

Using RAMPAGE to identify and annotate regulatory elements in insect genomes

R. Taylor Raborn^{*1,2} and Volker P. Brendel^{1,2}

¹Department of Biology, Indiana University

²School of Informatics and Computing, Indiana University

Department of Biology and School of Informatics and Computing,
Indiana University

212 S. Hawthorne Drive 205 Simon Hall, Bloomington, IN 47401, USA
<http://www.brendelgroup.org>

Abstract. Application of Transcription Start Site (TSS) profiling technologies, coupled with large-scale next-generation sequencing (NGS) has yielded valuable insights into the location, structure and activity of promoters across diverse metazoan model systems. In insects, TSS profiling has been used to characterize the promoter architecture of *D. melanogaster*, and, shortly thereafter, to reveal widespread transposon-driven alternative promoter usage.

In this chapter we highlight the utility of one TSS profiling method, RAMPAGE (RNA annotation and mapping of promoters for analysis of gene expression), for the precise, quantitative identification of promoters in insect genomes. We demonstrate this using our tools GoRAMPAGE and TSSrchitect, providing details instructions with the aim of taking the user from raw reads to processed results.

Keywords: *cis*-regulatory regions, promoter architecture, transcription initiation, transcription start sites (TSSs)

1 Introduction

1.1 TSS Profiling Identifies Promoters at Genome-Scale

The promoter, defined in eukaryotes as the genomic region bound by RNA Polymerase II immediately prior to transcription initiation [1], is the site where regulatory signals unite to direct gene expression. The identification of promoter regions is a valuable step for understanding the *cis*-regulatory signals that are present in an organism, and is important for genome annotation. However, despite the rapid accumulation of genome sequences across metazoan and arthropod diversity, accurate annotation of promoter regions remains sparse. This is because—empirical mapping of TSSs—precisely identifying sequence motifs that demarcate the promoter is unreliable. In contrast with current *in*

* Correspondence: rtraborn@indiana.edu

silico approaches, direct mapping of TSSs identifies the location of the core promoter. Cap Analysis of Gene Expression (CAGE) [2], one of the first methods devised to identify 5'-ends of mRNAs at large-scale, involves selective capture of 5'-capped transcripts, first-strand reverse-transcription and ligation of a short oligonucleotide (CAGE tag). CAGE was initially utilized by the FANTOM (Functional Annotation of the Mammalian Genome) consortium to identify promoter architecture in human and mouse [3], providing the first glimpse of the global landscape of transcription initiation. At the onset of the NGS era, CAGE was coupled with massively-parallel sequencing to generate 5'-ends of mRNAs at substantially higher scale. This advance provided more extensive coverage of the expressed transcriptome, and provided increased sensitivity for quantitative measurements *i.e.* measurement of promoter activity.

1.2 Promoter Architecture of *Drosophila melanogaster*

Hoskins and colleagues [4] performed CAGE in *D. melanogaster* as part of the modENCODE consortium, identifying promoters at large-scale and characterizing the promoter architecture of an insect genome for the first time. Hoskins [4] indicated that TSS distributions at *Drosophila* promoters exhibit a range of shapes that can be generally grouped into two major classifications: *peaked* and *broad*. Peaked promoters have a single, major TSS position occupying a narrow genomic region, whereas broad promoters lack a single, major TSS and contain TSSs across a wider region [5][6]. The authors also showed a strong association between promoter class and motif composition (consistent with previous findings [5, 7]). Peaked promoters were associated with positionally-enriched *cis*-regulatory motifs including TATA, Initiator (Inr) and DPE, while broad promoters contained an enrichment of less-well characterized motifs, including *Ohler6* and *Ohler7* [8]. The existence of two promoter classes appears to be conserved among metazoans, and has been reported (using TSS profiling methodologies) in insects, cladocerans [9], fish [10] and mammals [11, 6].

1.3 Promoter Structure of Insects

Beyond *D. melanogaster*, few investigations have utilized TSS profiling in insect genomes. As a consequence, what is known about promoter architecture in insects is largely restricted to the *Drosophila* genus. As part of the modENCODE effort, CAGE was performed in multiple tissues and developmental stages of the *Drosophila pseudoobscura*. TSSs were found to be highly similar between species: more than 80% of TSSs (81%) of aligned, CAGE-identified TSSs from *D. pseudoobscura* were positioned within 20nt of their counterparts in *D. melanogaster*. An enrichment of the CA dinucleotide was detected at the TSS ($[-1, +1]$), and the motifs corresponding to TATA, Inr and DPE were positioned at the same locations relative to the TSS in both species. The one other insect species for which TSS profiling has been applied is the Tsetse fly (*Glossina morsitans morsitans*) [12]. Using TSS-seq (specifically Oligo-capping; for details on this method see [13]), the authors identified 3134 mapping to 1424 genes. The authors found

a preference for CA and AA dinucleotides at the TSS, and observe the major core promoter elements observed in *Drosophila*: TATA, Inr, DPE, in addition to MTE (Motif Ten Element). As in *D. melanogaster*, peaked promoters were more likely to contain TATA and Inr than broad promoters. While the taxonomic sampling of species for TSS profiling has been limited, the existing studies are sufficient to provide a general picture of insect promoter architecture. A major demarcation between the promoter architecture of insects and mammals appears to be the large fraction of mammalian promoters found in CpG islands [12]. CpG island promoters (CPIs) form the largest class of promoter in mammals [14]; by contrast, CPIs are not known to exist as a class in invertebrates.

1.4 Paired-end TSS Profiling with RAMPAGE

The most recent major methodological advance in TSS Profiling is RAMPAGE (RNA Annotation and Mapping of Promoters for the Analysis of Gene Expression) . RAMPAGE is a protocol for 5'-cDNA sequencing that combines cap trapping and template-switching with paired-end sequence information. A key advantage of generating paired-end sequence is transcript connectivity, which provides a direct link between a given 5'-end and its associated mRNA molecule. Because short or spurious RNAs are found within the transcriptome, transcript connectivity allows the TSSs (and thus promoters) of full-length mRNAs to be unambiguously identified, which benefits genome annotation. Batut and colleagues generated libraries from total RNA isolated from 36 stages across the life cycle of *D. melanogaster* providing a comprehensive gene expression and promoter atlas for fruit fly and in the process demonstrating the utility of RAMPAGE. RAMPAGE is currently being applied as part of the latest iteration of ENCODE to identify promoters in human, but as of this writing it has not been applied to any non-*Drosophila* insect species. In anticipation of the future application of TSS profiling into other insect model systems here we provide a documented protocol for the computational processing RAMPAGE data, using selected libraries from Batut *et al.*. This method will consist of two parts: first, we will process, filter and align the sequenced RAMPAGE libraries to the *D. melanogaster* genome. Second, we will identify TSSs and promoters from the aligned sequences and associate them with coding regions. In closing, we will consider further applications of this data and discuss the utility of reproducible workflows in bioinformatic analysis.

2 Materials

The analyses described herein require a workstation capable for modern bioinformatics. An intermediate understanding of the Linux/Unix command line will be extremely useful, although we make efforts to explain the procedures with clarity. In addition, it will likely be necessary for the participant to have superuser privileges on the machine. If you do not have a machine (or access to one) that meets

these requirements, it is recommended that you consider cloud-based cyberinfrastructure, including Amazon Web Services (AWS; <https://aws.amazon.com/>) or CyVerse (<http://www.cyverse.org/>). The former is a well-known pay-per-use solution, while the latter is an NSF-funded resource that is made freely available to the public.

2.1 Hardware Requirements

- x86-64 compatible processors
- At least 8GB RAM
- 30GB+ hard disk space

2.2 Software Requirements

- Operating system: 64 bit Linux (preferred) or Mac OS X (with Command Line Tools from XCode)
- R (version 3.4)
- Bioconductor (version 3.5)
- FASTX-Toolit (version 0.0.13)
- Samtools (version 1.3 or above)
- SRA Toolkit (version 2.3.4-2 or above)
- STAR aligner (version 2.4 or above)
- TagDust (version 2.33)

2.3 Installation of R packages

For installation of the software listed above, please follow the instructions provided by each respective package. Part of our analysis will require the use of R packages found in the Bioconductor suite. To install Bioconductor, please type the following from an R console:

```
source("https://bioconductor.org/biocLite.R")
biocLite()
```

We will use the R package *TSRchitect* to identify promoters from aligned RAMAPGE libraries. First, we will need to install a series of prerequisite packages to *TSRchitect* from Bioconductor. Please install these packages as follows (as before, from an R console):

```
source("https://bioconductor.org/biocLite.R")
biocLite(c("AnnotationHub", "BiocGenerics", "BiocParallel",
"ENCODEExplorer", "GenomicAlignments", "GenomeInfoDb",
"GenomicRanges", "IRanges", "methods",
"Rsamtools", "rtracklayer", "S4Vectors",
"SummarizedExperiment"))
```

To install *TSRchitect*, please type the following from an R console:

```

131 source("https://bioconductor.org/biocLite.R")
132 biocLite("TSRchitect")

```

133 Finally, please confirm that TSRchitect has been installed correctly by load-
 134 ing it from your R console as follows:

```

135 library(TSRchitect)

```

136 3 Methods

137 3.1 Retrieving the RAMPAGE sequence data from NCBI's 138 Gene Expression Omnibus (GEO)

139 To begin our analysis, we must download the RAMPAGE data to our work-
 140 station. We will utilize tools provided by the SRA Toolkit, which should
 141 already be installed on your machine (see **Materials**). The command *fastq-*
 142 *dump* allows one to directly retrieve data from the GEO database using
 143 the appropriate identifier(s). While there are 36 RAMPAGE libraries in the
 144 Batut *et al.* dataset, we will select a subset of these to analyze here. We
 145 will compare samples from selected embryonic (E01h-E03h) and larval (L1-
 146 L3) tissues, representing the beginning and end of embryonic development.
 147 For more information about the experiment and the available RAMPAGE li-
 148 braries, please see the following link: <https://www.ncbi.nlm.nih.gov/Traces/study/?acc=SRP011193>
 149 First, let's proceed with the libraries from early embryonic tissues. Note
 150 that since these fastq files are paired-end, we use the argument *-split-files*
 151 to generate separate files for each read pair.

```

152 mkdir fastq_files #creating a new folder to house the downloaded files
153 cd fastq_files #moving into this directory
154 fastq-dump --split-files SRR424683
155 fastq-dump --split-files SRR424684
156 fastq-dump --split-files SRR424685

```

157 We continue by downloading the RAMPAGE libraries from late embryonic
 158 tissues:

```

159 fastq-dump --split-files SRR424707
160 fastq-dump --split-files SRR424708
161 fastq-dump --split-files SRR424709

```

162 Once the download of the aforementioned files are complete, you should see
 163 a total of 12 (6x2) separate fastq files in your current working directory:

```

164 ls -l *.fastq | wc -l

```

3.2 Creating symlinks to the files

Our workflow expects fastq files that have the format “*.R1/R2.clipped.fq”. Rather than rename them, we can simply create brand new symbolic links to the files, as follows:

```
mkdir symlinks
ln -s SRR424683_1.fastq symlinks/E01h.R1.clipped.fq #embryonic libraries
ln -s SRR424683_2.fastq symlinks/E01h.R2.clipped.fq
ln -s SRR424684_1.fastq symlinks/E02h.R1.clipped.fq
ln -s SRR424684_2.fastq symlinks/E02h.R2.clipped.fq
ln -s SRR424685_1.fastq symlinks/E03h.R1.clipped.fq
ln -s SRR424685_2.fastq symlinks/E03h.R2.clipped.fq

ln -s SRR424707_1.fastq symlinks/L1.R1.clipped.fq #larval libraries
ln -s SRR424707_2.fastq symlinks/L1.R2.clipped.fq
ln -s SRR424708_1.fastq symlinks/L2.R1.clipped.fq
ln -s SRR424708_2.fastq symlinks/L2.R2.clipped.fq
ln -s SRR424709_1.fastq symlinks/L3.R1.clipped.fq
ln -s SRR424709_2.fastq symlinks/L3.R2.clipped.fq
```

3.3 Downloading genomic data from *D. melanogaster*

Now that we have the fastq files from the RAMPAGE libraries downloaded and named appropriately, we now must retrieve the genome assembly and rRNA sequences from *D. melanogaster*. The genome assembly is required for aligning the RAMPAGE reads, and the rRNA sequences are required to filter out matching reads in the sequenced RAMPAGE libraries, since our sample is intended to contain only capped RNA transcripts. Please download the rRNA sequences from the link we provide below. These sequences were retrieved separately from Genbank at the NCBI database.

Please download the assembly from the ENSEMBL database as follows:

```
wget ftp://ftp.ensembl.org/pub/release-78/fasta/drosophila_melanogaster/dm
#uncompressing the file
gzip -d \texttt{Drosophila\_melanogaster.BDGP5.dna.toplevel.fa.gz}
```

The rRNA sequences are found at the following link: <https://iu.box.com/s/3a5lqbo58qlykhmqxw00h2uo>. You should see a file entitled "Dmel_rRNA.fasta" in your current directory.

```
head -n 3
>ref|\texttt{NR\_133562.1}| Drosophila melanogaster 28S ribosomal RNA (28S)
TTATATACAACCTCAACTCATATGGGACTACCCCTGAATTTAAGCATATTAATTAGGGGAGGAAAAGAA
ACTAACAAGGATTTCTTAGTAGCGGCGAGCGAAAAAGAAAACAGTTCAGCACTAAGTCACTTTGTCTATA
```

3.4 Filtering and alignment of RAMPAGE reads using GoRAMPAGE

At this stage we are ready to commence with the rRNA filtering and alignment of the RAMPAGE libraries. We will use GoRAMPAGE, a tool we developed, to perform these tasks in a concerted workflow. GoRAMPAGE runs TagDust [?] to remove rRNA and low-complexity reads, and uses STAR [?] to align RAMPAGE (or other paired-end) reads to a given genome assembly.

Preparing the output directory It will also be necessary to create an output directory under "outputDir" for the results. GoRAMPAGE expects the results of a given step to be in place prior to initiating a run, so we'll need to create the appropriate folders before proceeding. Please do this as follows:

```
mkdir output #omit if you already have an output directory selected
mkdir output/reads
mkdir output/reads/clipped
```

Setting up the GoRAMPAGE job Now, once this is complete, please copy the contents of the "symlinks" directory that you created earlier (*i.e.* all of the *.fq files) into the "clipped/" directory. Please refer to the script "GoRAMPAGE_script_MMB.sh" and (using a text editor) provide the appropriate paths to the genome assembly, output directory (see above) and rRNA sequences. Note that if you are running this on a cluster with a job scheduler you'll need to add the necessary headers to the top of the script and submit the job in the appropriate manner. The script can be executed as follows:

```
\texttt{./GoRAMPAGE\_script\_MMB.sh}
#alternatively 'sh \texttt{GoRAMPAGE\_script\_MMB.sh}'
```

If everything is working correctly you should start to see the results of the job being written to the file "errScript". You can inspect the progress during the run using the *less* command.

```
less -S errScript
```

Should the run fail before completion, any associated error messages will be printed to the errScript file. Once the job is complete, you should see the message "GoRAMPAGE job is complete!" appear on the command-line terminal.

Inspecting the rRNA filtering results To evaluate the results from Step 3 (rRNA filtering), please navigate to the top level of the "output" directory and open the file "LOGFILES". You'll see the recorded progress of the program Tagdust and a record of the results. We notice that (for the L3h library)

1046448 of reads (78.1%) were "extracted", meaning that slightly more than 20% of reads were removed because of matches with ribosomal sequences. The removed reads from all libraries are found in the "dusted_discard" directory, and the extracted reads are found in the current directory. Due to their sheer abundance within cells, ribosomal RNA sequences are an inevitable contaminant within TSS profiling libraries. For analysis purposes, it is important that these sequences be removed, which is what we have done here. Confident that this step was conducted successfully, we can proceed to the next step.

Evaluating the alignments The folder "alignments/" in your GoRAMAPGE output folder will now contain 6 .bam files, each representing the distinct RAMAPGE libraries selected for our analysis. Typing "ls -l" from the command line will show that these files are symlinks to the original alignment files found in the "STARoutput/" directory. "STARoutput/", as its name suggests, contains the output from the STAR alignment, and this includes the alignment files "*.sortedByCoord.out.bam", and four additional log files. The files with the suffix "*.STAR.Log.final.out" each contain a summary of the alignment, such as the number of input reads, the percentage of uniquely-mapped reads and the percentage of unmapped reads. An inspection of these log files indicates that the alignments have similar mapping rates (70-80%), a reasonable outcome for our purposes.

Now that our RAMPAGE libraries are filtered and aligned, we can commence with the second half of our analysis, which is identifying TSSs and promoters from RAMPAGE libraries.

3.5 Promoter identification from aligned RAMPAGE libraries

4 Notes

Acknowledgments

Disclosure Declaration

The authors declare that they have no competing interests.

5 Figures

For L^AT_EX users, we recommend using the *graphics* or *graphicx* package and the `\includegraphics` command.

Please check that the lines in line drawings are not interrupted and are of a constant width. Grids and details within the figures must be clearly legible and may not be written one on top of the other. Line drawings should have

277 a resolution of at least 800 dpi (preferably 1200 dpi). The lettering in figures
 278 should have a height of 2 mm (10-point type). Figures should be numbered and
 279 should have a caption which should always be positioned *under* the figures, in
 280 contrast to the caption belonging to a table, which should always appear *above*
 281 the table; this is simply achieved as matter of sequence in your source.

282 Please center the figures or your tabular material by using the `\centering`
 283 declaration. Short captions are centered by default between the margins and
 284 typeset in 9-point type (Fig. 1 shows an example). The distance between text
 285 and figure is preset to be about 8 mm, the distance between figure and caption
 286 about 6 mm.

287 To ensure that the reproduction of your illustrations is of a reasonable quality,
 288 we advise against the use of shading. The contrast should be as pronounced as
 289 possible.

290 If screenshots are necessary, please make sure that you are happy with the
 print quality before you send the files.

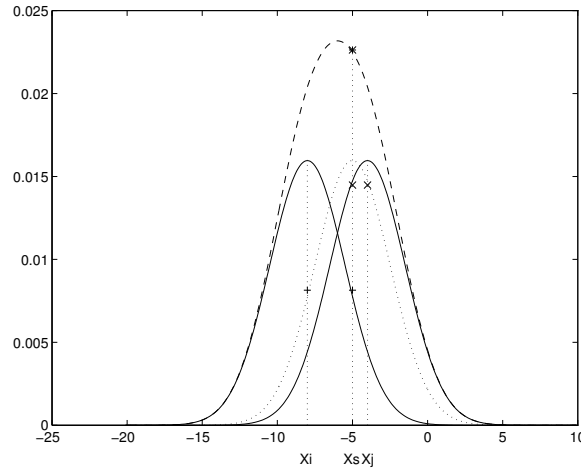


Fig. 1. One kernel at x_s (*dotted kernel*) or two kernels at x_i and x_j (*left and right*) lead to the same summed estimate at x_s . This shows a figure consisting of different types of lines. Elements of the figure described in the caption should be set in italics, in parentheses, as shown in this sample caption.

291 Please define figures (and tables) as floating objects. Please avoid using op-
 292 tional location parameters like “[h]” for “here”.
 293

294 5.1 Formulas

295 Displayed equations or formulas are centered and set on a separate line (with an
 296 extra line or halfline space above and below). Displayed expressions should be
 297 numbered for reference. The numbers should be consecutive within each section

or within the contribution, with numbers enclosed in parentheses and set on the right margin – which is the default if you use the *equation* environment, e.g.,

$$\psi(u) = \int_o^T \left[\frac{1}{2} (A_o^{-1}u, u) + N^*(-u) \right] dt . \quad (1)$$

Equations should be punctuated in the same way as ordinary text but with a small space before the end punctuation mark.

5.2 Footnotes

The superscript numeral used to refer to a footnote appears in the text either directly after the word to be discussed or – in relation to a phrase or a sentence – following the punctuation sign (comma, semicolon, or period). Footnotes should appear at the bottom of the normal text area, with a line of about 2 cm set immediately above them.¹

5.3 Program Code

Program listings or program commands in the text are normally set in typewriter font, e.g., CMTT10 or Courier.

Example of a Computer Program

```
program Inflation (Output)
  {Assuming annual inflation rates of 7%, 8%, and 10%,...
   years};
  const
    MaxYears = 10;
  var
    Year: 0..MaxYears;
    Factor1, Factor2, Factor3: Real;
  begin
    Year := 0;
    Factor1 := 1.0; Factor2 := 1.0; Factor3 := 1.0;
    WriteLn('Year 7% 8% 10%'); WriteLn;
    repeat
      Year := Year + 1;
      Factor1 := Factor1 * 1.07;
      Factor2 := Factor2 * 1.08;
      Factor3 := Factor3 * 1.10;
      WriteLn(Year:5,Factor1:7:3,Factor2:7:3,Factor3:7:3)
    until Year = MaxYears
  end.
```

(Example from Jensen K., Wirth N. (1991) Pascal user manual and report. Springer, New York)

¹ The footnote numeral is set flush left and the text follows with the usual word spacing.

334 5.4 Citations

335 For citations in the text please use square brackets and consecutive numbers:
336 [?], [?], [?] – provided automatically by L^AT_EX's `\cite ... \bibitem` mechanism.

337 5.5 Page Numbering and Running Heads

338 There is no need to include page numbers. If your paper title is too long to serve
339 as a running head, it will be shortened. Your suggestion as to how to shorten it
340 would be most welcome.

341 6 References

References

1. J. T. Kadonaga, "Perspectives on the RNA polymerase II core promoter." *Wiley Interdisciplinary Reviews: Developmental Biology*, vol. 1, no. 1, pp. 40–51, Jan. 2012.
2. R. Kodzius, M. Kojima, H. Nishiyori, M. Nakamura, S. Fukuda, M. Tagami, D. Sasaki, K. Imamura, C. Kai, M. Harbers, Y. Hayashizaki, and P. Carninci, "CAGE: cap analysis of gene expression." *Nature Methods*, vol. 3, no. 3, pp. 211–222, Mar. 2006.
3. P. Carninci, T. Kasukawa, S. Katayama, J. Gough, M. C. Frith, N. Maeda, R. Oyama, T. Ravasi, B. Lenhard, C. Wells, R. Kodzius, K. Shimokawa, V. B. Bajic, S. E. Brenner, S. Batalov, A. R. R. Forrest, M. Zavolan, M. J. Davis, L. G. Wilming, V. Aidinis, J. E. Allen, A. Ambesi-Impimbato, R. Apweiler, R. N. Aturaliya, T. L. Bailey, M. Bansal, L. Baxter, K. W. Beisel, T. Bersano, H. Bono, A. M. Chalk, K. P. Chiu, V. Choudhary, A. Christoffels, D. R. Clutterbuck, M. L. Crowe, E. Dalla, B. P. Dalrymple, B. de Bono, G. Della Gatta, D. di Bernardo, T. Down, P. Engstrom, M. Fagiolini, G. Faulkner, C. F. Fletcher, T. Fukushima, M. Furuno, S. Futaki, M. Gariboldi, P. Georgii-Hemming, T. R. Gingeras, T. Gojobori, R. E. Green, S. Gustincich, M. Harbers, Y. Hayashi, T. K. Hensch, N. Hirokawa, D. Hill, L. Huminiecki, M. Iacono, K. Ikeo, A. Iwama, T. Ishikawa, M. Jakt, A. Kanapin, M. Katoh, Y. Kawasaki, J. Kelso, H. Kitamura, H. Kitano, G. Kollias, S. P. T. Krishnan, A. Kruger, S. K. Kummerfeld, I. V. Kurochkin, L. F. Lareau, D. Lazarevic, L. Lipovich, J. Liu, S. Liuni, S. McWilliam, M. Madan Babu, M. Madera, L. Marchionni, H. Matsuda, S. Matsuzawa, H. Miki, F. Mignone, S. Miyake, K. Morris, S. Mottagui-Tabar, N. Mulder, N. Nakano, H. Nakauchi, P. Ng, R. Nilsson, S. Nishiguchi, S. Nishikawa, F. Nori, O. Ohara, Y. Okazaki, V. Orlando, K. C. Pang, W. J. Pavan, G. Pavesi, G. Pesole, N. Petrovsky, S. Piazza, J. Reed, J. F. Reid, B. Z. Ring, M. Ringwald, B. Rost, Y. Ruan, S. L. Salzberg, A. Sandelin, C. Schneider, C. Schönbach, K. Sekiguchi, C. A. M. Semple, S. Seno, L. Sessa, Y. Sheng, Y. Shibata, H. Shimada, K. Shimada, D. Silva, B. Sinclair, S. Sperling, E. Stupka, K. Sugiura, R. Sultana, Y. Takenaka, K. Taki, K. Tammoja, S. L. Tan, S. Tang, M. S. Taylor, J. Tegner, S. A. Teichmann, H. R. Ueda, E. van Nimwegen, R. Verardo, C. L. Wei, K. Yagi, H. Yamanishi, E. Zabarovsky, S. Zhu, A. Zimmer, W. Hide, C. Bult, S. M. Grimmond, R. D. Teasdale, E. T. Liu, V. Brusic, J. Quackenbush, C. Wahlestedt, J. S. Mattick, D. A. Hume, C. Kai, D. Sasaki, Y. Tomaru, S. Fukuda, M. Kanamori-Katayama, M. Suzuki, J. Aoki, T. Arakawa, J. Iida, K. Imamura, M. Itoh, T. Kato, H. Kawaji, N. Kawagashira, T. Kawashima, M. Kojima, S. Kondo, H. Konno, K. Nakano, N. Ninomiya, T. Nishio, M. Okada, C. Plessy, K. Shibata, T. Shiraki, S. Suzuki, M. Tagami, K. Waki, A. Watahiki, Y. Okamura-Oho, H. Suzuki, J. Kawai, Y. Hayashizaki, F. Consortium, R. G. E. R. Group, and G. S. G. N. P. C. Group, "The transcriptional landscape of the mammalian genome," *Science (New York, NY)*, vol. 309, no. 5740, pp. 1559–1563, Sep. 2005.
4. R. A. Hoskins, R. A. Hoskins, J. M. Landolin, J. M. Landolin, J. B. Brown, J. B. Brown, J. E. Sandler, J. E. Sandler, H. Takahashi, H. Takahashi, T. Lassmann, T. Lassmann, C. Yu, C. Yu, B. W. Booth, B. W. Booth, D. Zhang, D. Zhang, K. H. Wan, K. H. Wan, L. Yang, L. Yang, N. Boley, N. Boley, J. Andrews, J. Andrews, T. C. Kaufman, T. C. Kaufman, B. R. Graveley, B. R. Graveley, P. J. Bickel, P. J. Bickel, P. Carninci, J. W. Carlson, J. W. Carlson, S. E. Celniker,

- 390 and S. E. Celniker, "Genome-wide analysis of promoter architecture in *Drosophila*
391 *melanogaster*." *Genome Research*, vol. 21, no. 2, pp. 182–192, Feb. 2011.
- 392 5. E. A. Rach, H.-Y. Yuan, W. H. Majoros, P. Tomancak, and U. Ohler, "Motif
393 composition, conservation and condition-specificity of single and alternative tran-
394 scription start sites in the *Drosophila* genome." *Genome Biology*, vol. 10, no. 7, p.
395 R73, 2009.
- 396 6. B. Lenhard, A. Sandelin, and P. Carninci, "Metazoan promoters: emerging char-
397 acteristics and insights into transcriptional regulation." *Nature Reviews Genetics*,
398 vol. 13, no. 4, pp. 233–245, Apr. 2012.
- 399 7. T. Ni, D. L. Corcoran, E. A. Rach, S. Song, E. P. Spana, Y. Gao, U. Ohler,
400 and J. Zhu, "A paired-end sequencing strategy to map the complex landscape of
401 transcription initiation." *Nature Methods*, vol. 7, no. 7, pp. 521–527, Jul. 2010.
- 402 8. U. Ohler, G.-c. Liao, H. Niemann, and G. M. Rubin, "Computational analysis of
403 core promoters in the *Drosophila* genome." *Genome Biology*, vol. 3, no. 12, pp.
404 research0087.1–0087.12, 2002.
- 405 9. R. T. Raborn, K. Spitze, V. P. Brendel, and M. Lynch, "Promoter Architecture
406 and Sex-Specific Gene Expression in *Daphnia pulex*." *Genetics*, vol. 204, no. 2, pp.
407 593–612, Aug. 2016.
- 408 10. C. Nepal, Y. Hadzhiev, C. Previti, V. Haberle, N. Li, H. Takahashi, A. M. M.
409 Suzuki, Y. Sheng, R. F. Abdelhamid, S. Anand, J. Gehrig, A. Akalin, C. E. M.
410 Kockx, A. A. J. van der Sloot, W. F. J. van IJcken, O. Armant, S. Rastegar,
411 C. Watson, U. Strahle, E. Stupka, P. Carninci, B. Lenhard, and F. Muller, "Dy-
412 namic regulation of the transcription initiation landscape at single nucleotide res-
413 olution during vertebrate embryogenesis," *Genome Research*, vol. 23, no. 11, pp.
414 1938–1950, Nov. 2013.
- 415 11. P. Carninci, A. Sandelin, B. Lenhard, S. Katayama, K. Shimokawa, J. Ponjavic,
416 C. A. M. Semple, M. S. Taylor, P. G. Engström, M. C. Frith, A. R. R. For-
417 rest, W. B. Alkema, S. L. Tan, C. Plessy, R. Kodzius, T. Ravasi, T. Kasukawa,
418 S. Fukuda, M. Kanamori-Katayama, Y. Kitazume, H. Kawaji, C. Kai, M. Naka-
419 mura, H. Konno, K. Nakano, S. Mottagui-Tabar, P. Arner, A. Chesi, S. Gustincich,
420 F. Persichetti, H. Suzuki, S. M. Grimmond, C. A. Wells, V. Orlando, C. Wahlestedt,
421 E. T. Liu, M. Harbers, J. Kawai, V. B. Bajic, D. A. Hume, and Y. Hayashizaki,
422 "Genome-wide analysis of mammalian promoter architecture and evolution," *Na-
423 ture Genetics*, vol. 38, no. 6, pp. 626–635, Apr. 2006.
- 424 12. S. Mwangi, G. Attardo, Y. Suzuki, S. Aksoy, and A. Christoffels, "TSS seq based
425 core promoter architecture in blood feeding Tsetse fly (*Glossina morsitans mor-
426 sitans*) vector of Trypanosomiasis," *BMC Genomics*, vol. 16, no. 1, p. 722, Sep.
427 2015.
- 428 13. K. Tsuchihara, Y. Suzuki, H. Wakaguri, T. Irie, K. Tanimoto, S.-i. Hashimoto,
429 K. Matsushima, J. Mizushima-Sugano, R. Yamashita, K. Nakai, D. Bentley, H. Es-
430 umi, and S. Sugano, "Massive transcriptional start site analysis of human genes in
431 hypoxia cells," *Nucleic Acids Research*, vol. 37, no. 7, pp. 2249–2263, Apr. 2009.
- 432 14. N. Cvetesic and B. Lenhard, "Core promoters across the genome," *Nature Biotech-
433 nology*, vol. 35, no. 2, pp. 123–124, Feb. 2017.
- 434 15. T. Lassmann, "TagDust2: a generic method to extract reads from sequencing data,"
435 *BMC Bioinformatics*, vol. 16, no. 1, p. 1, Jan. 2015.
- 436 16. A. Dobin and T. R. Gingeras, "Optimizing RNA-Seq Mapping with STAR," in
437 *Transcription Factor Regulatory Networks*. New York, NY: Springer New York,
438 Apr. 2016, pp. 245–262.

439 In order to permit cross referencing within LNCS-Online, and eventually
 440 between different publishers and their online databases, LNCS will, from now
 441 on, be standardizing the format of the references. This new feature will increase
 442 the visibility of publications and facilitate academic research considerably. Please
 443 base your references on the examples below. References that don't adhere to this
 444 style will be reformatted by Springer. You should therefore check your references
 445 thoroughly when you receive the final pdf of your paper. The reference section
 446 must be complete. You may not omit references. Instructions as to where to find
 447 a fuller version of the references are not permissible.

448 We only accept references written using the latin alphabet. If the title of the
 449 book you are referring to is in Russian or Chinese, then please write (in Russian)
 450 or (in Chinese) at the end of the transcript or translation of the title.

451 The following section shows a sample reference list with entries for journal
 452 articles [?], an LNCS chapter [?], a book [?], proceedings without editors [?] and
 453 [?], as well as a URL [?]. Please note that proceedings published in LNCS are
 454 not cited with their full titles, but with their acronyms!

455 7 Checklist of Items to be Sent to Volume Editors

456 Here is a checklist of everything the volume editor requires from you:

- 457 ☐ The final L^AT_EX source files
- 458 ☐ A final PDF file
- 459 ☐ A copyright form, signed by one author on behalf of all of the authors of the
 460 paper.
- 461 ☐ A readme giving the name and email address of the corresponding author.