# A consensus-based ensemble approach to improve transcriptome assembly

Adam Voshall[1,2,3], Sairam Behera[2,4], Xiangjun Li[5,6], Xiao-Hong Yu[7], Kushagra Kapil[2], Jitender S. Deogun[2], John Shanklin[8], Edgar B. Cahoon[5,6], & Etsuko N. Moriyama[1,5]*

[1]School of Biological Sciences, University of Nebraska-Lincoln, Lincoln, NE 68588, USA

[2]Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588, USA

[3]Boston Children's Hospital/Harvard Medical School, Boston, MA 02115, USA

[4]Human Genome Sequencing Center, Baylor College of Medicine, Houston, TX 77030, USA

[5]Center for Plant Science Innovation, University of Nebraska-Lincoln, Lincoln, NE 68588, USA

[6]Department of Biochemistry, University of Nebraska-Lincoln, Lincoln, NE 68588, USA

[7]Department of Biochemistry and Cell Biology, Stony Brook University, Stony Brook, NY 11794, USA

[8]Biology Department, Brookhaven National Laboratory, Upton, NY 11973, USA

Adam Voshall[1,2,3], Sairam Behera[2], Xiangjun Li[4,5], Xiao-Hong Yu[6], Kushagra Kapil[2], Jitender S. Deogun[2], John Shanklin[7], Edgar B. Cahoon[4,5], & Etsuko N. Moriyama[1,4]*

*Correspondence: emoriyama2@unl.edu

## Abstract

**Background:** Systems-level analyses, such as differential gene expression analysis, co-expression analysis, and metabolic pathway reconstruction, depend on the accuracy of the transcriptome. Multiple tools exist to perform transcriptome assembly from RNAseq data. However, assembling high quality transcriptomes is still not a trivial problem. This is especially the case for non-model organisms where adequate reference genomes are often not available. Different methods produce different transcriptome models and there is no easy way to determine which are more accurate. Furthermore, having alternative-splicing events exacerbates such difficult assembly problems. While benchmarking transcriptome assemblies is critical, this is also not trivial due to the general lack of true reference transcriptomes.

**Results:** In this study, we first provide a pipeline to generate a set of the benchmark transcriptome and corresponding RNAseq data. Using the simulated benchmarking datasets, we compared the performance of various transcriptome assembly approaches including both *de novo* and genome-guided methods. The results showed that the assembly performance deteriorates significantly when alternative transcripts (isoforms) exist or for genome-guided methods when the reference is not available from the same genome. To improve the transcriptome assembly performance, leveraging the overlapping predictions between different assemblies, we present a new consensus-based ensemble transcriptome assembly approach, ConSemble.

**Conclusions:** Without using a reference genome, ConSemble using four *de novo* assemblers achieved an accuracy up to twice as high as any *de novo* assemblers we compared. When a reference genome is available, ConSemble using four genome-guided assemblies removed

many incorrectly assembled contigs with minimal impact on correctly assembled contigs, achieving higher precision and accuracy than individual genome-guided methods. Furthermore, ConSemble using *de novo* assemblers matched or exceeded the best performing genome-guided assemblers even when the transcriptomes included isoforms. We thus demonstrated that the ConSemble consensus strategy both for *de novo* and genome-guided assemblers can improve transcriptome assembly. The RNAseq simulation pipeline, the benchmark transcriptome datasets, and the script to perform the ConSemble assembly are all freely available from: http://bioinfolab.unl.edu/emlab/consemble/.

**Keywords**: RNAseq, Illumina, transcriptome assembly, *de novo* assembly, genome-guided assembly, ensemble assembly, simulation, benchmarking

## Background

A high quality and comprehensive transcriptome is required in many bioinformatics workflows [1-3]. Many methods have been developed for transcriptome assembly (*e.g.*, reviewed in [4]). However, assembled transcriptomes are often incomplete, especially for non-model organisms or highly divergent strains, and further improvements are needed in assembler performance [5].

Transcriptome assembly methods can be classified into two general categories: *de novo* assemblers that generate the assembly based solely on the RNAseq data (read sets) and genome-guided assemblers that use a reference genome or transcriptome. Most of the current *de novo* assemblers, such as Trinity [6], rely on kmer decomposition of the reads, where kmers are substrings of length *k*, and de Bruijn graph construction [7]. However, with this approach, there are tradeoffs with the choice of the kmer length. For example, while shorter kmers are more likely to fully cover the transcript sequences that need to be assembled, they are also more likely to cause ambiguity in assembly graphs when repetitive sequences exist among transcripts. Due to such tradeoffs, each transcript has an optimal kmer length that facilitates the accurate reconstruction of the full-length sequence. Consequently, using different methods or even using the same method with different kmer lengths will generate different sets of contigs. Isoforms, polyploidy, multigene families, and varying levels of gene expression, all contribute to complexity in transcriptome assembly. Therefore, to obtain the complete transcriptome assembly, multiple assemblers often need to be used with a broad parameter space [8-13].

Genome-guided assemblers, such as Cufflinks [14], avoid the ambiguity in kmer assembly graphs by mapping the RNAseq data to the reference genome and clustering the reads based on genomic location [1]. However reads mapping to multiple locations within a genome can cause ambiguity [15]. When the reference genome used has any sequence divergence from the target

genome, with more divergence, fewer reads can be mapped to the reference without gaps or mismatches. Each combination of read mapper and assembly algorithm handles these issues differently introducing inconsistent performance among assemblers.

While a core set of transcripts is more likely to be assembled correctly by multiple assemblers, many other transcripts may be missed depending on which specific algorithm and kmer length (for a *de novo* method) or read mapper (for a genome-guided method) are used. Through combining the results of multiple assemblers, ensemble assemblers such as EvidentialGene [16] and the method developed by Cerveau and Jackson [17] (we call their method "Concatenation") attempt to address the limitations of individual assemblers, retaining contigs that are more likely to be correctly assembled and discarding the rest. Both of these ensemble methods filter the contigs generated by multiple assemblers (usually *de novo*) by clustering the contigs and determining the representative contig based on both the entire nucleotide and predicted protein sequences. For Concatenation, contigs that contain only portions of longer contigs (without allowing any nucleotide change) are clustered and removed. EvidentialGene clusters contigs by nucleotide similarity (98% identity by default) based on the predicted coding sequences (CDS's) and classifies them into gene loci. From each cluster, it selects contigs including the longest CDS for the main set and those with distinct shorter CDS's for the alternative set. Similar to Concatenation, contigs that are fragments of other contigs are removed (D. Gilbert, personal communication). Both approaches greatly reduce the number of assembled contigs by removing redundant sequences. However, there is no guarantee that the correct sequence is retained as the representative for a given cluster or that each cluster represents a unique gene. In a new genome-guided ensemble assembler, TransBorrow, after extracting reliable subpaths supported by paired-end reads from a splice graph, it merges

103 transcripts assembled by multiple genome-guided methods and extracts reliable assembly
104 subpaths based on the number of assemblers that detect each subpath (transcript) [18]. While the
105 study showed superior performance of TransBorrow compared to individual genome-guided
106 assemblers, how the performance among ensemble assemblers differs has not been examined.
107 To evaluate the quality of transcriptome assemblies, several methods are available, such as
108 DETONATE [19], TransRate [15], and BUSCO [20]. The former two methods evaluate the
109 assembly quality based on how well the assemblies are explained by the RNAseq data used.
110 BUSCO evaluates the completeness of transcriptome assemblies by focusing on searching the
111 universal single-copy gene sets. These methods, however, cannot measure directly the accuracy
112 of each contig sequence. Moreover, without a benchmark transcriptome, transcripts that are not
113 assembled (missed) cannot be quantified and the impact of false positives tends to be
114 underestimated. In a recent study, a "real time" transcriptome was generated using the PacBio
115 long-read sequencing technology and used as a benchmark to evaluate *de novo* assemblies
116 generated from short-read RNAseq data obtained from the same biological samples [21].
117 However, obtaining good quality datasets that have been sequenced by both short- and long-read
118 technologies, especially from many different organisms with various conditions, is not easy nor
119 practical.
120 An alternative approach is to simulate the short-read sequencing based on a reference
121 genome and generate a benchmark dataset including the reference transcriptome and the RNAseq
122 reads. We previously used a simple simulation protocol and generated a benchmark
123 transcriptome based on a human genome [4]. It allowed us to compare transcriptome assembly
124 performance among different approaches. Our preliminary result showed that *de novo*

125 transcriptome assembly can be improved by taking the consensus among multiple *de novo*
126 assemblers.

127 In our present study, we generated several sets of benchmark transcriptomes with different
128 conditions. Using these simulated benchmark datasets as well as the real RNAseq data, and using
129 also various accuracy metrics, we performed a thorough assessment of currently available
130 methods based on various assembly conditions. We showed that the two currently available
131 ensemble methods that focus on thoroughness of assemblies (higher recall) retain significantly
132 high numbers of incorrectly assembled contigs. Instead, the overlaps between different
133 assemblies can be utilized to decrease false positives in transcriptome assembly. To demonstrate
134 the value of a consensus approach in ensemble transcriptome assembly, we present ConSemble,
135 a new ensemble transcriptome assembler. It combines the results from four transcriptome
136 assemblers including both *de novo* and genome-guided methods. For *de novo* assemblers,
137 assemblies generated using multiple kmer lengths are included. ConSemble successfully
138 improved the accuracy of transcriptome assembly over individual tools (both *de novo* and
139 genome-guided). Compared to other ensemble methods, ConSemble also showed consistent
140 improvements with fewer incorrectly assembled transcripts.
141

## 142 Results

### 143 Generation of simulated benchmark RNAseq data

144 To understand the exact performance of different transcriptome assembly methods, we first
145 generated simulated benchmark transcriptomes and corresponding RNAseq read sets as
146 illustrated in the Pipeline 1 in Additional file 1. Three benchmark datasets were generated. "No0-

147 NoAlt" and "Col0-Alt" are based on the *Arabidopsis thaliana* accessions Nossen (No-0) and
148 Columbia (Col-0), respectively. As the simplest benchmark dataset, no alternatively spliced
149 transcripts (isoforms) were included in "No0-NoAlt". Another benchmark dataset was generated
150 from the human reference genome (HG38). Table S1 in Additional file 2 shows the number and
151 distribution of isoforms included in each dataset.

152 The experimental design we used to compare performance of transcriptome assembly is
153 shown in Table S2 in Additional file 2. It allowed us to evaluate whether the transcriptome
154 assembly performance is affected by having isoforms, and for genome-guided methods, how
155 important the choice of the reference genome is. The assembly benchmarking process is
156 summarized in the Pipeline 2 in Additional file 1 (see **Materials and Methods** for further
157 details).

158

## 159 Performance comparison among *de novo* transcriptome assemblers

### 160 *Performance of individual de novo assembly methods*

161 Using the benchmark datasets, we first compared the following four *de novo* transcriptome
162 assemblers: Trinity [6], SOAPdenovo-Trans [22], IDBA-Tran [23], and rnaSPAdes [24, 25].
163 Among the three tests (Tests 1-3 in Table S2 in Additional file 2), Test 1 is the simplest where no
164 isoforms produced by alternative splicing are included in the RNAseq dataset (No0-NoAlt). Both
165 Tests 2 and 3 included isoforms.

166 All four *de novo* assemblers we tested overestimated the number of assembled contigs
167 regardless of whether isoforms are present ranging from 119% by IDBA-Tran for Test 3 to 203%
168 by rnaSPAdes for Test 2 (Table S3 in Additional file 2). The numbers of correctly assembled

169 contigs were low for all four methods. Trinity produced the most accurate assembly for all
170 datasets (*e.g.,* Precision=0.51, Recall=0.64, and *F*=0.57 for Test 1). All four assemblers
171 performed worse when isoforms were included in the dataset (Tests 2 and 3). The isoform
172 assembly performance was further compared in Table S4 in Additional file 2. When more than
173 one isoform existed for a gene (Categories 3-7), often none (Category 3) or only one isoform
174 (Category 4) could be correctly assembled by the *de novo* assemblers. Trinity was most
175 successful among the four *de novo* assemblers in identifying multiple isoforms (higher numbers
176 in Categories 5-7). It was the only *de novo* assembler that identified four or more isoforms (Fig.
177 S1 right panel in Additional file 3).
178 Contigs assembled by these four individual assemblers are compared in Fig. 1 (left column).
179 The 4-way intersection set included a large portion of correct contigs (40 ~ 45%, shown in black
180 letters) for all three datasets. It is also noteworthy that some contigs were correctly assembled
181 only by one assembler. In contrast, while the large portion of the contigs that were assembled
182 only by a single method were incorrect, the 4-way intersection set included only a small number
183 of incorrect contigs (0.6~2%, shown in red letters). These results illustrate a risk of choosing a
184 single *de novo* assembler.
185

186 ***Development of a consensus-based* de novo *ensemble assembler, ConSemble***

187 As described before, different sets of transcripts can be assembled optimally with different kmer
188 lengths. To explore if we can increase the number of correctly assembled contigs (*TP*) with using
189 multiple *de novo* assemblers, for each *de novo* assembler we pooled the assemblies by combining
190 the contigs produced using multiple kmer lengths. The range and number of kmer lengths used

191 for each method is as follows: IDBA-Tran with five kmers (20-60), SOAPdenovo-Trans with 16
192 kmers (15-75), rnaSPAdes with 14 kmers (19-71), and Trinity with four kmers (19-31).
193 As expected, this approach increased *TP* for all assemblers (Table S5 in Additional file 2;
194 Recall>0.67 for Test 1, >0.60 for Test 2, and >0.52 for Test 3). However, it also accumulated a
195 disproportionately large number of incorrectly assembled contigs (*FP*), greatly reducing the
196 accuracy of the assemblies (*e.g.*, Precision=0.05 for rnaSPAdes in Test 1). When we merged all
197 these assemblies (taking the union set of assemblies generated by all methods with multiple kmer
198 lengths; shown as "Merged" in Table S5 in Additional file 2), the number of correctly assembled
199 contigs increased further, although again at the cost of accumulating disproportionately more
200 incorrectly assembled contigs. For example, for the No0-NoAlt dataset (Test 1), merging all the
201 *de novo* assemblies produced 14,770 correctly assembled contigs (*TP*) from the 18,875
202 benchmark transcripts (Recall=0.78), but also had 549,859 incorrectly assembled contigs (*FP*)
203 decreasing the Precision significantly (0.03).
204 When we compared these pooled contig sets assembled by the four assemblers using multiple
205 kmers, we noted that the vast majority of the contigs that were assembled only by one or two
206 methods were incorrect (Fig. 1, right column). However, as the number of *de novo* assemblers
207 sharing a unique contig sequence increased, the likelihood that the contig was correctly
208 assembled also increased. A large majority (64 ~ 89%) of contigs produced by all four of the *de*
209 *novo* assemblers were correctly assembled regardless of the test dataset, which is 70~83% of all
210 correctly assembled contigs. Furthermore, for all test datasets, the number of contigs that were
211 incorrectly assembled by all four assemblers was consistently small (0.2~1.1% of all incorrectly
212 assembled contigs).

213 These observations indicate that by utilizing such consensus information, it is possible to
214 increase the number of correctly assembled contigs and at the same time reduce the number of
215 incorrectly assembled contigs, improving the overall assembly performance. To demonstrate this
216 approach, we present a new consensus-based ensemble transcriptome assembly method,
217 ConSemble, in which contigs shared by at least three of the four assemblers are retained for the
218 final assembly (see Pipeline 3 in Additional file 1 and **Materials and Methods** for further
219 details).
220

221 ***Performance of ConSemble compared against other* de novo *assembly methods***

222 The transcriptome assembly performance of ConSemble using the four *de novo* assemblers was
223 compared against the four individual *de novo* assemblers as well as two other ensemble methods
224 (EvidentialGene and Concatenation) that are also based on *de novo* assemblers. Specific details
225 for how each of these methods was run is described in **Materials and Methods**. The
226 performance statistics and comparisons are summarized in Fig. 2 and Table S6 in Additional file
227 2.

228 ConSemble3+d, the default assembly output of ConSemble based on four *de novo*
229 assemblers, effectively filtered out the incorrectly assembled contigs. When isoforms were not
230 included (the No0-NoAlt dataset), of 20,263 contigs produced by ConSemble3+d (107% of the
231 reference), 13,352 were correctly assembled (Precision=0.66 and Recall=0.71), achieving the
232 highest *F* score (0.68) among any *de novo* approach including both individual and ensemble
233 methods. EvidentialGene overestimated the number of transcripts significantly (417%). While
234 EvidentialGene recovered more benchmark transcripts (Recall=0.66) than Trinity (Recall=0.64;
235 Table S3 in Additional file 2), the best of the individual *de novo* assemblers, many contigs were

236 incorrectly assembled (Precision=0.16 compared to 0.51 for Trinity) leading to very low overall
237 performance (*F*=0.26). While Concatenation produced a lower number of incorrectly assembled
238 contigs (Precision=0.47) compared to Trinity, it did not achieve the accuracy level shown by
239 Trinity (*F*=0.52 compared to 0.57 by Trinity).

240 When isoforms were included in the dataset (the Col0-Alt and HG38 datasets, Tests 2 and 3),
241 the poor performance of the four individual *de novo* assemblers (shown in Table S3 in
242 Additional file 2) limited the potential for these *de novo* ensemble assembly methods.
243 ConSemble3+d slightly overestimated the number of transcripts present in the benchmark
244 datasets (109~112%). However, it again successfully reduced the numbers of incorrectly
245 assembled contigs outperforming all other *de novo* methods (Precision=0.53 and 0.47,
246 Recall=0.59 and 0.52, and *F*=0.56 and 0.49, respectively). EvidentialGene again had far more
247 contigs than the benchmark transcriptome (283% and 335% for the Col0-Alt and HG38 datasets,
248 respectively) resulting in very low overall performance (*F*=0.24 and 0.16, respectively). While
249 Concatenation showed much higher Precision with lower numbers of incorrectly assembled
250 contigs than EvidentialGene, the overall performance (*F*=0.51 and 0.38, respectively) did not
251 exceed those obtained by using only Trinity (*F*=0.53 and 0.45, respectively). The limited ability
252 of the most *de novo* assemblers to reconstruct multiple isoforms further limited ConSemble's
253 ability to accurately recover all the isoforms present in the merged assembly (Fig. S2 in
254 Additional file 3). However, although ConSemble3+d recovered fewer isoforms especially for
255 genes with five or more isoforms than Trinity, ConSemble3+d performed significantly better
256 than other individual *de novo* assemblers in terms of isoform identification (Fig. S1 right panel in
257 Additional file 3).

258 Among the three *de novo* ensemble assemblers, ConSemble3+d and Concatenation performs
259 virtually the same in terms of identifying isoforms up to five (Fig. S2 in Additional file 3). For
260 genes containing six or more isoforms, which correspond to only less than 2% of the benchmark
261 genes, Concatenation appears to be able to identify slightly more isoforms. However, with lower
262 Precision, Recall, and F scores with this method compared to ConSemble3+d (Table S6 in
263 Additional file 2), it sacrifices the recovery of transcripts from genes with five or fewer isoforms.
264 As described before, EvidentialGene and Concatenation cluster the contigs and choose the
265 representatives for the final assemblies. These approaches did not seem to significantly improve
266 the overall transcriptome assembly performance compared to the best individual *de novo*
267 assembler. However, as we observed in Table S5 in Additional file 2, the "Merged" *de novo*
268 assembly correctly assembled up to 78% of the transcripts in the benchmark. By utilizing
269 consensus information, the ConSemble approach successfully recovered many of these
270 transcripts without increasing the number of incorrectly assembled contigs (*FP*), improving the
271 overall assembly performance.
272

273 **Performance comparison among genome-guided assemblers**

274 The ConSemble approach can also apply to genome-guided methods (Pipeline 4 in Additional
275 file 1). Therefore, we compared the performance of genome-guided transcriptome assemblers
276 including both individual as well as ensemble methods.
277

278 ***Performance of individual genome-guided methods***

279 Using the benchmark datasets, we compared the following four genome-guided transcriptome
280 assemblers: Cufflinks [14], Bayesembler [26], Scallop [27], and StringTie2 [28, 29]. The

281 simplest experiment was to assemble the transcriptome where no alternative splicing happened
282 (the No0-NoAlt dataset) using the same No-0 genome as a reference (Test 4 in Table S2 in
283 Additional file 2). The genome-guided assemblers performed surprisingly poorly for this dataset
284 (Table S7 in Additional file 2). For all assemblies, only 71~75% of the assembled contigs were
285 correct (shown as Precision). StringTie2 recovered the most benchmark transcripts (Recall=0.80)
286 and Bayesembler the least (Recall=0.59).

287 The complexity of dealing with alternative splice forms is evident from the decreased
288 performance of all four assemblers even when using the same genome as the reference (Tests 6
289 and 8 in Table S7 in Additional file 2). While Bayesembler consistently produced the fewest
290 unique contigs, those assembled contigs were most accurate as shown by the consistently highest
291 values of Precision (>0.54). Scallop and StringTie2, on the other hand, produced larger numbers
292 of unique contigs recovering more benchmark transcripts as shown by the larger Recall values
293 (>0.48).

294 The results were similar when STAR [30] was used as the aligner (Table S8 in Additional
295 file 2). STAR with default parameters was shown to more accurately assign reads to the proper
296 genomic location than Tophat2 [31]. However, we found that the assembly results based on
297 STAR mapping were comparable to those based on Tophat2 showing no clear advantage in using
298 one aligner over the other. Since Tophat2 remains the most widely used read mapper for
299 genome-guided assemblies and is required for Bayesembler, only the results based on Tophat2
300 alignments are presented for further analyses.

301 The isoform assembly performance is compared further in Table S4 in Additional file 2.
302 When more than one isoform existed for a gene (Categories 3-7), Bayesembler and Scallop
303 recovered more isoforms (higher numbers in Categories 5-7). These two assemblers could

304 identify all isoforms for 50% or more of the genes that have isoforms (Category 6) or for about
305 one third of the genes that have three or more isoforms (Category 7). Among the four genome-
306 guided assemblers, only Bayesembler identified up to seven isoforms correctly, although the
307 success rates were not very high (Fig. S1 left panel in Additional file 3). None of the assemblers
308 could identify all isoforms when there were more than six isoforms.

309 The performance of all these assemblers was considerably worse when a non-identical
310 reference genome was used as the reference (Tests 5, 7, and 9 in Table S7 in Additional file 2).
311 While the total numbers of contigs produced were similar to the corresponding test results with
312 the same reference genomes (Tests 4, 6, and 8), the correctness of the contigs was greatly
313 diminished (*e.g.*, $F<0.34$) where, on average, only 32% of the assembled contigs were correct
314 (Precision) and only 33% of the benchmark transcripts were recovered (Recall). Of note is that
315 all the *de novo* assemblies had higher accuracy compared to the genome-guided assemblies
316 generated with the reference differing from the strain sequenced. Trinity, for example, assembled
317 41~51% of the contigs correctly (Precision) and recovered 50~64% of benchmark transcripts
318 (Recall) (Table S3 in Additional file 2), while all the genome-guided assemblers showed lower
319 than 38% for these statistics. In fact, the *de novo* assemblers performed at the level similar to
320 those with genome-guided assemblers with the same references (Precision>0.54 and
321 Recall>0.51; Tests 4, 6, and 8 in Table S7 in Additional file 2).

322 Contigs assembled by the four genome-guided assemblers are compared in Fig. 3. Similar to
323 what we observed with the *de novo* methods, while the 4-way intersection set always included
324 the largest numbers of correctly assembled contigs (shown in black letters), which combination
325 of assemblers shared more correctly assembled contigs depended on the dataset and the reference
326 genome used. Some contigs were correctly assembled only by one of the assemblers, while the

327 majority of such contigs were false positives (incorrectly assembled, shown in red letters).
328 Similar to what we observed with *de novo* assemblers, it illustrates the difficulty in choosing a
329 single genome-guided assembler for any transcriptome assembly problem.
330 We further compared the contigs generated by the four genome-guided assemblers with those
331 generated by the four *de novo* assemblers. The results showed that some contigs were assembled
332 correctly only by either genome-guided or *de novo* assemblers (Fig. 4). For example, for the
333 No0-NoAlt dataset, 396 contigs were produced only by the *de novo* assemblers. More contigs
334 were correctly identified uniquely by the *de novo* methods for other datasets (554 for Col0-Alt
335 and 1,836 for Human) despite the reduced performance of the *de novo* assemblers on these
336 datasets. It shows the advantage of using *de novo* assemblers especially when a good reference is
337 not available.
338
339 ***Performance of ConSemble compared against other genome-guided methods***
340 Since the individual genome-guided assemblers showed a similar trend in the overlap of
341 correctly assembled contigs as the *de novo* methods, the consensus-based ConSemble approach
342 with genome-guided methods is expected to improve the accuracy of the assembly. The
343 performance of ConSemble applied to the genome-guided assemblies (ConSemble3+g) is
344 compared with another genome-guided ensemble assembler TransBorrow [18] as well as
345 individual genome-guided assemblers in Fig. 5 and Table S9 in Additional file 2.
346 Compared to individual genome-guided methods, TransBorrow did not show significantly
347 different assembly performance. Moreover, TransBorrow could assemble a surprisingly small
348 number of contigs (<12%) from the Col0-Alt dataset (Test 6). The authors mentioned about the
349 possible issues associated to the complexity in building colored graphs depending on the

350 assembly included [18]. The Col0-Alt dataset may have presented one such challenge. In
351 contrast, ConSemble3+g showed significant reduction in miss-assembly (*FP*) without sacrificing
352 correct assembly (*TP*). For the No0-NoAlt dataset, it achieved a very high Precision (0.90). For
353 all benchmark datasets, ConSemble3+g assembly showed balanced and high accuracy ($F$=0.82,
354 0.67, and 0.52 for the No0-NoAlt, Col0-Alt, and HG38 datasets, respectively), which was
355 consistently better than TransBorrow as well as individual genome-guided methods. It should be
356 also noted that although TransBorrow recovered more correctly assembled contigs (*TP*) than
357 ConSemble, showing higher Recall values (except for Col0-Alt), due to a larger number of
358 incorrectly assembled contigs (*FP*), it showed lower Precision values, and hence lower overall
359 performance (F<0.76).
360 For genes with large numbers of isoforms, ConSemble3+g identified more isoforms than
361 other ensemble methods (Fig. S2 in Additional file 3). However, because many isoforms were
362 identified in only a limited number of genome-guided assemblies (Fig. S1 left panel in
363 Additional file 3), it also limits the ability of ConSemble3+g to identify multiple isoforms
364 especially for genes with five or more isoforms.
365

366 **Comparison of nucleotide- and protein-level performance**

367 In addition to the protein-level performance described above, the quality of assembled contigs at
368 the nucleotide sequence level was examined using DETONATE [19]. DETONATE provides a
369 reference-free model-based RSEM-EVAL score as well as reference-based scores ($F_1$ and KC).
370 The RSEM-EVAL metric given by DETONATE evaluates the transcriptome assembly
371 performance by comparing the assembled nucleotide sequences to the original reads [19].
372 Because genome-guided assembly returns the sequence of the reference genome rather than the

373 RNAseq data, using RSEM-EVAL on assemblies where the RNAseq data did not originate from
374 the reference genome used can lead to poor results. Therefore, for genome-guided assemblies,
375 only those produced using the same reference genome as the simulated RNAseq library (Tests 4,
376 6, and 8) were examined.

377 Many of the trends seen in the protein-level performance were also reflected in the nucleotide
378 assembly quality (Table S10 in Additional file 2). In general, the assemblies with higher *F* at the
379 protein level (Tables S3, S6, S7, and S9 in Additional file 2) also had relatively high
380 DETONATE scores (RSEM-EVAL and KC). Among the ensemble assemblers, most often
381 TransBorrow (except for Col0-Alt) and ConSemble3+g had the best scores in *F* and/or
382 DETONATE scores. While EvidentialGene consistently showed the lowest *F* at the protein level
383 as well as $F_1$ both at the nucleotide and contig levels, it was found to perform somewhat better
384 when the RSEM-EVAL scores were considered. As even a single nucleotide indel can have
385 major impacts on the predicted protein sequences, it is important to evaluate performance
386 statistics at both nucleotide and protein levels [32].

387 RSEM-EVAL and KC tend to provide higher scores for assemblies with longer sequences
388 that account for, *e.g.*, more of the kmers in the RNAseq data or the reference over the precision
389 of the contig sequence. To illustrate this issue, using the same ConSemble *de novo* assembly, we
390 generated two other collections of contig sets. In the ConSemble3+dLong set, instead of the
391 shortest nucleotide sequences for the default ConSemble3+d, the longest sequences were chosen
392 as the representative contigs from those coding identical protein sequences. In the
393 ConSemble3+dHigh set, contig sequences with the highest RSEM-EVAL scores were chosen
394 instead. Although these three ConSemble3+d assemblies have the same protein-level *F* scores
395 (for the same set of protein sequences), DETONATE scores were significantly different (Table

396 S10 in Additional file 2). By choosing the shortest nucleotide sequences, ConSemble assemblies
397 tend to truncate the untranslated regions (UTRs) of the transcripts. ConSemble3+dLong and
398 ConSemble3+dHigh showed similar and much higher DETONATE scores indicating longer
399 contigs are scored higher.
400

401 **Impact of using lower percent identity thresholds**

402 The results presented so far are based on the numbers of correctly assembled contigs where the
403 assembled sequences need to be fully 100% identical to the reference protein sequences. In
404 practice, this is more strict than necessary for many downstream analyses. We therefore
405 examined the impact of using lower identity thresholds on the assembly metrics.
406 The number of correctly assembled contigs identified at varied thresholds are shown in Fig.
407 S3 in Additional file 3. The increase in the performance metrics was most significant when the
408 threshold was reduced from 100% to 98%. The increase was generally proportional across all
409 methods with the exceptions of the genome-guided assemblies using different reference genomes
410 and EvidentialGene. With the 100% threshold, the performance of the genome-guided methods
411 was significantly lower than *de novo* methods when different reference genomes were used.
412 However, with less strict thresholds (<100%), their performance quickly recovered and became
413 better than that of some of the *de novo* assemblers, although it remained lower than genome-
414 guided assemblers using the same reference genome down to the lowest threshold (90%) used.
415 For the No0-NoAlt dataset, EvidentialGene recovered more correctly assembled contigs with
416 relaxed thresholds and resulted in Recall scores higher than other *de novo* methods (including
417 both ensemble and individual methods) (Table S11 in Additional file 2). However, since
418 EvidentialGene retained a large number of incorrectly assembled contigs (Precision<0.19

419 indicating only one in five contigs were correctly assembled), the accuracy level was kept very
420 low (F<0.31). ConSemble3+d remained the best performing non-genome guided method for all
421 three datasets. ConSemble3+g maintained the highest accuracy among all the methods we tested
422 including individual genome-guided and *de novo* methods. These trends were consistent in the
423 Col0-Alt and HG38 benchmark datasets as well (Tables S12 and S13 in Additional file 2).
424

425 **Assembling the real plant transcriptomes**

426 Assembly performance was further tested using the real RNAseq data from three plant species.
427 We included a strain of a model dicot *A. thaliana* where a high-quality genome and
428 transcriptome information is available, another dicot *Momordica charantia* (bitter melon, a
429 diploid), as well as the hexaploidal cotton (*Gossypium hirsutum*). This selection represents a
430 range of ploidies, which challenges transcriptome assembly methods. In contrast to testing using
431 simulated benchmark data, when the real RNAseq data are used, we do not know which contigs
432 are assembled correctly or incorrectly and hence, we cannot evaluate the assembly performance
433 using the same set of statistics. We instead used the number of genes identified from the
434 "Eudicotyledons" dataset of Benchmarking Universal Single-Copy Orthologs (BUSCO) [20] to
435 evaluate the thoroughness of each assembly. For *A. thaliana* Col-0, the number of contigs
436 matching with those in the Araport11 reference transcriptome at the protein level was also
437 counted. For the nucleotide level, the reference-free model-based RSEM-EVAL scores were
438 included. The results are summarized in Table 1.
439 For *A. thaliana* Col-0, all ensemble assemblers except TransBorrow produced more contigs
440 that matched Araport sequences and BUSCOs and had slightly better RSEM-EVAL scores than
441 the *de novo* assemblers. The very low performance by TransBorrow was consistent to what we

442 observed with the Col0-Alt benchmark dataset. All ensemble assemblers had better or
443 comparable performance than the genome-guided assemblers. While ConSemble3+g found
444 slightly fewer BUSCOs compared to SOAPdenovo-Trans, it had the second best RSEM-EVAL
445 following StringTie2. Among the *de novo* assembly-based ensemble methods, EvidentialGene
446 assembled the most BUSCOs but the fewest Araport-matching contigs and had the lowest
447 RSEM-EVAL score.

448 For *M. charantia*, all ensemble methods including TransBorrow recovered more BUSCOs
449 than *de novo* assemblers. EvidentialGene, Concatenation, and TransBorrow also recovered more
450 BUSCOs than the highest performing genome-guided method, StringTie2. Although
451 EvidentialGene recovered the most BUSCOs, it produced a significantly more contigs (324% of
452 the reference) indicating a high FP rate. While ConSemble3+d did not find as many BUSCOs as
453 StringTie2 did, it had the best RSEM-EVAL score among all assemblers. Concatenation
454 produced the fewest contigs, which was even fewer than those produced by Scallop, the best
455 genome-guided assembler. Among the genome-guided assemblers, Scallop and StringTie2 had
456 similar numbers of BUSCOs as well as RSEM-EVAL scores, while the number of contigs
457 produced by StringTie2 was close to the number reported for the reference transcriptome.

458 For the hexaploidal cotton (*G. hirsutum*), all methods produced significantly more contigs
459 than the 70,478 transcripts reported (except TransBorrow where the assembly largely failed). All
460 genome-guided methods showed high numbers of BUSCOs with StringTie2 producing the most.
461 Among the *de novo* assemblers, only Trinity produced a comparable number of BUSCOs. The
462 remaining *de novo* assemblers produced far fewer BUSCOs despite the high numbers of contigs
463 assembled, suggesting high false positive rates. Both EvidentialGene and Concatenation
464 produced large numbers of BUSCOs. However, EvidentialGene produced more than ten times

465 the number of sequences in the reference transcriptome and had a lower RSEM-EVAL score.
466 ConSemble3+d showed a significantly smaller number of BUSCOs and one of the worst RSEM-
467 EVAL scores. This may be due to the impact of a high ploidy [33]. It is discussed further below.
468

## 469 Discussion

470 Despite the number of transcriptome assemblers available, there is still no single assembler or
471 assembly strategy that performs best in all situations. Our simulation study clearly showed that
472 every assembler correctly assembled a set of contigs while missing transcripts correctly
473 assembled by other methods. When the reference genome was available from the same strain, the
474 genome-guided assemblers outperformed *de novo* and most ensemble assemblers. However, if
475 such reference sequences were not available, the performance of genome-guided assemblers
476 dropped below that of *de novo* assemblers. Ensemble assembly approaches can overcome some
477 of the limitations of individual assemblers. Our analysis showed that without requiring a
478 reference genome, ensemble *de novo* methods achieved the assembly performance comparable to
479 or higher than that of individual genome-guided methods.
480 Both EvidentialGene and Concatenation generally kept more of the correctly assembled
481 contigs produced by the individual *de novo* assemblers indicated by high Recall and BUSCO
482 scores. However, our simulation study showed that they also had many incorrectly assembled
483 contigs that decreased the overall assembly accuracy. Instead of simply choosing representative
484 contigs, ConSemble used consensus-based information to filter more reliable contigs. By pooling
485 multiple assemblies, especially also pooling those generated using multiple kmer lengths for *de*
486 *novo* methods, ConSemble increases the completeness of the assembled transcriptome. By taking
487 the consensus contig set from four contig libraries, both ConSemble3+d and ConSemble3+g

488 reduce the number of incorrectly assembled contigs significantly. This was observed in much
489 lower numbers of incorrectly assembled contigs than those obtained by the other ensemble
490 methods as well as all individual methods. For the three benchmark datasets, ConSemble3+d
491 showed comparable accuracies to the genome-guided assemblers with ideal reference genomes
492 and outperformed all individual *de novo* assemblers and the individual genome-guided
493 assemblers without good reference genomes. Furthermore, the consensus strategy used with
494 ConSemble allows the users to extract a set of contigs that are more likely to be correctly
495 assembled (ConSemble4 assembly) from the rest of the contigs (ConSemble3 assembly).
496 Any ensemble method will necessarily be limited by the performance of the individual
497 assemblies it is based on. If no individual method can correctly assemble a contig, no ensemble
498 assembly can recover it. ConSemble uses the overlap of any three of the four individual
499 assemblies (ConSemble3+). Therefore, any contigs that are correctly assembled by only one or
500 two assemblers are omitted from the final contig set. For the three benchmark transcriptome
501 assemblies, 10~20% of correctly assembled contigs (1,418 of 14,770, 1,988 of 11,179, and 2,139
502 of 11,267 in the No0-NoAlt, Col0-Alt, and HG38 datasets, respectively) were discarded in this
503 manner in ConSemble3+d. In Fig. S4 in Additional file 3, we examined how the assembly
504 performance is affected by expression levels of transcripts. It shows that assembly performance
505 is affected by the expression levels regardless of the datasets, and the *de novo* methods are
506 affected much more significantly than the genome-guided methods. Furthermore, the ConSemble
507 assembly (both ConSemble3+d and ConSemble3+g) performance was affected more for
508 transcripts whose expression levels are low (*e.g.,* bin # 1-3 in Fig. S4 in Additional file 3). As we
509 described before, especially when genes have many isoforms (*e.g.*, five or more) both
510 ConSemble3+d and ConSemble3+g missed many of such isoforms (Fig. S2 in Additional file 3).

511 However, as shown in "All Assemblies" in Fig. S1 in Additional file 3, many of these isoforms
512 in fact existed in at least one of the assemblies that were used as the contig libraries for
513 ConSemble assembly. These isoforms must have been unique to a single contig library or shared
514 only by two contig libraries, and subsequently filtered out during the process of consensus
515 assembly. Since the expression levels of isoforms from a gene are known to vary significantly, it
516 is possible that the isoforms that were missed by ConSemble3+ assembly were those with low
517 expression. This may also explain lower than expected levels of performance observed with all
518 assemblers especially when isoforms were included in the test datasets. To recover such low
519 expression transcripts, more refined assembly strategies beyond the current simple consensus
520 approach need to be considered.

521 A similar effect is found when performance among the assemblers included in ConSemble
522 varies widely. In the hexaploid cotton assembly, the quality of the ConSemble assembly
523 deteriorated noticeably. This result suggests that the increased ploidy may decrease the
524 likelihood that the true sequences are reconstructed by multiple *de novo* methods, consistent with
525 the reduced overlap between *de novo* assemblers in simulated polyploidy benchmarks [33]. Since
526 other ensemble approaches showed much larger BUSCOs for the cotton, it would be interesting
527 to consider an integrated approach. Further benchmarking studies incorporating varied ploidies
528 are necessary to determine the extent and impact of this issue.

529 ConSemble for *de novo* methods requires multiple different assemblers to run multiple times
530 over a range of kmer lengths. Therefore, the availability of computational resources could
531 become a limiting factor for deciding the number of assemblies ConSemble can utilize.
532 However, as each assembly process is independent, these jobs can be run in parallel depending
533 on the availability of the resources. This makes the overall assembly time roughly equivalent to

534 the time needed for the longest individual assembly reducing the necessary computational time
535 significantly.
536

## 537 Conclusions

538 We presented ConSemble, a new consensus-based ensemble transcriptome assembly approach. It
539 can be used with either *de novo* or genome-guided methods. Using ConSemble with *de novo*
540 methods, a transcriptome assembly produced showed the quality rivals that of genome-guided
541 assembly without requiring a reference genome and outperforms *de novo* and other ensemble
542 strategies. ConSemble with genome-guided methods further increases the accuracy of the
543 assembly over individual genome-guided methods. Both of *de novo* and genome-guided
544 ConSemble assembly achieve high performance by retaining the majority of the contigs correctly
545 assembled by individual assemblers and removing a significant number of misassembled contigs.
546 Further work is needed for identifying more correctly assembled contigs especially when
547 multiple lowly expressed isoforms exist or in the case of polyploidies. Improvements are needed
548 to retain more of the contigs that are correctly assembled only by a small number of assemblers
549 without accumulating incorrectly assembled contigs. We also showed the importance of
550 developing realistic simulated RNAseq benchmark datasets that allow evaluating the
551 performance of transcriptome assemblers under various conditions. It provides the only means of
552 assessing the accuracy of the transcripts assembled directly and quantitatively.
553

## 554 Materials and Methods

555 **Genomes and transcriptomes used**

556 RNAseq simulation and genome-guided assemblies of the *A. thaliana* accession Nossen (No-0)
557 were based on the assembly by Gan *et al.* [34] downloaded from the 1001 genomes project [35;
558 accession CS6805]. RNAseq simulations and genome-guided assemblies of the *A. thaliana*
559 accession Columbia (Col-0) were based on the TAIR reference genome (version 9) [36] and the
560 atRTD transcriptome dataset (version 3) [37]. The atRTD dataset was chosen over the TAIR
561 transcriptome (version 10) due to the higher prevalence of isoforms in the atRTD dataset,
562 allowing better testing of each assembler's ability to reconstruct alternative transcripts. The
563 Human RNAseq simulations were based on the HG38 reference genome and transcriptome [38;
564 GCF_000001405.39]. To evaluate the impact of using a different reference for this RNAseq
565 library, the reads were also mapped to the HuaXia1 (HX1) reference genome (available from
566 http://hx1.wglab.org).

567 Three datasets were used for the assembly of real RNAseq data. The *A. thaliana* Col-0
568 dataset consisting of 116M 76bp read pairs sequenced using Illumina Genome Analyzer II is
569 from Marquez *et al.* [39], which is available at National Center for Biotechnology Information
570 (NCBI) (SRA047499). For *M. charantia* (bitter melon), a dataset consisting of 228M 100bp read
571 pairs was produced by sequencing of RNA samples from leaves, roots, flowers, and seeds using
572 Illumina HiSeq 2500 (NCBI: SRR3535137, SRR3535138, SRR3535144, and SRR3535149).
573 The reference genome was from Urasaki *et al.* [40] (NCBI: BDCS01000001–BDCS01001052).
574 For *G. hirsutum* (upland cotton), an RNAseq dataset consisting of 117M 200bp read pairs was
575 produced from RNA samples from leaves, roots, flowers, and seeds using Illumina HiSeq 4000
576 (NCBI: SRR7689126- SRR7689129). The genome assembly used as the reference is the
577 allotetraploid L. acc. TM-1 [41; http://www.cottongen.org].

578

579 **Production of simulated benchmark transcriptomes and read sets**

580 Simulated benchmark transcriptomes and corresponding RNAseq read sets were produced from
581 *A. thaliana* and humans as follows. A modified Flux Simulator v1.2.1 pipeline [42] illustrated in
582 the Pipeline 1 in Additional file 1 was used to produce ~250M 76bp read pairs for each dataset.
583 The No0-NoAlt transcriptome was generated based on the *A. thaliana* No-0 genomic sequence
584 and no alternative splice events were included. The Col0-Alt transcriptome was generated based
585 on the *A. thaliana* Col-0 genomic sequence and the version 3 atRTD transcriptome model. The
586 expressed transcriptome in this dataset contains zero to nine alternative splice events (one to ten
587 isoforms) per gene where only the isoforms unique at the protein level were included. The
588 transcriptome for the Human dataset was based on the HG38 reference genome and
589 transcriptome. It contains zero to 14 alternative splice events (1 to 15 isoforms). Only reference
590 transcripts with full coverage of RNAseq data (all positions are required to be covered by at least
591 one read) were included in the benchmark datasets, as transcripts without full coverage cannot be
592 correctly assembled as a single contig. Table S1 in Additional file 2 shows the number and
593 distribution of isoforms per transcript for each dataset. All benchmark datasets used in this study
594 are available from: http://bioinfolab.unl.edu/emlab/consemble/

595

596 **Read processing**

597 The read pairs generated by Flux Simulator were quality filtered using Erne-filter 2.0 [43] using
598 the 'ultra-sensitive' flag with a minimum average quality of q20 and in paired-end mode. The
599 remaining reads were normalized using Khmer [44] with a kmer length of 32, an expected

600 coverage of 50x, and in paired-end mode. As normalization works at the read level, kmer-length
601 selection for this step has minimal impact on the unique kmers kept, and hence on the
602 performance of the assembly.
603

604 ## *De novo* assembly

605 *De novo* assembly was performed using Trinity 2.4.0 [6], SOAPdenovo-Trans 1.0.3 [22], IDBA-
606 Tran 1.1.1 [23], and rnaSPAdes 3.10.0 (using the rnaspades.py script) [25]. Each of the
607 assemblers was run using default settings as well as using multiple kmer lengths. With the
608 default setting, IDBA-Tran assembles across multiple kmer lengths and merges the results for the
609 final assembly. Trinity and SOAPdenovo-Trans use a single default value (25 and 23,
610 respectively). rnaSPAdes chooses the kmer length depending on the dataset. For this study, a
611 kmer length of 55 was chosen for both No0-NoAlt and Col0-Alt read sets and a kmer length of
612 37 was chosen for the HG38 set. Each method was also run with a range of kmer lengths ($k$) with
613 increments ($i$) as follows: IDBA-Tran with $k$=20-60 and $i$=10, SOAPdenovo-Trans with $k$=15-75
614 and $i$=4, rnaSPAdes with $k$=19-71 and $i$=4, and Trinity with $k$=19-31 and $i$=4.
615 The assemblers Velvet 1.2.10 [45] with Oases 0.2.08 [46] and Mira 4.0.2 [47] were excluded
616 from this study after the preliminary analysis due to their high computational requirements (both
617 resources and time), issues with program stability, or both.
618

619 ## Genome-guided assembly

620 The normalized reads were first mapped to the reference genomes using Tophat2 2.0.14 with
621 default settings [48]. Assemblies were performed using Cufflinks 2.2.1 [14], Bayesembler 1.2.0

622 [26], Scallop 0.10.2 [27], and StringTie2 2.0 [29]. Each assembler was executed on the default
623 settings. Assembled contig sequences were extracted using the gtf_to_fasta module from
624 Tophat2. The two *Arabidopsis* RNAseq datasets (No0-NoAlt and Col0-Alt) and the two human
625 RNAseq datasets (HG38 and HX1) were assembled using each of the two reference genomes. To
626 compare the impact of using different read mappers on assembly quality, additional assemblies
627 were also performed using STAR 2.4.2a [30] for read mapping.
628

629 **Ensemble methods**

630 For EvidentialGene (version 2017.03.09) [9], we chose the "okay" nucleotide contig set (okay.fa
631 and okalt.fa) produced by the tr2aacds.pl pipeline as the final output for our comparative
632 analysis. The assembly was performed using the same sets of four *de novo* assemblers and kmer
633 lengths as described above resulting in 39 assemblies in total. Concatenation originally used
634 three assemblers, Trinity, IDBA-Tran, and CLC (https://www.qiagenbioinformatics.com/), with
635 only one kmer length each [17]. For this study, we tested all combinations of three from the four
636 aforementioned *de novo* assemblers and kmer lengths. CLC was excluded from this study
637 because it is only available under a commercial license. Based on the numbers of correctly and
638 incorrectly assembled contigs, we decided to use the following three assemblies as the best
639 combination for Concatenation: IDBA-Tran with $k$=50, Trinity with $k$=31, and rnaSPAdes with
640 $k$=55. For TransBorrow (version 1.3) [18], the GTF files for the assemblies with the same
641 reference genome for each of the four genome-guided assemblers (Cufflinks, Bayesembler,
642 Scallop, and StringTie2) were manually merged into a single file. The TransBorrow assembly
643 was performed using default parameters.
644

645 **ConSemble, a new consensus-based ensemble assembly**

646 The Pipelines 3 and 4 in Additional file 1 summarize the process of the ConSemble
647 transcriptome assembly approach. Four transcriptome assembly methods, either *de novo* or
648 genome-guided, are used to generate four "contig libraries", each containing the unique protein
649 sequences produced by each method. For the *de novo* based assembly (Pipeline 3), each *de novo*
650 assembler is used with a given set of multiple kmer lengths (described above), producing a total
651 of 39 assemblies across the four methods. The assembly sets from each method are pooled to
652 produce four contig sets. In these pooled contig sets, unique contigs are identified based on the
653 coded protein sequences, producing four "pooled unique" contig sets. Each of the four pooled
654 unique contig sets is used as the "contig library" from each method. For the genome-guided
655 based assembly (Pipeline 4), the unique contig set (at the protein level) extracted from the
656 assembled contigs using each method is used as the "contig library". Based on the tradeoff
657 among the performance statistics (described below), we decided to take the set of contigs that are
658 found in the intersections of at least three contig libraries as the default output assembly. This
659 assembly is called ConSemble3+.

660 The 5' and 3' boundaries of the transcripts are handled differently depending on the
661 assembler. Although this does not affect the predicted open reading frame (ORF) and the coded
662 protein sequence, it affects the overall contig length and potentially the sequence of the
663 untranslated regions. Therefore, to minimize the impact of such differing behaviors between
664 assemblers, we concentrated only on the longest ORFs produced by ORFfinder [49] to compare
665 assembled contigs. When contigs with differing nucleotide sequences code entirely identical
666 protein sequences, the one with the shortest nucleotide sequence is chosen and included in the
667 final assembly. This is done to minimize the risk of including any chimera or over-assembled

668 sequences. Although all redundant contigs and those with no predicted protein product are
669 excluded from the final assembly, these contigs are saved in separate files and available for
670 additional analyses.

671 ConSemble assembly can be performed based on a smaller or larger number of the assembly
672 overlap. When only two or more assembly overlap was required (ConSemble2+d and
673 ConSemble2+g in Tables S6 and S9 in Additional file 2), more correctly assembled contigs (*TP*)
674 were recovered but at the cost of disproportionally more incorrectly assembled contigs
675 (*FP>>TP*) leading to higher Recall but lower Precision than ConSemble3+ (*e.g.,* Recall=0.74
676 and 0.84 and Precision=0.27 and 0.79, respectively, for No0-NoAlt). When the 4-way consensus
677 was used (ConSemble4d and ConSemble4g), while fewer correctly assembled contigs were
678 recovered than ConSemble3+ (*e.g*., Recall=0.65 and 0.50 for No0-NoAlt, respectively), it
679 reduced the number of incorrectly assembled contigs (*FP*) significantly only marginally
680 increasing the number of missing benchmark transcripts (*FN*). This led to significantly higher
681 Precision *(e.g*., 0.89 and 0.95 for No0-Alt, respectively) compared to ConSemble3+. Based on
682 these results, we chose ConSemble3+ as the main output of the ConSemble pipeline. Note that,
683 with its very high Precision, the ConSemble4 assembly can be used as part of post-processing of
684 the ConSemble3+ assembly to obtain a contig set with the highest confidence.

685 ConSemble is implemented in Perl. Source codes and test data are freely available at:
686 http://bioinfolab.unl.edu/emlab/consemble/
687

688 **Assembly benchmarking**

689 The experimental design we used to evaluate the assembly performance is summarized in Table
690 S2 in Additional file 2. The assembly benchmarking process is summarized in the Pipeline 2 in

691 Additional file 1. Each contig assembly was evaluated based on the accuracy of the coded
692 protein sequences in the ORFs predicted by ORFfinder [49]. This was done to avoid issues with
693 variation in how the assemblers define the 3' and 5'-ends of the contigs. Only contigs that fully
694 covered the benchmark protein sequences without any mismatches or gaps were considered
695 correctly assembled (true positives, TPs). Contigs with any amino acid mismatches and gaps or
696 those corresponding to benchmark sequences only partially are considered false positives (FPs).
697 Any transcript sequences in the benchmark dataset that were not assembled completely correctly
698 were considered false negatives (FNs).
699

700 **Performance statistics**

701 The overall performance of the assembly is determined as follows:

702 - *TP*: number of correctly assembled contigs
703 - *FP*: number of incorrectly assembled contigs
704 - *FN*: number of benchmark transcripts that are not assembled
705 - Recall (or sensitivity) = $TP/(TP+FN)$
706 - Precision = $TP/(TP+FP)$
707 - F-measure ($F$ or $F_1$) = $2(TP)/[2(TP)+FP+FN]$

708 Precision shows the proportion of correctly assembled contigs relative to all assembled contigs.
709 Recall also shows the proportion of correctly assembled contigs but relative to the number of
710 transcripts in the reference (actual positives). F-measure is a combined metric, the harmonic
711 mean of Precision and Recall. It therefore balances the concerns of Recall and Precision. F-
712 measure also does not require the number of true negatives (*TN*) to be known. This is useful

713 because a true negative is not defined for transcriptome assembly in this benchmarking. The
714 regular accuracy score, which is defined as ($TP+TN$)/($TP+FP+TN+FN$), also requires $TN$ and
715 cannot be calculated.
716

717 **Isoform detection performance**

718 As described before, only isoforms that produce distinct protein sequences are considered
719 different. This approach does not account for biologically important alternative splice events in
720 the UTRs, which can affect protein trafficking or translation without affecting protein sequences.
721 To simplify the analysis, the performance of the isoform detection was evaluated solely by the
722 number of correctly assembled isoforms without considering the abundance of each isoform.
723 Contigs that did not match completely any of the isoforms included in the benchmark dataset
724 were counted as incorrectly assembled (FP).
725

726 **Lower identity threshold**

727 To evaluate the effect of using % identity thresholds lower than 100% to identify TPs, identities
728 between each assembled contig and the benchmark transcript at the translated protein level, was
729 also calculated by dividing the edit distance by the length of the global alignment using the
730 python library for Edlib (version 1.3.8post2) [50].
731

732 **Venn diagram generation**

733 The contig overlap between assemblies was evaluated based on the coded protein sequences. The
734 Venn diagrams were generated using jvenn (version 1.0) [51].
735

736 **Assembly performance at the nucleotide level**

737 The quality of the nucleotide sequences produced by all of the assembly methods were examined
738 using DETONATE (version 1.11) [19]. DETONATE provides a reference-free model-based
739 RSEM-EVAL score as well as reference-based scores including the nucleotide and contig-level
740 $F_1$ and KC (kmer compression). RSEM-EVAL scores how well the contigs are supported by the
741 RNAseq data as well as the compactness of an assembly based on their joint probability. The
742 thresholds used to identify the contig-level TP to calculate $F_1$ score (see the formula above) is
743 99% identity and having ≤1% of insertions and deletions. The KC score is a combination of the
744 weighted kmer recall and inverse compression rate, the latter of which penalizes large
745 assemblies.
746

747 **BUSCO**

748 The thoroughness of the assemblies generated from the real plant RNAseq libraries was assessed
749 based on the number of complete genes identified from the Eudicotyledons obd10 dataset of
750 BUSCO (version 3.1.0) [20].
751

752 **Transcript expression level analysis**

To determine the impact of expression level on transcript assembly, the benchmark transcriptome was divided into ten equally sized bins by the expression level. The proportion of transcripts in each bin correctly assembled is determined by the number of benchmark transcripts with at least one exact match (full length with no gaps or mismatches) in an assembly divided by the total number of transcripts in the bin.

# Declarations

## Ethics approval and consent to participate

Not applicable.

## Consent for publication

Not applicable.

## Availability of data and materials

The source codes, benchmark datasets, and other datasets used and analyzed in this study are available from: http://bioinfolab.unl.edu/emlab/consemble/ .

## Competing interests

The authors declare that they have no competing interests.

## Funding

This work has been supported by the National Science Foundation under Grant Nos. 1339385 to JS, EBC, and ENM and 1557417 to EBC and ENM. The funder had no role in study design, data collection and analysis, decision to publish, or preparation of work included in this submission.

## Author's contributions

AV and ENM conceived and designed the research. AV wrote and tested the programs. XL, XHY, JS, and EBC generated RNAseq data. AV, SB, KK, and ENM analyzed and interpreted the results. JSD also interpreted the results. AV and ENM wrote and revised the manuscript. All authors read, discussed, and approved the manuscript.

## Acknowledgements

We would like to thank Dr. Don Gilbert (Indiana University) for his discussions and critical feedback regarding EvidentialGene and its performance.

787 ## References


788 1. Huang X, Chen XG, Armbruster PA: **Comparative performance of transcriptome**
789 **assembly methods for non-model organisms**. *BMC Genomics* 2016, **17**:523.
790 2. Wang S, Gribskov M: **Comprehensive evaluation of *de novo* transcriptome assembly**
791 **programs and their effects on differential gene expression analysis**. *Bioinformatics* 2017,
792 **33**(3):327-333.
793 3. Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, McPherson A, Szczesniak
794 MW, Gaffney DJ, Elo LL, Zhang X *et al*: **A survey of best practices for RNA-seq data**
795 **analysis**. *Genome Biol* 2016, **17**:13.
796 4. Voshall A, Moriyama EN: **Next-Generation Transcriptome Assembly: Strategies and**
797 **Performance Analysis**. In: *Bioinformatics in the Era of Post Genomics and Big Data.* Edited
798 by Adburakhmonov IY. Rijeka: IntechOpen; 2018.
799 5. Simonis M, Atanur SS, Linsen S, Guryev V, Ruzius FP, Game L, Lansu N, de Bruijn E, van
800 Heesch S, Jones SJ *et al*: **Genetic basis of transcriptome differences between the**
801 **founder strains of the rat HXB/BXH recombinant inbred panel**. *Genome Biol* 2012,
802 **13**(4):r31.
803 6. Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X, Fan L,
804 Raychowdhury R, Zeng Q *et al*: **Full-length transcriptome assembly from RNA-Seq data**
805 **without a reference genome**. *Nature biotechnology* 2011, **29**(7):644-652.
806 7. Martin JA, Wang Z: **Next-generation transcriptome assembly**. *Nat Rev Genet* 2011,
807 **12**(10):671-682.
808 8. Gruenheit N, Deusch O, Esser C, Becker M, Voelckel C, Lockhart P: **Cutoffs and k-mers:**
809 **implications from a transcriptome study in allopolyploid plants**. *BMC Genomics* 2012,
810 **13**:92.
811 9. Gilbert D: **Gene-omes built from mRNA seq not genome DNA**. In: *7th Annual Arthropod*
812 *Genomics Symposium: June 13-June 15 2013; Notre Dame, IN*.

813 10. Durai DA, Schulz MH: **Informed kmer selection for de novo transcriptome assembly**.
814 *Bioinformatics* 2016, **32**(11):1670-1677.
815 11. Rana SB, Zadlock FJt, Zhang Z, Murphy WR, Bentivegna CS: **Comparison of De Novo**
816 **Transcriptome Assemblers and k-mer Strategies Using the Killifish, Fundulus**
817 **heteroclitus**. *PloS one* 2016, **11**(4):e0153104.
818 12. Holding ML, Margres MJ, Mason AJ, Parkinson CL, Rokyta DR: **Evaluating the Performance**
819 **of De Novo Assembly Methods for Venom-Gland Transcriptomics**. *Toxins (Basel)* 2018,
820 **10**(6).
821 13. Holzer M, Marz M: **De novo transcriptome assembly: A comprehensive cross-species**
822 **comparison of short-read RNA-Seq assemblers**. *Gigascience* 2019, **8**(5).
823 14. Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, Pimentel H, Salzberg SL, Rinn JL,
824 Pachter L: **Differential gene and transcript expression analysis of RNA-seq experiments**
825 **with TopHat and Cufflinks**. *Nat Protoc* 2012, **7**(3):562-578.
826 15. Smith-Unna R, Boursnell C, Patro R, Hibberd JM, Kelly S: **TransRate: reference-free quality**
827 **assessment of *de novo* transcriptome assemblies**. *Genome Res* 2016, **26**(8):1134-1144.
828 16. Gilbert DG: **Genes of the pig, *Sus scrofa*, reconstructed with EvidentialGene**. *PeerJ* 2019,
829 **7**:e6374.
830 17. Cerveau N, Jackson DJ: **Combining independent *de novo* assemblies optimizes the**
831 **coding transcriptome for nonconventional model eukaryotic organisms**. *BMC*
832 *Bioinformatics* 2016, **17**(1):525.
833 18. Yu T, Mu Z, Fang Z, Liu X, Gao X, Liu J: **TransBorrow: genome-guided transcriptome**
834 **assembly by borrowing assemblies from different assemblers**. *Genome Res* 2020,
835 **30**(8):1181-1190.
836 19. Li B, Fillmore N, Bai Y, Collins M, Thomson JA, Stewart R, Dewey CN: **Evaluation of *de***
837 ***novo* transcriptome assemblies from RNA-Seq data**. *Genome Biol* 2014, **15**(12):553.

838 20. Simao FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM: **BUSCO: assessing**
839 **genome assembly and annotation completeness with single-copy orthologs**.
840 *Bioinformatics* 2015, **31**(19):3210-3212.
841 21. Yang Y, Gribskov M: **The evaluation of RNA-Seq *de novo* assembly by PacBio long read**
842 **sequencing**. *bioRxiv* 2019:735621.
843 22. Xie Y, Wu G, Tang J, Luo R, Patterson J, Liu S, Huang W, He G, Gu S, Li S *et al*:
844 **SOAPdenovo-Trans: de novo transcriptome assembly with short RNA-Seq reads**.
845 *Bioinformatics* 2014, **30**(12):1660-1666.
846 23. Peng Y, Leung HC, Yiu SM, Lv MJ, Zhu XG, Chin FY: **IDBA-tran: a more robust de novo**
847 **de Bruijn graph assembler for transcriptomes with uneven expression levels**.
848 *Bioinformatics* 2013, **29**(13):i326-334.
849 24. Bushmanova E, Antipov D, Lapidus A, Prjibelski AD: **rnaSPAdes: a *de novo* transcriptome**
850 **assembler and its application to RNA-Seq data**. *Gigascience* 2019, **8**(9).
851 25. Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS, Lesin VM, Nikolenko
852 SI, Pham S, Prjibelski AD *et al*: **SPAdes: a new genome assembly algorithm and its**
853 **applications to single-cell sequencing**. *J Comput Biol* 2012, **19**(5):455-477.
854 26. Maretty L, Sibbesen JA, Krogh A: **Bayesian transcriptome assembly**. *Genome Biol* 2014,
855 **15**(10):501.
856 27. Shao M, Kingsford C: **Accurate assembly of transcripts through phase-preserving graph**
857 **decomposition**. *Nature biotechnology* 2017, **35**(12):1167-1169.
858 28. Pertea M, Pertea GM, Antonescu CM, Chang TC, Mendell JT, Salzberg SL: **StringTie**
859 **enables improved reconstruction of a transcriptome from RNA-seq reads**. *Nature*
860 *biotechnology* 2015, **33**(3):290-295.
861 29. Kovaka S, Zimin AV, Pertea GM, Razaghi R, Salzberg SL, Pertea M: **Transcriptome**
862 **assembly from long-read RNA-seq alignments with StringTie2**. *Genome Biol* 2019,
863 **20**(1):278.

30. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, Batut P, Chaisson M, Gingeras TR: **STAR: ultrafast universal RNA-seq aligner**. *Bioinformatics* 2013, **29**(1):15-21.
31. Baruzzo G, Hayer KE, Kim EJ, Di Camillo B, FitzGerald GA, Grant GR: **Simulation-based comprehensive benchmarking of RNA-seq aligners**. *Nat Methods* 2017, **14**(2):135-139.
32. Watson M, Warr A: **Errors in long-read assemblies can critically affect protein prediction**. *Nature biotechnology* 2019, **37**(2):124-126.
33. Voshall A, Moriyama EN: **Next-generation transcriptome assembly and analysis: Impact of ploidy**. *Methods* 2020, **176**:14-24.
34. Gan X, Stegle O, Behr J, Steffen JG, Drewe P, Hildebrand KL, Lyngsoe R, Schultheiss SJ, Osborne EJ, Sreedharan VT *et al*: **Multiple reference genomes and transcriptomes for Arabidopsis thaliana**. *Nature* 2011, **477**(7365):419-423.
35. 1001 Genomes Consortium: **1,135 genomes reveal the global pattern of polymorphism in *Arabidopsis thaliana***. *Cell* 2016, **166**(2):481-491.
36. Swarbreck D, Wilks C, Lamesch P, Berardini TZ, Garcia-Hernandez M, Foerster H, Li D, Meyer T, Muller R, Ploetz L *et al*: **The Arabidopsis Information Resource (TAIR): gene structure and function annotation**. *Nucleic Acids Res* 2008, **36**(Database issue):D1009-1014.
37. Zhang R, Calixto CP, Tzioutziou NA, James AB, Simpson CG, Guo W, Marquez Y, Kalyna M, Patro R, Eyras E *et al*: **AtRTD - a comprehensive reference transcript dataset resource for accurate quantification of transcript-specific expression in *Arabidopsis thaliana***. *New Phytol* 2015, **208**(1):96-101.
38. Schneider VA, Graves-Lindsay T, Howe K, Bouk N, Chen HC, Kitts PA, Murphy TD, Pruitt KD, Thibaud-Nissen F, Albracht D *et al*: **Evaluation of GRCh38 and *de novo* haploid genome assemblies demonstrates the enduring quality of the reference assembly**. *Genome Res* 2017, **27**(5):849-864.

890 39. Marquez Y, Brown JW, Simpson C, Barta A, Kalyna M: **Transcriptome survey reveals**
891 **increased complexity of the alternative splicing landscape in Arabidopsis**. *Genome Res*
892 2012, **22**(6):1184-1195.
893 40. Urasaki N, Takagi H, Natsume S, Uemura A, Taniai N, Miyagi N, Fukushima M, Suzuki S,
894 Tarora K, Tamaki M *et al*: **Draft genome sequence of bitter gourd (*Momordica charantia*),**
895 **a vegetable and medicinal plant in tropical and subtropical regions**. *DNA Res* 2017,
896 **24**(1):51-58.
897 41. Zhang T, Hu Y, Jiang W, Fang L, Guan X, Chen J, Zhang J, Saski CA, Scheffler BE, Stelly
898 DM *et al*: **Sequencing of allotetraploid cotton (*Gossypium hirsutum* L. acc. TM-1)**
899 **provides a resource for fiber improvement**. *Nature Biotechnol* 2015, **33**:531.
900 42. Griebel T, Zacher B, Ribeca P, Raineri E, Lacroix V, Guigo R, Sammeth M: **Modelling and**
901 **simulating generic RNA-Seq experiments with the flux simulator**. *Nucleic Acids Res*
902 2012, **40**(20):10073-10083.
903 43. Del Fabbro C, Scalabrin S, Morgante M, Giorgi FM: **An extensive evaluation of read**
904 **trimming effects on Illumina NGS data analysis**. *PloS one* 2013, **8**(12):e85024.
905 44. Crusoe MR, Alameldin HF, Awad S, Boucher E, Caldwell A, Cartwright R, Charbonneau A,
906 Constantinides B, Edvenson G, Fay S *et al*: **The khmer software package: enabling**
907 **efficient nucleotide sequence analysis**. *F1000Res* 2015, **4**:900.
908 45. Zerbino DR, Birney E: **Velvet: algorithms for de novo short read assembly using de**
909 **Bruijn graphs**. *Genome Res* 2008, **18**(5):821-829.
910 46. Schulz MH, Zerbino DR, Vingron M, Birney E: **Oases: robust de novo RNA-seq assembly**
911 **across the dynamic range of expression levels**. *Bioinformatics* 2012, **28**(8):1086-1092.
912 47. Chevreux B, Pfisterer T, Drescher B, Driesel AJ, Muller WEG, Wetter T, Suhai S: **Using the**
913 **miraEST assembler for reliable and automated mRNA transcript assembly and SNP**
914 **detection in sequenced ESTs**. *Genome Res* 2004, **14**(6):1147-1159.
915 48. Trapnell C, Pachter L, Salzberg SL: **TopHat: discovering splice junctions with RNA-Seq**.
916 *Bioinformatics* 2009, **25**(9):1105-1111.

917 49. Wheeler DL, Church DM, Federhen S, Lash AE, Madden TL, Pontius JU, Schuler GD, Schriml
918 LM, Sequeira E, Tatusova TA *et al*: **Database resources of the National Center for**
919 **Biotechnology**. *Nucleic Acids Res* 2003, **31**(1):28-33.
920 50. Sosic M, Sikic M: **Edlib: a C/C ++ library for fast, exact sequence alignment using edit**
921 **distance**. *Bioinformatics* 2017, **33**(9):1394-1395.
922 51. Bardou P, Mariette J, Escudie F, Djemiel C, Klopp C: **jvenn: an interactive Venn diagram**
923 **viewer**. *BMC Bioinformatics* 2014, **15**:293.
924 52. Cheng CY, Krishnakumar V, Chan AP, Thibaud-Nissen F, Schobel S, Town CD: **Araport11:**
925 **a complete reannotation of the *Arabidopsis thaliana* reference genome**. *Plant J* 2017,
926 **89**(4):789-804.
927

928

# Additional files

## Additional file 1

Format: PDF

File content: Pipelines 1-4.

## Additional file 2

Format: PDF

File content: Supplementary Tables S1-S13.

## Additional file 3

Format: PDF

File content: Supplementary Figs. S1-S4.

942 **Table 1.** Performance analysis of real plant transcriptome assembly

| Assembler | *A. thaliana* Col-0 (48,359)[a] | | | *M. charantia* (45,859)[a] | | | *G. hirsutum* (70,478)[a] | | |
|---|---|---|---|---|---|---|---|---|---|
| | # of contigs[b] | BUSCO[c] | EVAL[d] | # of contigs[b] | BUSCO[c] | EVAL[d] | # of contigs[b] | BUSCO[c] | EVAL[d] |
| **[Genome-guided]** | | | | | | | | | |
| Bayesembler | 45,583 (94) [10,611] | 1,677 | -3.18 | 44,524 (97) | 1,629 | -1.99 | NA[e] | NA[e] | NA[e] |
| Cufflinks | 34,771 (72) [9,886] | 1,593 | -2.95 | 38,655 (84) | 1,662 | -1.64 | 98,590 (140) | 1,908 | -3.44 |
| Scallop | 73,172 (151) [13,272] | 1,841 | -2.82 | 68,593 (150) | 1,855 | -1.49 | 183,625 (261) | 1,967 | -3.32 |
| StringTie2 | 37,211 (77) [12,619] | 1,819 | -2.82 | 44,149 (90) | 1,847 | -1.51 | 108,389 (154) | 1,998 | -3.24 |
| **[*De novo*]** | | | | | | | | | |
| IDBA-Tran | 52,982 (110) [6,871] | 1,287 | -2.92 | 106,170 (232) | 657 | -1.82 | 215,470 (306) | 411 | -5.71 |
| rnaSPAdes | 101,444 (210) [8,521] | 1,475 | -2.74 | 91,791 (200) | 1,051 | -1.50 | 211,024 (299) | 1,533 | -3.66 |
| SOAPdenovo | 68,141 (141) [8,973] | 1,689 | -2.93 | 115,117 (251) | 1,776 | -1.47 | 224,718 (318) | 971 | -6.22 |
| Trinity | 50,023 (103) [7,501] | 1,573 | -2.89 | 90,347 (197) | 1,779 | -1.44 | 212,852 (302) | 1,921 | -3.60 |
| **[Ensemble]** | | | | | | | | | |
| EvidentialGene | 83,749 (173) [10,892] | 1,899 | -2.87 | 148,938 (324) | 1,983 | -1.60 | 726,828 (1,031) | 2,070 | -4.07 |
| Concatenation | 70,444 (146) [12,184] | 1,855 | -2.71 | 66,613 (145) | 1,923 | -1.53 | 97,419 (138) | 2,010 | -3.76 |
| ConSemble3+d | 49,713 (103) [12,659] | 1,833 | -2.73 | 90,781 (198) | 1,858 | -1.42 | 114,974 (163) | 1,312 | -5.53 |
| ConSemble3+g | 24,355 (50) [11,322] | 1,587 | -1.18 | 24,699 (74) | 1,718 | -2.20 | 49,530 (54) | 1,689 | -1.13 |
| TransBorrow | 1,521 (3) [734] | 29 | -2.40 | 53,191 (116) | 1,896 | -1.66 | 6,234 (9) | 79 | -2.83 |

943 [a]Number of protein-coding transcripts reported [36, 40, 41].

944 [b]Proportion (%) of unique protein sequences in the assembly relative to the number of proteins in
945 the reference transcriptome is shown in parentheses. Number of contigs whose sequences
946 matched those in the Araport11 reference transcriptome [52] is shown in square brackets.

947 [c]Number of complete BUSCOs identified from the 2,121 orthologs in the Eudicot set.

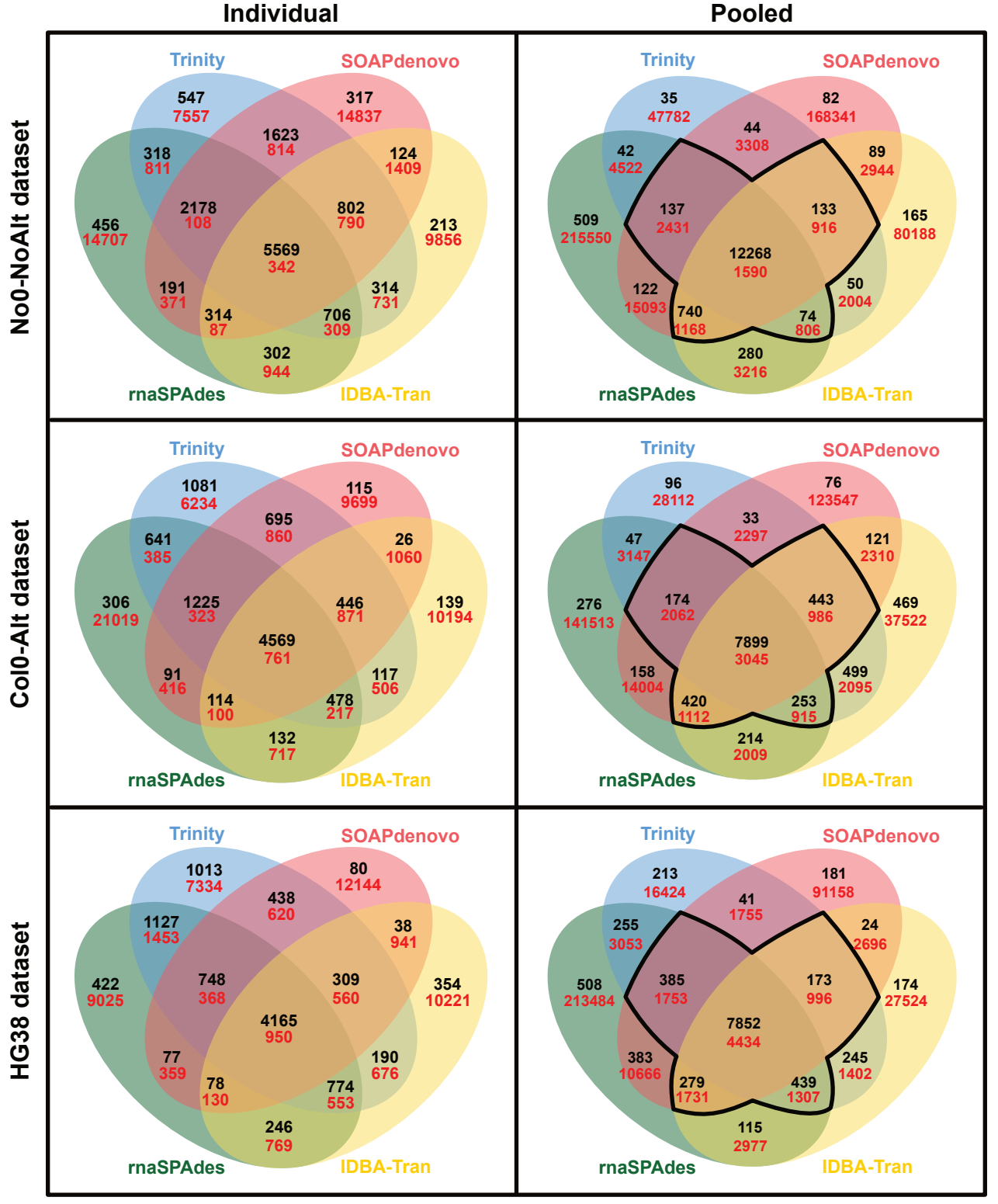
948 [d]RSEM-EVAL scores ($x10^{-9}$) from DETONATE.

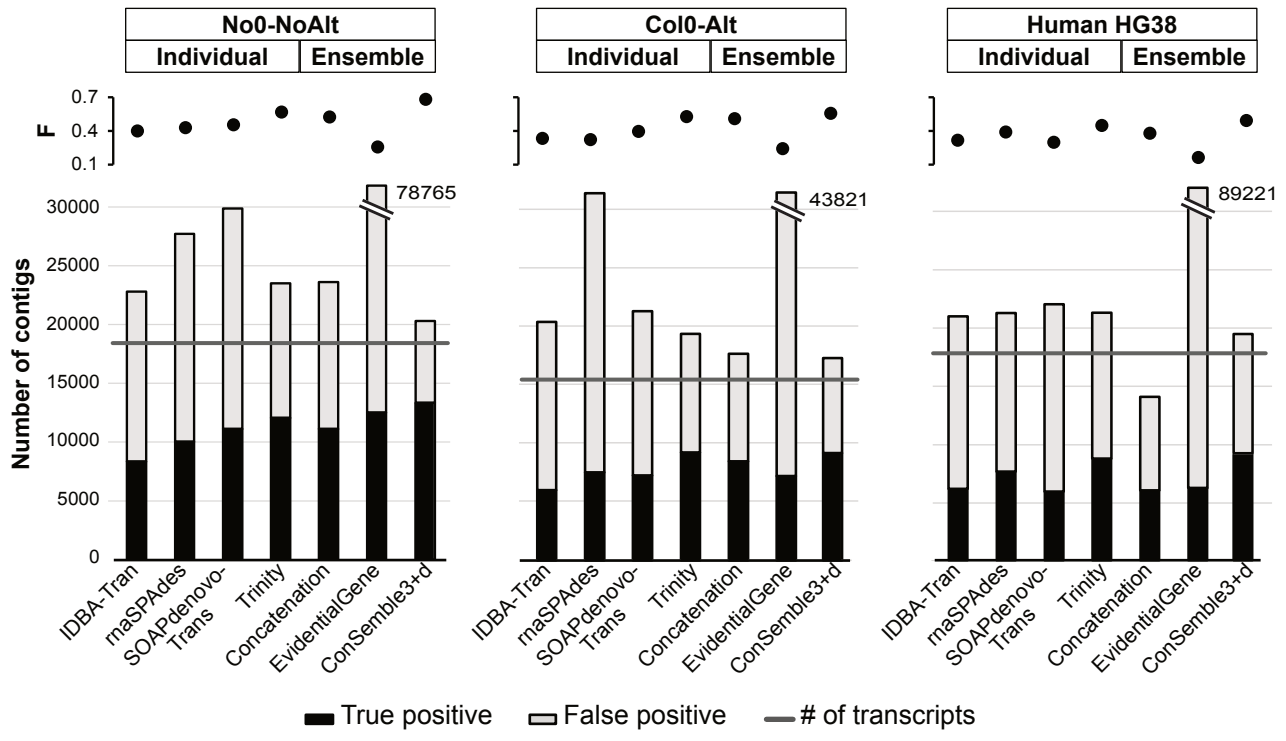
949 [e]Bayesembler was unable to run on this dataset.

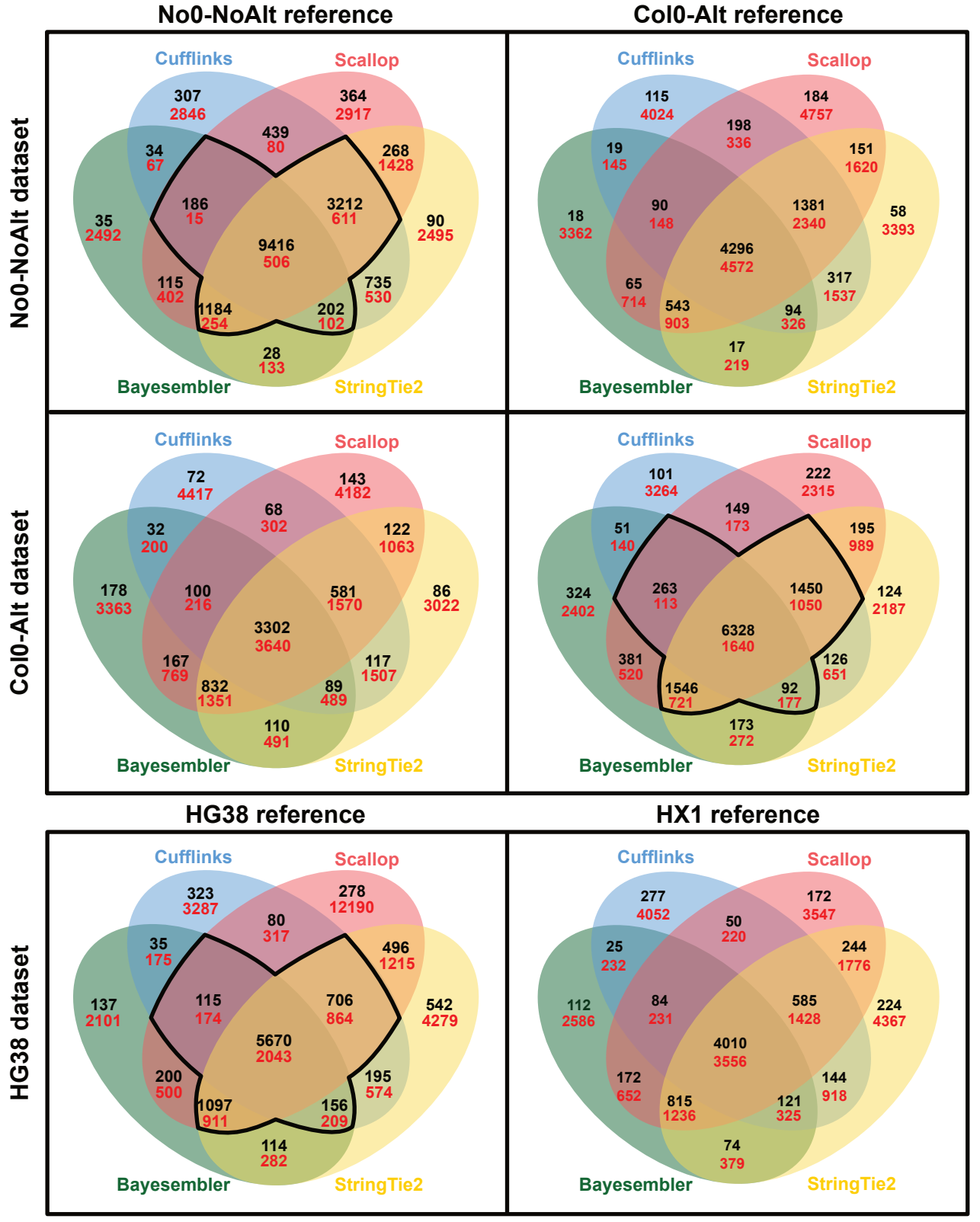
950 ## Figure legends

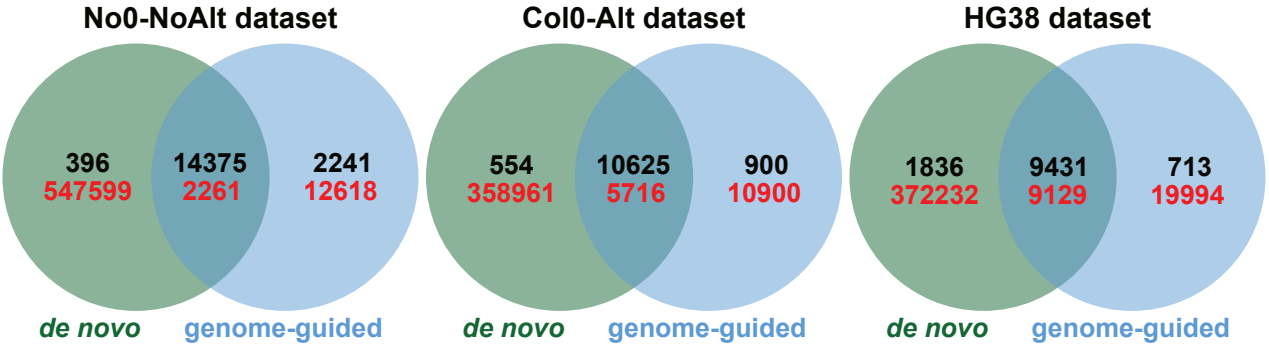
951 **Fig. 1 Numbers of assembled contigs shared between the four *de novo* assemblers.** The
952 numbers of correctly (black) and incorrectly (red) assembled contigs are shown. The three
953 benchmark datasets (No0-NoAlt, Col0-Alt, and Human HG38) were assembled by the four *de*
954 *novo*. Results based on individual methods using the default settings are shown under
955 "Individual". Results based on comparing the "pooled unique" contig sets assembled using the
956 four methods with multiple kmer lengths are shown under "Pooled". All contigs were compared
957 at the protein level. The outlined region represents where the shared correct and incorrect
958 contigs were counted for the ConSemble3+d assembly (shown as *TP* and *FP* in Table S6 in
959 Additional file 2).
960
961 **Fig. 2 Comparison of *de novo* assembler performance on the three benchmark datasets.**
962 For the individual *de novo* assemblers, results shown were obtained with their default settings.
963 See Tables S3 and S6 in Additional file 2 for details.
964
965 **Fig. 3 Numbers of assembled contigs shared between the four genome-guided**
966 **assemblers.** Rows and columns are based on the simulated RNAseq dataset and the reference
967 genome used for the transcriptome assembly. The numbers of correctly (black) and incorrectly
968 (red) assembled contigs are shown. All contigs were compared at the protein level. The outlined
969 region represents where the shared correct and incorrect contigs were counted for the
970 ConSemble3+g assembly using the same reference genomes (shown as *TP* and *FP* in Table
971 S9 in Additional file 2).
972

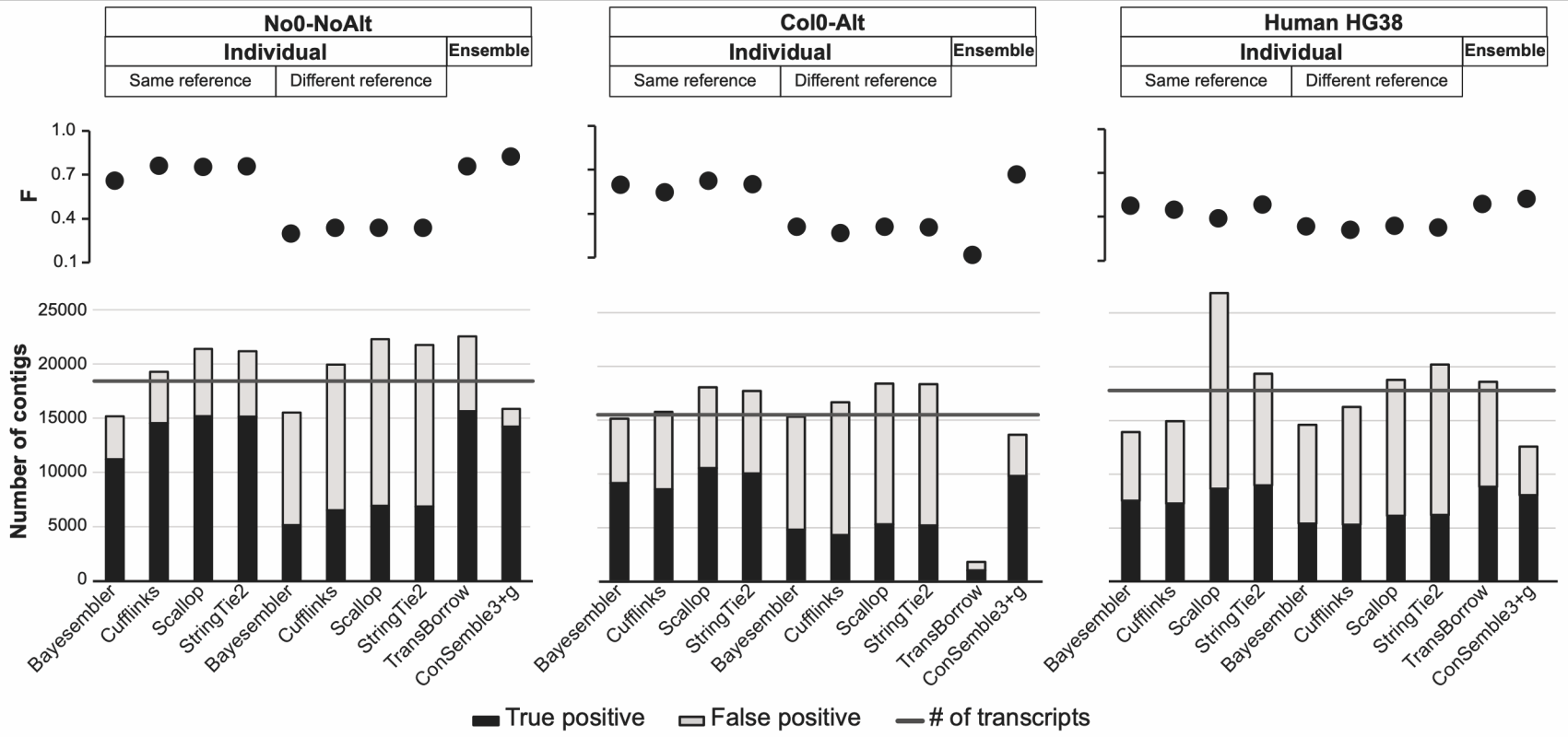
973 **Fig. 4 Numbers of assembled contigs shared between *de novo* and genome-guided**
974 **assemblies.** The "Merged" assemblies in Table S5 in Additional file 2 were used for the *de*
975 *novo* assembly datasets. The genome-guided assembly is the union set of the assemblies
976 generated by the four genome-guided methods using the same reference genomes (Tests 4, 6,
977 and 8 in Table S2 in Additional file 2). The numbers of correctly (black) and incorrectly (red)
978 assembled contigs are shown. All contigs were compared at the protein level.
979
980 **Fig. 5 Comparison of genome-guided assembler performance on the three benchmark**
981 **datasets.** See Tables S7 and S9 in Additional file 2 for details.
982


Individual
Pooled
No0-NoAlt dataset
Col0-Alt dataset
HG38 dataset
Trinity
SOAPdenovo
rnaSPAdes
IDBA-Tran
547
7557
317
14837
1623
814
318
811
124
1409
2178
108
802
790
456
14707
213
9856
5569
342
191
371
314
731
314
87
706
309
302
944
35
47782
82
168341
44
3308
42
4522
89
2944
137
2431
133
916
509
215550
165
80188
12268
1590
122
15093
50
2004
740
1168
74
806
280
3216
1081
6234
115
9699
695
860
641
385
26
1060
306
21019
1225
323
446
871
139
10194
4569
761
91
416
117
506
114
100
478
217
132
717
96
28112
76
123547
33
2297
47
3147
121
2310
276
141513
174
2062
443
986
469
37522
7899
3045
158
14004
499
2095
420
1112
253
915
214
2009
1013
7334
80
12144
438
620
1127
1453
38
941
422
9025
748
368
309
560
354
10221
4165
950
77
359
190
676
78
130
774
553
246
769
213
16424
181
91158
41
1755
255
3053
24
2696
508
213484
385
1753
173
996
174
27524
7852
4434
383
10666
245
1402
279
1731
439
1307
115
2977


No0-NoAlt
Individual
Ensemble
Col0-Alt
Individual
Ensemble
Human HG38
Individual
Ensemble
F
0.7
0.4
0.1
Number of contigs
30000
25000
20000
15000
10000
5000
0
78765
43821
89221
IDBA-Tran
rnaSPAdes
SOAPdenovo-Trans
Trinity
Concatenation
EvidentialGene
ConSemble3+d
True positive
False positive
# of transcripts


No0-NoAlt reference
Col0-Alt reference
HG38 reference
HX1 reference
No0-NoAlt dataset
Col0-Alt dataset
HG38 dataset
Cufflinks
Scallop
Bayesembler
StringTie2
307
2846
364
2917
439
80
34
67
268
1428
186
15
3212
611
35
2492
90
2495
9416
506
115
402
735
530
1184
254
202
102
28
133
115
4024
184
4757
198
336
19
145
151
1620
18
3362
90
148
1381
2340
58
3393
4296
4572
65
714
317
1537
543
903
94
326
17
219
72
4417
143
4182
68
302
32
200
122
1063
178
3363
100
216
581
1570
86
3022
3302
3640
167
769
117
1507
832
1351
89
489
110
491
101
3264
222
2315
149
173
51
140
195
989
324
2402
263
113
1450
1050
124
2187
6328
1640
381
520
126
651
1546
721
92
177
173
272
323
3287
278
12190
80
317
35
175
496
1215
137
2101
115
174
706
864
542
4279
5670
2043
200
500
195
574
1097
911
156
209
114
282
277
4052
172
3547
50
220
25
232
244
1776
112
2586
84
231
585
1428
224
4367
4010
3556
172
652
144
918
815
1236
121
325
74
379


No0-NoAlt dataset
396
547599
14375
2261
2241
12618
de novo
genome-guided
Col0-Alt dataset
554
358961
10625
5716
900
10900
de novo
genome-guided
HG38 dataset
1836
372232
9431
9129
713
19994
de novo
genome-guided


No0-NoAlt
Individual
Ensemble
Same reference
Different reference
Col0-Alt
Individual
Ensemble
Same reference
Different reference
Human HG38
Individual
Ensemble
Same reference
Different reference
F
1.0
0.7
0.4
0.1
Number of contigs
25000
20000
15000
10000
5000
0
Bayesembler
Cufflinks
Scallop
StringTie2
Bayesembler
Cufflinks
Scallop
StringTie2
TransBorrow
ConSemble3+g
True positive
False positive
# of transcripts