Supplementary Information
LongQC: A quality control tool for third generation sequencing long read data

Y. Fukasawa[*,1], L. Ermini[*], H. Wang[*,2], K. Carty[*], M-S. Cheung[*,1]

[*]King Abdullah University of Science and Technology (KAUST), Core Labs, Thuwal, Makkah, Saudi Arabia, 23955-6900
1Co-corresponding authors. To whom correspondence should be addressed: yoshinori.fukasawa@kaust.edu.sa, nicole.cheung@kaust.edu.sa
2Present address: Oxford Nanopore Technologies, Shanghai, 200000, China

Supplementary Materials and Methods

In-house Sequel empirical data

In-house empirical data were produced with the aim to generate challenging DNA sequences to be analysed by LongQC. We define those challenging sequences as DNA reads generated by produced poor quality single-molecule real-time (SMRT) libraries. These are libraries carrying insert DNA not properly repaired and carrying not well purified library template hampering therefore the action of the sequencing DNA polymerase.

DNA Extraction

Escherichia coli cultures were grown overnight on LB liquid medium at 37°C, and genomic DNA was extracted using a modified phenol-chloroform based extraction protocol (Syn and Swarup 2000). Briefly 3 ml of culture of *E. coli* grown were pellet to cells and resuspended in 0.75 mL 1% NaCl following by a further precipitation and resuspension in 0.75mL TES (10 mM Tris-HCl, 10 mM EDTA, pH 8.0, 2% SDS). After incubation at 75 °C for 15 minutes genomic DNA was extracted sequentially by using equal volumes of phenol:chloroform (3:1 v/r) and chloroform. Aqueous phase was recovered from which the DNA fraction was precipitated by centrifugation after the addition of 110 µL of 3 M sodium acetate (pH 5.2) and 1 mL of Isopropanol. Pellet was first washed with absolute ethanol (500 µL) and extracted genomics DNA was resuspended in 150 µL TE (10 mM Tris-HCl, 2 mM EDTA, pH 8.0) with 1µL 50 µg/mL RNase. Genomics DNA was stored at 4 °C and processed the following day for SMRT library preparation.

SMRT library preparation and sequencing

Genomic DNA (7.5 µg for each library) was sheared to 10 kb using Covaris g-Tube according to the manufacturer’s protocol and libraries were further size selected using BluePippin (Sage Scientific, Beverly, MA). The SMRTbell libraries were produced following the standard library protocols of the Pacific Biosciences DNA template preparation kit (Pacific Biosciences, Menlo Park, CA) with few modifications in order to generate poor quality SMRT libraries.
Only one 70% ethanol wash step was carried out for each purification step and suboptimal conditions for DNA pol ExoIII (Rogers and Weiss 1980) and ExoVII (Chase and Richardson 1974) activities (incubation at 45 °C for 30 minutes) were used during the DNA repairing step. Libraries were sequenced with Sequel platform in two SMRT cells with version 2 sequencing chemistry. Movie time was 10 hours.

Simulated Dataset

For PacBio data simulation, PBSIM was applied (Ono et al. 2013). The genome of *E. coli* strain K12 was used as a template. To generate non-sense reads, low accuracy reads were sampled from the reversed reference at higher error rate (--accuracy-mean 0.45 –accuracy-sd 0.02). For normal reads, moderate accuracy reads were generated from the reference genome (--accuracy-mean 0.85 –accuracy-sd 0.02). Simulated data for ONT reads was generated by NanoSim (Yang et al. 2017). The error profile trained on 1D reads from R9 flowcell using *E. coli* genome was applied. To get the certain fraction of non-sense reads, "Aligned / Unaligned ratio" parameter was adjusted. The other parameters were kept the default.

Quantification of actual fraction of non-sense reads

We mapped actual reads from PacBio RS-II, PacBio Sequel, and ONT MinION using minimap2 ver. 2.6. For PacBio data, we applied homopolymer-compression for k-mer which has 15bp and base-level alignment was conducted '-Hk15 –c'. For ONT reads, k-mer size was 12bp and base-level alignment was also conducted '-k12 –c'. Other parameters were set to default. The criterion previously used for mappable reads (https://github.com/rrwick/Basecalling-comparison) was applied to quantify the non-sense fraction: reads covered at 50% or more by the reference genome are classified as mappable reads, and the others were marked as non-sense reads. In addition, all datasets were also evaluated by blastn version 2.7.1+ (Altschul et al. 1990). Blast hits were first collected using a slightly relaxed E-value ('-evalue 0.0001 -task blastn -perc_identity 50'), and then further filtered with more stringent E-values using slightly modified script for blast (available at figshare). The more stringent E-value threshold was adjusted to match the false positive rate for minimap2 result (1E-8 for *E. coli* genome, 2.5E-21 for *C. elegans* genome, 1E-7 for *D. melanogaster* genome, 1E-21 for *A. thaliana* genome). Reads mapped to the reversed reference were treated as spurious false positive hits (Schwartz et al. 2003). Comparison with minimap2 results is summarized in Table S9. For Iso-Seq mapping, we mapped reads from PacBio Sequel using minimap2 ver. 2.11 with '-x splice -uf -C5 -c'. The unmapped reads of Iso-Seq dataset were further mapped to the same reference using '-Hk15 –c'.

Additional analysis of non-sense reads for in-house empirical datasets

Among the unmapped reads of the Sequel challenging datasets 1 and 2 to *E. coli* genome, we noticed numerous unmapped reads could be mapped to the PacBio spike-in control DNA. Of note spike-in reads are by default automatically removed by the Sequel platform and in these particular cases some reads seem to be leaked due to unknown reasons. Spike-in control-like reads were marked by minimap2 with the same parameter mentioned above ('-Hk15 -c'). Fraction of non-sense reads was quantified by unmarked reads to avoid overestimation of the fraction. Lists of such marked reads are available at figshare.

To exclude any contamination within our in-house datasets, we blasted non-sense reads of in our in-house *E. coli* datasets against the nt database. For Sequel in-house *E. coli* 1, Sequel in-house *E. coli* 2, and MinION *E. coli* 1D datasets, 10% of reads were randomly sampled because of computational time. Seqtk toolkit (https://github.com/lh3/seqtk) commit d210c57 was used for random sampling. The same parameters of blastn mentioned above were applied. MEGAN6 was used to screen species in the blast hits (Arumugam et al. 2019). "Max Expected", "Min Percent Identity", and "Min Support Percent" of LCA parameters in MEGAN6 were set to 1E-8, 50.0, and 0.1, respectively

Q-value estimation by DASCRUBBER suite

DASCRUBBER commit 8b737e4 (https://github.com/thegenemyers/DASCRUBBER) was applied to compute q-values for Sequel in-house *E. coli* datasets and Sequel in-house challenging *E. coli* datasets. DASCRUBBER depends on programs in DAZZ_DB (https://github.com/thegenemyers/DAZZ_DB) and DALINER2 (https://github.com/thegenemyers/DALIGNER), and we used implementations in commit number 034f1ab and efb48c3, respectively. Masking before overlap computation is done by DBdust in DAZZ_DB, and we omit repeat and tandem repeat masking parts. Because *E. coli* genome is not very repetitive, and computational time is greatly reduced when those steps are omitted.

Briefly, a database is generated by fasta2DB and DBsplit command with '-s1000 -x16' followed by DBdust command. Overlaps were computed by HPC.daligner with '-mdust -M60 -T40 -l1'. Coverage track was added to the database by Catrack program after coverage computation done by DAScover. Finally, DASqv was applied for the database to compute q-values with '-v -c {coverage}'. For -c {coverage} option, the numbers that total bases in each dataset divided by 5M bp were given.

Overlap finding and estimation of non-sense read fraction

To determine fraction of non-sense reads, which are randomly generated or highly erroneous reads, LongQC first subsamples sequences at random (10,000 as default). Next, overlapping region for each subsampled read are thoroughly searched against the entire dataset. LongQC employs minimap2 for this calculation and computes a rough coverage for each read after filtering. Filters used here were presented in a previous study with a slight modification (Li 2016): Maximum length of over-hanged region, minimum overlap ratio of read, and minimum overlap length are set to 2000, 0.4 and 0, respectively.

In addition, two different chaining score thresholds of minimap2 are applied for overlap region filtering. Colinear and shared k-mers between two sequences are searched with allowing certain gap lengths, and here a chain refers to a region having consecutive and shared k-mers ('chained' colinear k-mers) between two reads (Li 2018). Intuitively, higher score chains have more shared k-mers and smaller gap lengths. One threshold, t1, is the minimum threshold for chains, and this value is set to 40 and is universal for all platforms. The other threshold, t2, must be higher than t1 and is important for screening of non-sense reads. Coverage is computed for each read using both t1 and t2, and regardless of t1 coverage a read is marked as non-sense if no position is covered by at least *n* chains having higher scores than t2. In default, *n* is set to 3, and t2 is platform specific: 80 for PacBio (Table S1) and 160 for ONT reads (Table S2).

Per-read coverage calculation

The modified version of minimap2 implementation is used in LongQC. LongQC filters out long and moderately masked reads from minimap2 targets because of long computational time required for such reads. Fraction of masked bases on a read is computed by the DUST algorithm (Morgulis et al. 2006). The implementation in minimap2 (Li 2018) ver. 2.6 was used. Reads having more than 0.5Mbp long in length and at least 20% of masked bases or reads having 0.02Mbp long in length and at least 40% of masked bases are excluded from the targets.

Estimating per-read sequence error

Alignment-free error (divergence) estimation model using k-mer has been suggested and applied because of lower computational cost than base-level alignment (Ondov et al. 2016; Li 2018). Error rate *e* can be modeled by the total number of k-mer *n* and the number of error-free k-mer *m* in a query

$$\hat{e} = {}^{1}/_{k} \log {}^{n}/_{m}$$

This model requires a reference sequence and therefore is not suitable for any reference free statistics. However, if errors occur independently and randomly throughout the DNA template and coverage is homogeneous (Carneiro et al. 2012), the number of matches between erroneous k-mers and between error-free k-mers should be different. Erroneous k-mers indeed are expected to show fewer matches. Consequently, the above model can be updated computing *m* (here after m_c) by the number of times a k-mer appears in the data without the need of a reference. In order to compute m_c we first define a threshold level for all *n* k-mers, as follow:

$$\bar{t} = \frac{\sum_{i=0}^{n-1} c_i}{n}$$

where

n: the total number of k-mers on a query read

c: number of counts a query k-mer full (100%) matches with k-mers of other reads (full set)

we can then estimate m_c as follow:

$$m_c = \sum_{i=0}^{n-1} \delta_i \begin{cases} \delta_i = 1;\ if\ c_i > \bar{t} \\ \delta_i = 0 \end{cases}$$

In other words, if a k-mer is observed more than the average, that k-mer is simply treated as error-free k-mer

The error rate *e* can be then estimated without the need of a reference as follow

$$\hat{e} = {}^{1}/_{k} \log {}^{n}/_{m_c}$$

Estimation of coverage distribution and calculated genome/transcriptome size

From distribution of per-read coverage, LongQC then estimates sequencing depth for the sample dataset. The default statistical model for genome is a two-component Gaussian mixture model (GMM): one for true distribution and the other is for the background noise coming from highly sensitive setting of overlap finding. The top 15% of reads with highest per-read coverage is currently ignored as outliers. GMM is estimated from the rest of the data, and estimated multiple components are sorted by π_i/σ_i, where π_i is the mixing coefficient and σ_i is the standard deviation of the i-th component, respectively. The top component in the sorted list is chosen (Stauffer and Grimson 1999). For transcriptome data, mixture of a lognormal distribution and a Gaussian distribution is applied: lognormal distribution is used to cope with higher dispersion in this case. The implementation in scikit-learn is used (Pedregosa et al. 2011). Once the coverage is estimated, the calculated genome or transcriptome size can be computed from the throughput and the estimated coverage.

If the fraction of non-sense reads is unusually high (> 40%) or per-read coverage distribution has its peak around zero, GMM hardly discriminate between true distribution and background noise and LongQC also computes genome or transcriptome size using the Poisson model (Lander and Waterman 1988). Two scenarios can be considered in this case: 1) good quality but insufficient amount of data, namely zero-inflation (See Figure S3. 3x coverage is minimum requirement) or 2) problematic data where non-sense reads exist at a high percentage. Estimated size can be used to discriminate two scenarios. Let $\hat{p}$ be an observed non-sense read rate and ϵ be a true non-sense read rate. The coverage is computed from the fraction of gap region which has zero coverage under Poisson model as

$$\lambda = -\log(\hat{p} - \epsilon)$$

Crude genome or transcriptome size is then computed

$$\hat{G} = \frac{N(1-\epsilon)}{\lambda}$$

, where N is the total size of the given data. Although the true non-sense read rate ϵ is unknown, we can empirically determine a certain range of this value for a normal case.
Zero-inflation is considered for computing λ. If $\hat{p} \cong \epsilon$, this estimation does not work, hence, the estimated size shall be smaller than the actual size.

Parameterization of length distribution

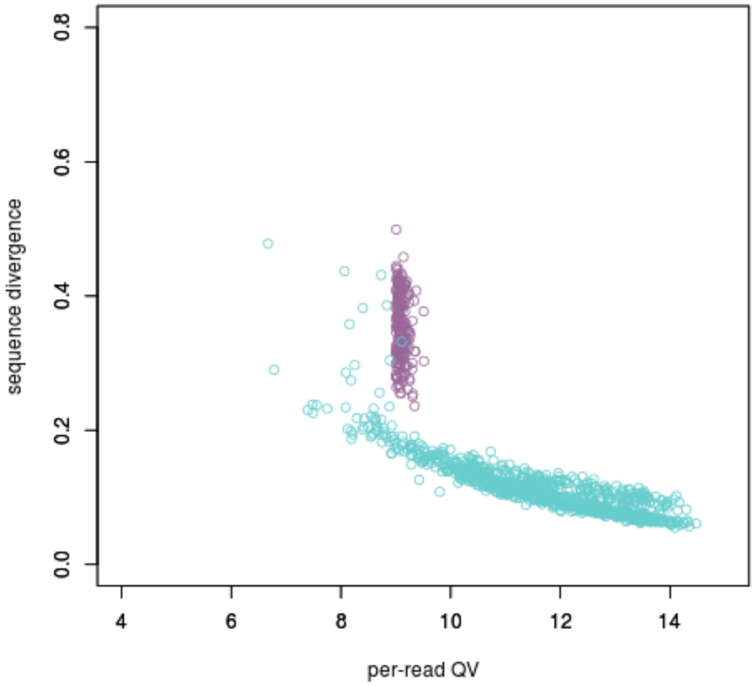
Length distribution of long reads generally shows skewed distribution with long tail if size selection is not conducted at all or is performed moderately. LongQC fits a gamma distribution to read length data because its shape parameter summarizes the observed distribution well. The distribution fitting step uses the implementation in SciPy.

GC content and sequence complexity calculation

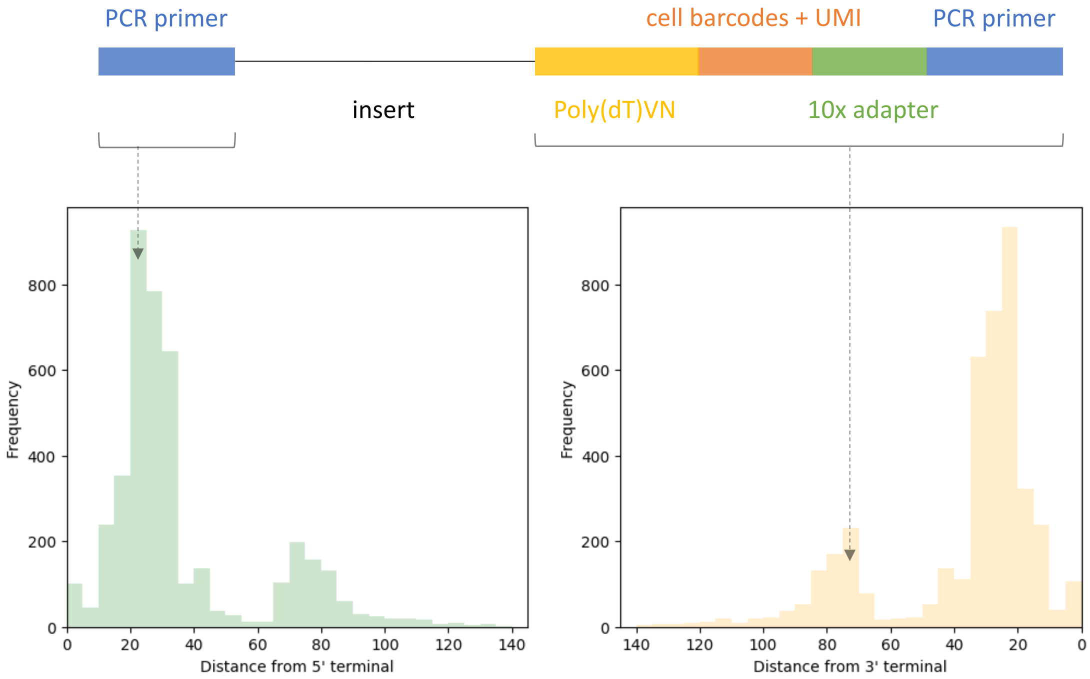
GC content of a genome has long been used as a standard statistic, and the distribution of per-read GC fraction is widely used for short read QC. In order to deal with the read length variability of TGS LongQC employs two different approaches to calculate the GC fraction. The first procedure computes the GC fraction on the whole read, independently from the read length and plots the distribution of per-chunk GC content. The second one instead standardizes per read length calculating the GC fraction on short fixed-length substrings (150bp as default).
In the latter case, because of the huge number of resulting short reads, randomly selected (20% as default) substrings are used to reduce computational time. The probability density function of both per-whole read and per-substring distributions are further estimated by kernel density estimation using Gaussian kernel implemented in SciPy. The bandwidth was determined by Scott's rule (Scott 1992).

The DUST algorithm(Morgulis et al. 2006) is applied to detect low-complexity region and compute low-complex fraction within a read. The implementation in minimap2 (Li 2018) ver. 2.6 was used.

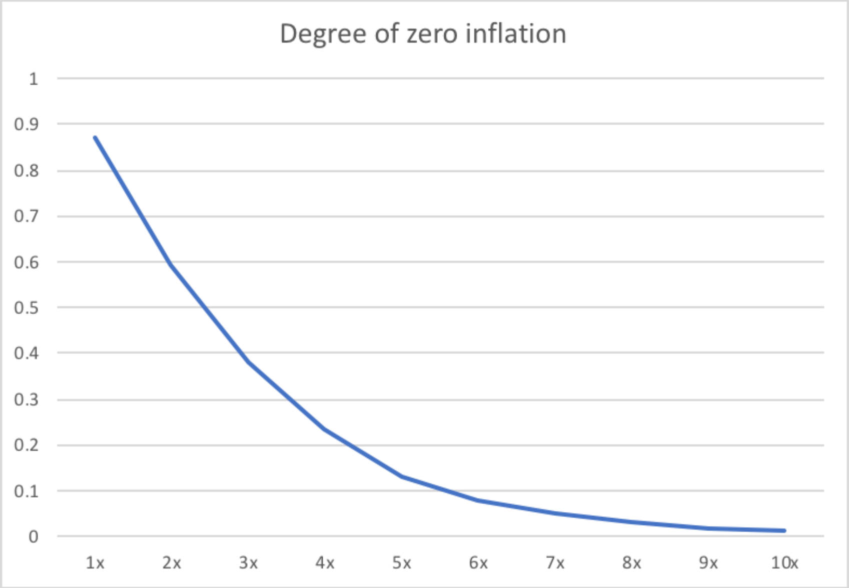
Supplementary Figures



Supplementary Figure S1: Estimated per-read error rate versus per-read QV (base-level quality scores assigned by a sequencer on a read were averaged). Purple dots represent HQNR (Mojarro et al. 2018), high quality noisy reads, and light blue dots represent the other normal reads from the same dataset (1000 reads were randomly subsampled for comparison). HQNR reads show high divergence regardless of high predicted quality score.



Supplementary Figure S2: An example of flanking region analysis using single cell data on Sequel (Gupta et al. 2018). Characteristic two peaks for each read are observed in histograms. Symmetric distribution is expected from the nature of sequencing protocol. The length of adapters having poly-T varies because of diverse poly-A length; as a result, the peak at the 3' terminal is expected to be shallower, and such trend is indeed observed.



Supplementary Figure S3: Effect of zero inflation on low coverage datasets. X-axis shows dataset depth and y-axis describes the estimated non-sense read fraction. Simulated 1x to 10x depth datasets were generated by PBSIM. Lack of coverage (<4x) wrongly classifies reads as non-sense reads.

Supplementary Tables

Supplementary Table S1: Search of good empirical threshold for PacBio datasets. Sum of squared difference was minimum at t2 = 80. Here, non-sense read fraction quantified by minimap2 were assumed to be true rate. See 'Overlap finding' in Supplementary Materials and Methods for further details of t2.

	t2 = 60	t2 = 80	t2 = 100	TRUE
Sequel in-house *E. coli* 1	0.0478	0.0575	0.0669	0.06713023
Sequel in-house *E. coli* 2	0.1004	0.1161	0.1304	0.12029256
RS-II in-house *E. coli*	0.0191	0.0291	0.0363	0.03447949
RS-II PB *E. coli*	0.0399	0.0509	0.0627	0.04862082
Sum of squared difference	0.001081953	0.00014445	0.00030375	

Supplementary Table S2: Search of good empirical threshold for Nanopore datasets. Sum of squared difference was minimum at t2 = 160. Here, non-sense read fraction quantified by minimap2 were assumed to be true rate. See 'Overlap finding' in Supplementary Materials and Methods for further details of t2.

	t2 = 120	t2 = 140	t2 = 160	t2 = 180	TRUE
MinION *E. coli* 1D	0.1432	0.1518	0.1579	0.162	0.157
MinION *E. coli* 1D^2	0.0651	0.0672	0.0695	0.0716	0.071628
Sum of squared difference	0.000233055	4.6647E-05	5.3384E-06	2.5001E-05	

Supplementary Table S3: The performance of the non-sense read fraction estimation using the real datasets. The value in parenthesis for *A. thaliana* shows the unmapped fraction against the reference genome before *E. coli* read filtering.

	PB *C.elegans*	PB *D.melanogaster*	ONT *A.thaliana*
True fraction	0.0440	0.0412	0.0995 (0.1760)
Estimated fraction	0.0413	0.0276	0.1050

Supplementary Table S4: Detected species in non-sense reads of our in-house *E.coli* datasets using blastn and nt.

	The number of reads	Zero hit at 1E-4	>=1% species
Sequel in-house *E. coli* 1[a]	4747	1851	*E. coli* (15.6%)
Sequel in-house *E. coli* 2[a]	13039	6082	*E. coli* (18.9%)

RS-II *E. coli* data1	2184	276	*E. coli* (45.8%)
MinION *E. coli* 1D[a]	8377	5402	*E. coli* (1.0%)
MinION *E. coli* 1D^2	2078	1508	*E. coli* (7.8%)
Sequel in-house challenging data 1	24686	17174	*E. coli* (2.1%), pb synthetic read (1.6%)
Sequel in-house challenging data 1	39933	30826	*E. coli* (1.7%), pb synthetic read (1.1%)

a: 10% random sampling was conducted because of computational time.

Supplementary Table S5: The estimation performance of the non-sense read fraction against the simulated datasets using NanoSim and PBSIM. For NanoSim sets, 80000 reads were generated after specifying noisy read rate. For PBSIM, 30x depth reads were mixed with 20x or 7.5x depth noisy reads to fit them 40% or 20% non-sense read rate.

	NanoSim40	NanoSim20	PBSIM40	PBSIM20
True fraction	0.4000	0.2000	0.4000	0.2000
Estimated fraction	0.4180	0.2190	0.4050	0.2120

Supplementary Table S6: QV thresholds for data quality estimated by DASqv, the fractions of the worst quality (q=50) segment, and the fractions of poor-quality reads defined by that segment.

	Thresholds[b]	Proportion of q=50 segment	Proportion of poor-quality reads[c]
Normal in-house *E. coli* dataset 1	19, 24	0.047	0.072
Normal in-house *E. coli* dataset 2	21, 28	0.080	0.143
Challenging in-house *E. coli* dataset 1[a]	20, 25	0.066	0.226
Challenging in-house *E. coli* dataset 2[a]	21, 27	0.094	0.384

a: Contaminated spiked-in control reads were removed from these datasets.
b: QV thresholds computed by DASqv. The two values refer to thresholds for good (80th percentile) and bad (93rd percentile) segments, respectively.
c: If more than 50% of segments have q=50 on a read, that read is defined as a poor-quality read here.

Supplementary Table S7: The performance of the non-sense read fraction estimation using the Iso-Seq datasets. The values in parenthesis show unmapped rate without consideration of mis-assignment of samples.

	Sequel Humming bird	Sequel Zebra Finch
True fraction	0.0456 (0.0569)	0.0377 (0.0530)
Estimated fraction	0.0407	0.0485

Supplementary Table S8: Spearman's rank correlation between estimated divergence and read identity. In all cases, correlations were statistically significant (p-value < 2.2e-16).

		Spearman's rank correlation
PacBio	RS-II in-house *E. coli*	-0.9499143
	RS-II PB *E. coli*	-0.9416994
	Sequel in-house *E. coli* 1	-0.8537211
	Sequel in-house *E. coli* 2	-0.8998917
	RS-II PB *C. elegans*	-0.8603884
	RS-II PB *D. melanogaster*	-0.7414686
ONT	MinION in-house *E. coli* 1D	-0.9116604
	MinION in-house *E. coli* 1Dsq	-0.9084482
	A. thaliana	-0.833329

Supplementary Table S9: Quantification of the non-sense read fraction for genome sequencing datasets by blastn.

	minimap2	blastn
Sequel in-house *E. coli* 1	0.0671	0.0609
Sequel in-house *E. coli* 2	0.1220	0.1080
RS-II in-house *E. coli* data1	0.0345	0.0323
RS-II *E. coli* data2	0.0486	0.0496
MinION in-house *E. coli* 1D	0.1570	0.1610
MinION in-house *E. coli* 1D^2	0.0716	0.0749
PB *C. elegans*	0.0440	0.0394
PB *D. melanogaster*	0.0412	0.0507
ONT *A. thaliana*	0.0995	0.1060

Supplementary Table S10: Runtime summary for LongQC and DASCRUBBER on normal and challenging datasets. Mean and standard deviation from three iterations are shown below. For DASCRUBBER, runtime is summation of multiple steps: from header conversion to adding coverage track to the DAZZLER database.

	LongQC	DASCRUBBER
Normal in-house *E. coli* dataset 1	10m30s ± 26s	347m 41s[b]
Normal in-house *E. coli* dataset 2	14m58s ± 19s	517m 0s[b]
Challenging in-house *E. coli* dataset 1[a]	1m29s ± 6s	3m34s ± 24s
Challenging in-house *E. coli* dataset 2[a]	1m22s ± 7s	2m11s ± 15s

a: Contaminated spiked-in control reads were removed from these datasets.

b: One iteration was applied.

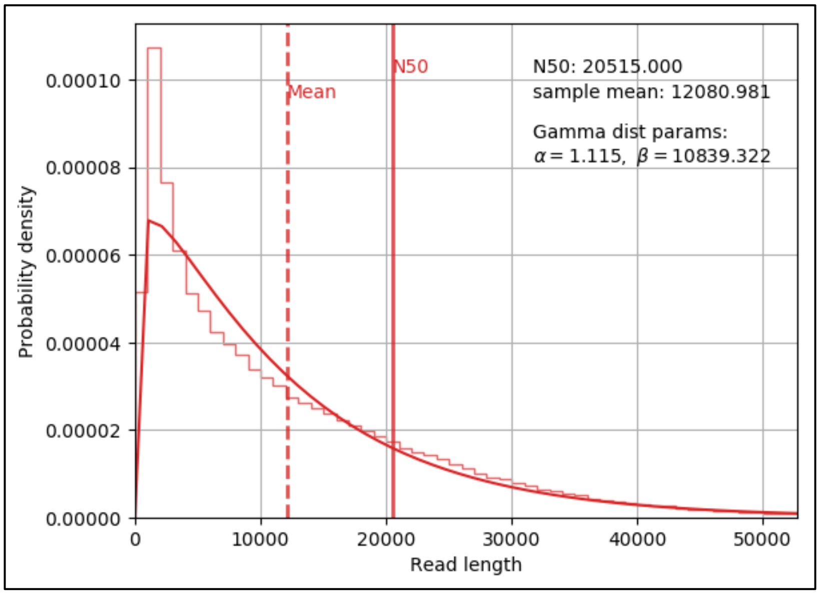
Outline of LongQC output

LongQC generates various statistics and plots for quality assessment purposes. The main output is a single HTML summary file containing plots and statistics. In addition, text-based statistics are also written in a json file. Per-read statistics are written into other text files and users can check further in details by reading or parsing those text files if needed. Explanations for each subsection in the HTML output file are given below. For illustration purpose, example plots for the Arabidopsis dataset (Michael et al. 2018) are also shown (contaminated reads were filtered. See the main text result for details).

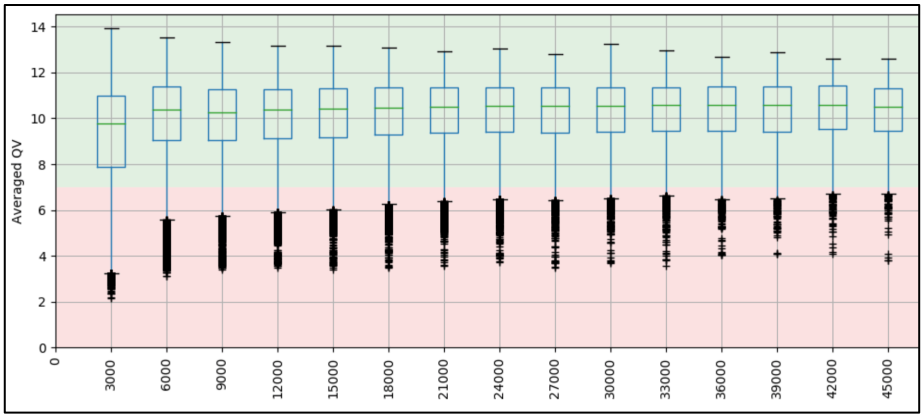
General statistics
This subsection in the HTML and json files summarizes and shows general statistics of a dataset: throughput, number of reads, and estimated non-sense read fraction.

Adapter statistics
LongQC provides a rapid adapter sequence removal functionality using Edlib, which is one of the fastest algorithms for alignment (Sosic and Sikic 2017). This subsection shows results of adapter sequence search. Adapter sequence can be provided to the program, otherwise pre-defined representative adapter sequence for ONT 1d, ONT 1d^2, PacBio Sequel or PacBio RS-II is applied to the dataset according to the platform chosen. The number of reads having adapter-like (75% or higher identity) sequences in terminals is shown. The average length of adapter-like sequences is computed if such sequences are observed. The average length should be consistent with the actual adapter used in the sequencing kit and the mode of the frequency distribution shown in the flanking region analysis plots would be at a non-zero value (see subsection “Flanking region analysis”).

Read length
This panel shows the length distribution for all reads in a given dataset. Typical genome sequencing data from ONT show unimodal exponential distribution, therefore, alpha parameter of Gamma distribution is smaller than 2. Transcripts, size selected libraries (such as PacBio data) or highly fragmented samples will show a higher alpha value because of the skewness to the left.



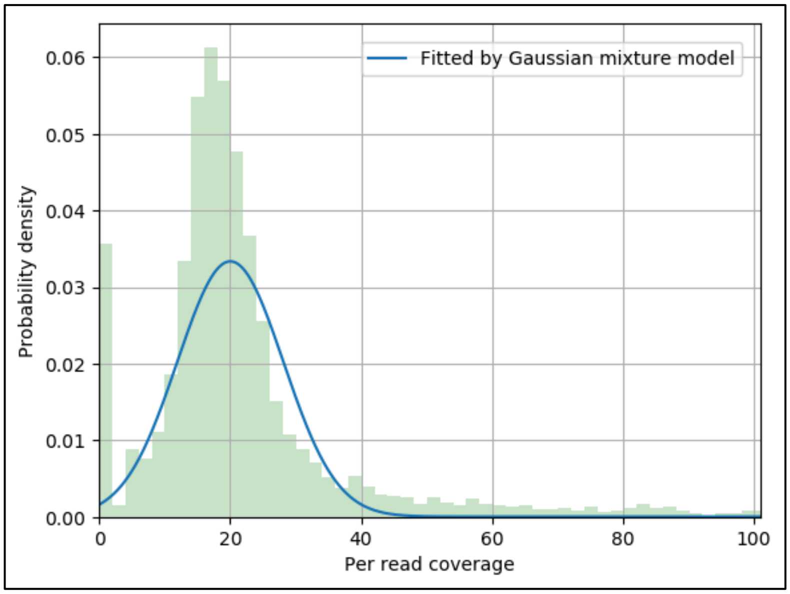
Per-read quality
This panel shows boxplots for per-read QV if QV is available in the file. The x-axis is not the position of reads but binned read length. QV threshold is set to 7, which is equivalent to 20% error rate. High quality library dataset should show high QV values regardless of read length and the median is expected to be higher than 7.



Per-read coverage
Per-read coverage subsection presents coverage statistics computed from the overlapping information between subsampled reads and the entire dataset.

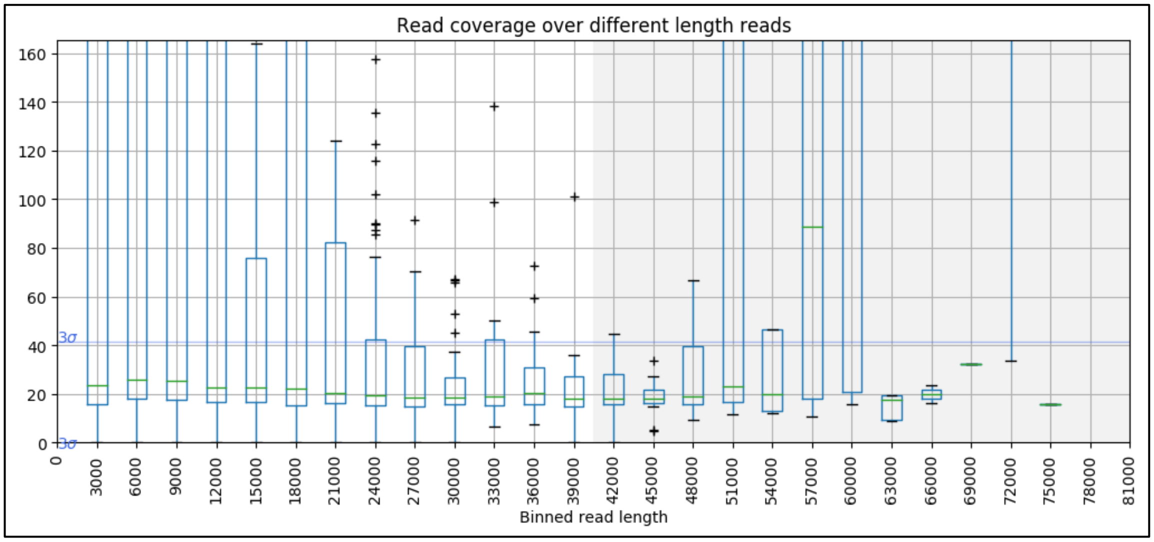
1) <u>Per read coverage distribution</u>

The first plot is an overview of per-read coverage. A single peak is expected except for metagenomic samples. LongQC automatically fits a curve using Gaussian mixture model (for genome) or a mixture of Gaussian and lognormal distribution (for transcriptome) to discriminate the true peak from the background. Mean/Median is then used for rough genome/transcriptome size estimation. Multiple peaks, when the library is not metagenomic, indicate that the dataset has an overdispersion of coverage distribution.



2) Read coverage over different length reads

This plot shows fluctuations of per-read coverage over different read lengths. In genome sequencing data, per-read coverage is expected to follow normal distribution and therefore fluctuation of medians for per-read coverage should be within a certain range (e.g. 3 standard deviations) regardless of read length. A significant deviation can highlight potential issues (Figure 2). Our experience suggested that such deviation could indicate contamination of different scale genomes/low quality library (i.e. overloading in PacBio).



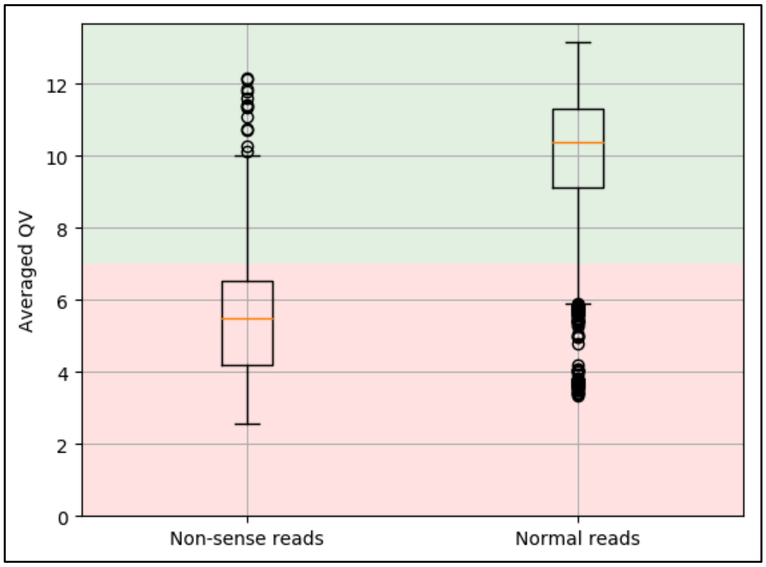
3) QV for normal and non-sense reads

Normal reads should show higher per-read QV value than that of non-sense reads. Ideally, median of normal reads should be higher than 7. This panel can be interpreted into 3 scenarios:

I. Possible low coverage. If medians of per-read QV for both normal and non-sense reads are placed in green area, this could indicate low coverage. When the dataset has low coverage, some good reads will appear to be unmappable simply due to insufficient depth achieved in the dataset and therefore are falsely classified as non-sense reads despite of their high QV.

II. Noisy dataset. When the medians for both normal and non-sense reads are smaller than 7 the dataset is particularly noisy. Further downstream analysis can be adversely affected.

III. Good dataset. When the median for non-sense reads is lower than 7 and the median for normal reads is higher than 7 the dataset is considered canonical.

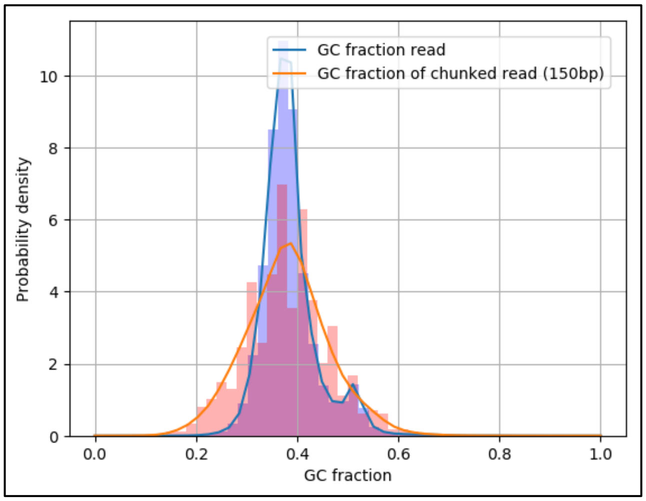


It is worth mentioning that mean/median coverage shown in this section can be lower than coverage obtained by mapping reads to a reference. Mapping to uncorrected error-prone sequences in the case of LongQC is less sensitive and coverage could be affected. Similarly, estimated genome/transcriptome size could be larger than the actual size because of this effect.

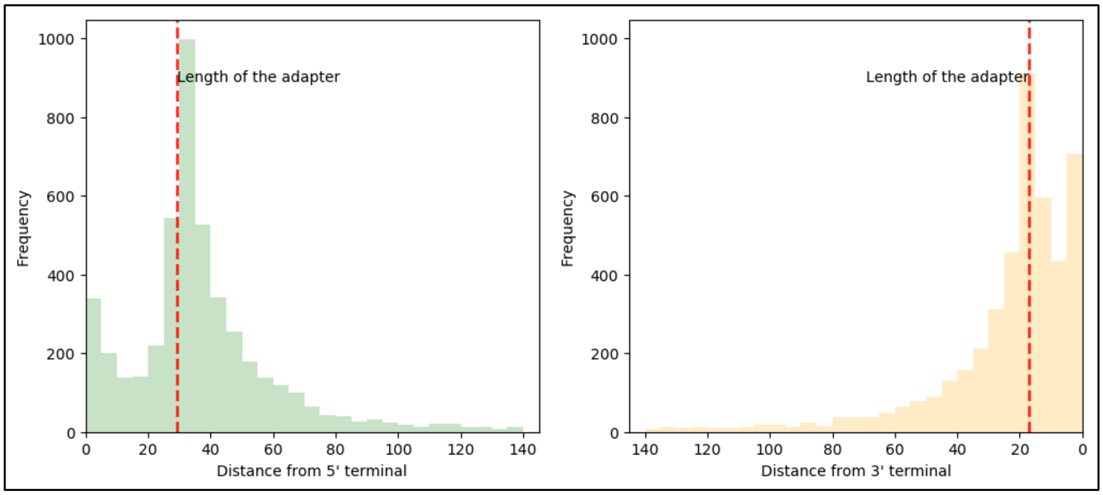
GC contents

GC content is shown in this subsection by two different distributions, computed from the same dataset. The first distribution is calculated using end-to-end full length reads, and the second one is computed from chunked subsequences. The first one should show sharper distribution, because of a smaller deviation reflecting the use of longer sequences.

Of note the full length read GC content distribution can look slightly different if the same sample is sequenced with different run/platforms (e.g. *E. coli* sequenced on Sequel and MinION). Each platform, and even different version of chemistry on the same platform, has different characteristic read length, and different modes in read length can affect the full length read GC content distribution. The chunked approach instead is more insensitive to sequencing or platform differences and does not show this bias since the length is standardized. When using a mixed dataset of different samples/organism, either or both the full length and the chunked approach can be used as a proxy for inferring sample contaminations showing a divergent GC distribution.



Flanking region analysis
These plots can be used to interrogate the presence of specific sequences like adapters. If there are no artificial sequences such as adapter sequences, the mode of the frequency distribution should be at 0 for both terminals and frequency should steeply decline from 0. Otherwise, specific patterns should reflect the characteristics of terminal sequences present for each application. If adapter-like sequences are observed by sequence search using Edlib (see above), average length of such sequences is plotted as a dashed vertical red line.



Sequence complexity
This panel shows distribution of per-read sequence complexity score computed by DUST algorithm (Morgulis et al. 2006).

Per-read sequence error
This is given as a part of text summary file for subsampled genomics DNA sequences. Empirically estimated error rate is computed for randomly sampled sequences (See computational details in the materials and methods below).

Literature Cited

Altschul, S.F., W. Gish, W. Miller, E.W. Myers, and D.J. Lipman, 1990 Basic local alignment search tool. *J Mol Biol* 215 (3):403-410.

Arumugam, K., C. Bagci, I. Bessarab, S. Beier, B. Buchfink *et al.*, 2019 Annotated bacterial chromosomes from frame-shift-corrected long-read metagenomic data. *Microbiome* 7 (1):61.

Carneiro, M.O., C. Russ, M.G. Ross, S.B. Gabriel, C. Nusbaum *et al.*, 2012 Pacific biosciences sequencing technology for genotyping and variation discovery in human data. *BMC Genomics* 13:375.

Chase, J.W., and C.C. Richardson, 1974 Exonuclease VII of Escherichia coli. Mechanism of action. *J Biol Chem* 249 (14):4553-4561.

Gupta, I., P.G. Collier, B. Haase, A. Mahfouz, A. Joglekar *et al.*, 2018 Single-cell isoform RNA sequencing characterizes isoforms in thousands of cerebellar cells. *Nat Biotechnol*.

Lander, E.S., and M.S. Waterman, 1988 Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics* 2 (3):231-239.

Li, H., 2016 Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. *Bioinformatics* 32 (14):2103-2110.

Li, H., 2018 Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*.

Michael, T.P., F. Jupe, F. Bemm, S.T. Motley, J.P. Sandoval *et al.*, 2018 High contiguity Arabidopsis thaliana genome assembly with a single nanopore flow cell. *Nat Commun* 9 (1):541.

Mojarro, A., J. Hachey, G. Ruvkun, M.T. Zuber, and C.E. Carr, 2018 CarrierSeq: a sequence analysis workflow for low-input nanopore sequencing. *Bmc Bioinformatics* 19.

Morgulis, A., E.M. Gertz, A.A. Schaffer, and R. Agarwala, 2006 A fast and symmetric DUST implementation to mask low-complexity DNA sequences. *Journal of Computational Biology* 13 (5):1028-1040.

Ondov, B.D., T.J. Treangen, P. Melsted, A.B. Mallonee, N.H. Bergman *et al.*, 2016 Mash: fast genome and metagenome distance estimation using MinHash. *Genome Biology* 17.

Ono, Y., K. Asai, and M. Hamada, 2013 PBSIM: PacBio reads simulator--toward accurate genome assembly. *Bioinformatics* 29 (1):119-121.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion *et al.*, 2011 Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12:2825-2830.

Rogers, S.G., and B. Weiss, 1980 Exonuclease III of Escherichia coli K-12, an AP endonuclease. *Methods Enzymol* 65 (1):201-211.

Schwartz, S., W.J. Kent, A. Smit, Z. Zhang, R. Baertsch *et al.*, 2003 Human-mouse alignments with BLASTZ. *Genome Res* 13 (1):103-107.

Scott, D.W., 1992 *Multivariate density estimation: theory, practice, and visualization*. New York: John Wiley & Sons.

Sosic, M., and M. Sikic, 2017 Edlib: a C/C++ library for fast, exact sequence alignment using edit distance. *Bioinformatics* 33 (9):1394-1395.

Stauffer, C., and W.E.L. Grimson, 1999 Adaptive background mixture models for real-time tracking, pp. 2246 in *cvpr*. IEEE.

Syn, C.K., and S. Swarup, 2000 A scalable protocol for the isolation of large-sized genomic DNA within an hour from several bacteria. *Anal Biochem* 278 (1):86-90.

Yang, C., J. Chu, R.L. Warren, and I. Birol, 2017 NanoSim: nanopore sequence read simulator based on statistical characterization. *Gigascience* 6 (4):1-6.