

Terminus enables the discovery of data-driven, robust transcript groups from RNA-seq data

Hirak Sarkar^{1,2}, Avi Srivastava³, Héctor Corrada Bravo^{1,2}, Michael I. Love^{4,5} and Rob Patro^{1,2,*}

¹Department of Computer Science and ²Center for Bioinformatics and Computational Biology, University of Maryland, College Park, MD 20742, USA, ³Department of Computer Science, Stony Brook University, Stony Brook, NY 11794, USA, ⁴Department of Biostatistics, University of North Carolina-Chapel Hill, Chapel Hill, NC 27510, USA and ⁵Department of Genetics, University of North Carolina-Chapel Hill, Chapel Hill, NC 27514, USA

*To whom correspondence should be addressed.

Abstract

Motivation: Advances in sequencing technology, inference algorithms and differential testing methodology have enabled transcript-level analysis of RNA-seq data. Yet, the inherent inferential uncertainty in transcript-level abundance estimation, even among the most accurate approaches, means that robust transcript-level analysis often remains a challenge. Conversely, gene-level analysis remains a common and robust approach for understanding RNA-seq data, but it coarsens the resulting analysis to the level of genes, even if the data strongly support specific transcript-level effects.

Results: We introduce a new data-driven approach for grouping together transcripts in an experiment based on their inferential uncertainty. Transcripts that share large numbers of ambiguously-mapping fragments with other transcripts, in complex patterns, often cannot have their abundances confidently estimated. Yet, the total transcriptional output of that group of transcripts will have greatly reduced inferential uncertainty, thus allowing more robust and confident downstream analysis. Our approach, implemented in the tool *terminus*, groups together transcripts in a data-driven manner allowing transcript-level analysis where it can be confidently supported, and deriving transcriptional groups where the inferential uncertainty is too high to support a transcript-level result.

Availability and implementation: *Terminus* is implemented in Rust, and is freely available and open source. It can be obtained from <https://github.com/COMBINE-lab/Terminus>.

Contact: rob@cs.umd.edu

Supplementary Information: [Supplementary data](#) are available at *Bioinformatics* online.

1 Introduction

RNA sequencing has become the *de facto* standard for analyzing transcriptomes, and has found myriad applications from differential expression analysis, to the discovery and assembly of new isoforms. Despite its widespread use, transcriptome analysis via RNA-seq poses a number of computational challenges. For example, reliably mapping or aligning short RNA-seq reads to the reference transcriptome and quantifying the abundance of transcripts is central to most typical RNA-seq analyses. Due to the nature of shared sequences in the reference transcriptome and genome, the simple problem of finding the locus of origin for a particular read sequence can be quite difficult. The complexity results both from alternative splicing, where the isoforms of a single gene can share multiple identical exons, and from very similar sequences arising in families of related genes. These challenges make the read sequence alone insufficient to determine the origin of the sequencing read. In such a scenario, a single read often maps equally well to multiple reference sequences. This ambiguity in determining the exact target sequence propagates directly to the process of quantification, where it becomes hard to

determine transcript-level expression when there is insufficient evidence to choose some transcripts over others as the true origin of sequencing reads.

To tackle these challenges, there has been tremendous growth in the space of computational tools that can effectively align (Dohin *et al.*, 2013; Kim *et al.*, 2013; Langmead and Salzberg, 2012) short RNA-seq reads to transcriptome and tools that can quantify (Diao *et al.*, 2014; Glam *et al.*, 2012; Li and Dewey, 2011; Patro *et al.*, 2014, 2017; Tress *et al.*, 2011) the abundance of transcripts. However, the inherent uncertainty in transcript abundance that results from ambiguous fragment alignment, even after attempting to model this uncertainty in either a maximum likelihood or Bayesian estimation framework, makes it difficult—and in some cases impossible—to provide a single accurate estimate for the number of reads originating from a specific transcript in a given sample. Gibbs sampling is a useful technique for estimating marginal or joint statistics of complex posterior distributions, and has been used by *rerseq* (Tress *et al.*, 2011), *BitSeq* (Glam *et al.*, 2012), *RSEM* (Li and Dewey, 2011) and *Salmon* (Patro *et al.*, 2017) in uncertainty quantification in expression estimation. This provides downstream

tools with the ability to analyze the full posterior, instead of just providing a point estimate.

Furthermore, different downstream tools such as mmDiff (Tanno *et al.*, 2014), IsoDE (Al Seem *et al.*, 2014), sleuth (Pimentel *et al.*, 2017) and wish (Zhu *et al.*, 2019) make use of these estimates in order to estimate differentially expressed transcripts or genes with higher accuracy and robustness. While taking samples from the posterior probability distribution provides some insight about the validity of the point estimates for transcript abundances, and this uncertainty can be propagated for the purposes of differential testing, it is often possible for a particular transcript to not exhibit expression in the point estimate altogether, in which case it becomes invisible to the analysis tools. Such cases were noted by Tanno *et al.* (2014), who further suggested grouping together transcripts whose abundance could not be confidently estimated into transcriptional groups. Specifically, it was demonstrated that there exist numerous cases where the abundance of an individual isoform cannot be reliably estimated, but the abundance of a small group of related isoforms can be determined accurately and robustly.

In a distinct context, but as a result of the same underlying cause of fundamentally multimapping reads, Robert and Watson (2013) note that these difficulties in mapping can lead to errors in quantification that affect genes of relevance to human disease. Crucially, they highlight that this issue occurs even at the level of genes, and is of concern even if one is not performing a transcript-level analysis. In addition to describing this issue, they identify specific groups of disease-related 938 genes that are affected by this problem, and suggest a gene-level analysis approach whereby groups of genes that share multimapping reads are tested jointly for the purposes of expression estimation and differential analysis. While this approach is quite robust, it is also very conservative, since it precludes transcript-level analysis altogether. Furthermore, depending on the set of all fragments sequenced in a sample, it may still be possible to confidently assess the abundance of a gene, or even a single transcript, even if it shares a large number of multimapping reads with other sequenced targets. There may then be utility in directly examining the posterior distributions of abundance estimates to determine when multimapping leads to a high degree of uncertainty in estimating expression, and when, despite the presence of multimapping reads, the abundance of a transcriptional target can be confidently assessed.

The mmCollapse tool (Tanno *et al.*, 2014) exploits the posterior samples generated by mmseq (Tanno *et al.*, 2011) to identify transcripts with highly anti-correlated posterior distributions. Some of the transcripts in these groups would otherwise not be properly estimated (we could have estimated abundances of 0), or would have such variable posterior estimates that they could neither be quantified confidently nor meaningfully tested for differential expression. However, when the transcripts are tested as inferential groups among the experimental samples, they can be robustly quantified and the group can be assessed for differential expression.

One of the major caveats of this approach is the particular choice of summary statistics used in order to identify similar groups. Specifically, mmCollapse attempts to group transcripts such that the minimum pairwise correlation is not too low. While this is a useful feature to assess, we find that it does not always accord with intuition about which transcripts should be grouped as it does not specifically account for inferential uncertainty would be reduced by grouping a specific pair of transcripts. Furthermore, the approach taken by mmCollapse is both extremely memory intensive (as all posterior samples for all expressed transcripts and for all samples in the experiment must be held in memory simultaneously) and quite time consuming, as the posterior correlations are computed in every iteration of the algorithm, even among completely unrelated transcripts that have little chance of producing promising candidates for grouping. Finally, mmCollapse is, arguably, overly conservative in the constraints it places on transcripts that may be considered as candidates for grouping. For example, only transcripts with no uniquely mapping reads are considered as potential candidates for collapse. However, we observe that even transcripts with a few uniquely mapping reads may exhibit a large degree of uncertainty in their

quantification estimates depending on the total number of reads mapping to such transcripts and the complexity of the patterns of multimapping with related transcripts.

Terminus, the tool presented in this article, attempts to address these shortcomings. It takes motivation from mmCollapse (Tanno *et al.*, 2014) as well as from the method proposed in ‘Surface simplification using quadric error metrics’ (Garland and Heckbert, 1997), a notable work in the field of computer graphics, in which densely tessellated shapes are simplified by approximating (coarsening) the mesh that represents the object. In ‘Surface simplification using quadric error metrics’, Garland and Heckbert (1997) argue that one way of achieving a visually appealing approximation is to start with the equivalent network of the visual model and repeatedly contract edges of the network in a manner that leads to minimal visual distortion of the overall shape.

In the same spirit, terminus reformulates the problem of discovering meaningful inferential groups as a graph simplification problem in which the (sparse) graph that defines what transcripts should be considered as candidates for collapsing is constructed by the read multimapping and conditional probability structure conveyed via the range-factorised equivalence classes (Zakari *et al.*, 2017) produced by Salmon. This avoids the need to even consider the vast majority of possible collapses. Further, terminus uses the reduction in inferential relative variance (Zhu *et al.*, 2019)—the reduction in inferential uncertainty that would result by grouping together pairs of transcripts—directly as a metric for optimisation. We show that this approach is extremely computationally efficient, and that it leads to groups of transcripts that are both biologically and inferentially meaningful. In complex transcriptomes (like human or mouse), our approach reduces the memory requirement by over two orders of magnitude compared to mmCollapse, and is simultaneously two orders of magnitude faster. We validate our results in both simulated and experimental datasets, and present time and memory benchmarks for running these tools.

2 Materials and methods

Numerous quantification tools, including mmseq and Salmon, encode the structure of mapping ambiguity in the form of equivalence classes. Terminus makes use of a collection of range-factorised equivalence classes (Zakari *et al.*, 2017) obtained from Salmon.

The overview of the mathematical model for Salmon is outlined in Supplementary Section S1. Here we would discuss the data structures that are relevant for terminus. Given a set of transcripts T and a set of read sequences R , a set of equivalence classes $\mathcal{E}$ is defined as a function from the domain of set of transcripts $t \in T$ to a natural number denoting the number of reads that are mapped to that group of transcripts, formally $\mathcal{E} : \mathcal{P}(T) \rightarrow \mathbb{N}^+$, where $\mathcal{P}(T)$ is a power set of all transcripts. In practice the size of the domain of $\mathcal{E}$ is much smaller than $2^{|T|} - 1$. Furthermore Salmon extends the notion of naive equivalence classes by adding the measure of mapping quality. Range-factorised equivalence classes can be defined as, $\mathcal{E} : \mathcal{P}(T \times \mathbb{N}^+) \rightarrow \mathbb{N}^+$. In effect each equivalence class consists of a set of pairs $\{t_i, w_i\}$ denoting transcript t_i and a number $w_i \in [0, 1]$ representing the average conditional probability with which the fragments in this equivalence class arose from transcript t_i . Depending on the granularity with which the range-factorised equivalence classes are defined, the equivalence relation between fragments is determined by the bin, within the range of conditional probabilities, into which they fall with respect to each transcript to which they map. Complete details of how this relation is defined can be found in Zakari *et al.* (2017). Along with these preliminary structures, Salmon also computes Gibbs chains $\mathcal{G}$, providing samples from the posterior distribution of the model. To be precise, each element $g_i \in \mathcal{G}$ contains the estimated fragment counts for transcript t_i taken over all (possibly shared) iterations of Gibbs sampling.

Our goal is to use the range-factorised equivalence classes $\mathcal{E}$ and the posterior samples $\mathcal{G}$ to determine groups of transcripts that exhibit high inferential uncertainty, and then to collect them together into robust transcriptional groups for which the posterior

uncertainty is considerably lower. To avoid the exponential space of possible groupings, we use the structure over transcripts induced by $\mathcal{E}$ to guide our search, and further restrict each iteration of the algorithm to perform a single pairwise collapse.

First, we collapse transcripts that appear in exactly the same set of equivalence class labels and that have near-identical conditional probability vectors. These are transcripts for which, even without examining the posterior samples, it is clear that no inference algorithm will have sufficient information to tell apart. Specifically, we accomplish this collapse using a partition refinement algorithm (Purge and Tarjan, 1987) where all transcripts start out within a single partition P_0 . We then iterate over $\mathcal{E}$ and determine the partitions that should be induced with respect to the current equivalence class $e \in \mathcal{E}$. This is simply the subset of transcripts that have nearly identical conditional probabilities with respect to e . For each such subset r in e , we refine the current partitioning P_i into P_{i+1} by replacing each set S_j in P_i that contain elements from r with $S_j \cup r$ and $S_j \setminus r$. This process is performed iteratively until we have processed all of the equivalence classes in $\mathcal{E}$.

Next, we define a graph $P = (V, E)$, constructed over $\mathcal{E}$ designed to encode the likely candidate transcripts for grouping. We define V to be the set of transcripts quantified, where $[u_i, v_j]$ is an edge in E if the transcript u_i and v_j co-occur in some equivalence class, and either of indication function $h(u_i)$ or $h(v_j)$ is true, and $s(u_i, v_j) \leq \tau$. Here $h(\cdot)$ is an indicator function that determines if a transcript is, when considered in isolation, a good candidate for possible grouping. We define $h(\cdot)$ as,

$$h(u_i) = \begin{cases} 1, & \text{if } \text{mean}(\mathbf{g}_i) \geq 1 \text{ and } \frac{\max(\mathbf{g}_i) - \min(\mathbf{g}_i)}{\text{mean}(\mathbf{g}_i)} > 1 \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where 1 is a user-defined parameter set to 0.1 by default. The score function $s(u_i, v_j) \leq \tau$ is designed to measure the improvement (decrease) in inferential uncertainty obtained by grouping u_i and v_j together. We define

$$s(u_i, v_j) = \text{infRV}_{\mathbf{g}_i + \mathbf{g}_j} = \frac{\text{infRV}_{\mathbf{g}_i} + \text{infRV}_{\mathbf{g}_j}}{2} \quad (2)$$

where,

$$\text{infRV}_{\mathbf{g}} = \frac{\max(\mathbf{g}, \text{var}(\mathbf{g}) - \text{mean}(\mathbf{g}))}{\text{mean}(\mathbf{g}) + \beta} + s \quad (3)$$

Where β is a pseudocount (we use 5 in tsemonia) and s is a small global shift (we use 0.01 in tsemonia). The $\text{infRV}_{\mathbf{g}}$ function measures the inferential relative variance, and the definition is taken from [Zhu et al. \(2019\)](#). The motivation for dividing by the mean is to stabilize the quantification of uncertainty across transcripts with low or high expression. Here, a negative value of $s(\cdot, \cdot)$ indicates that, when we sum the posterior samples of the two transcriptional units, the inferential relative variance is less than the average of the inferential relative variance of the individual units. When the value of $s(\cdot, \cdot)$ is sufficiently low (see [Fig. 1a](#)), then grouping these units together results in a substantial reduction in uncertainty by treating the pair of transcriptional units as a single group. A specific example of such collapse for two transcripts from human is shown in [Figure 2](#). In practice, we set the threshold for grouping (τ) in a data-driven manner by constructing a "background" distribution over the values resulting from evaluating $s(\cdot, \cdot)$ on a large number of randomly selected, expressed transcripts. We note that almost all pairs are not good candidates for collapsing, and so these random draws allow us to approximate sampling from the null distribution of scores for transcripts that should not be grouped. We call this background distribution $\mathcal{B}$.

Given a set of transcripts T , the possible number of transcript pairs to consider for collapse can be $O(|T|^2)$. For an organism with a large number of annotated transcripts (such as human or mouse), it is not computationally feasible to enumerate over such a large distribution. Therefore, for choosing a desirable threshold τ , we use an iterative sampling approach described in Algorithm 1. Specifically,

Algorithm 1: Threshold τ selection algorithm

```

Input  $T, \beta$ , selection_sim
Output  $\tau_{\text{max}}$ 
 $\tau_{\text{old}} \leftarrow 0$ ;
 $\tau_{\text{max}} \leftarrow 0$ ;
converged  $\leftarrow$  False;
 $B \leftarrow \{(i, j) | i, j \in T, s(i, j) = \text{selection\_sim}\}$ ;
 $B \leftarrow \emptyset$ ;
while not converged do
  for  $(i, j) \in B$  do
     $B = B \cup s(i, j)$ ;
  end
   $\tau_{\text{new}} \leftarrow \text{percentile}(B, 2.5)$ ;
   $\text{diff} \leftarrow |\tau_{\text{old}} - \tau_{\text{new}}|$ ;
  if  $\text{diff} < \delta$  then
    converged  $\leftarrow$  True;
  end
  selection_sim  $\leftarrow$  selection_sim  $\cdot \beta$ ;
   $\tau_{\text{old}} \leftarrow \tau_{\text{new}}$ ;
   $B \leftarrow \emptyset$ ;
end
return  $\tau_{\text{max}}$ ;

```

we exponentially increase the number of samples we draw until the resulting threshold τ that we would infer changes by less than some small quantity (we use 0.1%). The convergence of Algorithm 1 (steps plotted in [Fig. 1b](#)) ensures that the final threshold captures a close enough approximation to the true desired value. We set the threshold as a quantile of the background distribution, and choose 2.5% by default. The choice of the percentile is empirical, and chosen so that very few "independent" transcripts might be mistakenly grouped together. This parameter can be used to control sparsity of the graph.

[Figure 1a](#) plots the relation between the scoring function $s(i, j)$ defined in [Equation \(2\)](#) and sum of the means of Gibbs samples $\mathbf{g}_i$ and $\mathbf{g}_j$, corresponding to individual transcripts i and j . The red line shows where the empirical cut-off falls (which changes from one experiment to another). The points below the red line are candidates for grouping.

Given a graph P , a set Gibbs samples $\mathcal{G}$, and a threshold τ , tsemonia follows an iterative algorithm for collapsing transcripts. By construction of P , an edge $[u_i, v_j] \in E$ has three different attributes: (i) the score $s([u_i, v_j])$, (ii) a set of equivalence classes $\text{split}([u_i, v_j])$ where u_i and v_j co-occur and (iii) the total number of reads $c([u_i, v_j])$ that are shared between transcripts corresponding to nodes u_i and v_j . Tsemonia starts off by constructing a min-heap H ([Cormen et al., 2009](#)) over the set of edges E where the key for each edge is by the score function evaluated on the vertices sharing this edge. Tsemonia then iterates over H until it becomes empty, at each step collapsing the edge that was popped from the heap.

In each iteration i , starting with the current state of graph and the Gibbs samples, $(P_i, \mathcal{G}_i)$, tsemonia pops an edge $[u_i, v_j]$ from the heap with the minimum score and collapses the corresponding end points of the edge and produces a state $(P_{i+1}, \mathcal{G}_{i+1})$. The actual collapse process has a number of cases, but in essence, after collapsing, nodes u_i and v_j in the graph P , becomes a single node in P_{i+1} . Simultaneously, the corresponding vectors $\mathbf{g}_i$ and $\mathbf{g}_j$ from $\mathcal{G}_i$ are added to obtain $\mathbf{g}_{i+j}$ (to make collapsing more efficient, we associate the collapsed pair with some node f from the original endpoints) in $\mathcal{G}_{i+1}$.

The collapsing algorithm involves updating the information of the existing edges, and pushing appropriate edges on the heap. At any iteration, given a graph P , and the edge $[u_i, v_j]$ that has to be collapsed, tsemonia deletes the edge $[u_i, v_j]$ and includes either of the

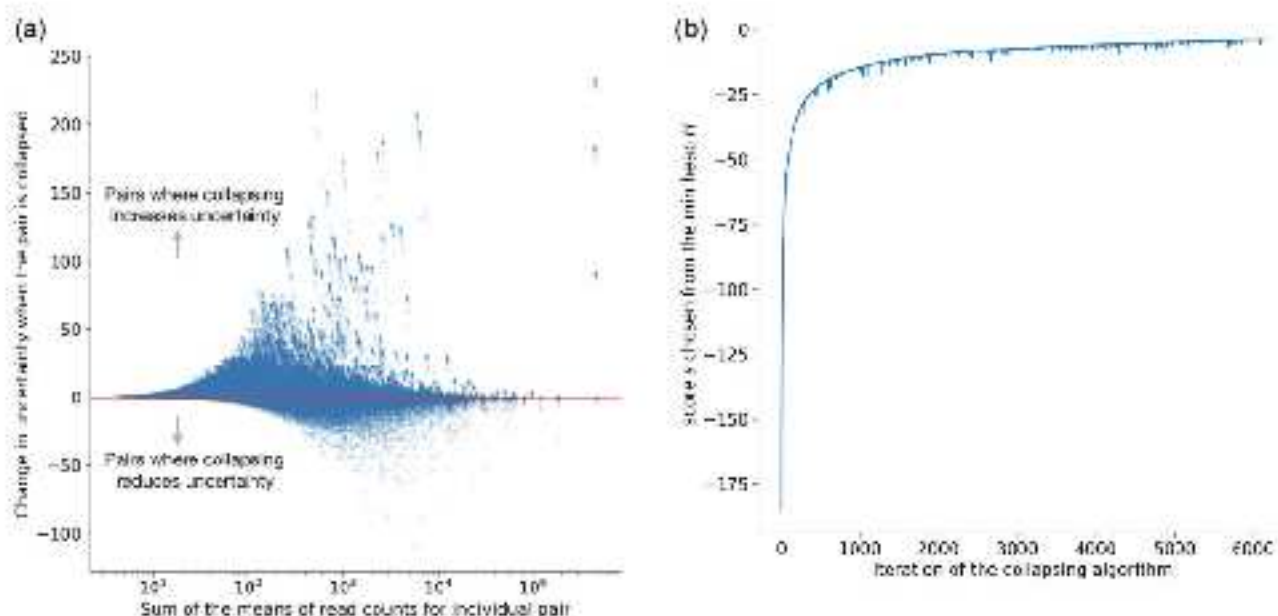


Fig. 1. The informational gain versus the sums of the candidate pairs and the convergence of terminus algorithm. (a) The reduction in uncertainty, defined by the score metric in Equation (2) versus the sum of the mean read counts for the individual transcripts of the pair. The cut-off is chosen in a data-driven manner. (b) The convergence of minimum InfRV in consecutive iterations

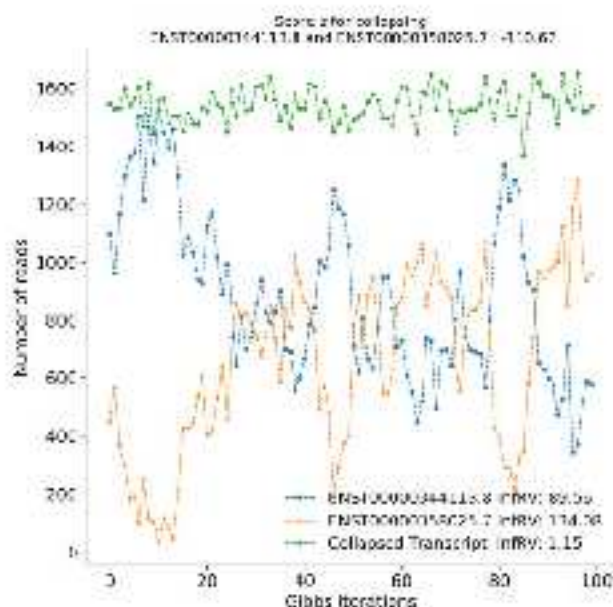


Fig. 2. Demonstration of uncertainty reduction by collapsing. Individual transcripts ENST00000344113.8 and ENST0000035025.7 are collapsed, and the corresponding InfRV is also reduced.

two nodes, say u_i (without loss of generality, in practice keeping the smaller numeric index, i.e. $i < j$) in R_{t+1} . Terminus updates G_{t+1} as $G_t \leftarrow G_t \cup E$.

The edge set of R_{t+1} is determined as follows given the set of adjacent nodes of u_i and u_j in P_t denoted as $\text{adj}(u_i)$ and $\text{adj}(u_j)$, any affected node $x \in \text{adj}(u_i) \cup \text{adj}(u_j)$ is handled in one of the three cases (See Fig. 3) below:

Case 1: Given $x \in \text{adj}(u_i)$ and $x \notin \text{adj}(u_j)$, an edge $\{xu_i\}$ is added to R_{t+1} , terminus recalculates $a(x, u_i)$. If $a(x, u_i)$ is smaller than τ , then the edge with the updated $a(x, u_i)$ is pushed into the min heap H . The $\text{cpl}(u_i)$ and corresponding counters for the pair remain unchanged as this edge already existed in R_t .

Case 2: Given $x \notin \text{adj}(u_i)$ and $x \in \text{adj}(u_j)$, a new edge $\{xu_i\}$ is added to R_{t+1} , terminus calculates $a(x, u_i)$. If $a(x, u_i)$ is smaller than τ , then the edge with the updated $a(x, u_i)$ is pushed into the min heap H . The $\text{cpl}(u_i)$ and corresponding counters for $\{xu_i\}$ are copied over to edge $\{xu_j\}$, since in the new round u_i and u_j are deemed to be identical.

Case 3: Given $x \in \text{adj}(u_i)$ and $x \in \text{adj}(u_j)$, since x, u_i and u_j forms a triangle in R_t . From all three edges only the edge $\{xu_i\}$ is added to R_{t+1} . Terminus calculates $a(x, u_i)$. If $a(x, u_i)$ is smaller than τ , then the edge with the updated $a(x, u_i)$ is pushed into the min heap H . The equivalence class for the newly added edge $\{xu_i\}$ is recalculated as: $\text{cpl}(xu_i) = \text{cpl}(xu_i) \cup \text{cpl}(xu_j)$. The union of the equivalence class lets merge the edge from over counting those equivalence classes that are shared between all three memberships x, u_i and u_j . In the same fashion, the reads that are shared between x and u_j are maintained in the edge $\{xu_j\}$. The shared read counts are calculated as follows:

$$c(\{xu_i\}) = c(\{xu_i\}) + c(\{xu_j\}) = \sum_{e \in \cap_{i,j} e} c(e)$$

Here, $\cap_{i,j} e$ signifies a set of shared equivalence classes.

All other nodes and edges remain unchanged in the iteration and are simply 'copied over' to R_{t+1} . Similarly, all elements of G_t would be copied over to G_{t+1} except the changed element g , and removal of the element g . We observe that the cardinality of P and G monotonically decreases, ensuring convergence.

Note that throughout the algorithm, the heap maintains the property that any edge currently present in the heap has a score less than empirical threshold τ . However, when one endpoint of an edge is modified, the corresponding score of the edge is not directly modified in the heap; such an edge is considered to be 'stale'. In order to keep the process of grouping efficient, terminus resets the same graph in all iterations, keeping state information (i.e. a 'last modified' timestamp) encoded as an attribute of the edge. When an edge is updated terminus just updates the corresponding attribute. When an edge is popped from the heap, we first check to ensure that it is not stale before processing the collapse. If the edge is stale, then we

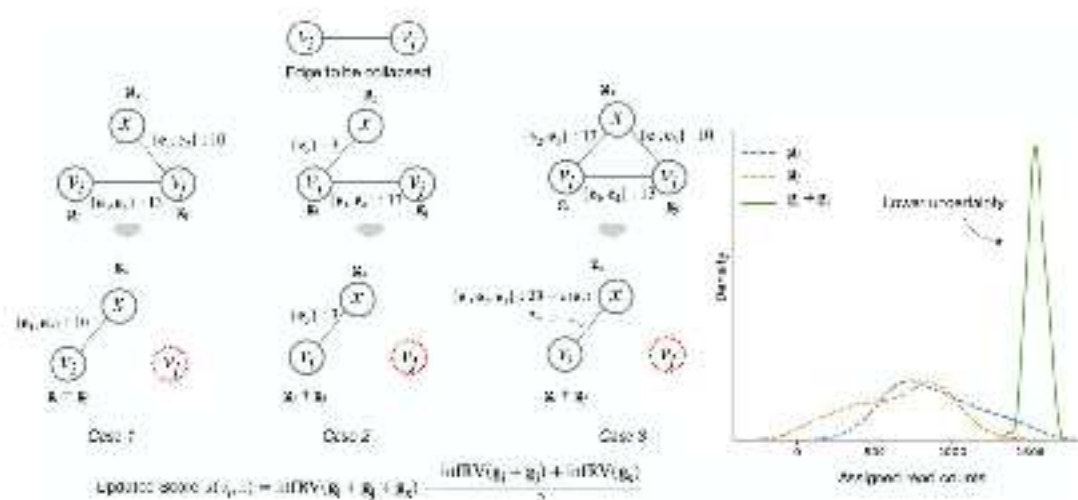


Fig. 3. Different possible scenarios that can arise while collapsing a node in graph R . In cases 1 and 2, as the edge property either same in same or transferred to the new edge. In case 3, the construction of edge explained in case 2 is either the problem of over counting already counted equivalence class (picking count of equivalence class $v_i(v_j)$). The right most plot shows the distribution of the individual Gtfs samples in dotted line along with the Gtfs samples of the collapsed group with much lower uncertainty.

re-evaluate its score and either perform the collapse or re-insert the edge in the heap.

Most typical RNA-seq experiments are comprised of multiple replicates. The process described above, however, is carried out individually per-sample. *Termius* takes a two-step approach in order to find coherent groups across the multiple replicates that comprise an experiment. It first groups the individual samples separately, and writes the groups for each sample. This step can be trivially parallelized. After obtaining individual groups, *termius* follows a consensus algorithm in order to find a set of groups to use across all samples.

The consensus procedure starts with individual groups, and constructs a union graph by treating each of the groups as a complete connected undirected graph (clique). For example given two different groups $[u_i, v_i, w_i]$ from one sample and $[u_j, v_j]$ from another, *termius* first constructs a weighted triangle with end points u_i, v_i and w_i where each edge has a weight 1. While considering the second group, *termius* increases the weight of edge $[u_i, v_j]$ by 1. This iterative procedure generates a weighted graph consisting any edge in the graph represents two transcripts that belong to the same group in at least one sample. *Termius* further prunes the union graph and removes the edges that have a weight below a user-defined threshold. The consensus mechanism ensures that a pair of transcripts should at least co-occur in a specified number of samples to qualify for the final grouping. Subsequently the final group that is common for all the samples is extracted by writing the connected components of the pruned graph. Note that these groups are deemed to be universal across samples and replicates. The default consensus threshold is $\frac{1}{w}$ where w is the number of samples in the experiment.

2.1 Datasets and evaluation

Different datasets, including both simulated and experimental data and varying across different organisms were used to demonstrate the benefit of using collapsed groups and the ability of *termius* to determine meaningful and robust groups.

Simulated dataset on Mouse. We used polyester *Pearce et al. (2013)*-generated simulated RNA-seq data, created by *Love et al. (2015)*, to assess accurate estimation when the true expression is known. The actual experiment was derived from a joint distribution of mean and dispersion values from GENCODE samples (*Lappalainen et al., 2013*). For the control experiment, we have chosen a 4 versus 4 subset from the original set of 12 versus 12 samples. The experiments consist of paired-end 100 base pair reads [see *Love et al. (2015)* for more details]. We refer to this dataset as *simulated 4 versus 4 mouse dataset*.

Simulated dataset with diploid transcriptomes from Mouse. The presence of a fully diploid transcriptome exacerbates the challenges caused by transcript sequence similarity. The mapping uncertainty in such case can greatly impact the quantification estimates. To capture such an extreme scenario, we have produced a diploid transcriptome following the same pipeline described by *Bagheri et al. (2018)*. The diploid transcriptome (named as the $N \times P$ transcriptome) combines a cross between NOD/ShiLAK (NOD) and PWK/Phj (PWK) strains of mice. The final transcriptome is obtained by joining progeny-mouse-references on the hybrid gtf and reference genome file. The hybrid gtf is produced by running *g2gtf* and *mouse* (*Bagheri et al., 2018*). The full script for producing such a transcriptome is provided in the repository. The paired-end read files from $N \times P$ transcriptome are generated using *polyester* with the true counts obtained from running *Salmon* on a real mouse RNA-Seq experiment (accession number SRR207106). We refer to this dataset as *simulated allelic mouse dataset*.

Experimental RNA-seq samples from *Drosophila* (2011): The experiment widely known as *pasilla* (*Bencks et al., 2011*) is an ensemble of 6 versus 6 RNA-seq experiment (with NCIMGEO accession numbers GSM461176 to GSM461181) that studies the effect of RNAi knockdown of *Pasilla*, which is the ortholog in *Drosophila melanogaster* of *NOVA1* and *NOVA2* mammalian genes. The same experiment is also used by *Toren et al. (2014)* to demonstrate the effect of grouping.

Given the above mixture of simulated and real experiments, we have run two sets of tools to produce groups: (i) *termius* on the output of *Salmon* and (ii) *revelotape*. There are different ways to produce input for *revelotape*. The *revelotape* run is preceded by a run of *revelotape*, which takes BAM files as input. To make the comparison of *revelotape* and *termius* as consistent as possible, we have used *Salmon*-generated BAM files for running the *revelotape* pipeline. For all experiments, *Salmon* is run with `-numCpuThreads 100` option in order to generate Gtfs samples. We used the `-hardFilter` parameter for producing the BAM files. In case of hits with different mapping scores, `-hardFilter` keeps only the hits with the best score. This parameter is chosen carefully to follow the equivalent Bowtie (*Largemad et al., 2009*) parameters mentioned in *Toren et al. (2011, 2014)*.

Evaluation of the quality of the collapsed groups is inherently a difficult task. Since there can be many possible groupings given a transcriptome dataset, comparing one grouping versus another requires biological validation. For simulated datasets, we validated the results by using the Spearman correlation and mean absolute relative difference (MARD) between the grouped estimates and corresponding true abundances. To be precise a given a grouping P is

defined as a partition over the set of all transcripts, allowing singleton partitions, denoting the un-grouped transcripts from their own groups. While assessing a particular grouping P , we induce the same partitioning over the ground truth abundances. Given a set of true read counts $p_{\text{true}} = [p_{11}, \dots, p_{1M}]$ for N transcripts and the estimated counts $p_{\text{est}} = [p_{21}, \dots, p_{2M}]$ where M is the number of groups (including singleton groups), we define a partition P induced on p_{true} as

$$P_{\text{true}} = \left\{ \sum_{i \in \psi_k} p_{1i}, \forall \psi_k \in P \right\}$$

We use P_{true} and p_{est} for each of the two tools evaluated in the current manuscript.

3 Results

3.1 Quantification comparison on simulated data

The two different simulated datasets that are used to demonstrate the utility of the grouping algorithm implemented in *Terminus* pose challenges of distinct natures. While the simulated 4 versus 4 human dataset is designed to capture aspects of real-world human transcript expression, the simulated allelic dataset from mouse represents the tremendous sequence ambiguity imposed by a diploid transcriptome and its resulting effect on quantification. We demonstrated that, on both the datasets, *Terminus* improves the accuracy of quantification results over *Salmon* and *rerollapase* under diverse metrics.

The simulated 4 versus 4 human dataset contains realistic GC bias estimated by *alpine* [Love et al., 2016] from experimental samples from the GEOVADIS study [Lappalainen et al., 2013]. If these realistic biases are not properly modeled, accurate quantification at the transcript level is impaired. By virtue of the design matrix, this experiment simulates differentially expressed transcripts and the nontrivial variability between the samples can also pose challenging problems to the collapsing algorithm. The global Spearman correlation and MARD values presented in Table 1 summarizes these metrics across eight different samples and takes the average. While *Salmon* itself performs fairly well in this dataset, we observe the groups induced by *Terminus* further improves the accuracy of abundance estimates. The poor performance of *rerollapase* is due to very noisy low abundance values given to truly overexpressed transcripts. The performance of *rerollapase* improves substantially when only considering truly expressed transcripts.

Figure 4 compares the quantification results at a more granular level under different constraints on the true and estimated counts. Among transcripts that are truly expressed, *rerollapase* more often mis-assigns reads compared to *Salmon* or *Terminus*. The skewness of histogram from *rerollapase* suggests that it tends to underestimate the true counts of transcripts. This effect is also visible in the corresponding scatter plot at the top left corner. The spread of the red points signify deviation from the true counts. The scatter plot from *Terminus* shrinks these mis-estimates toward the diagonal by putting them into groups improving the overall correlation.

A similar plot for simulated allelic dataset from mouse is shown in Figure 5. Owing to the diploid transcriptome, we see considerable ambiguity in the underlying sequence. It reduces the overall correlation for all the tools and further affects the convergence of the *rerollapase* collapsing algorithm. *Terminus* consistently produces

estimated counts closer to truth. The corresponding scatter plot captures the shrinkage of the transcripts groups toward the diagonal which are otherwise mis-estimated by *Salmon*.

Allelic imbalance Due to presence of highly similar pairs in a diploid transcriptome, it is often a challenging task for a quantification tool to assign reads to the correct allele, especially when both of the alleles are equally likely in terms of the confidence of the alignment. The proportion of two different alleles present in an experiment is often termed as the *allelic imbalance*. When the allelic imbalance is close to 0.5—when both alleles are expressed equally—accurate estimation becomes particularly difficult. The reason being the fact that there is almost equal prior for assigning a read to either of the candidates. In such cases, the maximum likelihood estimators often prefer one allele over the another, mis-estimating the resulting abundances. We pinpointed such cases by investigating cases where the true allelic imbalance is restricted to an interval of 0.45–0.55, focusing on the region where the uncertainty among the alleles is highest.

Supplementary Figure S8 plots the allelic imbalance predicted by *Salmon* (in y-axis ratio of the expression values for individual alleles estimated by *Salmon*) versus the true allelic imbalance (ratio of true expression of the alleles of a transcript). The color of the point is determined by the fact if the transcript is grouped by *Terminus* (dark blue when grouped and orange when not). To get a closer look at the mis-estimation problem in these cases, Supplementary Figure S9 zooms the x-axis to the range 0.45–0.55. We observe for transcripts with true allelic imbalance 0.5 *Salmon* either over-estimates or under-estimates the true allelic imbalance, resulting in the spread through the entire y-axis. Meanwhile, the cluster of the points near the very end of $y = 0.5$ vertical line suggests that in those case all the reads are assigned to one of the alleles leading to one allelic imbalance estimates of 0 and 1. As the color (signifying grouped/un-grouped status) suggests, in those cases, *Terminus* correctly identifies the present uncertainty and groups the alleles together.

To quantify the benefit of grouping in the context of allelic imbalance, we have considered the cases of mis-estimated dominant alleles. Specifically, such cases occur when the allelic imbalance ratio is reversed and the allele that truly has higher expression is estimated to have lower expression and vice versa. We measured the proportion (r_{mis}) of such cases versus the total number expressed alleles (counting an allele pair only once). We also measured the same metric after grouping. That is, we measured the ratio (r_{group}) where the numerator is the number of mis-estimated dominant alleles that are not grouped, and the denominator is the number of expressed alleles where neither allele from the pair was grouped. For the simulated allelic dataset, we found r_{mis} to be 0.1 and r_{group} to be 0.04, signifying the fact that *Terminus* is capable of preferentially grouping together alleles whose allelic ratios are highly uncertain, and where the estimates of *Salmon* are otherwise most likely to be incorrect.

3.2 Quantification on real datasets

The *Pastila* dataset is an experimental RNA-seq experiment in *Drosophila*. It comprises of both single-end and paired-end RNA-seq reads. Owing to rapid alternative splicing (AS), prevalent in *Drosophila*, the organism becomes specifically interesting in the context of evaluating uncertainty induced by extensive AS. Almost 20–37% multi-exon genes are alternatively spliced [Gillman et al., 2016]. The isoforms within a gene that share one or multiple long exons are often very hard to distinguish, resulting in low-confidence estimates. As the dataset does not contain the true quantification values, we measured other biological attributes to validate the grouping.

From the 30 397 transcripts, *Terminus* and *rerollapase* grouped 7904 and 4328 transcripts respectively, distributed in 3023 and 1833 groups. As expected, most of the transcripts in the group originate from the same gene. When groups contain transcripts from multiple genes, these genes belong to either the same gene family or to a closely related gene family.

Figure 6 depicts this phenomenon through a histogram plot. In Figure 6a, the x-axis is the number of genes to which all the transcript within one group can be mapped. For example, for groups

Table 1. Spearman correlation and mean absolute relative difference (MARD) are calculated with respect to ground truth under for both the simulated 4 versus 4 human dataset (termed as Human), and the simulated allelic dataset from mouse (termed as Mouse)

Dataset	Correlation (Spearman)			MARD		
	<i>Salmon</i>	<i>Rerollapase</i>	<i>Terminus</i>	<i>Salmon</i>	<i>Rerollapase</i>	<i>Terminus</i>
Human	0.94	0.79	0.96	0.11	0.15	0.09
Mouse	0.91	0.81	0.93	0.12	0.12	0.07

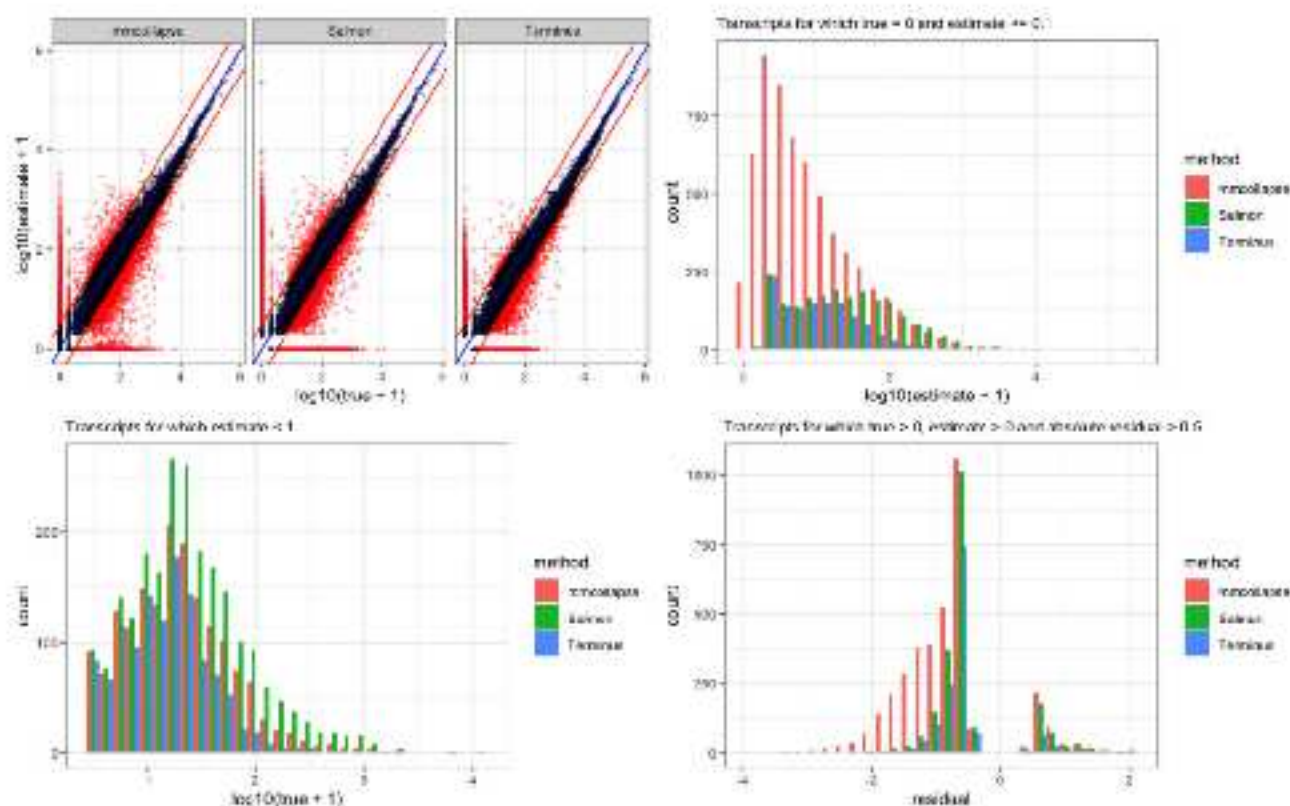


Fig. 4. A comparative view of three different tools on the simulated 4 versus 4 luciferase dataset. The scatter plot on the top right panel assesses the performance of at least one tool for three tools *rsem-collapse*, *Salmon* and *TMM*. The histogram at the top right corner shows the frequency of non-estimated transcripts which are not expressed in the ground truth. Here only those transcripts for which the tools have at least assigned one read are considered. This helps to get rid of the noise introduced by *rsem-collapse* that leads to degenerate values. The histogram at the bottom right shows the frequency of lowly expressed transcripts. The rest of the transcripts (that are expressed) are shown in the bottom right plot. Here the x-axis (plotted as residual) is the difference between the log transformed values of true and estimated expression. Negative residual values signify underestimation, while the positive values signify overestimation.

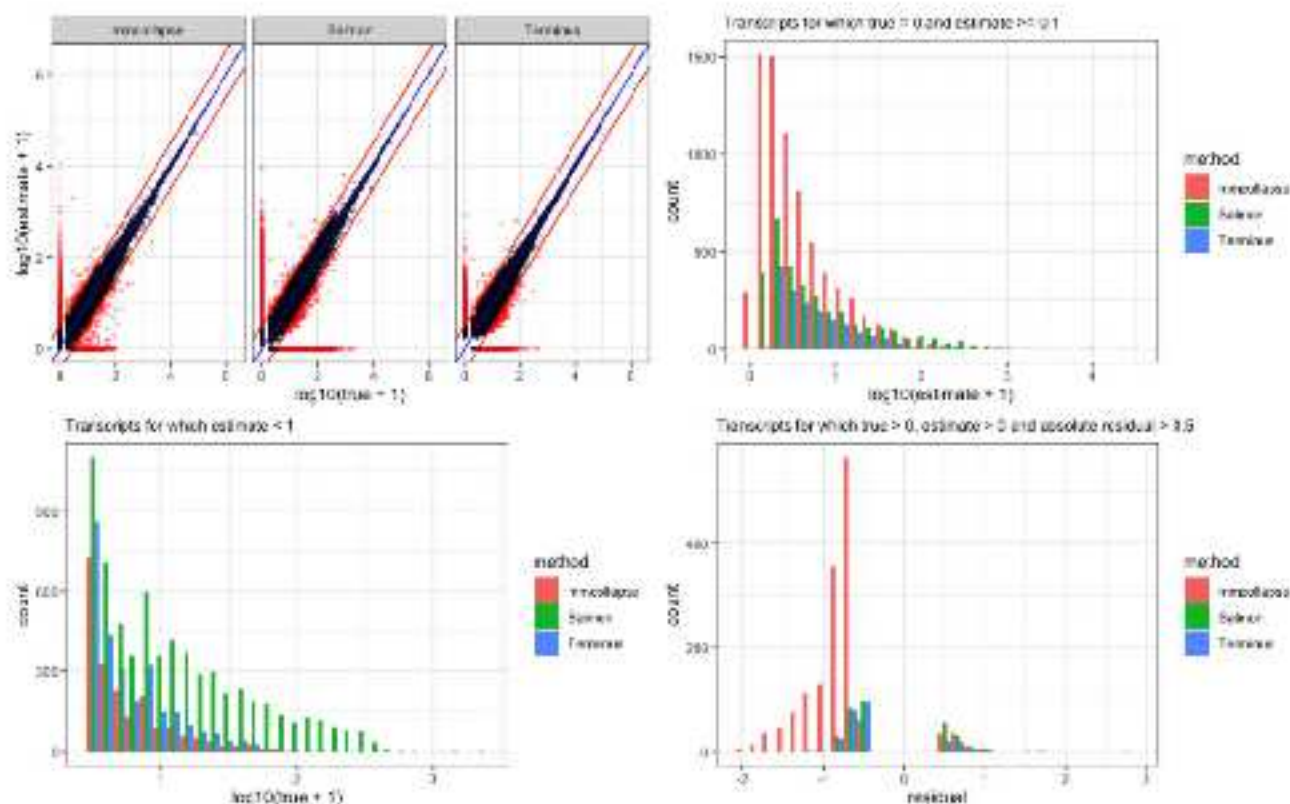


Fig. 5. Accuracy of *Salmon* with and without grouping and *rsem-collapse* on the simulated allelic dataset is shown in the plot. The structure is similar to that of **Figure 4**

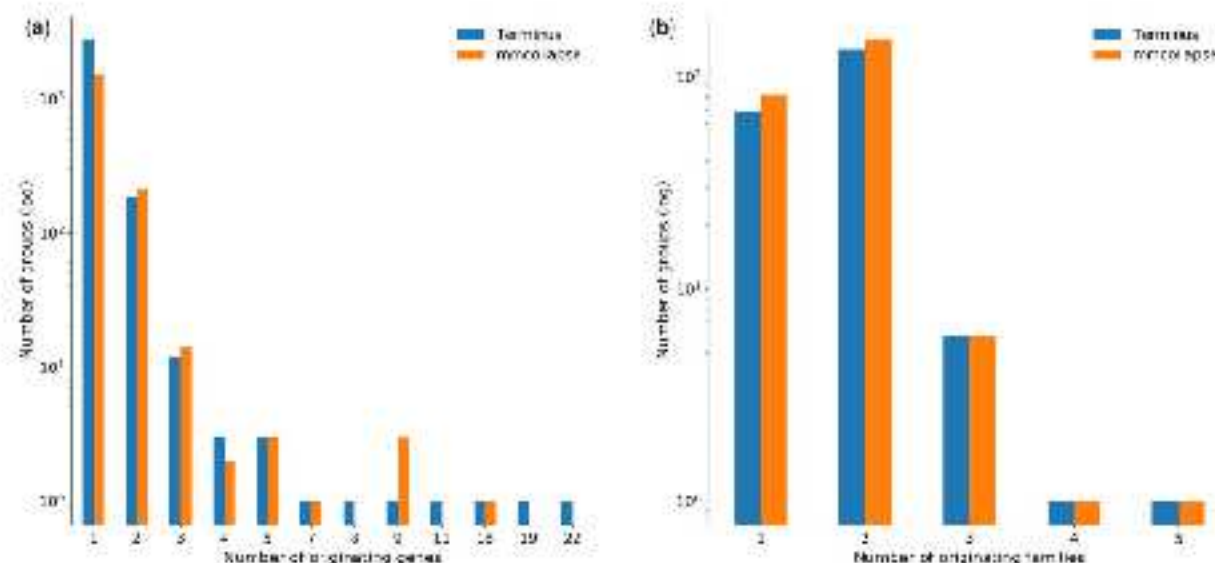


Fig. 6. Biological significance of the groups produced by Terminus and *rsemcollapse*. (a) The x-axis shows the number of genes that the transcript of the groups can be mapped back to, and the y-axis shows number of such groups. (b) The x-axis shows the number of gene families that the transcript can be mapped back to, and the y-axis is the frequency of such mappings.

that belong to the bar corresponding to $x=1$, all transcripts come from a single gene meaning that all the transcripts are isoforms of each other. Similarly, Figure 6b shows another level of summarization, where the transcripts of a group are mapped back to their gene families. We see in most of the cases that the groups can be mapped back to a small number of gene families. This behavior is expected, as gene families share considerable sequence information and are likely to give rise to related and consistent expression estimates. Note that *Terminus* is agnostic to the annotation of the underlying organism, yet the data-driven partitions inherently identify the genes and families annotated in the underlying reference.

We further observe the group sizes (defined by number of transcripts in a group) generated by *Terminus* (largest one being 54) tend to be larger than that of *rsemcollapse* (largest one being 19). Often these large groups also identify gene families that share large numbers of genes. One such example from the groups formed by *Terminus*, consisting 54 transcripts is from the *para* gene and from the same family of parathyroid hormone-related protein. This group of proteins are included in many other groups. Another such group comes from *gata* or *slowpoke* that regulates the release of a neurotransmitter.

3.3 Computational performance

Table 2 shows the computational performance of *Terminus* versus *rsemcollapse*. *Terminus* takes considerably less time to compute the groups and requires much lower memory to run. This enhanced performance derives from two key attributes of *Terminus*. First, *Terminus* does not consider collapsing transcript pairs that do not appear in any equivalence class. This results in the evaluation of many fewer pairs. Second, *Terminus* uses the underlying graph structure induced by the equivalence classes to order and prioritize the collapses, avoiding the need to constantly recompute candidate pairs. We observe that the performance of *rsemcollapse*, both in terms of running time and required memory, greatly varies from one sample to another. One possible reason for such behavior could be the variation in the number of non-unique transcripts. While, in the case of the simulated allelic dataset, *rsemcollapse* ran for a long time (we terminated the run after 24 h), for *Pan10a* dataset it finishes relatively quickly. However, we observe that the memory requirement of the tool often made it very difficult to test it with multiple threads. In those cases, the *rsemcollapse* run had to be restricted to 1 thread (e.g. on the simulated allelic dataset, running *rsemcollapse* with even 1 thread required ~21.6G of RAM). The enormous speed benefits of *Terminus* suggest that it can be easily incorporated as a

Table 2. The table shows construction time and memory requirements for *rsemcollapse* and *Terminus*.

Dataset	Memory (MB)		Time (h:m:s)	
	<i>rsemcollapse</i>	<i>Terminus</i>	<i>rsemcollapse</i>	<i>Terminus</i>
Human	370 675	2841	3:46:37	2:04
Mouse	225 457	485	38:55:44	0:12
Drosophila	4104	310	2:10	0:10

part of standard lightweight RNA-seq workflow for finding out groups, with very little computational overhead. We note that we have not included the time and memory requirement for *Salmon* and *rerseq* as *rsemcollapse* is the tool compared with *Terminus*.

4 Conclusion and future work

The presence of non-unique sequences can greatly affect the accuracy of transcript quantification. Careful analysis of the pooled or samples from the underlying probabilistic model not only provides a measure of uncertainty around a point estimate of abundance, but also indicates which groups of transcripts may have abundances that are particularly difficult to distinguish individually but which have estimable abundance as a group. *Terminus* demonstrates that Glib samples can be used to identify groups of transcripts that exhibit high inferential uncertainty on their own, but which exhibit much lower uncertainty as a group. We show how *Terminus* uses the information encoded in large-factored equivalence classes, readily available after quantification with *Salmon*, to tremendously accelerate the grouping process. *Terminus* writes the new expression estimates and pooled or samples with the group information in the same exact format as that of *Salmon*, enabling any downstream pipeline that accepts a similar format (Zhu et al., 2019) to directly run on *Terminus* output.

The groups computed by *Terminus* represent abundance estimates reported at the resolution that is actually supported by the underlying experimental data. In a typical experiment, this is neither at the gene level nor the transcript level. Some transcripts, even from complex, multi-isoform genes, can have their abundances accurately estimated and with low uncertainty, while other transcripts cannot. Rather than pre-defining the resolution at which the analysis will be

performed, and subjecting the results to either overfitting uncertainty or to insufficient biological resolution, *testimonium* allows the determination of transcriptional groups whose abundance can be confidently estimated in a given dataset, and represents, in this sense, a data-driven approach to transcriptome analysis.

Further, we demonstrate that *testimonium* creates biologically relevant groups that reflect the underlying hierarchy of genes and gene families. This shows the potential of *testimonium* to be utilized for applications of data-driven clustering of biological sequences, such as clustering *de novo* contigs (where the annotation is not known) or for clustering-related strains in metagenomic samples.

From a conceptual perspective, *testimonium* provides a novel approach for grouping complex interactions between biological sequence without having the prior information about the annotation itself. It first prunes the possible space of pairwise collapses by examining the structure induced by the single-factorial equivalence classes, and later, uses a iterative greedy technique to collapse transcripts that locally maximize the objective being optimized (i.e. the reduction in inferential relative variance). It is possible that this generic framework can be extended to other aspects of grouping and clustering such as taxonomic classification.

Funding

This work was funded by National Institute of Health [R01 EB022857 to M.L. and R.P., R01 MH113649, R01 CA141338, R01 CA188701 to M.L., R01 GM14167, R24 MH14311 to H.S.]. By National Science Foundation [PCN-1730472, CNS-1763623 to R.P.]. The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript.

Disclosure

R.P. is a co-founder of Ocean Genomics Inc.

Conflict of Interest: none declared.

References

Alizadeh, A. et al. (2014) Bootstrap-based differential gene expression analysis for RNA-Seq data with and without replicates. *BMC Genomics*, **15**, 52.

Brooks, A.N. et al. (2011) Conservation of a RNA regulatory map between *Drosophila* and *Caenorhabditis*. *Genome Res.*, **21**, 193–202.

Cormac, I.E. et al. (2009) *Introduction to Algorithms*. MIT Press, Cambridge, MA, USA.

Diao, E. et al. (2014) ORMAN: optimal resolution of ambiguous RNA-Seq multi-mappings in the presence of novel isoforms. *Bioinformatics*, **30**, 644–651.

Debin, A. et al. (2013) STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, **29**, 15–21.

Fraser, A.C. et al. (2013) Polycomb: simulating RNA-seq datasets with differential transcript expression. *Bioinformatics*, **31**, 1772–1784.

Garland, M. and Eckert, P.E. (1997) *Statistical significance using quadratic error metrics*. In: *Proceedings of the 1997 Annual Conference on Computer Graphics and Interactive Techniques*, pp. 209–216. Los Angeles, California, USA.

Gilchrist, J. et al. (2016) Alternative splicing within and between *Drosophila* species, sexes, tissues, and developmental stages. *PLoS Genet.*, **12**, e1006464.

Glass, E. et al. (2012) Identifying differentially expressed transcripts from RNA-seq data with biological variation. *Bioinformatics*, **28**, 1721–1728.

Kan, J. et al. (2013) EBAT: a fast aligned aligner with low memory requirements. *Nat. Methods*, **10**, 357–362.

Langmead, B. and Salzberg, S.L. (2012) Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.

Langmead, B. et al. (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.*, **10**, R25.

Lappalainen, T. et al. (2013) Transcriptome and genome sequencing uncovers functional variation in humans. *Nature*, **501**, 576–581.

Li, B. and Dewey, C.N. (2011) RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics*, **12**, 323.

Love, M.I. et al. (2016) Modeling of RNA-seq frequency across a broad range of systematic errors in transcript abundance estimation. *Nat. Biotechnol.*, **34**, 1287–1291.

Love, M.I. et al. (2014) Simultaneous downstream statistical analysis of differential transcript usage following mRNA quantification. *BioRxiv*, **7**, 932.

Paige, J. and Tarjan, R.E. (1987) Three partition refinement algorithms. *SIAM J. Comput.*, **16**, 973–993.

Park, J. et al. (2014) Isoform enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nat. Biotechnol.*, **32**, 462–464.

Park, J. et al. (2017) Salmon provides fast and bias-aware quantification of transcript expression. *Nat. Methods*, **14**, 417–419.

Placette, J.E. et al. (2017) Differential analysis of RNA-seq incorporating quantification uncertainty. *Nat. Methods*, **14**, 687–692.

Raghunathan, N. et al. (2013) Hierarchical analysis of RNA-seq reads improves the accuracy of allele-specific expression. *Bioinformatics*, **29**, 2177–2184.

Robert, C. and Wernke, M. (2013) Errors in RNA-Seq quantification affect genes of interest to human disease. *Genome Biol.*, **14**, 177.

Tsuru, R. et al. (2011) IsoType: a tool for isoform specific expression estimation using multi-mapping RNA-seq reads. *Genome Biol.*, **12**, R13.

Tsuru, R. et al. (2014) Flexible analysis of RNA-seq data using mixed effects models. *Bioinformatics*, **30**, 1021–1028.

Zakaria, M. et al. (2017) Improved data-driven likelihood formulations for transcript abundance estimation. *Bioinformatics*, **33**, 2612–2620.

Zhu, A. et al. (2013) Nonparametric expression analysis using inferential replicates. *Nucleic Acids Res.*, **41**, e128.