

1 **Title:**  
2 debar, a sequence-by-sequence denoiser for COI-5P DNA barcode data  
3

4 **Authors**

5 Cameron M. Nugent<sup>1,2,\*</sup>  
6 Tyler A. Elliott<sup>2</sup>  
7 Sujeevan Ratnasingham<sup>2</sup>  
8 Paul D. N. Hebert<sup>2</sup>  
9 Sarah J. Adamowicz<sup>1</sup>

10  
11 <sup>1</sup>Department of Integrative Biology, University of Guelph. Guelph, Ontario, Canada

12 <sup>2</sup>Centre for Biodiversity Genomics, University of Guelph. Guelph, Ontario, Canada

13 \*Corresponding author: [nugentc@uoguelph.ca](mailto:nugentc@uoguelph.ca)

14

15

16 **Abstract**

17

18 DNA barcoding and metabarcoding are now widely used to advance species discovery and  
19 biodiversity assessments. High-throughput sequencing (HTS) has expanded the volume and  
20 scope of these analyses, but elevated error rates introduce noise into sequence records that can  
21 inflate estimates of biodiversity. Denoising—the separation of biological signal from instrument  
22 (technical) noise—of barcode and metabarcode data currently employs abundance-based  
23 methods which do not capitalize on the highly conserved structure of the cytochrome *c* oxidase  
24 subunit I (COI) region employed as the animal barcode. This manuscript introduces debar, an R  
25 package that utilizes a profile hidden Markov model to denoise indel errors in COI sequences  
26 introduced by instrument error. In silico studies demonstrated that debar recognized 95% of  
27 artificially introduced indels in COI sequences. When applied to real-world data, debar reduced  
28 indel errors in circular consensus sequences obtained with the Sequel platform by 75%, and  
29 those generated on the Ion Torrent S5 by 94%. The false correction rate was less than 0.1%,  
30 indicating that debar is receptive to the majority of true COI variation in the animal kingdom. In  
31 conclusion, the debar package improves DNA barcode and metabarcode workflows by aiding the  
32 generation of more accurate sequences aiding the characterization of species diversity.

33

34 **Keywords:** COI, DNA barcode, metabarcode, denoising, Markov model, biodiversity

## 35 **Introduction**

36

37 Motivated by global biodiversity decline, conservation policies and strategies are being  
38 implemented to mitigate extinction rates (Driscoll *et al.* 2018; Baynham-Herd *et al.* 2018).  
39 Accurate assessments of biodiversity and its change over time are critical to support conservation  
40 strategies, to remediate environmental damage, and to manage natural resources, but this  
41 information is lacking for most ecosystems (Sogin *et al.* 2006; Hajibabaei *et al.* 2016; Hebert *et*  
42 *al.* 2016; D’Souza & Hebert 2018).

43 DNA barcoding provides a technological solution to the problem of identifying  
44 organisms and characterizing biodiversity (Hebert *et al.* 2003; Hubert & Hanner 2015). Instead  
45 of identifying specimens through morphological study, standardized DNA regions—termed  
46 DNA barcodes—are used to identify specimens belonging to known species and to recognize  
47 new taxa. Reflecting advances in sequencing technology, DNA barcode studies are expanding in  
48 scale from analyzing single specimens to characterizing bulk samples, an approach termed  
49 metabarcoding, as well as multi-marker and metagenomics approaches (Taberlet *et al.* 2012;  
50 Cristescu 2014; Hajibabaei *et al.* 2016; Wilson *et al.* 2019). These advances are providing newly  
51 detailed information on species diversity in different geographic regions and habitats (Hajibabaei  
52 *et al.* 2012; Hebert *et al.* 2016; Delabye *et al.* 2019; Lopez-Vaamonde *et al.* 2019) while also  
53 aiding the identification of invasive species (Brown *et al.* 2016; Xu *et al.* 2017), food web  
54 analysis (Wirta *et al.* 2014; Kanuisto *et al.* 2017), and environmental monitoring (Hajibabaei *et*  
55 *al.* 2016; Stat *et al.* 2017; Cordier *et al.* 2019).

56 Despite the broad adoption of DNA barcoding and metabarcoding, a fundamental  
57 problem persists. Efforts to quantify biodiversity from barcode and metabarcode data can be

58 strongly affected by analytical methodology (Clare *et al.* 2016; Braukmann *et al.* 2019). For  
59 example, if high-throughput sequence (HTS) data are cleaned suboptimally, the estimated  
60 number of taxa may be grossly inflated as variation introduced by sequencing (technical) errors  
61 are interpreted as biological variation (Hardge *et al.* 2018).

62 To reduce the impact of technical errors, sequence reads are often clustered into  
63 operational taxonomic units (OTUs) at specific identity thresholds (Elbrecht *et al.* 2018). Several  
64 software packages have attempted to increase the accuracy of this OTU method by separating  
65 biological signal from technical noise (Rosen *et al.* 2012; Callahan *et al.* 2016; Edgar 2016;  
66 Amir *et al.* 2017; Elbrecht *et al.* 2018; Kumar *et al.* 2018; Nearing *et al.* 2018). Many standard  
67 denoisers, such as DADA2 (Callahan *et al.* 2016), Deblur (Amir *et al.* 2017), and UNOISE  
68 (Edgar 2016), utilize cluster-based approaches, custom error models, or pre-clustering algorithms  
69 to account for and correct technical errors. Comparative studies have shown that all three of  
70 these methods outperform threshold-based OTU-clustering approaches (Nearing *et al.* 2018). It  
71 has also been shown that they produce similar estimates of species richness and relative  
72 abundance, but significantly different values for alpha diversity (intra-habitat diversity) and the  
73 number of unique exact sequence variants (ESVs) (Nearing *et al.* 2018). When a highly  
74 conserved protein-coding region, such as cytochrome *c* oxidase subunit I (COI), is employed as  
75 the barcode, structural information can be leveraged to improve denoising. The adoption of this  
76 approach can improve the accuracy of alpha-diversity estimates and the quality of identified  
77 barcode sequences by ensuring barcodes conform to biological reality. Additionally, rare  
78 sequences or important intra-species variants need not be discarded based solely on their  
79 abundance and can be retained with higher confidence if they conform to the expected gene  
80 structure. This latter benefit will be particularly valuable for work on hyper-diverse communities,

81 (e.g. tropical insects) and for analyses of metabarcode data, where uneven sampling is often the  
82 norm and the resolution of intra-species variation is challenging (Elbrecht *et al.* 2018; Nearing *et*  
83 *al.* 2018; Braukmann *et al.* 2019; Zizka *et al.* 2020).

84         Hidden Markov models (HMMs) are probabilistic representations of sequences that allow  
85 unobserved (hidden) states to be inferred through the observation of a series of non-hidden states  
86 (Durbin *et al.* 1998; Wilkinson 2019). HMMs have been applied widely in the analysis of  
87 biological sequences, in areas such as sequence alignment and annotation (Durbin *et al.* 1998;  
88 Eddy 1998). Profile Hidden Markov models (PHMMs) are a variant well suited for the  
89 representation of biological sequences with a shared evolutionary origin (Durbin *et al.* 1998;  
90 Eddy 1998, 2009). They are probabilistic models that contain position-specific information about  
91 the likelihood of potential characters (base pairs or amino acid residues) at the given position in  
92 the sequence (emission probabilities) and the likelihood of the observed character given the  
93 previously observed character in the sequence (transition probabilities). Once a PHMM is trained  
94 on a set of sequences, the Viterbi algorithm can be used to obtain the path of hidden states that  
95 align the novel sequences to the PHMM (Durbin *et al.* 1998). The Viterbi path is comprised of  
96 hidden match states (indicating the observed character matches to a position in the PHMM) and  
97 non-match states: either inserts or deletions. In the context of error correction, hidden non-match  
98 states identify the most likely positions at which novel sequences deviate from the PHMM's  
99 statistical profile. In this manner, individual sequences can be queried for evidence of insertion  
100 or deletion (indel) errors and adjusted in a statistically informed manner. The conserved protein-  
101 coding structure of the most common animal barcode gene, COI, and the wealth of available  
102 training sequences (Ratnasingham & Hebert 2007) for this region have allowed PHMMs to be  
103 successfully applied in the detection of technical errors in novel barcode sequences (Nugent *et*

104 *al.* 2020). Correction of technical indel errors in data from protein-coding barcode sequences is  
105 an important development as it maximizes the likelihood that both the nucleotide and amino acid  
106 sequences correspond to the true biological sequence. Mitigation of indels arising due to  
107 technical errors also makes sequence reads from a given specimen more directly comparable,  
108 allowing low-frequency point mutations to be eliminated when multiple reads are available for a  
109 given biological sequence. Here, we aim to extend the use of PHMMs in COI data processing to  
110 allow for the sequence-by-sequence correction (denoising) of technical errors.

111 This study had four primary goals: (1) design a denoising tool for COI barcode data that  
112 utilizes PHMMs to identify and correct insertion and deletion errors resulting from technical  
113 error; (2) test the tool's performance and optimize its default parameters by denoising a set of  
114 10,000 barcode sequences with artificially introduced indel errors; (3) develop, implement, and  
115 evaluate a workflow for denoising DNA barcode data produced through single-molecule, real  
116 time (SMRT) sequencing of 29,525 specimens on the Sequel platform (Pacific Biosciences); and  
117 (4) denoise a DNA metabarcode mock community data set using *debar* and evaluate the  
118 improvement in quality of consensus sequences and the ability to resolve intra-OTU haplotype  
119 variation. The denoiser resulting from this work, *debar* (DENOISING BARcodes), is a free,  
120 publicly available package written in R that is available through CRAN ([https://CRAN.R-](https://CRAN.R-project.org/package=debar)  
121 [project.org/package=debar](https://CRAN.R-project.org/package=debar)) and GitHub (<https://github.com/CNuge/debar>).

122

## 123 **Materials and Methods**

124

### 125 **Implementation**

126 The debar utility includes several customizable steps which denoise DNA barcode and  
127 metabarcode data (Figure 1; Supplementary File 1). Corrections with debar are based upon the  
128 comparison of input sequences with a nucleotide-based profile hidden Markov model (PHMM)  
129 (model training detailed in Nugent *et al.* 2020) using the Viterbi algorithm (Durbin *et al.* 1998).  
130 Briefly, debar's PHMM was trained using a curated set of 11,387 COI-5P barcode sequences  
131 obtained from the Barcode of Life Data Systems (BOLD: [www.boldsystems.org](http://www.boldsystems.org)) public database  
132 that were checked to ensure: (i) the sequence was >600 bp in length, (ii) taxonomy was known to  
133 a genus level, (iii) there were no missing base pairs, (iv) the amino acid sequence did not contain  
134 stop codons, and (v) BOLD's internal check for contaminants was negative (Nugent *et al.* 2020).  
135 The Viterbi path produced through alignment of the sequence to the PHMMs is used to match  
136 the input sequence to the PHMM (by finding the first set of 10 consecutive match states which  
137 indicate the absence of indels for the given 10 base pairs). The read is then adjusted to account  
138 for detected insertions or deletions (Figure 1). Three consecutive nucleotide insertions or  
139 deletions are permitted (not adjusted) as sequences of this kind are more likely to reflect true  
140 biological variants than technical errors (they do not result in reading frame shifts and may  
141 reflect an insertion or deletion of an amino acid in a functional protein-coding gene). The  
142 probability of such changes through sequencing error is relatively low (i.e. for the Pacific  
143 Biosciences Sequel platform the baseline probability of three consecutive deletions would be  
144 0.05% (baseline delete probability) cubed, or 0.000125%).

145 The denoising of sequences with debar is controlled using a suite of parameters (Figure  
146 1). The censorship parameter is most important as it controls the size of the masks (substitution  
147 of nucleotides for placeholder N characters) applied around sequence adjustments. This option is  
148 designed to prevent the introduction of errors that would be caused if the denoising process

149 deleted the wrong base pair or inserted a placeholder in the incorrect position. Derivation of the  
150 default value for the censorship parameter is detailed in the Methods and Results sections. The  
151 package also enables the translation of denoised sequences to amino acids to confirm that  
152 denoised outputs conform to the expected properties of the protein-coding gene region. Because  
153 *debar* can interface directly with fasta and fastq files, it enables file-to-file denoising in addition  
154 to denoising within an R programming environment. The default PHMM used for denoising by  
155 *debar* represents the complete 657bp barcode region of COI. The package also permits the use of  
156 customized PHMMs provided by a user, which allows the denoiser to be applied to data from  
157 other gene regions or for the denoiser to be targeted to a specific user-defined subsection of the  
158 COI barcode. Training of a PHMM for a new barcode or gene is supported by the R package  
159 *aphid* (Wilkinson 2019), while sub-setting of *debar*'s default PHMM is enabled by the R package  
160 *coil* (Nugent et al. 2020). Details of the package's components together with a demonstration of  
161 its implementation is available in the package's vignette (Supplementary File 1).

162

## 163 **Quantification of package performance**

### 164 **Simulated error data**

165 The *debar* package was tested using a phylogenetically stratified random sample of publicly  
166 available COI-5P sequences with artificially introduced indels. This test was designed to assess  
167 the accuracy of sequence corrections and to obtain a quantitatively informed set of default  
168 parameters for the denoising process. A random sample of 10,000 animal COI-5P sequences  
169 (excluding those used in PHMM model training) were obtained from BOLD and cleaned using  
170 the steps described in Nugent *et al.* 2020 (methods section – BOLD data acquisition). Errors  
171 were introduced into each sequence in accordance with the statistical error profile of the Pacific



172 Biosciences Sequel based upon the error profile for COI barcode region in Hebert *et al.* (2018).  
173 This profile indicated a baseline indel rate of 0.1% (insertions and deletions equally likely), a  
174 baseline substitution rate of 0.5%, and an elevated indel rate for long homopolymers (repeat  
175 length of 6,7, and 8+ with indel probabilities of 0.75%, 1.2%, and 3.8%, respectively) (Hebert *et*  
176 *al.* 2018). The location of all errors was recorded so that accuracy of subsequent corrections  
177 could be evaluated. Sequences were iteratively processed, and errors were limited to a single  
178 insertion or deletion error of one base pair in length (with the error introduction process being  
179 repeated for the original sequence when more than one indel occurred), which allowed for the  
180 accuracy of corrections to be assessed without the need to consider interaction effects.

181         The resultant sequences, each with one indel, were then denoised with debar ('denoise'  
182 function, using the parameter `sensor_length = 0`). The outputs of the denoise function were  
183 queried to determine the number and location of indel corrections applied by debar. This  
184 information was compared to the recorded ground truth error locations to quantify the following:  
185 1) the frequency with which debar located and exactly corrected indels, 2) the miss distance  
186 (number of nucleotide positions) between introduced errors and corrections applied in instances  
187 where debar did not correct the indel errors in exactly the correct position, and 3) the frequency  
188 at which debar applied an incorrect number of sequence corrections (i.e. 0 correction or 2+  
189 corrections). If one correction was made and the distance between the correction and true indel  
190 position was 0, then the correction was considered accurate. Corrections were also considered  
191 accurate if all base pairs between the correction location and the true indel position were the  
192 same (*i.e.* if base pair 2 in the homopolymer "TTTTT" was an insertion, but the 5<sup>th</sup> T in the  
193 sequence was removed by debar, this is functionally an exact correction as the true sequence is  
194 restored). All other corrections at inexact positions were considered inaccurate, and the distance

195 (number of positions) between the correction and true indel location was recorded. The mean and  
196 standard deviation of the miss distance were determined and used to select the default  
197 `sensor_length` parameter for the `debar` package, equal to the mean miss distance plus 2 standard  
198 deviations ( $\text{sensor\_length} = \text{ceiling}(\mu_{\text{miss\_distance}} + (2 \times \sigma_{\text{miss\_distance}}))$ ). This value was selected as  
199 it would be expected to avoid the introduction of an error for > 95% of inexact corrections.  
200 Sequences where no corrections or multiple corrections were made had their outputs inspected  
201 further to determine if other parts of the denoising pipeline (e.g. the check for stop codons in the  
202 translated amino acid sequence or trimming of sequence edges in the framing process) removed  
203 the error or led to the complete rejection of the sequence.

204

#### 205 **False correction rate**

206 The performance of `debar` on sequences with no indel errors was also quantified to determine the  
207 frequency and cause of erroneous corrections applied to cleaned, publicly available COI-5P  
208 barcode sequences with no known technical errors. A random sample of 10,000 sequences from  
209 all the animal COI-5P barcode sequences available on BOLD was obtained (Supplementary File  
210 2) meeting the following criteria was obtained: 1) the barcode was publicly available on the  
211 BOLD database, 2) the barcode was > 600bp in length, 3) the barcode did not contain missing  
212 characters (“N”) in the Folmer region, 4) the corresponding amino sequence did not contain stop  
213 codons, 5) the result of BOLD’s internal check for contaminants was negative, and 6) the  
214 sequence was not used in PHMM training and the simulated error dataset. Sequences were  
215 processed using `debar`’s `denoise` function (`sensor_length = 0`). All sequences that had corrections  
216 applied, or that were flagged for rejection, were counted and examined in detail to search for  
217 evidence of the proximal cause of the false correction. To search for evidence of taxonomic bias,

218 the taxonomy associated with all falsely corrected sequences were tallied at the order level, and  
219 manually examined for evidence of bias.

220

### 221 **Denoising PacBio Sequel data**

222 We quantified the performance of debar on raw DNA barcode sequence data by interfacing with  
223 the existing mBRAVE workflow (<http://www.mbrave.net>) used to process DNA barcode circular  
224 consensus sequences (CCS) obtained with the Sequel platform. A custom analysis pipeline  
225 (Supplementary File 3) was constructed to analyze and denoise the final set of CCS barcodes  
226 produced by the mBRAVE workflow (one CCS per OTU) (Figure 2). The pipeline was designed  
227 to search the final barcodes produced by mBRAVE for evidence of indel errors (by considering  
228 the translated amino acid sequence with the R package coil (Nugent *et al.* 2020)), denoise all the  
229 associated CCS with detected errors using the debar package, and then regenerate a consensus  
230 barcode sequence using the denoised data to produce a final, denoised barcode sequence for each  
231 specimen (Figure 2).

232 The outputs of this analysis were examined to determine if the debar pipeline decreased  
233 the number of technical errors in the barcode sequences and that those barcode sequences  
234 resulted in likely amino acid sequences when translated. Initial quantification of the  
235 improvement was conducted by comparing the number of barcode sequences whose amino acid  
236 sequences were flagged by the R package coil (Nugent *et al.* 2020, default parameters) before  
237 and after denoising. Barcodes are flagged by coil when they possess a stop codon when  
238 translated to amino acids or when the resultant amino acid sequence is improbable, both  
239 indicating that the sequence likely possesses an indel error.

240           Since the coil and debar packages both employ the same nucleotide profile hidden  
241 Markov model (coil also utilizes an amino acid PHMM), an independent test of pipeline  
242 effectiveness was also conducted. The effectiveness of the denoising pipeline was quantified by  
243 submitting both the original and denoised barcode sequences to BOLD. It was used to determine  
244 the number of original barcodes and denoised barcodes with evidence of stop codons after  
245 aligning the sequences using the BOLD's hidden Markov model (a model developed  
246 independently of the debar PHMM) and translating the sequence using the appropriate  
247 translation table corresponding to the taxonomic information accompanying the sequence record.  
248 Comparison of these numbers made it possible to quantify the increase in barcode-compliant  
249 sequences (i.e. those with no stop codon) produced by debar. Additionally, the Sequence Quality  
250 Report on BOLD was examined to determine the number of unknown nucleotides ("N") in the  
251 barcode sequences after denoising. The report categorizes barcode quality as: high (<1% Ns),  
252 medium (<2% Ns), low (<4% Ns), or unreliable (>4% Ns), and the number of barcodes in these  
253 different categories was recorded.

254

### 255           **Denoising metabarcode data**

256 To characterize debar's performance on metabarcode data, we analyzed a metabarcode dataset  
257 for a mock arthropod community (Braukmann *et al.* 2019). These data derived from a single  
258 sequencing run on an Ion Torrent S5 on COI amplicons generated by pooled DNA extracts from  
259 abdomens from single specimens of 369 arthropod species (methods described in detail in  
260 Braukmann *et al.* 2019). Sequences were from a 407bp fragment of the COI barcode region  
261 targeted using the primers MlepF1 and LepR1 (Hebert *et al.* 2004; Braukmann *et al.* 2019).  
262 Following amplification and sequencing on the Ion S5, quality control, sequence dereplication,

263 chimeric read filtering, matching to reference sequences, and clustering were performed on  
264 mBRAVE (Braukman *et al.* 2019). Two sets of data resulted from this process, a set of 123,926  
265 unique sequences that were assigned to 398 different Barcode Index Numbers (BINs)  
266 (Ratnasingham and Hebert 2013) through the comparison to reference sequences (matched at  
267 >98% similarity), and a set of 2,199 unique sequences not matching to available references that  
268 were clustered into an additional 1,255 OTUs at a 97% similarity threshold (using clustering  
269 algorithm described in Braukmann *et al.* 2019).

270 All sequences were denoised using debar's `denoise_list` function and a custom nucleotide  
271 PHMM. The custom PHMM was a 398bp subset of the complete COI PHMM (PHMM profile  
272 positions 250 – 648), corresponding to a segment of the Folmer (Folmer *et al.* 1994) region  
273 targeted by the metabarcoding primers. The PHMM was created using coil's 'subsetPHMM'  
274 function (Nugent *et al.* 2020). After denoising, two tests were conducted to determine if  
275 denoising improved the quality of the metabarcode pipeline's output data.

276 First, for each BIN and OTU consensus sequences were generated using denoised  
277 sequences and the debar function 'consensus\_sequence'. These consensus sequences were  
278 assessed for evidence of stop codons using coil and the same custom PHMMs used in denoising  
279 (function `coi5p_pipe` with the additional parameter: `trans_table = 5`). This test revealed the  
280 number of denoised consensus sequences which contained a stop codon when translated to  
281 amino acids, indicating an indel error persisted in the nucleotide sequence. The centroid  
282 sequences for the BINs and OTUs were used as a baseline metric for the number of barcode-  
283 compliant sequences. For each BIN, centroid sequences were obtained by clustering the  
284 sequences in the group using the R package `kmer`'s 'otu' function (parameters: `k = 4`, `threshold =`  
285 `0.95`) (Wilkinson 2018, Version 1.0.0). For the OTUs, centroids were obtained from data

286 generated by mBRAVE. All centroids were assessed with coil (Nugent *et al.* 2020, Version 1.0),  
287 and the number of barcode-compliant representative sequences for the original centroids and the  
288 final consensus sequences was compared.

289 Secondly, the individual sequences within each BIN and OTU were analyzed with coil to  
290 determine the number that were likely error free, as evidenced by the absence of stop codons  
291 after translation. This assessment was repeated on the denoised reads to determine the  
292 effectiveness of debar in correcting errors in individual sequences and to reveal if the denoising  
293 process improved the resolution of ESVs for subsequent analysis of intra-species genetic  
294 variation by placing the ESVs in reading frame and reducing the frequency of identified indel  
295 errors.

296

297

## 298 **Results**

### 299 **Quantification of package performance**

#### 300 **Simulated error data**

301 Debar was used to correct 10,000 barcodes, each with a single indel error (Supplementary File  
302 1). The denoised sequences and associated data were compared to the ground truth error  
303 locations to determine the accuracy of corrections applied by debar (Figure 3). For 9,459  
304 sequences (95.59%), a single correction was applied by debar, indicating that the package  
305 correctly identified the type of error in these sequences. However, debar either failed to  
306 recognize an indel or made too many corrections (2+) in the other 541 sequences. No correction  
307 was made for most (426) of these sequences, meaning that debar's PHMM did not identify the  
308 indel error. The overlooked indels were largely restricted to the terminal regions of the sequence;

309 75% (329/426) of them were positioned within 20 base pairs of the read termini (Figure 4),  
310 regions that only comprised 5% (40bp/650bp) of the sequences. The cause of this is that the  
311 debar denoising algorithm uses the first observation of 10 consecutive bp matching to the  
312 PHMM to establish the corrective window. Errors on the periphery of sequences therefore lead  
313 to trimming of the sequence (via the `keep_flanks` function) instead of indel correction. A  
314 substantial fraction of the remaining uncorrected indel errors (43) occurred between positions  
315 452 to 465 (Figure 4), a region associated with a 3bp indel present in some animal groups and  
316 absent in others. Its presence reduced the PHMM's indel detection ability in this region due to  
317 greater true variability. Not all unidentified indels were retained in the final output sequences as  
318 double checks of debar (employing the `keep_flanks` and `aa_check` parameters) identified many  
319 (266/426 – 62%) of the uncorrected sequences and either omit the problem region or flag the  
320 sequence as likely to contain an error. Therefore, debar's double checks allow many false  
321 negatives to be trimmed or flagged as problematic.

322 For 119 sequences (1.2%), two or more corrections were applied by debar when only a  
323 single indel existed (Figure 3). In contrast to the false negatives, debar's double checks only  
324 captured three of the false positives. Many of the false corrections appeared to be the presence of  
325 indels near codons that are not present in all animals. Due to true biological variation in the  
326 training data, these regions of the PHMM have higher probabilities of transitioning from a match  
327 state to an insert or delete state, and therefore indels in these locations are sometimes handled  
328 incorrectly (i.e. the sequence is characterized as having two deleted base pairs, when there was a  
329 1bp insertion). Because false corrections of this type result in sequences that conform to the  
330 structure of the protein-coding gene region (i.e. a lack of stop codons in the amino acid  
331 sequence), they are not identified by debar's `aa_check` function.

332 The 9,459 sequences for which the presence of a single indel was correctly identified  
333 were further analyzed to determine how accurately they were located (Figure 3). The analysis  
334 showed that debar was able to exactly locate and correct 5,847 (61.81% of sequences in single  
335 correction category) of the indel errors in the dataset. For the other 3,612 sequences (38.19% of  
336 the single corrections category), the indel corrections were not placed in exactly the correct  
337 position (Figure 5). For these sequences, the average distance between the true indel location and  
338 the applied correction was 2.31 base pairs (standard deviation = 1.9767).

339 These results were used to select a default censorship value for debar to ensure that  
340 inexactly identified indel errors are masked in most sequences (Figure 1). A default censorship  
341 length of 7 (the average miss distance plus two times the standard deviation, rounded up) was  
342 selected in order to mask the true error in >95% of instances where inexact corrections were  
343 applied, thereby successfully denoising sequences, albeit with some associated loss of  
344 information in the sequences, which can be overcome by building a consensus sequence when  
345 multiple reads are available for an individual.

346 Overall, denoising of the 10,000 barcodes with the default censorship parameter  
347 (`cancel_length = 7`) resulted in 9,309/10,000 (93.09%) of sequences with errors being  
348 successfully denoised. The additional double check parameters (`aa_check = True`, `keep_flanks =`  
349 `False`) captured, but did not correct, 269 (2.69%) errors. The debar package thereby corrected or  
350 removed 95.74% of sequences with indel errors (Figure 3).

351

### 352 **False correction rate**

353 A set of 10,000 barcode sequences with no known indel errors was analyzed with debar to  
354 determine the incidence of erroneous corrections. Nearly all sequences (99.91%) were not altered



355 nor flagged as erroneous. Nine sequences were erroneously corrected, and none were flagged for  
356 rejection. These sequences included a single sequence from each of five orders and four  
357 sequences from the order Diptera (flies). Interestingly, the four Diptera sequences that were  
358 incorrectly altered all belonged to the same genus: *Culicoides*. They represented 4/58 of all  
359 sequences from the family Ceratopogonidae that were in dataset, indicating that the performance  
360 issue was isolated to this single genus.

361         These results indicate that debar deals well with variation in COI sequences across most  
362 of the animal kingdom, but that it displays some taxonomic bias in performance. This is a  
363 limitation of debar, as any genus with a COI profile that systematically deviates from the COI  
364 PHMM used in debar will be erroneously denoised. The benefit of the conservative censorship  
365 approach used in the package is that although these reads are erroneously adjusted, the  
366 corrections made are masked by Ns, and the entire sequence is not rejected. Rather, only a small  
367 section of the sequences is lost, as if it were to contain an indel error. Most of any falsely  
368 corrected sequences can thereby be recovered, and in most instances, this would be sufficient to  
369 identify associated taxonomy and inform biological conclusions.

370

### 371         **Denoising PacBio Sequel data**

372 We applied debar in the analysis of real DNA barcode data by developing a processing pipeline  
373 (Figure 2 – hereafter ‘the debar pipeline’) and compared the amount of technical noise in the  
374 barcodes before and after processing. A set of 29,525 consensus barcode sequences derived from  
375 processing data from four Sequel runs were obtained from mBRAVE and were re-processed with  
376 the debar pipeline (Table 1).

377 Analysis of the consensus barcodes with coil (step ii. of the debar pipeline) flagged 3,495  
378 (11.8% of total) of consensus sequences due to the detection of a stop codon in the translated  
379 sequence or due to the presence of an unexpected amino acid (log likelihood score below the  
380 default threshold). The large number of flagged sequences is likely reflective of false positives  
381 (sequences flagged by coil that lack indel errors due to the incorrect establishment of reading  
382 frame). In fact, 2,418 sequences (8.1% of total, 69.2% of flagged sequences) were flagged due to  
383 the presence of a stop codon, and 1,282 of them (4.3% of total, 36.7% of flagged sequences)  
384 contained a stop codon in all three forward reading frames, providing extremely strong evidence  
385 of an indel error (i.e. a low likelihood of being a false positive).

386 After denoising, the output sequences were again assessed with coil (step viii. of the  
387 debar pipeline) and this analysis revealed that debar had corrected many indel errors (Table 1,  
388 Table 2). Only 1,123 (3.8%) of the final barcode sequences were flagged by coil's `coi5p_pipe`  
389 function, suggesting that 66.8% (2,335) of the flagged sequences were successfully denoised.  
390 When comparison was restricted to the 2,418 sequences with stop codons, only 176 were still  
391 flagged as containing stop codons, indicating that 92.7% (2,242/2,418) of the sequences in this  
392 subcategory were effectively denoised. A more conservative estimate of correction success was  
393 provided by the subset of flagged sequences with stop codons in all reading frames. Of these  
394 sequences, 1106/1282 (86.27%) passed the coil check following denoising, suggesting the  
395 successful correction of an indel error and improved representation of the true sequence.  
396 External quantification of the debar pipeline's denoising ability was obtained by the submission  
397 of pre- and post- pipeline barcode sequences to BOLD (<http://www.boldsystems.org>). The  
398 sample size for this test was smaller as BOLD requires taxonomic designations and this  
399 information was only provided by mBRAVE for 27,041 sequences. The total number of original

400 sequences flagged by BOLD due to its detection of a stop codon was 1,515 (6.3%), a  
401 considerably lower frequency than reported by coil on the initial pipeline inputs. Of the 1,515  
402 sequences with initial evidence of stop codons, 14 were rejected outright by the debar pipeline,  
403 223 were flagged but not successfully corrected, 147 were unflagged and not corrected, and  
404 1,131 had no evidence of errors following denoising (Table 3). Based on this assessment with  
405 BOLD, the debar pipeline produced a 75% reduction in the number of errors in the dataset from  
406 6.3% (1,515) to 1.6% (384). Of the remaining 384 errors, the majority (223) were detected as  
407 problematic and flagged as erroneous by debar. As a consequence, the debar pipeline reduced the  
408 number of unidentified errors by >90% (from 1,515 to 147) in the barcode dataset (Table 3).

409         The denoising of the barcodes with the debar pipeline did not result in sequences with  
410 large amounts of missing information. Of the 29,525 output barcodes, 28,802 were high quality  
411 (<1% Ns), 11 were medium quality (<2% Ns), 498 were low quality (<4% Ns), and 214 were  
412 unreliable (>4% Ns). There was a strong negative relationship between the number of CCS  
413 available for a sample and the amount of missing information in the final barcode sequence  
414 (Figure 6).

415

#### 416         **Denoising metabarcode data**

#### 417         **Consensus sequence quality**

418 Metabarcode data from a mock arthropod community were also denoised followed by  
419 comparison of original sequences to the denoised consensus sequences to determine if the debar  
420 improved sequence quality (Table 4). Of the original centroid sequences for the 398 BINs,  
421 125/398 (31.4%) contained evidence of indel errors when analyzed with coil. Following  
422 denoising and consensus sequence generation via debar, the number of barcode-compliant

423 outputs was considerably higher with only 7/394 (1.8%) displaying evidence of indel errors.  
424 Four BINs had all their component sequences rejected by debar so no consensus sequences were  
425 generated. The rate of apparent indel errors was higher in the centroids of the 1255 OTUs; 681  
426 (54%) displayed evidence of a stop codon when analyzed with coil, suggesting the presence of  
427 indels in more than half of the sequences representing each OTU. The consensus sequences  
428 produced through denoising and consensus sequence generation with debar were of apparent  
429 higher quality as only 134 (10.6%) displayed evidence of a stop codon when analyzed with coil.  
430 An additional 31 OTUs (2.5%) failed to produce a valid consensus sequence after denoising  
431 because all their component sequences were rejected by debar.

432         The corrections did cause some loss of information; 46/394 (11.7%) of the consensus  
433 sequences for the BIN groups contained at least one ‘N’ due to ambiguous or censored base pairs  
434 in their component reads, and 861/1255 (68.6%) of the OTU consensus sequences contained at  
435 least one ‘N’. The number of ‘Ns’ per sequence was generally low for the BINs (median = 0; 12  
436 sequences with 14 or more ‘Ns’) but was higher for the OTUs (median number of ‘Ns’ = 15),  
437 indicating there was on average one correction per OTU (correction of an indel, plus the seven  
438 bp mask in either direction result in 14 (insertion) or 15 (deletion) consecutive ‘Ns’). There was  
439 a positive relationship between the number of sequences within an OTU and the completeness of  
440 information in the final consensus sequence.

441

#### 442         **ESV data quality**

443         Data analysis on mBRAVE revealed 398 BINs represented by 123,926 unique  
444 dereplicated reads as well as 1255 OTUs lacking taxonomic assignment that were represented by  
445 2199 unique sequence reads. When original sequences were checked with coil, it indicated that

446 61,351/123,926 (49.5%) of BIN sequences and 1310/2199 (59.97%) of the OTU sequences  
447 displayed strong evidence of an indel error as they contained a stop codon when translated. By  
448 contrast, following denoising with debar the incidence of stop codons was far lower as just  
449 2858/122,349 (2.3%) of the BIN sequences and 418/2,145 (19.49%) of the OTU sequences had  
450 evidence of indels. This result indicated that denoising of individual sequences reduced the  
451 incidence of apparent indel errors by over 95% for the BINs (58,593 fewer indel errors) and by  
452 68% for the OTUs (892 fewer indel errors). Most sequences were subjected to at least one indel  
453 correction by debar, with 85,298/122,349 (69.7%) of the final BIN sequences and 1387/2145  
454 (64.7%) of final OTU sequences containing at least one 'N' character. Low abundance OTUs in  
455 the data set represented by biologically valid sequences need not be discarded solely due to their  
456 low abundance and could be further inspected for putative evidence of rare community members.  
457  
458  
459

## 460 **Discussion**

461  
462 This manuscript introduces debar, a PHMM-based denoiser, and demonstrates how it can  
463 improve the quality of sequence data used for both DNA barcode library construction and for  
464 metabarcode studies by correcting indels introduced by sequencing error. We first evaluated its  
465 effectiveness through an *in silico* study that tested its capacity to recognize and repair reference  
466 barcodes with artificially introduced indels. Debar was shown to be effective, as it corrected  
467 >95.7% of the errors and applied erroneous adjustments to less than 0.01% of correct sequences.  
468 This strong performance extended to real-world data sets. Debar reduced the rate of frameshift  
469 indels by 75% in sequence records generated by the long-read Sequel platform, generating more  
470 barcode-compliant sequences, most with little or no missing information. Debar also improved  
471 the quality of metabarcode data generated by the ION S5 allowing for ESVs to be considered  
472 with higher confidence and for the recovery of higher-quality representative sequences for  
473 OTUs.

474 Denoising sequences with artificial errors and known ground truths showed that the  
475 corrections performed by debar were imperfect, with the exact indel location being identified  
476 only 61.8% of the time. The application of a default 7bp censorship on both sides of putative  
477 indel corrections proved to be an effective means of masking most errors, improving the  
478 denoiser's error removal rate to >95.75%. This high error removal rate involves a tradeoff, as  
479 sequence adjustments are accompanied with a loss of 14 base pairs of information. This  
480 information loss is an acceptable cost, as it ensures that all remaining base pairs can be  
481 considered with high confidence. The nature of high-throughput sequence data, namely that there  
482 are usually multiple sequencing reads for a given specimen available, can help mitigate the loss

483 of information. Corrected sequences from a specimen or OTU can be used in conjunction with  
484 one another, filling in the different censored locations and overcoming the loss of information.  
485 The censorship of bases adjacent to indel corrections is an optional parameter that users may  
486 alter to suit their needs. Smaller censorship values, or no censorship at all, would result in less  
487 loss of information per sequence, but would come at the cost of more errors remaining in the  
488 final data.

489 Denoising of real DNA barcode data obtained from sequencing of specimens on the  
490 Pacific Biosciences Sequel platform resulted in higher-quality output sequences. An exact metric  
491 quantifying the improvement is, however, difficult to state with certainty, as the ground truth of  
492 the sequences is not known. The independent tests of the sequences through submission of  
493 consensus sequences to BOLD before and after denoising provided a conservative estimate of  
494 the debar package's effectiveness. Conservatively, this test showed a 75% reduction in the  
495 number of barcode sequences with technical indel errors after application of the debar pipeline  
496 and a low false negative rate (147 unidentified errors out of 1,515 total putative errors). This is  
497 an important improvement because the Pacific Biosciences Sequel platform is used at the Centre  
498 for Biodiversity Genomics to produce high-quality reference barcodes for the barcoding research  
499 community (Hebert *et al.* 2018). Accuracy of these sequences is therefore important; the debar  
500 package is shown to improve sequence quality, yielding more biologically likely and therefore  
501 reliable outputs. The generation of barcode sequences is also made more efficient. By increasing  
502 the rate of barcode-compliant outputs from 93.7% to 98%, fewer samples require reprocessing or  
503 resequencing.

504 Understanding within-species patterns of genetic diversity is an essential metric for  
505 characterizing community health. High intra-species genetic diversity is assumed to indicate

506 healthy ecosystems, comprised of large and stable populations with the standing genetic  
507 variation needed to survive environmental stressors (Zizka *et al.* 2020). The characterization of  
508 ESVs within OTUs can provide intra-species diversity measures for member species of a  
509 community (Frøslev *et al.* 2017). The initial check of the sub-OTU sequence data from the mock  
510 community sequenced with IonTorrent revealed a high rate of putative indel errors (54% of  
511 sequences), which would lead to a gross over estimation of the number of ESVs within the  
512 OTUs. The reduction of the error rate after denoising with debar allows for a more accurate  
513 examination of intra-OTU ESVs and therefore allows for more accurate assessments of intra-  
514 species diversity and community health, despite the fact that debar is not capable of eliminating  
515 non-indel errors from sequences. Even with the improvements to ESV quality by debar, intra-  
516 species diversity estimates will likely remain inflated to some extent, as the sequence-by-  
517 sequence corrections applied by debar exclusively account for indel errors while substitution  
518 errors could persist within the data.

519         We have demonstrated that debar is an effective means of reducing technical errors in  
520 DNA barcode and metabarcode data, but the package is not without limitations. The package is  
521 designed to correct insertion and deletion errors, but these are not the only technical issues that  
522 can lead to inflated biodiversity estimates. The program is not an effective means of identifying  
523 or correcting chimeric sequences or non-animal COI biological contaminants and should these  
524 exist within an input data set they are likely to go uncorrected. Additionally, debar does not have  
525 the ability to correct substitution errors on a sequence-by-sequence basis. Because of indel  
526 correction, denoised sequences are aligned, and nucleotide positions become directly comparable  
527 across different sequences from a given specimen or OTU. Random point substitution errors can  
528 thereby be corrected in consensus sequence generation, through the ‘majority rule’ approach



529 debar uses in base calling. However, if systematic errors exist (i.e. most sequences possess the  
530 same substitution), few sequences are available for consensus sequence generation, or ESVs are  
531 being examined, then substitution errors may persist in the data. An additional source of error  
532 unaccounted for by debar is contaminant sequences. It has been demonstrated previously that the  
533 PHMM utilized in debar is not an effective means of separating animal barcode sequences from  
534 off-target barcodes derived from bacteria, plant, fungi, or other origins (Nugent *et al.* 2020).  
535 Taken together, these limitations show that debar cannot single handedly address the technical  
536 challenges associated with DNA barcoding. The tool is likely most effective when applied in  
537 conjunction with existing barcode and metabarcode workflows and improves the quality of final  
538 sequences if the inputs have been filtered based on quality, had primers removed, and been  
539 cleaned of chimeric and contaminant sequences. The sequence-by-sequence denoising approach  
540 of debar means that it is a flexible tool capable of integrating into analysis pipelines for  
541 sequencing data from various sources. Application of debar in tandem with conventional,  
542 clustering-based denoising tools would likely lead to the highest quality assessment of  
543 biodiversity. Following OTU generation with other tools, using debar to denoise all reads within  
544 a given OTU prior to consensus sequence generation would maximize accuracy of the consensus  
545 sequence while conforming to the conserved structure of the COI barcode region. The removal  
546 of intra-OTU noise can also improve the accuracy of alpha-diversity estimates. Additionally,  
547 application of debar in the denoising of rare, low-abundance sequences not present in the OTUs  
548 would allow these data to be further examined with higher confidence, revealing biological  
549 insights that would be overlooked in conventional workflows.

550           The PHMM denoising technique used by debar is an effective barcode-focused  
551 framework that can be extended to fit a variety of needs. Data from only two sequencing

552 platforms were tested in this study: the Pacific Biosciences Sequel and Thermo IonTorrent S5.  
553 Since the PHMM used in debar is barcode specific and not sequencer specific, debar can be  
554 effectively applied in denoising of barcode data obtained from any sequencing platform.  
555 However, the effectiveness of the denoiser will depend on the types and rates of technical errors  
556 associated with a given platform. When applied to data from sequencers such as the Illumina  
557 MiSeq, the rate of technical errors corrected by debar will be lower, as this platform is more  
558 prone to introduction of substitution, as opposed to indel, errors (Schirmer *et al.* 2015). Although  
559 the debar package contains a PHMM for only the common animal barcode COI, the denoising  
560 algorithm can in the future be extended and applied in the correction of data for other DNA  
561 barcodes with conserved structures.

562

## 563 **Conclusion**

564 This study has described debar, an R package for denoising DNA barcode data, and  
565 demonstrated its ability to correct indels in both barcode and metabarcode sequences due to  
566 instrument error. In each dataset, debar improved sequence quality. It reduced the apparent  
567 number of indels by 75% in data generated by Sequel, increasing the proportion of sequences  
568 that met the quality standards required to qualify as a reference barcode. The merits of debar for  
569 metabarcode analysis were twofold, allowing more likely consensus sequences to be obtained for  
570 OTUs, and for intra-OTU variation to be quantified with higher confidence. Overall, debar is a  
571 robust utility for identifying deviations from the highly conserved protein-coding sequence of the  
572 COI barcode region. Corrections informed by its use improve the separation of true biological  
573 variation from technical noise, with low frequencies of false corrections. Integration of debar

574 into the workflows for processing barcode and metabarcode data will allow biological variation  
575 to be characterized with higher accuracy.

576 **Acknowledgements**

577

578 This research was supported by grants from Genome Canada through Ontario Genomics and  
579 from the Ontario Ministry of Economic Development, Job Creation and Trade. The funders  
580 played no role in study design or decision to publish. This research was enabled in part by  
581 resources provided by Compute Canada ([www.computecanada.ca](http://www.computecanada.ca)). We thank Tony Kuo and  
582 Thomas Braukmann for aid with data acquisition and interpretation and Tony for helpful  
583 comments on the manuscript.

584

585 **References**

586

587 Amir, A., McDonald, D., Navas-Molina, J. A., Kopylova, E., Morton, J. T., Xu, Z. Z., ... &  
588 Knight, R. (2017). Deblur rapidly resolves single-nucleotide community sequence  
589 patterns. *MSystems*, 2(2).

590

591 Baynham-Herd, Z., Amano, T., Sutherland, W. J., & Donald, P. F. (2018). Governance explains  
592 variation in national responses to the biodiversity crisis. *Environmental*  
593 *Conservation*, 45(4), 407-418.

594

595 Braukmann, T. W., Ivanova, N. V., Prosser, S. W., Elbrecht, V., Steinke, D., Ratnasingham, S., ...  
596 & Hebert, P. D. N. (2019). Metabarcoding a diverse arthropod mock  
597 community. *Molecular Ecology Resources*, 19(3), 711-727.

598

599 Brown E.A., Chain, F. J., Zhan, A., MacIsaac, H. J., & Cristescu, M. E. (2016). Early detection  
600 of aquatic invaders using metabarcoding reveals a high number of non-indigenous  
601 species in Canadian ports. *Diversity and Distributions*, 22(10), 1045-1059.

602

603 Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J. A., & Holmes, S. P.  
604 (2016). DADA2: high-resolution sample inference from Illumina amplicon data. *Nature*  
605 *Methods*, 13(7), 581.

606

607 Clare, E. L., Chain, F. J., Littlefair, J. E., & Cristescu, M. E. (2016). The effects of parameter  
608 choice on defining molecular operational taxonomic units and resulting ecological  
609 analyses of metabarcoding data. *Genome*, 59(11), 981-990.

610

611 Cordier, T., Lanzén, A., Apothéoz-Perret-Gentil, L., Stoeck, T., & Pawlowski, J. (2019).  
612 Embracing environmental genomics and machine learning for routine  
613 biomonitoring. *Trends in Microbiology*, 27(5), 387-397.

614

615 Cristescu, M. E. (2014). From barcoding single individuals to metabarcoding biological  
616 communities: towards an integrative approach to the study of global biodiversity. *Trends*  
617 *in Ecology & Evolution*, 29(10), 566-571.

618

619 Delabye, S., Rougerie, R., Bayendi, S., Andeime-Eyene, M., Zakharov, E. V., deWaard, J. R., ...  
620 & Mavoungou, J. F. (2019). Characterization and comparison of poorly known moth  
621 communities through DNA barcoding in two Afrotropical environments in  
622 Gabon. *Genome*, 62(3), 96-107.

623

624 Durbin, R., Eddy, S. R., Krogh, A., & Mitchison, G. (1998). *Biological sequence analysis:*  
625 *probabilistic models of proteins and nucleic acids*. Cambridge University Press.

626

627 Driscoll, D. A., Bland, L. M., Bryan, B. A., Newsome, T. M., Nicholson, E., Ritchie, E. G., &  
628 Doherty, T. S. (2018). A biodiversity-crisis hierarchy to evaluate and refine conservation  
629 indicators. *Nature Ecology & Evolution*, 2(5), 775-781.

630

- 631 Eddy, S. R. (1998). Profile hidden Markov models. *Bioinformatics* (Oxford, England), 14(9),  
632 755-763.  
633
- 634 Eddy, S. R. (2009). A new generation of homology search tools based on probabilistic inference.  
635 In *Genome Informatics 2009: Genome Informatics Series Vol. 23* (pp. 205-211).  
636
- 637 Edgar, R. C. (2016). UNOISE2: improved error-correction for Illumina 16S and ITS amplicon  
638 sequencing. *BioRxiv*, 081257  
639
- 640 Elbrecht, V., Vamos, E. E., Steinke, D., & Leese, F. (2018). Estimating intraspecific genetic  
641 diversity from community DNA Metabarcoding Data. *PeerJ*, 6, e4644.  
642
- 643 Folmer, O., Black M., Hoeh W., Lutz R, Vrijenhoek, R. (1994). DNA primers for amplification  
644 of mitochondrial cytochrome c oxidase subunit I from diverse metazoan  
645 invertebrates. *Mol Mar Biol Biotechnol*, 3(5), 294-9.  
646
- 647 Frøslev, T. G., Kjølner, R., Bruun, H. H., Ejrnæs, R., Brunbjerg, A. K., Pietroni, C., & Hansen, A.  
648 J. (2017). Algorithm for post-clustering curation of DNA amplicon data yields reliable  
649 biodiversity estimates. *Nature Communications*, 8(1), 1-11.  
650
- 651 Hajibabaei, M., Spall, J. L., Shokralla, S., & van Konynenburg, S. (2012). Assessing biodiversity  
652 of a freshwater benthic macroinvertebrate community through non-destructive  
653 environmental barcoding of DNA from preservative ethanol. *BMC Ecology*, 12(1), 28.  
654
- 655 Hajibabaei, M., Baird, D. J., Fahner, N. A., Beiko, R., & Golding, G. B. (2016). A new way to  
656 contemplate Darwin's tangled bank: how DNA barcodes are reconnecting biodiversity  
657 science and biomonitoring. *Philosophical Transactions of the Royal Society B: Biological  
658 Sciences*, 371(1702), 20150330.  
659
- 660 Hebert, P. D. N., Cywinska, A., Ball, S. L., & Dewaard, J. R. (2003). Biological identifications  
661 through DNA barcodes. *Proceedings of the Royal Society of London. Series B:  
662 Biological Sciences*, 270(1512), 313-321.  
663
- 664 Hebert, P. D. N., Ratnasingham, S., Zakharov, E. V., Telfer, A. C., Levesque-Beaudin, V., Milton,  
665 M. A., ... & DeWaard, J. R. (2016). Counting animal species with DNA barcodes:  
666 Canadian insects. *Philosophical Transactions of the Royal Society B: Biological  
667 Sciences*, 371(1702), 20150333.  
668
- 669 Hebert, P. D. N., Braukmann, T. W., Prosser, S. W., Ratnasingham, S., DeWaard, J. R., Ivanova,  
670 N. V., ... & Zakharov, E. V. (2018). A Sequel to Sanger: amplicon sequencing that  
671 scales. *BMC Genomics*, 19(1), 219.  
672
- 673 Hubert, N., & Hanner, R. (2015). DNA barcoding, species delineation and taxonomy: a historical  
674 perspective. *DNA Barcodes*, 3(1), 44-58.  
675

- 676 Kaunisto, K. M., Roslin, T., Sääksjärvi, I. E., & Vesterinen, E. J. (2017). Pellets of proof: First  
677 glimpse of the dietary composition of adult odonates as revealed by metabarcoding of  
678 feces. *Ecology and Evolution*, 7(20), 8588-8598.  
679
- 680 Kumar, V., Vollbrecht, T., Chernyshev, M., Mohan, S., Hanst, B., Bavafa, N., ... & Golden, M.  
681 (2019). Long-read amplicon denoising. *Nucleic Acids Research*, 47(18), e104-e104.  
682
- 683 Lopez-Vaamonde, C., Sire, L., Rasmussen, B., Rougerie, R., Wieser, C., Allaoui, A. A., ... &  
684 Lees, D. C. (2019). DNA barcodes reveal deeply neglected diversity and numerous  
685 invasions of micromoths in Madagascar. *Genome*, 62(3), 108-121.  
686
- 687 Nearing, J. T., Douglas, G. M., Comeau, A. M., & Langille, M. G. (2018). Denoising the  
688 Denoisers: an independent evaluation of microbiome sequence error-correction  
689 approaches. *PeerJ*, 6, e5364.  
690
- 691 Nugent, C. M., Elliott, T. A., Ratnasingham, S., & Adamowicz, S. J. (2020). Coil: an R package  
692 for cytochrome C oxidase I (COI) DNA barcode data cleaning, translation, and error  
693 evaluation. *Genome*. 63(6):291-305.  
694
- 695 Ratnasingham, S., & Hebert, P. D. N. (2013). A DNA-based registry for all animal species: the  
696 Barcode Index Number (BIN) system. *PloS One*, 8(7).  
697
- 698 Rosen, G., Garbarine, E., Caseiro, D., Polikar, R., & Sokhansanj, B. (2008). Metagenome  
699 Fragment Classification Using *N*-Mer Frequency Profiles. *Advances in*  
700 *Bioinformatics*, 2008.  
701
- 702 Schirmer, M., Ijaz, U. Z., D'Amore, R., Hall, N., Sloan, W. T., & Quince, C. (2015). Insight into  
703 biases and sequencing errors for amplicon sequencing with the Illumina MiSeq  
704 platform. *Nucleic Acids Research*, 43(6), e37-e37.  
705
- 706 Sogin, M. L., Morrison, H. G., Huber, J. A., Welch, D. M., Huse, S. M., Neal, P. R., ... & Herndl,  
707 G. J. (2006). Microbial diversity in the deep sea and the underexplored "rare  
708 biosphere". *Proceedings of the National Academy of Sciences*, 103(32), 12115-12120.  
709
- 710 Stat, M., Huggett, M. J., Bernasconi, R., DiBattista, J. D., Berry, T. E., Newman, S. J., ... &  
711 Bunce, M. (2017). Ecosystem biomonitoring with eDNA: metabarcoding across the tree  
712 of life in a tropical marine environment. *Scientific Reports*, 7(1), 1-11.  
713
- 714 Taberlet, P., Coissac, E., Hajibabaei, M., & Rieseberg, L. H. (2012). Environmental  
715 DNA. *Molecular Ecology*, 21(8), 1789-1793.  
716
- 717 Wilkinson SP. (2018) kmer: an R package for fast alignment-free clustering of biological  
718 sequences. R package version 1.0.0. <https://cran.r-project.org/package=kmer>  
719
- 720 Wilkinson, S. P. (2019). aphid: an R package for analysis with profile hidden Markov  
721 models. *Bioinformatics*, 35(19), 3829-3830.

722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735

Wilson, J. J., Brandon-Mong, G. J., Gan, H. M., & Sing, K. W. (2019). High-throughput terrestrial biodiversity assessments: mitochondrial metabarcoding, metagenomics or metatranscriptomics?. *Mitochondrial DNA Part A*, 30(1), 60-67.

Wirta, H. K., Hebert, P. D. N., Kaartinen, R., Prosser, S. W., Várkonyi, G., & Roslin, T. (2014). Complementary molecular information changes our perception of food web structure. *Proceedings of the National Academy of Sciences*, 111(5), 1885-1890.

Zizka, V. M., Weiss, M., & Leese, F. (2020). Can metabarcoding resolve intraspecific genetic diversity changes to environmental stressors? A test case using river macrozoobenthos. *BioRxiv*.



736

737 **Data Accessibility Statement**

738

739 DNA barcode sequences used in training of the Profile Hidden Markov Models are available in  
740 the Supplementary data of the following paper: <https://doi.org/10.1139/gen-2019-0206>. DNA  
741 barcode sequences used in model testing are available in this manuscript's Supplementary files.

742 The R source code for the debar package is available on GitHub:

743 <https://github.com/CNuge/debar>. Additional data and code available on request from the authors.

744

745

746 **Author Contributions**

747

748 The study was conceived and designed by SJA, PDNH, SR, and CMN. The programming of the  
749 debar package was performed by CMN. Analyses of package performance were performed by  
750 CMN with resources, design, and other assistance provided by TAE, SR, and SJA. The initial  
751 draft of the manuscript was written by CMN and SJA. All authors contributed to the editing of  
752 the manuscript.

753

754 **Tables and Figures**

755

756 **Table 1.** Summary of the results for the 29,525 barcode sequences (produced from PacBio  
757 Sequel data analyzed using the mBRAVE platform) after processing with the debar pipeline.  
758

<b>PacBio Sequel run</b>	<b>Run 1</b>	<b>Run 2</b>	<b>Run 3</b>	<b>Run 4</b>	<b>Total</b>
<b>Consensus sequences generated</b>	7,518	7,373	7,235	7,399	29,525
<b>Consensus sequences flagged by coil for indel error</b>	869	817	900	909	3,495 (11.8%)
<b>Rejected by debar denoising</b>	8	4	16	9	37 (0.1%)
<b>Sequences flagged by coil post-denoising</b>	256	285	305	277	1,123 (3.8%)
<b>Sequences corrected</b>	605	528	579	623	2,335 (66.8% of flagged sequences)

759

760

761 **Table 2.** Assessment of the correction ability of the debar pipeline for the subset of sequences in  
 762 the high-confidence error set. This set of sequences was flagged by coil and produced a stop  
 763 codon when translated within all reading frames. The top half of the table indicates the number  
 764 of sequences flagged by coil as likely to be erroneous, based on the log likelihood values of the  
 765 sequences. Results are shown for sequences both before and after the denoising process. The  
 766 bottom half of the table contains the number of sequences flagged by coil as likely to be  
 767 erroneous, based on the presence of a stop codon in the amino acid sequence resulting from the  
 768 censored translation of the framed nucleotide sequence. This high success for the stop-codon  
 769 metric (86.3% of errors removed) indicates that the pipeline is an effective means of correcting  
 770 frameshift-causing insertion or deletion errors. The relatively lower success at correcting  
 771 sequences with low log likelihood values suggests that frameshift-causing errors are not the only  
 772 set of errors being flagged by coil, and that non-frameshift errors are not effectively corrected by  
 773 the debar pipeline.  
 774

<b>PacBio Sequel run</b>	<b>Run 1</b>	<b>Run 2</b>	<b>Run 3</b>	<b>Run 4</b>	<b>Total</b>
<b>Original flagged</b>	551	547	609	610	2,317
<b>Flagged post-denoising</b>	254	280	300	271	1,105
<b>Corrected</b>	53.9%	48.8%	50.7%	55.6%	52.3%
<b>PacBio Sequel run</b>	<b>Run 1</b>	<b>Run 2</b>	<b>Run 3</b>	<b>Run 4</b>	<b>Total</b>
<b>Original stop codon</b>	319	295	318	350	1,212
<b>Stop codon post-denoising</b>	43	42	36	55	176
<b>Corrected stop codons</b>	86.5%	85.7%	88.7%	84.2%	86.3%

775  
 776  
 777  
 778  
 779  
 780  
 781

782 **Table 3.** Result of the BOLD Data System evaluation of debar denoising workflow's  
 783 effectiveness. The number of sequences identified by BOLD as containing stop codons, before  
 784 and after processing with the denoising pipeline (Figure 2). Only the 27,041 specimens with  
 785 barcodes and taxonomic information produced through the processing of PacBio Sequel data on  
 786 the MBRAVE platform were considered, as BOLD requires taxonomic information for assessing  
 787 the presence of stop codons. The rows break the sequences down into categories, which indicate  
 788 the source of the post-denoising sequence that was submitted to BOLD for assessment.  
 789

Sequence category	Total Sequence count	Stop codon count		Percent error reduction
		Original	Post-denoising	
Unaltered	23,992	88	88 <sup>†</sup>	-
Denoised, altered	2,265	1,190	59 <sup>†</sup>	95%
Flagged for potential error, unaltered	701	223	223*	-
Flagged and rejected	16	14	14	-
Labelled as <i>Wolbachia</i> by MBRAVE	67	0	0	-
<b>Total</b>	27,041	1,515 (6.3%)	384 (1.6%)	74.66%
<b>Total, non-flagged only</b>	26,257	1,278 (4.8%)	147 (0.6%)	88.5%

790 † The sum of these categories (shown in the final row of the column) represents the false  
 791 negative rate for the denoising pipeline. These are the 0.6% (147/27,041) of sequences that  
 792 appear to contain true stop codons that were not flagged for denoising, or that were denoised  
 793 unsuccessfully and not flagged as potential errors.

794 \* The false positive rate of the denoising pipeline is the number of sequences in this category  
 795 that do not in fact contain a stop codon. There is a total of 478 (701-223) false positives and an  
 796 overall false positive rate of 1.8% (478/27,041). Since this set of sequences are flagged for  
 797 potential errors, as opposed to being outright rejected, additional inspection of sequences in this  
 798 category can separate the unsuccessfully denoised sequences with true errors from those that do  
 799 not contain an error.

800

801 **Table 4.** Assessment of the sequence quality for data from a mock community of arthropods  
 802 sequenced in bulk using a Thermo Fisher Ion Torrent and processed on the mBRAVE platform.  
 803 Sequencing and processing results in two sets of data, groups of sequences assigned to BINs and  
 804 groups of sequences clustered into OTUs. The representative sequences (centroids before  
 805 denoising, consensus after denoising) and all individual sequences were checked with the R  
 806 package coil for evidence of frameshifts (stop codons in amino acid sequence) before and after  
 807 denoising to see if processing the data with the debar package resulted in higher quality barcode  
 808 sequences.  
 809

Sequences analyzed	Sequence data source	Original		After debar denoising	
		Total count	Stop codon count	Total count	Stop codon count
Representative sequences	Assigned to BINs	398	125 (31.4%)	394	7 (1.8%)
	OTUs	1,255	681 (54%)	1,224	134 (10.6%)
ESVs	Assigned to BINs	123,926	61351 (49.5%)	122,349	2858 (2.3%)
	OTUs	2,199	1310 (59.57%)	2,145	418 (19.49%)

810  
 811  
 812  
 813  
 814  
 815

## A – Input

```
S1 : CTATACTTAATCTTTGGCGCATGAGCTGGT...GGAGGAGACCCAGTTTTATACCAACACCTT
S2 : ACCCTATACTTTATTTTGAATTTGATCA...GGAGGGGACCCTATTTTATACCAACATTTA
S3 : GGGTACTCTGTACCTAATCTTCGGAGCATGAGCC...GGAGGAGAACCCAGTACTATACCAACACCTACGCG
```

## B – PHMM compare, frame

```
000111111111111111111111111111111111...11111111111111111111111111
S1 : ---CTATACTTAATCTTTGGCGCATGAGCTGGT...GGAGGAGACCCAGTTTTATACCAACACCTT

111111111111111101111111111111...111111111111111111111111111111
S2 : ACCCTATACTTTATTTTGAATTTGATCA...GGAGGGGACCCTATTTTATACCAACATTTA

111111111111111111111111111111...111111112111111111111111111111
S3 : ACTCTGTACCTAATCTTCGGAGCATGAGCC...GGAGGAGAACCCAGTACTATACCAACACCTA
```

## C – Adjust sequence, censorship

```
000111111111111111111111111111111111...11111111111111111111111111
S1 : ---CTATACTTAATCTTTGGCGCATGAGCTGGT...GGAGGAGACCCAGTTTTATACCAACACCTT

11111111111111110 111111111111...111111111111111111111111111111
S2 : ACCCTATACNNNNNNNNNNNNNNNGATCA...GGAGGGGACCCTATTTTATACCAACATTTA

111111111111111111111111111111...111111112111111111111111111111
S3 : ACTCTGTACCTAATCTTCGGAGCATGAGCC...GNNNNNNN NNNNNNNCTATACCAACACCTA
```

816

## D – Output sequences

```
S1 : ---CTATACTTAATCTTTGGCGCATGAGCTGGT...GGAGGAGACCCAGTTTTATACCAACACCTT
S2 : ACCCTATACNNNNNNNNNNNNNNNGATCA...GGAGGGGACCCTATTTTATACCAACATTTA
S3 : GGGTACTCTGTACCTAATCTTCGGAGCATGAGCC...GNNNNNNN NNNNNNNCTATACCAACACCTACGCG
```

## E – Consensus of adjusted sequences

```
S2A : ACCCTATACNNNNNNNNNNNNNNNGATCA...GGAGGGGACCCTATTTTATACCAACATTTA
S2B : ACCCTATACTTTATTTTGAATTTGATCA...NNNNNNGACCCTATTTTATACCAACATTTA
S2C : ACCCTATACTTTATTTTGAATTTGATCA...NNNNNNNNNNNNNTTTTATACCAACATTTA
S2FINAL : ACCCTATACTTTATTTTGAATTTGATCA...GGAGGGGACCCTATTTTATACCAACATTTA
```

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

**Figure 1.** Diagram demonstrating the debar package’s denoising workflow. Blue indicates nucleotides that are part of the barcode region and orange nucleotides in bold font indicate technical errors or sequence from outside of the barcode region.

**A.** The debar package operates on a sequence-by-sequence basis, taking each input and constructing a custom DNaseq object. A DNaseq object can receive a DNA sequence, an identifier, and optionally a sequence of corresponding PHRED quality scores. Although not utilized in the denoising, indel-correcting adjustments to the sequence are applied to the PHRED scores as well, so that quality information can be carried from input to output.

**B.** Following DNaseq object construction, the sequence is compared to the PHMM using the Viterbi algorithm. By default, the full length (657bp) COI-5P PHMM contained in debar is used to evaluate the sequence. When required, a user may pass a custom PHMM corresponding to a subsection of the COI-5P region (specified using the coil package’s subsetPHMM function) or a custom PHMM trained on user-defined data (Wilkinson 2019). The frame function isolates the correction window, which is the section of the sequence matching the PHMM (the first 10 consecutive base pairs matching to the PHMM on the leading and trailing edges of the sequence establish the section of the input on which subsequent corrections are applied).

834 **C.** The adjust function traverses the section of the sequence and Viterbi path defined by the  
835 frame function. When evidence of an inserted base pair ('2' label in the Viterbi path) is  
836 encountered, the corresponding base pair is removed. When evidence of a deleted base pair is  
837 encountered (a '0' label in the Viterbi path) a placeholder 'N' nucleotide is inserted. Exceptions  
838 are made for triple inserts or triple deletes (three consecutive '0' or '2' labels), which are skipped  
839 by the adjustment algorithm, as they are indicative of mutations that would not have a large  
840 impact on the structure of the protein-coding gene region and could reflect biological amino acid  
841 indels. The total number of adjustments made by debar is limited by the parameter 'adjust\_limit'  
842 (default = 5), sequences requiring adjustments in excess of this number are flagged for rejection,  
843 as this high frequency of indels is likely not the result of technical error, but rather other sources  
844 of noise such as pseudogenes. Following adjustment, a mask of placeholder 'N' nucleotides is  
845 applied to base pairs flanking the corrected indel (default is 7bp in each direction, see Figure 3.  
846 For derivation of default). Masking of 7bp flanks adjacent to each correction allows imprecise  
847 corrections to effectively correct sequence length and also mask true indel locations in the  
848 majority of instances.

849 **D.** Following adjustment, the denoised sequences are output by debar. By default, the outputs  
850 will include trailing sequence outside of the correction window. Leading information outside of  
851 the correction window is dropped, so that sequences are aligned with a common starting position.  
852 A user can choose to keep only the correction window, or have both flanking regions appended  
853 back on to the sequence output.

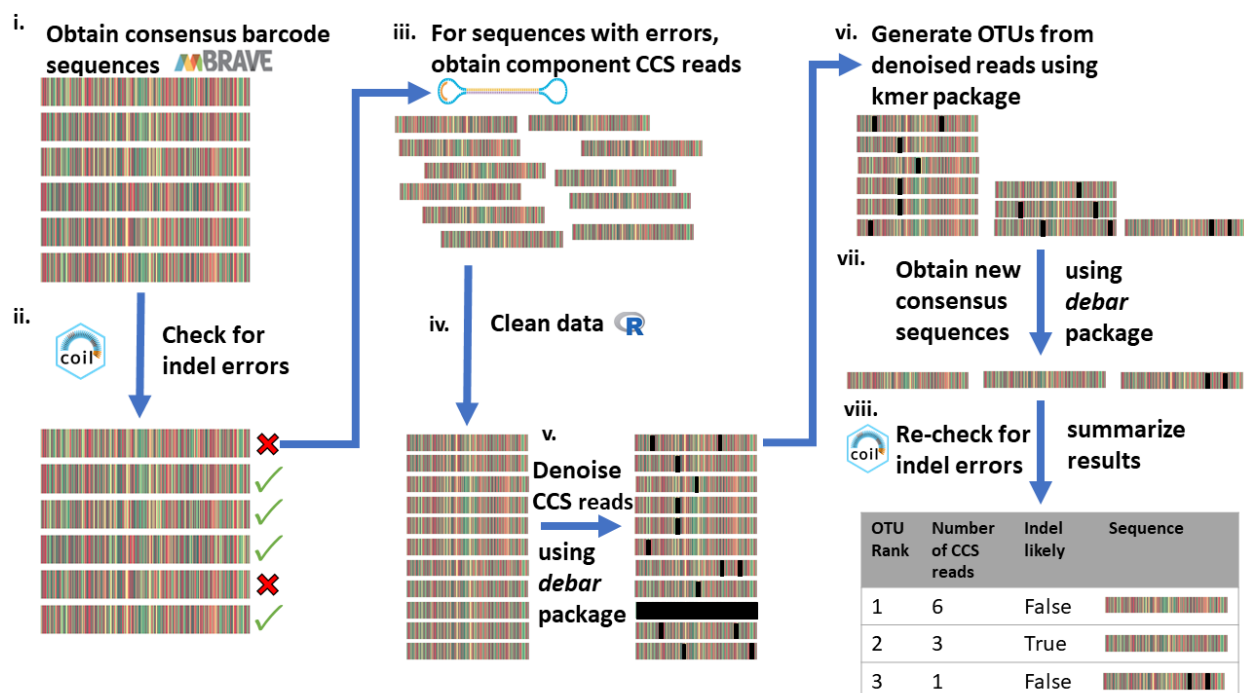
854 **E.** If multiple denoised sequences are available (for either a given specimen in the case of  
855 barcoding or a given OTU in metabarcoding) then the consensus of the denoised sequences can  
856 be taken. The consensus function assumes the sequences have been denoised and their left flanks  
857 removed; as a result, they are aligned to one another. The modal base pair for each position is  
858 then taken to generate a consensus sequence, and in the case of ties, a placeholder "N" character  
859 is added to the consensus.

860

861

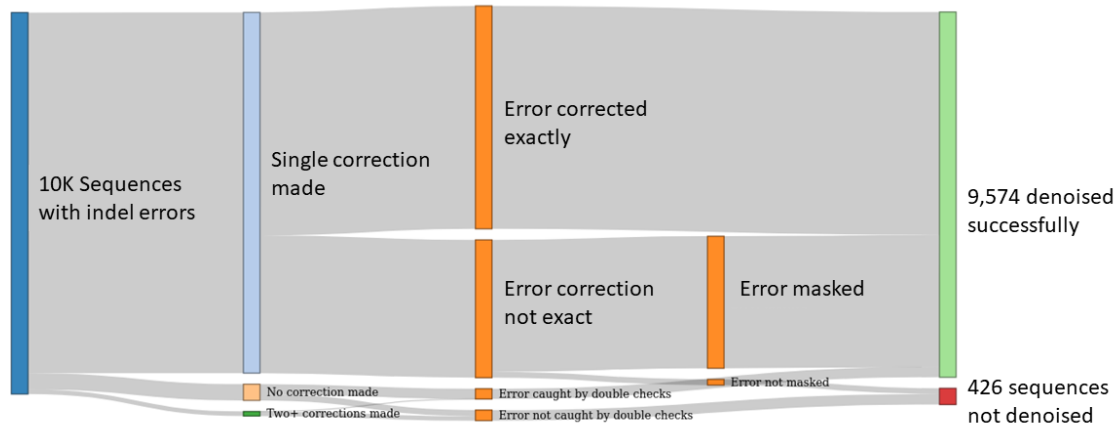
862

863



864  
 865 **Figure 2.** Diagram of the denoising workflow used to improve the quality of barcodes produced  
 866 by processing Pacific Biosciences Sequel circular consensus data on the mBRAVE platform. **(i)**  
 867 Pacific Biosciences Sequel data are processed on the mBRAVE platform, and an initial set of  
 868 barcode sequences is produced. **(ii)** The set of consensus barcode sequences produced by the  
 869 mBRAVE platform are obtained and analyzed with the coil package, using the ‘coi5p\_pipe’  
 870 function (default parameters). Sequences displaying evidence of an indel (either the presence of a  
 871 stop codon when translated to amino acids or an amino acid sequence with a low likelihood  
 872 score) are retained for further denoising. **(iii)** For each barcode with evidence of an error, all  
 873 component CCS reads (and associated metadata) derived from the given specimen are obtained  
 874 from mBRAVE. **(iv)** Based on the mBRAVE metadata, sequences are trimmed to remove  
 875 primers, MID tags, and adapter sequence. The reverse complement of reads are taken when  
 876 required. **(v)** The ‘denoise\_list’ function of debar is used to denoise all CCS reads (options:  
 877 dir\_check = FALSE, keep\_flanks = ‘right’, censor\_length = 7). Rejected reads (those flagged by  
 878 the denoise\_list function) are removed from the dataset. **(vi)** For each specimen, the reads are  
 879 clustered into OTUs using the R package kmer (clustering threshold = 0.975). This is done to  
 880 mitigate the influence of outlier CCS or contaminant sequences. **(vii)** For each OTU, a consensus  
 881 sequence is generated using debar’s ‘consensus’ function. For each specimen, OTUs are ranked  
 882 based on the number of component CCS reads they contain. **(viii)** The consensus sequences are  
 883 reassessed with coil. If the top-ranked consensus sequence now passes the coil check, it is  
 884 deemed to have been successfully denoised, and it is selected as the output barcode. If not, the  
 885 check is repeated for the second-ranked consensus sequence (when available), and this output is  
 886 retained if it is barcode compliant. If neither the first nor second highest ranked consensus  
 887 sequence passes the coil check, then the original (pre-denoising process) barcode is retained, as  
 888 no meaningful improvement was made. In this situation the barcode is flagged as likely to  
 889 contain an error.

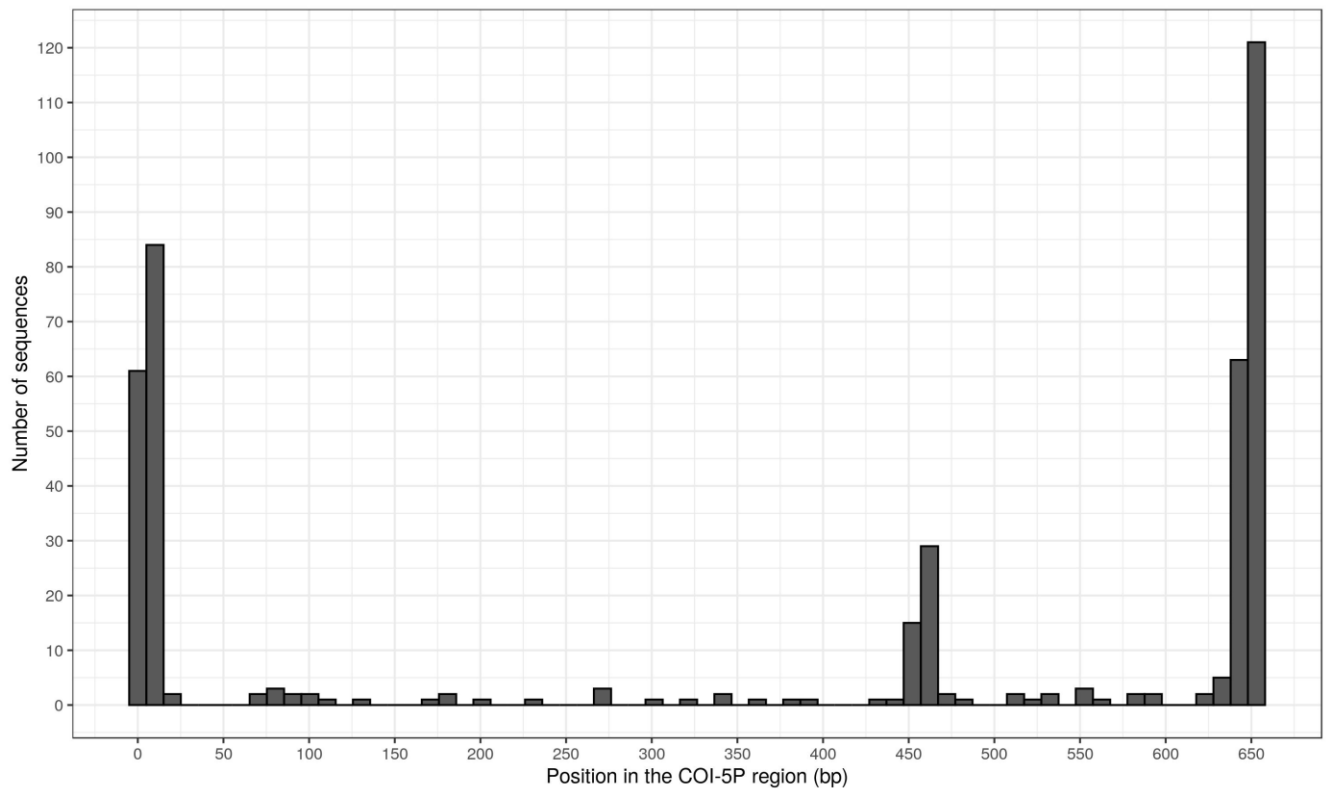




890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903

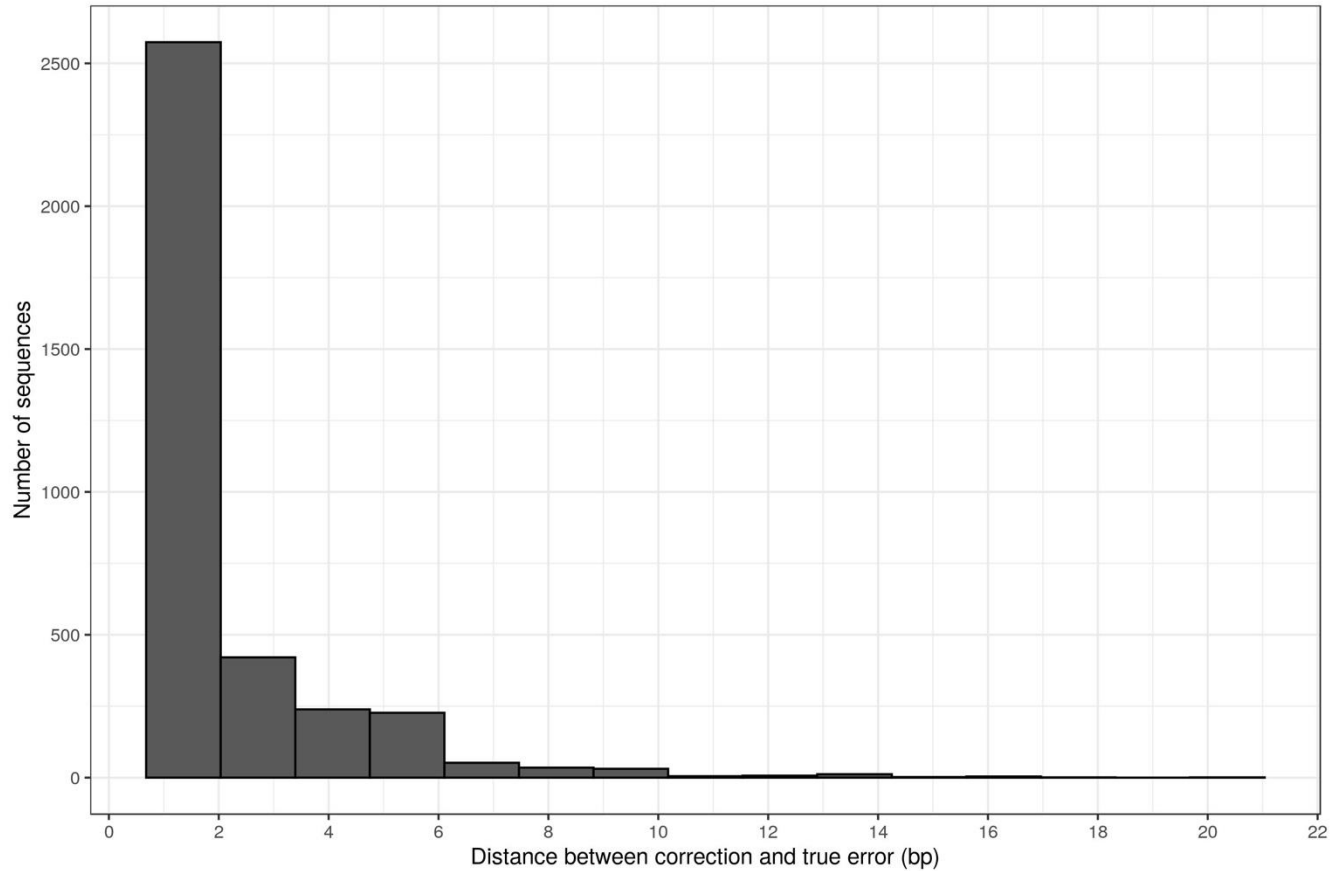
**Figure 3.** The debar package’s denoising of 10,000 COI sequences containing single insertion or deletion errors. So that exact error positions were known, errors were artificially introduced in accordance with known probabilities for COI DNA barcode data from the PacBio Sequel platform (Hebert *et al.* 2018). Denoising was accomplished through altering sequences in accordance with the Viterbi path yielded by comparison to the PHMM. The correct number of adjustments was made for 9,455 sequences, and 61.8% of these corrections located the indel exactly. Masking of 7bp flanks adjacent to each correction allowed imprecise corrections to correct sequence length and mask the true indel location 96% of the time. For the 545 instances where an incorrect number of adjustments were made, 269 were caught through query of the amino acid sequence for stop codons and the trimming of spurious matches at the edge of sequences. Overall, 95.74% of errors were effectively corrected or identified as erroneous.

904



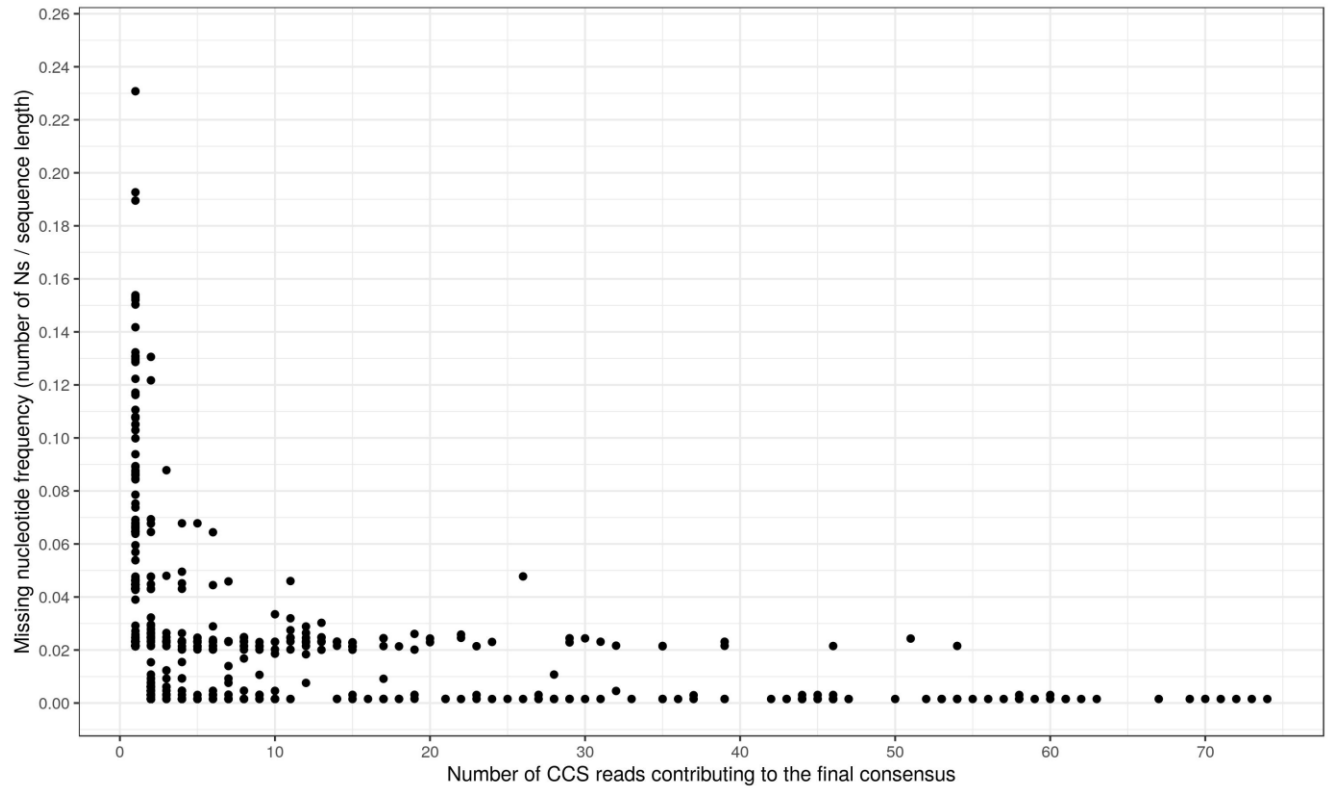
905  
906  
907  
908  
909  
910

**Figure 4.** Histogram indicating the position in the COI-5P region of the 426 uncorrected indel errors from the 10,000-sequence artificial error dataset. The x axis indicates the base pair position in the COI-5P profile, and the y axis displays the number of sequences that contained an uncorrected error at the given range of positions (bins of 10 base pair positions).



911  
912  
913  
914  
915  
916  
917

**Figure 5.** Histogram showing number of base pairs between inexact corrections applied by debar and the ground truth error location for the given sequence. In total 3,612 sequences (36.12%) had errors that were denoised inexactly, and corrections were an average of 2.31 bp (sd = 1.9767) away from the exact ground truth error location.



918  
919  
920  
921  
922  
923  
924  
925  
926  
927

**Figure 6.** Relationship between the amount of missing data in the final denoised barcode sequences (number of Ns divided by the total length of the sequence) and the number of CCS reads that contributed to the generation of the barcode. The figure displays only the 1,008 denoised barcode sequences submitted to BOLD that contained at least one “N” (the remaining 28,517 barcode sequences in the BOLD submission did not contain an “N”).

928 **Supplementary Information**

929

930 **Supplementary File 1** ('S1-single\_errors\_in\_10k\_sequences.csv') The 10,000 COI barcode  
931 sequences with single introduced indel errors that were used to test debar and calibrate the  
932 default parameters.

933

934 **Supplementary File 2** ('S2-control\_denoising\_no\_errors.csv') The 10,000 COI barcode  
935 sequences with no known indel errors used to assess the false correction rate of debar

936

937 **Supplementary File 3** ('S3-single\_file\_pipeline') Scripts and example data for the denoising  
938 pipeline developed to process COI DNA barcode sequence data produced using the Pacific  
939 BioSciences Sequel sequencer and mBRAVE platform

940

941 **Supplementary File 4** Scripts and example data for the denoising pipeline developed to process  
942 COI DNA metabarcode sequence data produced using the IonTorrent S5 sequencer and the  
943 mBRAVE platform

944

945 **Supplementary File 5** Vignette demonstrating the functionality of the debar package. The  
946 vignette is also available as part of the R package

947 (<https://github.com/CNuge/debar/tree/master/vignettes>)