptome annotation as a reference for annotation and assigning alternative splicing types. The *de novo* workflow comprises merging the alignment files for experimental groups; next, we use `StringTie` (v2.1.5)[31] to obtain group-specific transcriptome annotations, which are then subsequently combined with `gffcompare`[32] in the guided mode. We use this novel annotation for downstream analysis, including naming genes and transcripts and assigning AS types when possible. Novel SJ, not included in the reference transcript annotation, are also annotated. Currently, Baltica determines the following types: exon skipping (ES), alternative 3' splice-site (A3SS), alternative 5' splice-site (A5SS). The AS type assignment procedure occurs by comparing SJ to overlapping exons features, detected in the *de novo* annotation. We can determine the AS type using distance rules between the start and end coordinates of the SJ and its overlapping exons. Associating the SJ to transcripts enables the study of the splicing event in the context of the transcript sequence and structure. Finally, the framework produces a report that summarizes integration results. It provides an overview of the integration results and an HTML table with one SJ per row, the methods score, SJ annotation, and link to the UCSC Genome Browser[33].

Benchmark

Methods to detect DJU from RNA-seq are valuable tools for prioritizing mechanisms driving splicing changes. From a classification perspective, differential splicing methods aim to classify introns that are truly differently used from the other introns. We approach the differential splicing identification as a binary classification problem. Thus, the positive instance, the differently spliced intron, is more relevant than the negative instance. For the SIRV dataset, introns in the SIRV transcriptome that have fold-change $\neq$ 1 were considered positive, while others introns that were not changing (fold-change of 1) were negative instances. For the paired Illumina-ONT RNA-seq dataset, introns with p-value <0.05 were considered positive instances. This value was obtained with the `edgeR::diffSpliceDGE` function. The set of introns from the SIRV or ONT RNA-seq dataset were used as a reference, so the results among methods are comparable. A true-positive instance (TP) was defined as truly changing and correctly classified, while false-positive (FP) was a negative instance classified as positive. Accordingly, true negatives (TN) and false-negatives (FN) were true negative instances that were correctly or incorrectly called, respectively. The following metrics are defined:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

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

$$Recall\ (or\ sensitivity) = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{FP + TN}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

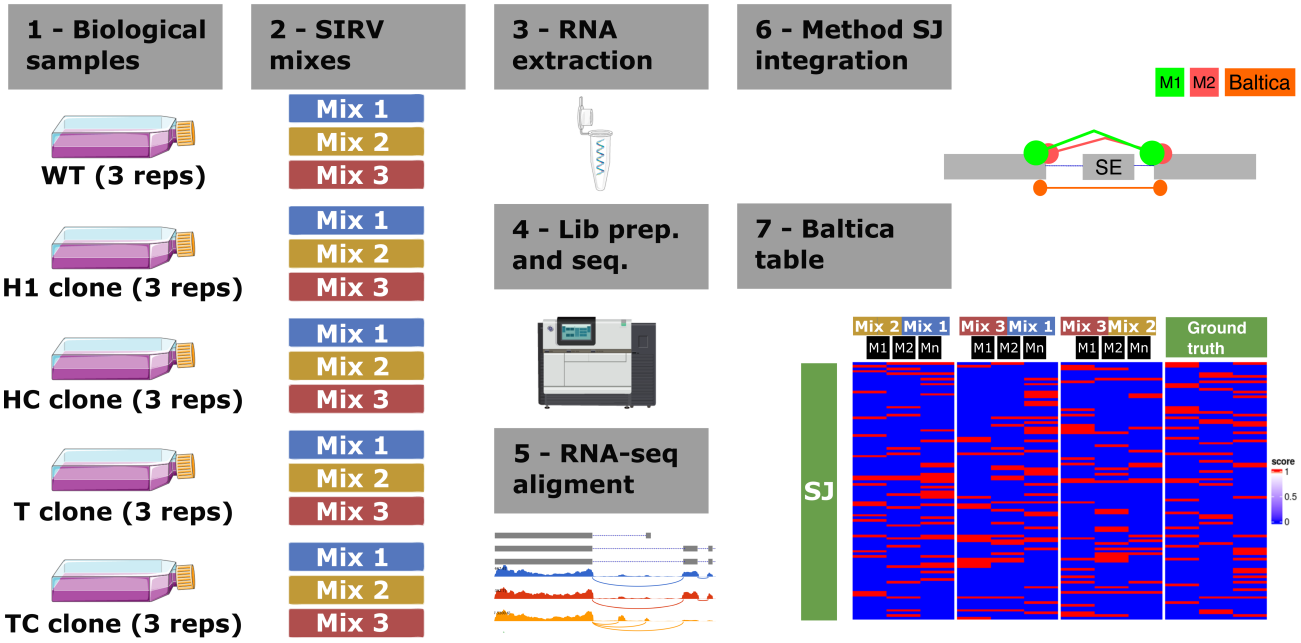
True positive rate (TPR) and false positive rate (FPR) are synonymous to recall and 1− specificity, respectively. Receiver Operating Characteristic (ROC) curve, Precision-Recall (PR) curve, and area under curve (AUC) were computed with ROCR[34]. Confusion matrix and associated statistics report were computed with caret[35], and for that, method scores were made binary using the 0.95 threshold. Heatmap and UpSet plots were created with ComplexHeatmap package[36].

Meta-classifier to identify differential splicing

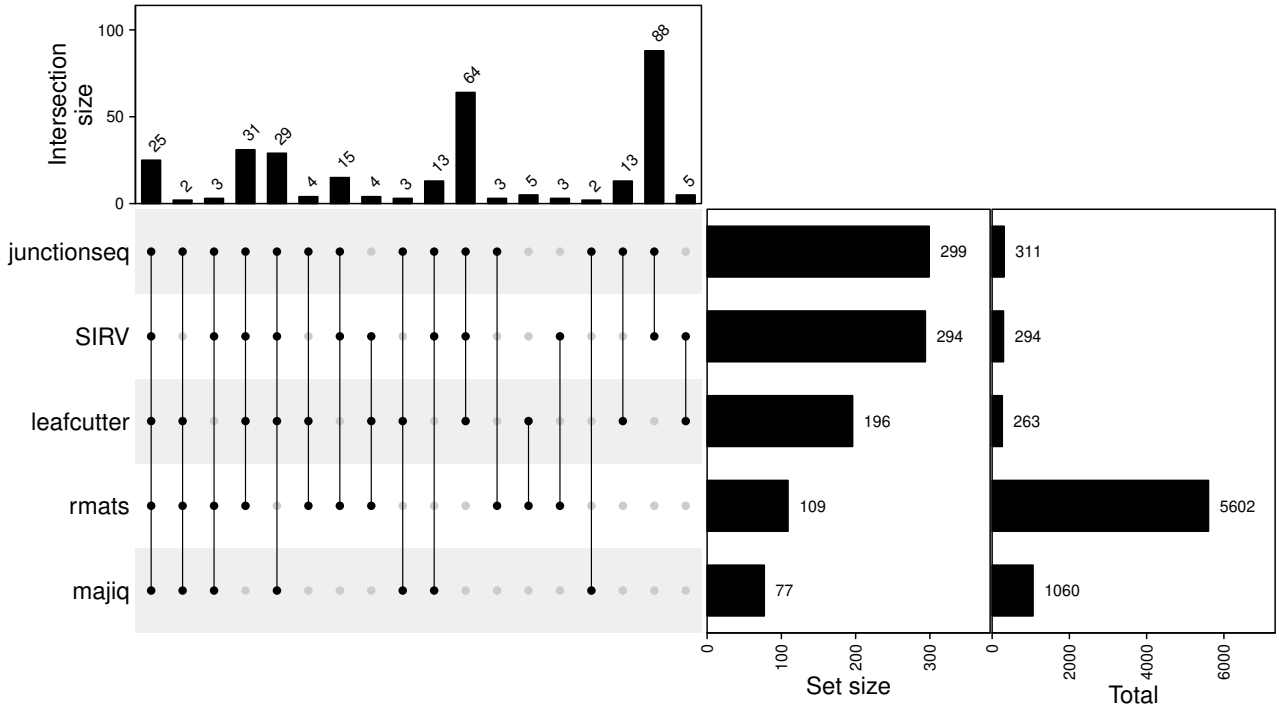
We propose a machine learning approach for a meta-classifier that combines the score of the DJU methods workflows implemented in Baltica. To do so, we train the models with the matrix of DJU scores from the second dataset, with matched third-generation sequencing. The dataset was split into training and testing data (80% vs 20%). DJU scores from the Illumina and ONT RNA-seq were used as input and target values, respectively. Feature selection proceeded with the mlxtend package[37] using either a method score column or a combination of the 4 score columns. Next, a grid search was performed with parameters for the Gradient Boosting Classifier (GBC) and Logistic Regression Classifier (LRC) algorithms implemented in scikit-learn (v0.24.2)[38], for features listed in Table 2, and the combination of columns. The grid search aimed to maximize the area under the ROC curve. This experiment allows us to compare the classification performance from the meta-classifier and classifiers trained from a single method score.

RNA libraries preparation, sequencing, and alignment for the SIRV dataset

Figure 2 schematizes the application of Baltica to the SIRV dataset. Figure 2b compares DJU calls by the 4 methods and the ground-truth. Cell lines, RNA extraction, and RNA-seq were described in Gerbracht and collaborators[39]. In short, we obtained 15 libraries from Flp-In T-REx 293 cells, extracted the RNA fraction with TrueSeq Stranded Total RNA kit (Illumina), followed by ribosomal RNA depletion, with RiboGold Plus kit and Spike-In RNA Variants (Lexogen SIRV, Set-1, Iso Mix E0, E1 and E2, cat 025.031) input. Libraries were sequenced with an Illumina HiSeq4000 sequencer using PE 100bp protocol, which yielded around 50 million reads per sample. Data were deposited in ArrayExpress (E-MTAB-8461). Sequenced reads' adapters and low-quality bases were trimmed, and reads mapping to human precursor ribosomal RNA were discarded. The remaining reads aligned with the human genome (version 38, EnsEMBL 90) extended with the SIRV annotation. In the DJU method benchmarking context, we are not interested in the actual biological condition but the SIRV transcriptome changes. Our experimental design does not



(a) Schematic view for benchmark with the SIRV dataset.



(b) Integrated SJ from DJU methods.

Fig. 2. Integrated DJU results for the SIRV dataset. (a) The experimental design from Gerbracht et al.[39] has five biological groups in replicates, and Table S1 matches the biological samples groups to SIRV mixes and samples identifiers. SIRV mixes were included in a design not confounded to the biological groups. As detailed in Section 2.6, after RNA extraction, library preparation, and sequencing, the sequencing reads were aligned to the human genome extended with the SIRV genome. We apply Baltica workflows, as described in Section 2.1. To integrate the results, we first split AS events into individual SJ that are contained in each event. Next, we correct the start and end coordinates from SJ of multiple methods. Once SJ were integrated, we observed that the statistically significant SJ for `JunctionSeq` (padjust < 0.05), `LeafCutter` (p.adjust < 0.05), `Majiq` (probability_non_changing < 0.05) and `rMATS` (FDR < 0.05) have limited overlap with SJ that are known to change in the SIRV transcriptome. The score is defined as $1 - padjust$, where padjust is the metric for the statistical test from each metric. In the figure, $M_1, M_2, \ldots, M_n$ represent the multiple DJU methods. The UpSet plot in (b) shows distinct sets of introns called significant by combinations of methods and the SIRV annotation (294 true positive SJ). The intersection and set sizes show hits to annotated SIRV introns, while the total column shows the size of hits to the combined human transcriptome and SIRV annotation. The SIRV transcriptome has 98 distinct introns that change in fold change among the mixes. We omit the complement set for the combinations.

confound a SIRV mix with the biological conditions, as detailed in Supplementary Table S1, so any AS events identified within human chromosomes are false calls. The SIRV transcriptome comprises seven genes, 101 transcripts, 138 unique introns, of which 98 change among the 3 mixes, leading to 294 changing introns. In conclusion, the 3 SIRV mixtures in the context of the complex human transcriptome allow us to compare the performance of the DJU methods.

RNA libraries preparation, sequencing, and alignment for the matched ONT RNA-seq and Illumina RNA-seq datasets

The cell lines, RNA extraction, library preparation, and RNA-seq have been described in Boehm et al. (2021)[7]. In detail, wild type (WT) or SMG7 knockout (KO) Flp-In-T-REx-293 cells were seeded on 2x 10 cm plates in high-glucose, GlutaMAX DMEM (Gibco) supplemented with 9% fetal bovine serum (Gibco) and 1x Penicillin Streptomycin (Gibco) at a density of 2.5x106 cells per plate and reverse transfected using 6.25 μl Lipofectamine RNAiMAX and 150 pmol of the respective siRNA (Luciferase as control for WT, SMG6 for SMG7 KO cells) according to the manufacturer's instructions. Cells were harvested after 72 h with 2 ml of peqGOLD TriFast (VWR Peqlab) per plate and total RNA was isolated following the manufacturer's instructions. The following changes were made: Instead of 200 μl chloroform, 150 μl 1-Bromo-3-chloropropane (Molecular Research Center, Inc.) was used. RNA was resuspended in 40 μl RNase-free water. 100 μg of total RNA was subjected to 2 rounds of consecutive poly(A)-enrichment by using 200 μl Dynabeads Oligo (dT)25 and following the manufacturer's instructions. Poly(A)-enriched RNA was eluted with 22 μl RNase-free water and subsequently used for library preparation and ONT direct RNA sequencing. A total of 4 replicates per condition were sequenced. Reads were aligned with minimap2 (v2.17)[40]. We assembled the Nanopore reads and estimated junction read counts based on this new assembly with Stringtie2 (v2.1.1) and Ballgown (v2.14.0)[41]. DJU calls were computed with with EdgeR (v3.24.0)[42] using the `diffSpliceDGE` function. We use $1-$ p-value as the DJU score.

Results

DJU method performance comparison

We present benchmark experiments comprising the 4 DJU methods and 2 datasets, the SIRV dataset, schematized in Figure 2, and the paired Illumina-ONT-seq dataset.

Benchmarking the SIRV dataset

The SIRV dataset comprises 414 SJ from the 3 comparisons with the 3 SIRV mixes. The benchmark shows that `JunctionSeq` outperforms all of the other methods (Figure 3). `JunctionSeq` ranks top with an AUC of 0.87, followed by `LeafCutter` (AUC 0.60), `Majiq` (AUC 0.59), and `rMATS` (AUC 0.53). While `JunctionSeq` performs best, `rMATS` performs close to a random classifier. Supplementary Figure S3 shows the PR curve, which shows the trade-off between precision and recall independent of method score. Of note, `Majiq` presents a low score variability.

All methods have a specificity ≥ 0.68, which represents a fair predictive performance for negative instances (Supplementary Figure S2 and confusion matrices in Supplementary Section S-X.6). Methods' scores show only a limited agreement across each other (Supplementary Figure S4).

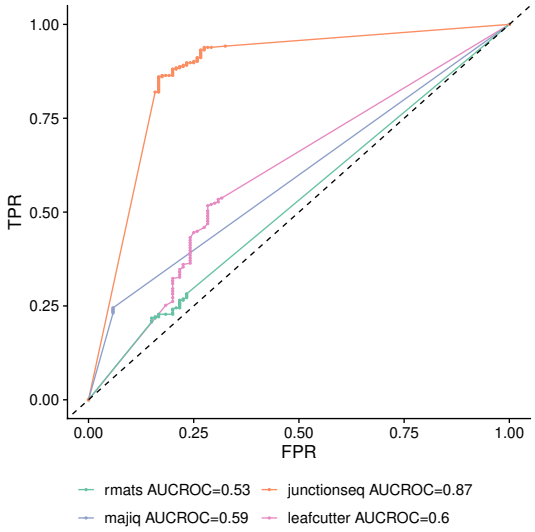


Fig. 3. Baltica benchmark for DJU methods with the SIRV dataset. The introns matching the SIRV transcriptome validated on whether these introns change or not in a given comparison. The performance rank for both curves is consistent between the ROC and PR curves (Supplementary Figure 3): `JunctionSeq` is the top ranking method, followed by `LeafCutter`, `Majiq`, and `rMATS`. AUC: area under the curve; PR: precion-recall; ROC: Receiver Operating Characteristic; TPR: True Positive Rate; FPR: False Positive Rate.

Benchmarking ONT direct RNA-seq dataset as validation

To complement the size limitation of the SIRV dataset, we also benchmark an alternative dataset. Different to the SIRV dataset, this is not a *bona fide* ground-truth dataset. However, we observe similar pattern for the ROC curves of both benchmarks (Figure 3 and Figure 4). Interestingly, we note a reduction of the differences in AUCROC among methods in the second dataset (Figure 4). Most notably, `rMATS` AUCROC increased from 0.53 to 0.65. In terms of AUCPR metric, `JunctionSeq` ranks first with AUCPR of 0.72, followed by `Majiq` (0.61), `LeafCutter` (0.57) and `rMATS` (0.48) (Supplementary Figure S6). Out of 20,744 introns labeled as positive, 21% were called by all 4 methods, and 3 out 4 methods already call 43% (Figure S5).

Table 1 compares the recall, specificity, and F1 metrics for the 2 benchmarks. There are multiple factors to explain the differences between the 2 benchmarks. Of note, the ratio of positive and negative instances change from 2/3 to 1/4 between the SIRV and paired Illumina-ONT RNA-seq datasets. This change can partially explain the overall gain in F1 for all methods, but `JunctionSeq`.

Methods reach a consensus in terms of classification of negative instances. In contrast, for positive instances, the consensus is not as clear (Supplementary Figure S7), and methods scores complement each other. The correlation among scores was low, with a maximum of 0.55 for the `LeafCutter` and `rMATS` pair (Supplementary Figure S8). The relatively low correlation and complementary nature of method scores for positive instances have motivated us to test the performance of a meta-classifier for differential splicing identification.

Meta-classifier with combined DJU scores

Meta-classifiers may improve classification performance by combining the decision of multiple classifiers. To test the

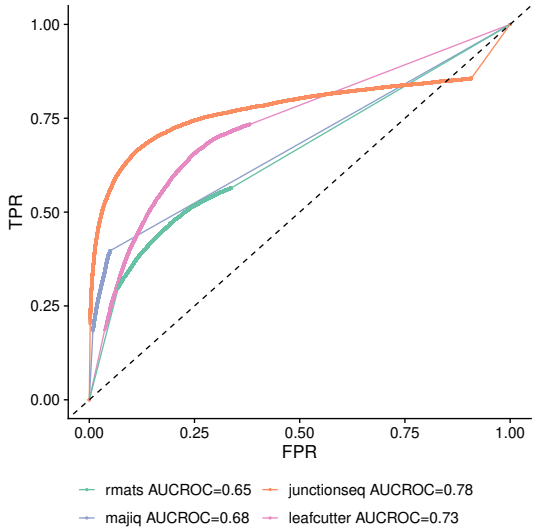


Fig. 4. Baltica benchmark with a paired Illumina-ONT RNA-seq datasets. Performance for DJU methods executed with Illumina RNA-seq and validated with Nanopore RNA-seq show similarities with the benchmark with the SIRV dataset. The classification performance ranks are the same as the benchmark with the previous dataset, in Figure 3. `JunctionSeq` outperforms the other methods in both AUCROC and AUCPR (Supplementary Figure S6). However, in this case, the performance difference among methods is smaller than the benchmark with the SIRV dataset.

	SIRV			Illumina-ONT RNA-seq		
	Recall	Spec.	F1	Recall	Spec.	F1
`rMATS`	0.26	0.77	0.39	0.51	0.76	0.61
`JunctionSeq`	0.91	0.74	0.82	0.65	0.90	0.75
`Majiq`	0.24	0.94	0.39	0.35	0.96	0.51
`LeafCutter`	0.54	0.68	0.60	0.70	0.70	0.70

Table 1. Prediction metrics for the benchmark in the SIRV and ONT RNA-seq datasets. See the complete report at the Supplementary Section S-X.6 (SIRV) and S-X.11 (Illumina-ONT RNA-seq). Spec: Specificity.

application of meta-classifiers to the differential splicing identification task, we have trained 2 models, the LRC and GBC, by fitting them with either a single feature (one method score) or 4 features, one for each method score.

Because models trained in a different set of features may require different sets of parameters, we apply a grid search over the parameters listed in Table 2. We observe the predictors with the combined scores outperform models fitted with scores from a single method score, independent of other parameters and the machine learning algorithm. Table 3 compares these results. To our surprise, the LRC algorithm performs competitively with the GBC algorithm. The GBC algorithm scores an AUCROC of 0.92 in the training set and 0.91 in the testing set, confirming the model's ability to generalize to unseen data. This result demonstrates that combining scores from multiple DJU methods is favorable for differential splicing identification. It improves the predictive performance of the classifier targeting class determined by an orthogonal set, the ONT RNA-seq.

GBC Parameter	**Parameter space**
learning_rate	1, 0.5, 0.25, 0.1, 0.05, 0.01
max_depth	1, 2, 4, 8
subsample	0.5, 0.8, 1.0
min_samples_split	0.2, 0.4, 0.8
n_estimators	1, 2, 4, 8, 16, 32, 64, 100, 200
LRC Parameter	**Parameter space**
penalty	l1, l2
C	$1.0 \times 10^{-3}, 5.6 \times 10^{2}, 3.1, 1.7 \times 10^{2}, 1.0 \times 10^{4}$

Table 2. Parameter space for the grid search procedure with Gradient Boosting Classifier (GBC) and Logistic Regression Classifier (LRC). The procedure aims to maximize the ROC AUC (Area under the Receiver Operating Characteristic Curve) metric of scikit-learn's implementation of the machine learning algorithms. In addition to these parameters, we compare classifiers trained in a single method versus a classifier trained on the 4 methods to test whether the differential splicing identification task benefits from a meta classifier.

	GBC Mean (SD) AUCROC	LRC Mean (SD) AUCROC
Combined	0.92 (0.003)	0.91 (0.002)
`JunctionSeq`	0.88 (0.004)	0.88 (0.004)
`LeafCutter`	0.82 (0.003)	0.83 (0.003)
`rMATS`	0.69 (0.005)	0.70 (0.006)
`Majiq`	0.68 (0.006)	0.68 (0.005)

Table 3. Comparison of the top-scoring meta-classifiers performance. Mean (standard deviation) AUCROC for the 10-fold cross-validation. Each row represents the top-scoring model for models trained on the combined features of a single feature. SD: Standard deviation.

Discussion

Baltica aims to enable the study of integrated results from DJU methods. To achieve that goal, the framework provides workflows[11] and containers[12] to execute the said methods. Next, it combines and re-annotates the results and reports them as an interactive table, as illustrated in Figure 5.

The main challenge for data integration with DJU method results is the difference in individual method implementation. For example, `JunctionSeq` does not produce splicing events. It computes the fold-change of introns, and because it adopts a distinct metric, it handles the effect size comparison with other tools inviable. DJU methods also use different definitions to define the coordinate of the AS events. Due to that, we have decided not to integrate effect-size, the PSI, from various methods, despite understanding this attribute is critical for alternative splicing identification.

This manuscript applies a benchmark to 2 independent datasets, the SIRV and the paired Illumina-ONT RNA-seq datasets. Other DJU method performance comparisons have used simulated datasets or datasets with a small subset of PCR-validated AS events[43, 9]. These 2 approaches do not fully appreciate the complexity of the RNA-seq experiment. Other benchmarks integrate splicing events on the gene level. Baltica can systematically resolve the different coordinate systems and compare multiple methods without these limitations.

The benchmark with the SIRV transcriptome is a special case that embeds a small complex transcriptome into a human transcriptome. The SIRV transcriptome has 138 introns, of

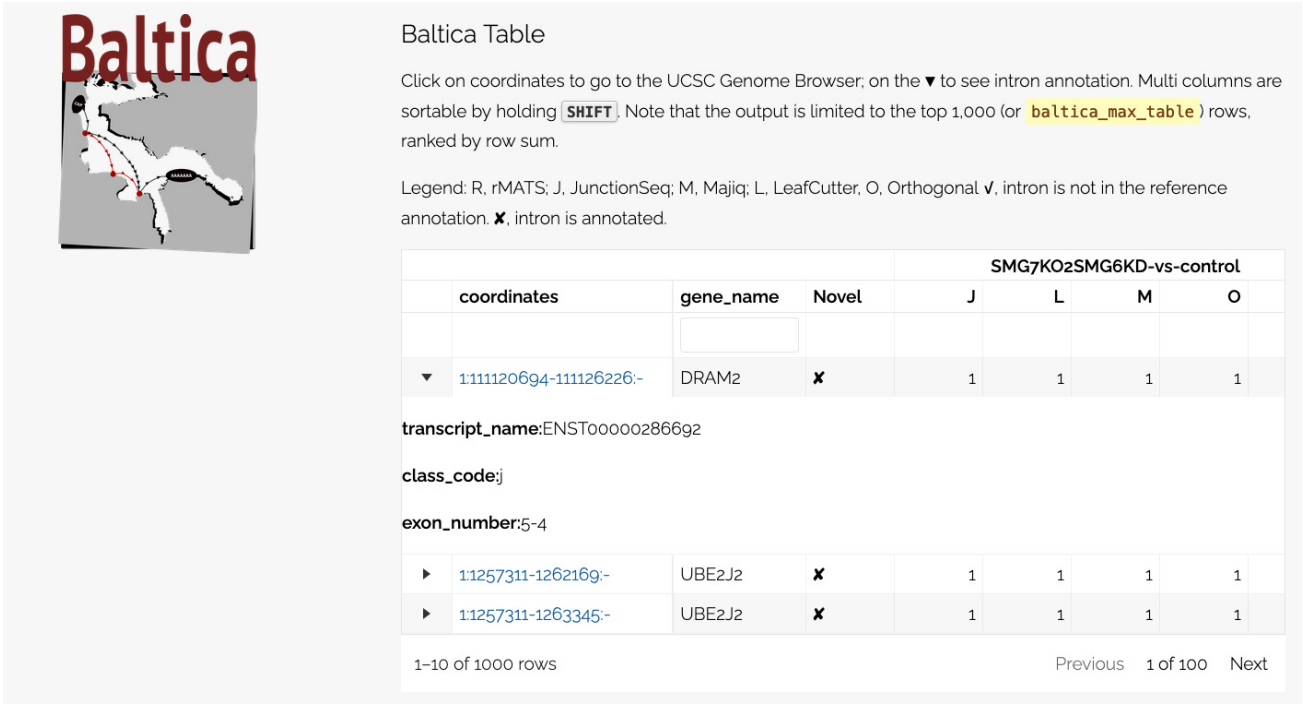


Fig. 5. Interative Baltica table. The image shows a static view of the Baltica report, focusing on the table component. The table is pre-sorted by sums of the scores columns, and it can be sorted by the scores or filtered by gene name. In addition, annotation is provided for introns that match the *de novo* transcriptome annotation. Introns that are annotated are flagged in the **Novel** column. **J**, **L**, **M**, **R** and **O** stand for JunctionSeq, LeafCutter, Majiq, rMATs and orthogonal scores, respectively. The **gene_name**, **class_code**, **transcript_name** and **exon_number** come from StringTie integration. The **class_code** matches how novel transcripts compare to annotated ones. The rMATS score column is missing from the image.

which 2/3 change between the 3 contrasts. All SIRV introns are annotated, benefiting methods that rely on the transcriptome annotation, such as `JunctionSeq`. In addition, it contains almost 2:1 number positive to negative instances ratio, donor and acceptor splice with non-canonical sequence, and transcripts in opposite strands that share the identical introns. These attributes make the SIRV an important dataset for the benchmark of differential splicing identification methods.

In addition, we also benchmark the methods with the paired Illumina and ONT RNA-seq. Second-generation sequencing, primarily Illumina RNA-seq, has promoted many discoveries in differential AS. However, Illumina RNA-seq is limited by its relatively short read length, leading to a limited resolution of one single intron at a time. Third-generation RNA-seq technology, represented by Iso-seq from Pacific Biosciences (PacBio) and ONT RNA-seq, overcome this issue by offering longer sequencing reads than second-generation RNA-seq. The longer reads enable unambiguous matching to multiple introns in transcript isoforms and thus allow a better resolution of the transcriptome structure[44, 45]. Hybrid sequencing approaches pairing third-generation sequencing, and second-generation sequencing can benefit from both the deep coverage and the long-reads to improve AS identification task[46, 45, 47, 48]. This dataset allows us to compare the methods scores on 65,408 introns that have been tested for DJU with EdgeR in the ONT RNA-seq dataset.

We understand the Nanopore DJU scores are not a *bona fide* ground-truth, but only an orthogonal approach to the splicing identification problem. Although the apparent differences between the SIRV and the ONT RNA-seq datasets, the benchmark results were remarkably similar. Specifically, the ranks of DJU methods were the same in both datasets. On the other hand, the difference in F1 metric among methods was less pronounced in the second dataset.

In addition to the considerations detailed above, readers should interpret the benchmark results in the context of the 2 datasets. For example, the 2 benchmark use cases use transcriptomes with known introns, and they should benefit from methods that rely on a complete annotation. However, to a certain extent, current methods like `LeafCutter` and `Majiq` don't rely on the annotation for the intron count modeling. Also, while `Majiq`, `LeafCutter`, and `rMATS` output events that comprise multiple SJ, `JunctionSeq` output a single p-value per intron. Moreover, for the second benchmark, one must keep in mind that `EdgeR`, which was used to analyze the Nanopore data, and `JunctionSeq` use a similar statistical model.

We demonstrate that the integration of DJU methods results can be helpful for intron prioritization. To do so, we have trained multiple machine learning models using 2 algorithms and five feature sets, one for each method score or the combination of the 4. We observed that models trained in the combined feature set performed better than methods trained in a single feature independent of machine algorithms. The LRC method has fewer parameters, requires fewer input data, and is faster to train but has lower predictive performance than GBC[49]. We used the LRC algorithm as a base model, intending to build upon it with the GCB. However, the 2 models presented a comparable performance. We provide the source code for model training and validation so that users can apply the method in their datasets. To our knowledge, this approach is the first method to take advantage of DJU scores to train a meta-classifier. However, this practice is common for

bioinformatics practices and has been used to train predictors for protein secundary structure [50]. The integration helps prioritize introns for further experimental validation and may expand our knowledge on functional consequences of AS events.

We plan to extend Baltica in the future with additional DJU methods workflows and will include unit tests for tracking output differences due to changes in software versions. We invite the user community to follow the Baltica repository `https://github.com/dieterich-lab/baltica` for the code and update documentation and to participate in Baltica development.

Key Points

- Methods to identify differential splicing changes are critical to detect the association of introns to genomic features such as genetic variants or splicing factor binding sites.
- However, these methods differ in many aspects, and the lack of standardization hampers the comparison of their results
- Baltica enables reproducible execution and integration of DJU methods. The integrated results allow benchmarking the different methods, and it reveals `JunctionSeq` ranks first in F1 metric across 2 independent benchmarks.
- Meta-classifiers trained on methods scores outperform all models trained in single method scores, demonstrating the data integration advantage.

Acknowledgments

We are grateful to members of the Dieterich Lab for feedback on this work. Specifically, we would like to thank Tobias Jakobi and Harald Wilhelmi for the technical setup and feedback. The authors would like to thank Jennifer Gerbracht for sharing her dataset and her feedback on the earlier stages of the project. All authors acknowledge the support of the West-German Genome Center (WGGC) and, in particular, Tobias Lautwein.

Competing interests

Authors declare no competing interests.

Author contributions

V.B. has tested the framework. V.B and N.G. have performed cell culture, library preparation, and sequencing. C.D. has processed the data. C.D. and N.G. supervised the project. T.B.B. processed data, developed and implemented the computational workflows and analysis scripts, tested the framework, led the analyses, contributed to the figures, and wrote the paper. All authors contributed to the research, including analysis, manuscript writing, and feedback for the framework design and development.

Data availability

The SIRV RNA-seq is available at ArrayExpress E-MTAB-8461. The paired RNA-Seq and Nanopore-Seq are available from CD and NG upon reasonable request. In addition, data matrices used for the benchmark and meta-prediction are available at https://doi.org/10.5281/zenodo.5643428.

Funding

This work has been supported by Informatics for Life, funded by the Klaus Tschira Foundation and the Deutsche Forschungsgemeinschaft (DFG, DI1501/8-2, and GE2014/6-2). Nanopore sequencing was supported by a DFG Sequencing Call grant to CD (DI1501/12-1, project 423957469).

Thiago Britto-Borges is a postdoctoral researcher in Biomedical Data Science at the University Hospital Heidelberg.

Volker Boehm is a postdoctoral researcher in RNA Biology at the Institute for Genetics, University of Cologne.

Niels Gehring is a professor at the Institute for Genetics, University of Cologne.

Christoph Dieterich is a full professor and head of the Klaus Tschira Institute for Integrative Computational Cardiology at the University Hospital Heidelberg.

References

1. Jing Liu, Xu Kong, Mengkai Zhang, Xiao Yang, and Xiuqin Xu. RNA binding protein 24 deletion disrupts global alternative splicing and causes dilated cardiomyopathy. *Protein & Cell*, 10(6):405–416, September 2018.
2. Marina M. Scotti and Maurice S. Swanson. RNA mis-splicing in disease. *Nature Reviews Genetics*, 17(1):19–32, November 2015.
3. Javier Tapial, Kevin C.H. Ha, Timothy Sterne-Weiler, André Gohr, Ulrich Braunschweig, Antonio Hermoso-Pulido, Mathieu Quesnel-Vallières, Jon Permanyer, Reza Sodaei, Yamile Marquez, Luca Cozzuto, Xinchen Wang, Melisa Gómez-Velázquez, Teresa Rayon, Miguel Manzanares, Julia Ponomarenko, Benjamin J. Blencowe, and Manuel Irimia. An atlas of alternative splicing profiles and functional associations reveals new regulatory programs and genes that simultaneously express multiple major isoforms. *Genome Research*, 27(10):1759–1768, aug 2017.
4. Kishore Jaganathan, Sofia Kyriazopoulou Panagiotopoulou, Jeremy F. McRae, Siavash Fazel Darbandi, David Knowles, Yang I. Li, Jack A. Kosmicki, Juan Arbelaez, Wenwu Cui, Grace B. Schwartz, and et al. Predicting splicing from primary sequence with deep learning. *Cell*, 176(3):535–548.e24, Jan 2019.
5. Yang I. Li, Bryce van de Geijn, Anil Raj, David A. Knowles, Allegra A. Petti, David Golan, Yoav Gilad, and Jonathan K. Pritchard. RNA splicing is a primary link between genetic variation and disease. *Science*, 352(6285):600–604, Apr 2016.
6. Eric L. Van Nostrand, Peter Freese, Gabriel A. Pratt, Xiaofeng Wang, Xintao Wei, Rui Xiao, Steven M. Blue, Jia-Yu Chen, Neal A. L. Cody, Daniel Dominguez, and et al. A large-scale binding and functional map of human RNA-binding proteins. *Nature*, 583(7818):711–719, Jul 2020.
7. Volker Boehm, Sabrina Kueckelmann, Jennifer V. Gerbracht, Sebastian Kallabis, Thiago Britto-Borges, Janine Altmüller, Marcus Krüger, Christoph Dieterich, and Niels H. Gehring. SMG5-SMG7 authorize nonsense-mediated mRNA decay by enabling SMG6 endonucleolytic activity. *Nature Communications*, 12(1), Jun 2021.
8. Michael H. Radke, Victor Badillo-Lisakowski, Thiago Britto-Borges, Dieter A. Kubli, René Jüttner, Pragati Parakkat, Jacobo Lopez Carballo, Judith Hüttemeister, Martin Liss, Arne Hansen, Christoph Dieterich, Adam E.

Mullick, and Michael Gotthardt. Therapeutic inhibition of RBM20 improves diastolic function in a murine heart failure model and human engineered heart tissue. *Science Translational Medicine*, 13(622), dec 2021.

9. Arfa Mehmood, Asta Laiho, Mikko S Venäläinen, Aidan J McGlinchey, Ning Wang, and Laura L Elo. Systematic evaluation of differential splicing tools for RNA-seq studies. *Briefings in Bioinformatics*, 21(6):2052–2065, Dec 2019.
10. Yarden Katz, Eric T Wang, Edoardo M Airoldi, and Christopher B Burge. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nature Methods*, 7(12):1009–1015, Nov 2010.
11. Felix Mölder, Kim Philipp Jablonski, Brice Letcher, Michael B. Hall, Christopher H. Tomkins-Tinch, Vanessa Sochat, Jan Forster, Soohyun Lee, Sven O. Twardziok, Alexander Kanitz, and et al. Sustainable data analysis with Snakemake. *F1000Research*, 10:33, Apr 2021.
12. Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. Singularity: Scientific containers for mobility of compute. *PLOS ONE*, 12(5):e0177459, May 2017.
13. Björn Grüning, Ryan Dale, Andreas Sjödin, Brad A. Chapman, Jillian Rowe, Christopher H. Tomkins-Tinch, Renan Valieris, and Johannes Köster. Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nature Methods*, 15(7):475–476, Jul 2018.
14. Liguo Wang, Shengqin Wang, and Wei Li. RSeQC: quality control of RNA-seq experiments. *Bioinformatics*, 28(16):2184–2185, June 2012.
15. Simon Andrews, Felix Krueger, Anne Segonds-Pichon, Laura Biggins, Christel Krueger, and Steven Wingett. FastQC. Babraham Institute, January 2012.
16. Philip Ewels, Måns Magnusson, Sverker Lundin, and Max Käller. MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*, 32(19):3047–3048, Jun 2016.
17. Shihao Shen, Juw Won Park, Zhi-xiang Lu, Lan Lin, Michael D. Henry, Ying Nian Wu, Qing Zhou, and Yi Xing. rmats: Robust and flexible detection of differential alternative splicing from replicate rna-seq data. *Proceedings of the National Academy of Sciences*, 111(51):E5593–E5601, Dec 2014.
18. Stephen W. Hartley and James C. Mullikin. Detection and visualization of differential splicing in RNA-seq data with JunctionSeq. *Nucleic Acids Research*, page gkw501, June 2016.
19. Jorge Vaquero-Garcia, Alejandro Barrera, Matthew R Gazzara, Juan Gonzalez-Vallinas, Nicholas F Lahens, John B Hogenesch, Kristen W Lynch, and Yoseph Barash. A new view of transcriptome complexity and regulation through the lens of local splicing variations. *eLife*, 5, February 2016.
20. Yang I. Li, David A. Knowles, Jack Humphrey, Alvaro N. Barbeira, Scott P. Dickinson, Hae Kyung Im, and Jonathan K. Pritchard. Annotation-free quantification of RNA splicing using LeafCutter. *Nature Genetics*, 50(1):151–158, December 2017.
21. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2021.
22. M. Lawrence, R. Gentleman, and V. Carey. rtracklayer: an R package for interfacing with genome browsers. *Bioinformatics*, 25(14):1841–1842, May 2009.
23. Michael Lawrence, Wolfgang Huber, Hervé Pagès, Patrick Aboyoun, Marc Carlson, Robert Gentleman, Martin T. Morgan, and Vincent J. Carey. Software for computing and annotating genomic ranges. *PLoS Computational Biology*, 9(8):e1003118, Aug 2013.
24. Hadley Wickham, Mara Averick, Jennifer Bryan, Winston Chang, Lucy McGowan, Romain François, Garrett Grolemund, Alex Hayes, Lionel Henry, Jim Hester, and et al. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686, Nov 2019.
25. Juan L. Trincado, Juan C. Entizne, Gerald Hysenaj, Babita Singh, Miha Skalic, David J. Elliott, and Eduardo Eyras. SUPPA2: fast, accurate, and uncertainty-aware differential splicing analysis across multiple conditions. *Genome Biology*, 19(1), Mar 2018.
26. Kuan-Ting Lin and Adrian R Krainer. PSI-Sigma: a comprehensive splicing-detection method for short-read and long-read RNA-seq analysis. *Bioinformatics*, 35(23):5048–5054, May 2019.
27. Stephen W. Hartley and James C. Mullikin. QoRTs: a comprehensive toolset for quality control and data processing of RNA-seq experiments. *BMC Bioinformatics*, 16(1), Jul 2015.
28. S. Anders, A. Reyes, and W. Huber. Detecting differential usage of exons from RNA-seq data. *Genome Research*, 22(10):2008–2017, June 2012.
29. Kelsy C. Cotto, Yang-Yang Feng, Avinash Ramu, Zachary L. Skidmore, Jason Kunisaki, Megan Richters, Sharon Freshour, Yiing Lin, William C. Chapman, Ravindra Uppaluri, and et al. RegTools: Integrated analysis of genomic and transcriptomic data for the discovery of splicing variants in cancer. pre-print, Oct 2018.
30. Thiago Britto Borges, Tobias Jakobi, and Volker Böhm. dieterich-lab/baltica: v1.1, September 2021.
31. Sam Kovaka, Aleksey V. Zimin, Geo M. Pertea, Roham Razaghi, Steven L. Salzberg, and Mihaela Pertea. Transcriptome assembly from long-read RNA-seq alignments with StringTie2. *Genome Biology*, 20(1), Dec 2019.
32. Geo Pertea and Mihaela Pertea. GFF utilities: GffRead and GffCompare. *F1000Research*, 9:304, Sep 2020.
33. W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and a. D. Haussler. The human genome browser at UCSC. *Genome Research*, 12(6):996–1006, May 2002.
34. T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer. ROCR: visualizing classifier performance in R. *Bioinformatics*, 21(20):3940–3941, Aug 2005.
35. Max Kuhn. Building predictive models in R using the caret package. *Journal of statistical software*, 28(1):1–26, 2008.
36. Zuguang Gu, Roland Eils, and Matthias Schlesner. Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics*, 32(18):2847–2849, May 2016.
37. Sebastian Raschka. Mlxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack. *Journal of Open Source Software*, 3(24):638, 2018.
38. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
39. Jennifer V Gerbracht, Volker Boehm, Thiago Britto-Borges, Sebastian Kallabis, Janica L Wiederstein, Simona Ciriello,

Dominik U Aschemeier, Marcus Krüger, Christian K Frese, Janine Altmüller, and et al. CASC3 promotes transcriptome-wide activation of nonsense-mediated decay by the exon junction complex. *Nucleic Acids Research*, 48(15):8626–8644, Jul 2020.

40. Heng Li. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, 34(18):3094–3100, May 2018.
41. Mihaela Pertea, Geo M Pertea, Corina M Antonescu, Tsung-Cheng Chang, Joshua T Mendell, and Steven L Salzberg. Stringtie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nature Biotechnology*, 33(3):290–295, Feb 2015.
42. M. D. Robinson, D. J. McCarthy, and G. K. Smyth. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, Nov 2009.
43. Ruolin Liu, Ann E Loraine, and Julie A Dickerson. Comparisons of computational methods for differential alternative splicing detection using RNA-seq in plant systems. *BMC Bioinformatics*, 15(1), Dec 2014.
44. Bo Wang, Elizabeth Tseng, Michael Regulski, Tyson A Clark, Ting Hon, Yinping Jiao, Zhenyuan Lu, Andrew Olson, Joshua C. Stein, and Doreen Ware. Unveiling the complexity of the maize transcriptome by single-molecule long-read sequencing. *Nature Communications*, 7(1), Jun 2016.
45. Mohan T. Bolisetty, Gopinath Rajadinakaran, and Brenton R. Graveley. Determining exon connectivity in complex mRNAs by nanopore sequencing. *Genome Biology*, 16(1), Sep 2015.
46. K. F. Au, V. Sebastiano, P. T. Afshar, J. D. Durruthy, L. Lee, B. A. Williams, H. van Bakel, E. E. Schadt, R. A. Reijo-Pera, J. G. Underwood, and et al. Characterization of the human ESC transcriptome by hybrid sequencing. *Proceedings of the National Academy of Sciences*, 110(50):E4821–E4830, Nov 2013.
47. Matthew T Parker, Katarzyna Knop, Anna V Sherwood, Nicholas J Schurch, Katarzyna Mackinnon, Peter D Gould, Anthony JW Hall, Geoffrey J Barton, and Gordon G Simpson. Nanopore direct RNA sequencing maps the complexity of Arabidopsis mRNA processing and m6A modification. *eLife*, 9, Jan 2020.
48. Laura Schulz, Manuel Torres-Diz, Mariela Cortés-López, Katharina E. Hayer, Mukta Asnani, Sarah K. Tasian, Yoseph Barash, Elena Sotillo, Kathi Zarnack, Julian König, and et al. Direct long-read RNA sequencing identifies a subset of questionable exitrons likely arising from reverse transcription artifacts. *Genome Biology*, 22(1), Jun 2021.
49. Tjeerd van der Ploeg, Peter C Austin, and Ewout W Steyerberg. Modern modelling techniques are data hungry: a simulation study for predicting dichotomous endpoints. *BMC Medical Research Methodology*, 14(1), Dec 2014.
50. Alexey Drozdetskiy, Christian Cole, James Procter, and Geoffrey J. Barton. JPred4: a protein secondary structure prediction server. *Nucleic Acids Research*, 43(W1):W389–W394, Apr 2015.

Supplementary Materials for Baltica: integrated splice junction usage analysis

Here we provide the supplementary information to the manuscript Baltica: integrated splice junction usage analysis, by Thiago Britto-Borges, Volker Boehm, Niels H. Gehring and Christoph Dieterich.

Experimental design for the SIRV dataset.

Biological sample	Spike-in mix	Sample id
WT_1	Mix1_1	106030
WT_2	Mix2_1	106032
WT_3	Mix3_1	106034
H1 clone_1	Mix1_2	106036
H1 clone_2	Mix2_2	106038
H1 clone_3	Mix3_2	106040
HC clone_1	Mix1_3	106042
HC clone_2	Mix2_3	106044
HC clone_3	Mix3_3	106046
T clone_1	Mix1_4	106048
T clone_2	Mix2_4	106050
T clone_3	Mix3_4	106052
TC clone_1	Mix1_5	106054
TC clone_2	Mix2_5	106056
TC clone_3	Mix3_5	106058

Table S1. The biological sample, spike-in mix, and sample identifier associations for samples in dataset E-MTAB-8461. Related to Figure 2. Note that the biological samples and the spike-in mixes are not confounded.

Popularity of DJU methods.

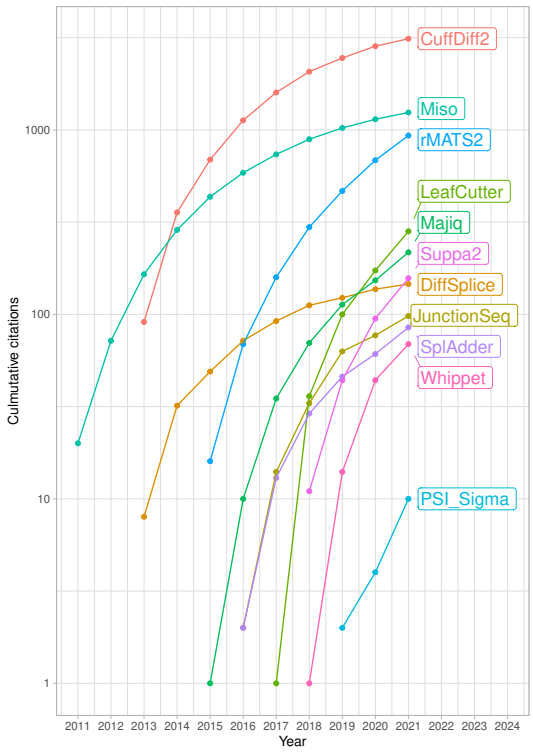


Fig. S1. The popularity of selected methods for differential splicing identification. The citation over time for selected DJU methods. The y-axis is log_{10} transformed. Data sourced from https://scholar.google.com/ using the `scholar` R package.

Heatmap for the SIRV benchmark

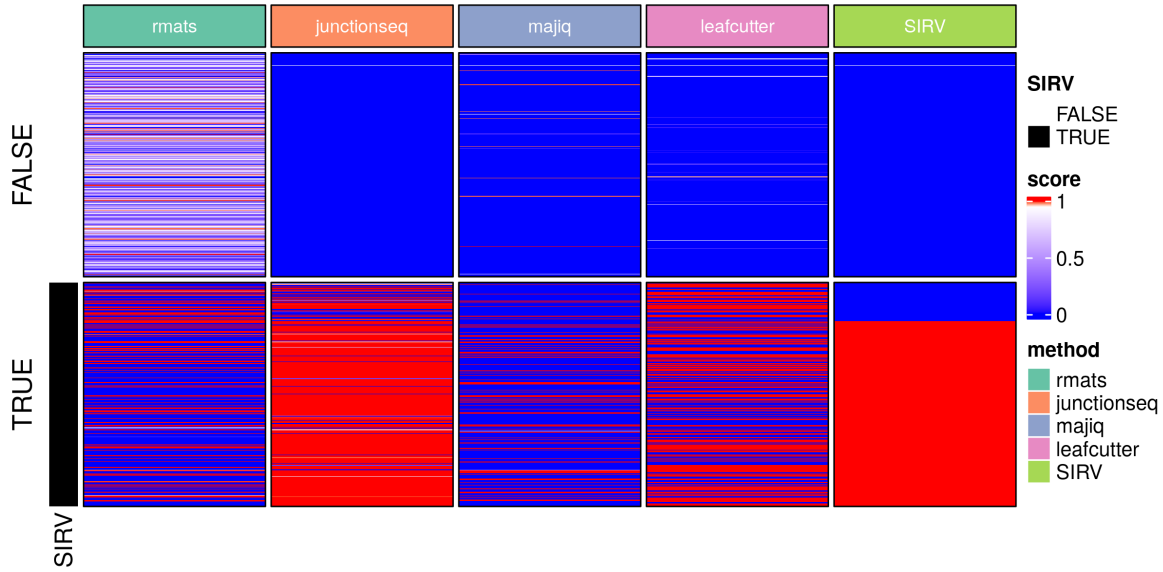


Fig. S2. Four hundred fourteen introns were randomly sampled from the human transcriptome (top) and the same amount in the SIRV transcriptome (bottom). Introns in the SIRV transcriptome are annotated, as shown in the leftmost annotation. Colorbar shows method score as divergent color from red (1) to blue (0), and the highest score better the performance classification. The methods score for negative instances, in the top panel, agreement while for positive, for instance, the 2/3 in the bottom panel, are not.

SIRV benchmark precision-recall curve

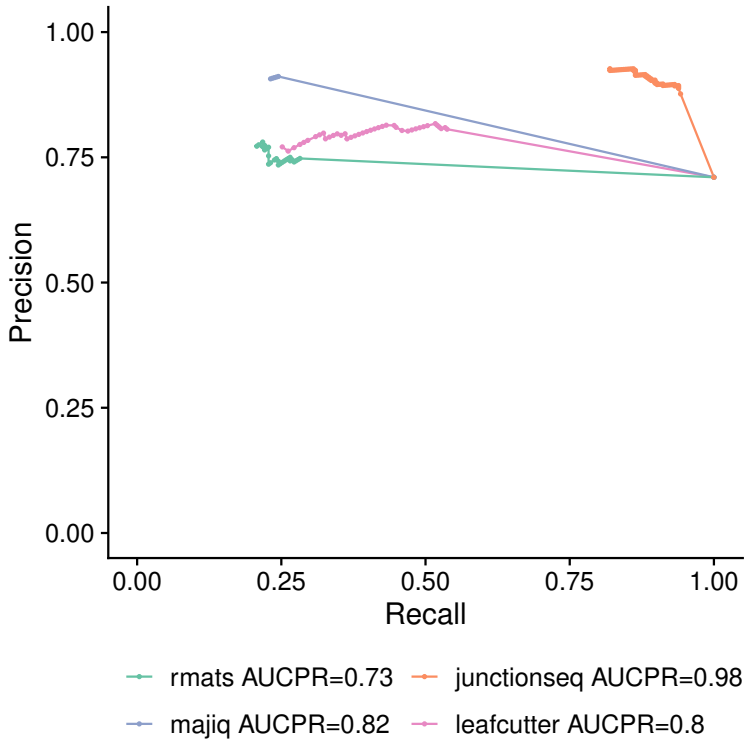


Fig. S3. SIRV benchmark precision-recall curve. Overall, changes in recall have only a slight effect on precision for all the methods. `JunctionSeq` ranks first (AUCPR=0.98), followed by `Majiq` (AUCPR=0.82), `LeafCutter` (AUCPR=0.8), and `rMATS` (AUCPR=0.73). PR: precion-recall; AUCPR: area under the precision-recall curve. Relates to Figure 3.

Spearman correlation for the SIRV benchmark

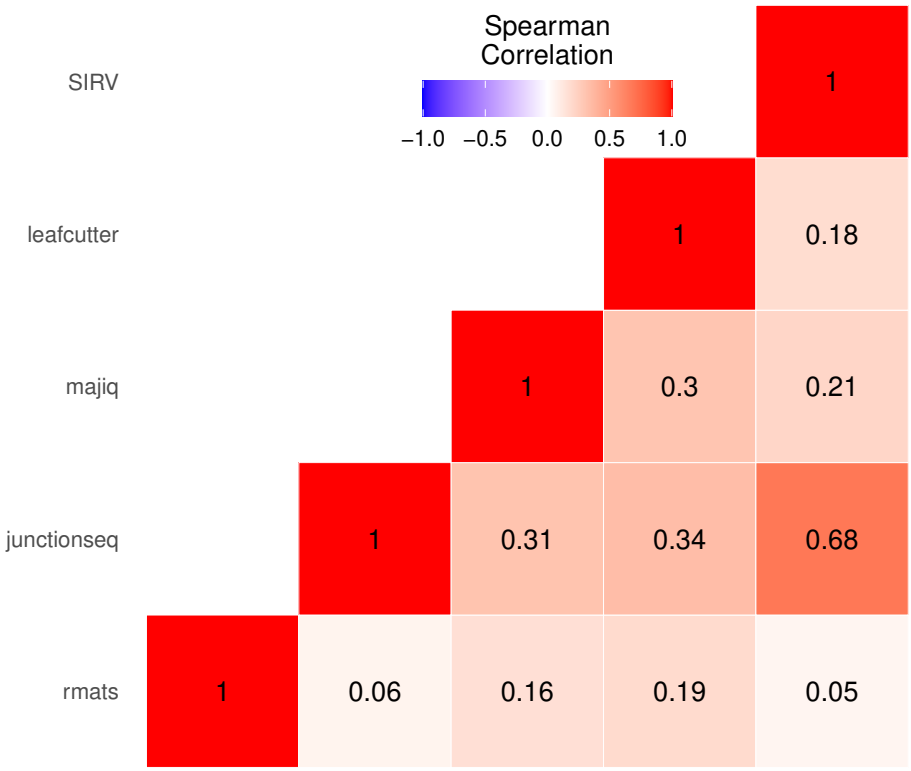


Fig. S4. Spearman correlation among methods and the SIRV ground-truth. Scores were log transformed with the function $f(x) = -log_{10}(x + 1e^{-10})$ and only introns from the SIRV transcriptome were tested. Overall, pairs of methods have a low correlation with an mean of 0.25.The maximum value is of 0.68 for l and ground-truth. .

Perfomance metrics and confusion matrix for the SIRV benchmark

Metrics were obtained with `caret::confusionMatrix` function. True and false instances were separate with the 0.95 threshold.

```
JunctionSeq

Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE    89   26
     TRUE     31  268

               Accuracy : 0.8623
                 95% CI : (0.8253, 0.894)
    No Information Rate : 0.7101
    P-Value [Acc > NIR] : 1.957e-13

                  Kappa : 0.6614

 Mcnemar's Test P-Value : 0.5962

            Sensitivity : 0.9116
            Specificity : 0.7417
         Pos Pred Value : 0.8963
         Neg Pred Value : 0.7739
             Prevalence : 0.7101
         Detection Rate : 0.6473
   Detection Prevalence : 0.7222
```

```
      Balanced Accuracy : 0.8266

       'Positive' Class : TRUE
```

```
rMATS

Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE    92  213
     TRUE     28   81

               Accuracy : 0.4179
                 95% CI : (0.3699, 0.467)
    No Information Rate : 0.7101
    P-Value [Acc > NIR] : 1

                  Kappa : 0.029

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.2755
            Specificity : 0.7667
         Pos Pred Value : 0.7431
         Neg Pred Value : 0.3016
             Prevalence : 0.7101
         Detection Rate : 0.1957
   Detection Prevalence : 0.2633
      Balanced Accuracy : 0.5211

       'Positive' Class : TRUE
```

```
Majiq

Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE   113  224
     TRUE      7   70

               Accuracy : 0.442
                 95% CI : (0.3935, 0.4913)
    No Information Rate : 0.7101
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1171

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.2381
            Specificity : 0.9417
         Pos Pred Value : 0.9091
         Neg Pred Value : 0.3353
             Prevalence : 0.7101
         Detection Rate : 0.1691
   Detection Prevalence : 0.1860
      Balanced Accuracy : 0.5899

       'Positive' Class : TRUE
```

i

```
LeafCutter

Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE    82  136
     TRUE     38  158

               Accuracy : 0.5797
                 95% CI : (0.5305, 0.6277)
    No Information Rate : 0.7101
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1778

 Mcnemar's Test P-Value : 1.93e-13

            Sensitivity : 0.5374
            Specificity : 0.6833
         Pos Pred Value : 0.8061
         Neg Pred Value : 0.3761
             Prevalence : 0.7101
         Detection Rate : 0.3816
   Detection Prevalence : 0.4734
      Balanced Accuracy : 0.6104

       'Positive' Class : TRUE
```

Baltica integration for the ONT RNA-seq dataset

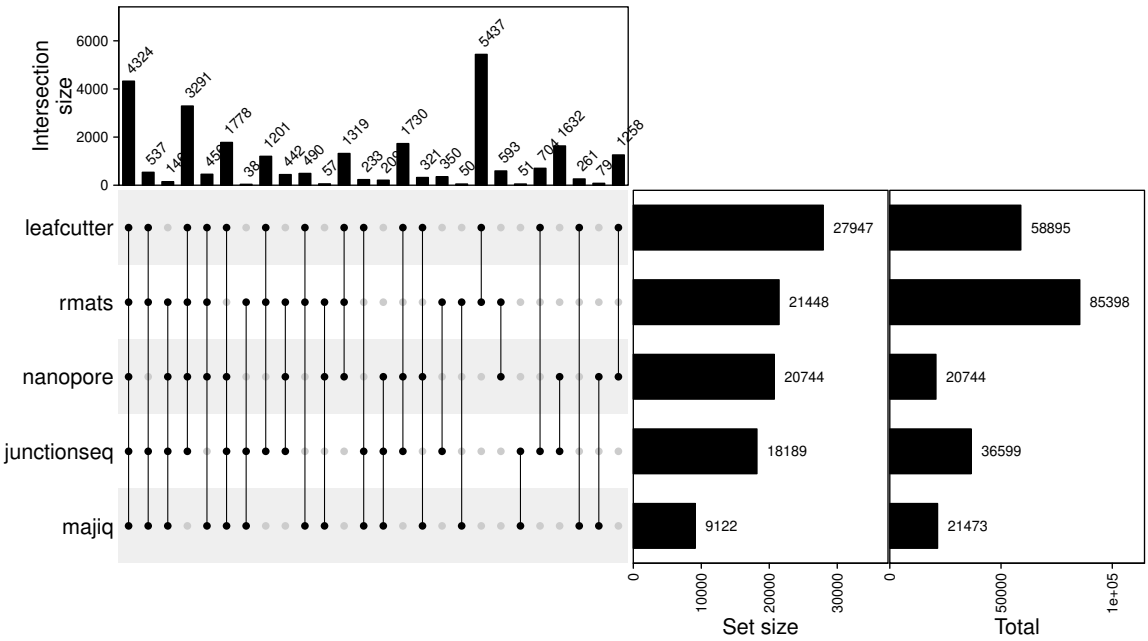


Fig. S5. Baltica integrated SJ for the ONT dataset. The plot shows distinct sets of introns with *score* > 0.95 by combinations of methods and the DJU for the ONT RNA-seq. The complement sets, combinations with a degree of 1, were omitted. The intersection and set sizes refer to the number of the sets not omitted, while the total shows the total number of calls.

ONT RNA-seq benchmark precision-recall curve

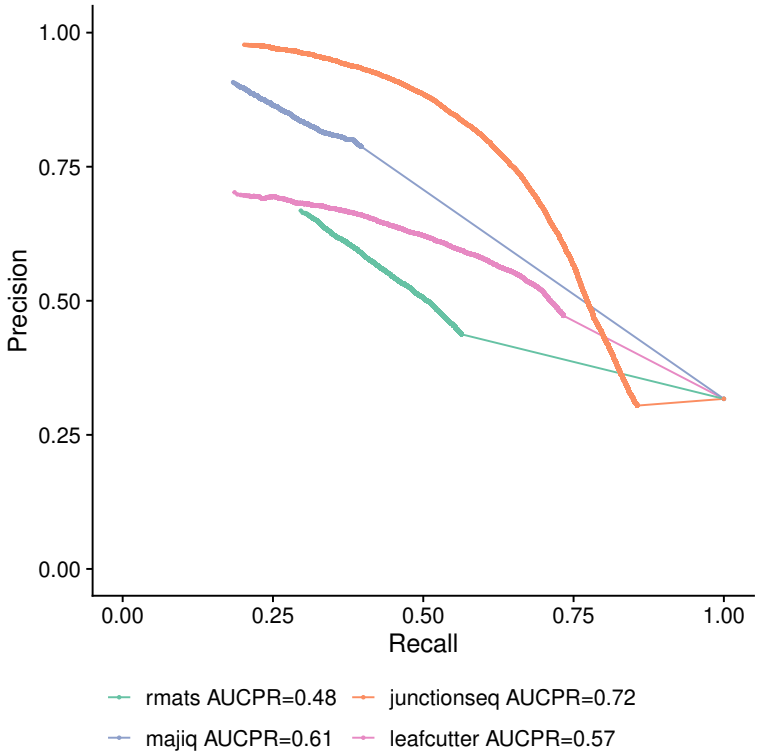


Fig. S6. ONT RNA-seq benchmark precision-recall curve. Different from the SIRV benchmark, the precision-recall curve for the ONT RNA-seq benchmark shows negative correlation between precision and recall. However, the method rank is consistent with the SIRV benchmark with `JunctionSeq` first (AUCPR=0.72), and then `Majiq` (AUCPR=0.61), `LeafCutter` (AUCPR=0.57), and `rMATS` (AUCPR=0.48). PR: precion-recall; AUCPR: area under the precision-recall curve. Relates to Figure 4

Heatmap for benchmark with the ONT RNA-seq dataset

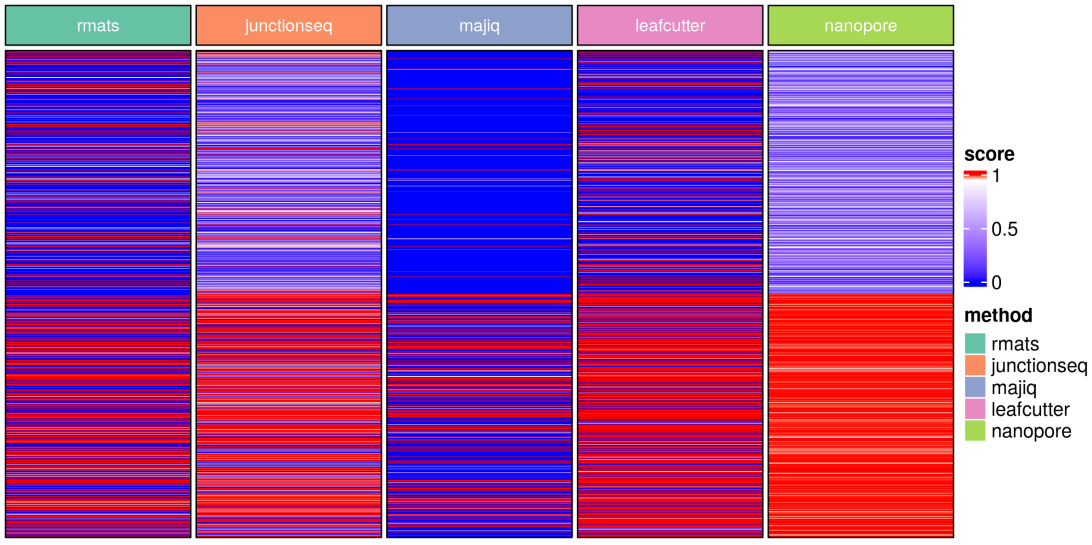


Fig. S7. The 0.95 thresholds separate five hundred negative (top) and positive instances (bottom) detected by the ONT RNA-seq. Colorbar shows method score as divergent color from red (1) to blue (0), higher is better. Related with Supplementary Figure S2

Spearman correlation among methods scores and ONT RNA-seq DJU method for the ONT RNA-seq dataset

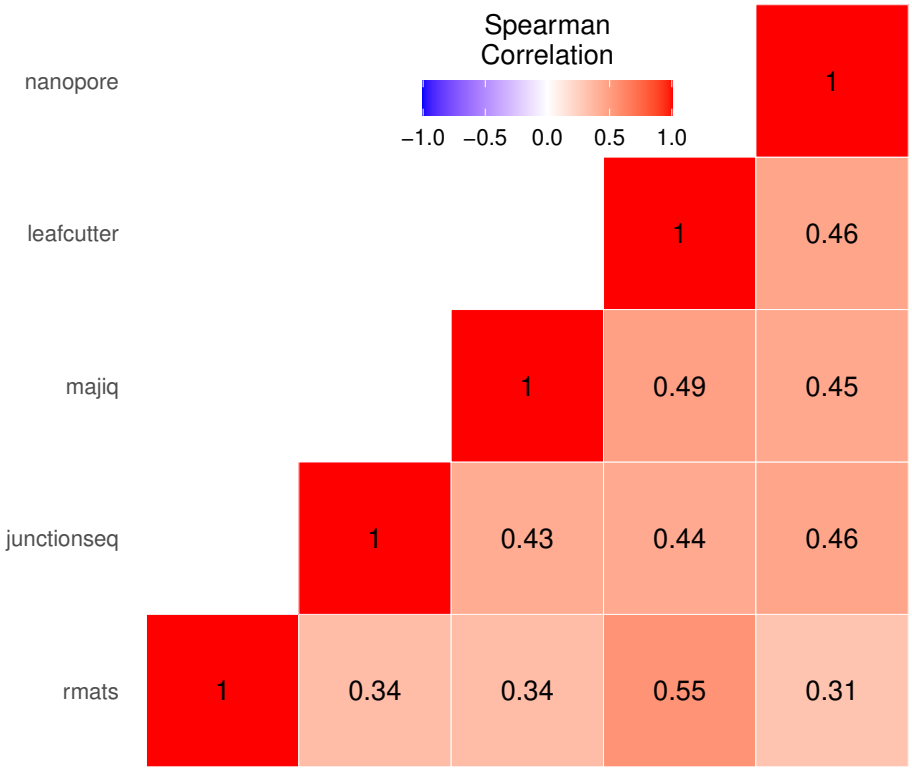


Fig. S8. Spearman correlation among scores from DJU methods and ONT RNA-seq dataset DJU. Overall, methods scores show higher correlation than in the SIRV benchmark (Supplementary Figure S4), however the agreement between to methods is still limited, whith the maximum value of 0.55 for `rMATS` versus `LeafCutter`. Scores were transformed with $f(x) = -log_{10}(x + 1e^{-10})$ and only introns presented in the ONT RNA-seq dataset were tested.

Confusion matrix for the benchmark with the ONT dataset

```
JunctionSeq

Confusion Matrix and Statistics

          Reference
Prediction FALSE  TRUE
     FALSE 40027  7192
     TRUE   4637 13552

               Accuracy : 0.8192
                 95% CI : (0.8162, 0.8221)
    No Information Rate : 0.6829
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5682

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.6533
            Specificity : 0.8962
         Pos Pred Value : 0.7451
         Neg Pred Value : 0.8477
             Prevalence : 0.3171
         Detection Rate : 0.2072
   Detection Prevalence : 0.2781
```

```
      Balanced Accuracy : 0.7747

       'Positive' Class : TRUE
```

```
rMATS

Confusion Matrix and Statistics

          Reference
Prediction FALSE  TRUE
     FALSE 33844 10116
     TRUE  10820 10628

               Accuracy : 0.6799
                 95% CI : (0.6763, 0.6835)
    No Information Rate : 0.6829
    P-Value [Acc > NIR] : 0.947

                  Kappa : 0.2677

 Mcnemar's Test P-Value : 1.182e-06

            Sensitivity : 0.5123
            Specificity : 0.7577
         Pos Pred Value : 0.4955
         Neg Pred Value : 0.7699
             Prevalence : 0.3171
         Detection Rate : 0.1625
   Detection Prevalence : 0.3279
      Balanced Accuracy : 0.6350

       'Positive' Class : TRUE
```

```
Majiq

Confusion Matrix and Statistics

          Reference
Prediction FALSE  TRUE
     FALSE 42912 13374
     TRUE   1752  7370

               Accuracy : 0.7687
                 95% CI : (0.7655, 0.772)
    No Information Rate : 0.6829
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.3718

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.3553
            Specificity : 0.9608
         Pos Pred Value : 0.8079
         Neg Pred Value : 0.7624
             Prevalence : 0.3171
         Detection Rate : 0.1127
   Detection Prevalence : 0.1395
      Balanced Accuracy : 0.6580

       'Positive' Class : TRUE
```

LeafCutter

Confusion Matrix and Statistics

```
          Reference
Prediction FALSE  TRUE
     FALSE 31194  6267
     TRUE  13470 14477

               Accuracy : 0.6982
                 95% CI : (0.6947, 0.7018)
    No Information Rate : 0.6829
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.3626

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.6979
            Specificity : 0.6984
         Pos Pred Value : 0.5180
         Neg Pred Value : 0.8327
             Prevalence : 0.3171
         Detection Rate : 0.2213
   Detection Prevalence : 0.4273
      Balanced Accuracy : 0.6982

       'Positive' Class : TRUE
```