2

Quality Control and Data Preprocessing

Stuart M. Brown

It is an essential part of any next-generation sequencing (NGS) experiment or core laboratory workflow to assess the quality of the data produced by each run of the sequencing machine. Various quality metrics may be assessed at the run, flowcell, lane, or sample level; these metrics provide information and may detect problems related to sample preparation, multiplexing, mixing, and sample loading in each lane, as well as the chemistry and optics of the sequencing machine. Some quality control (QC) metrics may identify problems that can be corrected or ameliorated by data cleaning procedures.

Run and lane level information is produced by the sequencing machine during the run and may be assessed with tools such as the Illumina Real-Time Analysis (RTA) and Sequencing Analysis Viewer (SAV) applications:

> Real Time Analysis runs locally on the instrument control computer and performs base call and quality scoring. The analysis is performed during the chemistry and imaging cycles of a sequencing run, which saves downstream analysis time and allows the operator to quickly decide whether or not the run is progressing as expected. . . . The Sequencing Analysis Viewer is an application that allows real-time views of important quality metrics generated by the real-time analysis software.[1]

The SAV provides information about cluster density and the base quality per cycle in each lane of the flow cell (see Fig. 1). Additional internal metrics are provided such as signal intensity, signal-to-noise ratio, percent of clusters for which a base is called, percent of base calls that pass filter, phasing and prephasing (percent of molecules in a cluster for which the addition of bases falls behind or jumps ahead of the current cycle), and an error rate computed from **alignment** of reads from a spike-in PhiX control.

[1] From Illumina RTA software support (http://support.illumina.com/sequencing/sequencing_software/real-time_analysis_rta.html).

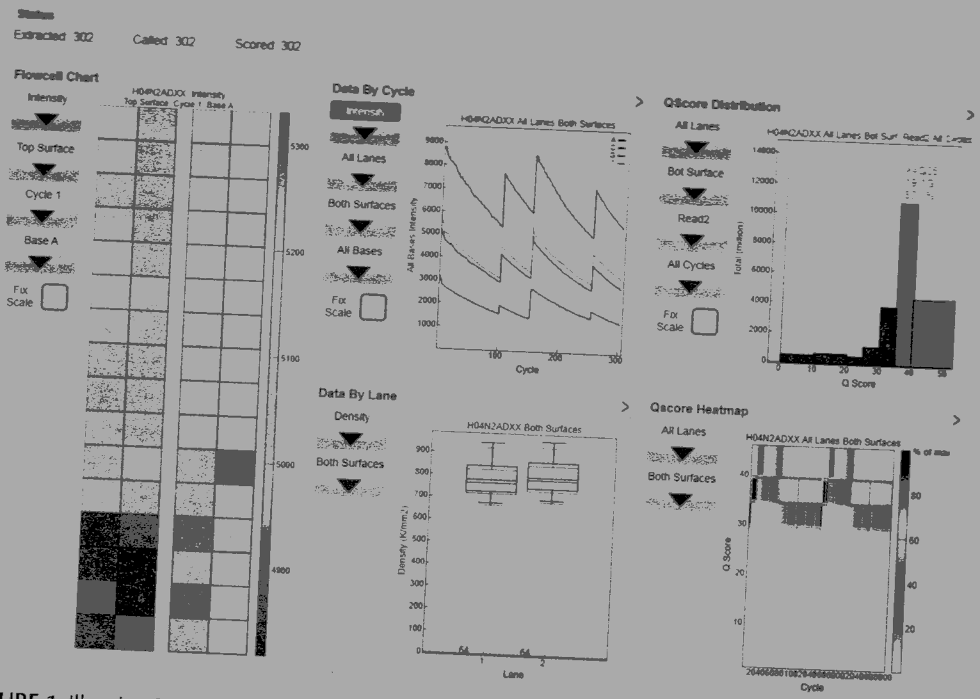


FIGURE 1. Illumina SAV display showing a currently running sequencer run with color-coded representations of progress on individual tiles within a flow cell, cluster density, per cycle base composition, and quality score distributions.

Illumina has developed its own Library QC workflow as part of the MiSeq Reporter software that summarizes overall run quality after the completion of the run. This method can be used with the MiSeq machine to quickly and inexpensively evaluate the quality of sequencing libraries before more extensive sequencing on HiSeq machines. Some statistics, such as number of clusters and percentage of pass-filter reads, overlap with the metrics provided by SAV. The MiSeq software computes many metrics based on alignment with BWA (Li and Durbin 2010) to a **reference sequence** (using low sensitivity parameters for greater speed) including fragment size distribution, percentage of forward and reverse reads aligned to ~~the~~ reference, percentage of sequence duplicates (reported as library diversity), mismatch rate (includes both sequencing error and single-nucleotide polymorphisms [SNPs]), and **coverage** over the length of the reference sequence (primarily for polymerase chain reaction [PCR] **amplicon** sequencing targets).

MULTIPLEXING

Many NGS experiments do not require the full output of a "lane" or entire flow cell from a sequencing machine. Multiplexing is an experimental design that uses a bar coding scheme to tag individual samples with different adapter sequences, mix them together in a single sequencing library, and then use bioinformatics methods to sort out the samples by identifying the different bar codes in the output data file. Ideally, the bar codes for different mixed samples should all differ by more than one base, so that a single sequencing error does not result in the misclassification of a sample. For metagenomic experiments using the 454 sequencer, Hamady et al. (2008) developed a set of 1544 eight-base error-correcting bar codes based on Hamming distances. The bar codes were further optimized for GC content, eliminating consecutive triplets of the same base, and self-complementary sequences. Caporaso et al. (2011) adapted the bar code scheme for Illumina MySeq and HiSeq sequencers, publishing a list of 2167 12-base bar codes based on a Golay error-correcting **algorithm**, and filtered for self-complementarity.

Illumina has developed its own bar coding system, which has some novel features. Rather than including the bar code sequence at the end of the adpaters so that it becomes part of the **sequence read** adjacent to the cloned **DNA fragments**, the Illumina bar code index sequence is located a dozen or more bases upstream on the adapter. Illumina includes an entirely separate sequencing reaction with its own **sequencing primer** and reagent cycles to capture the sequence of the 8-base bar code, which is stored in a separate sequencing data file from the information captured from the insert DNA. For paired-end sequencing, it is possible to include two separate bar codes at each end ("dual indexed libraries"). Illumina currently (in 2014) sells kits that allow for a total of 96 different bar code combinations to be mixed in a single lane (12 at one end, 8 at the other). However, care must be taken when mixing a small number of bar-coded samples. The Illumina chemistry is sensitive to the sequence composition at each base position, requiring at least one sample with G/T (green laser) and one with A/C (red laser) at each base position of the bar code. The Illumina Experiment Manager (a stand-alone software tool to create sample sheets) will identify bad bar code combinations (see Figs. 2 and 3).

Illumina includes a demultiplexing option in both MiSeq Reporter and HiSeq CASAVA operating software. These methods allow for one base mismatch between

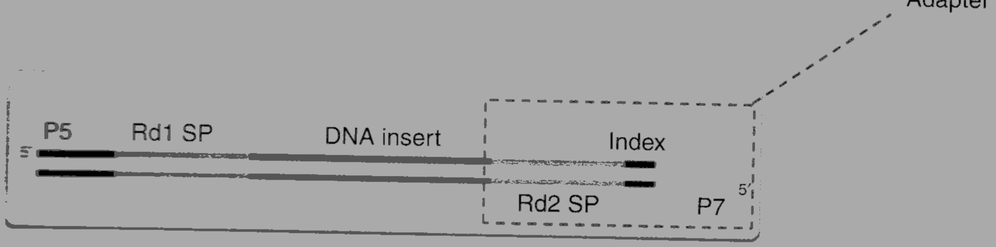


FIGURE 2. Illumina bar code multiplex system includes the bar code index within the adapter. (Used with permission from Illumina Inc. 2008.)

Good examples				Bad examples			
Index 1		Index 2		Index 1		Index 2	
705	GGACTCCT	503	TATCCTCT	705	GGACTCCT	502	CTCTCTAT
706	TAGGCATG	503	TATCCTCT	705	TAGGCATG	502	CTCTCTAT
701	TAAGGCGA	504	AGAGTAGA	701	TAAGGCGA	503	TATCCTCT
702	CGTACTAG	504	AGAGTAGA	702	CGTACTAG	503	TATCCTCT
	✓✓✓✓✓✓✓✓		✓✓✓✓✓✓✓✓		✓✓✓✓✓✓✓✓		✓✓✓✓XXXX

✓= signal in both color

X = signal missing in one color channel

FIGURE 3. Examples of good and bad index combinations.

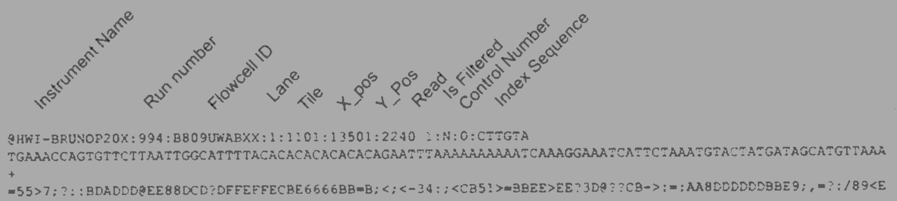
the sequenced bar code and the reference. Quality scores for the bar code bases are not used. Reads are then split into separate **FASTQ files** for each bar coded sample. The index sequence for the bar code is added to the end of the header line of each read in the FASTQ file (see Fig. 4). Reads with more than one base mismatch are dumped into a FASTQ file of "undetermined_indicies." It may be possible to recover some reads from this file by comparison with the set of bar codes used, but a sequencing run with lots of bad bar code reads may be unusable.

OTHER DEMULTIPLEXING METHODS

The metagenomics software suite Quantitative Insights into Microbial Ecology (QIIME) has its own bar code demultiplexing program (demultiplex_fasta.py) that operates on FASTA input files, assuming the bar code is at the beginning of each read.

FASTX-Toolkit

FASTX-Toolkit (http://hannonlab.cshl.edu/fastx_toolkit/commandline.html#fastx_barcode_splitter_usage) has a bar code splitter (fastx_barcode_splitter.pl) that reads a FASTA or FASTQ input file and splits it into individual files for each bar code with a user-specified number of allowed mismatches. The bar code sequence itself



```
@HWI-BRUNOP20X:994:B809UWABXX:1:1101:13501:2240 1:N:0:CTTGTA
TGAAACCAGTGTTCTTAATTGGCATTTTACACACACACACACAGAATTTAAAAAAAAAATCAAAGGAAATCATTCTAAATGTACTATGATAGCATGTTAAA
+
=55>7;?::BDADDD@EE88DCD?DFFEFFECBE6666BB=B;<;<-34:;<CB51>=BBEE>EE?3D@??CB->:=:AA8DDDDDDBBE9;,=?:/89<E
```

FIGURE 4. Illumina FASTQ format contains information in the header line for the Instrument Name, Run number, Flowcell ID, Lane, Tile, X and Y grid position, Read quality filter, and the Bar Code Index Sequence. (Used with permission from Illumina Casava v1.8.2 Users Guide, Illumina Inc., 2011.)

is trimmed off from each sequence read. The bar codes can be specified at the beginning or end of each read sequence. Sequence quality values are not used.

ea-utils

ea-utils (https://code.google.com/p/ea-utils/wiki/FastqMultx) has a demultiplexing tool for FASTQ files called **FastqMultx** with similar features—split FASTQ file by bar code, allow mismatches, and trim bar code from each sequence.

Sabre

Sabre (https://github.com/najoshi/sabre) is a demultiplexing tool from Najoshi (creator of Sickle and Scythe). Sabre takes an input FASTQ file and an input bar code data file and outputs the reads demultiplexed into separate files using the file names from the data file. The bar codes will be stripped from the reads and the quality values of the bar code bases will also be removed. Any reads with unknown bar codes get put into the "unknown" file specified on the command line. The -m option allows for mismatches in the bar codes.

SEQUENCE QUALITY

Many investigators who send samples to a collaborating laboratory, core facility, or commercial service provider for sequencing will only receive a FASTQ file for each sample and will not have access to SAV reports. The FastQC program (http://www.bioinformatics.babraham.ac.uk/projects/fastqc) has become the standard method for quality assessment of FASTQ files (or **SAM/BAM files**). The metrics provided by FastQC for each data file include total number of sequences, sequence length, overall %GC, a graph of per base sequence quality, mean quality per read, base composition and %GC along the length of all reads, mean %GC per read, per position count of "N" (undetermined) bases, sequence duplication levels, overrepresented sequences, and overrepresented ***k*-mer** content. FastQC is available as an interactive web-based program hosted by the Babraham Institute, or as a free downloadable Java program available under the GPL v3.

Interpretation of QC metrics depends on the nature of the samples being sequenced, sample preparation methods, and the performance of the sequencing machine. The FastQC program provides a set of green/orange/red flags that correspond to good/warning/failure for specific metrics, but these warnings assume a sample of random and diverse DNA fragments, such as a whole-**genome** shotgun library. Other samples, such as **RNA-seq**, targeted sequencing of PCR amplicons, restriction enzyme digests, or fragments containing adapters internal to the sequencing adapters, may receive warnings, yet fully meet the goals of the experiment.

The per base graph of Q scores produced by FastQC has become a standard representation of sequencing quality (see Fig. 5). The x-axis represents position on the reads and the y-axis is sequence quality on the Phred scale, where green is considered good ($Q > 28$), orange moderate ($28 > Q > 20$), and red poor ($20 > Q$). Many data sets show a decline in quality at the 3′ end, which may be trimmed off to improve overall data quality. Low-quality bases may also be observed at the 5′ end. However, a low-quality region (or single base) at any location within the central portion of the sequence is indicative of a serious problem with the sequencing machine. The per sequence quality graph for a good sequencing run shows a fairly tight range of overall quality scores above Q30. A wide distribution of quality scores or a bimodal distribution indicates problems sequencing some clusters, which is generally indicative of a problem with the sequencer (imaging, flow cell chemistry, cluster amplification,

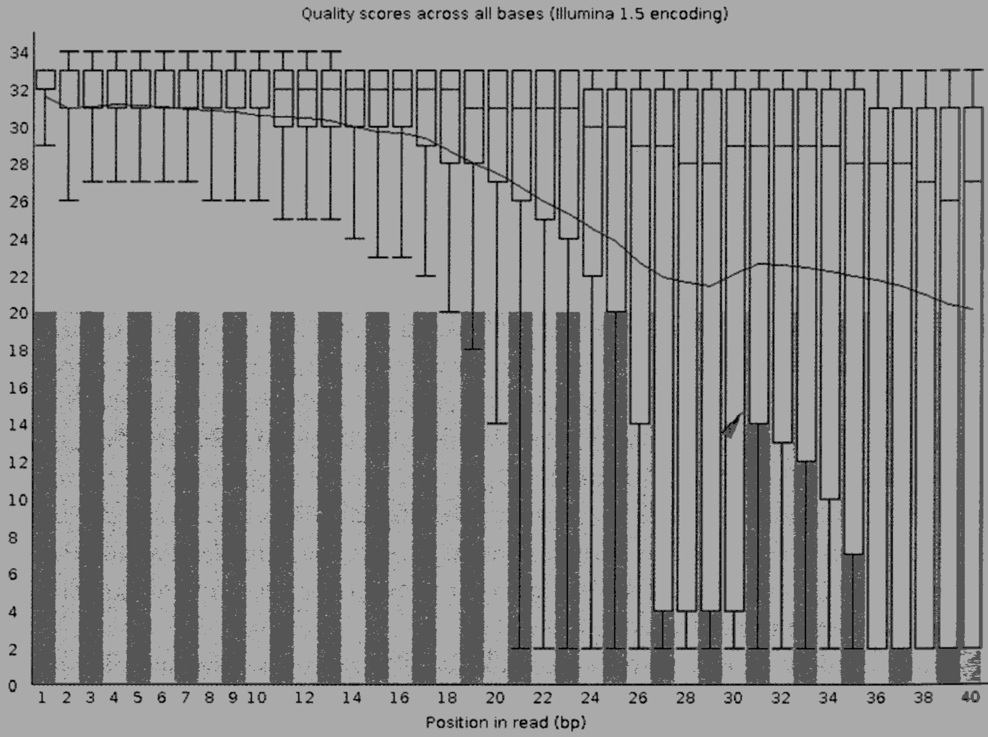


FIGURE 5. BoxWhisker plot of per base quality produced by FastQC. In this sample, sequence quality declines toward the 3′ end of the reads and the median sequence quality (red line) is below Q30 after about cycle 24. This sequence data file would be unacceptable for most experiments. (From http://www.bioinformatics.babraham.ac.uk/projects/fastqc/bad_sequence_fastqc.html, reprinted with permission from Simon Andrews.)

etc). Similar Q-score boxplots are produced by the FASTX-Toolkit (fastq_quality_boxplot_graph.sh) and PRINSEQ.

%GC

The overall %GC of a sample is a property of a particular organism or any other source of the DNA sample. However, the graph of %GC may contain some interesting information. An overall %GC different than expected for the target organism could indicate a bias in the sequencing library or an issue with the sequencing machine. A change in %GC at a particular base in the reads may indicate a problem at a cycle in the sequencing process or a bias created during sample preparation. The DNA fragments from a single organism are expected to form a normal distribution of %GC. A bimodal distribution could be indicative of contamination from a different species (e.g., bacteria in a fruit fly sample, human contamination of a microbial metagenomic sample). (See Fig. 6.)

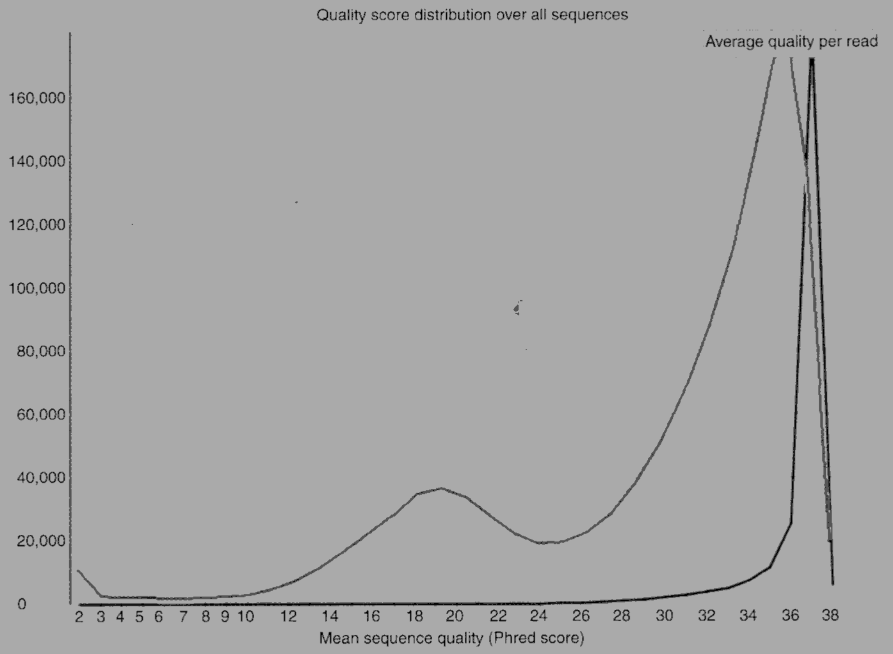


FIGURE 6. Mean sequence quality graph. The *x*-axis is the overall average quality value (**Phred score**) of each read and the *y*-axis is the number of reads observed per quality level. The red line shows a sequence quality graph of a good sample with a tight distribution of reads with Q scores > Q36 and a small tail of low-quality sequences. The blue line shows a set of sequences with a much wider distribution of quality scores and a distinct second peak of much lower-quality sequences, indicative of problems with the sequencing machine. (From http://www.bioinformatics.babraham.ac.uk/projects/fastqc/bad_sequence_fastqc.html, reprinted with permission from Simon Andrews.)

Per Base "N" Content

DNA sequencers have traditionally used the letter "N" to represent a base that cannot be called. This is equivalent to a quality of zero, but the use of N is helpful because some analysis and visualization software ignores quality information. Ns occur more frequently at the 3′ and 5′ ends of sequences. If any internal positions have more than a few percent Ns, it is indicative of a serious problem with the operation of the sequencer. FastQC raises a warning if any position shows >5% Ns and a failure is reported if any position has >20% Ns.

Sequence Length Distribution

Illumina sequence reads are all the same length, which is determined by the number of cycles of sequencing chemistry that are run. PacBio, 454, and other technologies produce reads of variable length. The graph of sequence length distribution (frequency of reads at each length) can be useful in understanding the quality of a sequencing library and in comparing different sample data files.

Duplicate Sequences

FastQC analyzes the first 200,000 reads in each FASTQ file and compares them with all of the sequences in the entire file to count duplicates, using an exact sequence match over the first 50 bp of the sequence. A graph shows the sequence copy number on the *x*-axis, and the *y*-axis represents the observed fraction of the genome that contains each number of copies. A whole-genome shotgun library should have a relatively low level of duplication (10%–20% is typical), so higher levels would be a cause for concern. RNA-seq libraries generally have more duplication (30%–40%) because highly expressed genes will be present in many copies; however, much higher levels of duplication may be an indication that the library contains a high level of rRNA sequences or concatemers of primer sequences. Enriched libraries such as **exon** capture or **ChIP-seq** may have a high level of duplication because of deep coverage, but bias due to PCR overamplification is also a concern. A targeted sequencing protocol that sequences PCR amplicons will have very high duplication levels, so this metric may not be useful.

Overrepresented Sequences

A typical whole-genome (or RNA) sequencing library will contain a diverse mixture of fragments so that no single DNA sequence will make up more than a tiny percentage of the entire FASTQ file. The FastQC Overrepresented Sequences module identifies any sequence that makes up >0.1% of a FASTQ file. The same alignment method is used as for the Duplicate Sequences module described above (exact matching between the first 50 bp of the first 200,000 sequences vs. the entire file). Any overrepresented sequences that are found are compared with a file of known contaminants that include **Illumina sequencing** adapters, PhiX sequences, human and mouse rRNA sequences, etc.

DATA PREPROCESSING

NGS data often suffers from deterioration of quality toward the 3′ end of reads and/or a fraction of reads with low overall quality. NGS library preparations may contain small insert fragments and/or adapter concatemers that lead to the presence of adapter sequences in the reads. NGS libraries may also contain DNA from nontarget species or other types of contaminants such as rRNA and mtDNA in RNA-seq data sets. Problems with sequence quality may be diagnosed from a standard QC analysis of FASTQ files, but other types of contaminants may not be detected unless the data is specifically screened by alignment to contaminant reference sequences. Depending on experimental design, NGS data files may be preprocessed in a standard workflow that includes identification and removal of adapter sequences, removal of low-quality sequences, trimming of low-quality ends from reads that contain regions with high-quality bases, and removal of contaminating sequences of various types. Quality trimming has been clearly shown to increase the quality and reliability of de novo genome assembly, transcriptome assembly, **metagenomics**, gene expression profiling with RNA-seq, and genotyping (**variant detection**) while reducing computational resources needed (Del Fabbro et al. 2013).

Removal of Adapter Sequences

Removal of adapter sequences is usually performed before any other sequence trimming or filtering, because even low-quality sequences can be recognized as adapters, and the presence of adapters may interfere with the identification of other types of contaminants. Trimming of low-quality sequences first may make it more difficult to recognize adapter sequences. Adapter contamination is often caused by sequencing of a DNA fragment that is shorter than the read length. The beginning of the read contains data from the target fragment, but when the end of the insert is reached, the sequencer continues to read through into the adapter. This results in an adapter sequence toward the 3′ end of the read. This is a serious problem because adapter sequences at the end of reads can prevent alignment to a **reference genome** or create mismatches in the alignment that may lead to false-positive detection of **sequence variants**. In **de novo assembly**, adapter sequences can block **contig** assembly or create false overlaps between unrelated **sequence fragments**.

The Illumina multiplex system uses a bar code sequence located inside of the adapters, which is read by a separate sequencing primer in a separate set of sequencing cycles. The Illumina machine software (RTA) identifies these bar codes and demultiplexes the reads into separate FASTQ files for each sample (the reads do not contain the bar code sequence, but it is appended at the end of the header; see Fig. 4). However, in some custom multiplexing systems, bar code sequences are included downstream from the standard sequencing primer, so the bar code appears at the 5′ end of the reads. Custom demultiplexing software is required to identify these bar codes, sort the samples into separate data files, and trim off the bar code sequences.

The FASTX-Toolkit, written by Assaf Gordon in Greg Hannon's laboratory at Cold Spring Harbor, provides an extremely useful set of QC and data preprocessing functions for FASTQ files including fastx_clipper for adapter removal. Fastx_clipper recognizes and clips an adapter sequence with the option to discard sequences that do not contain the adapter. It also has the option to discard all sequences with undetermined (N) bases.

NGS QC Toolkit (Patel and Jain 2012) provides a similar set of tools. Adapter sequences are removed based on matching (ungapped alignment) of 20-bp user-specified adapter sequences versus 50 bp at the 5′ and 3′ ends of each read, allowing for only one base mismatch.

Trimmomatic is the most widely used adapter removal and quality trimming program for Illumina FASTQ files, written in Java by Anthony Bolger at the Max Planck Institute (Lohse et al. 2012; Bolger et al. 2014). Trimmomatic contains a built-in database of known Illumina adapter sequences for Truseq2 and TruSeq3 sample kits. Additional adapter sequences can be added as FASTA sequences, but should also include the reverse complement of each sequence. Adapters are first recognized by an ungapped seed alignment of 16-bp sections of adapter to the entire read with a maximum mismatch of 1 or 2 bases. When the 16-bp seed alignment matches above a threshold score, the entire adapter is aligned to the read with a method similar to **Smith–Waterman alignment**.

In a **paired-end sequencing** run on a short insert, the same fragment will have adapter sequences at the 3′ and 5′ ends of both forward and reverse reads, and the entire insert sequence will be included in both reads, which allows Trimmomatic to detect short adapter sequences with much greater sensitivity and precision (see Fig. 6). Trimmomatic also includes features for quality trimming using either a user-specified sliding window (window size and average quality score cutoff) or an adaptive quality trim method called MaxInfo that allows the user to specify a "strictness" value that controls a trade-off between preserving read length (maximizes total data from the run) versus removal of incorrect bases (maximizes quality of the data).

AdapterRemoval (Lindgreen et al. 2012) is another program; it uses a similar approach with greater sensitivity than Trimmomatic, but at the cost of greater false positives (removal of nonadapter sequences). (See Fig. 7.)

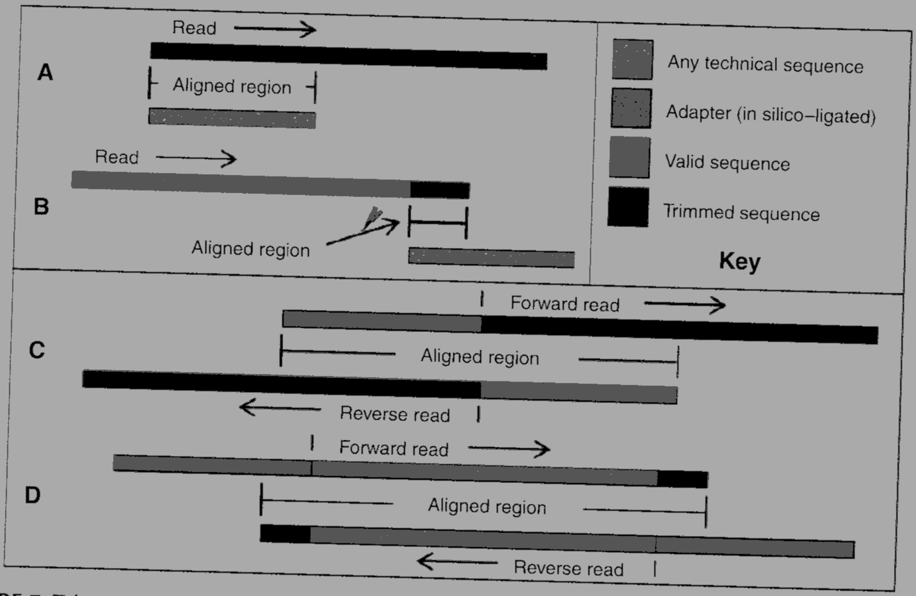


FIGURE 7. Trimmomatic strategy to identify adapter sequences at the end of paired-end sequence reads with short insert size. (*A*) The read is entirely composed of an adapter sequence, so it is easily aligned with the sequence in the screening database. (*B*) The read contains only a short portion of an adapter sequence, which cannot be recognized by alignment if it is below a detection threshold. (*C*) A paired-end sequencing of a read containing only an adapter. (*D*) Short adapter sequence is detected and trimmed at both ends of a set of **paired-end reads**. (Reprinted from http://www.usadellab.org/cms/uploads/supplementary/Trimmomatic/TrimmomaticManual_V0.32.pdf; see also Bolger and Giorgi 2014, with permission from Oxford University Press.)

Scythe (Vince Buffalo 2011–2012) is a freely available Linux command line tool on GitHub (https://github.com/najoshi/scythe). Scythe uses a naïve Bayesian approach to classify contaminant substrings in sequence reads. It considers quality information, which can make it robust in picking out 3′-end adapters, which often include poor-quality bases. Scythe compares the probability of two likelihood models for the 3′ and 5′ ends of each sequence: Does the sequence match the contaminant adapter sequences (in a FASTA file) or is it a random sequence? By default, the prior contamination rate is set at 0.05 and the minimum number of matching bases is set at 5. Scythe can be used on paired-end data, but it trims the forward and reverse reads independently, so it does not benefit from overlap information to identify small inserts.

Cutadapt, written by Marcel Martin (2011) at TU Dortmund University, recognizes and removes adapter sequences from NGS FastQC files using a sensitive gapped semiglobal alignment method that tolerates mismatches and **indels**. It has a quality trim option to remove low-quality bases from the 3′ ends of reads before alignment to an adapter, but it does not make use of quality scores to evaluate the alignments. Cutadapt is written in Python and is actively maintained in 2014.

Quality Trim

NGS data often suffer from lower quality at the 3′ end of reads and/or at the 5′ end. Removal of low-quality reads improves alignment to the reference genome and dramatically reduces false-positive SNP calls (by 3×) without a noticeable loss of coverage on the reference genome (Del Fabbro et al. 2013). The QUAKE read correction method identifies and removes reads that contain rare *k*-mers (likely sequencing errors) and shows improved speed and quality of de novo genome assembly (Kelley et al. 2010).

Some current de novo assemblers such as ABySS and ALLPATHS-LG incorporate data quality filters within the details of their algorithm (removing rare *k*-mers, which usually contain sequencing errors, from the **de Brujin graph** before contig generation). Quality filtering before assembly with these tools will have little useful effect and may remove some data that could otherwise be used to build longer contigs. In the case of RNA-seq, trimming may produce a detrimental trade-off between sensitivity (total number of aligned reads) and specificity (number of correctly aligned reads).

Trimmomatic removes Ns and very low-quality bases one by one from both the 3′ and the 5′ sequence ends, and then a sliding window/quality function is used that removes low-quality bases only from the 3′ end. Typical settings use a window size of 4 bases and a quality threshold of Q20 (there are no defaults), so that bases are removed one by one from the 3′ end until the average quality within the window is greater than or equal to the threshold.

FASTX Quality Filter throws away all reads that do not meet a specified minimum average quality. Parameters are the Q-score threshold and the percentage of bases in the read that must have quality at or above the threshold.

FASTX Quality Trimmer scans through all reads from the 5′ to the 3′ end, and when it encounters a base with a quality score of less than the quality threshold, trims off the rest of the read and then subsequently removes reads shorter than length threshold.

Sickle uses an adaptive window that is set to 10% of the read length (i.e., 10 bases for a 100-bp read) and a target minimum Q value (default Q20). The window starts at the 5′ end of the read and trims bases one by one from the 5′ end until the average quality within the window is greater than or equal to the target. Then the window continues to slide toward the 3′ end until the average quality drops below the target. At that point all bases within the window and all remaining 3′ bases are deleted. This is similar to using a trimming window that slides inward from both the 5′ and 3′ ends; however, a stretch of low-quality sequences somewhere in the middle of the read will cause all downstream 3′ sequences to be removed.

SolexaQA finds the largest contiguous stretch of bases within each read with bases all having quality values above a threshold (default Q13) and removes all bases outside of that window.

PRINSEQ provides both a downloadable (stand-alone Perl script) and a web-based tool (http://edwards.sdsu.edu/cgi-bin/prinseq/prinseq.cgi) for quality assessment and preprocessing of raw sequencing data. Reads can be filtered (and discarded) by average quality score, minimum or maximum %GC content, and maximum percentage of Ns. Reads can be trimmed using a sliding window of user-specified size from the 3′ and/or 5′ ends and a *Q*-value threshold for the average score within the window.

Cutadapt is primarily an adapter removal tool; however, it provides a trim-qualities parameter that is designed to remove low-quality bases from the ends of reads before identifying adapter sequences. The trimming algorithm is the same as the one used by BWA: Subtract a *Q*-value cutoff from all qualities; compute partial sums from all indices to the end of the sequence; and cut the sequence at the index at which the sum is minimal.

Filter Contaminating Sequences

The definition of contaminating sequences can differ in different types of NGS experiments. In standard genome sequencing for variant detection, **de novo sequencing**, or enrichment analysis (ChIP-seq or its many variants), contaminating sequences might come from other organisms such as bacteria or viruses (sample contaminants, laboratory contaminants, or DNA present in reagents used for sample preparation). In metagenomic analysis, contaminating sequences might be from human, mouse, or whatever organism served as host for the microbes. In RNA-seq, ribosomal RNA and mitochondrial RNA are considered contaminants. These sequences are generally removed by alignment of the entire FASTQ file to a reference or "contaminant" sequence file.

When a specific set of contaminating sequences are known to occur in an NGS experiment, such as rRNA in RNA-seq or human DNA in metagenomics, the FASTQ files can be preprocessed by an ad hoc strategy with a standard NGS alignment tool such as BWA or Bowtie using the contaminants as the reference genome. Unmatched sequences are redirected to a new "cleaned" output file, which is used for all subsequent analyses. The widely used MG-RAST metagenomics server uses a Bowtie search against the human genome (or other model organisms) to filter data in its annotation pipeline (Meyer et al. 2008).

NGS samples may contain unknown contaminating sequences from one or many different organisms. Hadfield and Eldridge (2014) have developed a multi-genome alignment tool (MGA) to screen for contaminants in FASTQ files. MGA is optimized for speed so that it can be run routinely as part of a NGS data processing pipeline for all samples produced in a sequencing facility. MGA takes a subsample of reads (100,000 reads per lane or per FASTQ file), trims the reads to 36 bases, and then uses Bowtie to align against a collection of genome sequences from bacteria, viruses, fungi, and laboratory animals. MGA also screens for adapter contamination by more sensitive alignment to a set of adapter and primer sequences using the Exonerate **sequence alignment** tool (Slater and Birney 2005). This is run using a local alignment model with affine gaps, similar to the Smith–Waterman–Gotoh algorithm (Smith and Waterman 1981; Gotoh 1982). The result of MGA screening is an easy-to-read graphical report that identifies contamination, but it does not produce a clean data file with contaminating sequences removed.

DeconSeq Human Contaminant Filter

The **Human Microbiome Project** generated large data sets of **16S ribosomal DNA** (rDNA) amplicons and shotgun metagenomic data from samples associated with the human body (skin, oral, fecal, nasal, vaginal, etc.) that contained widely varying amounts of human DNA. Because the goal of the project was to make microbiome data sets widely available for reanalysis by many researchers, and the research subjects did not consent to make their own personal genomes public, the samples must be filtered for contaminating human DNA. The extremely large amount of DNA data made the use of BLAST impractical. The DeconSeq program was developed to meet the needs of this project (Schmieder and Edwards 2011). DeconSeq uses BWA-SW (Burrows–Wheeler Aligner) to find short, nearly exact matches between NGS reads and the human reference genome (or any other reference genome installed by the user) using a combination of high percent identity (default = 94%) and read coverage (how much of the read is similar to the reference sequence, default = 90%). DeconSeq is available as stand-alone source code (Perl scripts and modified BWA code) freely distributed under GPL3 and as a web application (http://edwards.sdsu.edu/cgi-bin/deconseq/deconseq.cgi).

The Human Microbiome Project (HMP) also developed the Best Match Tagger (BMTagger) to computationally filter the human sequence from microbial whole-genome shotgun (WGS) sequences. BMTagger is a heuristic tool that discriminates between human reads and microbial reads without doing an alignment of all reads to the human genome. The algorithm discriminates between human reads and microbial reads by comparing consecutive sequences of 18-mer-length nucleotides found in the sequence reads with those found in the human genome. Reads are sorted into three classes: Those that contain few consecutive human-matching 18-mers (<10% of the read) are tagged as nonhuman, those that contain >80% matching 18-mers are tagged as human, and those with an intermediate amount of matching 18-mers are tagged as undetermined and further tested by an alignment procedure called srprism that guarantees to find matches with up to two errors in reads that are at least 32 bp long. The BMTagger has been adopted as standard operating procedure (SOP) for human contaminant screening of metagenomic data for the HMP (http://www.hmpdacc.org/doc/HumanSequenceRemoval_SOP.pdf) and it has been incorporated in the CloVR virtual machine, which is available on the academic DIAG cloud and the Amazon EC2 commercial service (Angiuoli et al. 2011).

SOP RECOMMENDATIONS

Overall QC Evaluation: FastQC

FastQC provides a comprehensive selection of quality metrics on raw FASTQ files with easy-to-read graphical output.

Adapter Removal: Trimmomatic

Adapter removal is essential for all types of NGS data analysis. Trimmomatic has a superior algorithm for paired-end data to detect adapter contamination of NGS reads in FASTQ files caused by small insert size. It is widely used by de novo genome sequencing projects and highly cited (113 Google Scholar citations).

Quality Trimming

Removal of uncalled "N" bases from sequence ends and of reads containing large numbers of Ns is beneficial. Quality trimming is not always essential or beneficial for de novo assembly or RNA-seq applications when assemblers and aligners have built-in quality filtering methods. Trimming with moderate quality thresholds (Q10 to Q20) is recommended for de novo assembly. Trimming is not advised before RNA-seq alignment with TopHat. Trimming greatly improves genotyping/variant detection pipelines by reducing false positives, so a higher-quality threshold (Q20 to Q30) may be beneficial. Quality thresholds should be adjusted to ensure that the majority of the data set is included in the high-quality fraction.

A preprocessing workflow recommended in the Trimmomatic manual is as follows:

- Remove Illumina adapters provided in the TruSeq3-PE.fa file (provided). Initially Trimmomatic will look for seed matches (16 bases) allowing maximally two mismatches. These seeds will be extended and clipped if in the case of paired-end reads a score of 30 is reached (about 50 bases), or in the case of single-ended reads a score of 10 (about 17 bases).
- Remove leading low-quality or N bases (below quality 3).
- Remove trailing low-quality or N bases (below quality 3).

- Scan the read with a 4-base-wide sliding window, cutting when the average quality per base drops below 15.
- Drop reads that are <36 bases long after these steps.

REFERENCES

Angiuoli SV, Matalka M, Gussman G, Galens K, Vangala M, Riley DR, Arze C, White JR, White O, Fricke WF. 2011. CloVR: A virtual machine for automated and portable sequence analysis from the desktop using cloud computing. *BMC Bioinformatics* **12:** 356.

Bolger AM, Lohse M, Usadel B. 2014. Trimmomatic: A flexible trimmer for Illumina sequence data. *Bioinformatics* **30:** 2114–2120.

Caporaso JG, Lauber CL, Walters WA, Berg-Lyons D, Lozupone CA, Turnbaugh PJ, Fierer N, Knight R. 2011. Global patterns of 16S rRNA diversity at a depth of millions of sequences per sample. *Proc Natl Acad Sci* **108** (Suppl 1): 4516–4522.

Del Fabbro C, Scalabrin S, Morgante M, Giorgi FM. 2013. An extensive evaluation of read trimming effects on Illumina NGS data analysis. *PLoS ONE* **8:** e85024.

Hadfield J, Eldridge MD. 2014. Multi-genome alignment for quality control and contamination screening of next-generation sequencing data. *Front Genet* **5:** 31.

Hamady M, Walker JJ, Harris JK, Gold NJ, Knight R. 2008. Error-correcting barcoded primers for pyrosequencing hundreds of samples in multiplex. *Nat Methods* **5:** 235–237.

Illumina Inc. 2008. Multiplexed sequencing with the Illumina Genome Analyzer System. www.illumina.com.

Joshi NA, Fass JN. 2011. Sickle: A sliding-window, adaptive quality-based trimming tool for FastQ files (Version 1.33) [Software]. Available at https://github.com/najoshi/sickle.

Kelley DR, Schatz MC, Salzberg SL. 2010. Quake: Quality-aware detection and correction of sequencing errors. *Genome Biol* **11:** R116.

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

Lindgreen S. 2012. AdapterRemoval: Easy cleaning of next-generation sequencing reads. *BMC Res Notes* **5:** 337.

Lohse M, Bolger AM, Nagel A, Fernie AR, Lunn JE, Stitt M, Usadel B. 2012. RobiNA: A user-friendly, integrated software solution for RNA-Seq-based transcriptomics. *Nucleic Acids Res* **40** (Web Server issue): W622–W627.

Martin M. 2011. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet J* **17:** 10–12 (see http://journal.embnet.org/index.php/embnetjournal/article/view/200).

Meyer F, Paarmann D, D'Souza M, Olson R, Glass EM, Kubal M, Paczian T, Rodriguez A, Stevens R, Wilke A, et al. 2008. The Metagenomics RAST server—A public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics* **9:** 386.

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

Schmieder R, Edwards R. 2011. Fast identification and removal of sequence contamination from genomic and metagenomic datasets. *PLoS ONE* **6:** e17288.

Slater GS, Birney E. 2005. Automated generation of heuristics for biological sequence comparison. *BMC Bioinformatics* **6:** 31.

WWW RESOURCES

ftp://ftp.ncbi.nlm.nih.gov/pub/agarwala/bmtagger/screening.pdf Rotmistrovsky K, Agarwala R. 2011. BMTagger: Best Match Tagger for removing human reads from metagenomics data sets.

http://hannonlab.cshl.edu/fastx_toolkit FASTX-Toolkit, Cold Spring Harbor Laboratory.

http://res.illumina.com/documents/products/appnotes/appnote_miseq_libqc.pdf MiSeq System Application, Illumina Inc., 2013.

http://support.illumina.com/sequencing/sequencing_software/real-time_analysis_rta.html Illumina RTA software support.

http://www.bioinformatics.babraham.ac.uk/projects/fastqc Andrews S. 2010. FastQC, a quality control tool for high throughput sequence data.

http://www.hmpdacc.org/doc/HumanSequenceRemoval_SOP.pdf Human Sequence Removal, National Center for Biotechnology Information.

https://github.com/najoshi/scythe Scythe—a Bayesian adapter trimmer. Vince Buffalo.