FREIE UNIVERSITÄT BERLIN

Mason – A Read Simulator for Second Generation Sequencing Data

Manuel Holtgrewe

B-10-06
October 2010

FACHBEREICH MATHEMATIK UND INFORMATIK
SERIE B • INFORMATIK

Abstract

We present a read simulator software for Illumina, 454 and Sanger reads. Its features include position specific error rates and base quality values. For Illumina reads, we give a comprehensive analysis with empirical data for the error and quality model. For the other technologies, we use models from the literature. It has been written with performance in mind and can sample reads from large genomes. The C++ source code is extensible, and freely available under the GPL/LGPL.

1. Introduction

Second generation sequencing technologies yield DNA sequence data at unprecedented high throughput. The generated data has many applications, including genome re-sequencing, and structural variant detection.

Various sequencing technologies are commercially available: Instruments based on pyrosequencing by Roche/454 Life Sciences [MEA+05], reversible terminator chemistry by Illumina [BBS+08], and sequencing by oligonucleotide ligation and detection by Applied Biosystems are examples of second-generation sequencing. Helicos Biosciences offers the first commercial instrument for single-molecule sequencing.

The Short Read Archive (SRA) offers huge numbers of freely available read data. However, when developing, testing and evaluation software that processes sequencing data, using real-world data only is not desirable. While evaluating performance on real-world data is indispensable, simulated data nicely complements real data.

First, the original sample location in the genome is not known for real-world data while it is available for simulated data. Second, the data in the SRA is often more than one year old since authors publish their data in the SRA after the publication of their paper. This means that the technology used for reads in the SRA is older, e.g. with shorter read numbers org having no mate pairs. Third, simulation allows to consider certain characteristics of data in an isolated way. For example, one can increase the error rate in the simulated reads to show the robustness of an algorithm.

Because of this, many authors use simulated data for evaluating their algorithms. Many authors use their own, possibly ad-hoc, software for generating such reads. However, there are also some publications that only deal with the simulation of reads.

In [Mye99], Myers describes celsim, a program for the simulation of Sanger reads. Since the genomes of important model organisms were not known at that time, it also allows the synthesis of genomes. In [ROA+08], Richter et al. describe MetaSim, a program for the simulation of reads. The focus of MetaSim is metagenomics and it allows to sample reads from a larger set of genomes and also to artificially let these genomes evolve. Recently, in [BML+10] , Balzer et al. describe flowsim, a simulator for pyrosequencing reads, based on the analysis of empirical data.

Table 1 shows properties of these read simulators and Mason, the software described in this paper. The source code of Mason is freely available, it supports the generation

Feature	celsim	MetaSim	flowsim	Mason
Source code available			✓	✓
Sanger reads	✓	✓		✓
Sanger qualities	✓			✓
454 reads		✓	✓	✓
454 qualities			✓	✓
Illumina reads		✓		✓
Illumina qualities				✓

Table 1: Properties of read simulation software. Citations are: celsim [Mye99], MetaSim [ROA+08], flowsim [BML+10]. Mason is described in this article.

of Illumina, 454 and Sanger reads. Our group has successfully used the simulator for benchmarking software for read mapping, read correction and transcript quantification.

The read simulator has been implemented using C++ using the SeqAn library.

2. Simulation Models

The simulation program framework is described in Section 3. In this section, we describe the simulation model parts that depend on the simulated technology: The length of the physical sample (reference sequence infix), the model for simulating sequencing errors and base quality values.

Following George Box' words "All models are wrong but some are useful," the aim is not to find a model that fits reality completely. Rather, we want to find *simple* that show important characteristics of the simulated sequencing technologies.

2.1. Model for Illumina Reads

Illumina sequencing works by synthesizing one strand of the reads base by base: The reads are arranged in beads of identical reads. Chemicals flow along the beads in cycles (A, T, C, G). In each cycle, one base binds in each read. The binding of the bases emits light which is recorded by a camera. Based on the recorded points of light emission, it is decided which base has been bound most probably.

In our model, we ignore the death of beads and assume each bead emits a signal in each step. However, the steps can be faulty: The signal can be misinterpreted, not be detected, or the signal can be emitted in a later step. The faults correspond to single base polymorphisms, deleted, and inserted bases in the read.

It was shown in [DLBH08] that there are context dependent biases for errors. We ignore this fact for our model, as is usually the case in read simulation. However, we incorporate the positon of a base in the read, for the simulation of errors: Because of properties of the chemical process, the probability for an error raises with every step.

Our first draft model (which we will revise below) is thus:

Read Set	Species	Length [bp]	Submission	Platform
SRR026674	fly	36	2009/09/28	Illumina GA 2
SRR026675	fly	36	2009/09/28	Illumina GA 2
SRR026676	fly	36	2009/09/28	Illumina GA 2
SRR049254	fly	100	2010/05/20	Illumina GA 2
SRR038098	yeast	20	2010/03/16	Illumina GA 2
SRR003673	yeast	36	2008/08/12	"Illumina" (1G?)

Table 2: Properties of the read sets chosen for determining position-depending probabilities.

- The read length ℓ is constant.
- The probability for mismatches, insertions, deletions at position i in the read is given by $p_m(i)$, $p_d(i)$, and $p_i(i)$.
- On mismatches, the replacing base is chosen at random, excluding the replaced base.

We performed an experimental study to determine these position specific probabilities.

We use the program RazerS [WER+09] for mapping reads with full sensitivity against the reference genome with a minimal identity of 90 %. RazerS thus finds all best hits (see [HWRE10] for a definition of "all best hits") of up to 10 % errors (relative to the read length) for each read. This means that for each read, all matches with the lowest distance are found. The resulting multi read alignment is then analyzed for positional mismatches, inserts and deletions.

When speaking about matching, mismatching, inserted and deleted bases, we are always considering the multi read alignment returned by RazerS. We are aware that this is biased towards the alignment algorithms and scoring schemes used inside RazerS. However, any semiglobal alignment schema will have a bias.

We ran the analysis on Illumina reads of fly and yeast.

We chose flybase release 5.27 of Drosophila melanogaster as the reference sequence. As reads, we chose the read sets with SRA accession numbers SRR049254 (100 bp reads) and SRR026674-SRR026676 (36 bp reads). We concatenated chromosomes NC_001133-NC_001148 and NC_001224 for Saccharomyces cerevisiae as the reference sequence. As reads, we chose SRR038098 (20 bp reads) and SRR003673 (36 bp reads). This is summarized in Table 2.

2.1.1. Overall Errors

First, consider Table 3. This table shows the percentage of reads with a given number of errors for different read sets (the numbers for SRR026674-6 are very similar). As we can see, for the 100 bp reads, about 25 % of all reads have more than 2 errors.

errors	percentage			
	SRR038098	SRR003673	SRR026674	SRR049254
0	95.39	78.41	74.77	43.52
1	3.58	11.98	17.25	20.85
2	1.02	5.61	5.64	10.87
3	0.01	3.97	2.33	6.61
4	0	0.01	0.01	4.51
5	0	0	0	3.34
6	0	0	0	2.68
7	0	0	0	2.23
8	0	0	0	1.94
9	0	0	0	1.73
10	0	0	0	1.55
11	0	0	0	0.05

Table 3: Percentage of reads with a given number of errors.

2.1.2. Positional Errors

We show each of the following plots in two versions.

- One for read sets SRR026674, SRR049254, SRR038098, and SRR003673 (*sets A*) to show the variance between the results from different studies.
- One for the read sets SRR026674, SRR026675, and SRR026676 (*sets B*) to show the variance within the same experiment.

The plots themselves can be found in Appendix A. Figures 1, 2, and 3 show the positional mismatch, insertion and deletion error rates. We can see there are differences between the insertion error rates in sets A (Figure 1a) and even those from the same experiment (sets B, Figure 1b).

The reads from sets B were sequenced in 2008, being one of the first reads to be sequenced in paired-end mode outside Illumina ([Tho10]). The authors consider these reads to be of low quality mostly due to source prep. Newer but not yet published read sets have a much better quality, according to the authors. Note that while the consistently growing error rates in the order SRR026674, SRR026675, SRR026676 suggest instrument drift, the authors did not confirm this. The runs were not necessarily done in this order. Two runs were made at the same day, one on the next. A new flow cell is used for each run, so cleaning is not an issue either.

Positional insertion rates can be seen in Figure 2. The rates being 0 for the first and last base are caused by the alignment algorithm in RazerS: At the end of the program, the reads are aligned semiglobally against the reference sequence and the gap penalties are slightly larger than mismatch penalties. This means, alignments like these:

```
... CAACAAC-AACAACAACAA-CAACAACAACAA ...
             ||||||||||
           AAACAACAACAAA
```

are replaced with alignment like these:

```
... CAACAACAACAACAACAACAACAACAACAA ...
             ||||||||||
           AAACAACAACAAA
```

The half-moon shape of the error rates in Figure 2b for sets B also have to be explained to be alignment algorithm artifacts. Note that the insert rates are one order of magnitude lower than the mismatch rates. The closer a mismatch occurs towards the ending of an alignment, the higher is the probability that an inserted base shifts the leading or trailing bases to match to a non-significant but random match. In sets SRR049254, SRR038098, and SRR003673, the right tip of the half-moon is much larger than the left one. This can be explained by the fact that the insertion rates are much higher towards the end of the read, and for these reads, the mismatch error rate grows strong towards the end than for read sets B.

A similar explanation holds for the shapes of the deletion rate curves shown in Figure 3. We hope to correct such alignment algorithm biases by a multi read realignment program such as seqcons [RKD+09] in the future. Currently, bugs in this program prevent us from using it in a whole-genome setting.

We consider the insert and deletion probabilities to be independent of the position and calibrate them with average rates from the middle of the analyzed reads. Thus, our error model for Illumina reads is:

- The read length ℓ is constant.
- The probability for mismatches at position i in the read is given by $p_m(i)$.
- The probability for insertions and deletions is the same at each position and given by p_i and p_d.
- On mismatches, the replacing base is chosen at random, excluding the replaced base.

The position dependent mismatch probability can be given explicitely as an empirical curve. Another possibility is to specify it as a two-piece affine function in the position relative to the read length. From 0 up to a position x with a flat slope and from x to 1 with a steeper slope. The empirical data suggests 2/3 as the value for x.

2.1.3. Positional Qualities

Figures 4, 5, 6, 7, 8 show the positional mean qualities of matching, mismatching, inserted bases and those of the bases before/after deleted bases. The vertical bars show the standard deviations.

We first note that the qualities before and after deleted bases hardly differ. Second, the qualities for inserted bases roughly follow those of neighbours of deleted bases. Third, the qualities of matching and mismatching bases don't differ much at the beginning but quickly separatetowards the ending. The qualities of inserted and deleted bases lie in between, first following the quality of matching bases up the relative position of 1/3.

In Section 2.1.2, we described methodological problems of determining whether a base is inserted/deleted or mismatching. Because of this, inserted and deleted bases could in fact be mismatching or matching bases. For now, we decide only to differentiate between mismatches and the rest of error types. Hopefully, a realignment step can clarify whether this is adequate or not.

We decide to model the qualities as position specific normal distributions. One for mismatching bases, and one for all other bases. The position depending means follow a linearly falling ramps, the standard deviations follow a linearly raising ramp.

2.2. Model for 454 LS Reads

454 pyrosequencing works by synthesizing one strand of a read homopolymer by homopolymer (where a homopolymer is the repetition of the same base). For each synthesized hompolymer, light is emitted with a light intensity depending on its length ℓ. Background noise is simulated as light with log-normally distributed intensity.

In the original publication [MEA+05], the authors give normal distributions with a mean of ℓ and standard deviation of $k \cdot \ell$ where k is a fixed proportionality factor. The read simulator MetaSim, described in [ROA+08] models the standard deviation as by $k \cdot \ell$ by default, to be more consistent with "basic statistics." We do the same. We model background noise as a log-normal distribution with mean 0.2 and standard deviation 0.1. We remind the reader that often, the formula for the log-normal distribution function is given with the mean and standard deviation of the underlying normal distribution. In this case, mean and standard deviation of the log-normal distribution have to be transformed.

In the simulation, the light intensities are simulated following the normal distributions, with an additive background noise. Consistent with MetaSim and flowsim [BML+10], we call bases within one flow cycle using a simple Bayes base caller. We use the log-odds transformed error probability as returned by the Bayes base caller as quality values, as does [BML+10].

2.3. Model for Sanger Reads

For Sanger reads, we follow the model from celsim as proposed in [Mye99]:

- Read lengths are either uniformly sampled from an interval or normally distributed with given mean and standard deviation.
- Insertions, deletions and mismatches are randomly distributed with position dependent probabilities. The position dependent probabilities are computed by ramp functions with configureable probabilities at the beginning and the end.

- The quality simulation is the same as for Illumina reads, as described in Section 2.1.3.

3. Simulation Framework

This section gives a short overview of the implementation. The read simulation framework works as follows:

First, the reference sequence is loaded. Alternatively, a random sequence with a given background distribution can be generated.

Second, model specific parameters are computed or loaded from a file. For example, the empirical error distribution for Illumina reads can be loaded in this step.

Third, haplotypes are simulated from the reference sequence: The reference sequence is taken and changes are applied to it (actually, we only store a list with modifications for shorter running times). At each position, a base substition, an insertion, or a deletion is applied with user defined probabilites. The length of insertions and deletions is randomly picked, inserted bases are picked uniformly at random.

Fourth, the reads are simulated. Section 2 describes the parts that depend on the simulated technology. For each read:

- Pick the read length, depending on the simulation model.
- Pick physical sample location and haplotype.
- Depending on the simulation model, generate edit string and a buffer with inserted, substituted bases.
- Simulate qualities, depending on the edit string and the simulation model.
- If mate pairs are to be simulated then pick location for the mate and perform the upper steps for the mate.
- Add metainformation about the reads sample location, originally sample reference infix and edit string into the sequence descriptor.

Fifth, the reads are written out into a FASTA/FASTQ file. Optionally, the program writes out the alignment of the reads against the reference sequence in a SAM [LHW+09] file.

4. Conclusion and Future Work

4.1. Conclusion

We have presented a read read simulator for Illumina, 454 and Sanger reads. It incorporates position-specific error rates and base quality distributions. Its source code is freely available hat `http://www.seqan.de/projects/mason.html`.

4.2. Future Work

Using empirical distributions and simulating degradation for 454 reads following [BML+10] would be very useful. Furthermore, a future version should support the simulation of SOLiD color space reads.

The simulation of Helicos reads is another point. However, direct access to raw sequencing Helicos sequencing data would be necessary for this, similar to the work in [BML+10].

5. Acknowledgements

I thiank Anne-Kathrin Emde, David Weese, and Knut Reinert for enlightening discussions on 2GS technologies.

References

[BBS+08] David R Bentley, Shankar Balasubramanian, Harold P Swerdlow, Geoffrey P Smith, John Milton, Clive G Brown, Kevin P Hall, Dirk J Evers, Colin L Barnes, Helen R Bignell, Jonathan M Boutell, Jason Bryant, Richard J Carter, R Keira Cheetham, Anthony J Cox, Darren J Ellis, Michael R Flatbush, Niall A Gormley, Sean J Humphray, Leslie J Irving, Mirian S Karbelashvili, Scott M Kirk, Heng Li, Xiaohai Liu, Klaus S Maisinger, Lisa J Murray, Bojan Obradovic, Tobias Ost, Michael L Parkinson, Mark R Pratt, Isabelle M J Rasolonjatovo, Mark T Reed, Roberto Rigatti, Chiara Rodighiero, Mark T Ross, Andrea Sabot, Subramanian V Sankar, Aylwyn Scally, Gary P Schroth, Mark E Smith, Vincent P Smith, Anastassia Spiridou, Peta E Torrance, Svilen S Tzonev, Eric H Vermaas, Klaudia Walter, Xiaolin Wu, Lu Zhang, Mohammed D Alam, Carole Anastasi, Ify C Aniebo, David M D Bailey, Iain R Bancarz, Saibal Banerjee, Selena G Barbour, Primo A Baybayan, Vincent A Benoit, Kevin F Benson, Claire Bevis, Phillip J Black, Asha Boodhun, Joe S Brennan, John A Bridgham, Rob C Brown, Andrew A Brown, Dale H Buermann, Abass A Bundu, James C Burrows, Nigel P Carter, Nestor Castillo, Maria Chiara E Catenazzi, Simon Chang, R Neil Cooley, Natasha R Crake, Olubunmi O Dada, Konstantinos D Diakoumakos, Belen Dominguez-Fernandez, David J Earnshaw, Ugonna C Egbujor, David W Elmore, Sergey S Etchin, Mark R Ewan, Milan Fedurco, Louise J Fraser, Karin V Fuentes Fajardo, W Scott Furey, David George, Kimberley J Gietzen, Colin P Goddard, George S Golda, Philip A Granieri, David E Green, David L Gustafson, Nancy F Hansen, Kevin Harnish, Christian D Haudenschild, Narinder I Heyer, Matthew M Hims, Johnny T Ho, Adrian M Horgan, Katya Hoschler, Steve Hurwitz, Denis V Ivanov, Maria Q Johnson, Terena James, T A Huw Jones, Gyoung-Dong Kang, Tzvetana H Kerelska, Alan D Kersey, Irina Khrebtukova, Alex P Kindwall, Zoya Kings-

bury, Paula I Kokko-Gonzales, Anil Kumar, Marc A Laurent, Cynthia T Lawley, Sarah E Lee, Xavier Lee, Arnold K Liao, Jennifer A Loch, Mitch Lok, Shujun Luo, Radhika M Mammen, John W Martin, Patrick G McCauley, Paul McNitt, Parul Mehta, Keith W Moon, Joe W Mullens, Taksina Newington, Zemin Ning, Bee Ling Ng, Sonia M Novo, Michael J O'Neill, Mark A Osborne, Andrew Osnowski, Omead Ostadan, Lambros L Paraschos, Lea Pickering, Andrew C Pike, Alger C Pike, D Chris Pinkard, Daniel P Pliskin, Joe Podhasky, Victor J Quijano, Come Raczy, Vicki H Rae, Stephen R Rawlings, Ana Chiva Rodriguez, Phyllida M Roe, John Rogers, Maria C Rogert Bacigalupo, Nikolai Romanov, Anthony Romieu, Rithy K Roth, Natalie J Rourke, Silke T Ruediger, Eli Rusman, Raquel M Sanches-Kuiper, Martin R Schenker, Josefina M Seoane, Richard J Shaw, Mitch K Shiver, Steven W Short, Ning L Sizto, Johannes P Sluis, Melanie A Smith, Jean Ernest Sohna Sohna, Eric J Spence, Kim Stevens, Neil Sutton, Lukasz Szajkowski, Carolyn L Tregidgo, Gerardo Turcatti, Stephanie Vandevondele, Yuli Verhovsky, Selene M Virk, Suzanne Wakelin, Gregory C Walcott, Jingwen Wang, Graham J Worsley, Juying Yan, Ling Yau, Mike Zuerlein, Jane Rogers, James C Mullikin, Matthew E Hurles, Nick J McCooke, John S West, Frank L Oaks, Peter L Lundberg, David Klenerman, Richard Durbin, and Anthony J Smith. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218):53–9, November 2008.

[BML+10] S. Balzer, K. Malde, A. Lanzen, A. Sharma, and I. Jonassen. Characteristics of 454 pyrosequencing data–enabling realistic simulation with flowsim. *Bioinformatics*, 26(18):i420–i425, September 2010.

[DLBH08] Juliane C Dohm, Claudio Lottaz, Tatiana Borodina, and Heinz Himmelbauer. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic acids research*, 36(16):e105, September 2008.

[HWRE10] Manuel Holtgrewe, David Weese, Knut Reinert, and Anne-Kathrin Emde. Benchmark for Second-Generation Read Mapping. *Unpublished.*, 2010.

[LHW+09] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, and Richard Durbin. The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, 25(16):2078–9, August 2009.

[MEA+05] Marcel Margulies, Michael Egholm, William E Altman, Said Attiya, Joel S Bader, Lisa A Bemben, Jan Berka, Michael S Braverman, Yi-Ju Chen, Zhoutao Chen, Scott B Dewell, Lei Du, Joseph M Fierro, Xavier V Gomes, Brian C Godwin, Wen He, Scott Helgesen, Chun Heen Ho, Chun He Ho, Gerard P Irzyk, Szilveszter C Jando, Maria L I Alenquer, Thomas P Jarvie, Kshama B Jirage, Jong-Bum Kim, James R Knight, Janna R Lanza, John H Leamon, Steven M Lefkowitz, Ming Lei, Jing Li, Kenton L Lohman, Hong Lu, Vinod B Makhijani, Keith E McDade, Michael P McKenna, Eugene W

Myers, Elizabeth Nickerson, John R Nobile, Ramona Plant, Bernard P Puc, Michael T Ronan, George T Roth, Gary J Sarkis, Jan Fredrik Simons, John W Simpson, Maithreyan Srinivasan, Karrie R Tartaro, Alexander Tomasz, Kari A Vogt, Greg A Volkmer, Shally H Wang, Yong Wang, Michael P Weiner, Pengguang Yu, Richard F Begley, and Jonathan M Rothberg. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–80, September 2005.

[Mye99] Gene Myers. A dataset generator for whole genome shotgun sequencing. *Proceedings / ... International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology*, pages 202–10, January 1999.

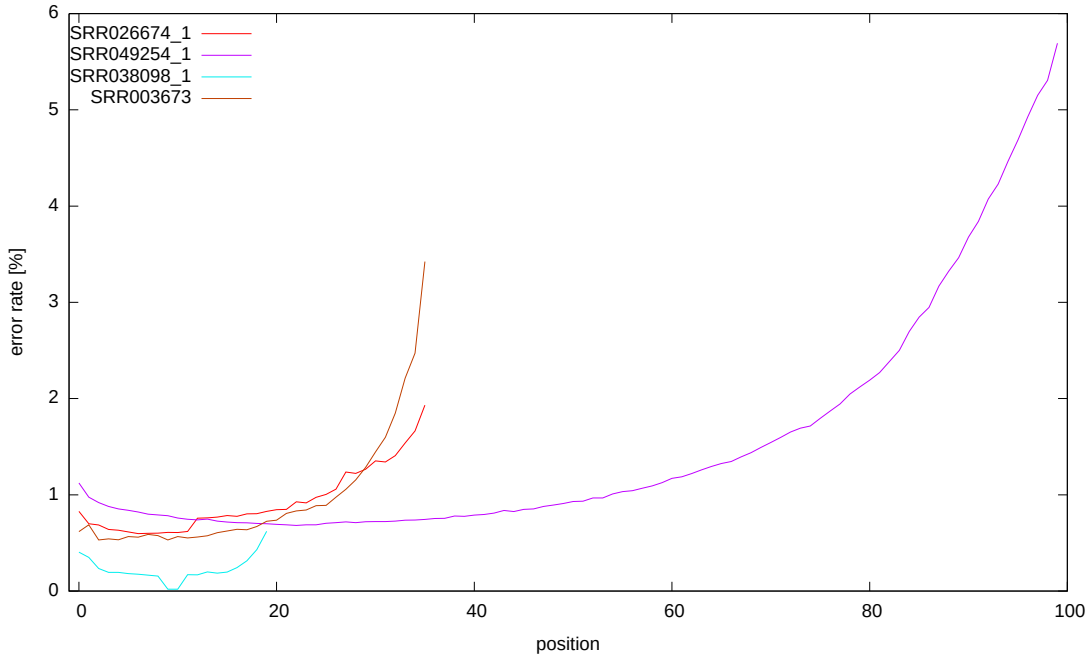
[RKD+09] Tobias Rausch, Sergey Koren, Gennady Denisov, David Weese, Anne-Katrin Emde, Andreas Döring, and Knut Reinert. A consistency-based consensus algorithm for de novo and reference-guided sequence assembly of short reads. *Bioinformatics (Oxford, England)*, 25(9):1118–24, May 2009.

[ROA+08] Daniel C Richter, Felix Ott, Alexander F Auch, Ramona Schmid, and Daniel H Huson. MetaSim: a sequencing simulator for genomics and metagenomics. *PloS one*, 3(10):e3373, January 2008.

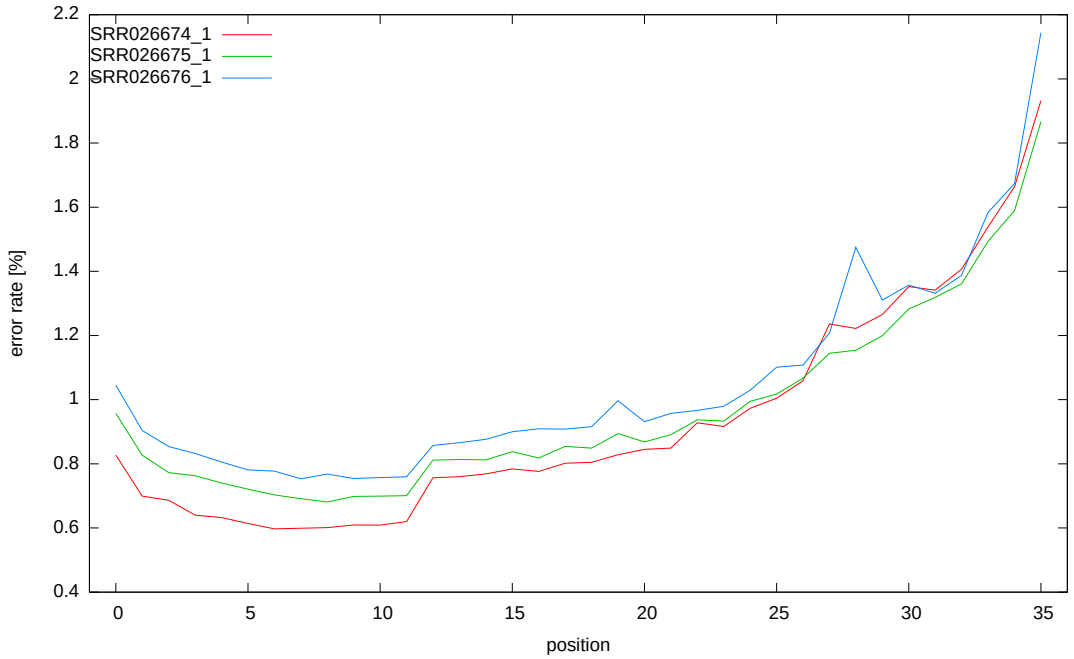
[Tho10] Kevin Thornton. Personal Communication, 2010.

[WER+09] David Weese, Anne-Katrin Emde, Tobias Rausch, Andreas Doring, and Knut Reinert. RazerS–fast read mapping with sensitivity control. *Genome Res*, 19(9):1646–1654, September 2009.

A. Positional Error Rate Plots

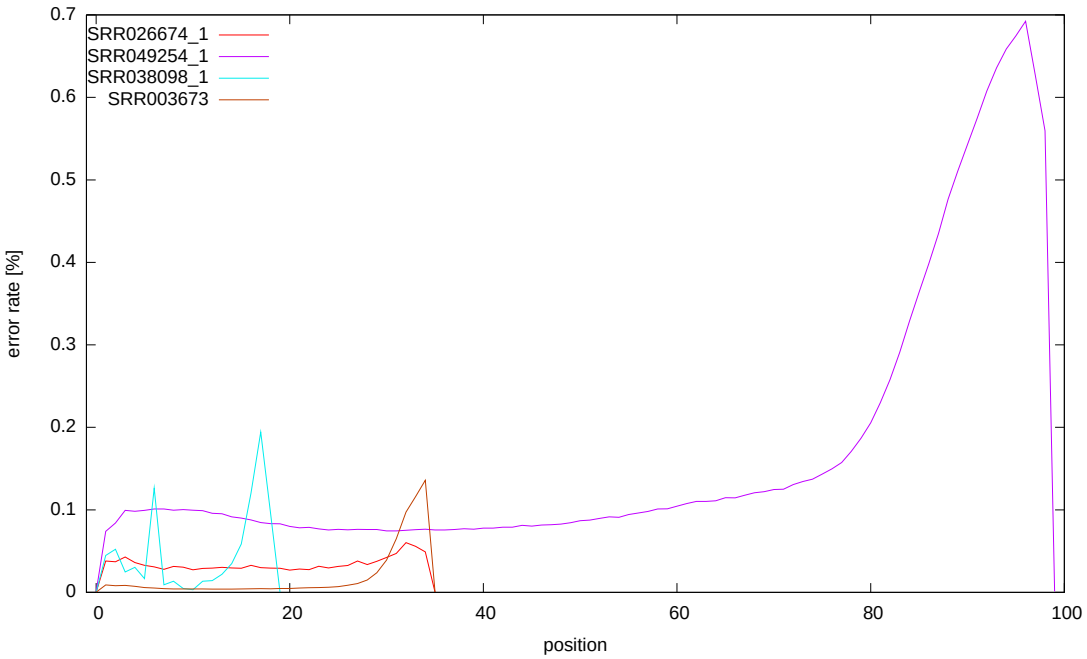


(a) Positional mismatch error rates in read sets A.

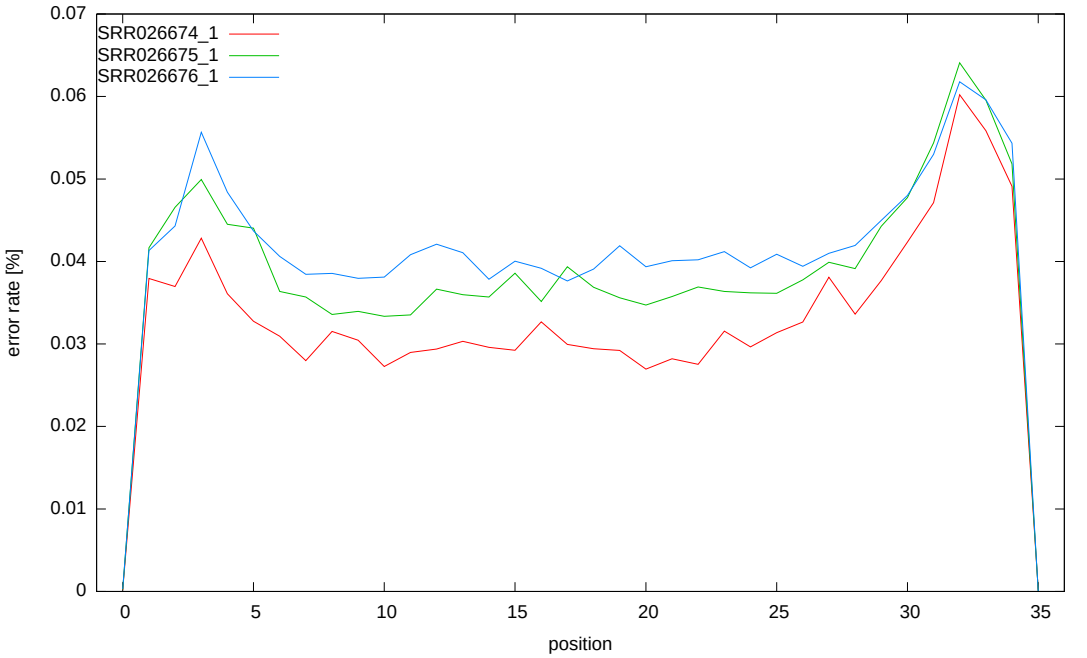


(b) Positional mismatch error rates in read sets B.

Figure 1: Positional mismatch error rates.

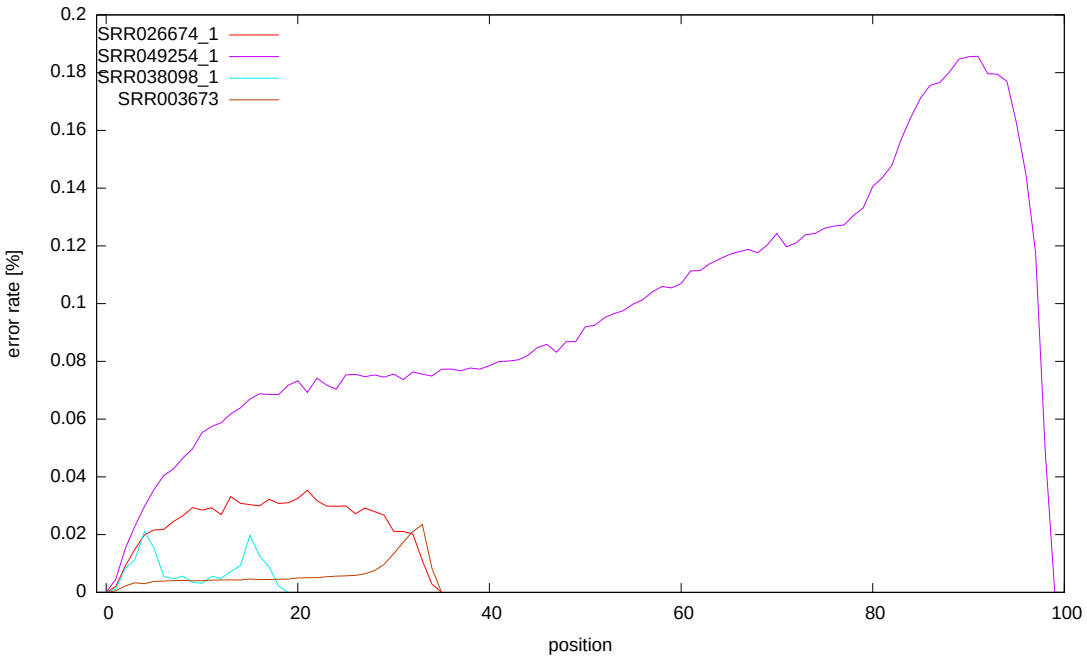


(a) Positional insert error rates in read sets A.

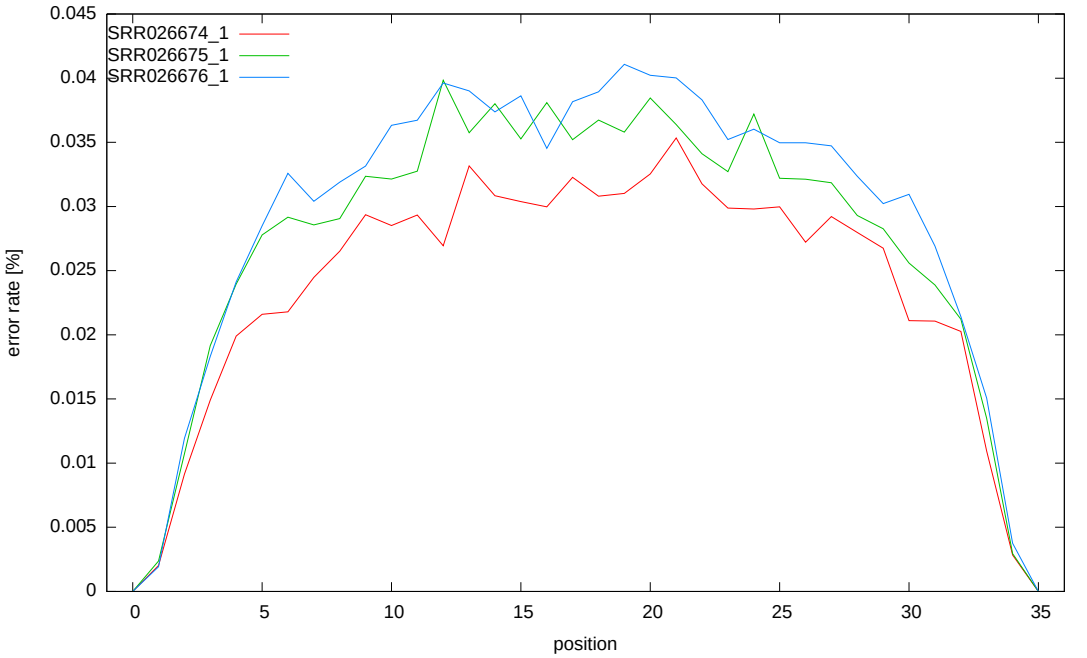


(b) Positional insert error rates in read sets B.

Figure 2: Positional insert error rates.



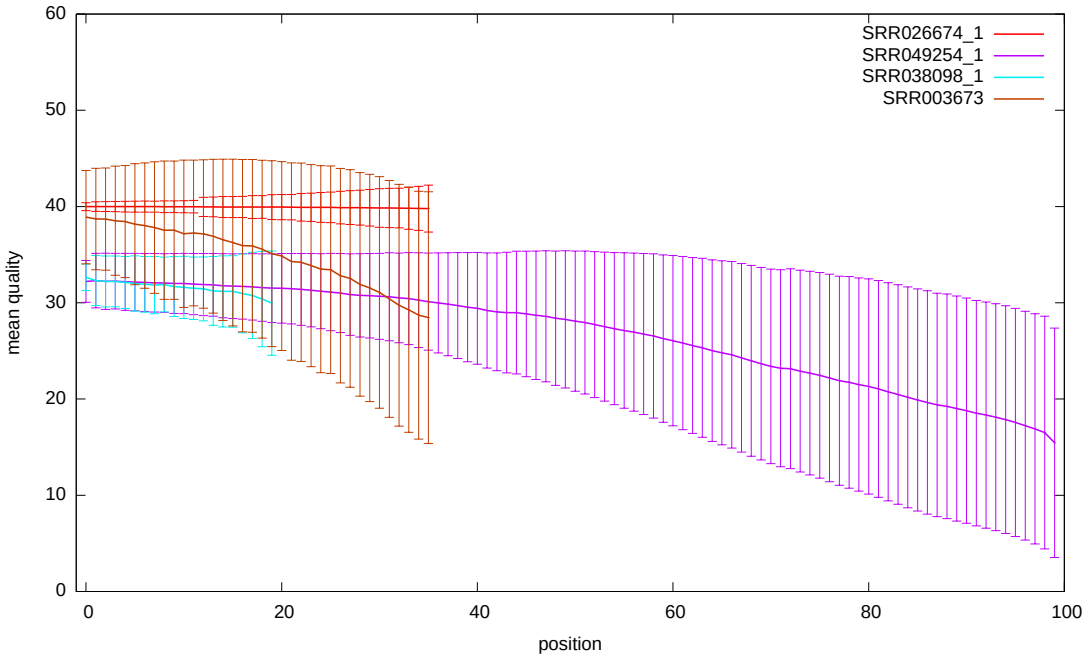
(a) Positional delete error rates in read sets A.



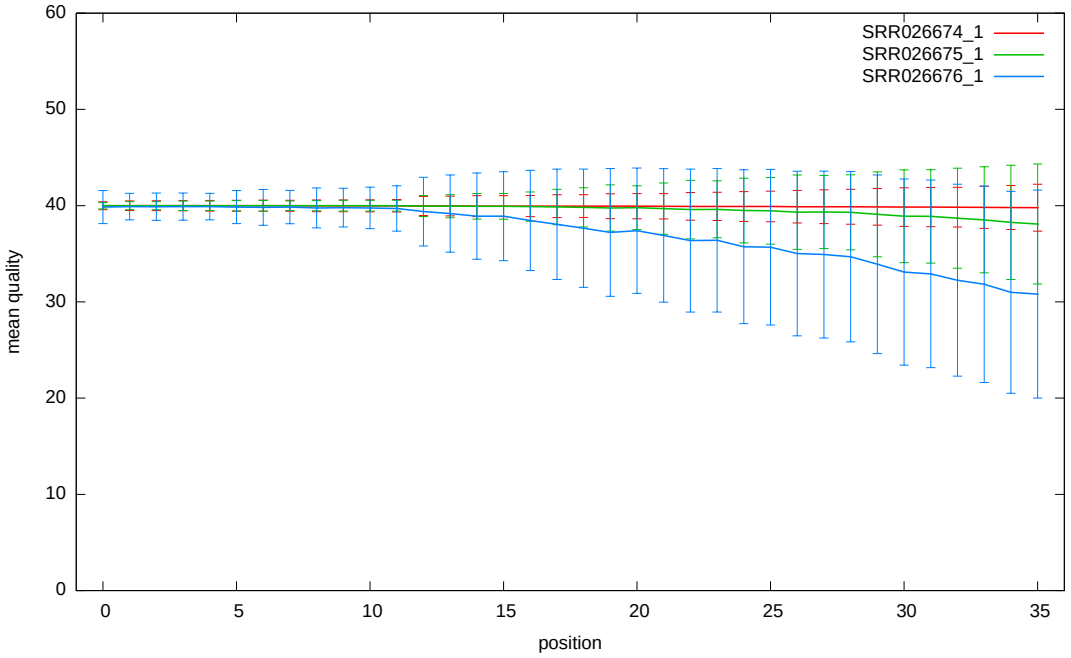
(b) Positional delete error rates in read sets B.

Figure 3: Positional delete error rates.

B. Positional Quality Value Plots

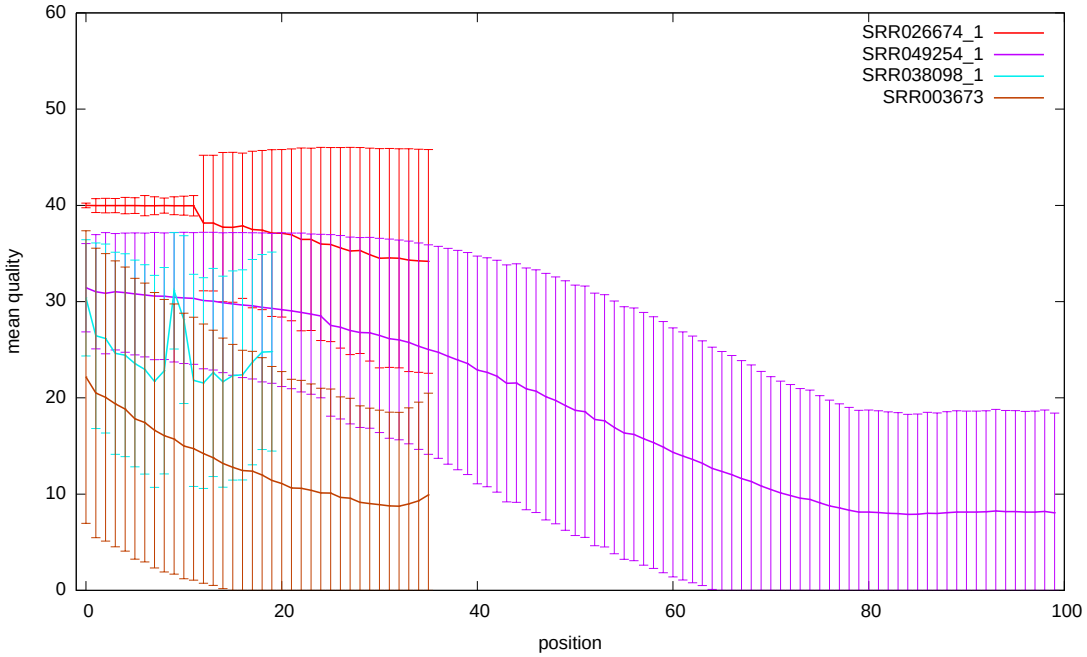


(a) Positional match quality values in read sets A.

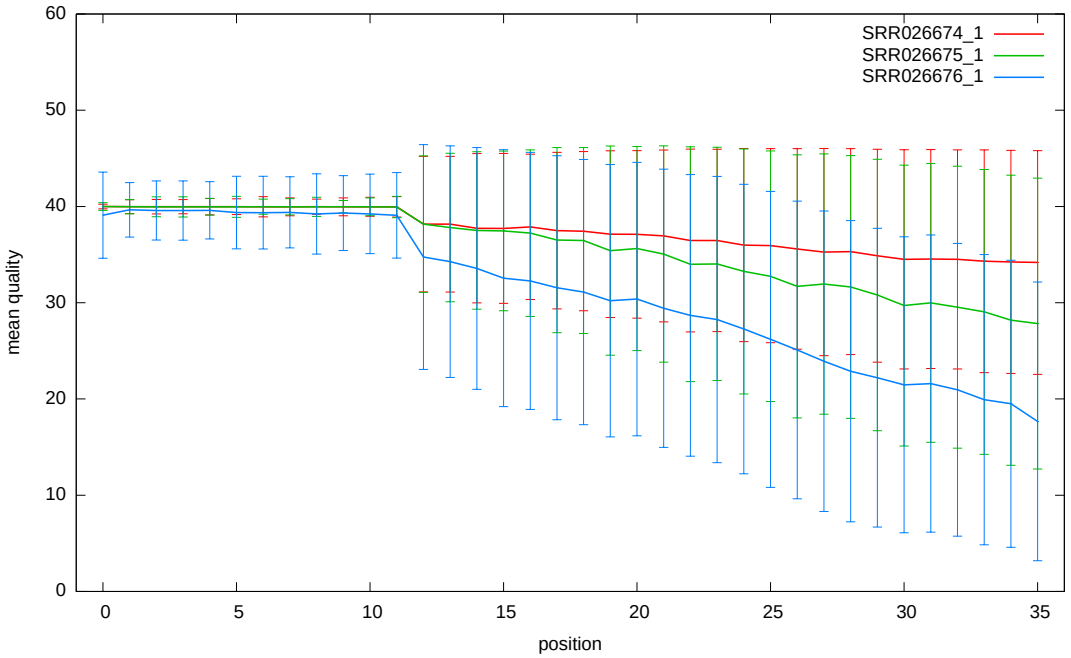


(b) Positional match quality values in read sets B.

Figure 4: Positional match base quality values.

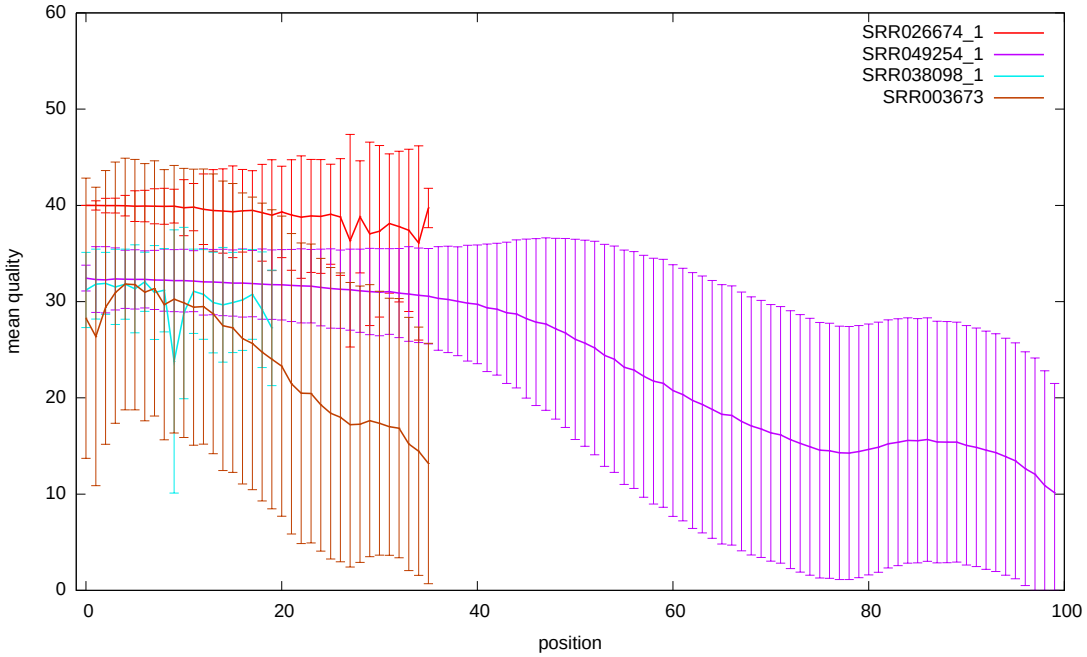


(a) Positional mismatch quality values in read sets A.

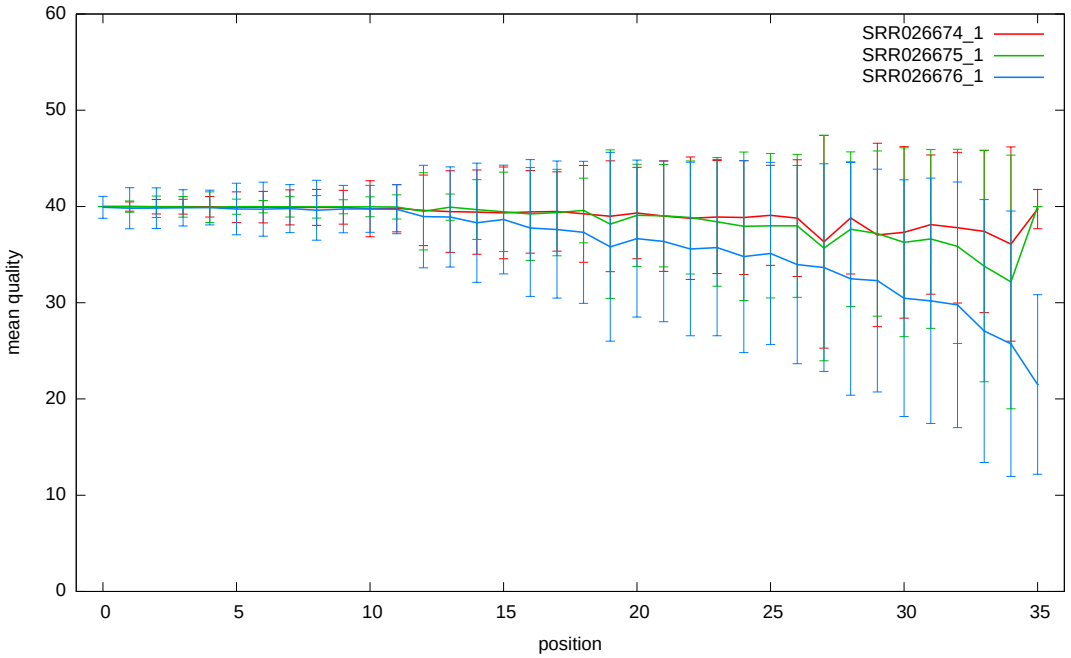


(b) Positional mismatch quality values in read sets B.

Figure 5: Positional mismatch base quality values.

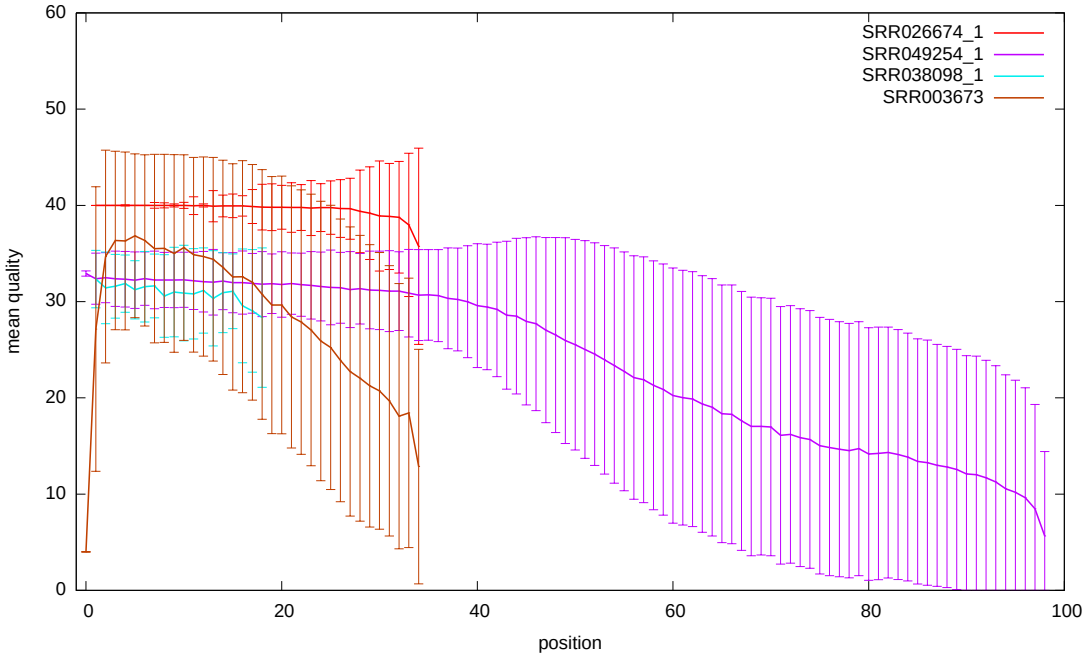


(a) Positional insert quality values in read sets A.

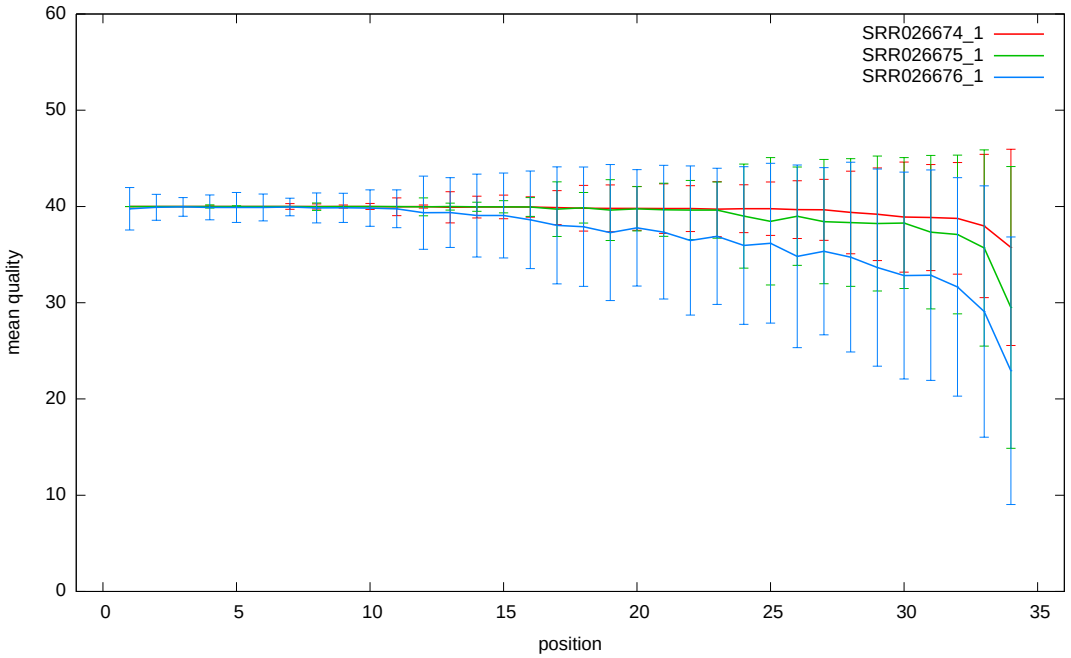


(b) Positional insert quality values in read sets B.

Figure 6: Positional inserted base quality values.

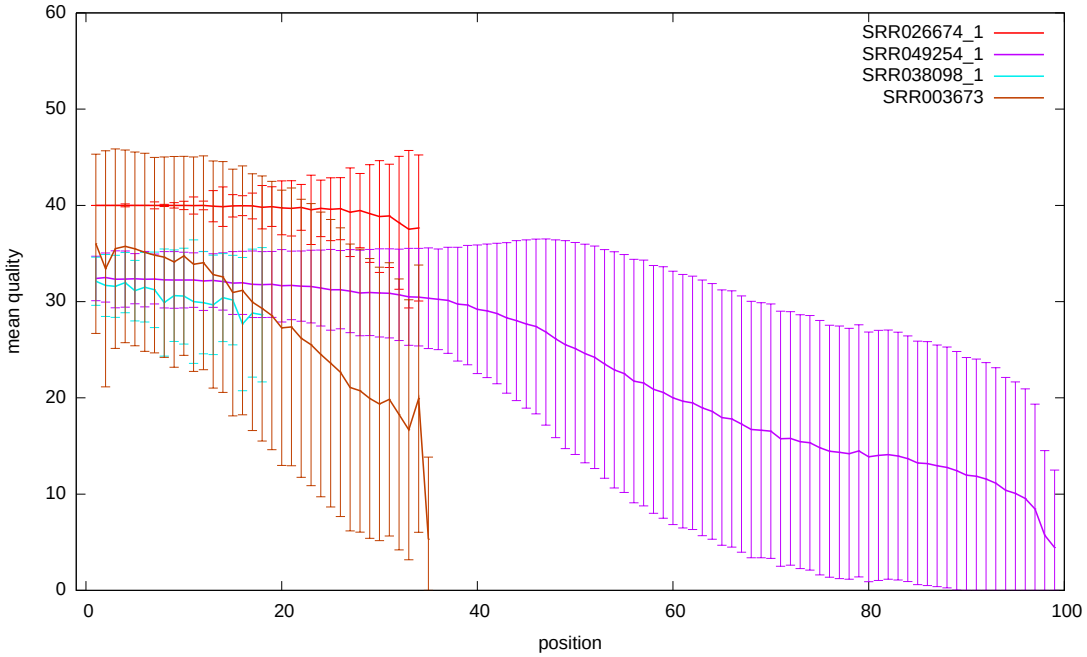


(a) Quality values of bases before deleted ones in read sets A.

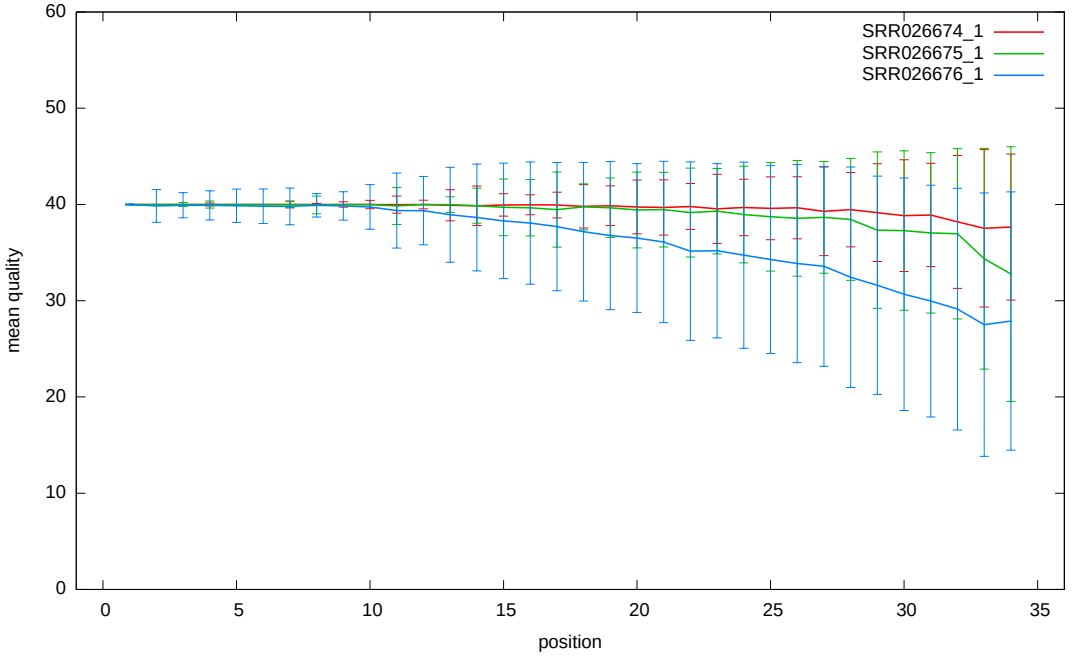


(b) Quality values of bases before deleted ones in read sets B.

Figure 7: Quality values of bases before deleted ones.



(a) Quality values of bases after deleted ones in read sets A.



(b) Quality values of bases after deleted ones in read sets B.

Figure 8: Quality values of bases after deleted ones.