**TITLE:**

OLego: *de novo* discovery of novel exon junctions and small exons from RNA-Seq data

**AUTHORS:**

Wu, Jie

Cold Spring Harbor Laboratory

Cold Spring harbor, NY

wuj@cshl.edu

Feng, Huijuan

Department of Systems Biology

Department of Biochemistry and Molecular Biophysics

Center for Motor Neuron Biology and Disease

Columbia University

New York, NY

[hf2304@columbia.edu](mailto:hf2304@columbia.edu)

Zhang, Chaolin

Department of Systems Biology

Department of Biochemistry and Molecular Biophysics

Center for Motor Neuron Biology and Disease

Columbia University

New York, NY

cz2294@columbia.edu

**CORRESPONDING AUTHOR:**

Zhang, Chaolin

cz2294@columbia.edu

**KEYWORDS:**

Bioinformatics, genomics, next generation sequencing (NGS), alignment, mapping, RNA-Seq, exon junction detection, splicing, small exons, micro-exons

**SHORT ABSTRACT:**

Here we present a protocol to use OLego to perform fast, accurate and sensitive alignment of spliced RNA-Seq reads. OLego is optimized for *de novo* detection of exon junctions and small exons by using small seeds with very efficient algorithm implementations.

**LONG ABSTRACT:**

OLego is a software tool specifically designed for *de novo* mapping of spliced RNA-Seq reads. It uses a seed-and-extend scheme to identify exon junctions and small exons from the reads. One major difference between OLego and other splice mappers is its ability to achieve high sensitivity by using strategic and efficient searches with very small seeds (12~15 nt for mammalian genomes). Meanwhile, to improve accuracy and to resolve ambiguous mapping at exon junctions, OLego uses a built-in regression model to score and rank exon junctions by considering both splice-site strength and intron size. To ensure efficient alignments of massive RNA-Seq datasets with this high-resolution search, we do not rely on external mappers but use Burrows–Wheeler transform (BWT) and full-text minute-space (FM)-index in multiple steps to efficiently map seeds, locate exon junctions and identify small exons. OLego is implemented in C++ supporting fully multi-threaded execution, and allows fast processing of large-scale data. In this article, we describe the workflow of OLego and present a standard protocol to run OLego for RNA-Seq alignment. Its performance was compared with other mainstream tools using both simulated and real experimental data to demonstrate its sensitivity, accuracy and efficiency.

**INTRODUCTION:**

Deep RNA sequencing (RNA-Seq) is a recently developed approach that uses the next-generation sequencing (NGS) technologies to profile the transcriptome at unprecedented depth and resolution, which has greatly facilitated the discovery and quantification of transcripts or transcript variants resulted from alternative splicing[1](#_ENREF_1). The first step to analyze RNA-Seq data is alignment of many millions of short reads to the reference genome to locate exon body reads and exon junction reads. The challenges of this step are two folds: First, short reads increase the possibility of having multiple hits on the genome by chance, and also make it difficult to identify exon junctions due to small overlaps in the exons across the junction. Second, the large amount of data requires optimizations in the algorithm to ensure high efficiency of alignment without sacrificing accuracy and sensitivity.

*De novo* discovery of exon junctions in RNA-Seq data has become an important resource to annotate gene structures and study gene expression regulation including alternative splicing. To overcome the aforementioned challenges, a number of algorithms have been developed in the recent years to specifically align spliced reads to exon junctions[2-8](#_ENREF_2). To locate exon junction reads, a common strategy used by most of the programs is the seed-and-extend method, although details may vary. In this method, each read is first split into two or more segments (seeds), which are then aligned to the genome independently without allowing splits. A search for an exon junction is then performed between two adjacent seeds that are mapped to two genomic loci separated by certain distance consistent with a candidate intron (double-anchor search). For a seed near the end of the read, the alignment of the single seed can also be used as an anchor to guide a search for a potential exon junction using the unaligned portion at the end of the read at a higher resolution (single-anchor search) (Figure 1).

Many of these software tools require an external mapper (e.g., Bowtie[9](#_ENREF_9)) for seed mapping, which prevented the use of short seeds in practice, because of formidable requirements on storage space for temporary files and computation time. This caveat largely limits the mapping resolution of these tools, and hence their sensitivity to detect exon junctions and small exons. To overcome this limitation, we recently designed a novel algorithm and program named OLego for fast and sensitive mapping of RNA-Seq reads using very small seeds[10](#_ENREF_10).

The overview of OLego is briefly described in the following workflow (Figure 1):

OLego first attempts to map unspliced reads to the genome. Reads which cannot be mapped within a certain number of mismatches or small insertions/deletions (indels) are processed in the following junction searching steps. Each read is segmented into multiple seeds. Small seed is used to improve sensitivity (default: 15 nt with 1-nt overlap, which is optimized for mammalian-sized genomes). In the next step, each seed of a read is mapped independently to the reference genome by querying the indexed genome. Their hits are then collected, sorted, and clustered into one or more potential alignments of the read. Afterwards, the hits in each potential alignment are further grouped into candidate exons. A candidate exon is defined when one or more seeds are mapped to a continuous genomic region while two consecutive candidate exons are separated by a large gap representing potential introns. Alignment of each candidate exon is then extended to identify approximate exon boundary positions. Then double-anchor and single-anchor junction searches are performed between each pair of consecutive candidate exons or at the ends of each alignment if unaligned sequence segments are present at the end of the read. In addition, potential small exons are searched when there are unaligned gaps between two candidate exons. Finally, exon junctions and exons are connected to obtain the complete alignment of the read. Multiple candidate alignments, if any, are ranked and filtered by considering the number of mismatches, splice site strength and intron sizes.

Since each read is aligned independently, multiple threads can be easily enabled for parallelization. For more details of the algorithm, please refer to the original publication[10](#_ENREF_10). Additional online documentation and software updates are available at http://zhanglab.c2b2.columbia.edu/index.php/OLego.

[Figure 1]

**PROTOCOL:**

1. Installation and preparation to run OLego

* 1. Installation

Note: The major components of OLego ( olego and olegoindex ) were written in C++ and can be installed and run on mainstream Unix-based systems (Linux or Mac OS X) with the GCC compiler installed. Installation of Perl and R is required by additional scripts in the software package for offline construction of the regression models that score the strength of exon junctions (models for human and mouse have been provided in the package), and post processing of read alignment, such as merging paired-end reads and identification of unique exon junctions.

* + 1. Download the most recent stable version of the OLego source code or executable binaries from SourceForge ( <http://sourceforge.net/projects/ngs-olego/files/>. Alternatively, use git to retrieve the most updated version of the source code from the repository:

git clone git://git.code.sf.net/p/ngs-olego/code olego

Note: Skip to 1.2 if the binary executable files are downloaded in this step.

* + 1. Decompress the package and change the current directory to OLego folder:

tar zxvf olego.src.v1.x.x.tgz

cd olego/

* + 1. Compile the source code. Two executable files (olego and olegoindex) will be created in the folder.

make

Note: For 32-bit systems, edit the “Makefile” to remove the “-m64” tag before compiling the code.

* 1. Prepare for the indexed reference genome

1.2.1) Download the FASTA genome sequence from the UCSC genome browser.(e.g., <http://hgdownload.soe.ucsc.edu/goldenPath/mm10/bigZips/chromFa.tar.gz>). Here the mouse genome is used as an example in this protocol.

1.2.2) Decompress the gz file and concatenate the files into a single FASTA file (e.g. mm10.fa). It is recommended that random chromosomes and haplotype sequences are typically excluded from the analysis.

tar zxvf chromFa.tar.gz

rm chr\*random.fa chrUn\*fa

cat \*.fa >mm10.fa

1.2.3) Run olegoindex to build the index for the reference genome:

olegoindex mm10.fa –p mm10

Note: There will be eight files (mm10.pac, mm10.ann, mm10.amb, mm10.rpac, mm10.bwt, mm10.rbwt, mm10.sa and mm10.rsa) generated.

* 1. Other required files

1.3.1) Download the pre-built exon junction database from the OLego website “http://zhanglab.c2b2.columbia.edu/index.php/OLego”. Decompress this file.

gunzip mm10.intron.hmr.bed.gz

1.3.1.1) Alternatively, create a custom junction database from transcript annotations (BED format file) using the Perl script included in the package (bed2junc.pl).

bed2junc.pl transcripts.bed transcripts.junc.bed

Note: It is strongly recommended that a database of annotated exon junctions should be provided to OLego (option -j) for more sensitive mapping.

1.3.2) Find the logistic regression models for mouse and human exon junctions included in the sub folder “models”. Use the script included in the package to build models for other species. Type “perl regression\_model\_gen/OLego\_regression.pl” in the olego folder to see the help message.

1. Align the reads using OLego
   1. Align paired-end reads separately:

olego -v -r models/mm.cfg -j mm10.intron.hmr.bed -o r1.sam mm10 r1.fq

olego -v -r models/mm.cfg -j mm10.intron.hmr.bed -o r2.sam mm10 r2.fq

2.1.1) Use option “-t” to specify the number of CPU cores to be used in the alignment.

2.1.2) Use “-M” to specify the maximum number of mismatches or indels allowed. The default will be determined based on the read length (Supplementary Table 1).

2.1.3) Use “-w” to control the seed size.

Note: The seeds will be evenly distributed on the read from the beginning to the end with a maximum overlap defined by --word-max-overlap. The default is 15 nt seed with 1-nt overlap, which is recommended for relatively long reads (e.g., >70 nt). A smaller seed (e.g., 12-14 nt) should be used for shorter reads in combination with different seed overlap size (e.g., 0-2 nt) to maximize the number of seeds that covers the read.

2.1.4) For strand-specific RNA-Seq libraries, check option “–-strand-mode”.

2.1.5) Use “-r” to specify the logistic regression model prepared in step 1.3.2.

2.1.6) For other options, refer to the online documentation.

* 1. Merge the mapping results from paired-end reads according to their distance and orientations to resolve ambiguities:

mergePEsam.pl r1.sam r2.sam merge.sam

2.2.1) Use “-d” to specify the maximum distance between the two ends on the reference. This script requires the two ends mapped to different strands by default, use “--ss” (requiring mapping to the same strand) or “--ns” (ignoring strand information) to change the behavior.

* 1. Optionally, re-align reads with the exon junctions identified from the data. Some reads supporting novel exon junctions that are failed during the first-pass alignment can be recovered because of more extensive searches for reads mapped to known exon junctions.

2.3.1) Convert the SAM alignment output file to BED format:

sam2bed.pl merge.sam merge.bed

2.3.1.1) Use “--uniq” to keep only reads aligned to unique loci (i.e., single hits). Use “--use-RNA-strand” to extract the RNA strand instead of the read strand for junction reads when the orientation can be inferred from the splice sites ( “+” is assigned to exonic reads) .

2.3.2) Extract the exon junctions from the BED file:

bed2junc.pl merge.bed merge.junc.bed

2.3.3) Remap with this new exon junction annotation file:

olego -v -r mm.cfg -j merge.junc.bed --non-denovo -o r1.remap.sam mm10.fa r1.fq

olego -v -r mm.cfg -j merge.junc.bed --non-denovo -o r2.remap.sam mm10.fa r2.fq

2.3.4) See Section 2.2 to merge the paired-end reads.

3 Post-processing and downstream analysis

3.1) Sort and convert the merge.sam into BAM format files. SAMtools[11](#_ENREF_11" \o "Li, 2009 #92) is required for this step:

samtools view -uSh merge.sam | samtools sort - merge.sort

3.2) Perform downstream analysis using other tools. Examples include quantification of gene expression (e.g., Cufflinks/Cuffdiff[12](#_ENREF_12" \o "Trapnell, 2010 #93) and HTSeq-count[13](#_ENREF_13)/DESeq[14](#_ENREF_14" \o "Anders, 2010 #95)/EdgeR[15](#_ENREF_15" \o "McCarthy, 2012 #96)), alternative splicing (e.g., Quantas: http://zhanglab.c2b2.columbia.edu/index.php/Quantas) and *de novo* transcriptome assembly (e.g., Trinity[16](#_ENREF_16) and Velvet[17](#_ENREF_17)).

**REPRESENTATIVE RESULTS:**

We previously assessed OLego (v1.0.0) using simulated data (10 million paired-end reads of 100 or 150 nt) and compared with other spliced alignment tools[10](#_ENREF_10) (TopHat v1.4.0 , ref.[3](#_ENREF_3), MapSplice v1.15.2, ref.[4](#_ENREF_4) and PASSion v1.2.1, ref.[7](#_ENREF_7)). For exon junction discovery, OLego had the lowest false negative rate (FNR) (6.8%~8.2%), which almost halved the FNRs from TopHat (12.8%~15.4%) and PASSion (14.8%~15.5%). Meanwhile, OLego achieved a high accuracy with high positive predictive values (PPVs) (97.7%~98.1%), which is comparable to TopHat v1.4.0 and better than MapSplice and PASSion. In addition, we demonstrated that OLego is very sensitive to discover small or micro-exons. For exons with a length between 9 and 39 nt, OLego achieved a much lower FNR (9.2%~13.4%), compared to TopHat v1.4.0 (31.5%~36.1%), MapSplice (19.4%~24.1%), and PASSion (21.3%~32.3%). OLego is particularly sensitive in detecting micro-exons smaller than 15 nt. While OLego discovered >75% of the exons in this range in the simulation test, the other programs discovered substantially lower percentages (6.1% ~ 47.4%).

The high sensitivity of exon-junction and small-exon discovery is attributed to the high-resolution searches using small seeds and optimization in various steps. Nevertheless, OLego is also among the fastest programs in terms of the mapping speed, especially when multiple threading is enabled. The high efficiency of alignment originates from the internal querying algorithm (BWT and FM-index [18](#_ENREF_18)) used in multiple mapping steps, the multi-threading implementation and several heuristics fine-tuned to limit the query complexity. In comparison with TopHat v1.4.0 , which was the fastest among the other three algorithms we tested, OLego achieved a comparable speed when using single CPU core, and 2-fold faster with 16 CPU cores (please see our initial publication[10](#_ENREF_10) for more details).

The use of small seeds is very computationally expensive. To find a fine balance between mapping speed and sensitivity, we have made additional optimization and improvements since the initial release of OLego. These improvements include the use of overlapping seeds starting from v1.1.2, such that a similar sensitivity can be achieved with slightly longer seeds (e.g., 15-nt seeds with 1 nt overlap vs. 14-nt seeds without overlap), while the mapping speed is greatly improved.

Since our initial publication, several other programs (e.g., STAR[19](#_ENREF_19)) were published and new versions have been released for some of the programs we previously compared. Therefore, we carried out additional comparisons of OLego v1.1.5 with TopHat v2.0.11[20](#_ENREF_20) and STAR v2.3.0e using the same simulated datasets[10](#_ENREF_10). Default parameters are used for both programs, except that intron sizes were limited to the range of 20~500,000 nt and the “--microexon-search” option was enabled for TopHat2. The primary alignments from each program were then extracted to evaluate the performance in *de novo* discovery of exon junctions and small exons.

Compared to the results from OLego v1.0.0, the new version (v1.1.5) achieved higher sensitivity with lower FNRs on exon-junction discovery (5.7%~7.2%). The sensitivity for small exon discovery also improved. For example, the FNRs for exons of 9~39 nt decreased from 9.2%~13.4% to 8%~11.1%. At the same time, the PPVs for both exon junction and small exon discovery are maintained at the same level (94.2% ~ 98.1%). In addition, the speed has been improved significantly because of optimizations implemented in the newer version. The computing time decreased from 0.8h to 0.51h to map 10 million 2×100-nt reads, and from 1.4h to 1.07h to map 10 million 2×150-nt reads (Table 1). The accuracy and sensitivity of TopHat2 are comparable to TopHat v1.4.0 . STAR has the highest sensitivity for overall junction discovery at the expense of slightly lower PPV. In addition, its FNRs on discovery of small exons are higher than OLego, especially for exons with a length between 9 and 15 nt (~70% compared to ~20%). In terms of speed STAR is 4-6 fold faster than OLego, and ~10 fold faster than TopHat2. This is not surprising because STAR uses uncompressed suffix arrays to index the reference genome. Consequently, STAR has a large memory footprint (>24G for mouse genome), which is 6 times larger than TopHat2 and OLego (<4G in general).

[Table 1]

In our previous comparison, we also ran OLego on an RNA-Seq dataset from mouse retina[21](#_ENREF_21) and identified 1,665 micro-exons between 9 nt and 27 nt, among which 630 were novel. We selected 15 potential novel cassette exons and performed RT-PCR validations. All these micro-exons were successfully validated and the inclusion ratios observed in RNA-Seq data and RT-PCR were highly correlated (Pearson correlation coefficient R=0.85)[10](#_ENREF_10).

In this new test, we ran OLego v1.1.5, TopHat v2.0.11 and STAR v2.3.0e on the same dataset and extracted the micro-exons discovered by each program. The new version of OLego found even more micro-exons compared to the old version (1,792 vs. 1,665). Compared to OLego, TopHat2 found substantially fewer (713) exons, including 566 exons annotated in the inclusive gene models (combined from 11 sources[10](#_ENREF_10),[21](#_ENREF_21)) and 431 exons annotated in RefSeq (these numbers are 1,067 and 731 for OLego). OLego also identified more novel micro-exons of high confidence (i.e., cassette exons flanked by annotated exons and supported by exon-junction reads[10](#_ENREF_10), 452 vs. 70). STAR identified more micro-exons in the range of 9~39nt (2,249) overall. However, only 898 of them are in the inclusive gene models, and only 320 un-annotated exons fell into the high-confidence novel exon category (which together accounts for 1,218 exons or 54% of total, as compared to 1,519 exons or 85% of total for OLego). These results indicate that STAR has a higher false positive rate and lower sensitivity compared to OLego in discovery of small and micro-exons (Table 2).

[Table 2]

We then looked into the 15 novel micro-exons successfully validated in our previous study[10](#_ENREF_10). Among these, STAR and TopHat2 identified 14 and 9 exons (both flanking exon junctions were identified with at least one supporting read), respectively (Supplementary Table 2). In addition, we estimated the inclusion ratios of these exons based on alignments by each program and correlated them with the inclusion ratios estimated from RT-PCR validations (Figure 2). We found that TopHat2 and STAR substantially underestimated the inclusion ratios of a portion of these micro-exons, because many reads supporting the inclusion isoforms failed in alignment (Supplementary Table 2).

[Figure 2]

[Supplementary Table 1]

[Supplementary Table 2]

**Figure 1: Workflow of OLego.** Exonic alignment is attempted first. Exon-junction alignment is then performed for unmapped reads through the following steps: seeding, seed mapping and hit clustering, candidate-exon identification and extension, single- and double- anchor searches. Finally, exons and exon-junctions are connected to identify the optimal complete alignment for each read.

**Figure 2: Comparison of OLego, TopHat2 and STAR on micro-exon discovery and quantification.** A set of 15 micro-exons validated by RT-PCR was used for this analysis. (a) An IGV screenshot shows a micro-exon in *Ykt6* (15nt) identified by OLego but missed by TopHat2 and STAR. (b) Correlations between inclusion ratios estimated from RNA-Seq data based on alignment by each program and RT-PCR validations.

**Table 1: Comparison of OLego, TopHat2 and STAR using simulated RNA-Seq data.** Two simulated datasets with different read lengths (10 million paired-end reads, 100 nt or 150 nt) were used for comparison. Sensitivity and accuracy of OLego (v1.1.5), TopHat (v2.0.11) and STAR (v2.3.0e) on both exon junction discovery and small exon discovery were evaluated. Mapping speed is based on 16 Intel Xeon CPU cores (2.0 GHz).

**Table 2. Numbers of micro-exons discovered in RNA-Seq data by different programs.**

**Supplementary Table 1. Default maximum number of mismatches or indels allowed for different read lengths.**

**Supplementary Table 2. Micro-exon discovery in real data.** The numbers of exon-junction reads mapped to the 15 RT-PCR-validated novel micro-exons by different programsin RNA-Seq data from mouse retina are shown. The inclusion ratios of these exons were computed using these junction reads and compared to the inclusion ratios estimated from RT-PCR analysis.

**DISCUSSION:**

OLego was specifically designed to align spliced RNA-Seq reads with high sensitivity using small seeds. As demonstrated by both simulated tests and real data analysis, this tool allows us to discover exon junctions and small exons *de novo* at high accuracy and sensitivity. In addition, despite the more exhaustive search at high resolution, OLego achieves high efficiency of read mapping, making it suitable for novel exon and exon-junction discovery in large-scale RNA-Seq data.

A unique feature of OLego is its ability to detect small and micro-exons which are generally more difficult to identify and hence frequently missed in previous studies and databases. These exons are particularly interesting, because many of them are alternatively spliced and under tissue-specific regulation22,23. In our comparisons, OLego identified substantially more small exons of high confidence from both simulated and real data. By analyzing RNA-Seq data obtained from various conditions, we will be able to obtain novel insights into these exons, which are excellent models to study mechanisms and function of alternative splicing regulation, given the very limited information encoded in the exon sequences.

To achieve the best sensitivity of this algorithm, we recommend users to specify a suitable combination of seed size and overlapping length (see Section 2.1.3) based on read length, and provide a comprehensive exon junction database (Section 1.3.1) during mapping. A remapping step can further rescue more reads mapped to novel junctions. To make the program even more sensitive to micro-exons, the minimum exon length can be decreased to as small as 6 nt using option “--min-exon” (the default is 9 nt). The accuracy of exon junction discovery can also be further improved by applying more stringent filters based on the numbers of supporting reads and anchor sizes.

To find a balance between sensitivity, accuracy and speed, we limit the *de novo* exon junction search to canonical (GT/AG) splice sites, because they account for about 99% of the mammalian splice sites. In addition, by default, we require prefect matches during certain steps in which querying of short sequences is involved (e.g., seed mapping, single-anchor search and small exon identification) to avoid very extensive search. However, these heuristics have relatively minimal effects on sensitivity in practice, according to the results in both simulated and real data tests.

The protocol presented in this paper can be readily combined with other tools for downstream analysis. OLego outputs the alignments in the standard SAM format which can be easily manipulated by SAMtools (e.g., converted to BAM files). Therefore, alignment results can serve as input for various tools for a range of different types of analysis, such as transcriptome reconstruction, differential gene expression and alternative splicing.

**ACKNOWLEDGMENTS:**

The work was supported by grants from National Institutes of Health (NIH) [R00GM95713 to C.Z.] and Simons Foundation Autism Research Initiative (297990).

**DISCLOSURES:**

The authors have no conflict of interests to disclose.

**REFERENCES**:

1 Wang, Z., Gerstein, M. & Snyder, M. RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet* **10**, 57-63, doi: 10.1038/nrg2484 (2009).

2 Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L. & Wold, B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat Meth* **5**, 621-628, doi: 10.1038/nmeth.1226 (2008).

3 Trapnell, C., Pachter, L. & Salzberg, S. L. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics* **25**, 1105-1111, doi: 10.1093/bioinformatics/btp120 (2009).

4 Wang, K. *et al.* MapSplice: accurate mapping of RNA-seq reads for splice junction discovery. *Nucleic Acids Res* **38**, e178, doi: 10.1093/nar/gkq622 (2010).

5 Au, K. F., Jiang, H., Lin, L., Xing, Y. & Wong, W. H. Detection of splice junctions from paired-end RNA-seq data by SpliceMap. *Nucleic Acids Res* **38**, 4570-4578, doi: 10.1093/nar/gkq211 (2010).

6 Huang, S. *et al.* SOAPsplice: genome-wide ab initio detection of splice junctions from RNA-Seq data. *Front Genet* **2**, doi: 10.3389/fgene.2011.00046 (2011).

7 Zhang, Y. *et al.* PASSion: a pattern growth algorithm-based pipeline for splice junction detection in paired-end RNA-Seq data. *Bioinformatics* **28**, 479-486, doi: 10.1093/bioinformatics/btr712 (2012).

8 Wu, T. D. & Nacu, S. Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics* **26**, 873-881, doi: 10.1093/bioinformatics/btq057 (2010).

9 Langmead, B., Trapnell, C., Pop, M. & Salzberg, S. L. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol* **10**, R25, doi: 10.1186/gb-2009-10-3-r25 (2009).

10 Wu, J., Anczukow, O., Krainer, A. R., Zhang, M. Q. & Zhang, C. OLego: fast and sensitive mapping of spliced mRNA-Seq reads using small seeds. *Nucleic Acids Res* **41**, 5149-5163, doi: 10.1093/nar/gkt216 (2013).

11 Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078-2079, doi: 10.1093/bioinformatics/btp352 (2009).

12 Trapnell, C. *et al.* Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotechnol* **28**, 511-515, doi: 10.1038/nbt.1621 (2010).

13 Anders, S., Pyl, P. T. & Huber, W. *HTSeq: A Python framework to work with high-throughput sequencing data*. (2014).

14 Anders, S. & Huber, W. Differential expression analysis for sequence count data. *Genome Biol* **11**, R106, doi: 10.1186/gb-2010-11-10-r106 (2010).

15 McCarthy, D. J., Chen, Y. & Smyth, G. K. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Res* **40**, 4288-4297, doi: 10.1093/nar/gks042 (2012).

16 Grabherr, M. G. *et al.* Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat Biotechnol* **29**, 644-652, doi: 10.1038/nbt.1883 (2011).

17 Zerbino, D. R. & Birney, E. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res* **18**, 821-829, doi: 10.1101/gr.074492.107 (2008).

18 Ferragina, P. & Manzini, G. Opportunistic data structures with applications. *Proc FOCS 2000*, 390-398 (2000).

19 Dobin, A. *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* **29**, 15-21, doi: 10.1093/bioinformatics/bts635 (2013).

20 Kim, D. *et al.* TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol* **14**, R36, doi: 10.1186/gb-2013-14-4-r36 (2013).

21 Grant, G. R. *et al.* Comparative analysis of RNA-Seq alignment algorithms and the RNA-Seq unified mapper (RUM). *Bioinformatics* **27**, 2518-2528, doi: 10.1093/bioinformatics/btr427 (2011).

22 Irimia, M. *et al.* A highly conserved program of neuronal microexons is misregulated in autistic brains. *Cell* **159**, 1511-1523, doi: 10.1016/j.cell.2014.11.035 (2014).

23 Li, Y. I., Sanchez-Pulido, L., Haerty, W. & Ponting, C. P. RBFOX and PTBP1 proteins regulate the alternative splicing of micro-exons in human brain transcripts. *Genome Res* **25**, 1-13, doi: 10.1101/gr.181990.114 (2015).