MOLECULAR ECOLOGY

Molecular Ecology (2014) **23**, 1679–1700 doi: 10.1111/mec.12680

INVITED REVIEWS AND SYNTHESES

Prevention, diagnosis and treatment of high-throughput sequencing data pathologies

XIAOFAN ZHOU and ANTONIS ROKAS

Department of Biological Sciences, Vanderbilt University, Nashville, TN 37235, USA

Abstract

High-throughput sequencing (HTS) technologies generate millions of sequence reads from DNA/RNA molecules rapidly and cost-effectively, enabling single investigator laboratories to address a variety of 'omics' questions in nonmodel organisms, fundamentally changing the way genomic approaches are used to advance biological research. One major challenge posed by HTS is the complexity and difficulty of data quality control (QC). While QC issues associated with sample isolation, library preparation and sequencing are well known and protocols for their handling are widely available, the QC of the actual sequence reads generated by HTS is often overlooked. HTS-generated sequence reads can contain various errors, biases and artefacts whose identification and amelioration can greatly impact subsequent data analysis. However, a systematic survey on QC procedures for HTS data is still lacking. In this review, we begin by presenting standard 'health check-up' QC procedures recommended for HTS data sets and establishing what 'healthy' HTS data look like. We next proceed by classifying errors, biases and artefacts present in HTS data into three major types of 'pathologies', discussing their causes and symptoms and illustrating with examples their diagnosis and impact on downstream analyses. We conclude this review by offering examples of successful 'treatment' protocols and recommendations on standard practices and treatment options. Notwithstanding the speed with which HTS technologies – and consequently their pathologies – change, we argue that careful QC of HTS data is an important – yet often neglected – aspect of their application in molecular ecology, and lay the groundwork for developing a HTS data QC 'best practices' guide.

Keywords: bioinformatics, high-throughput sequencing, next-generation sequencing, preprocessing, quality control, sequence read

Received 4 November 2013; revision received 17 January 2014; accepted 22 January 2014

Introduction

The recent invention and development of several different high-throughput sequencing (HTS) technologies represents a major breakthrough in data acquisition (Shendure & Ji 2008; Mardis 2013). With its broad applicability and unprecedented data generation ability, HTS has fuelled remarkable advances in multiple areas of biology (Mardis 2008; Schuster 2008; Werner 2010; Egan *et al.* 2012). Of particular interest to molecular ecologists has been the impact of HTS on the fields of ecology and evolutionary biology. By enabling the acquisition and analysis of transcriptome and genome data in a cost-effective fashion from virtually every organism, the advent of HTS has greatly expanded the scope of these fields, fundamentally changing the ranges and types of questions that can be addressed (Hudson 2008; Rokas & Abbot 2009; Tautz *et al.* 2010; Ekblom & Galindo 2011). Judging by how quickly HTS became the standard method for collecting large-scale DNA/RNA sequence data (Shendure & Lieberman Aiden 2012), by the recent emergence of bench-top (Loman *et al.* 2012) and even travel-size personal sequencers (Eisenstein

Correspondence: Antonis Rokas, Fax: +1 615 343 6707;
E-mail: antonis.rokas@vanderbilt.edu

2012), as well as by the increase in HTS capacity and flexibility (Glenn 2011), HTS will soon become, if not already, a standard technology for molecular ecology laboratories.

The novelty and enormity of HTS-generated sequence data, however, pose several novel and significant challenges that can greatly impact downstream data analyses. Currently, these challenges are augmented by the fact that an increasing number of HTS data analyses are performed by newcomers to the field, typically experimental biologists well trained in the wet laboratory techniques of the type used to generate HTS data but who usually lack the computational training required for HTS sequence data analysis.

Recently, many efforts have been launched to address these challenges. Notable examples include the development of integrative platforms, such as GALAXY (Goecks *et al.* 2010), which provide users convenient and intuitive web interfaces for accessing software tools that perform basic sequence read quality assessment and filtering, as well as numerous downstream analyses (e.g. the ability to map sequence reads against reference genomes). These platforms streamline users' ability to perform, share and replicate certain types of analyses on HTS data, in effect substantially reducing the burden of acquiring computational skills without, however, reducing the quality of the analysis output. Furthermore, a number of recent tutorials and reviews provide expert guidance to several major steps of HTS studies from platform choice and sample preparation to various downstream analyses (Oshlack *et al.* 2010; Glenn 2011; De Wit *et al.* 2012; Wolf 2013).

As data amount continues to increase, one critical part of HTS studies that is becoming increasingly important as well as increasingly complex is the quality control (QC) of the actual sequence read data. This aspect of QC is frequently overlooked (descriptions of QC protocols dealing with sequence read data are often absent from or barely mentioned in HTS studies, especially those done at small scale), even though it is increasingly recognized that HTS sequence read data harbour various quality issues, such as platform-specific error profiles (Glenn 2011; Quail *et al.* 2012), uneven sequencing quality across sequence reads (Dohm *et al.* 2008; Minoche *et al.* 2011) and contaminations (e.g. sequencing adapter/primer, untargeted organisms) (Longo *et al.* 2011; Dewoody *et al.* 2013). If undetected, such artefacts may lead to inaccurate data interpretation or even false discovery. Thus, good QC of sequence read data is vital for any HTS experiment not only because the concern of 'garbage in, garbage out' also applies to HTS data but also because proper QC of sequence read data can dramatically improve the accuracy and quality of results of downstream analyses (Taub *et al.* 2010; Bokulich *et al.* 2013; MacManes & Eisen 2013).

We believe that guidance and discussion of QC of HTS sequence read data is an urgent need faced by the HTS user community, which is evidenced by the frequent questions on this topic on forums such as SEQanswers (Li *et al.* 2012) and BioStar (Parnell *et al.* 2011). Although these online threads discussing individual QC-related topics are very useful, the lack of systematic surveys that examine the rationale and procedures of HTS data QC as well as the many available bioinformatics tools adds to the challenge that users face when analysing HTS sequence read data.

In this review, we present a comprehensive introduction to the diverse 'pathologies' (i.e. types of QC issues) frequently encountered in HTS sequence read data. Because QC during the library preparation step has been well covered (Gayral *et al.* 2011; De Wit *et al.* 2012; Wolf 2013), we focus exclusively on artefacts that affect HTS sequence read data and that can be detected and corrected after HTS data generation. We use examples from our own research as well as the literature to illustrate the diagnosis and treatment of various HTS sequence data pathologies and the often considerable positive impact that the addition of QC steps has on data and downstream analysis quality. Finally, we evaluate the performance of sets of QC tools with similar functions, opening the path towards the development of a set of best practices for HTS data QC. We centre our discussion on data generated by the Illumina technology due to its dominance in the HTS market and literature, yet QC tools designed for other HTS technologies are also discussed when applicable.

'Health check-up': general QC assessment measures for HTS sequence read data

Just like a complete health check-up is the first step in combating real diseases, the journey of HTS sequence read data QC starts with an overall assessment of its quality. Typical HTS data sets consist of millions of short sequence reads, each of which is several tens to hundreds of nucleotide bases long and is accompanied by quality values that measure the probability of incorrect base calling. To efficiently summarize sequence quality statistics from such large data sets, numerous tools have been developed (for a description of tool features and performance, see Box 1). Most of these tools have adopted a core set of QC metrics that evaluate essential aspects of HTS sequence data quality.

Box 1 Tools for assessing HTS sequence read data quality

General quality assessment of HTS sequence read data is a relatively straightforward task. The many available pieces of software developed for this purpose mainly differ in their functionality and efficiency. We compared the features of nine stand-alone tools, which are summarized in Table 1. While many tools can assess the distribution of quality, composition and sequence read length, FASTQC (http://www.bioinformatics.babraham.ac.uk/projects/fastqc) is the most versatile; it supports all HTS technologies, is compatible with compressed input files and provides useful information on duplication level and overrepresented sequences and *k*-mers. Some other tools, such as BIGPRE (Zhang *et al.* 2011), HTQC (Yang *et al.* 2013b) and SOLEXAQA (Cox *et al.* 2010), can perform tile-based quality assessment, which helps to identify problems that affect specific regions (known as tiles) on a flow cell (see TILEQC (Dolan & Denver 2008) for a more specialized tile-specific QC tool). Using these tools on the same high-quality data set examined in the 'Health Check-up' section (16.5 million 93-bp sequence reads) showed that tools written in compiled languages [i.e. HTQC, FASTQC and FASTX (STATISTICS; http://hannonlab.cshl.edu/fastx_toolkit)] were much faster than others (Fig. 1).

Table 1 Summary of tools for general quality assessment of HTS data.

Name	Supported technologies	Features	Link/Reference	Note
BIGPRE	Illumina, 454	CC, CR, DF, DL, PE, QC, QR, QT	http://sourceforge.net/projects/bigpre; (Zhang *et al.* 2011)	
FASTQC	All	CC, CR, DL, LD, OK, OS, PL, QC, QR	http://www.bioinformatics.babraham.ac.uk/projects/fastqc	Graphical interface; support compressed input
FASTQ-UTILS	FASTQ	LD, QC	http://ngsutils.org; (Breese & Liu 2013)	
FASTX (STATISTICS)	Illumina	CC, QC	http://hannonlab.cshl.edu/fastx_toolkit	
HTQC	Illumina	CC, DF, DL, LD, PE, PL, QC, QR, QT	http://sourceforge.net/projects/htqc; (Yang *et al.* 2013b)	
NGS QC TOOLKIT	Illumina, 454	CC, DF, LD, QC, QR	http://59.163.192.90:8080/ngsqctoolkit; (Patel & Jain 2012)	
PRINTSEQ	Illumina, 454	CR, DF, DL, LD, QC, QR	http://prinseq.sourceforge.net/index.html; (Schmieder & Edwards 2011b)	Additional features for metagenomic data
QRQC	FASTQ	CC, CR, LD, OK, QC, QR	http://www.bioconductor.org/packages/2.14/bioc/html/qrqc.html	R package in BIOCONDUCTOR
SOLEXAQA	Illumina	DF, LD, QC, QT	http://solexaqa.sourceforge.net; (Cox *et al.* 2010)	

CC, composition per cycle; CR, composition per read; DF, data filtration; DL, duplication level; LD, length distribution; OK, overrepresented *k*-mer; OS, overrepresented sequence; PE, supports paired-end reads; PL, parallelization; QC, quality per cycle; QR, quality per read; QT, quality per tile.

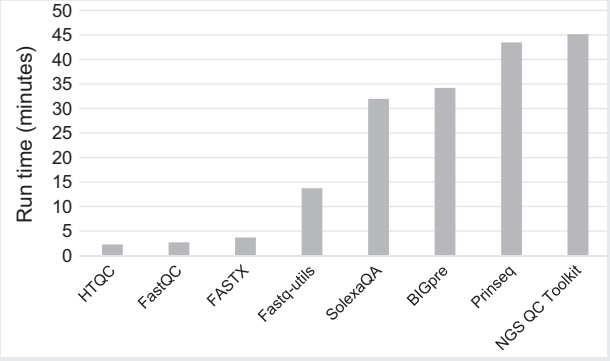

Fig. 1 Run-time comparison of general quality assessment tools. QRQC was not included in the evaluation because it requires use of the R environment. BIGPRE and SOLEXAQA can perform subsampling of sequence reads, which can substantially reduce their run time. Note that PRINSEQ is designed primarily for 454 technology, which generates fewer but longer sequence reads. The run time of each tool was measured 10 times, and the average value is shown. All tools were run on a single CPU thread on a desktop with two Intel CPUs and eight gigabytes physical memory.

A useful illustration of these core QC metrics is provided by the QC report generated by the popular tool FASTQC (http://www.bioinformatics.babraham.ac.uk/projects/fastqc) on a high-quality Illumina sequence read data set. We have also used this high-quality data set as a reference against which we compare other data sets that show a wide variety of quality problems. For this example, as well as for all other examples in this review, we have strived to present the original software-generated figures whenever possible or practical so that they most closely match what readers will likely encounter in their own analyses.

The most basic statistics for a HTS data set include the number of sequence reads and range of sequence read length (Fig. 1a; FASTQC also generates a detailed plot on read length distribution, which is not shown here). Typically, the sequence reads produced by Illumina sequencing are of identical length; variation in the length of sequence reads indicates that some kind of sequence read trimming has already been performed (e.g. the trimming of adapters that takes place automatically for data sets generated on Illumina MiSeq instruments). However, variation in the length of sequence reads is the expected outcome for sequence read data sets generated by other HTS technologies such as 454 and PacBio (Glenn 2011), making it more difficult to assess whether read trimming has already been performed for a given data set. In addition, FASTQC can

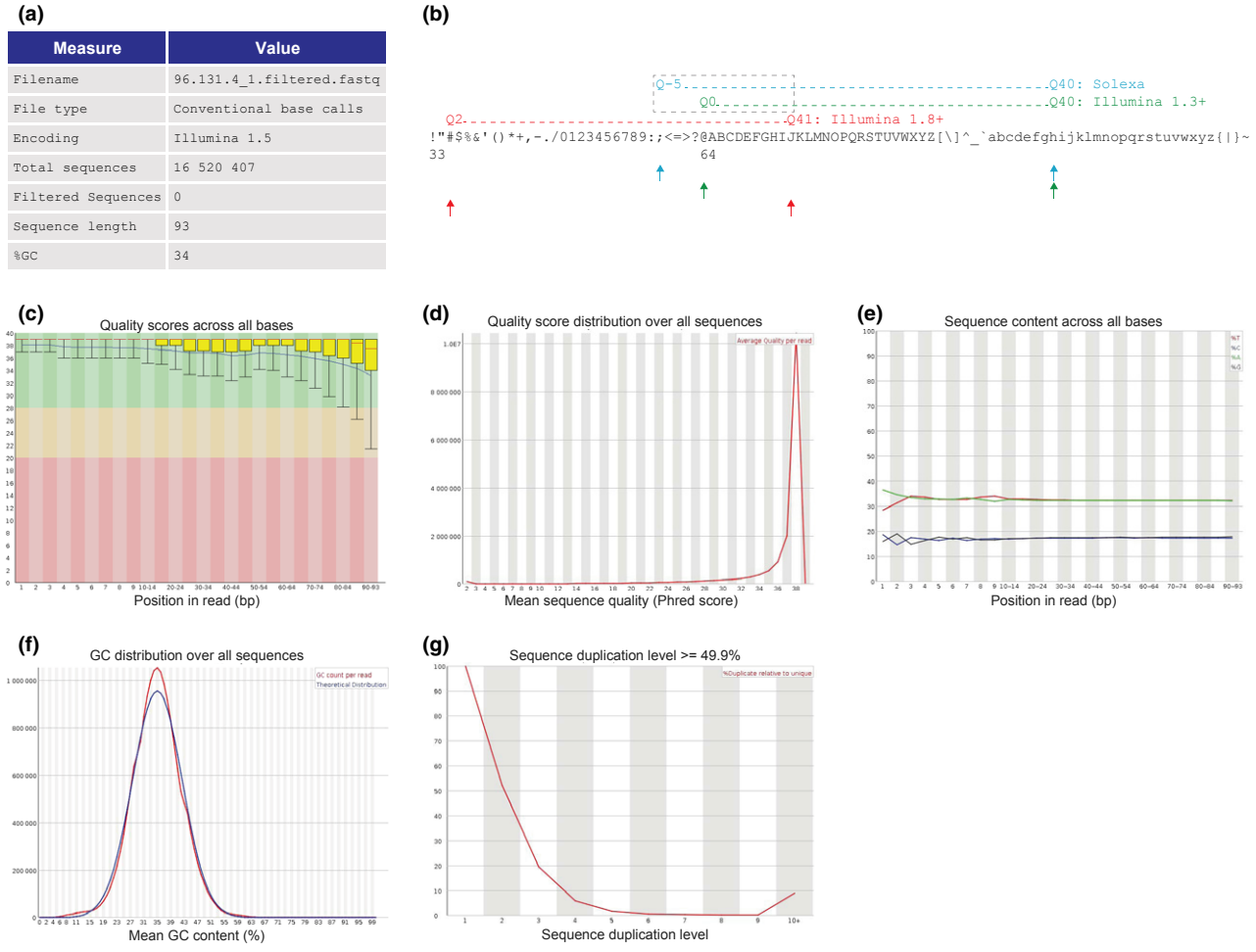

Fig. 1 Exemplar general quality assessment reports generated by FASTQC on a high-quality high-throughput sequencing data set. (a) Overall summary statistics. (b) Illustration of the three base quality score formats used by Illumina; 'Solexa' and 'Illumina 1.3+' scores represent different error rates for value below 10 (highlighted in the black dashed line box). (c) Distribution of quality score across bases. (d) Distribution of quality score across reads. (e) Distribution of sequence composition across bases. (f) Distribution of GC content across reads. (g) Estimated level of duplicate reads.

detect the format in which base quality scores are encoded. The standard output format for Illumina data is FASTQ where base quality scores are presented as ASCII printable characters; the format of data generated by other technologies is either FASTQ (e.g. data generated by Ion Torrent) or other formats that can be readily converted to FASTQ (e.g. the SFF files generated by 454 and the XSQ files generated by SOLiD) using tools such as SEQ_CRUMBS (for SFF files; http://github.com/JoseBlanca/seq_crumbs) and XSQ_TOOLS (for XSQ files; http://www.lifetechnologies.com/us/en/home/technical-resources/software-downloads/xsq-software.html).

Historically, three incompatible variants of FASTQ format have been used for Illumina-generated HTS data sets (Cock *et al.* 2010): 'Solexa', which encodes Solexa base quality scores with an ASCII offset of 64; 'Illumina 1.3+' (or 'Illumina 1.5+', 'Phred64'), which encodes Phred base quality scores with an ASCII offset of 64; and 'Illumina 1.8+' (or 'Sanger', 'Phred33'), which encodes Phred base quality scores with an ASCII offset of 33 (Fig. 1b). Getting the base quality score format right matters because many tools used for downstream HTS data analysis do not currently automatically detect the base quality score format of an HTS data set; thus, misspecification of base quality score format will confound downstream analyses such that a high-quality HTS data set in 'Phred33' format might be treated as a low-quality data set if analysed using software whose default format is 'Phred64'.

Two of the most important questions regarding any HTS sequence read data set are whether the data set is of high quality and whether it accurately represents the underlying biological sample. Answers to these questions can be obtained by examining quality metrics on the distribution of base quality score and nucleotide frequency for each sequencing cycle or read (Fig. 1c–f). Successful HTS experiments produce high average base quality scores for the vast majority of sequence reads across all sequencing cycles, although it is typical to observe a decline in base quality score towards the far 3′-end of Illumina sequence reads (Dohm *et al.* 2008; Minoche *et al.* 2011). For instance, FASTQC considers a HTS data set to be of high quality if the error rates for lower-quartile and median base quality scores are 10% or lower and ~0.3% or lower, respectively, for all sequencing cycles. In addition, given that the nucleotide bases sequenced at a given sequence cycle in most HTS experiments are randomly sampled, we expect the frequencies of the four nucleotides to be even across sequencing cycles and the overall GC content of the sequence reads to match that of the targeted sample (e.g. if the reads are from the exome of a particular species, we expect the GC content of the reads to match that of the exome of the species). Deviations from these expectations indicate the presence of biases or artefacts in the sequence read data set (see later section for detailed discussion).

Several other metrics can be highly informative in identifying potential problems. For instance, the plot of sequence duplication level (Fig. 1g) depicts the frequency of sequence reads which appear more than once in the data set (note that FASTQC only examines the first 200 000 reads, and if reads are longer than 75 bp, it checks only the first 50 bp). If the sampling of DNA fragments in HTS is truly random, the chance that two or more identical fragments will be sequenced will be small. Therefore, the duplication level is expected to be low and a high duplication level might suggest biased sampling. In contrast, it is normal to see higher levels of duplication in sequence read data sets stemming from the HTS of samples with highly uneven coverage, such as from RNA-Seq experiments. To help identify potential contamination, FASTQC also reports sequence reads with a high frequency of occurrence and matches them with the primers and adapters used in library preparation and sequencing.

Another way for evaluating a data set for biased sampling and/or contamination is by examining the data set for the presence of overrepresented *k*-mers. A *k*-mer is a sequence fragment of length *k* from a sequence read and is considered as overrepresented if its observed occurrence in the data set at a given read position is substantially higher than expected assuming random sampling. This approach has better sensitivity in detecting problems such as contaminations that affect only parts of sequence reads. For 'healthy' data sets, FASTQC simply reports 'no overrepresented *k*-mers (or sequences)', whereas for contaminated or otherwise biased data sets the program reports a plot of overrepresented *k*-mers (see later section for example).

'Common pathologies' of HTS sequence read data

Having established what is expected for a 'healthy' HTS data set, we move on to discuss the origins, diagnosis and treatment of various HTS sequence read data 'pathologies'. Under the category of 'common pathologies' falls a list of common problems in standard HTS sequence read data sets that are closely associated with library preparation and the sequencing process itself. Identifying and correcting these problems constitutes the major body of a typical QC workflow for HTS sequence read data.

Chastity-filtered reads

During Illumina sequencing, several metrics are used to monitor and control the quality of the generated data.

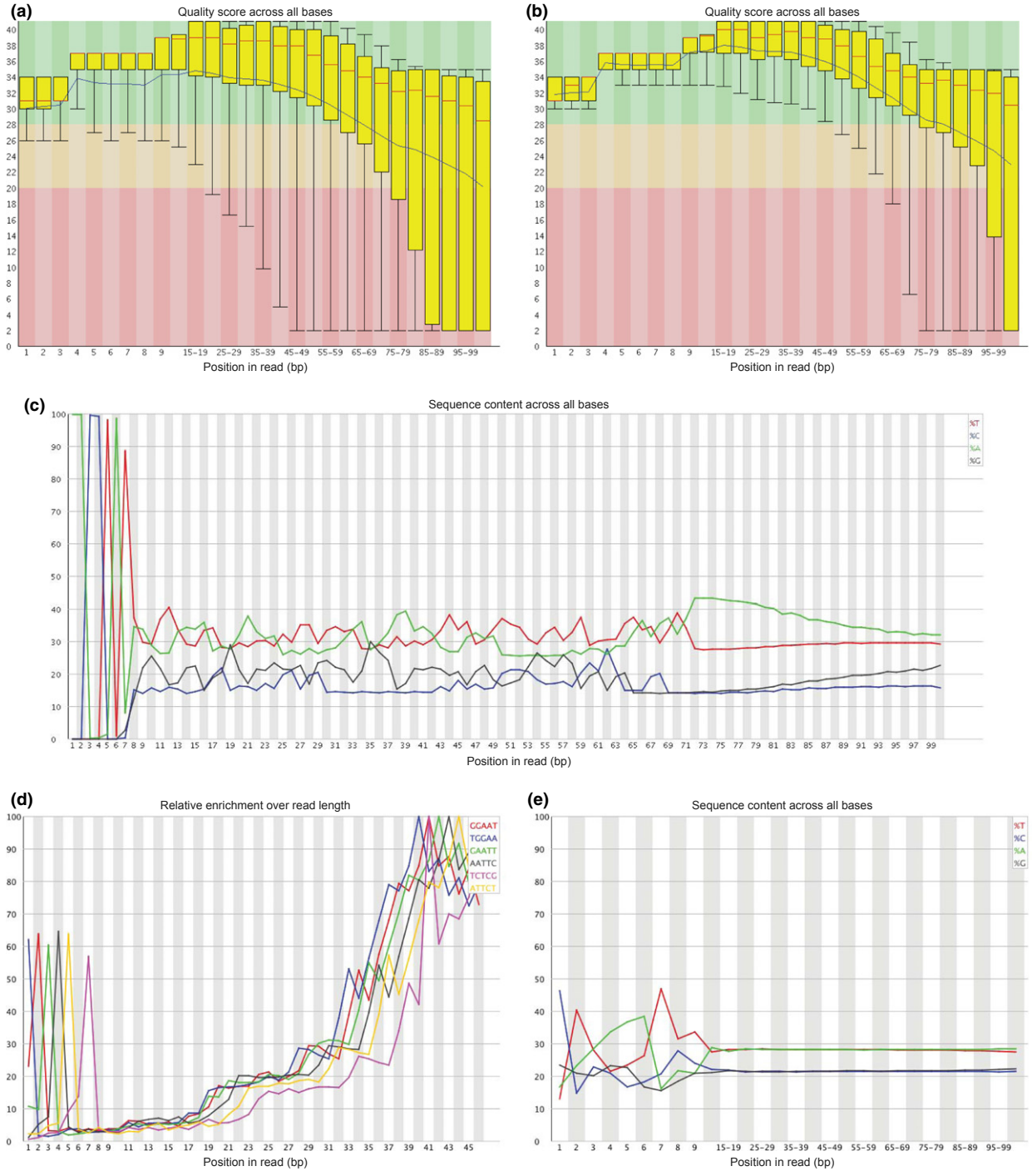

Fig. 2 Exemplar FASTQC plots demonstrating the diagnosis of common 'pathologies' present in high-throughput sequencing data. (a, b) Distribution of quality scores across bases for the same Illumina data set with (panel a) and without (panel b) removing *CHASTITY*-filtered reads. The overall quality of the data set is substantially improved after filtering, yet a decrease in average base quality score towards the 3′ end is still observed. In addition, the first few bases at the beginning of reads have lower quality. (c) Distribution of sequence composition across bases for a barcoded Illumina data set. This data set has untrimmed barcode sequence (almost all reads have the same sequence in the first six bases) and likely contamination of adapter sequences (irregular composition along the read). (d) Relative enrichment level of *k*-mers over read positions for the *Drosophila* nuclear run-on sequencing data set. The enrichment of multiple consecutive, overlapping *k*-mers near the 5′ and 3′ ends of sequence reads indicates adapter dimer and 3′ adapter contamination, respectively. (e) Distribution of sequence composition across bases for an Illumina RNA-seq data set. The variation in sequence composition at beginning of reads is caused by nonrandom priming and is commonly observed among RNA-seq data sets.

One such metric, known as CHASTITY, measures the signal purity in a given sequencing cycle by comparing signals from the four possible bases; it equals the ratio of the intensity of the brightest base over the summed intensity of the two brightest bases. A sequence read fails the CHASTITY filter if more than one of the first 25 sequencing cycles has a CHASTITY value lower than a threshold. It has been shown that the removal of these low-quality sequence reads can greatly reduce the error rate of HTS data (Minoche *et al.* 2011), whereas their inclusion may lead to dramatic decrease in the overall data quality (Fig. 2a–b). Some older versions of the Illumina Consensus Assessment of Sequence and Variation (CASAVA) software (versions 1.8.0 and 1.8.1) mandatorily report both the reads that passed and failed the CHASTITY filter, whereas in the most recent version (version 1.8.2) this report is an optional feature and is usually disabled by default (e.g. on MiSeq platform). The latest FASTQ format used by Illumina ('Illumina 1.8+') flags CHASTITY-filtered reads by including a 'Y' in their sequence identifiers. FASTQC (in CASAVA mode) reports the number of CHASTITY-filtered reads in a data set and, if present, such reads can be filtered by using custom script or compiled tools (e.g. the FASTQ_ILLUMINA_FILTER program available at http://cancan.cshl.edu/labmembers/gordon/fastq_illumina_filter).

Sequence reads of low quality

Powerful as they are, HTS technologies generate sequence read data sets with nontrivial amounts of sequencing errors (Glenn 2011), which can majorly confound downstream data analysis (Taub *et al.* 2010). Sequencing errors introduce artefactual discrepancies in the acquired sequence read data, thus complicating studies that rely on sequence read mapping. Sequencing errors are particularly harmful in *de novo* assembly studies; they prevent proper overlap between sequence reads and dramatically increase the complexity of the assembly process, leading to both heavier computational burden and lower assembly quality (Paszkiewicz & Studholme 2010).

The types and characteristics of sequencing errors vary among HTS technologies (Glenn 2011). In data generated by Illumina, the primary error type is incorrect base calling, which is more frequent towards the 3′ end of sequence reads mainly due to the phasing artefact (Dohm *et al.* 2008; Minoche *et al.* 2011). Specifically, for each read, the sequence at each cycle of the sequencing reaction in Illumina technology is determined based on the combined signal emanating from a large number of fragments, also known as cluster, generated from clonal amplification of the same template. Each cycle involves the addition of nucleotides with reversible terminator chemistry so that a single base is incorporated to all the fragments in each cluster being sequenced; at the end of each cycle, terminators are removed to allow further extension of the sequenced fragments in subsequent cycles (Mardis 2013). However, a small fraction of fragments in each cluster either fails to incorporate any base (phasing) because of incomplete removal of terminators from the previous cycle, or incorporates more than one base (prephasing) due to the lack of terminator in some nucleotides (Mardis 2013). The noise caused by these unsynchronized fragments in each cluster builds up with increasing number of sequencing cycles, leading to lower base quality scores. Other factors contributing to sequencing errors include interfering signals from neighbouring clusters or bases with similar emission spectra (e.g. between G and T) (Kircher *et al.* 2011). In addition, extraneous objects on the flow cell (e.g. dust, air bubbles) have negative impact on sequencing quality and may even be misidentified as sequencing clusters, which in turn generate sequence reads of low complexity (Kircher *et al.* 2011).

A commonly used strategy to handle sequencing errors and artefacts in HTS data is the trimming of low-quality sequence reads (Kircher *et al.* 2011; Minoche *et al.* 2011). It improves the overall data quality and can usually give rise to better results in downstream analyses (see Box 2 for an example on genome assembly). However, trimming can also lead to the loss of potentially useful information which may bias data interpretation (Delmont *et al.* 2013; Yang *et al.* 2013a). Therefore, determining how much trimming is necessary is not always straightforward (Bokulich *et al.* 2013; MacManes 2014). Nevertheless, data sets should be filtered for sequence reads that are unreliable or likely artefacts. In Illumina data, many sequence reads contain consecutive bases with a base quality score of 2 at their 3′ ends; these low-quality segments can be detected and masked by sequencing software. It has been shown that the removal of these unreliable bases has the most dramatic effect on reducing error rates (Minoche *et al.* 2011). In addition, as mentioned above, low-complexity reads likely represent sequencing artefacts; they can be removed using tools such as REAPER from the KRAKEN package (Davis *et al.* 2013).

Box 2 Adapter and quality trimming tools for HTS sequence read data

Adapter contamination is a prevalent problem in HTS studies, and many computational solutions have been developed to tackle it (Table 1). To better understand their performance, we evaluated these adapter trimming tools on simulated Illumina SE and PE data sets with various degrees of adapter contamination. We specifically focused on the removal of 3′ adapter contamination, not only because this is the most common type of adapter contamination in Illumina data but also because the lower base quality at 3′ end of sequence reads makes this task much more challenging than trimming adapters in the higher base quality 5′ end. We measured the performance of different adapter trimming tools following Lindgreen's approach (Lindgreen 2012). Briefly, we defined true positive (TP) as the number of reads that are correctly trimmed; true negative (TN) as the number of adapter free reads that are not trimmed; false positive (FP) as the number of reads where nonadapter sequences are trimmed; and false negative (FN) as the number of reads that still contain adapter sequence after trimming. We then calculated the Matthew's correlation coefficient (MCC), a commonly used quality measurement for binary classifier, based on the following formula:

$$\mathrm{MCC} = (\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}) / \sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}$$

As shown in Fig. 1, our results suggest that SCYTHE (http://github.com/vsbuffalo/scythe), CUTADAPT (Martin 2011) and ADAPTERREMOVAL (Lindgreen 2012) showed the best overall performance in trimming SE read data, and SEQPREP (http://github.com/jstjohn/SeqPrep), ADAPTERREMOVAL and TRIMMOMATIC (Lohse *et al*. 2012) in trimming PE read data. It should be noted that all tests were performed using the default settings of tools. While it is highly likely that tool performance could be improved through parameter adjustments, most researchers typically use these tools with their default parameter settings. In addition, we also compared the time efficiency of these programs and found that, while most tools had similar run time, a trade-off between performance and speed was observed in some cases (e.g. SEQPREP).

Many of these adapter trimming tools can also perform quality-based filtering, as well as some of the general QC assessment measures described above (Box 1). There are also dedicated quality-based trimming tools such as CONDETRI (Smeds & Kunstner 2011), SEQTK (http://github.com/lh3/seqtk) and SICKLE (http://github.com/vsbuffalo/scythe). Because the trimming strategies implemented in these tools differ from all others, comparison of their performance and efficiency is not straightforward. Nevertheless, tools that enable the use of input data in compressed format and support parallelization (e.g. TRIMMOMATIC) may have advantages in speed.

To highlight the importance of adapter and quality trimming, we present two case studies that nicely illustrate their impact on downstream analysis. The first case study is the analysis of *Drosophila melanogaster* HTS sequence reads aimed at identifying promoters bound by RNA polymerase II (Kwak *et al*. 2013), in which a large fraction of the fragments targeted for sequencing are very short (<50 bp, the length of the sequence reads) and thus the sequence reads are expected to contain a high degree of adapter contamination. We mapped sequence reads with and without adapter trimming against the fruit fly genome using BOWTIE2 (Langmead & Salzberg 2012) ('end-to-end' alignment mode) and found that both the number of total mapped reads and uniquely mapped reads significantly increased after adapter removal (Table 2). Similar results were obtained with another popular short read aligner BWA (Li & Durbin 2010) (Table 2). Mapping the original sequence reads using BOWTIE2 in 'local' alignment mode and BWA with 'soft clipping' enabled, both of which can omit unmatched nucleotides at read ends, also greatly improved mapping, although the trimming of adapter sequence prior to mapping yielded a slightly higher number of mapped reads (Table 2).

The second case study centres around the *de novo* genome assembly generated from a bacterial whole-genome sequencing data set of *Escherichia coli* (Adey *et al*. 2010). We compared the impact of several QC procedures on *de novo* genome assembly, including adapter removal (AR), quality-based trim (QT), error correction (EC) and their combinations. Assemblies were evaluated by their contig and scaffold N50 values (a commonly used statistic which refers to the largest contig/scaffold size such that half of the total assembly size is contained in contigs/scaffolds no shorter than this value). We found that: (i) as expected, QC procedures lead to significantly better *de novo* assembly; (ii) removal of adapter contamination improves assembly quality (e.g. QT vs. AR+QT, and EC vs. AR+EC; see Fig. 2 for details); and (iii) error correction produces superior results than quality trimming.

Table 1 Summary of tools for adapter trimming.

Name	Supported data type	Demultiplexing	5′-adapter	3′-adapter	Quality trim	Link/Reference	Note
ADAPTERREMOVAL	SE, PE		+	+	+	http://code.google.com/p/adapterremoval; (Lindgreen 2012)	
ALIENTRIMMER	SE, PE	+	+	+	+	ftp://ftp.pasteur.fr/pub/GenSoft/projects/AlienTrimmer; (Criscuolo & Brisse 2013)	
BTRIM	SE	+	+	+	+	http://graphics.med.yale.edu/trim; (Kong 2011)	
CUTADAPT	SE		+	+	+	http://github.com/marcelm/cutadapt; (Martin 2011)	Support colour-space data* and compressed input
EA-UTILS	SE, PE	+	+	+	+	http://code.google.com/p/ea-utils; (Aronesty 2013)	Support compressed input
FASTX (CLIPPER)	SE	+		+	+	http://hannonlab.cshl.edu/fastx_toolkit	
FLEXBAR	SE, PE	+	+	+	+	http://sourceforge.net/projects/flexbar; (Dodt *et al.* 2012)	Support colour-space data and compressed input; parallelization
KRAKEN (REAPER)	SE	+	+	+	+	http://www.ebi.ac.uk/research/enright/software/kraken; (Davis *et al.* 2013)	Support compressed input
NEXTCLIP	MP		+	+	+	http://github.com/richardmleggett/nextclip/; (Leggett *et al.* 2013)	Specially designed for Nextera MP data
SCYTHE	SE			+		http://github.com/vsbuffalo/scythe	Support compressed input
SEQPREP	PE			+		http://github.com/jstjohn/SeqPrep	Support compressed input
TAGDUST	SE		+	+		http://genome.gsc.riken.jp/osc/english/dataresource; (Lassmann *et al.* 2009)	
TRIM GLORE	SE, PE		+	+	+	http://www.bioinformatics.babraham.ac.uk/projects/trim_galore	Support compressed input
TRIMMOMATIC	SE, PE			+	+	http://www.usadellab.org/cms/index.php?page=trimmomatic; (Lohse *et al.* 2012)	Support compressed input; parallelization

SE, single-end reads; PE, paired-end reads; MP, mate-pair reads.

*Colour-space refers to the dibase encoding used by SOLiD sequencing technology, in which every possible dinucleotide is represented by one of four colours and each sequence read consists of a nucleotide at the beginning of read and a sequence of colours which indicate consecutive, overlapping dinucleotides starting from the first base position.

Fig. 1 Comparison of performance and run time of adapter trimming tools on simulated data sets. All sequence read test data were 100 bp long, paired-end (PE) Illumina reads simulated from human genome (0.1×) at 1% average error rate using PIRS (Hu *et al.* 2012), whose actual sequence fragment size follows a normal distribution. Simulated reads contained adapter sequence when the fragment size was smaller than read length (100 bp). To simulate different degrees of contamination, we generated a series of data sets with increasing fragment size distribution mean values (80–130 bp with a step size of 5 bp, corresponding to the dark grey to light grey colours in panel 1a and 1c). PE sequence reads were treated separately in the evaluation of single-end (SE) read tools. All reported run times were averaged values of 10 replicates. TAGDUST was not evaluated because it always removes the entire read if contamination is detected.

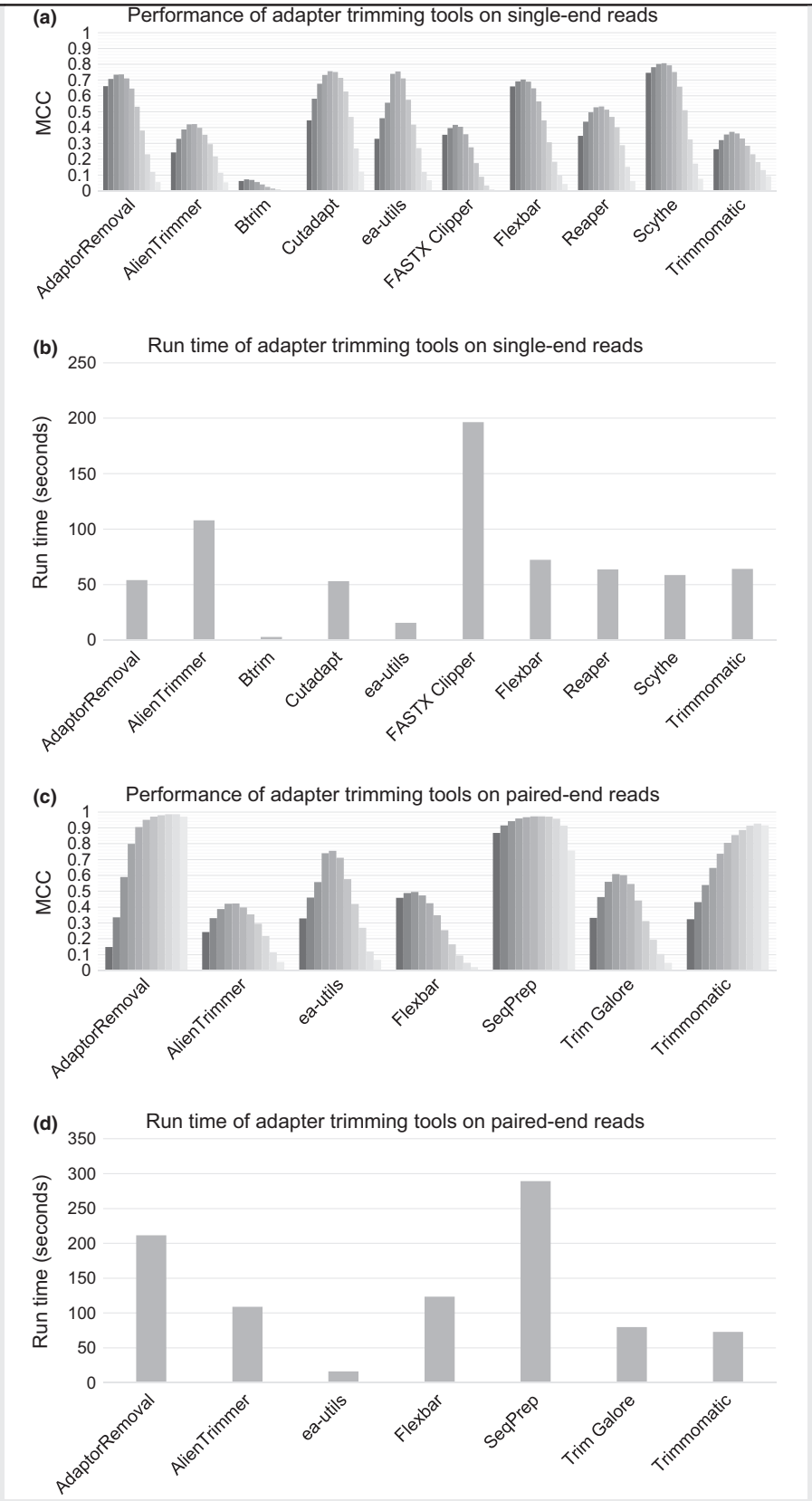

Table 2 Impact of adapter contamination on sequence read alignment. 'Original' and 'trimmed' refer to data sets before and after adapter removal. The original data set was also mapped with the 'local alignment' mode in BOWTIE2 ('Local') and 'soft clipping' function in BWA ('Soft clipping').

	BOWTIE2			BWA		
	Original	Trimmed	'Local'	Original	Trimmed	'Soft clipping'
Uniquely mapped	20 588 461 (49.63%)	31 155 597 (76.80%)	30 614 845 (73.80%)	15 598 067 (37.60%)	31 728 166 (78.21%)	29 515 461 (71.15%)
Total mapped	25 262 729 (60.90%)	38 272 839 (94.35%)	37 774 172 (91.06%)	19 521 568 (47.06%)	38 101 178 (93.92%)	34 954 500 (84.26%)
Total reads	41 484 479	40 566 782	41 484 479	41 484 479	40 566 782	41 484 479

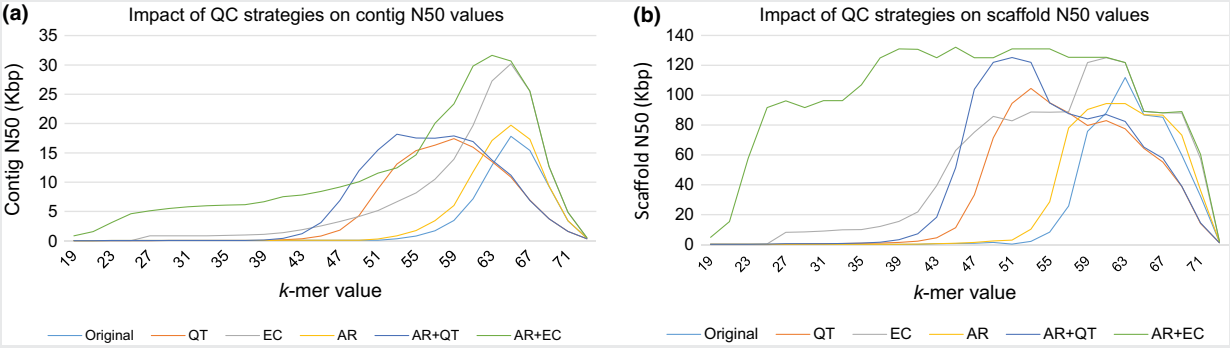

Fig. 2 Impact of different quality control (QC) strategies on *de novo* assembly of the *Escherichia coli* genome. We performed adapter trimming on the original data set and found that about 16% read pairs have adapter contamination; both the original and adapter trimmed data sets were then subjected to quality trimming and error correction. The six data sets were assembled using SOAPDENOVO (Li *et al.* 2010b) with multiple *k*-mer values. Two most commonly used measurements of assembly quality, contig N50 (a) and scaffold N50 (b) values were shown for *de novo* genome assemblies generated from the original data set and five other data sets derived from various combinations of QC procedures, including adapter removal ('AR'), quality-based trim ('QT') and error correction ('EC'). Each data set was assembled using *k*-mer values from 19 to 73.

Further quality-based trimming, if necessary, should be applied after the removal of adapter contamination (see next section), because bases with lower base quality scores can still be useful for the identification of adapter sequences. For the purpose of sequence read mapping, aligners that allow for soft clipping (i.e. ignoring mismatched bases at the beginning/end of sequence reads according to certain criteria) such as BWA (Li & Durbin 2010) and BOWTIE2 (Langmead & Salzberg 2012) can deal with mismatched bases on the fly. Alternatively, quality-based trimming could be applied beforehand if the 'soft clipping' option is not available during alignment (e.g. the popular spliced aligner for RNA-seq data, TOPHAT2 (Kim *et al.* 2013), does not support the 'local' alignment mode of BOWTIE2).

In *de novo* assembly studies, low-quality data can be partially rescued through error correction rather than being discarded entirely, which often leads to superior results (Box 2). Assuming that each base in the genome/transcriptome is sequenced multiple times and that sequencing errors occur randomly and infrequently, a sequence read containing errors can be corrected by comparing it to the many other reads that cover the same sequence region and whose sequences are without errors or do not contain the same errors. Error correction has been shown to significantly improve the quality of *de novo* assembly of both genomes (Salzberg *et al.* 2012) and transcriptomes (MacManes & Eisen 2013). Due to the significant benefits from error correction in *de novo* assembly, a number of tools are available to perform this task; many of them have been summarized in a recent review (Yang *et al.* 2013a), yet new tools are still being developed (Ilie & Molnar 2013; Le *et al.* 2013; Liu *et al.* 2013b; Marcais *et al.* 2013; Nikolenko *et al.* 2013; Sleep *et al.* 2013), including ones that are specially tailored for data sets with highly uneven coverage (e.g. SEECER for transcriptome data and BAYESHAMMER for single-cell sequencing studies). Some tools will automatically remove low-quality data remaining after error correction (Kelley *et al.* 2010; Marcais *et al.* 2013); alternatively, a final pass quality-based trimming may be

performed separately. For a more detailed discussion on error correction of HTS data, we refer interested readers to the review by Yang and coworkers (Yang *et al.* 2013a).

Recent sequence read data sets generated using the Illumina technology also show a drop in base quality scores towards the 5′ end of reads; specifically, in these data sets, the base quality scores appear to be capped at 34, 37 and 39 for the sequencing cycles 1–3, 4–8 and 9–13, respectively. This pattern consistently appears across data sets generated in different types of studies (e.g. DNA-seq, RNA-seq) or by different laboratories, suggesting that it is unlikely to be due to the poor sample/sequencing quality of particular studies or experiments. Rather, it seems that the sequencing software behaves more conservatively when assigning base quality scores for the first few bases. Therefore, these bases with seemingly lower quality are normal and may not need to be trimmed unless other artefacts coexist.

Adapter contamination

In HTS experiments, fragmented DNA/RNA molecules are ligated with adapters on both ends prior to sequencing. Adapter contamination is a common problem in HTS data, and it can occur in several scenarios: (i) if the sequence read length is larger than the fragment size, the read will include sequence of the adapter in its 3′ end (3′ contamination); (ii) adapters might be ligated together without any fragment insert in-between and the adapter itself is sequenced (adapter dimer); and (iii) sequence reads generated from certain library preparation protocols [e.g. Nextera mate-pair (MP) library] may also contain adapter sequences in their 5′ end (5′ contamination).

Adapter contamination has detrimental effects on various downstream analyses (Box 2). For instance, the presence of adapter sequences may prevent contaminated sequence reads from being correctly mapped to their reference sequences (Kircher *et al.* 2011), which may in turn affect mapping-based analyses such as variant calling and abundance estimation. Adapter contamination can also introduce noise into *de novo* assembly analyses (Martin & Wang 2011); in some extreme cases, the presence of adapter contamination can lead to highly fragmented genome assembly and lower the average size of assembled sequences by several orders of magnitude (see more details at http://pathogenomics.bham.ac.uk/blog/2013/04/adaptor-trim-or-die-experiences-with-nextera-libraries/). Finally, adapter contamination may bias data analysis and lead to incorrect conclusions. For example, Keegan and coworkers developed a novel method to evaluate the quality of HTS data by comparing putative duplicate sequence reads, and reported surprisingly high levels of error for multiple Illumina data sets (Keegan *et al.* 2012). However, it was found later that the majority of errors they identified were a result of adapter contaminations (Eren *et al.* 2013). Therefore, the removal of adapter contamination is critical to a successful HTS data analysis.

As discussed in the 'health check-up' section, adapter contamination can be identified by examining sequence composition and overrepresented sequences/*k*-mers. Irregular sequence composition along reads overrepresented sequences that match known adapter/primer sequences and consecutive overlapping *k*-mers that are overrepresented towards the 3′ end of reads all suggest adapter contamination (Fig. 2c–d). In sequence read mapping, adapter contamination can be handled by soft clipping-enabled aligners such as BWA and BOWTIE2, in a manner similar to the handling of sequencing errors near the ends of reads (see previous section); otherwise, adapter trimming has to be performed prior to any downstream analyses. When the sequences of adapters used in an HTS experiment are known, they can be filtered from the resulting sequence read data set using one of the many available tools (Box 2). If the sequences of adapters are not known, they can be first inferred from the data set using MINION from the KRAKEN package (Davis *et al.* 2013) and then trimmed accordingly.

Discordant paired-end sequence reads

Paired-end (PE) sequencing is a powerful option available on most HTS platforms; it sequences each DNA fragment from both ends, generating a pair of sequence reads. This pairing information is extremely useful for many applications, such as *de novo* genome and transcriptome assembly (Treangen & Salzberg 2012; Wolf 2013), identification of gene fusion events (Maher *et al.* 2009) and detection of genome structural variation (Alkan *et al.* 2011a). The sequence reads generated by PE sequencing are usually stored in two separate FASTQ files, each of which contains the reads of one or the other end; the pairing information is recorded by the fact that the reads of each pair have the same sequence identifier except for the one-digit label ('1' or '2') that distinguishes the two ends. However, many HTS downstream data analysis tools assume that the reads from a given pair appear in the same order in both files without explicitly checking the sequence identifier. Therefore, it is important to keep PE sequence reads in the exact same order in the two files to avoid unexpected results. For instance, PE sequence reads are important in *de novo* genome/transcriptome assembly as their longer length and pairing aid HTS assemblers in connecting short contigs into larger scaffolds. Incorrect pairing information will confound such analyses, because out-of-order PE data can create both false

positives, where contigs are mistakenly connected, and false negatives, where genuine scaffolds are missed.

The order of PE sequence reads might be disrupted during data processing if the files containing the sequence reads from the two ends are manipulated separately. Most likely this would occur when reads are filtered for low-quality or adapter contamination; some sequence read pairs may have one read discarded and the other retained, so that the two processed files will have different read numbers and unmatched read orders. In addition, although unusual, PE sequence reads might be out of order within original data sets (as encountered in one of our own studies as well as in some public data sets). Therefore, we recommend that researchers should always check the order of sequence reads in PE data sets, and fix it if necessary, prior to any analysis; one tool for validating sequence read order in PE data is available at http://www.mcdonaldlab.biology.gatech.edu/bioinformatics/FastqPairedEndValidator.pl; any discordant PE data sets can be fixed using PECOMBINER (De Wit *et al.* 2012).

Duplicate sequence reads

Duplicate sequence reads are frequently observed in HTS data sets (Mamanova *et al.* 2010); it is not rare for FASTQC to generate a 'warning' (when duplication level is ≥20%) or even 'failure' (when duplication level is ≥50%) during the evaluation of sequence duplication in a data set. Duplicate sequence reads have three main sources: (i) natural duplicates, which represent independent molecules with very high sequence similarity present in the original sample that when sequenced produce sequence reads that are identical; (ii) PCR duplicates, which all stem from the same original molecule through PCR amplification during HTS library preparation; and (iii) optical duplicates, which are generated by the same DNA cluster on a sequencing flow cell but are mistakenly identified as separate clusters by image capture software used during HTS. Ideally, one wishes to retain all natural duplicates in a HTS experiment, but remove all artefactual PCR and optical duplicates.

Among the three types of duplicates, the most easily identified are optical duplicates; they have the same sequence (or, if sequencing errors occur, highly similar ones) and neighbouring coordinates on the flow cell. Thus, duplicate sequence read artefacts stemming from optical duplicates can be removed using the MARKDUPLICATES program from the PICARD package (http://picard.sourceforge.net). Identifying and removing PCR duplicates is more challenging because natural duplicates and PCR duplicates are indistinguishable at the sequence read level. Because a large fraction of duplicate reads found in single-end (SE) read data sets is likely derived from distinct molecules (Bainbridge *et al.* 2010), blind removal of all duplicate reads will lead to the loss of all natural duplicates, which might result in substantial loss of genuine sequence data. In addition, the expected level of natural duplicates is positively correlated with the coverage of HTS experiment. The situation is greatly improved by using PE sequence reads because it is much less likely that independent molecules are identical at both ends, but the level of PCR duplicates may still be overestimated.

The best current solution to the problem of PCR duplicates is the use of library preparation protocols that are amplification-free (Kozarewa *et al.* 2009) or able to explicitly distinguish between PCR and natural duplicates (Shiroguchi *et al.* 2012). From the perspective of HTS sequence data QC, deciding what is the best treatment for duplicate reads largely depends on the nature of the study. The removal of duplicate reads is an essential step in the discovery of single-nucleotide polymorphisms because errors introduced in early cycles of amplification are shared by PCR duplicates and lead to high false-positive rate (DePristo *et al.* 2011). It is also usually recommended to remove duplicate sequence reads in ChIP-seq analyses for more reliable results (Chen *et al.* 2012). In these alignment-based analyses, duplicates are often defined as reads that are mapped to the exact same location in the reference sequence; they can be removed by PICARD or SAMTOOLS (Li *et al.* 2009). In *de novo* genome assembly, high levels of duplicates may negatively affect the construction of sequence scaffolds and increase memory usage (Martin & Wang 2011). In the absence of reference sequences, the removal of duplicates has to be based on comparing sequences of reads (Box 3). Alternatively, more sophisticated filtering options can be considered (see later discussion on digital normalization).

Box 3 Tools for sequence-based deduplication of HTS sequence reads

A common strategy for sequence-based deduplication is to collapse sequence reads or read pairs that have identical sequences, as implemented in tools such as FASTUNIQ (Xu *et al.* 2012), FASTX (COLLAPSER; http://hannonlab.cshl.edu/fastx_toolkit) and KRAKEN (TALLY) (Davis *et al.* 2013). While they are all able to process tens of millions of sequence reads within minutes, KRAKEN is more memory efficient than the other two tools and supports both SE and PE sequence reads. These tools also differ in how they report base quality values for collapsed reads; while FASTX only generates FASTA output, KRAKEN combines the highest score observed among all duplicates for each base, whereas

FASTUNIQ simply reports scores associated with the last appearing duplicate in the data set. The program FALCRUM (Burriesci *et al.* 2012) can collapse near-identical sequence reads in addition to identical ones by ignoring low-quality differences between reads, but can be substantially slower than tools that require exact match, thus limiting its utility for large data sets. Several other tools are available for removing duplicate reads from 454 data, including PRINSEQ (Schmieder & Edwards 2011b), PYROCLEANER (Jerome *et al.* 2011) and JATAC (Balzer *et al.* 2013).

Table 1 Summary of tools for sequence-based deduplication of HTS sequence reads.

Name	Supported data type	Quality value	Link/Reference	Note
FASTUNIQ	PE	Quality value of the last appearing duplicate	http://sourceforge.net/projects/fastuniq; (Xu *et al.* 2012)	
FASTX (COLLAPSER)	SE	NA	http://hannonlab.cshl.edu/fastx_toolkit/index.html	
FULCRUM	SE, PE, 454	Consensus of all duplicates	http://pringlelab.stanford.edu/projects.html; (Burriesci *et al.* 2012)	Can also remove near-identical reads
KRAKEN (TALLY)	SE, PE	Highest value among all duplicates for each base	http://www.ebi.ac.uk/research/enright/software/kraken; (Davis *et al.* 2013)	

The problem is more complicated for RNA-seq studies, where the range of transcript abundance can be extremely broad (Wang *et al.* 2009). On the one hand, the removal of duplicate reads will cause genes with higher transcript abundances to have their expression levels underestimated, as they are expected to have more natural duplicates. On the other hand, given that rates of PCR amplification are not equal for all genes, keeping all duplicate reads will overestimate the expression levels of genes that are more efficiently PCR amplified (Kozarewa *et al.* 2009). Here, the better strategy is to model the probability of natural duplicates in a given HTS sequence read data set and adjust the observed number of duplicate sequence reads accordingly. This solution is implemented by IRECKON (Mezlini *et al.* 2013) and RASTA (Baumann & Doerge 2013).

Biases

High-throughput sequencing experiments are complex, consisting of multiple steps from sample collection and library preparation to sequencing, during which various biases can be introduced (Wang *et al.* 2009; Taub *et al.* 2010; Aird *et al.* 2011). Perhaps one of the most disturbing biases for many researchers is the skewed sequence composition at the 5′ end of reads generated in transcriptome studies, which is readily apparent when examining the per-base nucleotide frequency plot generated by general quality assessment (Fig. 2e). This bias is caused by the use of random hexamer primers and may lead to nonuniform distribution of sequence reads in the transcriptome (Hansen *et al.* 2010). Trimming the affected bases from the 5′ end of sequence reads does not alleviate the problem because the bias in sequence read distribution has already been introduced (Hansen *et al.* 2010). Another potential reason for trimming is that the skewed base frequency present in 5′ end of sequence reads may compromise *de novo* transcriptome assembly. However, our analysis showed that such trimming would reduce the completeness of assembly, likely due to the loss of information (Box 4). To correct the nonrandom priming bias, Hansen *et al.* (2010) proposed an approach that weighs each sequence read according to its first heptamer. Other more general solutions are also available to deal with positional and sequence biases in sequence read distribution which can improve the estimation of gene expression levels (Li *et al.* 2010a; Roberts *et al.* 2011).

Box 4 Trimming the 5′ end of sequence reads with skewed base composition negatively affects *de novo* transcriptome assembly

To evaluate the impact of 5′-end trimming on the quality of *de novo* transcriptome assembly, we analysed Illumina RNA-seq data from mouse (Grabherr *et al.* 2011) and rice (Zhang *et al.* 2010), both of which have biased nucleotide frequencies at the 5′ end of sequence reads. For both species, we performed *de novo* transcriptome assembly on data sets without and with 5′-end trimming and obtained greater numbers of both assembled transcripts in total and those longer than 1 Kbp without trimming than with it (Table 1). We further compared the completeness of the assemblies generated from original and 'trimmed' data sets by measuring the number of annotated genes and isoforms that were fully or nearly fully recovered by assembled transcripts. Our results (Table 1) showed that, in both cases, 5′-end trimming leads to a

significantly lower number of recovered genes/isoforms. Our results highlight the importance of judicious use of QC procedures in HTS data analysis; while proper treatments of artefacts and errors can be greatly beneficial (see case studies in Box 2), too aggressive data filtering may lead to the loss of useful information and harm downstream analysis.

Table 1 Impact of 5′-end trimming on the completeness of *de novo* transcriptome assembly. For both mouse and rice, the 'trimmed' data sets were generated by removing the first 12 nucleotides from all sequence reads in the original transcriptome data. All data sets were assembled using SOAPDENOVO-TRANS (Xie *et al.* 2013). Assembled transcripts of mouse and rice were subsequently aligned against annotated transcripts in respective genomes using LASTZ (http://www.bx.psu.edu/miller_lab/) and an identity cut-off of 95%. A gene was considered recovered if at least one of its isoforms met the criteria on alignment coverage.

		Mouse		Rice	
		Original	Trimmed	Original	Trimmed
Number of assembled transcripts	All	51 961	46 044	80 879	75 056
	≥1 Kbp	12 883	11 889	15 922	13 026
Coverage = 100%	Genes	4707	3613	1371	710
	Isoforms	6821	4997	1503	757
Coverage ≥95%	Genes	9060	8366	5818	3765
	Isoforms	17 610	15 924	7329	4732

Another well-characterized bias that is prevalent in HTS data is the nonuniform relationship between GC content and read coverage (Dohm *et al.* 2008; Benjamini & Speed 2012; van Heesch *et al.* 2013); regions that are either low or high in GC content tend to have relatively lower coverage. GC-content bias is mainly caused by PCR amplification, which is a common library preparation step in HTS experiments (Kozarewa *et al.* 2009; Aird *et al.* 2011). Therefore, GC-content bias broadly affects HTS studies that rely on read coverage information, such as variant calling (Rieber *et al.* 2013), detection of copy number variation (Teo *et al.* 2012) and also *de novo* assembly (Chen *et al.* 2013). In comparative analyses (e.g. differential gene expression) where the same sequence regions (e.g. genes) are compared across samples, GC-content bias may be ignored if samples are equally affected; however, although not consistently observed in all studies (Dillies *et al.* 2012), GC-content bias has been found to be sample specific in some data sets (Pickrell *et al.* 2010). To better inform downstream analysis, the pattern of GC-content bias can be examined using RNA-SEQC (DeLuca *et al.* 2012) for transcriptome studies or using the COLLECTGCBIASMETRICS program from the PICARD package for HTS data in general. Tools for GC-content bias correction are available for different downstream applications, including CQN (Hansen *et al.* 2012) and EDASEQ (Risso *et al.* 2011) for RNA-seq, BEADS (Cheung *et al.* 2011) for ChIP-seq studies and GCCORRECT (Benjamini & Speed 2012) for other DNA sequencing in general.

High-throughput sequencing is still an emerging field; as our understanding of the different components of HTS experiments improves, previously unknown biases and artefacts will be characterized, and methods for correction will be developed accordingly. For example, a new type of artefact has been reported recently in deep coverage-targeted capture sequencing data; oxidative contaminants in DNA samples can induce DNA oxidation events during acoustic shearing which can later become transversion mutations (Costello *et al.* 2013). If uncharacterized, these artefactual mutations can be mistakenly reported as true biological events.

'Infectious pathology': library contamination

The aforementioned 'pathologies' largely stem from HTS-specific aspects of how experiments are performed and how sequence read data are generated. In addition, there are some old challenges which have become even more troublesome in the new context of HTS. One such example is contamination, where the data are 'infected' with sequences from undesired sources. Due to the unprecedented sequencing depths achieved by HTS technologies as well as because of their abilities to sequence directly pools of DNA fragments, HTS experiments are more likely to contain contaminant sequences than traditional Sanger experiments. Contamination can result from various reasons, including imperfect sample collection, human contamination and mix-up of samples. Factors such as close associations between organisms (e.g. hosts and pathogens or symbiotic microbes) also add to the complexity of the problem. Therefore, while it is possible to minimize contamination through careful experimental procedures or to enrich DNA from targeted sources (e.g. the approach to capture ancient human DNA which often comprises <1% of the DNA in specimens (Carpenter *et al.* 2013)), HTS read data should not

be assumed to be contamination free in general. In fact, contamination has been found to be prevalent in genome-scale studies; recent surveys have revealed widespread human DNA contamination in genome databases (Longo *et al.* 2011) and metagenomic data sets (Schmieder & Edwards 2011a). Such contaminant sequences should be removed to the degree possible because their inclusion will introduce noise into downstream analysis and may lead to incorrect conclusions [e.g. (Alkan *et al.* 2011b; Laurin-Lemay *et al.* 2012)].

The identification and removal of contamination from HTS data is particularly challenging due to the immense volume of generated sequence reads. Traditionally, sequence reads are compared against comprehensive databases (e.g. NCBI nonredundant nucleotide/protein database) to determine their sources of origin. Such an approach is computationally challenging for HTS data because of the great volume of sequence read data, typically hundreds or thousands of millions of sequence reads, which need to be examined. The computational burden can be greatly reduced if contamination sources are known; this knowledge can narrow the search space to a few species and allow for the use of tools optimized for mapping HTS reads against reference genomes. However, such information is not always available *a priori*, and other contamination may exist in addition to the 'known' one(s). In such cases, possible contamination sources can be inferred from the data. For example, QC-CHAIN (Zhou *et al.* 2013) delineates species composition based on ribosomal DNA sequences (see Fig. 3 for an example) and also estimates the relative abundance of contamination by comparing a random subset of reads against the NCBI nonredundant nucleotide database.

Once the sources of contamination are determined, subsequent decontamination can be carried out using DECONSEQ (Schmieder & Edwards 2011a). When reference sequences from the sources of contamination are not available (e.g. when the actual contaminant sequences are divergent from available reference sequences used to identify the contamination in the first place), a strategy based on searching for contamination after sequence read assembly may be preferable for better accuracy (Kostic *et al.* 2011; Leese *et al.* 2012); in other words, a *de novo* assembly of all the data is constructed first and then assembled contigs are searched against a nonredundant database to filter contigs that appear to be contaminants. This approach could be further augmented by incorporating information on the GC content and read coverage of assembled contigs (Kumar *et al.* 2013). Finally, in cases where only prokaryotic sequences are of interest, eukaryotic contaminants can be efficiently identified by using tools that evaluate sequence composition instead of similarity and *vice versa* [e.g. EU-DETECT (Mohammed *et al.* 2011)].

'Inherited pathology': complex genomes

Genome sequencing is one of the most powerful applications of HTS, and thousands of new genomes have been

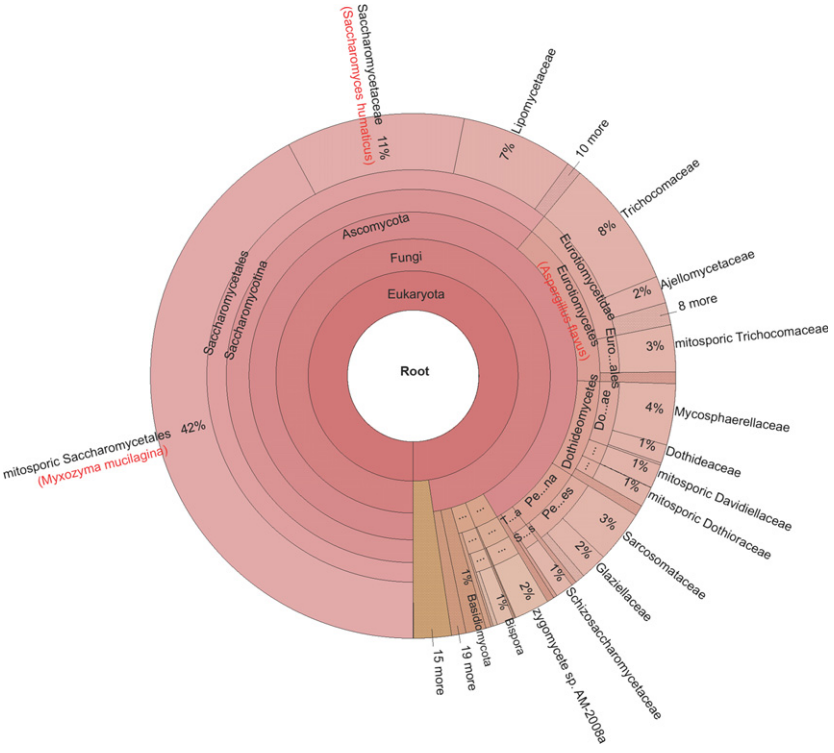

Fig. 3 Diagnosis of potential contamination in high-throughput sequencing sequence read data set. The plot generated by QC-CHAIN shows the taxonomic distribution of rDNA sequences identified in a fungal genomic sequencing data set. Besides the target species (*Myxozyma mucilagina*), this data set also contains sequences from other fungi, likely including species sequenced on the same lane (*Saccharomyces humaticus* and *Aspergillus flavus*), indicating potential cross-contamination during sample preparation.

successfully sequenced and assembled based on HTS data in the last few years (Pagani *et al.* 2012). However, all genomes are not equal; *de novo* assembly is inherently challenging for 'complex' genomes and has become even more so when the shorter sequence reads generated by HTS technologies are used. Features such as high repeat content, heterozygosity and polyploidy often make assembly difficult and give rise to assemblies that are highly fragmented or downright erroneous, often requiring custom study designs and sequencing strategies. For example, multiple rounds of inbreeding can reduce the level of heterozygosity (Hirsch & Buell 2013), while a combination of longer reads and PE reads with longer insert size [e.g. mate-pair (MP) reads with insert sizes of several Kbp] can improve the assembly of repetitive regions (Treangen & Salzberg 2012). In particular, promising results have recently been obtained by using the combination of Illumina PE/MP reads and longer reads generated by 454 and PacBio (Koren *et al.* 2012), or even by using the PacBio reads alone (Koren *et al.* 2013), to resolve complex repeat regions in bacterial and eukaryotic genomes.

These idiosyncrasies, which pose great challenges in designing algorithms for HTS data analysis, are inherent to the genomes of the sequenced organisms and cannot be said to be artefacts of the HTS data *per se*. However, these idiosyncrasies can often be learned from the data so that they better inform subsequent data analyses and the interpretation of results. The 'PREQC' component of SGA assembler (Simpson 2013) is a handy tool for this purpose; it can estimate the levels of heterozygosity, repetitive sequences and sequencing errors from sequence read data, as well as generate other informative plots on data properties to further facilitate genome assembly (see Fig. 4 for examples). If known ahead of time, highly heterozygous genomes can yield better assemblies with specialized programs [e.g. HAPSEMBLER (Donmez & Brudno 2011)], whereas assemblies of highly repetitive genomes can (and should) be examined for misassembly by using evaluation tools such as REAPR (Hunt *et al.* 2013).

'Fitness'

Besides the diagnosis and treatment of various pathologies of HTS data, there are also procedures that can be incorporated during the data preprocessing stage that improve data quality and boost downstream analysis, that is, to improve the 'fitness' of the data (Magoc & Salzberg 2011; Liu *et al.* 2012; Pell *et al.* 2012; Titus Brown *et al.* 2012). Here, we briefly introduce two of them.

Digital normalization

In *de novo* studies of transcriptome and metagenome where the relative abundance of transcripts and species can be extremely uneven, HTS experiments are usually performed at very high sequencing depth in order to achieve a comprehensive representation of the underlying sequence repertoire. However, highly abundant sequences, which can be readily covered at lower sequencing depths and are extremely well covered at higher sequencing depths, substantially add to the computational burden of *de novo* assembly. To overcome this burden, a 'digital normalization' approach was developed that reduces the coverage of highly abundant sequences to an even level (Titus Brown *et al.* 2012). This procedure greatly reduces the size of HTS data sets by mostly removing redundancy and sequencing errors

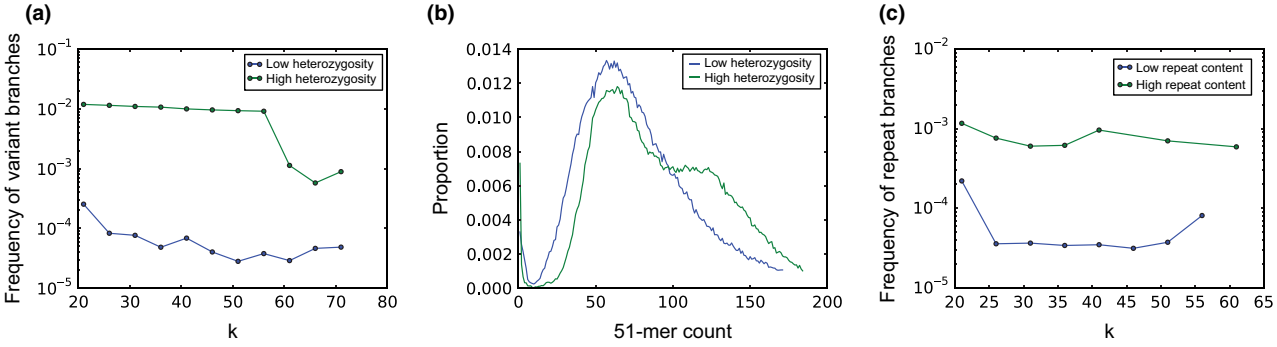

Fig. 4 Evaluation of genome 'complexity' from high-throughput sequencing sequence read data. The plots were generated by PREQC to evaluate the level of heterozygosity and repeat content. (a) Frequency of variant branches in the *de Bruijn* graphs constructed using different *k*-mer values. High frequency of variant branches indicates high level of heterozygosity. (b) Distribution of the counts of 51 mers. While the genome with low heterozygosity shows only one peak in the distribution, the highly heterozygous genome has an extra peak representing 51 mers (likely derived from heterozygous regions) whose counts are twice of that of the main peak. (c) Frequency of repeat branches in the *de Bruijn* graphs constructed using different *k*-mer values. High frequency of repeat branches indicates high level of repeat content.

contained in these redundant sequences but retaining real information, thus leading to much improved computational performance and largely comparable assembly quality (Titus Brown *et al.* 2012). For example, in an analysis of mouse transcriptome data, digital normalization removed 90.7% of the sequence reads containing virtually all (more than 99.99%) of the errors (Titus Brown *et al.* 2012). Digital normalization is a widely applicable, promising solution for the ever increasing size of HTS data sets, and has been integrated in TRINITY (Haas *et al.* 2013), which is one of the most popular *de novo* transcriptome assemblers.

Overlapping paired-end/mate-pair reads

One major limitation of many HTS technologies, including Illumina, is that their shorter sequence read lengths contribute to the difficulty of *de novo* assembly using HTS data. While the original sequence read length is determined by the sequencing technology being used, it is possible to obtain longer sequence reads computationally from PE data. This can be accomplished by connecting paired reads into a single sequence if they overlap with each other, which happens when the insert size is smaller than the summed length of both reads. Overlapping PE reads are highly useful in *de novo* studies; for example, they are required by the ALLPATHS-LG assembler (Ribeiro *et al.* 2012) to enable high-quality genome assembly. There are also stand-alone programs for merging PE reads, including aforementioned multipurpose QC tools (e.g. ADATPERREMOVAL, EA-UTILS and SEQPREP) and dedicated ones [e.g. SHERA (Rodrigue *et al.* 2010), FLASH (Magoc & Salzberg 2011), COPE (Liu *et al.* 2012), PANDASEQ (Masella *et al.* 2012), XORRO (Dickson & Gloor

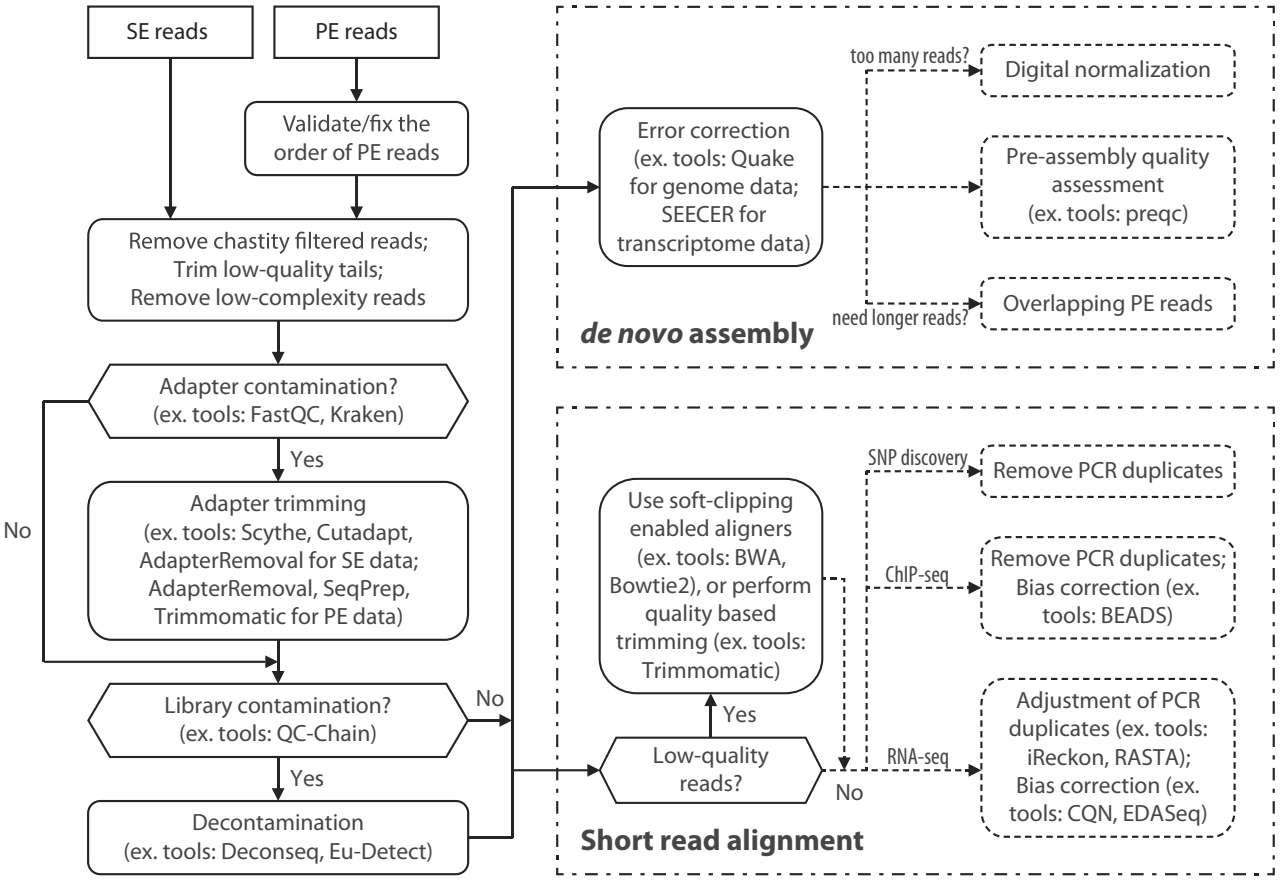

Fig. 5 A suggested workflow for quality control (QC) of high-throughput sequencing (HTS) sequence read data. The first few steps of this workflow (shown on the left side) summarize QC procedures that are likely relevant to most HTS data sets, including the validation of PE reads order, the removal of extremely low-quality and artefactual reads, adapter trimming and the removal of library contamination. The steps shown on the right side summarize QC procedures recommended for *de novo* assembly studies as well as QC procedures recommended for studies based on short read alignment (e.g. SNP calling, ChIP-seq and RNA-seq). Rounded rectangles represent QC procedures, hexagons represent decision-making steps, and dotted lines represent QC procedures that are optional or applicable to only certain types of studies. Example tools for various QC procedures are listed in brackets.

2013) and PEAR (Zhang *et al.* 2013)]. Alternatively, multiple read pairs can be merged together if overlap can be found between pairs, irrespective of whether reads overlap within each pair (Silver *et al.* 2013); this approach can thus be applied to MP data. In a more aggressive approach, sequences of the original fragments that give rise to each read pair can be inferred by mapping PE/MP reads to a preliminary assembly generated from the original sequence reads (Liu *et al.* 2013a).

Conclusion

Quality control of sequence read data generated by HTS technologies is an essential yet somewhat overlooked component of data analysis. In this review, we have outlined several of what we think ought to be standard QC procedures as well as major types of HTS data pathologies, and discussed currently available approaches for diagnosis and treatment. We hope this work will provide a useful guide for researchers working with HTS data, and also facilitate the future development of best practice guides for HTS data QC. To this end, we have constructed a workflow that summarizes both the QC steps that are common to HTS data in general and ones that are more relevant to certain types of studies (Fig. 5).

In the immediate future, greater efforts are needed in several directions. For example, systematic evaluations of tools with similar capabilities that build on the ones we have provided in the Boxes of this review would greatly help researchers better choose the appropriate one in their studies and aid in the identification of treatments that most improve data quality. In the context of integrated pipeline construction, it is also important to improve our understanding of how different combinations of QC procedures affect downstream analyses. Finally, perhaps the greatest challenge faced in efforts to improve sequence read data quality is the amazing rapidity with which the HTS field, and its associated data pathologies, is changing; new QC measurements will be needed to accommodate the new challenges that would accompany the development of HTS technologies. As molecular ecologists fully embrace the power of HTS technologies, our field will be well served to once again heed the Red Queen's advice to Alice: 'Now, here, you see, it takes all the running you can do, to keep in the same place. If you want to get somewhere else, you must run at least twice as fast as that!'

Acknowledgements

This work was conducted in part using the resources of the Advanced Computing Center for Research and Education at Vanderbilt University. This work was supported by funding to AR from the National Science Foundation Grant DEB-0844968 and by the March of Dimes.

References

Adey A, Morrison HG, Asan *et al.* (2010) Rapid, low-input, low-bias construction of shotgun fragment libraries by high-density *in vitro* transposition. *Genome Biology*, **11**, R119.

Aird D, Ross MG, Chen WS *et al.* (2011) Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. *Genome Biology*, **12**, R18.

Alkan C, Coe BP, Eichler EE (2011a) Genome structural variation discovery and genotyping. *Nature Reviews Genetics*, **12**, 363–376.

Alkan C, Sajjadian S, Eichler EE (2011b) Limitations of next-generation genome sequence assembly. *Nature Methods*, **8**, 61–65.

Aronesty E (2013) Comparison of sequencing utility programs. *The Open Bioinformatics Journal*, **7**, 1–8.

Bainbridge MN, Wang M, Burgess DL *et al.* (2010) Whole exome capture in solution with 3 Gbp of data. *Genome Biology*, **11**, R62.

Balzer S, Malde K, Grohme MA, Jonassen I (2013) Filtering duplicate reads from 454 pyrosequencing data. *Bioinformatics*, **29**, 830–836.

Baumann DD, Doerge RW (2013) Robust adjustment of sequence tag abundance. *Bioinformatics*. doi: 10.1093/bioinformatics/btt575.

Benjamini Y, Speed TP (2012) Summarizing and correcting the GC content bias in high-throughput sequencing. *Nucleic Acids Research*, **40**, e72.

Bokulich NA, Subramanian S, Faith JJ *et al.* (2013) Quality-filtering vastly improves diversity estimates from Illumina amplicon sequencing. *Nature Methods*, **10**, 57–59.

Breese MR, Liu Y (2013) NGSUtils: a software suite for analyzing and manipulating next-generation sequencing datasets. *Bioinformatics*, **29**, 494–496.

Burriesci MS, Lehnert EM, Pringle JR (2012) Fulcrum: condensing redundant reads from high-throughput sequencing studies. *Bioinformatics*, **28**, 1324–1327.

Carpenter ML, Buenrostro JD, Valdiosera C *et al.* (2013) Pulling out the 1%: whole-genome capture for the targeted enrichment of ancient DNA sequencing libraries. *American Journal of Human Genetics*, **93**, 852–864.

Chen Y, Negre N, Li Q *et al.* (2012) Systematic evaluation of factors influencing ChIP-seq fidelity. *Nature Methods*, **9**, 609–614.

Chen YC, Liu T, Yu CH, Chiang TY, Hwang CC (2013) Effects of GC bias in next-generation-sequencing data on *de novo* genome assembly. *PLoS One*, **8**, e62856.

Cheung MS, Down TA, Latorre I, Ahringer J (2011) Systematic bias in high-throughput sequencing data and its correction by BEADS. *Nucleic Acids Research*, **39**, e103.

Cock PJ, Fields CJ, Goto N, Heuer ML, Rice PM (2010) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Research*, **38**, 1767–1771.

Costello M, Pugh TJ, Fennell TJ *et al.* (2013) Discovery and characterization of artifactual mutations in deep coverage targeted capture sequencing data due to oxidative DNA damage during sample preparation. *Nucleic Acids Research*, **41**, e67.

Cox MP, Peterson DA, Biggs PJ (2010) SolexaQA: at-a-glance quality assessment of Illumina second-generation sequencing data. *BMC Bioinformatics*, **11**, 485.

Criscuolo A, Brisse S (2013) AlienTrimmer: a tool to quickly and accurately trim off multiple short contaminant sequences from high-throughput sequencing reads. *Genomics*, **102**, 500–506.

Davis MP, van Dongen S, Abreu-Goodger C, Bartonicek N, Enright AJ (2013) Kraken: a set of tools for quality control and analysis of high-throughput sequence data. *Methods*, **63**, 41–49.

De Wit P, Pespeni MH, Ladner JT *et al.* (2012) The simple fool's guide to population genomics via RNA-Seq: an introduction to high-throughput sequencing data analysis. *Molecular Ecology Resources*, **12**, 1058–1067.

Delmont TO, Simonet P, Vogel TM (2013) Mastering methodological pitfalls for surviving the metagenomic jungle. *BioEssays*, **35**, 744–754.

DeLuca DS, Levin JZ, Sivachenko A *et al.* (2012) RNA-SeQC: RNA-seq metrics for quality control and process optimization. *Bioinformatics*, **28**, 1530–1532.

DePristo MA, Banks E, Poplin R *et al.* (2011) A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, **43**, 491–498.

Dewoody JA, Abts KC, Fahey AL *et al.* (2013) Of contigs and quagmires: next-generation sequencing pitfalls associated with transcriptomic studies. *Molecular Ecology Resources*, **13**, 551–558.

Dickson RJ, Gloor GB (2013) XORRO: rapid paired-end read overlapper. *ArXiv e-Prints*, p. 4620, arXiv:1304.4620 [q-bio.GN].

Dillies MA, Rau A, Aubert J *et al.* (2012) A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in Bioinformatics*, **14**, 671–683.

Dodt M, Roehr J, Ahmed R, Dieterich C (2012) FLEXBAR—flexible barcode and adapter processing for next-generation sequencing platforms. *Biology*, **1**, 895–905.

Dohm JC, Lottaz C, Borodina T, Himmelbauer H (2008) Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Research*, **36**, e105.

Dolan PC, Denver DR (2008) TileQC: a system for tile-based quality control of Solexa data. *BMC Bioinformatics*, **9**, 250.

Donmez N, Brudno M (2011) Hapsembler: an assembler for highly polymorphic genomes. In: *Research in Computational Molecular Biology* (eds Bafna V, Sahinalp SC), pp. 38–52. Springer, Berlin/Heidelberg.

Egan AN, Schlueter J, Spooner DM (2012) Applications of next-generation sequencing in plant biology. *American Journal of Botany*, **99**, 175–185.

Eisenstein M (2012) Oxford Nanopore announcement sets sequencing sector abuzz. *Nature Biotechnology*, **30**, 295–296.

Ekblom R, Galindo J (2011) Applications of next generation sequencing in molecular ecology of non-model organisms. *Heredity*, **107**, 1–15.

Eren AM, Morrison HG, Huse SM, Sogin ML (2013) DRISEE overestimates errors in metagenomic sequencing data. *Briefings in Bioinformatics*. doi: 10.1093/bib/bbt010.

Gayral P, Weinert L, Chiari Y *et al.* (2011) Next-generation sequencing of transcriptomes: a guide to RNA isolation in nonmodel animals. *Molecular Ecology Resources*, **11**, 650–661.

Glenn TC (2011) Field guide to next-generation DNA sequencers. *Molecular Ecology Resources*, **11**, 759–769.

Goecks J, Nekrutenko A, Taylor J, Galaxy T (2010) Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology*, **11**, R86.

Grabherr MG, Haas BJ, Yassour M *et al.* (2011) Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature Biotechnology*, **29**, 644–652.

Haas BJ, Papanicolaou A, Yassour M *et al.* (2013) *De novo* transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis. *Nature Protocols*, **8**, 1494–1512.

Hansen KD, Brenner SE, Dudoit S (2010) Biases in Illumina transcriptome sequencing caused by random hexamer priming. *Nucleic Acids Research*, **38**, e131.

Hansen KD, Irizarry RA, Wu ZJ (2012) Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics*, **13**, 204–216.

van Heesch S, Mokry M, Boskova V *et al.* (2013) Systematic biases in DNA copy number originate from isolation procedures. *Genome Biology*, **14**, R33.

Hirsch CN, Buell CR (2013) Tapping the promise of genomics in species with complex, nonmodel genomes. *Annual Review of Plant Biology*, **64**, 89–110.

Hu X, Yuan J, Shi Y *et al.* (2012) pIRS: profile-based Illumina pair-end reads simulator. *Bioinformatics*, **28**, 1533–1535.

Hudson ME (2008) Sequencing breakthroughs for genomic ecology and evolutionary biology. *Molecular Ecology Resources*, **8**, 3–17.

Hunt M, Kikuchi T, Sanders M *et al.* (2013) REAPR: a universal tool for genome assembly evaluation. *Genome Biology*, **14**, R47.

Ilie L, Molnar M (2013) RACER: rapid and accurate correction of errors in reads. *Bioinformatics*, **29**, 2490–2493.

Jerome M, Noirot C, Klopp C (2011) Assessment of replicate bias in 454 pyrosequencing and a multi-purpose read-filtering tool. *BMC Research Notes*, **4**, 149.

Keegan KP, Trimble WL, Wilkening J *et al.* (2012) A platform-independent method for detecting errors in metagenomic sequencing data: DRISEE. *PLoS Computational Biology*, **8**, e1002541.

Kelley DR, Schatz MC, Salzberg SL (2010) Quake: quality-aware detection and correction of sequencing errors. *Genome Biology*, **11**, R116.

Kim D, Pertea G, Trapnell C *et al.* (2013) TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology*, **14**, R36.

Kircher M, Heyn P, Kelso J (2011) Addressing challenges in the production and analysis of illumina sequencing data. *BMC Genomics*, **12**, 382.

Kong Y (2011) Btrim: a fast, lightweight adapter and quality trimming program for next-generation sequencing technologies. *Genomics*, **98**, 152–153.

Koren S, Schatz MC, Walenz BP *et al.* (2012) Hybrid error correction and *de novo* assembly of single-molecule sequencing reads. *Nature Biotechnology*, **30**, 693–700.

Koren S, Harhay GP, Smith TP *et al.* (2013) Reducing assembly complexity of microbial genomes with single-molecule sequencing. *Genome Biology*, **14**, R101.

Kostic AD, Ojesina AI, Pedamallu CS *et al.* (2011) PathSeq: software to identify or discover microbes by deep sequencing of human tissue. *Nature Biotechnology*, **29**, 393–396.

Kozarewa I, Ning Z, Quail MA *et al.* (2009) Amplification-free Illumina sequencing-library preparation facilitates improved mapping and assembly of (G+C)-biased genomes. *Nature Methods*, **6**, 291–295.

Kumar S, Jones M, Koutsovoulos G, Clarke M, Blaxter M (2013) Blobology: exploring raw genome data for contaminants, symbionts and parasites using taxon-annotated GC-coverage plots. *Frontiers in Genetics*, **4**, 237.

Kwak H, Fuda NJ, Core LJ, Lis JT (2013) Precise maps of RNA polymerase reveal how promoters direct initiation and pausing. *Science*, **339**, 950–953.

Langmead B, Salzberg SL (2012) Fast gapped-read alignment with Bowtie 2. *Nature Methods*, **9**, 357–359.

Lassmann T, Hayashizaki Y, Daub CO (2009) TagDust–a program to eliminate artifacts from next generation sequencing data. *Bioinformatics*, **25**, 2839–2840.

Laurin-Lemay S, Brinkmann H, Philippe H (2012) Origin of land plants revisited in the light of sequence contamination and missing data. *Current Biology*, **22**, R593–R594.

Le HS, Schulz MH, McCauley BM, Hinman VF, Bar-Joseph Z (2013) Probabilistic error correction for RNA sequencing. *Nucleic Acids Research*, **41**, e109.

Leese F, Brand P, Rozenberg A *et al.* (2012) Exploring Pandora's box: potential and pitfalls of low coverage genome surveys for evolutionary biology. *PLoS One*, **7**, e49202.

Leggett RM, Clavijo BJ, Clissold L, Clark MD, Caccamo M (2013) NextClip: an analysis and read preparation tool for Nextera Long Mate Pair libraries. *Bioinformatics*. doi: 10.1093/bioinformatics/btt702.

Li H, Durbin R (2010) Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, **26**, 589–595.

Li H, Handsaker B, Wysoker A *et al.* (2009) The sequence alignment/map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

Li J, Jiang H, Wong WH (2010a) Modeling non-uniformity in short-read rates in RNA-Seq data. *Genome Biology*, **11**, R50.

Li Y, Hu Y, Bolund L, Wang J (2010b) State of the art *de novo* assembly of human genomes from massively parallel sequencing data. *Human Genomics*, **4**, 271–277.

Li JW, Schmieder R, Ward RM *et al.* (2012) SEQanswers: an open access community for collaboratively decoding genomes. *Bioinformatics*, **28**, 1272–1273.

Lindgreen S (2012) AdapterRemoval: easy cleaning of next-generation sequencing reads. *BMC Research Notes*, **5**, 337.

Liu B, Yuan J, Yiu SM *et al.* (2012) COPE: an accurate k-mer-based pair-end reads connection tool to facilitate genome assembly. *Bioinformatics*, **28**, 2870–2874.

Liu T, Tsai CH, Lee WB, Chiang JH (2013a) Optimizing information in next-generation-sequencing (NGS) reads for improving *de novo* genome assembly. *PLoS One*, **8**, e69503.

Liu Y, Schroder J, Schmidt B (2013b) Musket: a multistage k-mer spectrum-based error corrector for Illumina sequence data. *Bioinformatics*, **29**, 308–315.

Lohse M, Bolger AM, Nagel A *et al.* (2012) RobiNA: a user-friendly, integrated software solution for RNA-Seq-based transcriptomics. *Nucleic Acids Research*, **40**, W622–W627.

Loman NJ, Misra RV, Dallman TJ *et al.* (2012) Performance comparison of benchtop high-throughput sequencing platforms. *Nature Biotechnology*, **30**, 434–439.

Longo MS, O'Neill MJ, O'Neill RJ (2011) Abundant human DNA contamination identified in non-primate genome databases. *PLoS One*, **6**, e16410.

MacManes MD (2014) On the optimal trimming of high-throughput mRNA sequence data. *Frontiers in Genetics*, **5**, 13.

MacManes MD, Eisen MB (2013) Improving transcriptome assembly through error correction of high-throughput sequence reads. *PeerJ*, **1**, e113.

Magoc T, Salzberg SL (2011) FLASH: fast length adjustment of short reads to improve genome assemblies. *Bioinformatics*, **27**, 2957–2963.

Maher CA, Palanisamy N, Brenner JC *et al.* (2009) Chimeric transcript discovery by paired-end transcriptome sequencing. *Proceedings of the National Academy of Sciences of the United States of America*, **106**, 12353–12358.

Mamanova L, Andrews RM, James KD *et al.* (2010) FRT-seq: amplification-free, strand-specific transcriptome sequencing. *Nature Methods*, **7**, 130–132.

Marcais G, Yorke JA, Zimin A (2013) QuorUM: an error corrector for Illumina reads. *ArXiv e-Prints*, p. 3515, arXiv: 1307.3515 [q-bio.GN].

Mardis ER (2008) The impact of next-generation sequencing technology on genetics. *Trends in Genetics*, **24**, 133–141.

Mardis ER (2013) Next-generation sequencing platforms. *Annual Review of Analytical Chemistry*, **6**, 287–303.

Martin M (2011) *Cutadapt removes adapter sequences from high-throughput sequencing reads*.

Martin JA, Wang Z (2011) Next-generation transcriptome assembly. *Nature Reviews Genetics*, **12**, 671–682.

Masella AP, Bartram AK, Truszkowski JM, Brown DG, Neufeld JD (2012) PANDAseq: paired-end assembler for illumina sequences. *BMC Bioinformatics*, **13**, 31.

Mezlini AM, Smith EJ, Fiume M *et al.* (2013) iReckon: simultaneous isoform discovery and abundance estimation from RNA-seq data. *Genome Research*, **23**, 519–529.

Minoche AE, Dohm JC, Himmelbauer H (2011) Evaluation of genomic high-throughput sequencing data generated on Illumina HiSeq and genome analyzer systems. *Genome Biology*, **12**, R112.

Mohammed MH, Chadaram S, Komanduri D, Ghosh TS, Mande SS (2011) Eu-Detect: an algorithm for detecting eukaryotic sequences in metagenomic data sets. *Journal of Biosciences*, **36**, 709–717.

Nikolenko SI, Korobeynikov AI, Alekseyev MA (2013) BayesHammer: Bayesian clustering for error correction in single-cell sequencing. *BMC Genomics*, **14** (Suppl 1), S7.

Oshlack A, Robinson MD, Young MD (2010) From RNA-seq reads to differential expression results. *Genome Biology*, **11**, 220.

Pagani I, Liolios K, Jansson J *et al.* (2012) The Genomes OnLine Database (GOLD) v. 4: status of genomic and metagenomic projects and their associated metadata. *Nucleic Acids Research*, **40**, D571–D579.

Parnell LD, Lindenbaum P, Shameer K *et al.* (2011) BioStar: an online question & answer resource for the bioinformatics community. *PLoS Computational Biology*, **7**, e1002216.

Paszkiewicz K, Studholme DJ (2010) *De novo* assembly of short sequence reads. *Briefings in Bioinformatics*, **11**, 457–472.

Patel RK, Jain M (2012) NGS QC Toolkit: a toolkit for quality control of next generation sequencing data. *PLoS One*, **7**, e30619.

Pell J, Hintze A, Canino-Koning R *et al.* (2012) Scaling metagenome sequence assembly with probabilistic de Bruijn graphs.

Proceedings of the National Academy of Sciences of the United States of America, **109**, 13272–13277.
Pickrell JK, Marioni JC, Pai AA *et al.* (2010) Understanding mechanisms underlying human gene expression variation with RNA sequencing. *Nature*, **464**, 768–772.
Quail MA, Smith M, Coupland P *et al.* (2012) A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers. *BMC Genomics*, **13**, 341.
Ribeiro FJ, Przybylski D, Yin S *et al.* (2012) Finished bacterial genomes from shotgun sequence data. *Genome Research*, **22**, 2270–2277.
Rieber N, Zapatka M, Lasitschka B *et al.* (2013) Coverage bias and sensitivity of variant calling for four whole-genome sequencing technologies. *PLoS One*, **8**, e66621.
Risso D, Schwartz K, Sherlock G, Dudoit S (2011) GC-content normalization for RNA-Seq data. *BMC Bioinformatics*, **12**, 480.
Roberts A, Trapnell C, Donaghey J, Rinn JL, Pachter L (2011) Improving RNA-Seq expression estimates by correcting for fragment bias. *Genome Biology*, **12**, R22.
Rodrigue S, Materna AC, Timberlake SC *et al.* (2010) Unlocking short read sequencing for metagenomics. *PLoS One*, **5**, e11840.
Rokas A, Abbot P (2009) Harnessing genomics for evolutionary insights. *Trends in Ecology & Evolution*, **24**, 192–200.
Salzberg SL, Phillippy AM, Zimin A *et al.* (2012) GAGE: a critical evaluation of genome assemblies and assembly algorithms. *Genome Research*, **22**, 557–567.
Schmieder R, Edwards R (2011a) Fast identification and removal of sequence contamination from genomic and metagenomic datasets. *PLoS One*, **6**, e17288.
Schmieder R, Edwards R (2011b) Quality control and preprocessing of metagenomic datasets. *Bioinformatics*, **27**, 863–864.
Schuster SC (2008) Next-generation sequencing transforms today's biology. *Nature Methods*, **5**, 16–18.
Shendure J, Ji H (2008) Next-generation DNA sequencing. *Nature Biotechnology*, **26**, 1135–1145.
Shendure J, Lieberman Aiden E (2012) The expanding scope of DNA sequencing. *Nature Biotechnology*, **30**, 1084–1094.
Shiroguchi K, Jia TZ, Sims PA, Xie XS (2012) Digital RNA sequencing minimizes sequence-dependent bias and amplification noise with optimized single-molecule barcodes. *Proceedings of the National Academy of Sciences of the United States of America*, **109**, 1347–1352.
Silver DH, Ben-Elazar S, Bogoslavsky A, Yanai I (2013) ELOPER: elongation of paired-end reads as a pre-processing tool for improved *de novo* genome assembly. *Bioinformatics*, **29**, 1455–1457.
Simpson JT (2013) Exploring genome characteristics and sequence quality without a reference. *Bioinformatics*. doi: 10.1093/bioinformatics/btu023.
Sleep JA, Schreiber AW, Baumann U (2013) Sequencing error correction without a reference genome. *BMC Bioinformatics*, **14**, 367.
Smeds L, Kunstner A (2011) ConDeTri–a content dependent read trimmer for Illumina data. *PLoS One*, **6**, e26314.
Taub MA, Corrada Bravo H, Irizarry RA (2010) Overcoming bias and systematic errors in next generation sequencing data. *Genome Medicine*, **2**, 87.
Tautz D, Ellegren H, Weigel D (2010) Next generation molecular ecology. *Molecular Ecology* **19** (Suppl 1), 1–3.
Teo SM, Pawitan Y, Ku CS, Chia KS, Salim A (2012) Statistical challenges associated with detecting copy number variations with next-generation sequencing. *Bioinformatics*, **28**, 2711–2718.
Titus Brown C, Howe A, Zhang Q, Pyrkosz AB, Brom TH (2012) A reference-free algorithm for computational normalization of shotgun sequencing data. *ArXiv e-Prints*, p. 4802, arXiv:1203.4802 [q-bio.GN].
Treangen TJ, Salzberg SL (2012) Repetitive DNA and next-generation sequencing: computational challenges and solutions. *Nature Reviews Genetics*, **13**, 36–46.
Wang Z, Gerstein M, Snyder M (2009) RNA-Seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, **10**, 57–63.
Werner T (2010) Next generation sequencing in functional genomics. *Briefings in Bioinformatics*, **11**, 499–511.
Wolf JB (2013) Principles of transcriptome analysis and gene expression quantification: an RNA-seq tutorial. *Molecular Ecology Resources*, **13**, 559–572.
Xie Y, Wu G, Tang J *et al.* (2013) SOAPdenovo-trans: *de novo* transcriptome assembly with short RNA-Seq reads. *ArXiv e-Prints*, p. 6760, arXiv:1305.6760 [q-bio.GN].
Xu H, Luo X, Qian J *et al.* (2012) FastUniq: a fast *de novo* duplicates removal tool for paired short reads. *PLoS One*, **7**, e52249.
Yang X, Chockalingam SP, Aluru S (2013a) A survey of error-correction methods for next-generation sequencing. *Briefings in Bioinformatics*, **14**, 56–66.
Yang X, Liu D, Liu F *et al.* (2013b) HTQC: a fast quality control toolkit for Illumina sequencing data. *BMC Bioinformatics*, **14**, 33.
Zhang G, Guo G, Hu X *et al.* (2010) Deep RNA sequencing at single base-pair resolution reveals high complexity of the rice transcriptome. *Genome Research*, **20**, 646–654.
Zhang T, Luo Y, Liu K *et al.* (2011) BIGpre: a quality assessment package for next-generation sequencing data. *Genomics Proteomics & Bioinformatics*, **9**, 238–244.
Zhang J, Kobert K, Flouri T, Stamatakis A (2013) PEAR: a fast and accurate Illumina Paired-End reAd mergeR. *Bioinformatics*. doi: 10.1093/bioinformatics/btt593.
Zhou Q, Su X, Wang A, Xu J, Ning K (2013) QC-Chain: fast and holistic quality control method for next-generation sequencing data. *PLoS One*, **8**, e60234.

X. Z. and A. R. conceived this study, analysed the data, and wrote the article.

Data accessibility

The high-quality Illumina short read data set used for the illustration in Figure 1 and evaluation in Box 1, the simulated Illumina short read data sets used for the evaluation of adapter trimming tools in Box 3, the Illumina short read data set used for the illustration in Figure 3 and commands used for all QC tool evaluations performed in this review are available at: Dryad doi: 10.5061/dryad.h988s.