11/06/2014

RNA Seq analysis

Cleaning

Plateforme ABiMS

RNA Seq analysis : cleanning

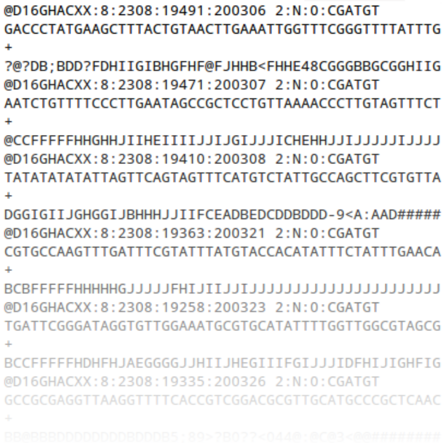
Data Cleaning

Why do we care about cleaning ?

?

Why do we care about cleaning ?

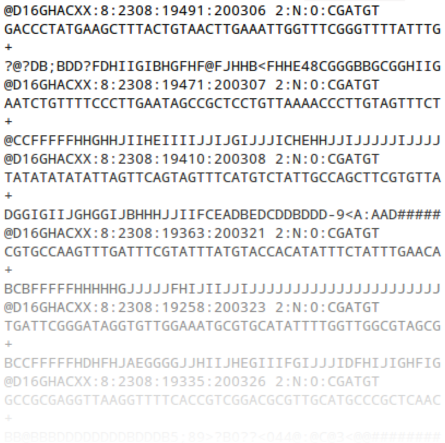
RAW SEQUENCES



```
@D16GHACXX:8:2308:19491:200306 2:N:0:CGATGT
GACCCTATGAAGCTTTACTGTAACTTGAAATTGGTTTCGGGTTTTATTTG
+
?@?DB;BDD?FDHIIGIBHGFHF@FJHHB<FHHE48CGGGBBGCGGHIIG
@D16GHACXX:8:2308:19471:200307 2:N:0:CGATGT
AATCTGTTTTCCCTTGAATAGCCGCTCCTGTTAAAACCCTTGTAGTTTCT
+
@CCFFFFFHHGHHJIIHEIIIIJJIJGIJJJICHEHHJJIJJJJJIJJJJ
@D16GHACXX:8:2308:19410:200308 2:N:0:CGATGT
TATATATATATTAGTTCAGTAGTTTCATGTCTATTGCCAGCTTCGTGTTA
+
DGGIGIIJGHGGIJBHHHJJIIFCEADBEDCDDBDDD-9<A:AAD#####
@D16GHACXX:8:2308:19363:200321 2:N:0:CGATGT
CGTGCCAAGTTTGATTTCGTATTTATGTACCACATATTTCTATTTGAACA
+
BCBFFFFFHHHHHGJJJJJFHIJIIJJIJJJJJJJJJJJJJJJJJJJJJJ
@D16GHACXX:8:2308:19258:200323 2:N:0:CGATGT
TGATTCGGGATAGGTGTTGGAAATGCGTGCATATTTTGGTTGGCGTAGCG
+
BCCFFFFFHDHFHJAEGGGGJJHIIJHEGIIIFGIJJJIDFHIJIGHFIG
@D16GHACXX:8:2308:19335:200326 2:N:0:CGATGT
GCCGCGAGGTTAAGGTTTTCACCGTCGGACGCGTTGCATGCCCGCTCAAC
+
BB@BBBDDDDDDDDBDDDB5;89>?B0??<044@:@C@3<@@########
```

Why do we care about cleaning ?

RAW SEQUENCES

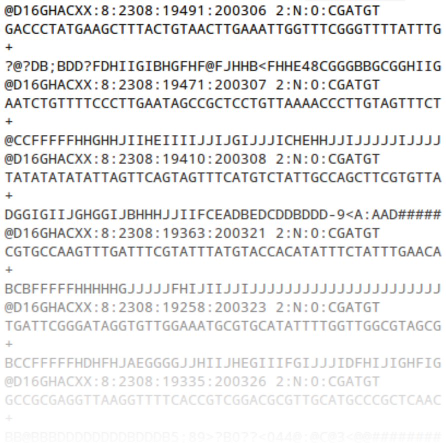


```
@D16GHACXX:8:2308:19491:200306 2:N:0:CGATGT
GACCCTATGAAGCTTTACTGTAACTTGAAATTGGTTTCGGGTTTTATTTG
+
?@?DB;BDD?FDHIIGIBHGFHF@FJHHB<FHHE48CGGGBBGCGGHIIG
@D16GHACXX:8:2308:19471:200307 2:N:0:CGATGT
AATCTGTTTTCCCTTGAATAGCCGCTCCTGTTAAAACCCTTGTAGTTTCT
+
@CCFFFFFHHGHHJIIHEIIIIJJIJGIJJJICHEHHJJIJJJJJIJJJJ
@D16GHACXX:8:2308:19410:200308 2:N:0:CGATGT
TATATATATATTAGTTCAGTAGTTTCATGTCTATTGCCAGCTTCGTGTTA
+
DGGIGIIJGHGGIJBHHHJJIIFCEADBEDCDDBDDD-9<A:AAD#####
@D16GHACXX:8:2308:19363:200321 2:N:0:CGATGT
CGTGCCAAGTTTGATTTCGTATTTATGTACCACATATTTCTATTTGAACA
+
BCBFFFFFHHHHHGJJJJJFHIJIIJJIJJJJJJJJJJJJJJJJJJJJJJ
@D16GHACXX:8:2308:19258:200323 2:N:0:CGATGT
TGATTCGGGATAGGTGTTGGAAATGCGTGCATATTTTGGTTGGCGTAGCG
+
BCCFFFFFHDHFHJAEGGGGJJHIIJHEGIIIFGIJJJIDFHIJIGHFIG
@D16GHACXX:8:2308:19335:200326 2:N:0:CGATGT
GCCGCGAGGTTAAGGTTTTCACCGTCGGACGCGTTGCATGCCCGCTCAAC
+
BB@BBBDDDDDDDDBDDDB5;89>?B0??<044@:@C@3<@@########
```

AMAZING TRANSCRIPTOME !!!

Why do we care about cleaning ?

RAW SEQUENCES

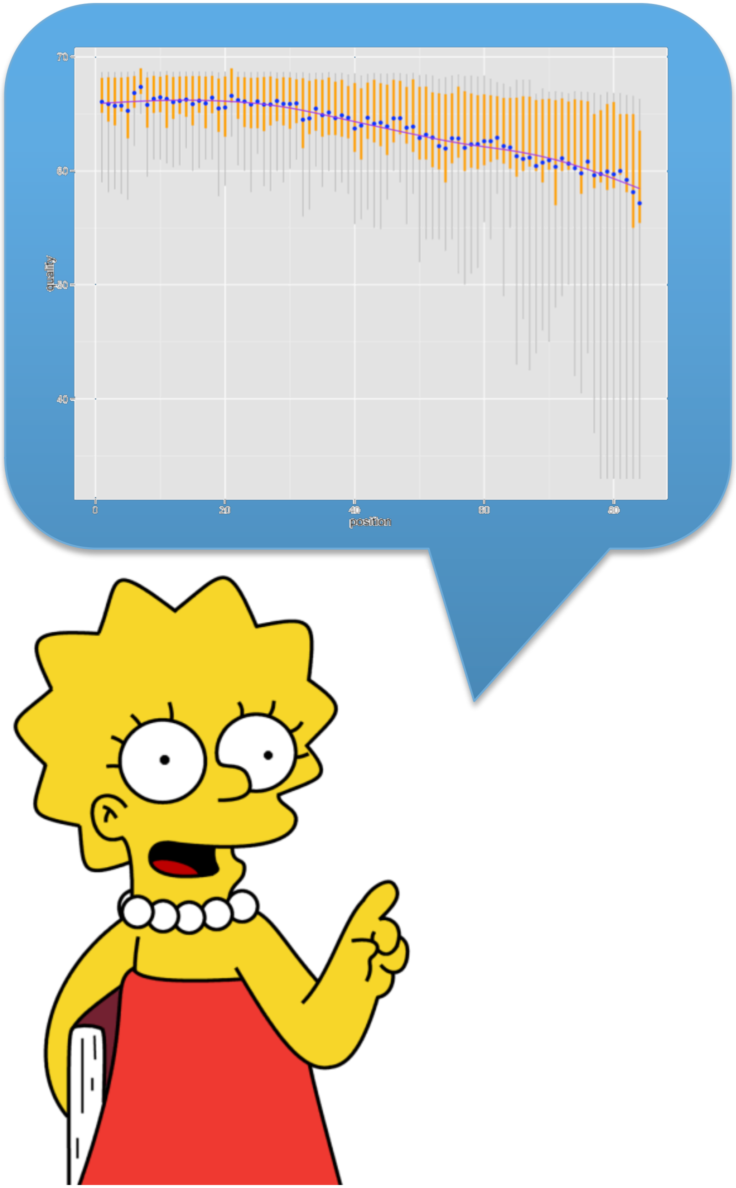


```
@D16GHACXX:8:2308:19491:200306 2:N:0:CGATGT
GACCCTATGAAGCTTTACTGTAACTTGAAATTGGTTTCGGGTTTTATTTG
+
?@?DB;BDD?FDHIIGIBHGFHF@FJHHB<FHHE48CGGGBBGCGGHIIG
@D16GHACXX:8:2308:19471:200307 2:N:0:CGATGT
AATCTGTTTTCCCTTGAATAGCCGCTCCTGTTAAAACCCTTGTAGTTTCT
+
@CCFFFFFHHGHHJIIHEIIIIJJIJGIJJJICHEHHJJIJJJJJIJJJJ
@D16GHACXX:8:2308:19410:200308 2:N:0:CGATGT
TATATATATATTAGTTCAGTAGTTTCATGTCTATTGCCAGCTTCGTGTTA
+
DGGIGIIJGHGGIJBHHHJJIIFCEADBEDCDDBDDD-9<A:AAD#####
@D16GHACXX:8:2308:19363:200321 2:N:0:CGATGT
CGTGCCAAGTTTGATTTCGTATTTATGTACCACATATTTCTATTTGAACA
+
BCBFFFFFHHHHHGJJJJJFHIJIIJJIJJJJJJJJJJJJJJJJJJJJJJ
@D16GHACXX:8:2308:19258:200323 2:N:0:CGATGT
TGATTCGGGATAGGTGTTGGAAATGCGTGCATATTTTGGTTGGCGTAGCG
+
BCCFFFFFHDHFHJAEGGGGJJHIIJHEGIIIFGIJJJIDFHIJIGHFIG
@D16GHACXX:8:2308:19335:200326 2:N:0:CGATGT
GCCGCGAGGTTAAGGTTTTCACCGTCGGACGCGTTGCATGCCCGCTCAAC
+
BB@BBBDDDDDDDBDDDB5;89>?B0??<044@:@C@3<@@########
```

NO !!

AMAZING
TRANSCRIPTOME !!!

Because…



- Unknown nucleotides
- Bad quality nucleotides
- Adaptors and primers sub-sequences
- Poly A/T tails
- Low complexity sequences
- rRNA sequences
- Contaminant sequences
- Short length sequences

But also:

- Removing singletons
- In-silico normalization
- Sequencing errors correction
- …

But first… What data do we have ?

?

NGS sequences

- Illumina, 454 (Roche), Ion Torrent, Solid, …
- Single, Paired-end, Mate pairs
- Sequences length: 25, 35, 50, 75, 100, 150, 250, 500, 700, 800, … base pairs
- File format: Fastq Phred+33, Fastq Phred+64, 2 files (.fasta + .qual), Colorspace

RNAseq

- Illumina, 454 (Roche), Ion Torrent, Solid, …
- Single, Paired-end, Mate pairs
- Sequences length: 25, 35, 50, 75, 100, 150, 250, 500, 700, 800, … base pairs
- File format: Fastq Phred+33, Fastq Phred+64, 2 files (.fasta + .qual), Colorspace

RNAseq

- Illumina, 454 (Roche), Ion Torrent, Solid, …
- Single, Paired-end, Mate pairs
- Sequences length: 25, 35, 50, 75, 100, 150, 250, 500, 700, 800, … base pairs
- File format: Fastq Phred+33, Fastq Phred+64, 2 files (.fasta + .qual), Colorspace

NGS data Quality Checking (QC)

- These apply to all NGS data (not just RNAseq).

- Some of these problems can be worked around but others indicate that the lane is bad & must be re-run (or a new library is needed).

- Bias should be corrected in reverse order of their generation
  1. Sequencing biases (bad quality, unknowns)
  2. Library preparation
     a. Adaptors and primers sequences
     b. Poly A/T tails
  3. Biological sample (low complexity, rRNA, contaminants)

- Our favorite NGS QC tools is FastQC.

http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

1. Sequencing biases

- Unknown nucleotides (Ns)
- Bad quality nucleotides
- Hexamers biases (random priming) ? (Illumina. Now corrected ?)

- Why do we need to correct those ?
  - To remove a lot of sequencing errors (detrimental to the vast majority of assemblers)
  - Because most de-bruijn graph based assemblers can't handle unknown nucleotides

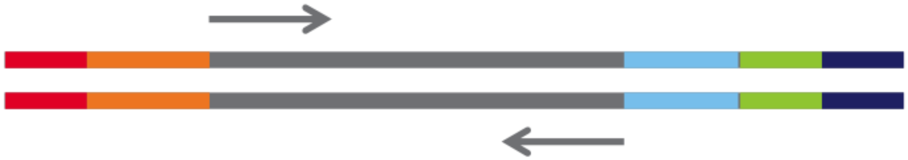
PRINSEQ

- http://prinseq.sourceforge.net/index.html
- Perl software for PReprocessing and INformation of SEQuence data
- Not the fastest, but very exhaustive
- 2 versions. We use the command-line version: prinseq_lite.pl
- But also: FASTX Toolkit, …

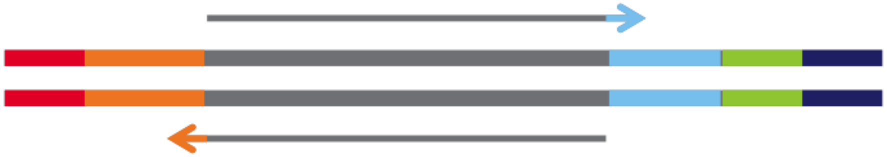
2. Adaptors & primers sequences

- Can be found in 3’ end if insert size is too short

Normal case:
insert size > sequencing length



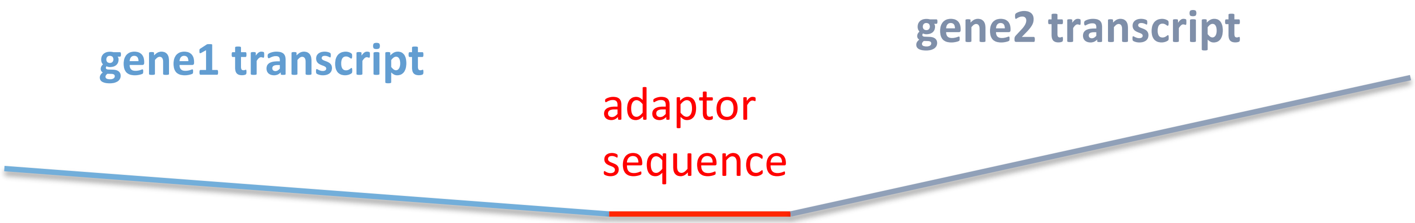
Abnormal case:
insert size < sequencing length



2. Adaptors & primers sequences

- Can be found in 3’ end if insert size is too short

- Why do we need to remove those ?
  - Because they can lead to “bridges” (links) between unrelated sequences (eg. 2 genes) and generate chimeras



Cutadapt

- http://code.google.com/p/cutadapt/

- Trimming of adaptors sequences from NGS data

- But also: trimmomatic, far, btrim, SeqTrim, TagCleaner, solexaQA, ...

3. Poly A/T tails, low complexity reads

- Some poly A/T tails can be left during library preparation

- Poly A/T or low complexity sequences can also lead to “bridges” between unrelated sequences and generate chimeras

```
>
ACGTAGCTACTAGCTGACGATTCCCGTAGATCATCGGATAAAAAAAAAAAAAAAAAAAAAAA
>
TTTTTTTTTTTTTTTTTTTTTTTTTTTACTGCGTAGCACATGGCTATTATTTCGGCCATCAA
>
CGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCGCG
>
ATGATGATGATGATGATGATGATGATGATGATGATGATGATGATGATGATGATGATGATGAT
```

PRINSEQ 2

- Trimming poly A/T tails
  - From 5’-end and 3’-end
  - w/ nucleotide nb >= 5
- Filtering low complexity sequences
  - Entropy < 70 (out of 100)
- Filtering short reads (< 50 nu)

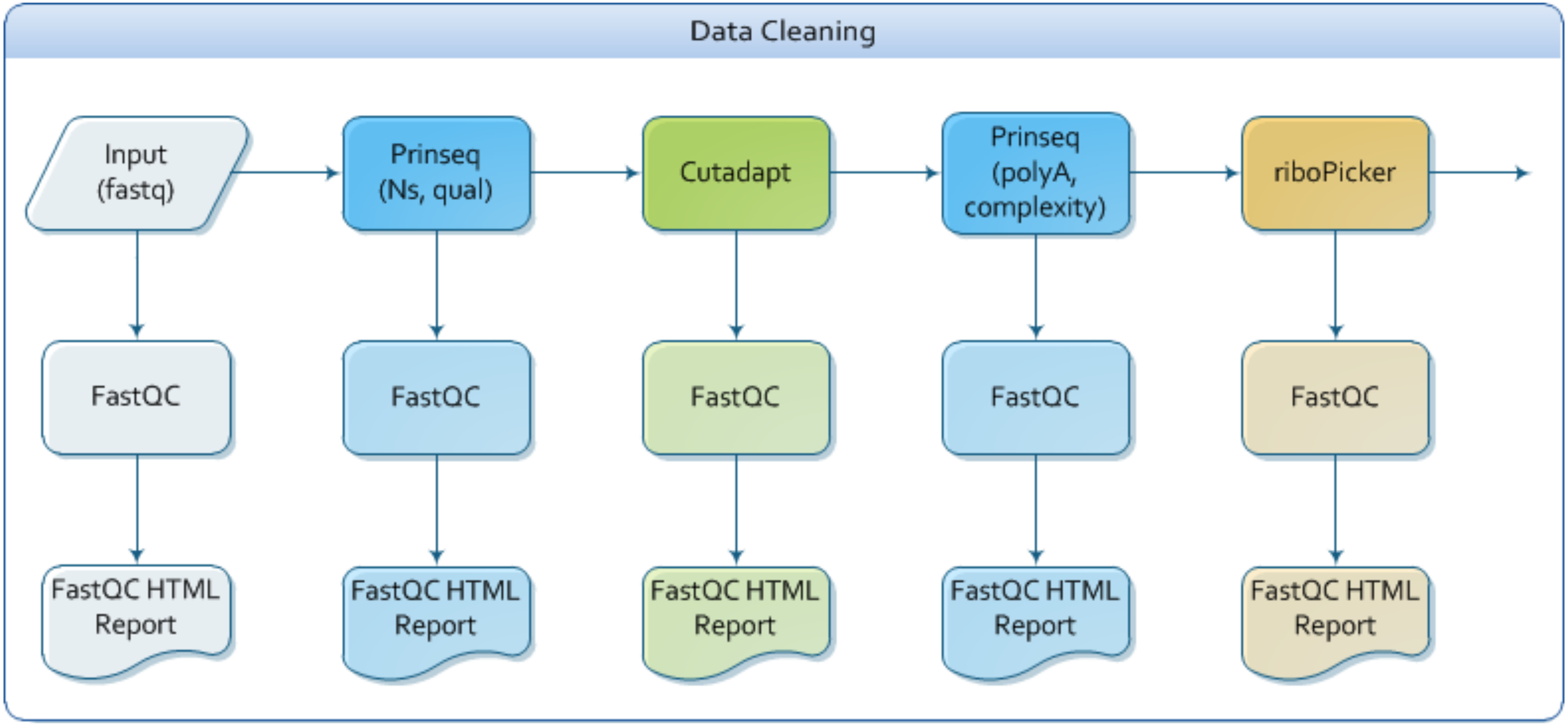
4. Contaminations

- Most RNA-seq libraries comprise ribosomal RNA that you may want to remove

- Contaminations can also occur with foreign RNA/DNA (PhiX, Bacteria, …)

riboPicker

- http://ribopicker.sourceforge.net/
- Easy identification and removal of rRNA-like sequences
- For RNAseq and DNAseq
- But also: SortMeRNA, DeconSeq, …


Data Cleaning
Input
(fastq)
Prinseq
(Ns, qual)
Cutadapt
Prinseq
(polyA,
complexity)
riboPicker
FastQC
FastQC
FastQC
FastQC
FastQC
FastQC HTML
Report
FastQC HTML
Report
FastQC HTML
Report
FastQC HTML
Report
FastQC HTML
Report

So… What data do **you** have ?

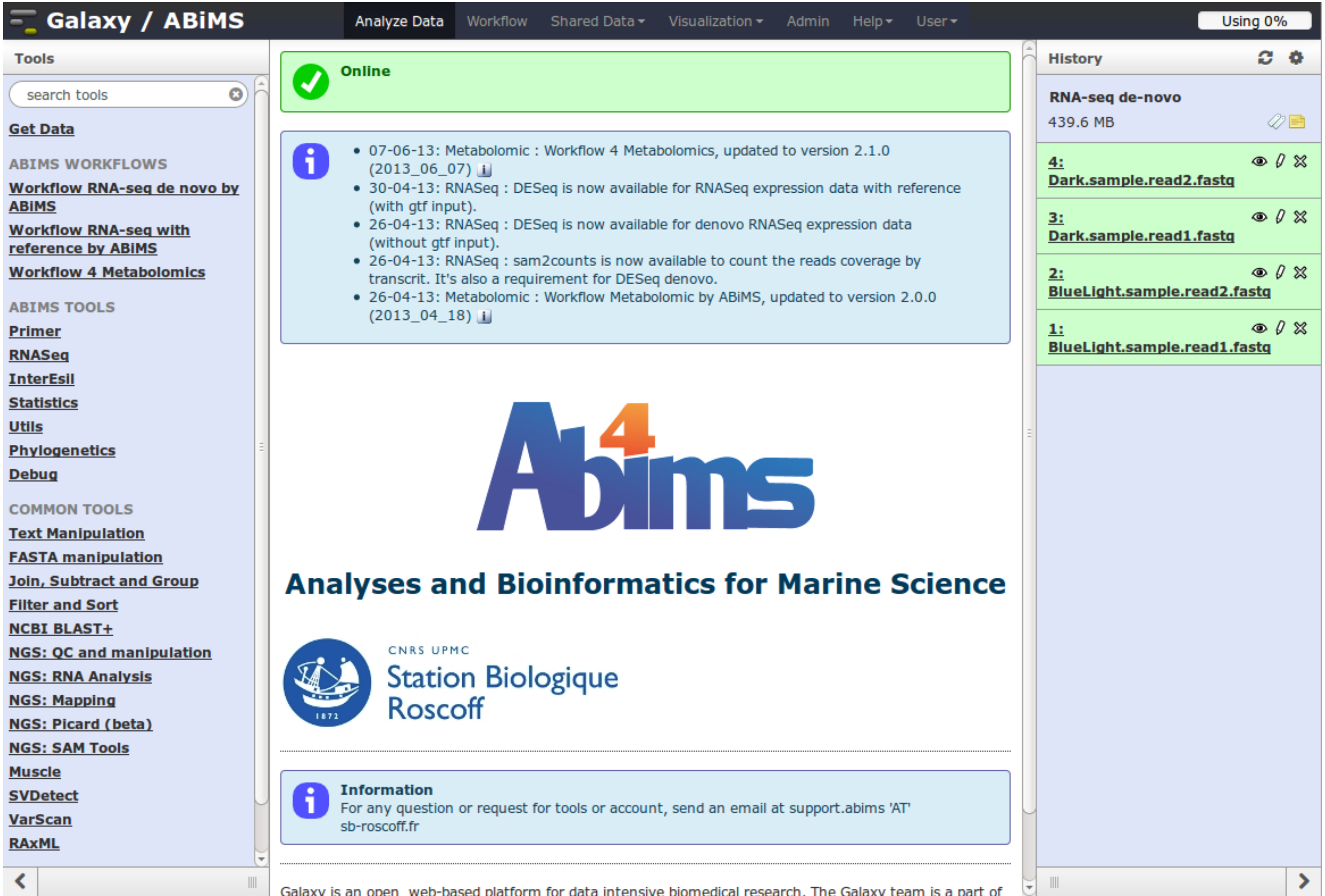
TP

But first, let’s retrieve it:

- History → Create New
- Shared Data → Data Libraries → RNA-seq de-novo
- Select all datasets and import to current history
- Name your new history

So... What data do you have ?

TP



Online

- 07-06-13: Metabolomic : Workflow 4 Metabolomics, updated to version 2.1.0 (2013_06_07)
- 30-04-13: RNASeq : DESeq is now available for RNASeq expression data with reference (with gtf input).
- 26-04-13: RNASeq : DESeq is now available for denovo RNASeq expression data (without gtf input).
- 26-04-13: RNASeq : sam2counts is now available to count the reads coverage by transcrit. It's also a requirement for DESeq denovo.
- 26-04-13: Metabolomic : Workflow Metabolomic by ABiMS, updated to version 2.0.0 (2013_04_18)

Analyses and Bioinformatics for Marine Science

CNRS UPMC
Station Biologique
Roscoff

Information
For any question or request for tools or account, send an email at support.abims 'AT' sb-roscoff.fr

Galaxy is an open, web-based platform for data intensive biomedical research. The Galaxy team is a part of

RNA-seq de-novo
439.6 MB

4: Dark.sample.read2.fastq

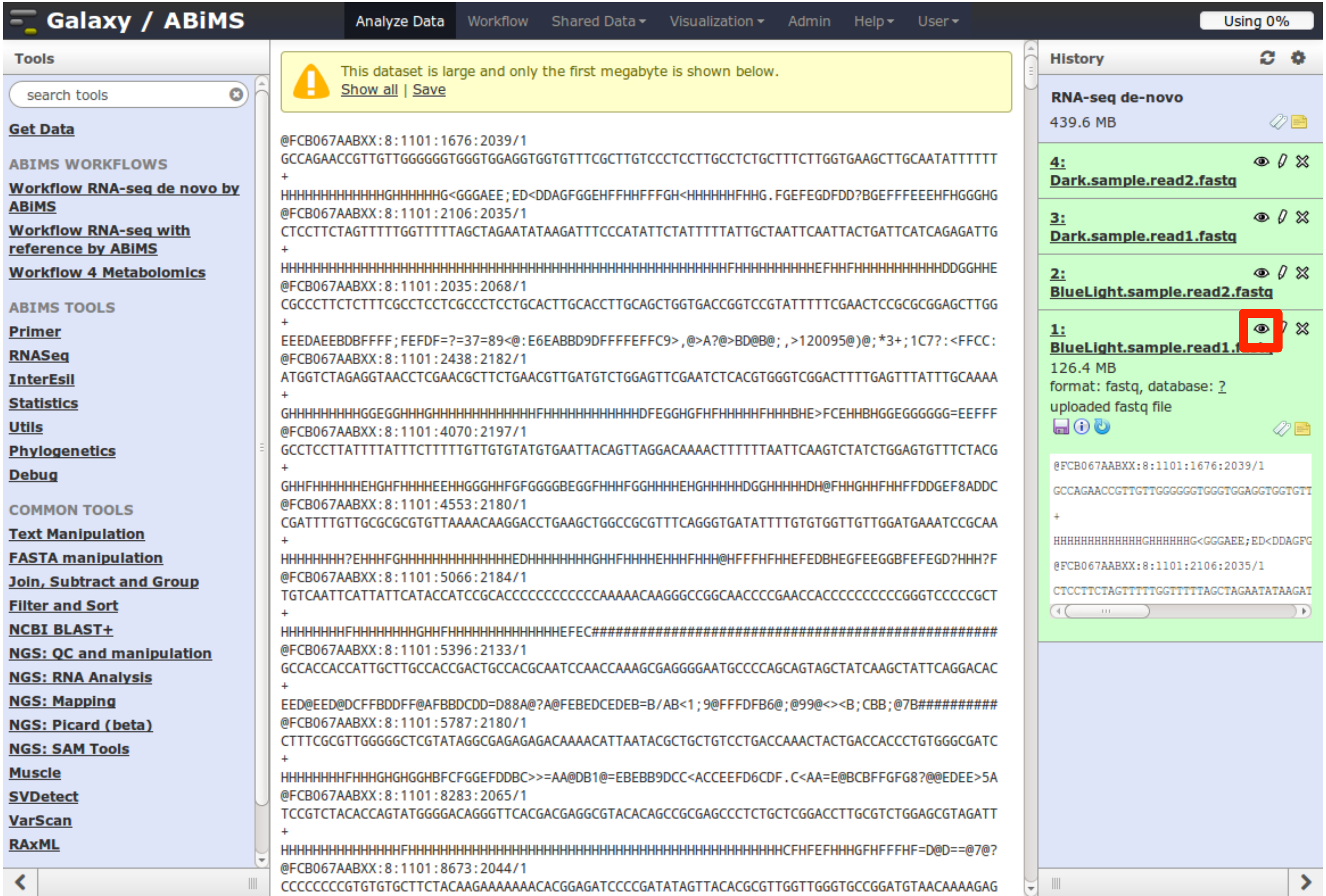
3: Dark.sample.read1.fastq

2: BlueLight.sample.read2.fastq

1: BlueLight.sample.read1.fastq

So... What data do **you** have ?

TP



This dataset is large and only the first megabyte is shown below.
Show all | Save

```
@FCB067AABXX:8:1101:1676:2039/1
GCCAGAACCGTTGTTGGGGGGTGGGTGGAGGTGGTGTTTCGCTTGTCCCTCCTTGCCTCTGCTTTCTTGGTGAAGCTTGCAATATTTTTT
+
HHHHHHHHHHHHHGHHHHHHG<GGGAEE;ED<DDAGFGGEHFFHHFFFGH<HHHHHHFHHG.FGEFEGDFDD?BGEFFFEEEHFHGGGHG
@FCB067AABXX:8:1101:2106:2035/1
CTCCTTCTAGTTTTTGGTTTTTAGCTAGAATATAAGATTTCCCATATTCTATTTTTATTGCTAATTCAATTACTGATTCATCAGAGATTG
+
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHFHHHHHHHHHHEFHHFHHHHHHHHHHHDDGGHHE
@FCB067AABXX:8:1101:2035:2068/1
CGCCCTTCTCTTTCGCCTCCTCGCCCTCCTGCACTTGCACCTTGCAGCTGGTGACCGGTCCGTATTTTTCGAACTCCGCGCGGAGCTTGG
+
EEEDAEEBDBFFFF;FEFDF=?=37=89<@:E6EABBD9DFFFFEFFC9>,@>A?@>BD@B@;,>120095@)@;*3+;1C7?:<FFCC:
@FCB067AABXX:8:1101:2438:2182/1
ATGGTCTAGAGGTAACCTCGAACGCTTCTGAACGTTGATGTCTGGAGTTCGAATCTCACGTGGGTCGGACTTTTGAGTTTATTTGCAAAA
+
GHHHHHHHHHGGEGGHHHGHHHHHHHHHHHHHHHFHHHHHHHHHHHHHDFEGGHGFHFHHHHHFHHHBHE>FCEHHBHGGEGGGGGG=EEFFF
@FCB067AABXX:8:1101:4070:2197/1
GCCTCCTTATTTTATTTCTTTTTGTTGTGTATGTGAATTACAGTTAGGACAAAACTTTTTTAATTCAAGTCTATCTGGAGTGTTTCTACG
+
GHHFHHHHHHHEHGHFHHHHEEHHGGGHHFGFGGGGBEGGFHHHFGGHHHHEHGHHHHHDGGHHHHHDH@FHHGHHFHHFFDDGEF8ADDC
@FCB067AABXX:8:1101:4553:2180/1
CGATTTTGTTGCGCGCGTGTTAAAACAAGGACCTGAAGCTGGCCGCGTTTCAGGGTGATATTTTGTGTGGTTGTTGGATGAAATCCGCAA
+
HHHHHHHH?EHHHFGHHHHHHHHHHHHHHHHEDHHHHHHHHGHHFHHHHEHHHFHHH@HFFFHFHHEFEDBHEGFEEGGBFEFEGD?HHH?F
@FCB067AABXX:8:1101:5066:2184/1
TGTCAATTCATTATTCATACCATCCGCACCCCCCCCCCCCAAAAACAAGGGCCGGCAACCCCGAACCACCCCCCCCCCGGGTCCCCCGCT
+
HHHHHHHHFHHHHHHHHHGHHFHHHHHHHHHHHHHHHEFEC##################################################
@FCB067AABXX:8:1101:5396:2133/1
GCCACCACCATTGCTTGCCACCGACTGCCACGCAATCCAACCAAAGCGAGGGGAATGCCCCAGCAGTAGCTATCAAGCTATTCAGGACAC
+
EED@EED@DCFFBDDFF@AFBBDCDD=D88A@?A@FEBEDCEDEB=B/AB<1;9@FFFDFB6@;@99@<><B;CBB;@7B##########
@FCB067AABXX:8:1101:5787:2180/1
CTTTCGCGTTGGGGGCTCGTATAGGCGAGAGAGACAAAACATTAATACGCTGCTGTCCTGACCAAACTACTGACCACCCTGTGGGCGATC
+
HHHHHHHHFHHHGHGHGGHBFCFGGEFDDBC>>=AA@DB1@=EBEBB9DCC<ACCEEFD6CDF.C<AA=E@BCBFFGFG8?@@EDEE>5A
@FCB067AABXX:8:1101:8283:2065/1
TCCGTCTACACCAGTATGGGGACAGGGTTCACGACGAGGCGTACACAGCCGCGAGCCCTCTGCTCGGACCTTGCGTCTGGAGCGTAGATT
+
HHHHHHHHHHHHHHHHFHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHCFHFEFHHHGFHFFFHF=D@D==@7@?
@FCB067AABXX:8:1101:8673:2044/1
CCCCCCCCGTGTGTGCTTCTACAAGAAAAAAACACGGAGATCCCCGATATAGTTACACGCGTTGGTTGGGTGCCGGATGTAACAAAAGAG
```

RNA-seq de-novo
439.6 MB

4: Dark.sample.read2.fastq

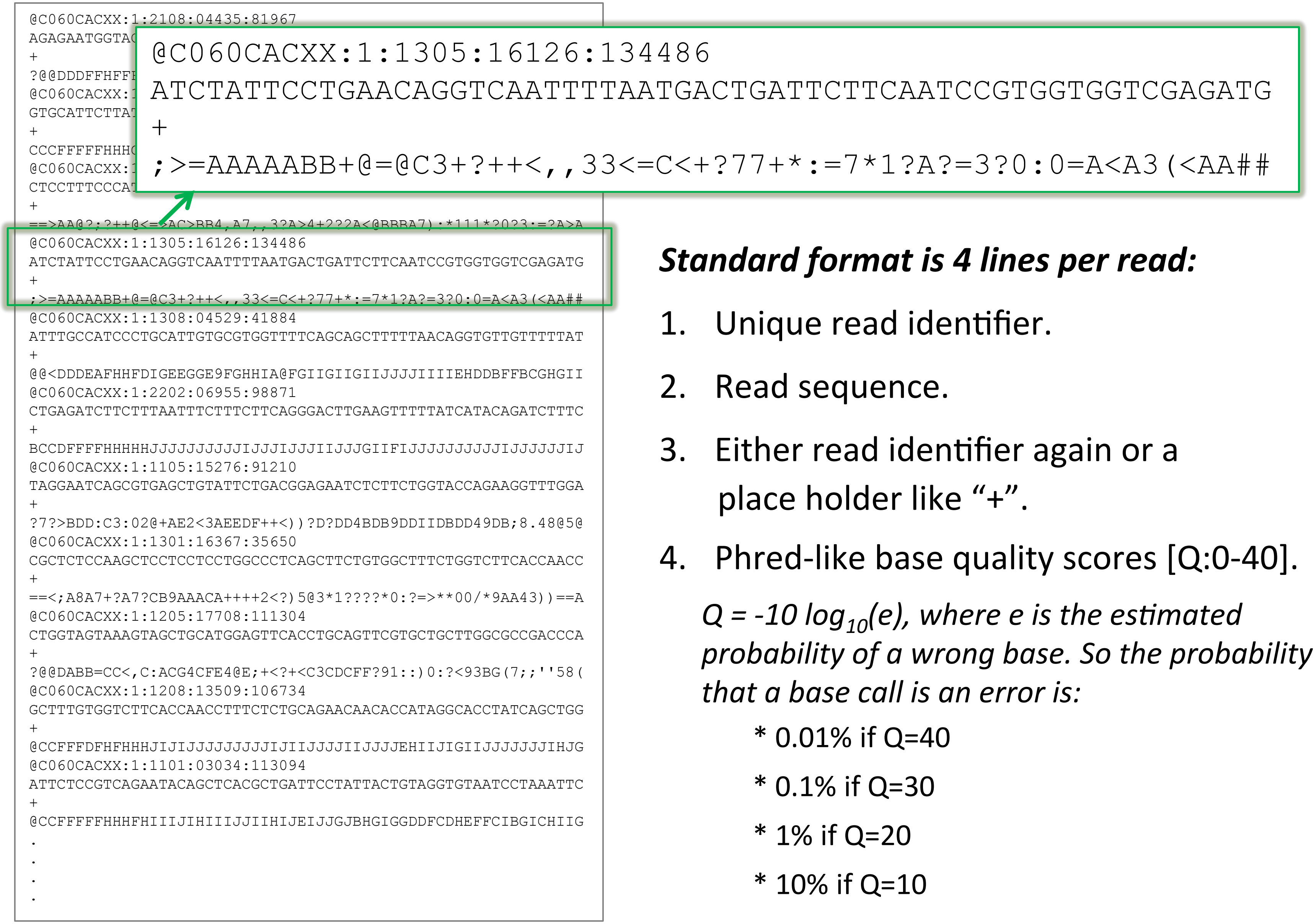
3: Dark.sample.read1.fastq

2: BlueLight.sample.read2.fastq

1: BlueLight.sample.read1.f
126.4 MB
format: fastq, database: ?
uploaded fastq file

```
@FCB067AABXX:8:1101:1676:2039/1
GCCAGAACCGTTGTTGGGGGGTGGGTGGAGGTGGTGTT
+
HHHHHHHHHHHHHGHHHHHHG<GGGAEE;ED<DDAGFG
@FCB067AABXX:8:1101:2106:2035/1
CTCCTTCTAGTTTTTGGTTTTTAGCTAGAATATAAGAT
```

NGS Data basics : FASTQ format, SE data



```
@C060CACXX:1:2108:04435:81967
AGAGAATGGTAC
+
?@@DDDFFHFFF
@C060CACXX:1
GTGCATTCTTAT
+
CCCFFFFFHHHC
@C060CACXX:1
CTCCTTTCCCAT
+
==>AA@?;?++@<=>AC>BB4,A7,,3?A>4+2?2A<@BBBA7):*111*?0?3:=?A>A
@C060CACXX:1:1305:16126:134486
ATCTATTCCTGAACAGGTCAATTTTAATGACTGATTCTTCAATCCGTGGTGGTCGAGATG
+
;>=AAAAABB+@=@C3+?++<,,33<=C<+?77+*:=7*1?A?=3?0:0=A<A3(<AA##
@C060CACXX:1:1308:04529:41884
ATTTGCCATCCCTGCATTGTGCGTGGTTTTCAGCAGCTTTTTAACAGGTGTTGTTTTTAT
+
@@<DDDEAFHHFDIGEEGGE9FGHHIA@FGIIGIIGIIJJJJIIIIEHDDBFFBCGHGII
@C060CACXX:1:2202:06955:98871
CTGAGATCTTCTTTAATTTCTTTCTTCAGGGACTTGAAGTTTTTATCATACAGATCTTTC
+
BCCDFFFFHHHHHJJJJJJJJJJIJJJIJJJIIJJJGIIFIJJJJJJJJJJIJJJJJJIJ
@C060CACXX:1:1105:15276:91210
TAGGAATCAGCGTGAGCTGTATTCTGACGGAGAATCTCTTCTGGTACCAGAAGGTTTGGA
+
?7?>BDD:C3:02@+AE2<3AEEDF++<))?D?DD4BDB9DDIIDBDD49DB;8.48@5@
@C060CACXX:1:1301:16367:35650
CGCTCTCCAAGCTCCTCCTCCTGGCCCTCAGCTTCTGTGGCTTTCTGGTCTTCACCAACC
+
==<;A8A7+?A7?CB9AAACA++++2<?)5@3*1????*0:?=>**00/*9AA43))==A
@C060CACXX:1:1205:17708:111304
CTGGTAGTAAAGTAGCTGCATGGAGTTCACCTGCAGTTCGTGCTGCTTGGCGCCGACCCA
+
?@@DABB=CC<,C:ACG4CFE4@E;+<?+<C3CDCFF?91::)0:?<93BG(7;;''58(
@C060CACXX:1:1208:13509:106734
GCTTTGTGGTCTTCACCAACCTTTCTCTGCAGAACAACACCATAGGCACCTATCAGCTGG
+
@CCFFFDFHFHHHJIJIJJJJJJJJJIJIIJJJJIIJJJJEHIIJIGIIJJJJJJJIHJG
@C060CACXX:1:1101:03034:113094
ATTCTCCGTCAGAATACAGCTCACGCTGATTCCTATTACTGTAGGTGTAATCCTAAATTC
+
@CCFFFFFHHHFHIIIJIHIIIJJIIHIJEIJJGJBHGIGGDDFCDHEFFCIBGICHIIG
.
.
.
.
```

```
@C060CACXX:1:1305:16126:134486
ATCTATTCCTGAACAGGTCAATTTTAATGACTGATTCTTCAATCCGTGGTGGTCGAGATG
+
;>=AAAAABB+@=@C3+?++<,,33<=C<+?77+*:=7*1?A?=3?0:0=A<A3(<AA##
```

Standard format is 4 lines per read:

1. Unique read identifier.
2. Read sequence.
3. Either read identifier again or a place holder like “+”.
4. Phred-like base quality scores [Q:0-40].

$Q = -10 \log_{10}(e)$, where e is the estimated probability of a wrong base. So the probability that a base call is an error is:

* 0.01% if Q=40

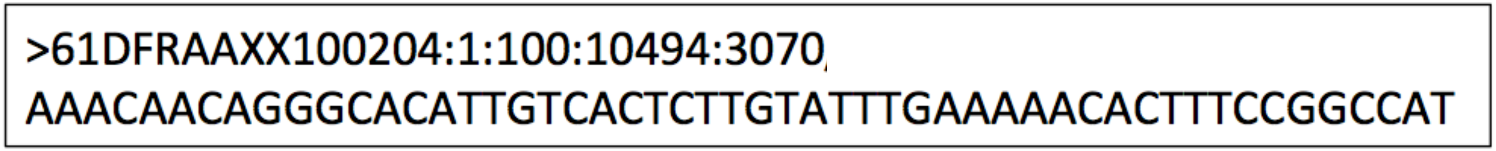
* 0.1% if Q=30

* 1% if Q=20

* 10% if Q=10

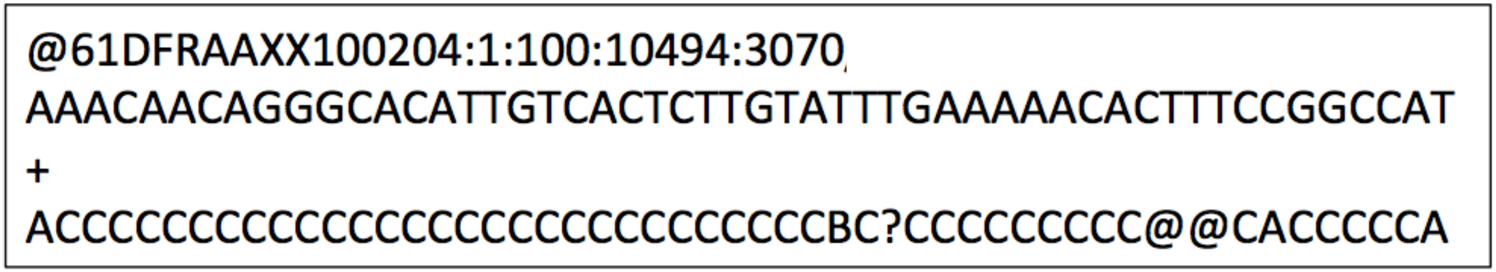
NGS Data basics : FASTQ format, SE data

FASTA format:



```
>61DFRAAXX100204:1:100:10494:3070
AAACAACAGGGCACATTGTCACTCTTGTATTTGAAAAACACTTTCCGGCCAT
```

FASTQ format:

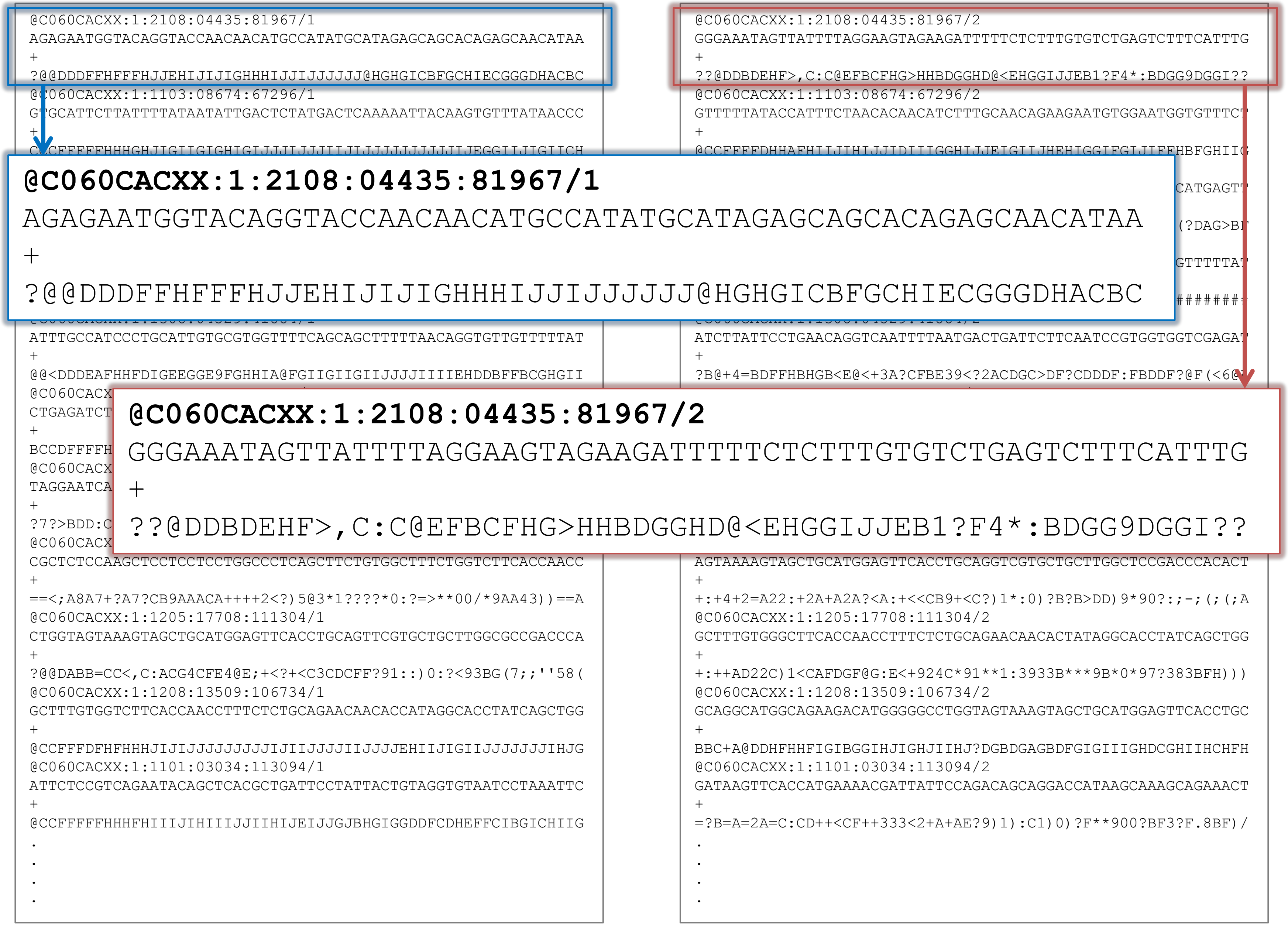


```
@61DFRAAXX100204:1:100:10494:3070
AAACAACAGGGCACATTGTCACTCTTGTATTTGAAAAACACTTTCCGGCCAT
+
ACCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBC?CCCCCCCCC@@CACCCCCA
```

Read

Quality values

NGS Data basics : FASTQ format, SE data



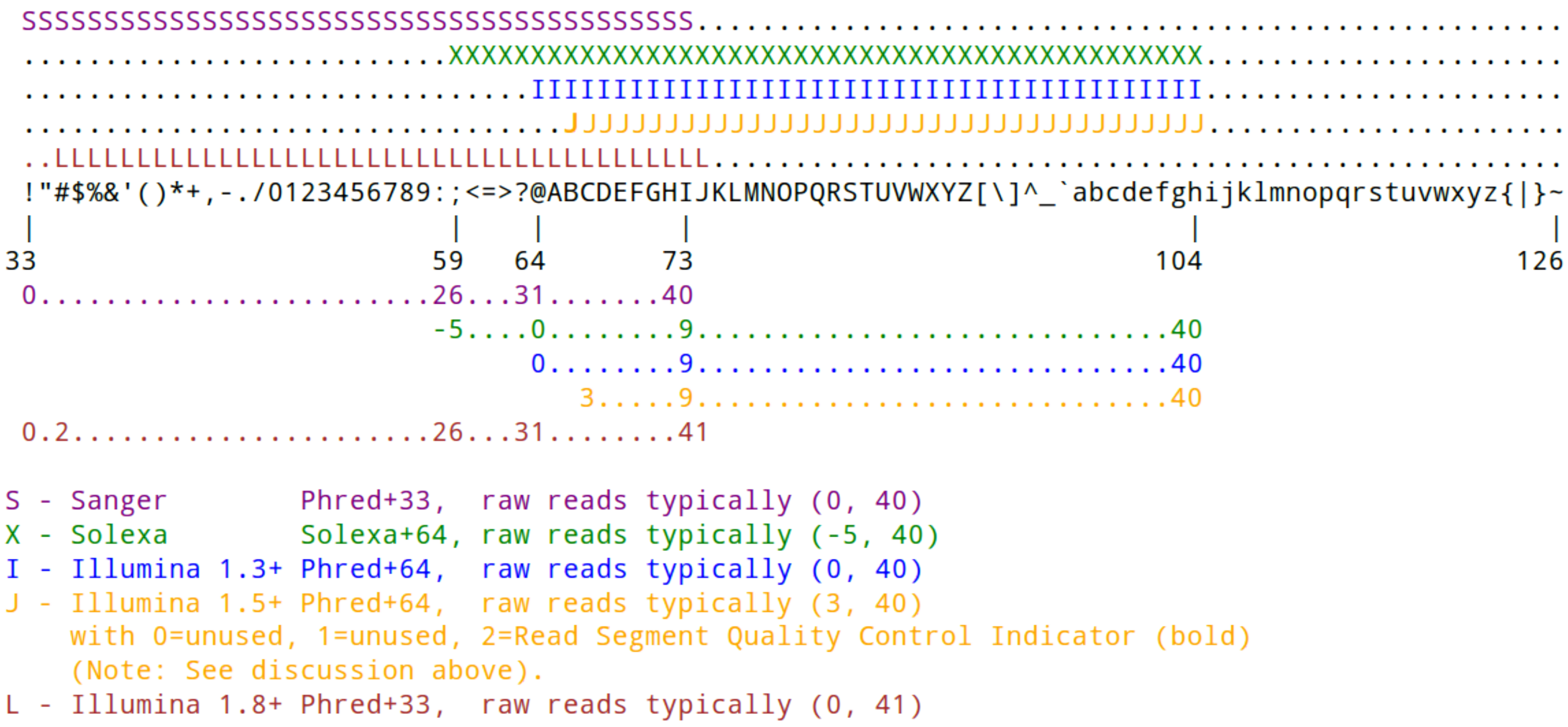
```
@C060CACXX:1:2108:04435:81967/1
AGAGAATGGTACAGGTACCAACAACATGCCATATGCATAGAGCAGCACAGAGCAACATAA
+
?@@DDDFFHFFFHJJEHIJIJIGHHHIJJIJJJJJJ@HGHGICBFGCHIECGGGDHACBC
@C060CACXX:1:1103:08674:67296/1
GTGCATTCTTATTTTATAATATTGACTCTATGACTCAAAAATTACAAGTGTTTATAACCC
+
CCCFFFFFHHHGHJIGIIGIGHIGIJJJIJJJIIJIJJJJJJJJJIJEGGIIJIGIICH
ATTTGCCATCCCTGCATTGTGCGTGGTTTTCAGCAGCTTTTTAACAGGTGTTGTTTTTAT
+
@@<DDDEAFHHFDIGEEGGE9FGHHIA@FGIIGIIGIIJJJJIIIIEHDDBFFBCGHGII
@C060CACX
CTGAGATCT
+
BCCDFFFFH
@C060CACX
TAGGAATCA
+
?7?>BDD:C
@C060CACX
CGCTCTCCAAGCTCCTCCTCCTGGCCCTCAGCTTCTGTGGCTTTCTGGTCTTCACCAACC
+
==<;A8A7+?A7?CB9AAACA++++2<?)5@3*1????*0:?=>**00/*9AA43))==A
@C060CACXX:1:1205:17708:111304/1
CTGGTAGTAAAGTAGCTGCATGGAGTTCACCTGCAGTTCGTGCTGCTTGGCGCCGACCCA
+
?@@DABB=CC<,C:ACG4CFE4@E;+<?+<C3CDCFF?91::)0:?<93BG(7;;''58(
@C060CACXX:1:1208:13509:106734/1
GCTTTGTGGTCTTCACCAACCTTTCTCTGCAGAACAACACCATAGGCACCTATCAGCTGG
+
@CCFFFDFHFHHHJIJIJJJJJJJJIJIIJJJJIIJJJJEHIIJIGIIJJJJJJJIHJG
@C060CACXX:1:1101:03034:113094/1
ATTCTCCGTCAGAATACAGCTCACGCTGATTCCTATTACTGTAGGTGTAATCCTAAATTC
+
@CCFFFFFHHHFHIIIJIHIIIJJIIHIJEIJJGJBHGIGGDDFCDHEFFCIBGICHIIG
.
.
.
.
```

```
@C060CACXX:1:2108:04435:81967/2
GGGAAATAGTTATTTTAGGAAGTAGAAGATTTTTCTCTTTGTGTCTGAGTCTTTCATTTG
+
??@DDBDEHF>,C:C@EFBCFHG>HHBDGGHD@<EHGGIJJEB1?F4*:BDGG9DGGI??
@C060CACXX:1:1103:08674:67296/2
GTTTTTATACCATTTCTAACACAACATCTTTGCAACAGAAGAATGTGGAATGGTGTTTCT
+
@CCFFFFDHHAFHIIJIHIJJIDIIIGGHIJJEIGIIJHEHIGGIFGIJIFFHBFGHIIG
CATGAGTT
(?DAG>BI
GTTTTTAT
########
ATCTTATTCCTGAACAGGTCAATTTTAATGACTGATTCTTCAATCCGTGGTGGTCGAGAT
+
?B@+4=BDFFHBHGB<E@<+3A?CFBE39<?2ACDGC>DF?CDDDF:FBDDF?@F(<6@
AGTAAAAGTAGCTGCATGGAGTTCACCTGCAGGTCGTGCTGCTTGGCTCCGACCCACACT
+
+:+4+2=A22:+2A+A2A?<A:+<<CB9+<C?)1*:0)?B?B>DD)9*90?:;-;(;(;A
@C060CACXX:1:1205:17708:111304/2
GCTTTGTGGGCTTCACCAACCTTTCTCTGCAGAACAACACTATAGGCACCTATCAGCTGG
+
+:++AD22C)1<CAFDGF@G:E<+924C*91**1:3933B***9B*0*97?383BFH)))
@C060CACXX:1:1208:13509:106734/2
GCAGGCATGGCAGAAGACATGGGGGCCTGGTAGTAAAGTAGCTGCATGGAGTTCACCTGC
+
BBC+A@DDHFHHFIGIBGGIHJIGHJIIHJ?DGBDGAGBDFGIGIIIGHDCGHIIHCHFH
@C060CACXX:1:1101:03034:113094/2
GATAAGTTCACCATGAAAACGATTATTCCAGACAGCAGGACCATAAGCAAAGCAGAAACT
+
=?B=A=2A=C:CD++<CF++333<2+A+AE?9)1):C1)0)?F**900?BF3?F.8BF)/
.
.
.
.
```

```
@C060CACXX:1:2108:04435:81967/1
AGAGAATGGTACAGGTACCAACAACATGCCATATGCATAGAGCAGCACAGAGCAACATAA
+
?@@DDDFFHFFFHJJEHIJIJIGHHHIJJIJJJJJJ@HGHGICBFGCHIECGGGDHACBC
```

```
@C060CACXX:1:2108:04435:81967/2
GGGAAATAGTTATTTTAGGAAGTAGAAGATTTTTCTCTTTGTGTCTGAGTCTTTCATTTG
+
??@DDBDEHF>,C:C@EFBCFHG>HHBDGGHD@<EHGGIJJEB1?F4*:BDGG9DGGI??
```

FASTQ quality encoding

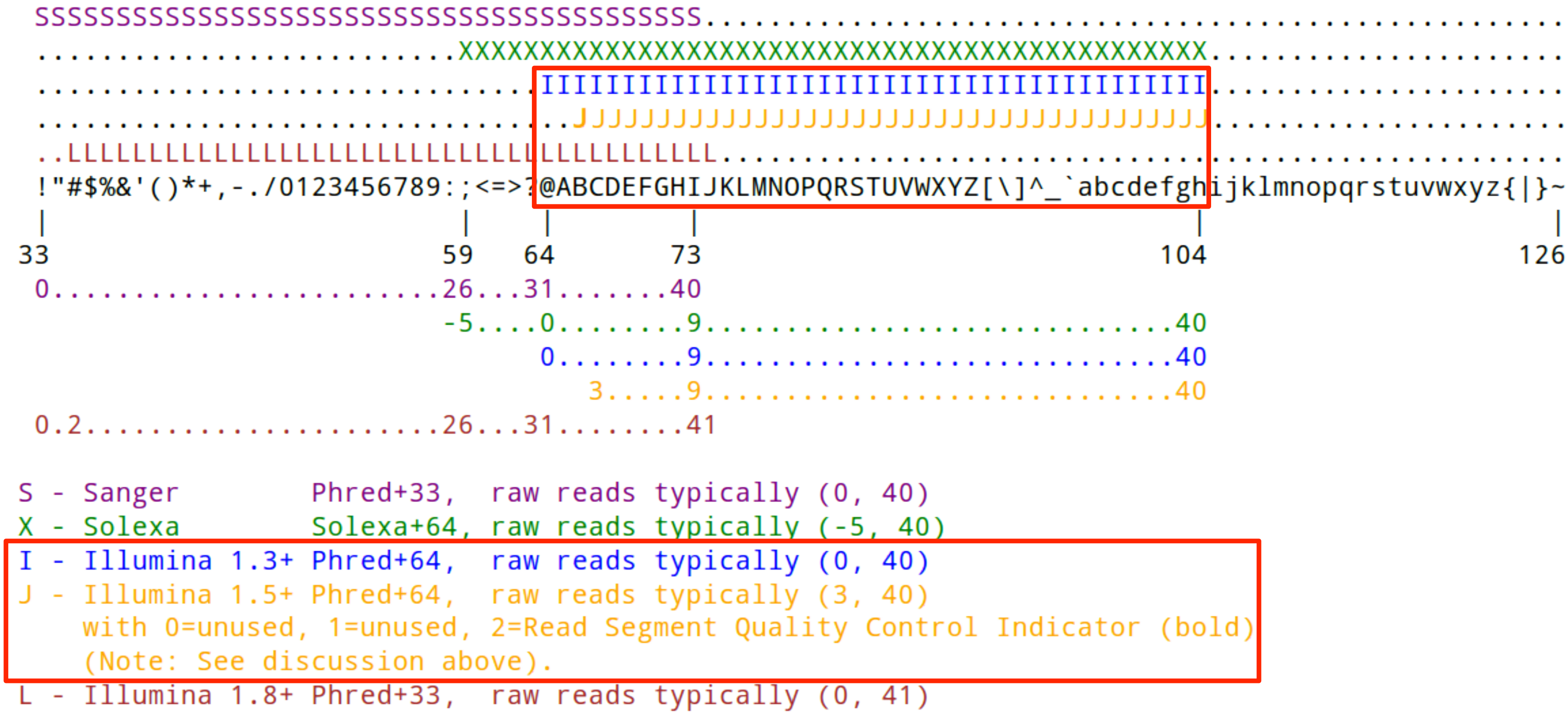


```
 SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
 ..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
 ...............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
 .................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
 ..LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL....................................................
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 |                         |    |        |                              |                     |
33                        59   64       73                            104                   126
 0........................26...31.......40
                          -5....0........9..............................40
                                0........9..............................40
                                   3.....9..............................40
 0.2.......................26...31........41

S - Sanger        Phred+33,  raw reads typically (0, 40)
X - Solexa        Solexa+64, raw reads typically (-5, 40)
I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
    with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
    (Note: See discussion above).
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

Thanks to Wikipedia… ;-)

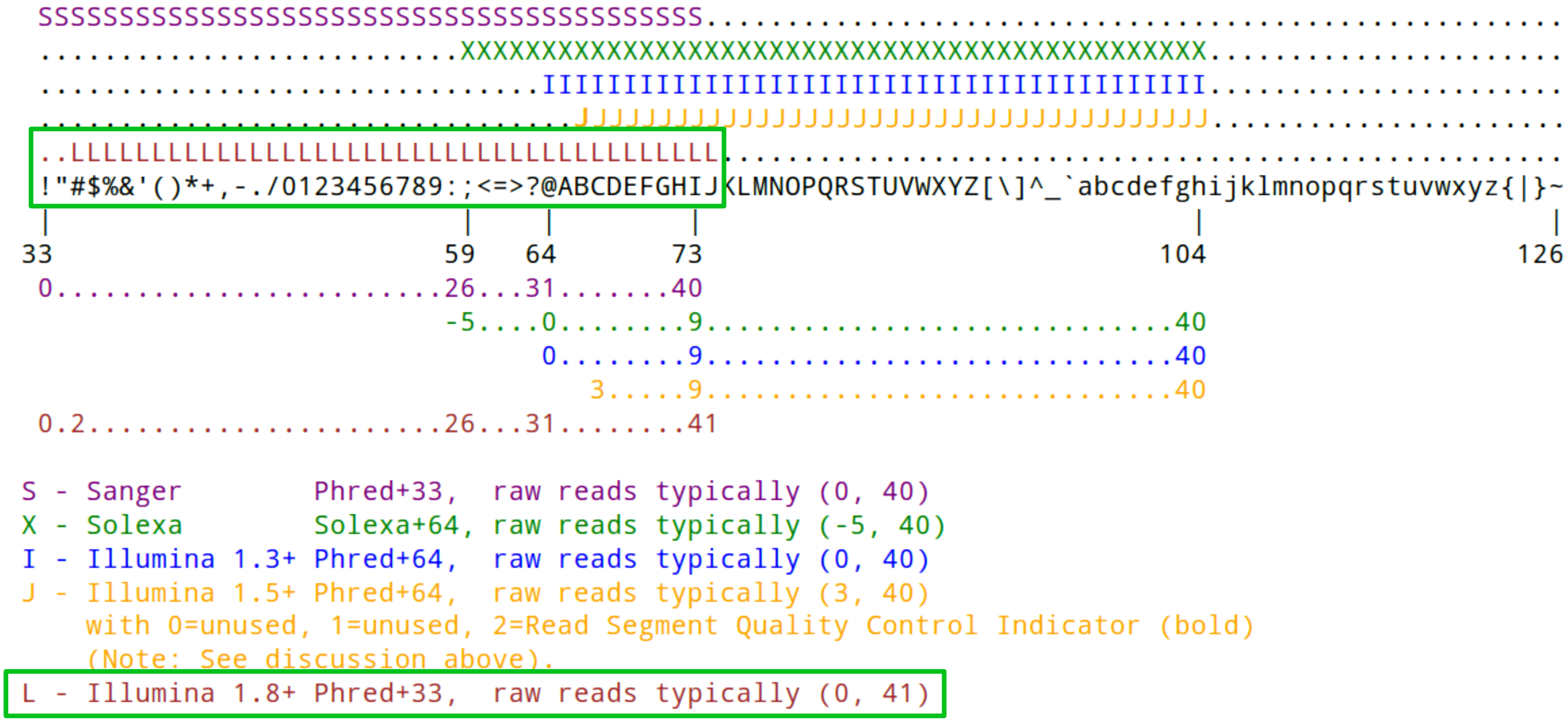
FASTQ quality encoding



```
  SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
  ..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
  ...............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
  .................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
  ..LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL....................................................
  !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
  |                         |    |        |                              |                     |
 33                        59   64       73                            104                   126
  0........................26...31.......40
                           -5....0........9.............................40
                                 0........9.............................40
                                    3.....9.............................40
  0.2......................26...31........41

 S - Sanger        Phred+33,  raw reads typically (0, 40)
 X - Solexa        Solexa+64, raw reads typically (-5, 40)
 I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
 J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
     with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
     (Note: See discussion above).
 L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

FASTQ quality encoding



```
 SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
 ..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
 ...............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
 .................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
 ..LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL....................................................
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 |                         |    |        |                              |                     |
33                        59   64       73                            104                   126
 0........................26...31.......40
                         -5....0........9.............................40
                               0........9.............................40
                                  3.....9.............................40
 0.2......................26...31........41

S - Sanger        Phred+33,  raw reads typically (0, 40)
X - Solexa        Solexa+64, raw reads typically (-5, 40)
I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
    with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
    (Note: See discussion above).
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

FASTQ quality encoding

```
@MERCURE_0127:7:1101:1162:2110#CTTGTA/1
TAATAACCCATTAAATACCAATCCAGAAAGCAGCGTGGGTTCAATTCCCAAGATCGGAAG
+MERCURE_0127:7:1101:1162:2110#CTTGTA/1
bbbeeeeegggggiiiihfgfffgihhiihfhfcab``aKZ^]b]]_]`b^^_b``[a__
@MERCURE_0127:7:1101:1182:2111#CTTGTA/1
ACTTACCTCCTGACCCCCCAAAGCCTACTCTCCACTTGCCTGGATGAGCGCAGCTCCAAC
+MERCURE_0127:7:1101:1182:2111#CTTGTA/1
bbbeeeeeggghiihhihiiiiiigaaabb`b`b]`b`b^``T]T]bc_aOEETR___BB
```

```
@HWI-ST227:191:D16GHACXX:8:2308:20216:200677 1:N:0:CGATGT
GCCATTGATGGTGGTGTGTGTTTGGTTGGTTGTTGGATGGGGGTGGGGGGTGTGGTGCG
+
++1BD2222==2A+2+2<3CFFIIA<E)1?C:)0?)*0*0?D@################
@HWI-ST227:191:D16GHACXX:8:2308:20300:200513 1:N:0:CGATGT
CGTTGTTCCTCGCGACGAGAAAAGTGCAGACGGTTTAGGGATCATCGGTATTTCGTGCG
+
?@?ADDDDDBCF@HIEIAGDHB;DDBHGIIEBG:FBDGHBD@CA+9:>098595?CCC<
```

FASTQ quality encoding

```
@MERCURE_0127:7:1101:1162:2110#CTTGTA/1
TAATAACCCATTAAATACCAATCCAGAAAGCAGCGTGGGTTCAATTCCCAAGATCGGAAG
+MERCURE_0127:7:1101:1162:2110#CTTGTA/1
bbbeeeeeggggiiiihfgfffgihhiihfhfcab``aKZ^]b]]_]`b^^_b``[a__
@MERCURE_0127:7:1101:1182:2111#CTTGTA/1
ACTTACCTCCTGACCCCCCAAAGCCTACTCTCCACTTGCCTGGATGAGCGCAGCTCCAAC
+MERCURE_0127:7:1101:1182:2111#CTTGTA/1
bbbeeeeeggggghiihhihiiiiiigaaabb`b`b]`b`b^``T]T]bc_aOEETR___BB
```

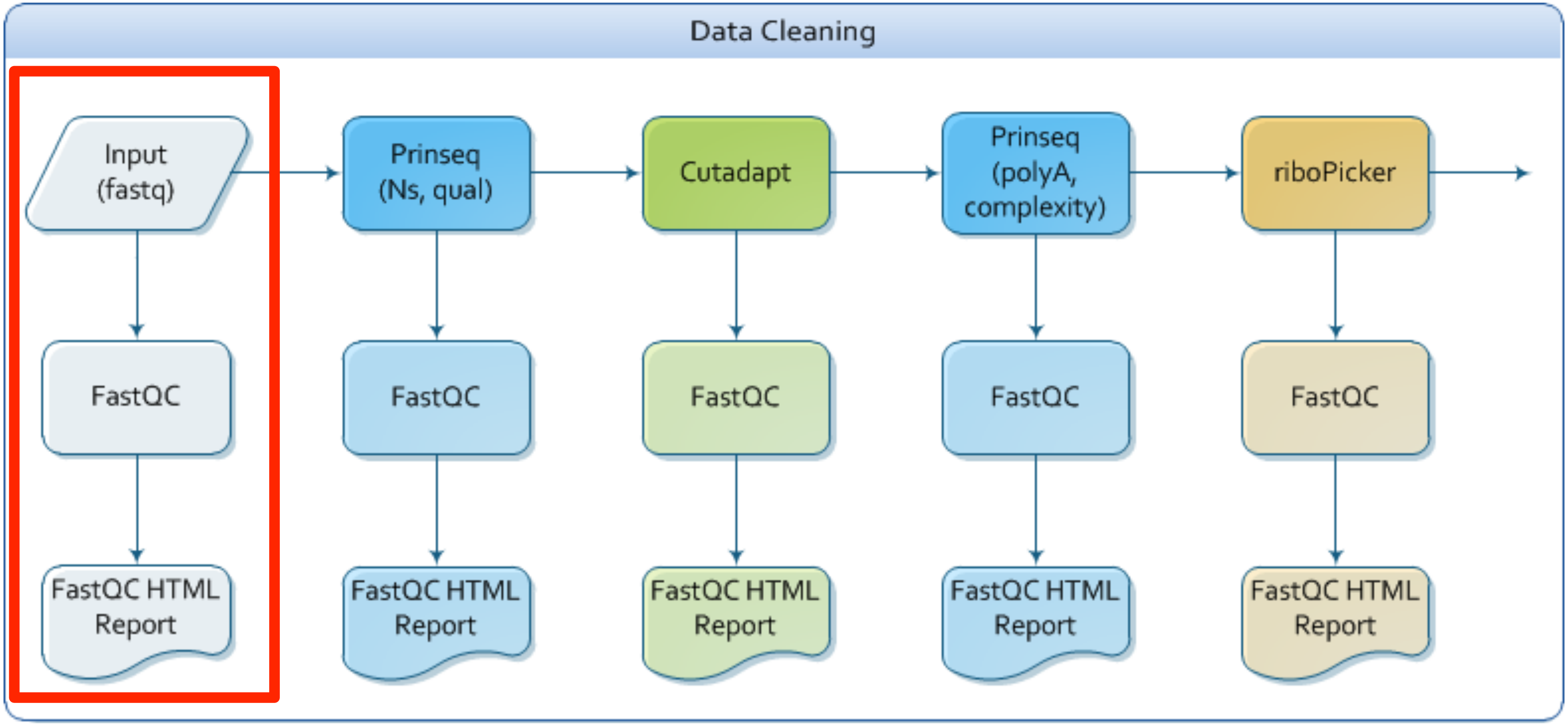
Phred+64

```
@HWI-ST227:191:D16GHACXX:8:2308:20216:200677 1:N:0:CGATGT
GCCATTGATGGTGGTGTGTGTTTGGTTGGTTGTTGGATGGGGGTGGGGGGTGTGGTGCG
+
++1BD2222==2A+2+2<3CFFIIA<E)1?C:)0?)*0*0?D@################
@HWI-ST227:191:D16GHACXX:8:2308:20300:200513 1:N:0:CGATGT
CGTTGTTCCTCGCGACGAGAAAAGTGCAGACGGTTTAGGGATCATCGGTATTTCGTGCG
+
?@?ADDDDDBCF@HIEIAGDHB;DDBHGIIEBG:FBDGHBD@CA+9:>098595?CCC<
```

Phred+33

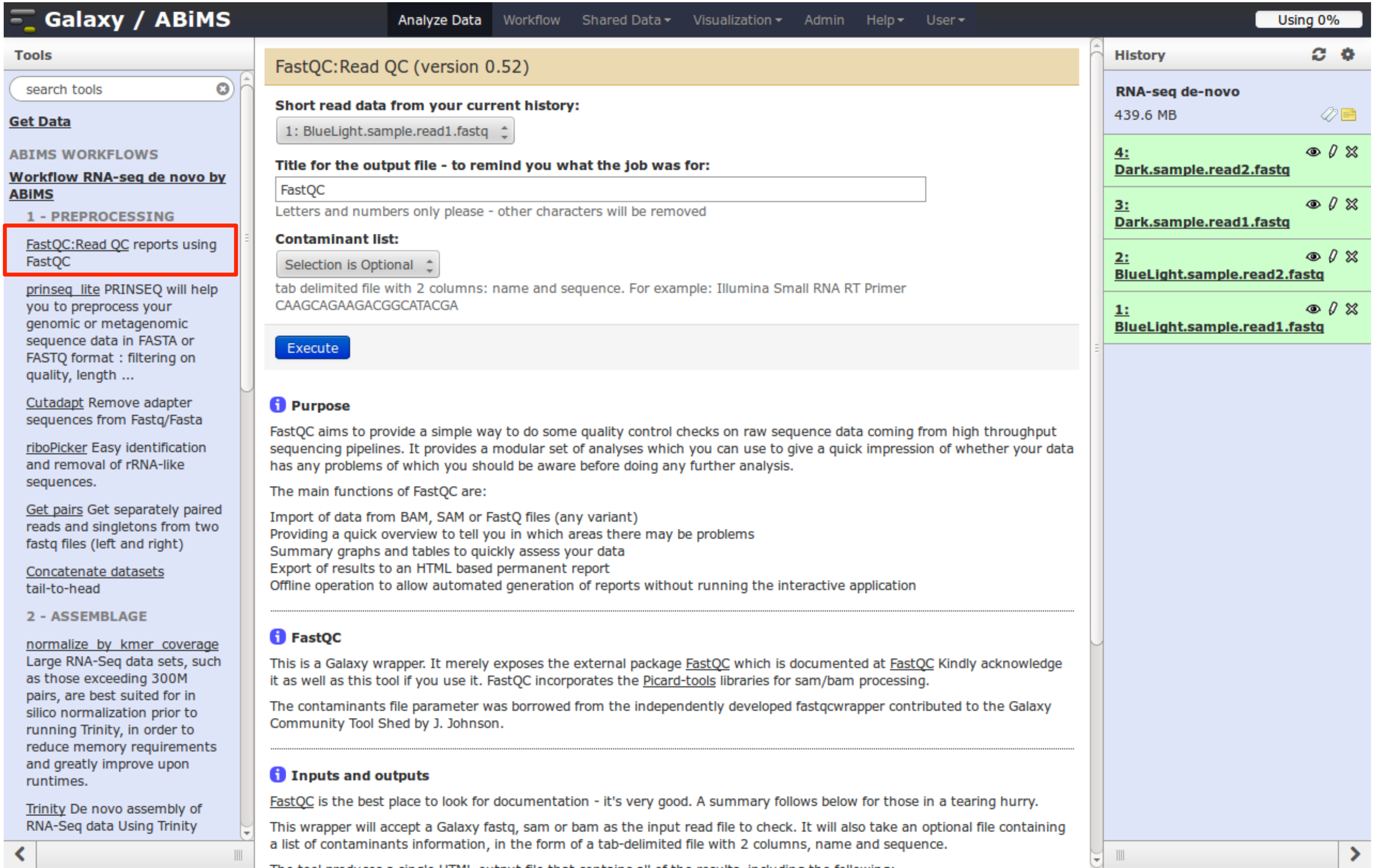
Station Biologique Roscoff

TP



FastQC

TP



Get Data

ABIMS WORKFLOWS

Workflow RNA-seq de novo by ABIMS

1 - PREPROCESSING

FastQC:Read QC reports using FastQC

prinseq_lite PRINSEQ will help you to preprocess your genomic or metagenomic sequence data in FASTA or FASTQ format : filtering on quality, length ...

Cutadapt Remove adapter sequences from Fastq/Fasta

riboPicker Easy identification and removal of rRNA-like sequences.

Get pairs Get separately paired reads and singletons from two fastq files (left and right)

Concatenate datasets tail-to-head

2 - ASSEMBLAGE

normalize_by_kmer_coverage Large RNA-Seq data sets, such as those exceeding 300M pairs, are best suited for in silico normalization prior to running Trinity, in order to reduce memory requirements and greatly improve upon runtimes.

Trinity De novo assembly of RNA-Seq data Using Trinity

FastQC:Read QC (version 0.52)

Short read data from your current history:
1: BlueLight.sample.read1.fastq

Title for the output file - to remind you what the job was for:
FastQC
Letters and numbers only please - other characters will be removed

Contaminant list:
Selection is Optional
tab delimited file with 2 columns: name and sequence. For example: Illumina Small RNA RT Primer CAAGCAGAAGACGGCATACGA

Execute

Purpose

FastQC aims to provide a simple way to do some quality control checks on raw sequence data coming from high throughput sequencing pipelines. It provides a modular set of analyses which you can use to give a quick impression of whether your data has any problems of which you should be aware before doing any further analysis.

The main functions of FastQC are:

Import of data from BAM, SAM or FastQ files (any variant)
Providing a quick overview to tell you in which areas there may be problems
Summary graphs and tables to quickly assess your data
Export of results to an HTML based permanent report
Offline operation to allow automated generation of reports without running the interactive application

FastQC

This is a Galaxy wrapper. It merely exposes the external package FastQC which is documented at FastQC Kindly acknowledge it as well as this tool if you use it. FastQC incorporates the Picard-tools libraries for sam/bam processing.

The contaminants file parameter was borrowed from the independently developed fastqcwrapper contributed to the Galaxy Community Tool Shed by J. Johnson.

Inputs and outputs

FastQC is the best place to look for documentation - it's very good. A summary follows below for those in a tearing hurry.

This wrapper will accept a Galaxy fastq, sam or bam as the input read file to check. It will also take an optional file containing a list of contaminants information, in the form of a tab-delimited file with 2 columns, name and sequence.

The tool produces a single HTML output file that contains all of the results, including the following:

RNA-seq de-novo
439.6 MB

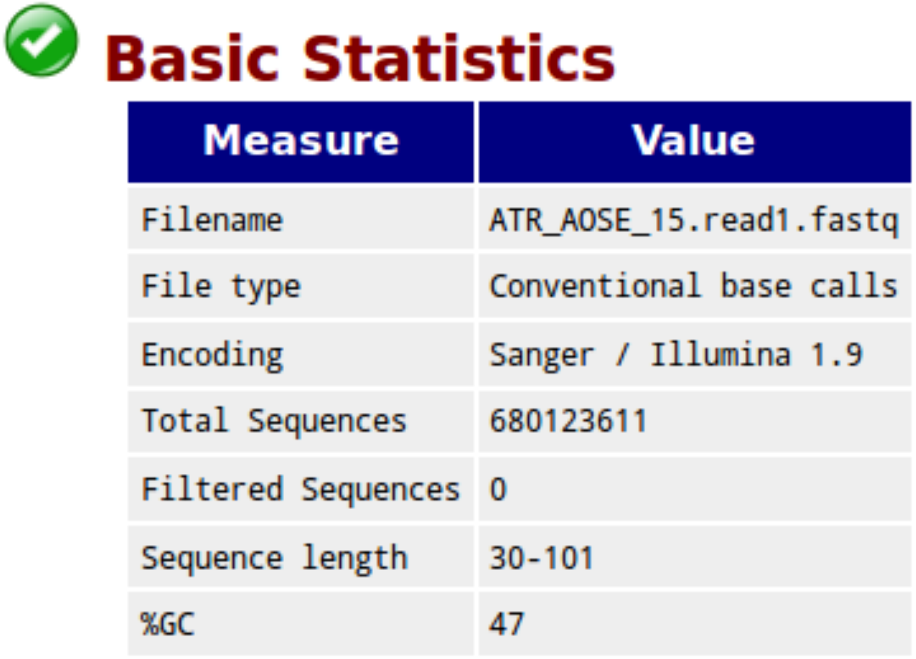
4: Dark.sample.read2.fastq

3: Dark.sample.read1.fastq

2: BlueLight.sample.read2.fastq

1: BlueLight.sample.read1.fastq

FastQC : Basic Statistics

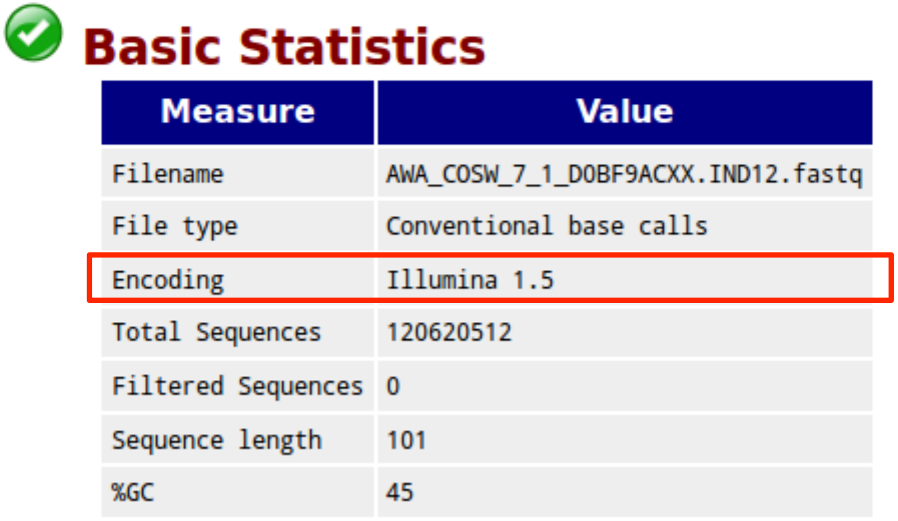


✓ Basic Statistics

Measure	Value
Filename	ATR_AOSE_15.read1.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	680123611
Filtered Sequences	0
Sequence length	30-101
%GC	47

FastQC : Basic Statistics

```
@MERCURE_0127:7:1101:1162:2110#CTTGTA/1
TAATAACCCATTAAATACCAATCCAGAAAGCAGCGTGGGTTCAATTCCCAAGATCGGAAG
+MERCURE_0127:7:1101:1162:2110#CTTGTA/1
bbbeeeeeggggggiiiiihfgfffgihhiihfhfcab``aKZ^]b]]_]`b^^_b``[a__
@MERCURE_0127:7:1101:1182:2111#CTTGTA/1
ACTTACCTCCTGACCCCCCAAAGCCTACTCTCCACTTGCCTGGATGAGCGCAGCTCCAAC
+MERCURE_0127:7:1101:1182:2111#CTTGTA/1
bbbeeeeeggggghiihhihiiiiiigaaabb`b`b]`b`b^`T]T]bc_aOEETR___BB
```

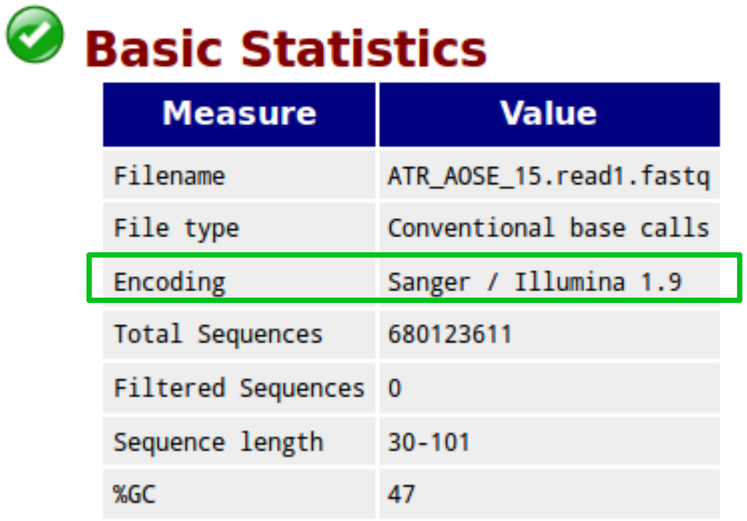


Basic Statistics

Measure	Value
Filename	AWA_COSW_7_1_D0BF9ACXX.IND12.fastq
File type	Conventional base calls
Encoding	Illumina 1.5
Total Sequences	120620512
Filtered Sequences	0
Sequence length	101
%GC	45

Phred+64

```
@HWI-ST227:191:D16GHACXX:8:2308:20216:200677 1:N:0:CGATGT
GCCATTGATGGTGGTGTGTGTTTGGTTGGTTGTTGGATGGGGGTGGGGGGTGTGGTGCG
+
++1BD2222==2A+2+2<3CFFIIA<E)1?C:)0?)*0*0?D@################
@HWI-ST227:191:D16GHACXX:8:2308:20300:200513 1:N:0:CGATGT
CGTTGTTCCTCGCGACGAGAAAAGTGCAGACGGTTTAGGGATCATCGGTATTTCGTGCG
+
?@?ADDDDDBCF@HIEIAGDHB;DDBHGIIEBG:FBDGHBD@CA+9:>098595?CCC<
```



Basic Statistics

Measure	Value
Filename	ATR_AOSE_15.read1.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	680123611
Filtered Sequences	0
Sequence length	30-101
%GC	47

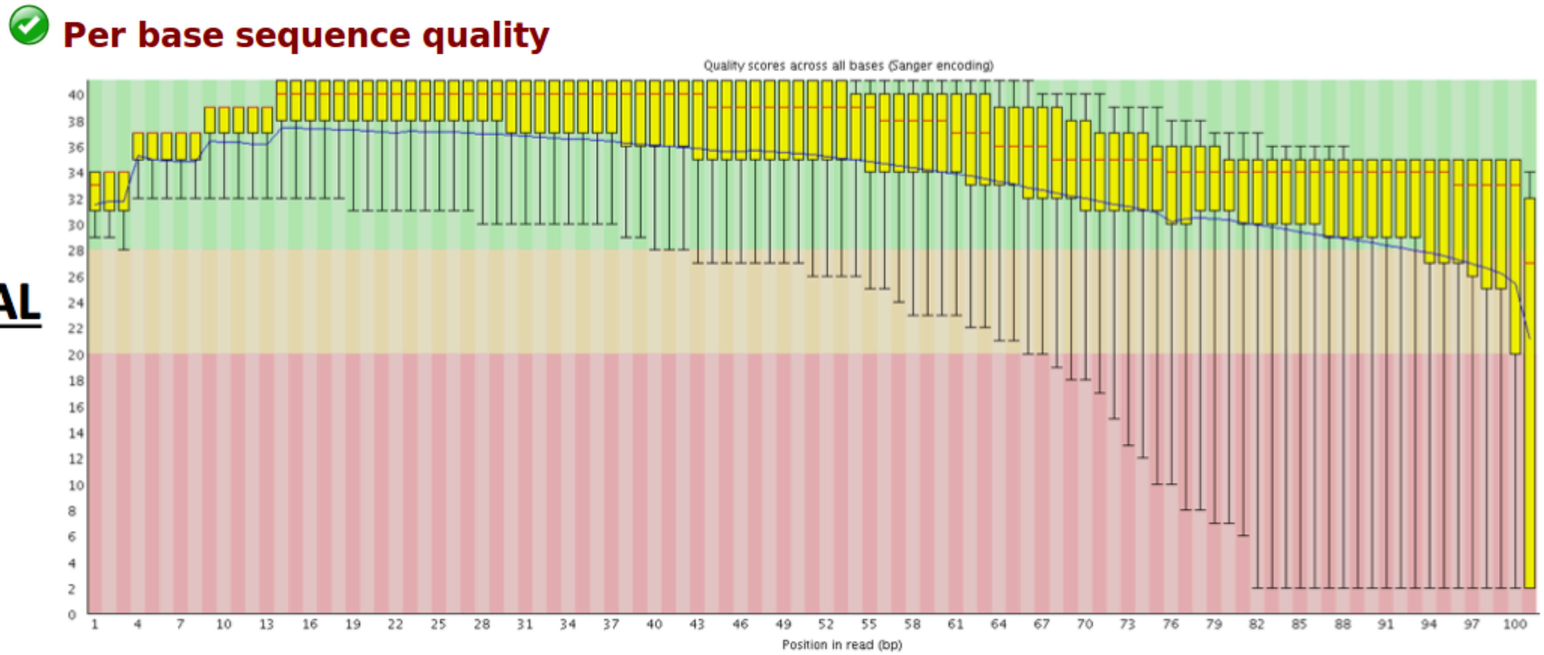
Phred+33

FastQC : Per base sequence quality

This plot shows the base quality score distribution for all reads in a lane, with each read position considered independently.

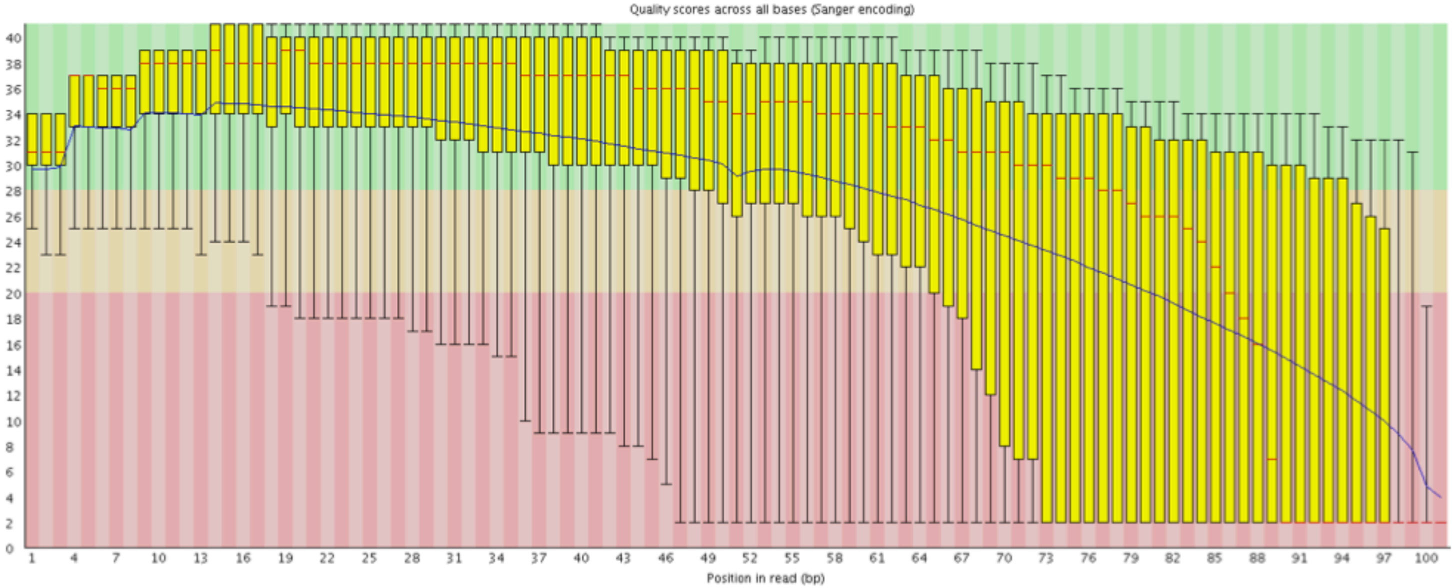
- x-axis = position in read (bp)
- y-axis = Phred-like base quality score [pink=0-20, tan=20-30, green=30-40]
- red bar = median score, blue line = mean score
- yellow box = 25th to 75th percentile, black whiskers = 10th to 90th percentile

GOOD/NORMAL LANE

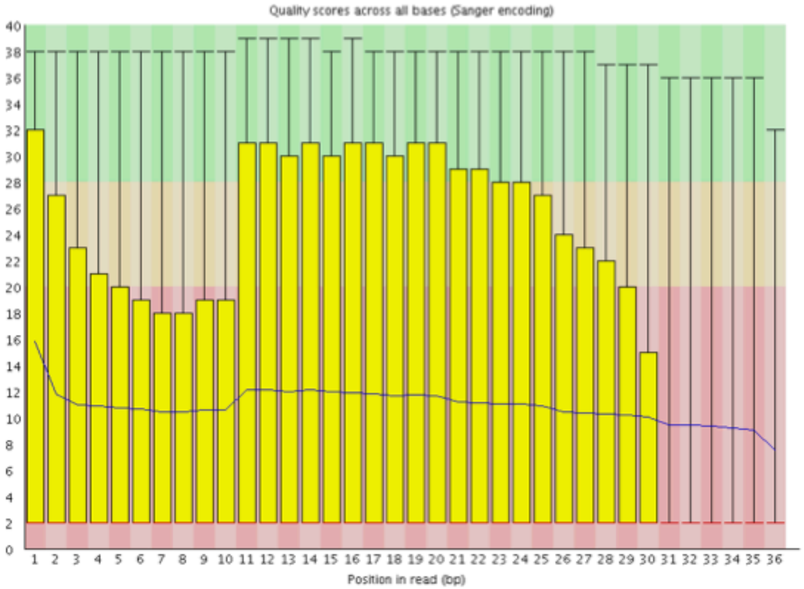

✓ Per base sequence quality

FastQC : Per base sequence quality

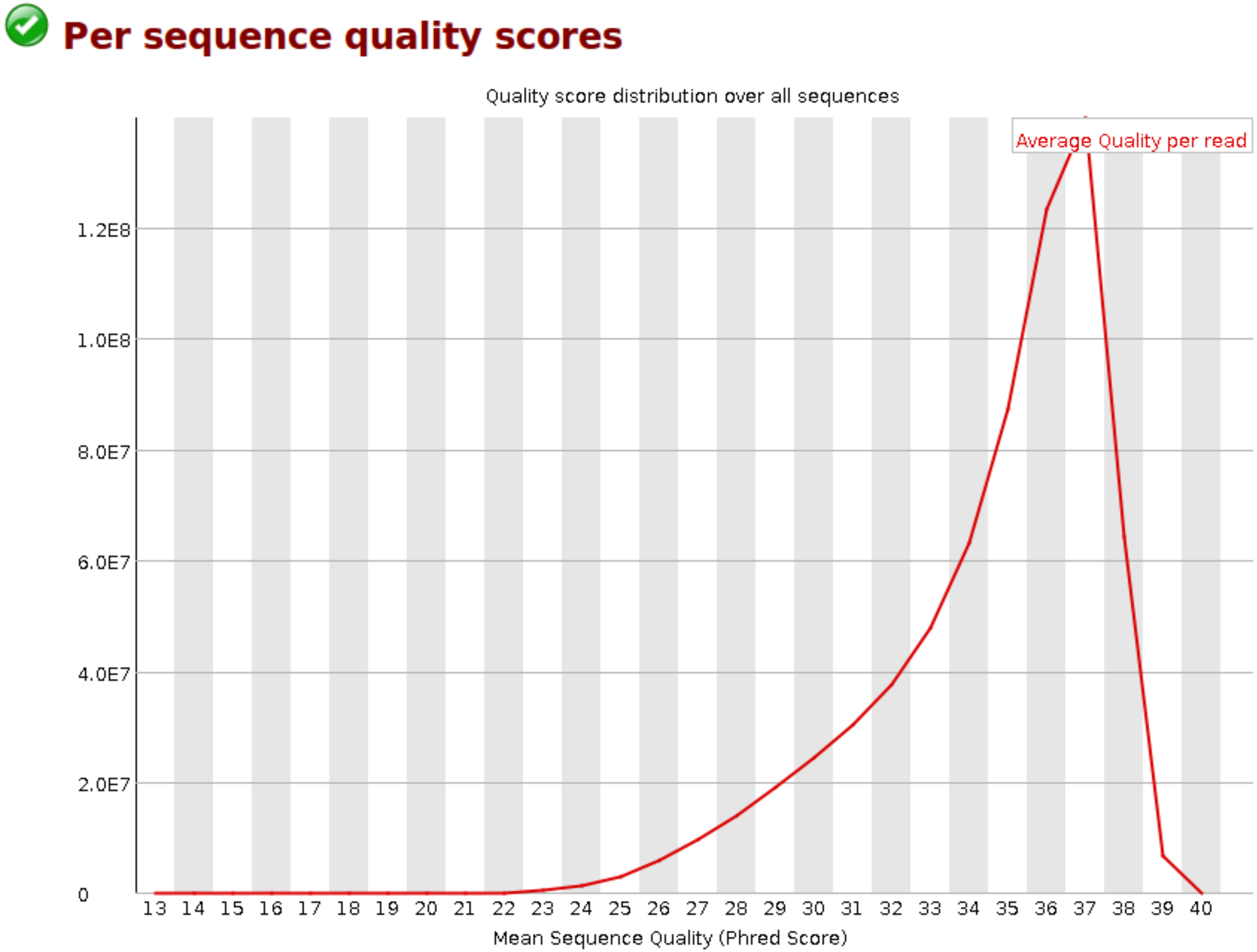
SALVAGEABLE LANE



FAILED LANE



FastQC: Per sequence quality scores



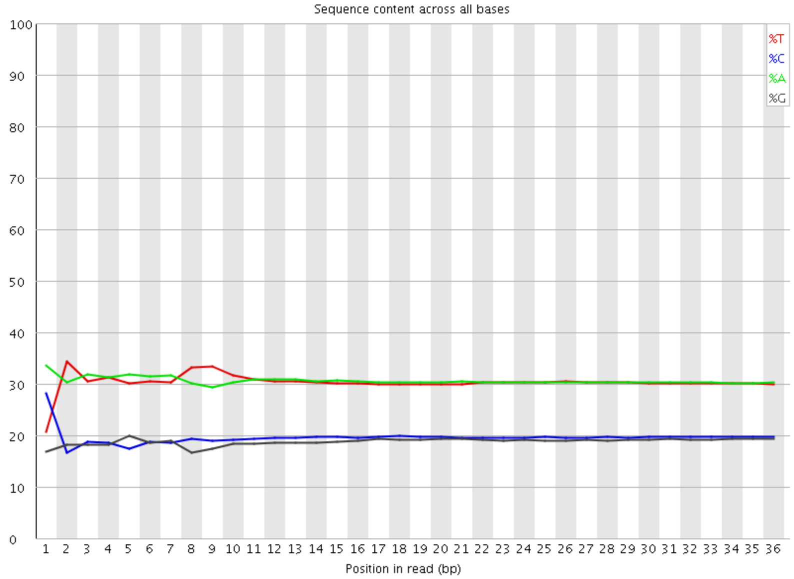
✓ Per sequence quality scores

FastQC: Per base sequence content

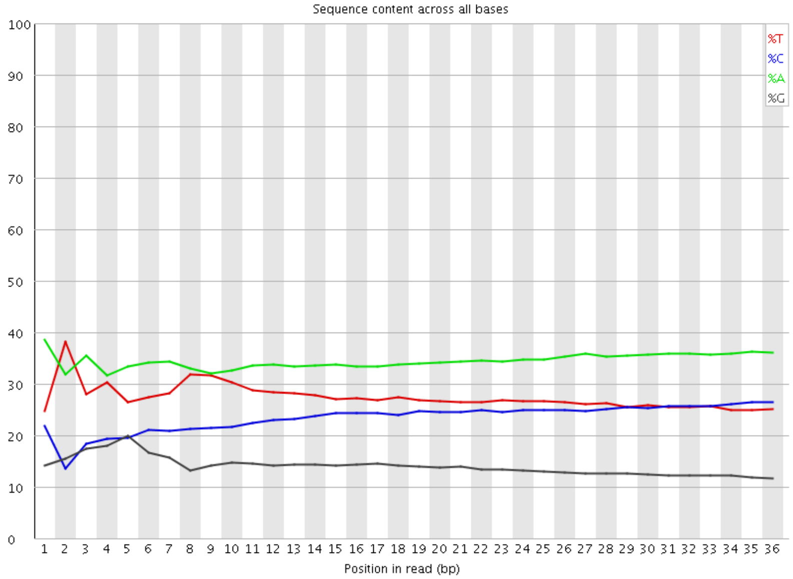
This plot shows the nucleotide distribution per read position for all reads in a lane.

- x-axis = position in read (bp)
- y-axis = % of all reads in the lane
- colors refer to individual nucleotides: **A**, **C**, **G**, **T**

GOOD LANE



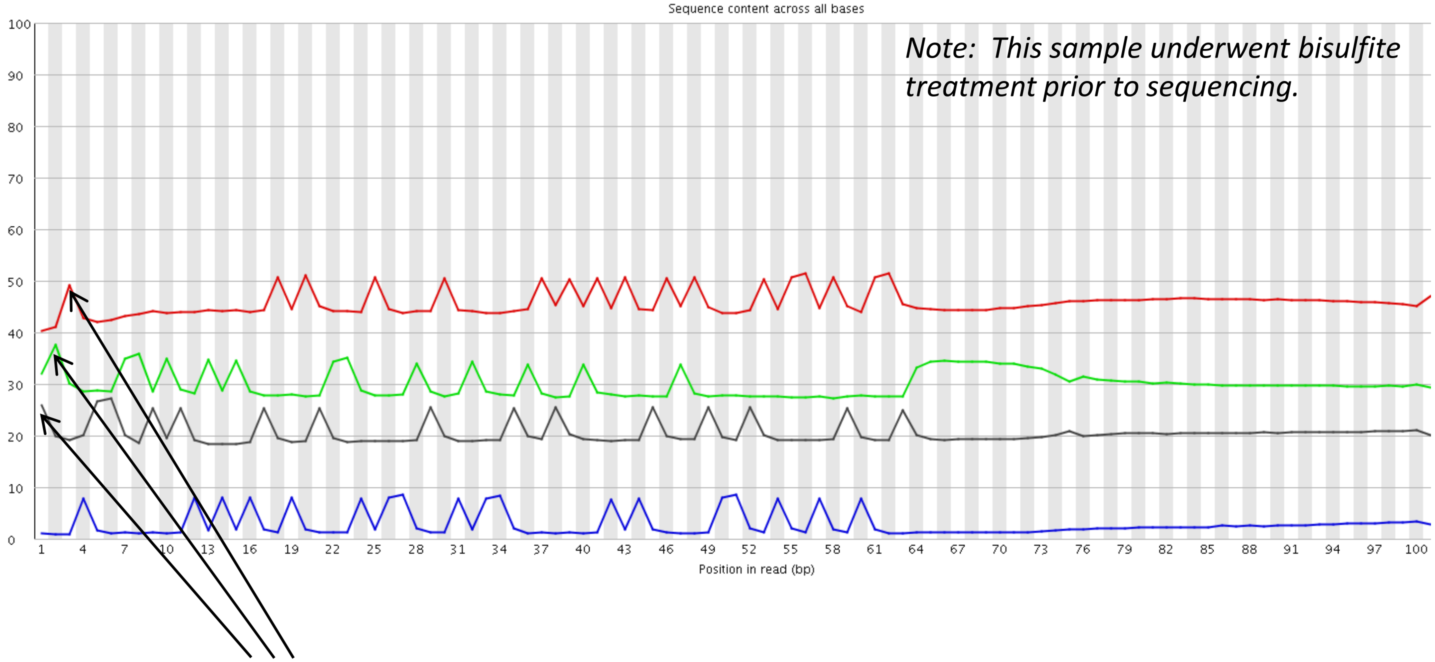
BAD LANE



Can this be fixed? **No.**

FastQC: Per base sequence content

This lane has a different problem – one sequence motif is highly over-represented.



Note: This sample underwent bisulfite treatment prior to sequencing.

primer/adapter sequence: **GATCGGAAGAGCACACGTCTGAACTCCAGTCACACAGTGATCTCGTATGCCGTCTTCTGCTTG**

In this lane, ~10% of reads have the adapter sequence & the rest are normal.

<u>Can this be fixed?</u> Yes. Simply remove the reads w/ adapter contamination, and everything that's left should be fine. (Talk to a bioinformatics analyst for help.)

FastQC: Per sequence GC content

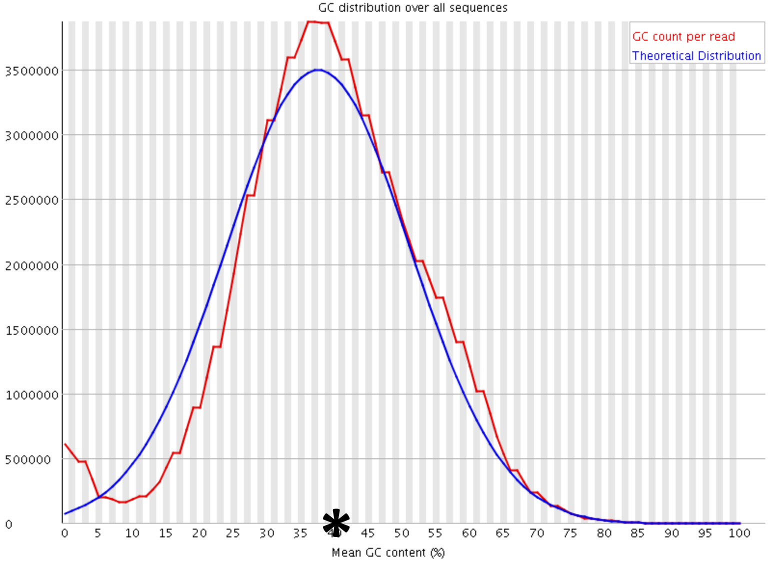
This plot shows the distribution of GC content per read for all reads in a lane.

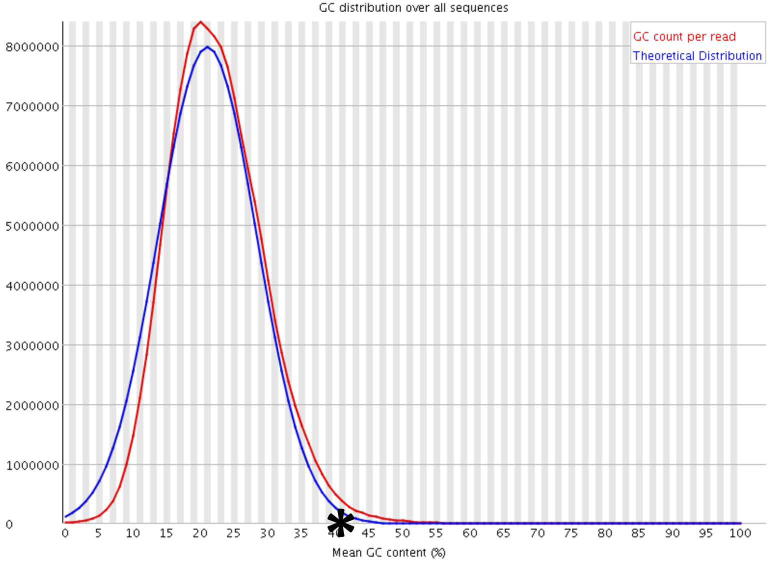
- x-axis = mean GC content (%)
- y-axis = # of reads
- red: observed read count, blue: theoretical distribution (given observed)

GOOD LANE

mouse genome ≈ 40% GC

BAD LANE

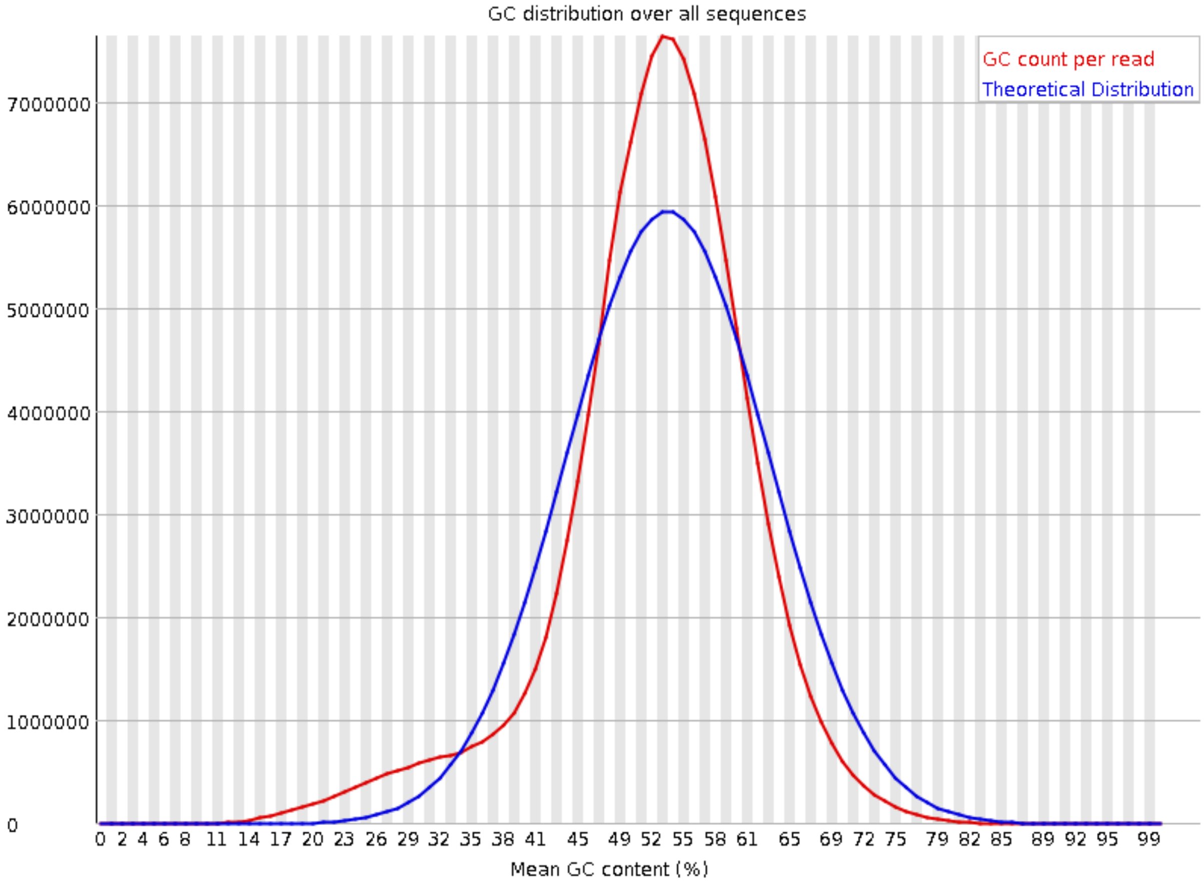




Can this be fixed? No.

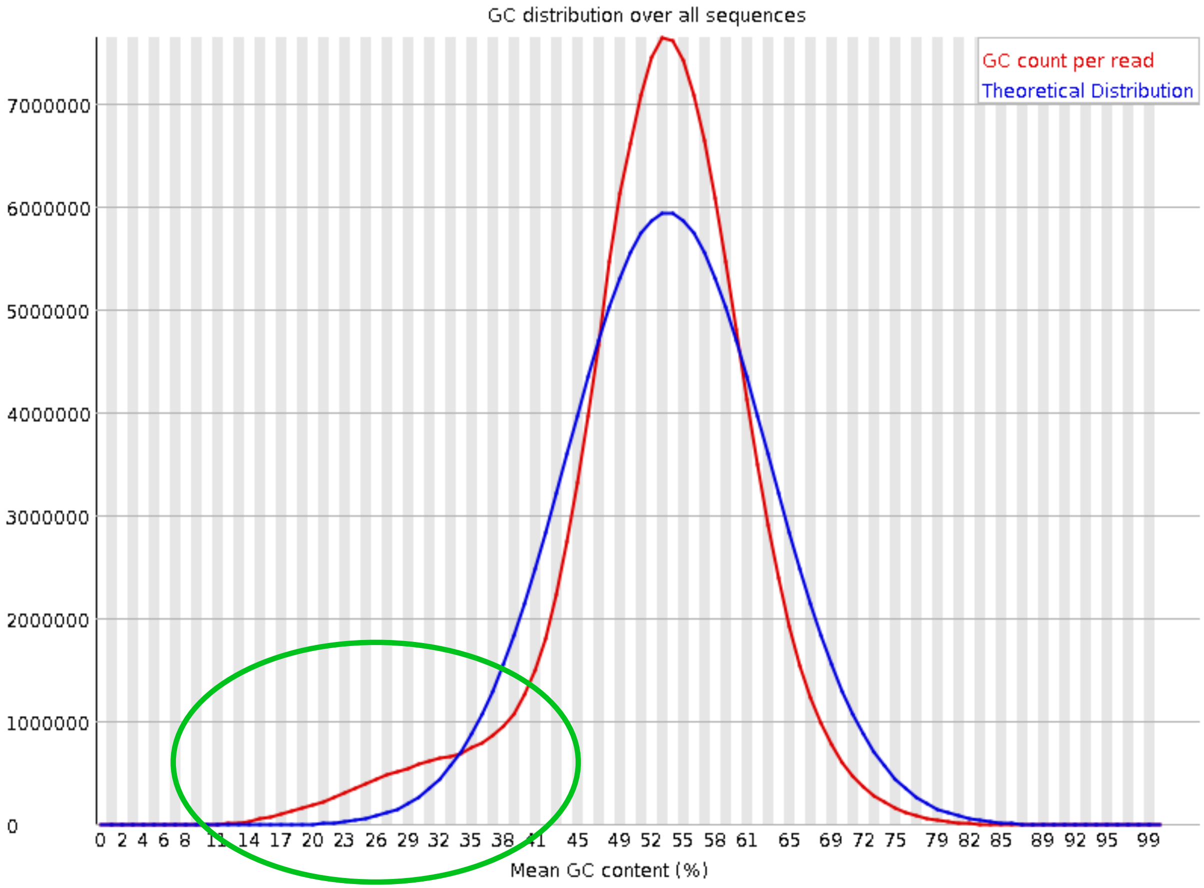
FastQC: Per sequence GC content

- A contamination ?



FastQC: Per sequence GC content

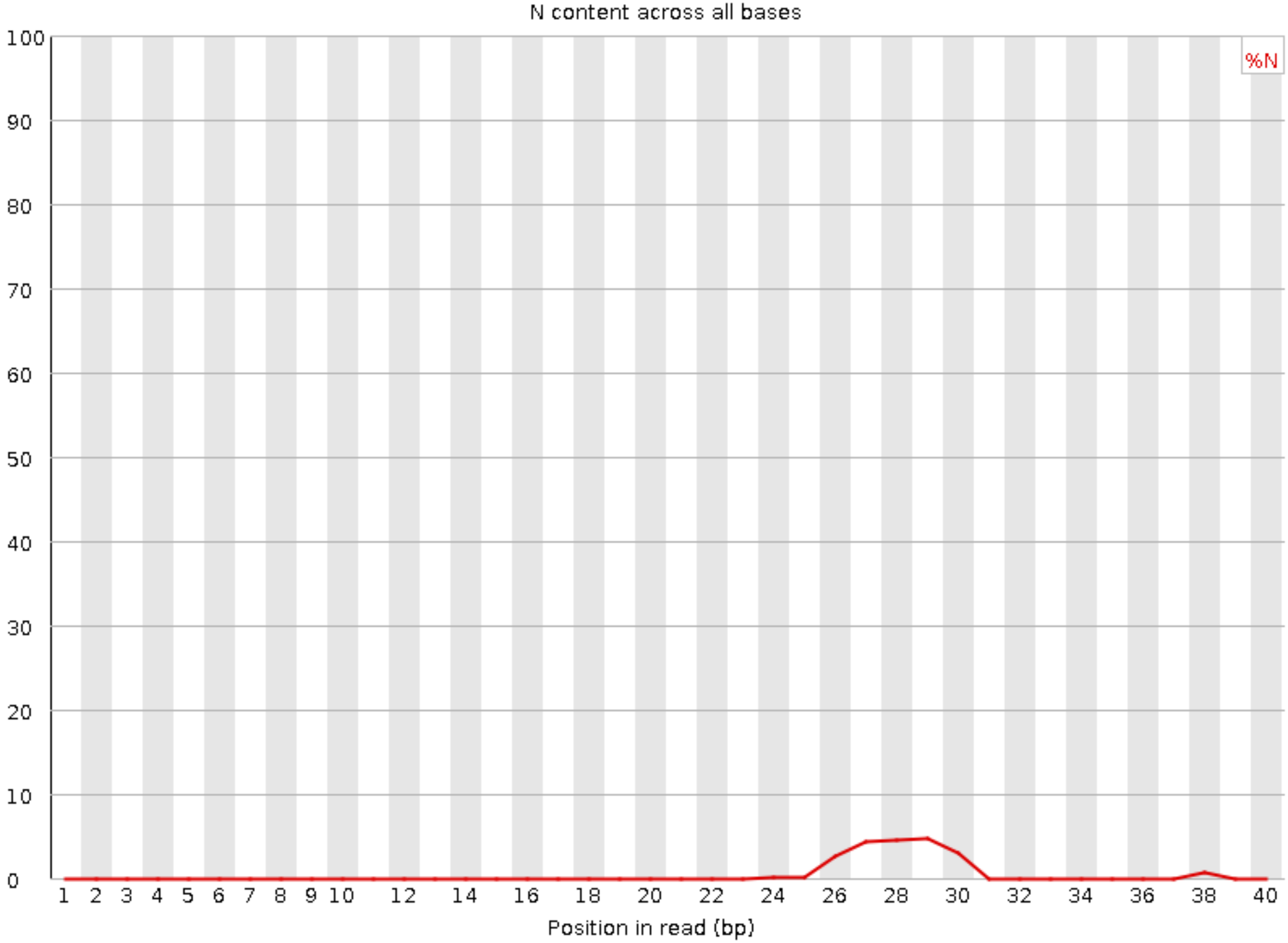
- A contamination ?



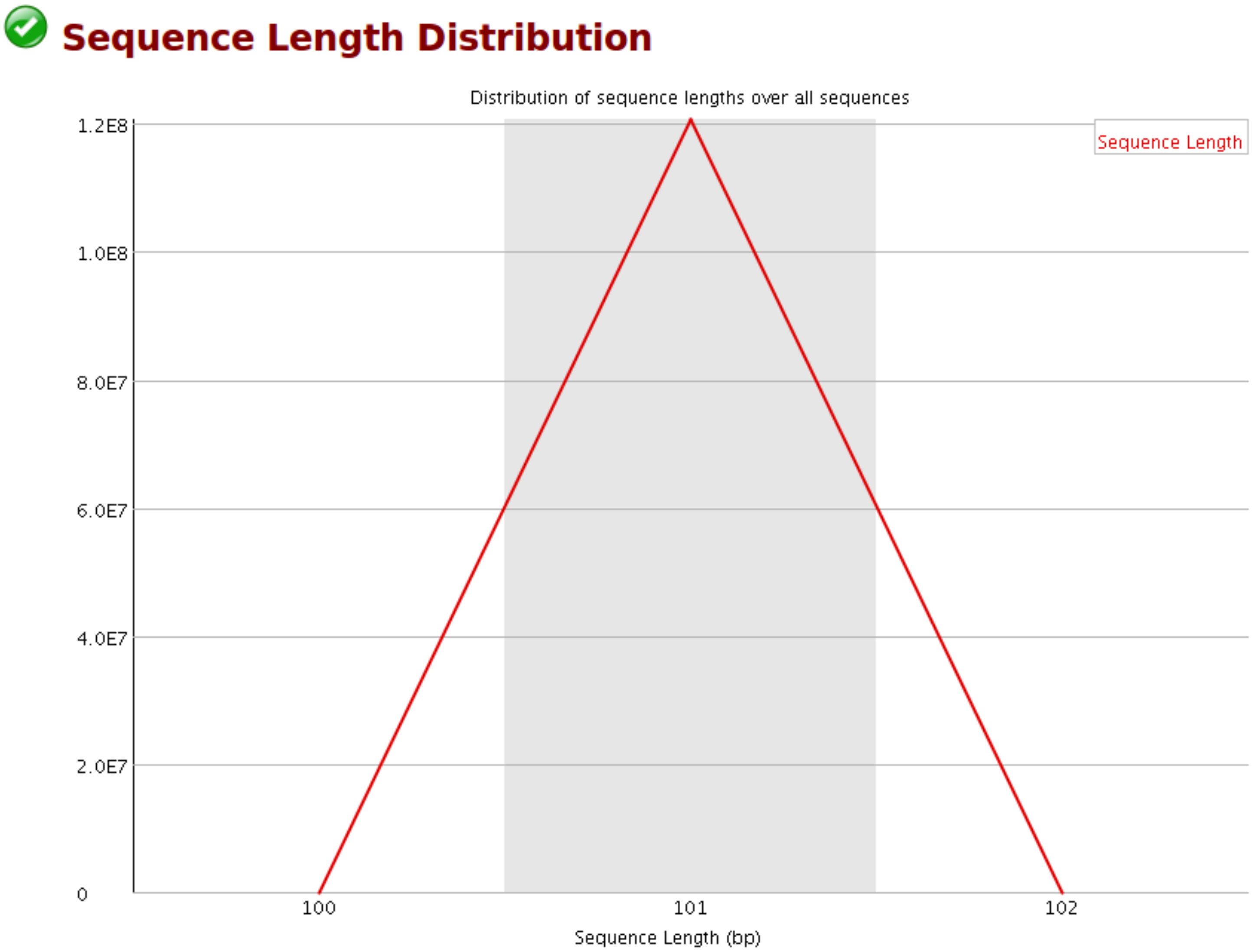
Can this be fixed ? Maybe…

FastQC: Per base N content

✓ Per base N content



FastQC: Sequence Length Distribution



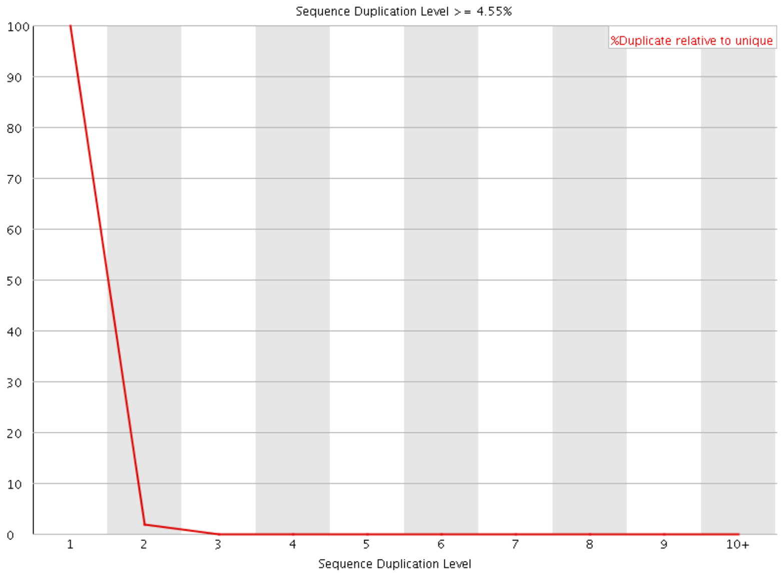
✓ Sequence Length Distribution

FastQC: Sequence Duplication Levels

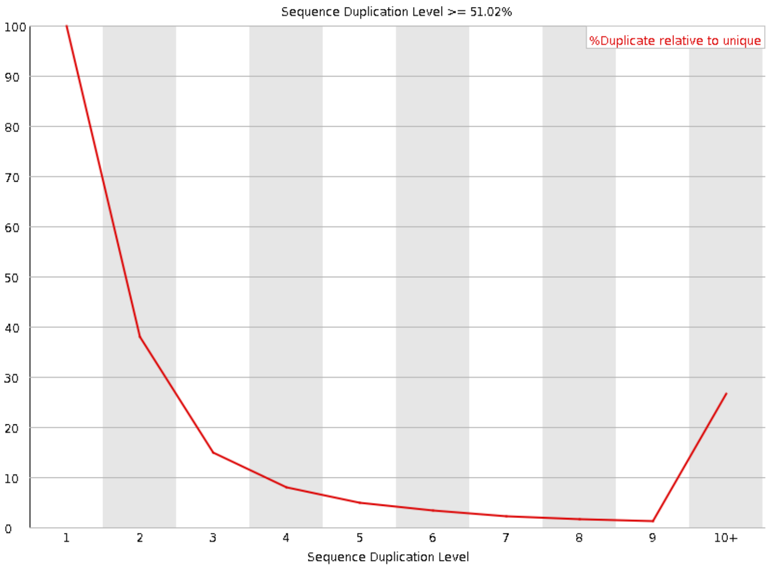
This plot shows the degree of duplication for a subset of reads in a lane.

- x-axis = sequence duplication level
- y-axis = % duplicates relative to unique reads

GOOD LANE

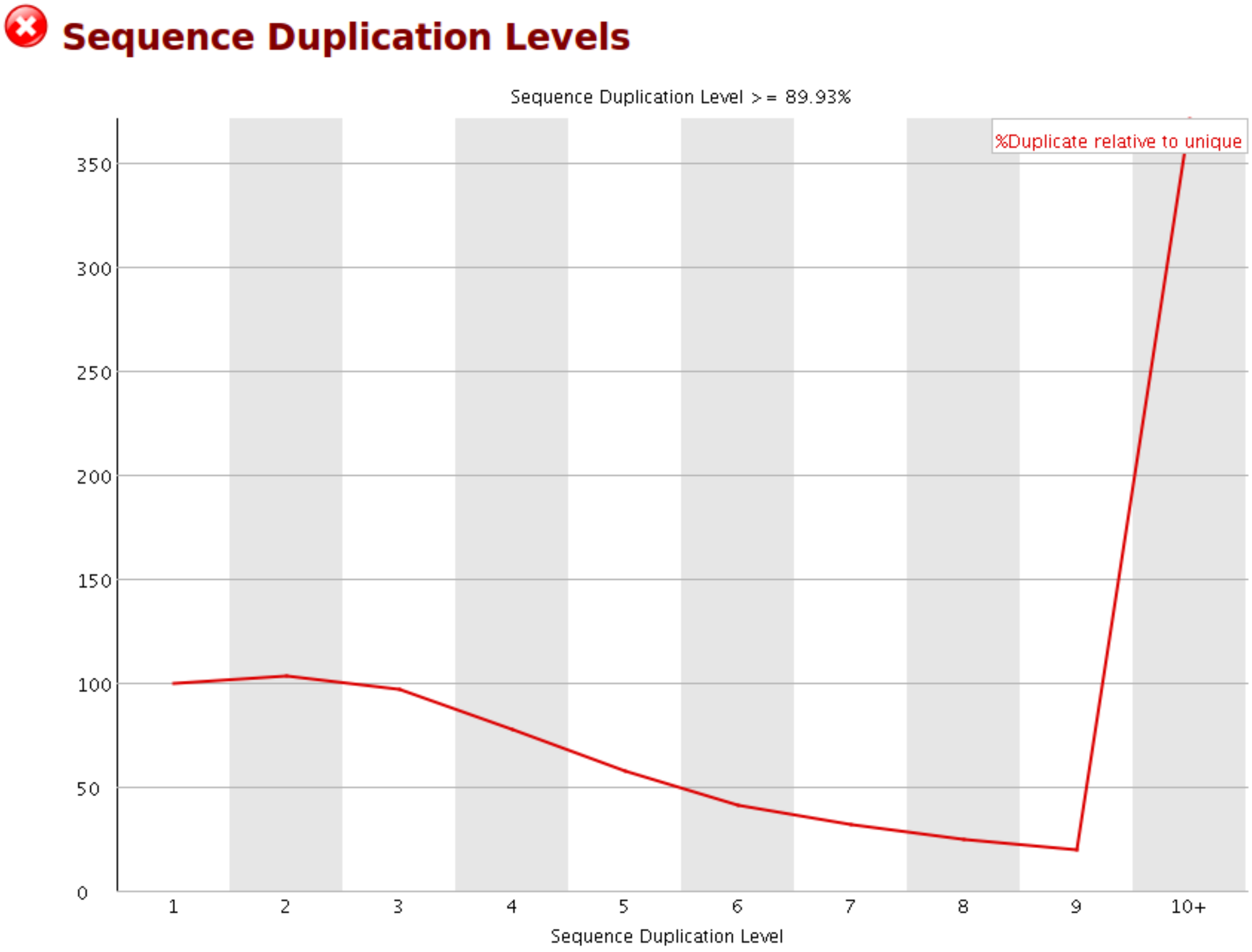


BAD LANE



Can this be fixed? **Maybe.**

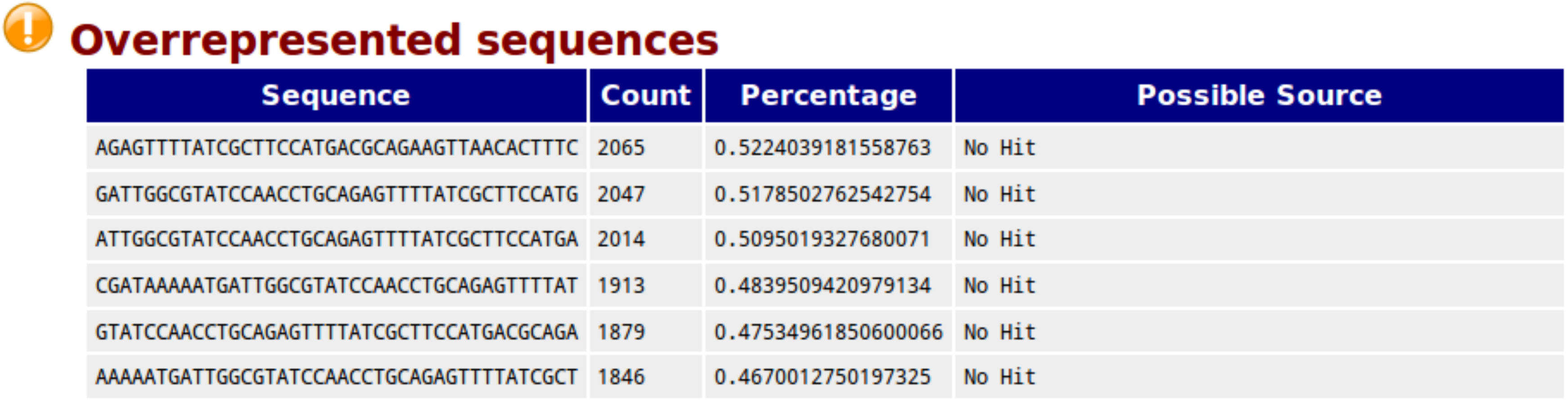
FastQC: Sequence Duplication Levels



Sequence Duplication Levels

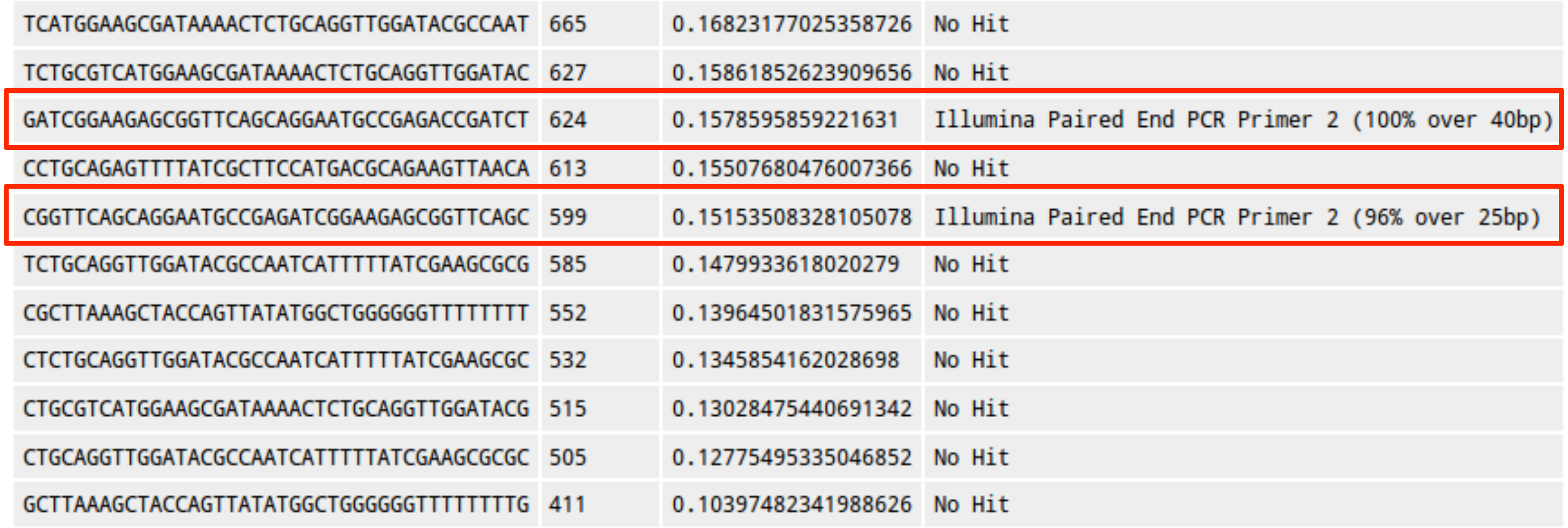
Can this be fixed? **Hem…**

FastQC: Overrepresented sequences



Overrepresented sequences

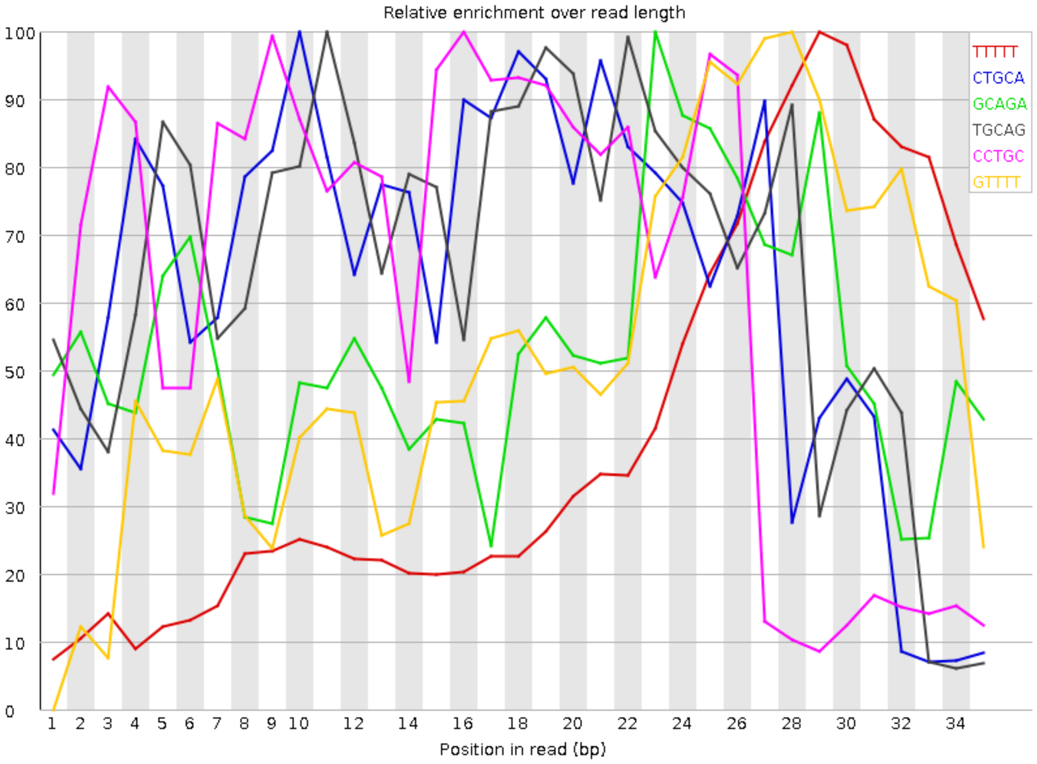
Sequence	Count	Percentage	Possible Source
AGAGTTTTATCGCTTCCATGACGCAGAAGTTAACACTTTC	2065	0.5224039181558763	No Hit
GATTGGCGTATCCAACCTGCAGAGTTTTATCGCTTCCATG	2047	0.5178502762542754	No Hit
ATTGGCGTATCCAACCTGCAGAGTTTTATCGCTTCCATGA	2014	0.5095019327680071	No Hit
CGATAAAAATGATTGGCGTATCCAACCTGCAGAGTTTTAT	1913	0.4839509420979134	No Hit
GTATCCAACCTGCAGAGTTTTATCGCTTCCATGACGCAGA	1879	0.47534961850600066	No Hit
AAAAATGATTGGCGTATCCAACCTGCAGAGTTTTATCGCT	1846	0.4670012750197325	No Hit

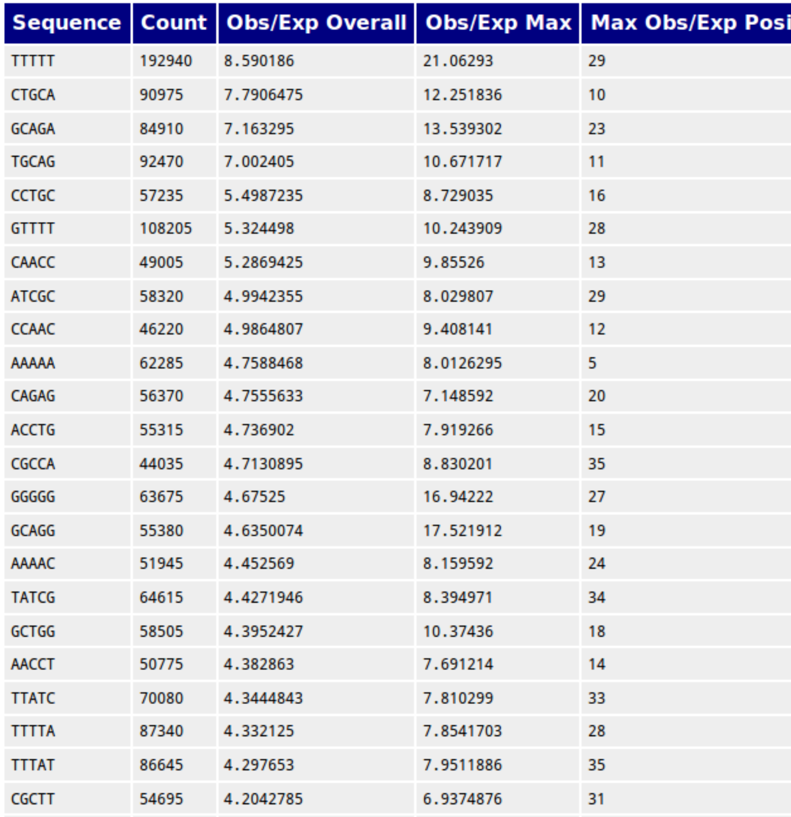


TCATGGAAGCGATAAAACTCTGCAGGTTGGATACGCCAAT	665	0.16823177025358726	No Hit
TCTGCGTCATGGAAGCGATAAAACTCTGCAGGTTGGATAC	627	0.15861852623909656	No Hit
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGATCT	624	0.157859585922163 1	Illumina Paired End PCR Primer 2 (100% over 40bp)
CCTGCAGAGTTTTATCGCTTCCATGACGCAGAAGTTAACA	613	0.15507680476007366	No Hit
CGGTTCAGCAGGAATGCCGAGATCGGAAGAGCGGTTCAGC	599	0.15153508328105078	Illumina Paired End PCR Primer 2 (96% over 25bp)
TCTGCAGGTTGGATACGCCAATCATTTTTATCGAAGCGCG	585	0.1479933618020279	No Hit
CGCTTAAAGCTACCAGTTATATGGCTGGGGGTTTTTTTT	552	0.13964501831575965	No Hit
CTCTGCAGGTTGGATACGCCAATCATTTTTATCGAAGCGC	532	0.1345854162028698	No Hit
CTGCGTCATGGAAGCGATAAAACTCTGCAGGTTGGATACG	515	0.13028475440691342	No Hit
CTGCAGGTTGGATACGCCAATCATTTTTATCGAAGCGCGC	505	0.12775495335046852	No Hit
GCTTAAAGCTACCAGTTATATGGCTGGGGGTTTTTTTTG	411	0.1039748234198862 6	No Hit

FastQC: Kmer Content

☒ Kmer Content

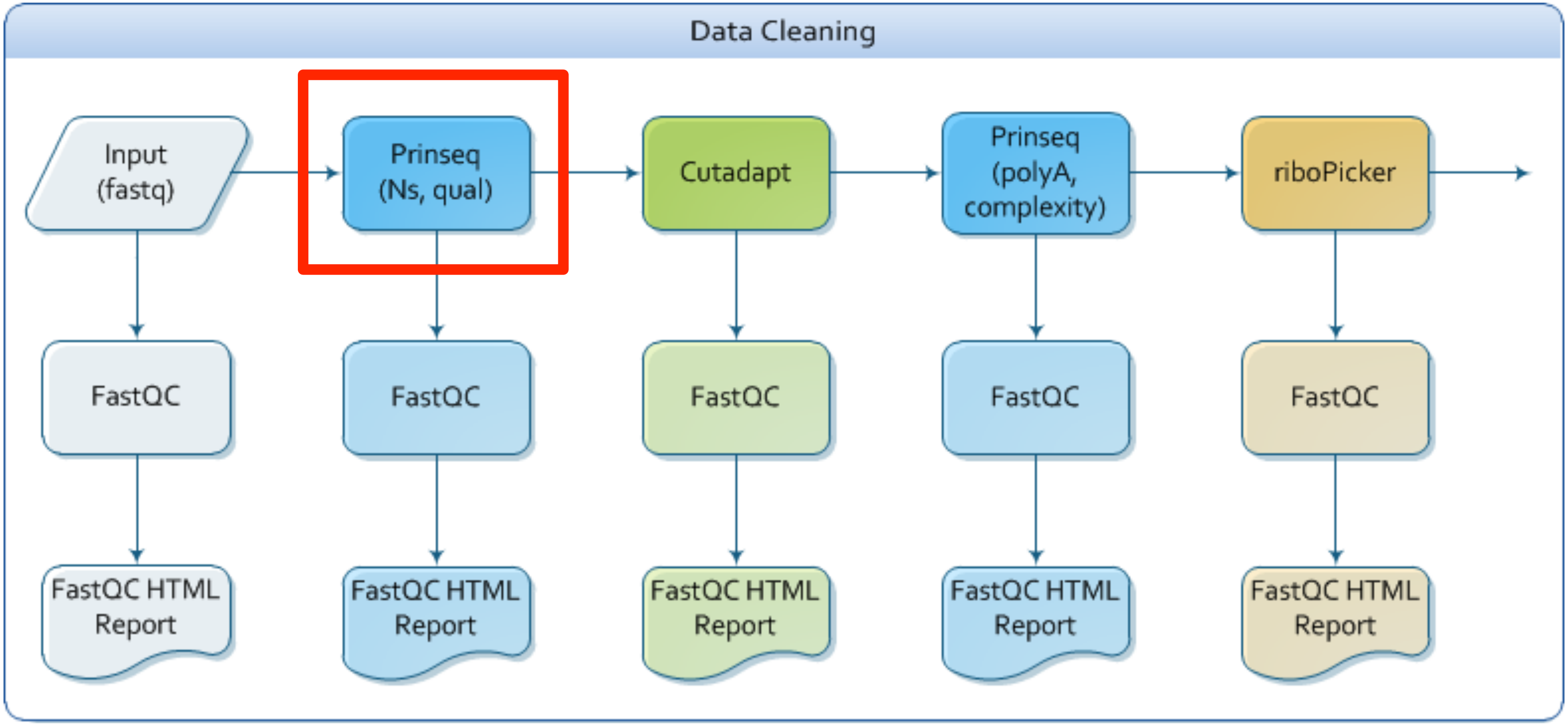




Sequence	Count	Obs/Exp Overall	Obs/Exp Max	Max Obs/Exp Posi
TTTTT	192940	8.590186	21.06293	29
CTGCA	90975	7.7906475	12.251836	10
GCAGA	84910	7.163295	13.539302	23
TGCAG	92470	7.002405	10.671717	11
CCTGC	57235	5.4987235	8.729035	16
GTTTT	108205	5.324498	10.243909	28
CAACC	49005	5.2869425	9.85526	13
ATCGC	58320	4.9942355	8.029807	29
CCAAC	46220	4.9864807	9.408141	12
AAAAA	62285	4.7588468	8.0126295	5
CAGAG	56370	4.7555633	7.148592	20
ACCTG	55315	4.736902	7.919266	15
CGCCA	44035	4.7130895	8.830201	35
GGGGG	63675	4.67525	16.94222	27
GCAGG	55380	4.6350074	17.521912	19
AAAAC	51945	4.452569	8.159592	24
TATCG	64615	4.4271946	8.394971	34
GCTGG	58505	4.3952427	10.37436	18
AACCT	50775	4.382863	7.691214	14
TTATC	70080	4.3444843	7.810299	33
TTTTA	87340	4.332125	7.8541703	28
TTTAT	86645	4.297653	7.9511886	35
CGCTT	54695	4.2042785	6.9374876	31

Quality cleanning

TP

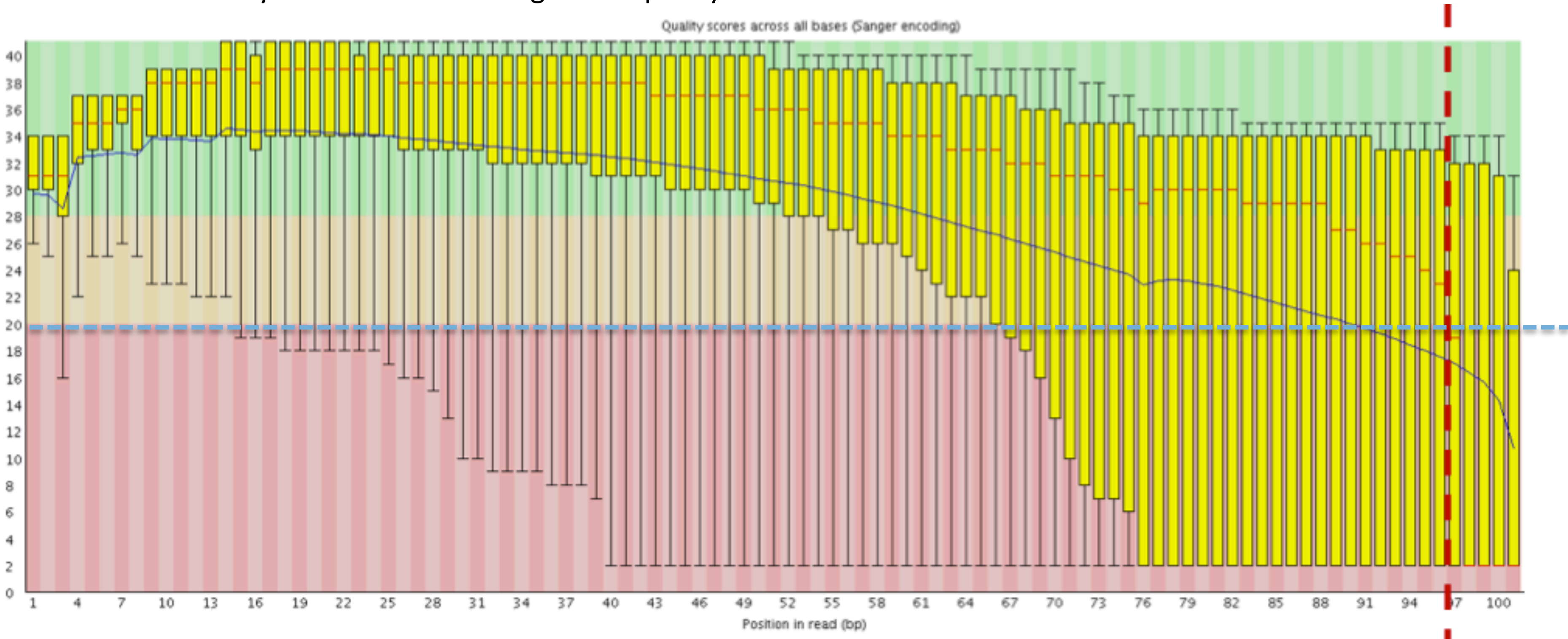


Quality Trimming & Filtering Example (1)

Most lanes will not have problems with sequence bias, GC content, adapters, etc.
Most lanes will have reads with base quality problems. Here is a typical example...
Note: Stringency of base qualities to retain is somewhat application-specific.

Step 1 = Trimming by base quality.
Trim right reads where the base quality falls below 20.

Step 2 = Filtering by base quality.
Retain only reads with an average base quality score ≥ 20.



PRINSEQ step 1

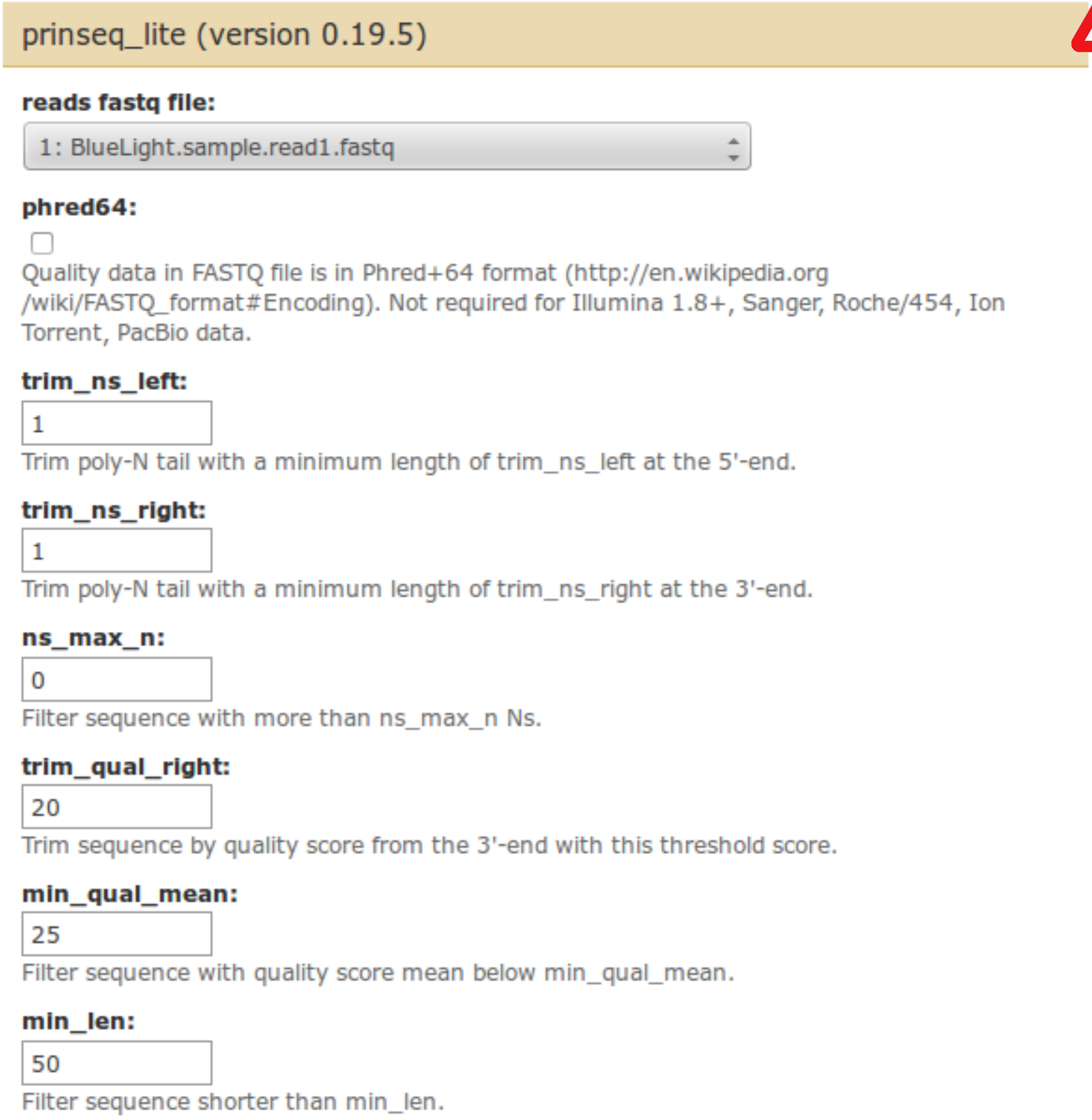
TP

- Removing all unknown nucleotides
  - First by trimming
  - Then by filtering

- Trimming, from 3’ end, nucleotides w/ Q < 20

- Filtering sequences
  - w/ average quality score < 25
  - w/ length < 50

TP



prinseq_lite (version 0.19.5)

reads fastq file:

1: BlueLight.sample.read1.fastq

phred64:

□

Quality data in FASTQ file is in Phred+64 format (http://en.wikipedia.org/wiki/FASTQ_format#Encoding). Not required for Illumina 1.8+, Sanger, Roche/454, Ion Torrent, PacBio data.

trim_ns_left:

1

Trim poly-N tail with a minimum length of trim_ns_left at the 5'-end.

trim_ns_right:

1

Trim poly-N tail with a minimum length of trim_ns_right at the 3'-end.

ns_max_n:

0

Filter sequence with more than ns_max_n Ns.

trim_qual_right:

20

Trim sequence by quality score from the 3'-end with this threshold score.

min_qual_mean:

25

Filter sequence with quality score mean below min_qual_mean.

min_len:

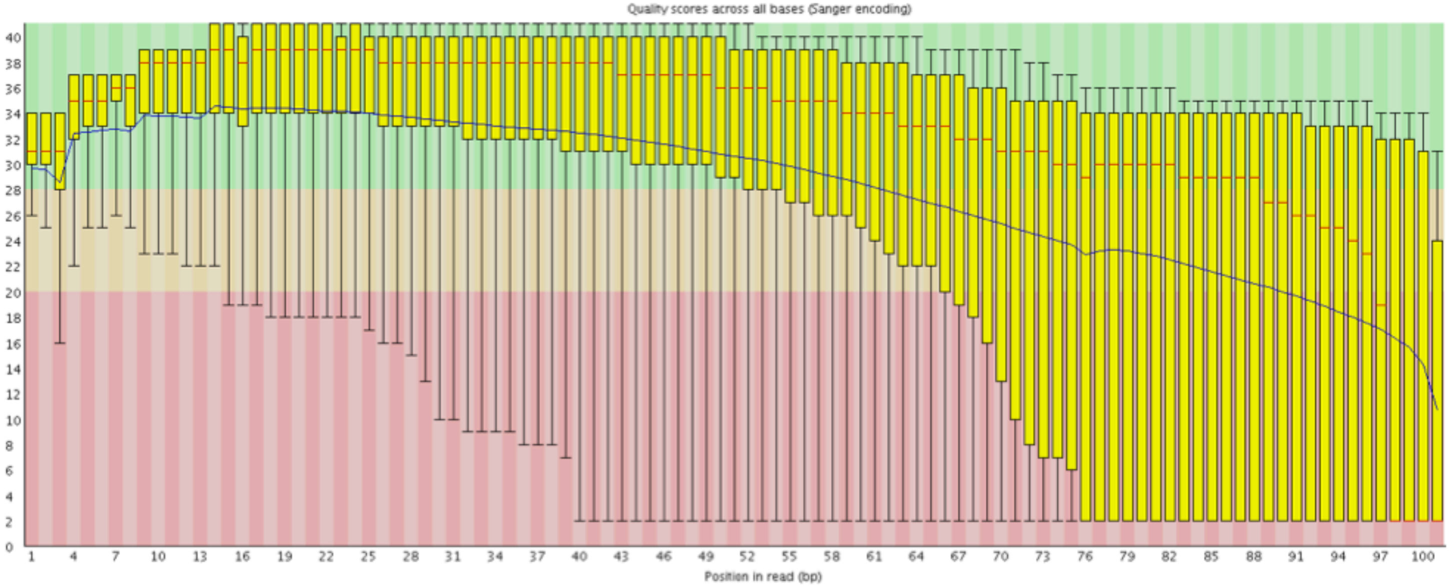
50

Filter sequence shorter than min_len.

Quality Trimming & Filtering Example (2)

BEFORE:

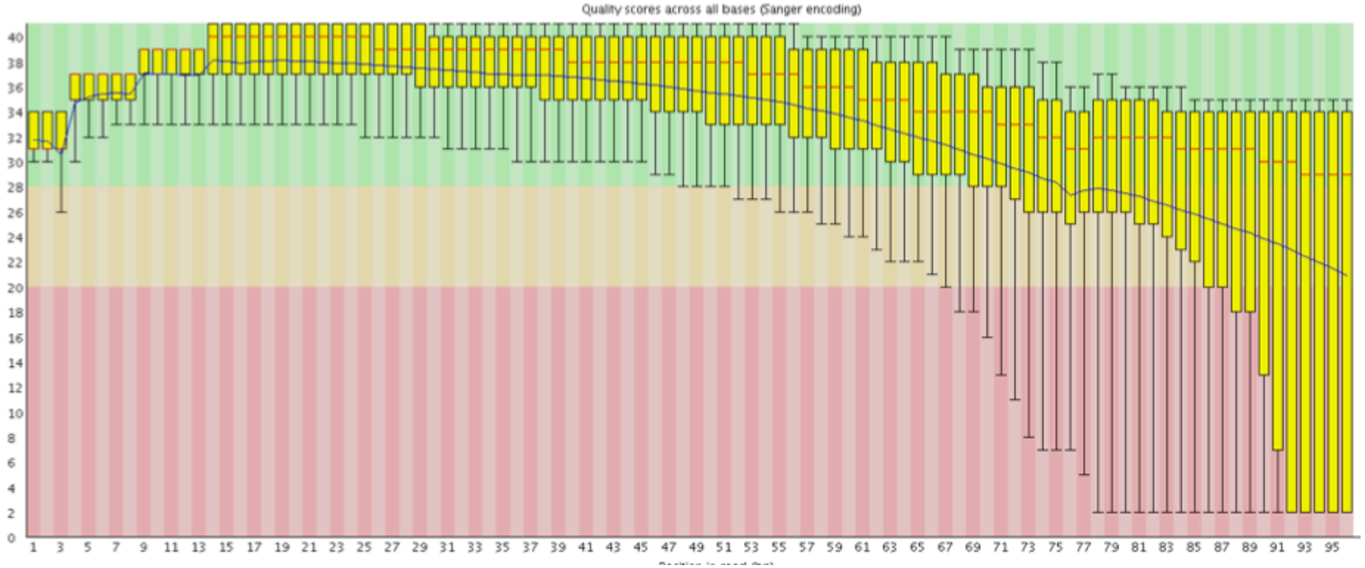
Raw data (i.e. untrimmed & unfiltered) →



AFTER:

Post-trimming & post-filtering base quality distribution →

~82% of reads in this lane pass this QC filter.



PRINSEQ : add stringency

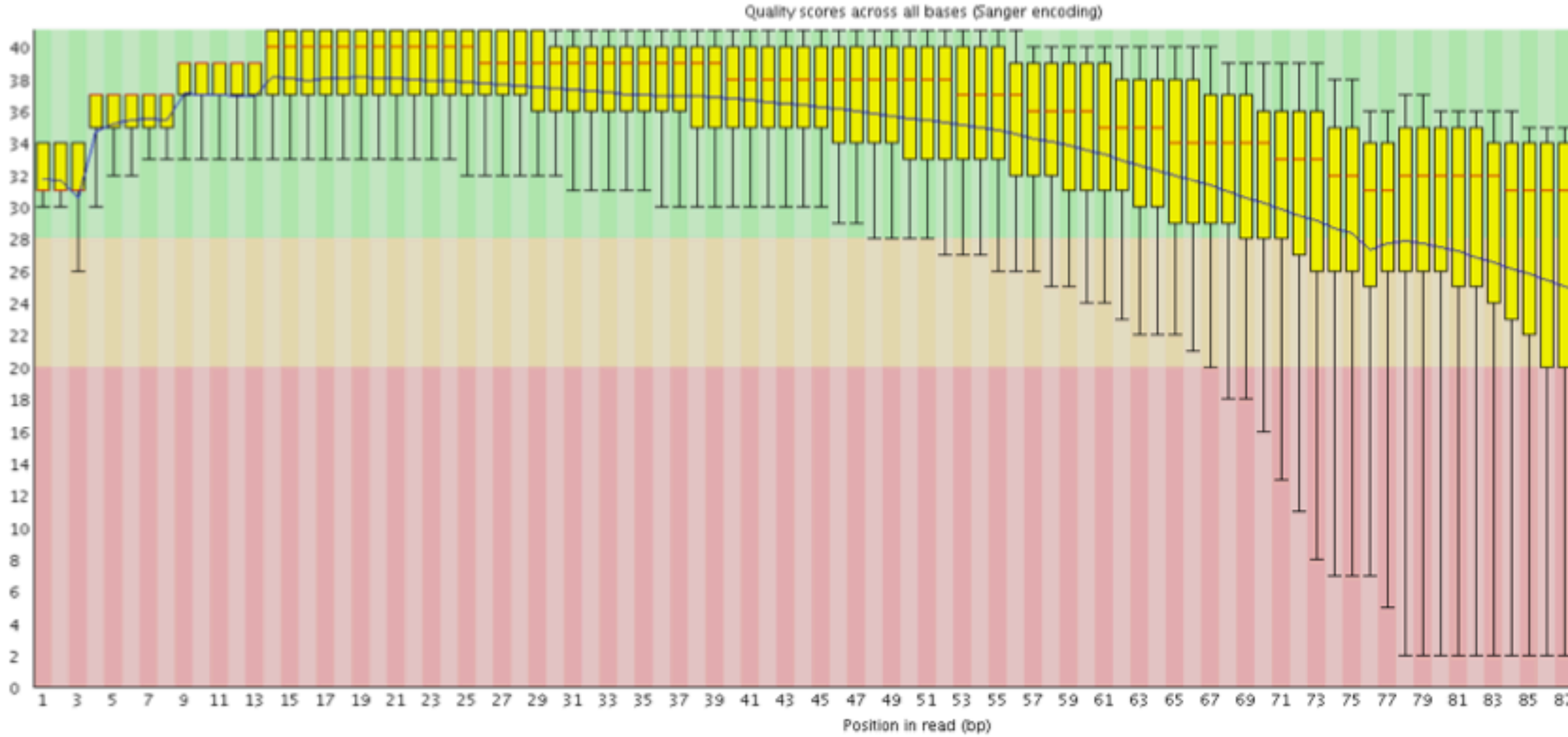
TP

- Removing all unknown nucleotides
  - First by trimming
  - Then by filtering
- Trimming, from 3’ end, nucleotides w/ Q < 20

  Q < 25
- Filtering sequences
  - w/ average quality score < 25 average Q < 30
  - w/ length < 50

Quality Trimming & Filtering Example (3)

More stringent



Trimming effect

Recent publications have identified contradictory results of the effects of trimming raw reads on the quality of the assembly

-> How de novo assemblers manage the variable reads size?

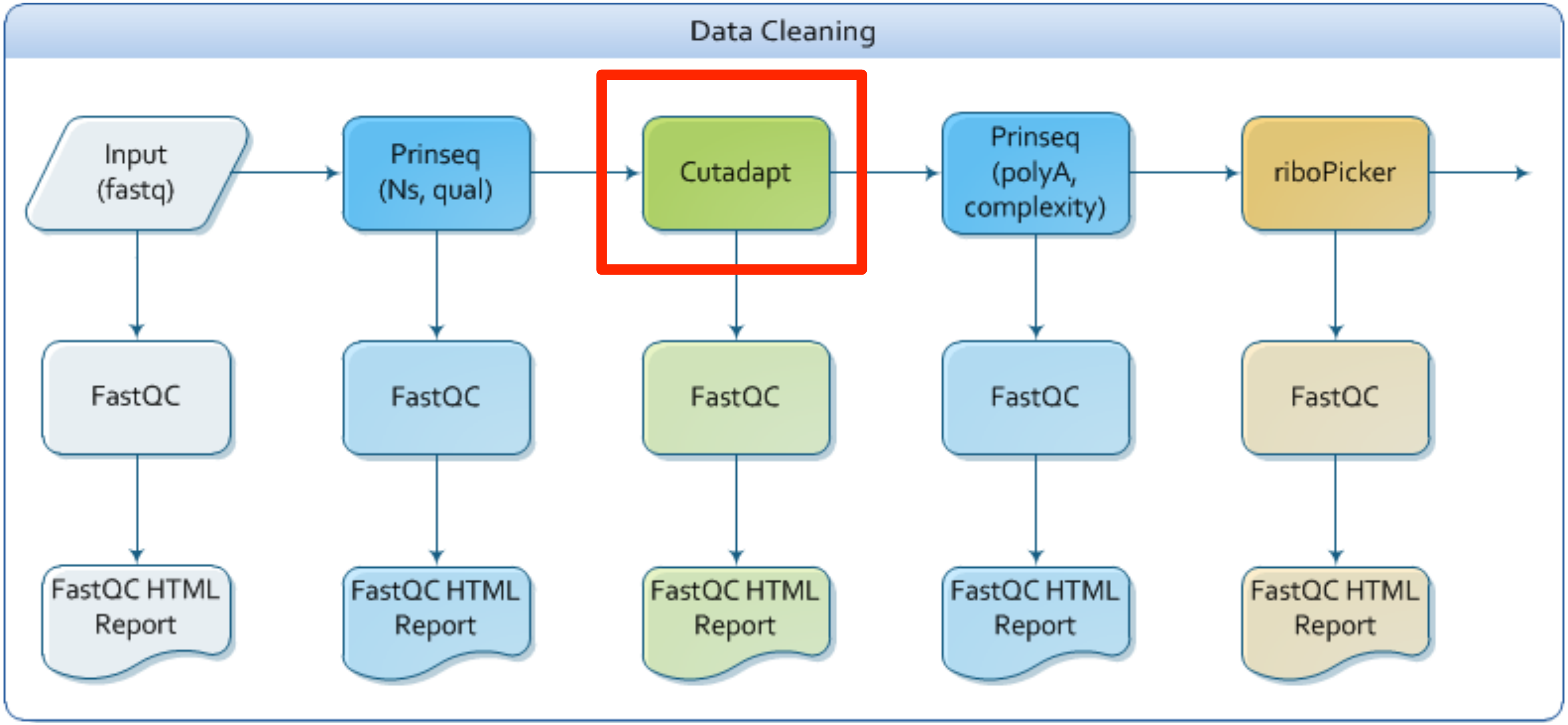
-> Should we prefer a complete removal of the read to the deletion of the only poor quality part?

-> Add later additional cleanning step

Del Fabbro, C., Scalabrin, S., Morgante, M., & Giorgi, F. M. (2013). An Extensive Evaluation of Read Trimming Effects on Illumina NGS Data Analysis. PLoS ONE, 8(12), e85024. doi:10.1371/journal.pone.0085024
MacManes, M. D. (2014, November). On the optimal trimming of high-throughput mRNAseq data. Biorxiv. doi:10.1101/000422
Sleep, J. A., Schreiber, A. W., & Baumann, U. (2013). Sequencing error correction without a reference genome. BMC Bioinformatics, 14(1), 367. doi:10.1186/gb-2011-12-11-r112

Adaptor cleanning

TP



Cutadapt

1. Compute optimal alignment between the read and the adapter sequences. The type of alignment produced is called end-space (or regular semi-global) alignment. It does not penalize initial or trailing gaps.
2. Depending on the parameter used (-a -b -g) cutadapt considers that you know where the adapter is located or not.

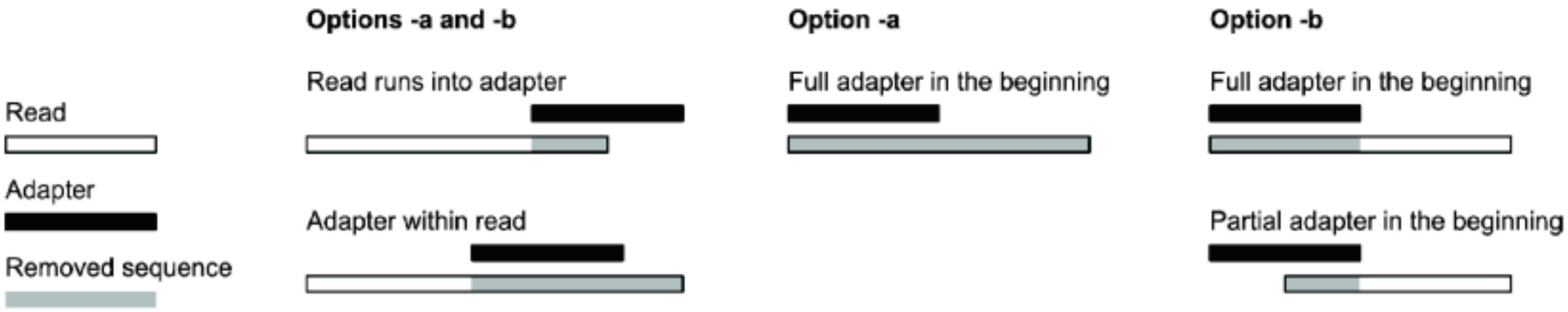


Figure 1. This illustration shows all possible alignment configurations between the read and adapter sequence. There are two different trimming behaviours, triggered by whether option "-a" or "-b" is used to provide the adapter sequence. Note that the case "Partial adapter in the beginning" is not possible with option "-a", as the alignment algorithm prevents it.

M. Martin. Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet.journal, North America, 17, May 2011. Available at: http://journal.embnet.org/index.php/embnetjournal/article/view/

Cutadapt

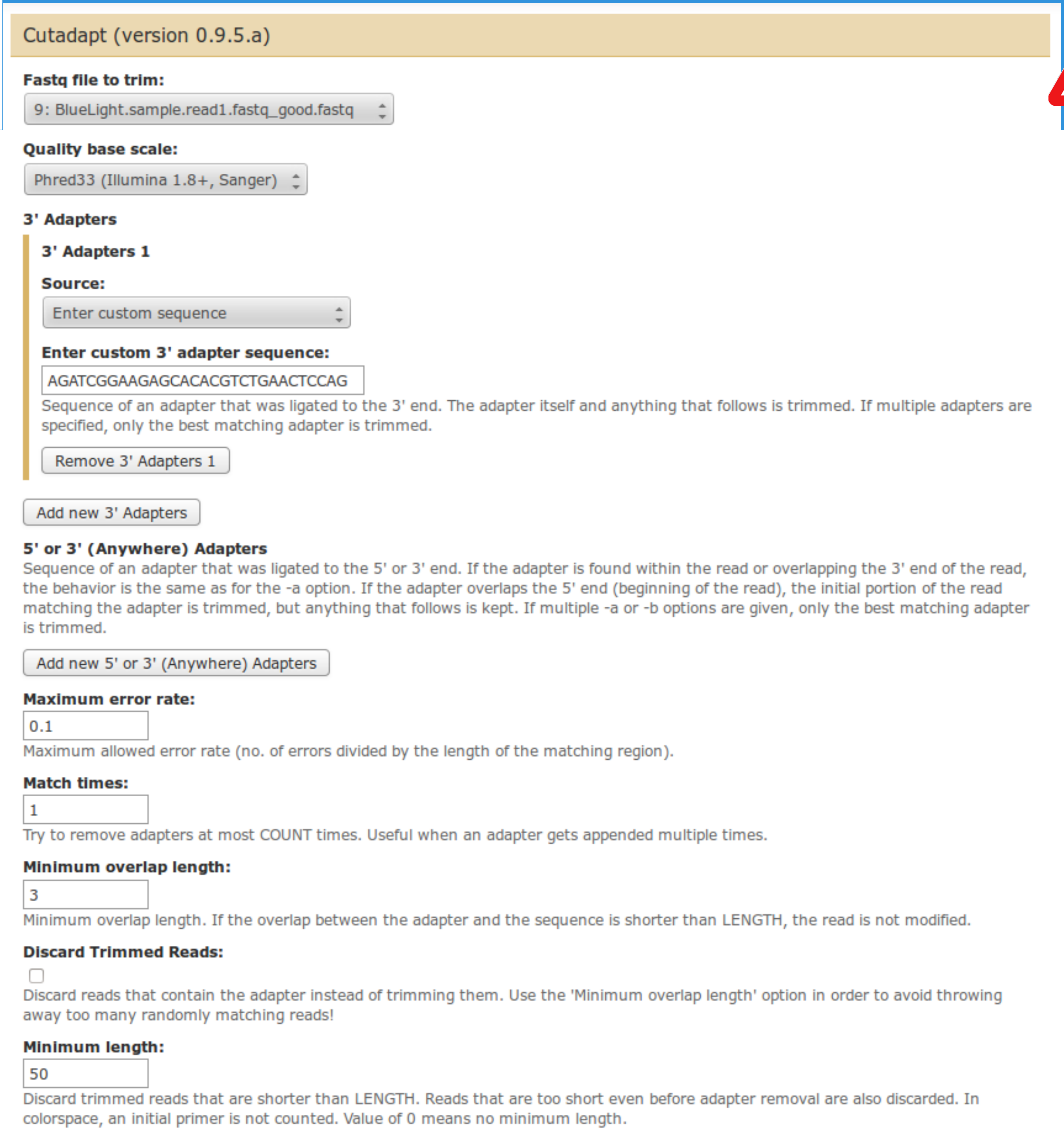
TP

- Trimming from 3’end

AGATCGGAAGAGCACACGTCTGAACTCCAG

- Filtering short reads (< 50 nu)

TP



Cutadapt (version 0.9.5.a)

Fastq file to trim:

9: BlueLight.sample.read1.fastq_good.fastq

Quality base scale:

Phred33 (Illumina 1.8+, Sanger)

3' Adapters

3' Adapters 1

Source:

Enter custom sequence

Enter custom 3' adapter sequence:

AGATCGGAAGAGCACACGTCTGAACTCCAG

Sequence of an adapter that was ligated to the 3' end. The adapter itself and anything that follows is trimmed. If multiple adapters are specified, only the best matching adapter is trimmed.

5' or 3' (Anywhere) Adapters

Sequence of an adapter that was ligated to the 5' or 3' end. If the adapter is found within the read or overlapping the 3' end of the read, the behavior is the same as for the -a option. If the adapter overlaps the 5' end (beginning of the read), the initial portion of the read matching the adapter is trimmed, but anything that follows is kept. If multiple -a or -b options are given, only the best matching adapter is trimmed.

Maximum error rate:

0.1

Maximum allowed error rate (no. of errors divided by the length of the matching region).

Match times:

1

Try to remove adapters at most COUNT times. Useful when an adapter gets appended multiple times.

Minimum overlap length:

3

Minimum overlap length. If the overlap between the adapter and the sequence is shorter than LENGTH, the read is not modified.

Discard Trimmed Reads:

☐

Discard reads that contain the adapter instead of trimming them. Use the 'Minimum overlap length' option in order to avoid throwing away too many randomly matching reads!

Minimum length:

50

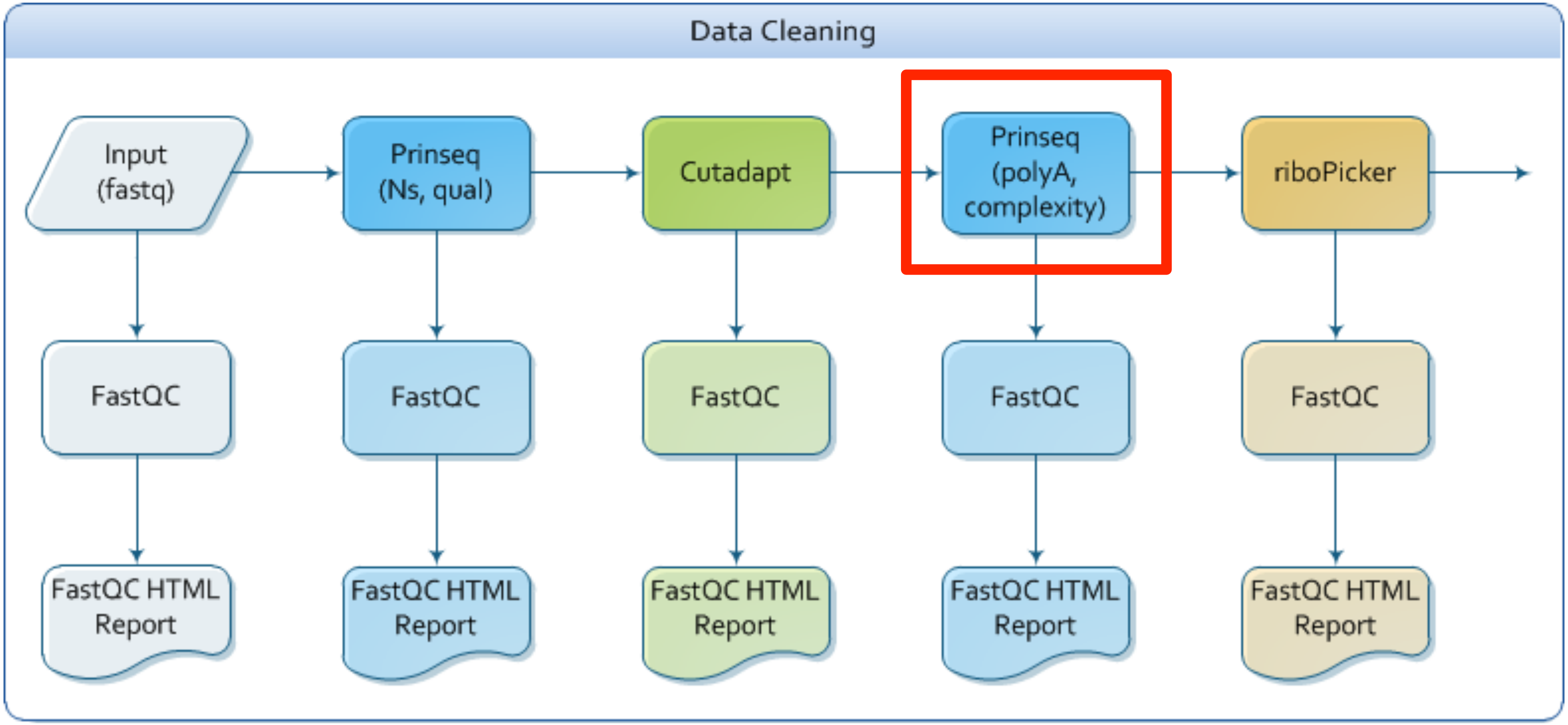
Discard trimmed reads that are shorter than LENGTH. Reads that are too short even before adapter removal are also discarded. In colorspace, an initial primer is not counted. Value of 0 means no minimum length.

CNRS UPMC
Station Biologique
Roscoff

ABiMS

PolyA and low complexity cleaning

TP

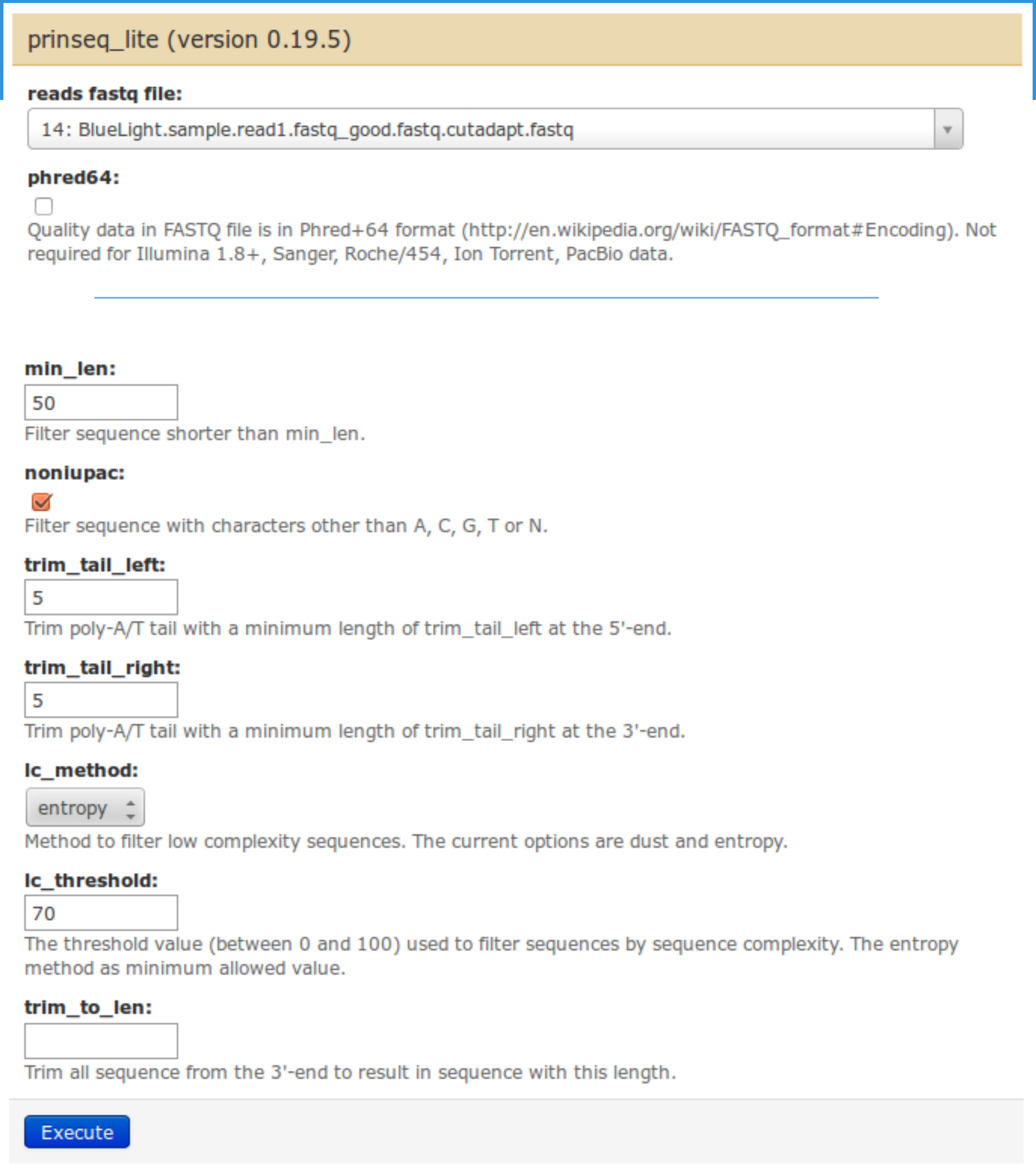


PRINSEQ step 2

TP

- Trimming poly A/T tails
  - From 5’-end and 3’-end
  - w/ nucleotide nb >= 5
- Filtering low complexity sequences
  - Entropy < 70 (out of 100)
- Filtering short reads (< 50 nu)

TP



prinseq_lite (version 0.19.5)

reads fastq file:

14: BlueLight.sample.read1.fastq_good.fastq.cutadapt.fastq

phred64:

☐

Quality data in FASTQ file is in Phred+64 format (http://en.wikipedia.org/wiki/FASTQ_format#Encoding). Not required for Illumina 1.8+, Sanger, Roche/454, Ion Torrent, PacBio data.

min_len:

50

Filter sequence shorter than min_len.

noniupac:

☑

Filter sequence with characters other than A, C, G, T or N.

trim_tail_left:

5

Trim poly-A/T tail with a minimum length of trim_tail_left at the 5'-end.

trim_tail_right:

5

Trim poly-A/T tail with a minimum length of trim_tail_right at the 3'-end.

lc_method:

entropy

Method to filter low complexity sequences. The current options are dust and entropy.

lc_threshold:

70

The threshold value (between 0 and 100) used to filter sequences by sequence complexity. The entropy method as minimum allowed value.

trim_to_len:

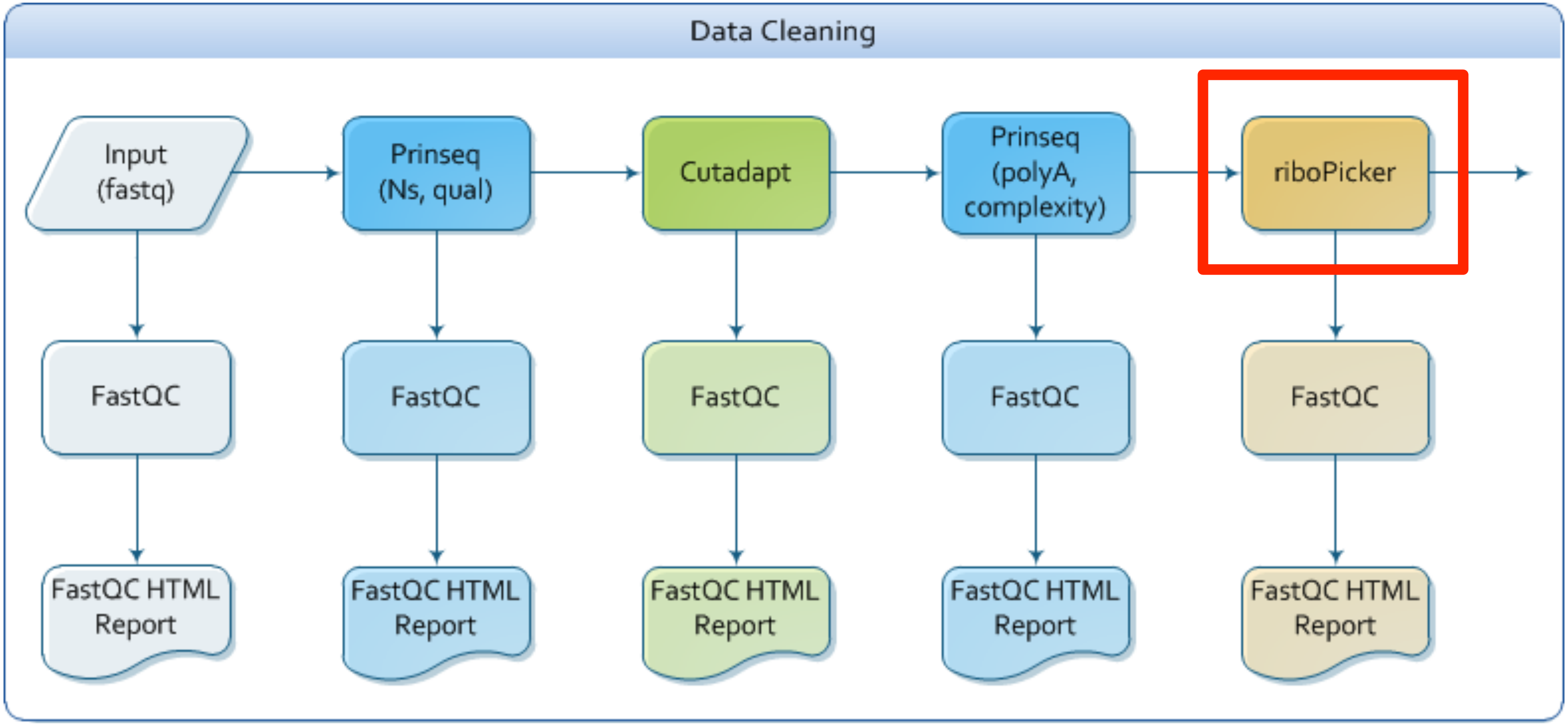
Trim all sequence from the 3'-end to result in sequence with this length.

PRINSEQ step 2

TP

- Trimming poly A/T tails
  - From 5’-end and 3’-end
  - w/ nucleotide nb >= 5
- Filtering low complexity sequences
  - Entropy < 70 (out of 100) Entropy < 50
- Filtering short reads (< 50 nu)

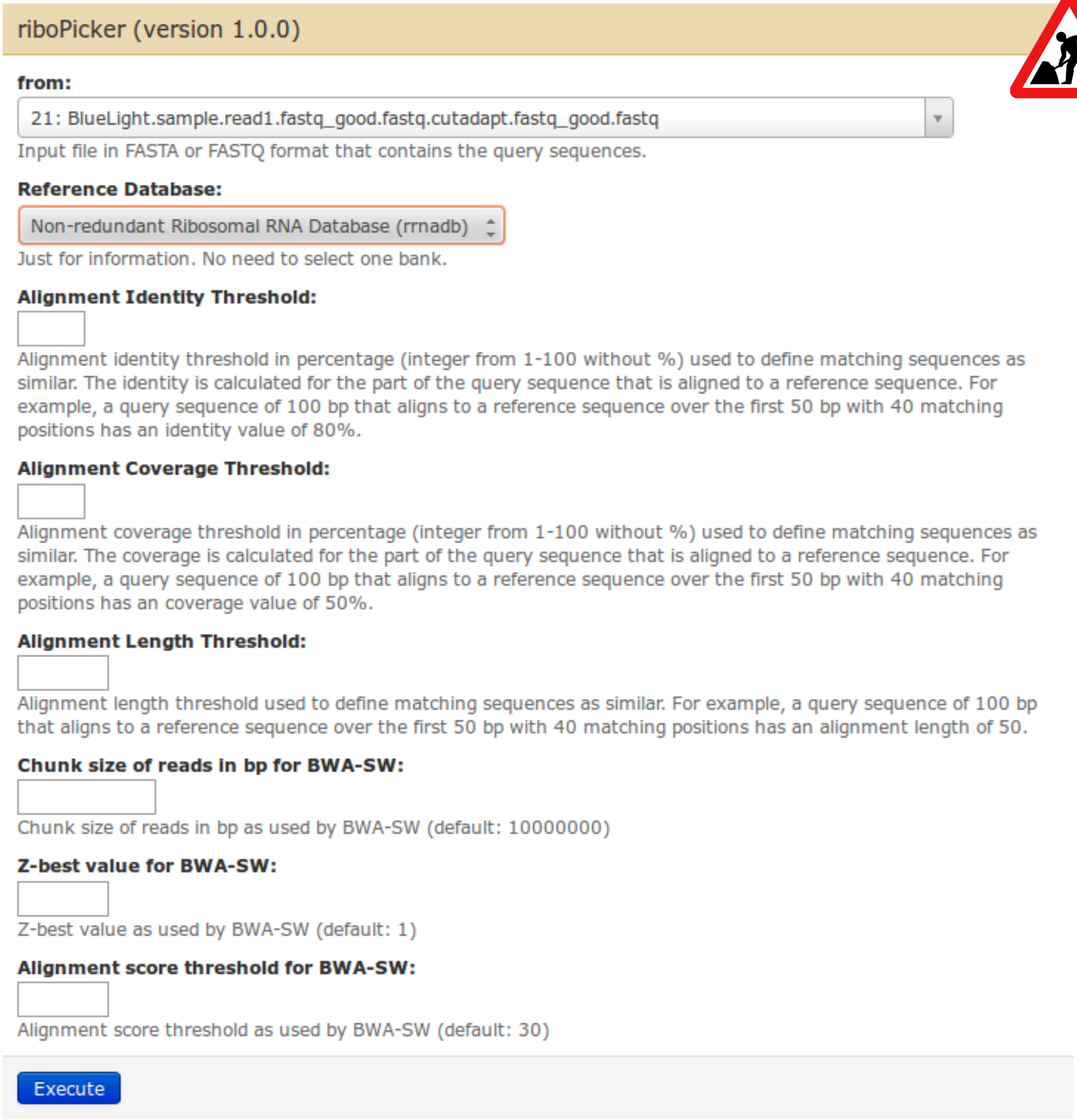
TP


Data Cleaning
Input (fastq)
Prinseq (Ns, qual)
Cutadapt
Prinseq (polyA, complexity)
riboPicker
FastQC
FastQC
FastQC
FastQC
FastQC
FastQC HTML Report
FastQC HTML Report
FastQC HTML Report
FastQC HTML Report
FastQC HTML Report

riboPicker

- Select “rrnadb” as the reference database

TP



riboPicker (version 1.0.0)

from:

21: BlueLight.sample.read1.fastq_good.fastq.cutadapt.fastq_good.fastq

Input file in FASTA or FASTQ format that contains the query sequences.

Reference Database:

Non-redundant Ribosomal RNA Database (rrnadb)

Just for information. No need to select one bank.

Alignment Identity Threshold:

Alignment identity threshold in percentage (integer from 1-100 without %) used to define matching sequences as similar. The identity is calculated for the part of the query sequence that is aligned to a reference sequence. For example, a query sequence of 100 bp that aligns to a reference sequence over the first 50 bp with 40 matching positions has an identity value of 80%.

Alignment Coverage Threshold:

Alignment coverage threshold in percentage (integer from 1-100 without %) used to define matching sequences as similar. The coverage is calculated for the part of the query sequence that is aligned to a reference sequence. For example, a query sequence of 100 bp that aligns to a reference sequence over the first 50 bp with 40 matching positions has an coverage value of 50%.

Alignment Length Threshold:

Alignment length threshold used to define matching sequences as similar. For example, a query sequence of 100 bp that aligns to a reference sequence over the first 50 bp with 40 matching positions has an alignment length of 50.

Chunk size of reads in bp for BWA-SW:

Chunk size of reads in bp as used by BWA-SW (default: 10000000)

Z-best value for BWA-SW:

Z-best value as used by BWA-SW (default: 1)

Alignment score threshold for BWA-SW:

Alignment score threshold as used by BWA-SW (default: 30)

CNRS UPMC
Station Biologique
Roscoff

riboPicker

- For additional databases (chloroplasts, mitochondrions, …) please contact your favorite bioinformatic analysts at support.abims@sb-roscoff.fr

Get Pairs

- Data cleaning is performed on every sequence file without using the paired information

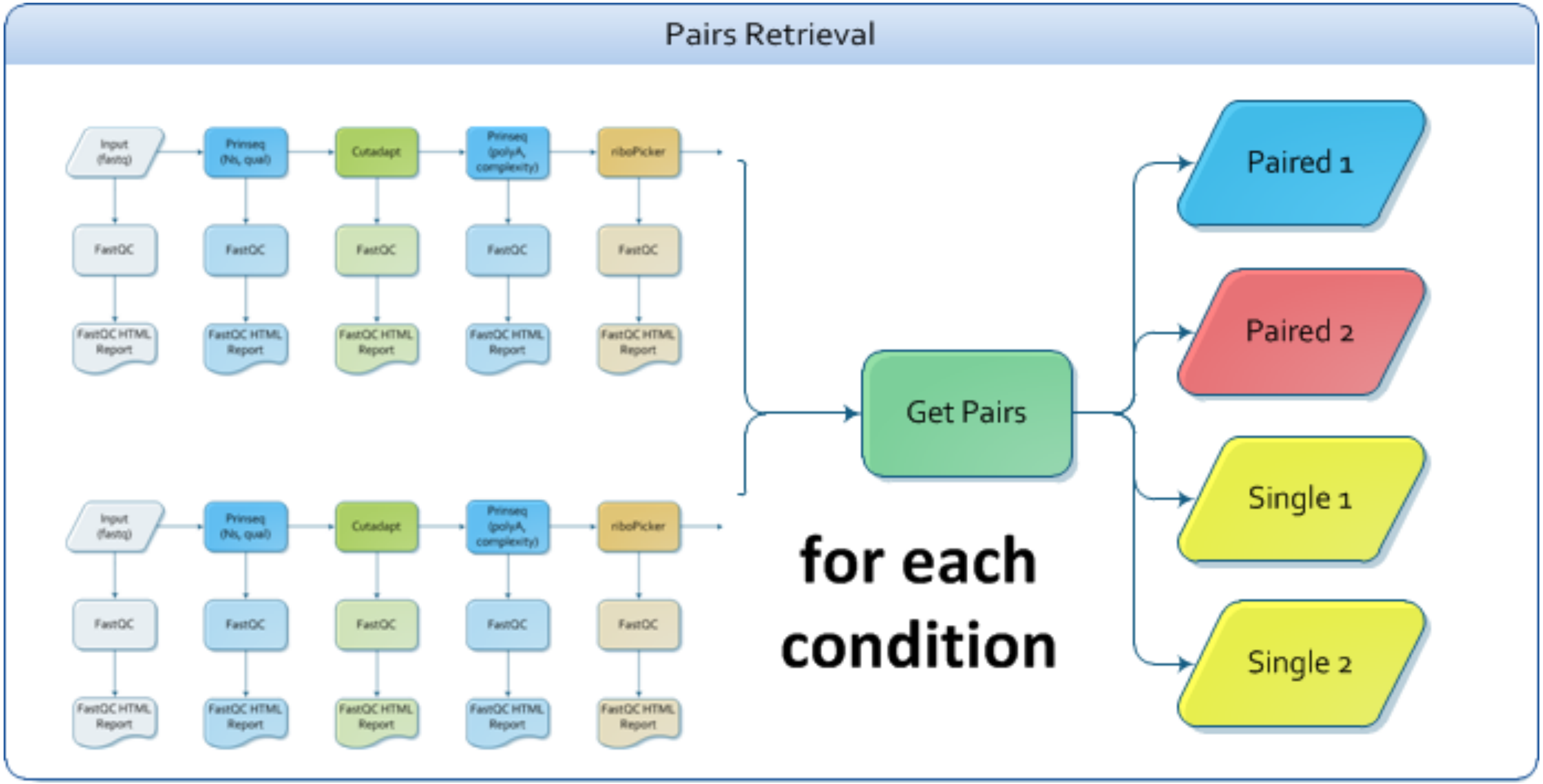
  ➔ Cleaning leads to singletons generation

- Very few tools can work with both paired reads and singletons

- For the next part of the pipeline we need to retrieve paired reads and isolate singletons

Get Pairs

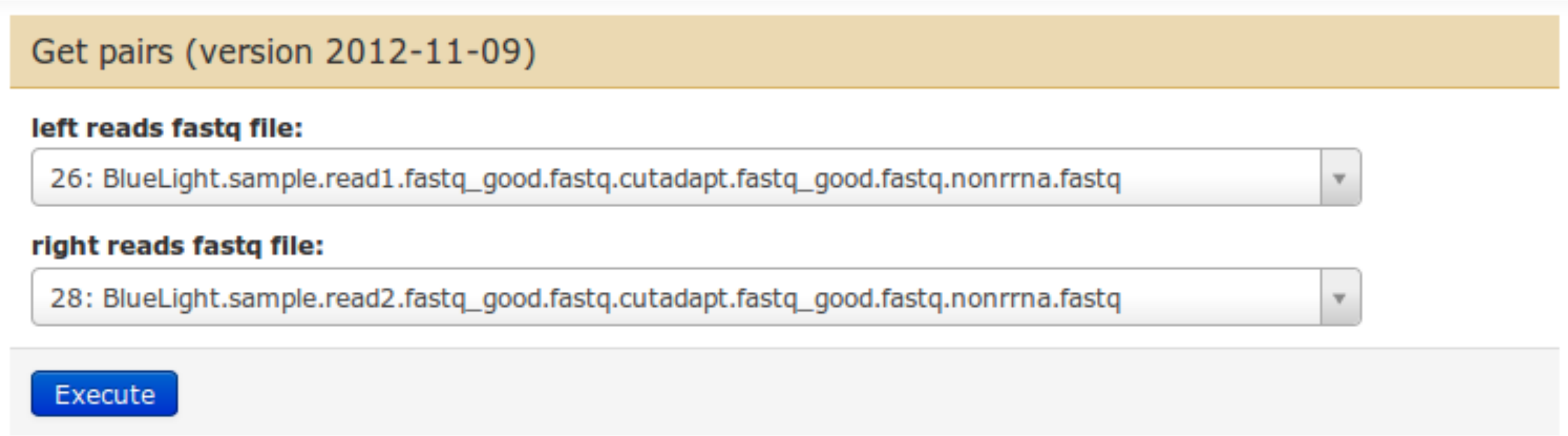
TP



for each condition

Get Pairs

TP



Get pairs (version 2012-11-09)

left reads fastq file:

26: BlueLight.sample.read1.fastq_good.fastq.cutadapt.fastq_good.fastq.nonrrna.fastq

right reads fastq file:

28: BlueLight.sample.read2.fastq_good.fastq.cutadapt.fastq_good.fastq.nonrrna.fastq

Additional optional step

FLASH (Fast Length Adjustment of SHort reads) is a very fast and accurate software tool to merge paired-end reads.

- FLASH is designed to merge pairs of reads when the original DNA fragments are shorter than twice the length of reads.
- The resulting longer reads can significantly improve genome assemblies. They can also improve transcriptome assembly when FLASH is used to merge RNA-seq data

Sequencing error corrections.

Error occur during the sequencing process. These errors impact the assembly process (less identity, larger graphs,...)

Removing these errors before assembly :

- Limits the errors in the contigs
- Speeds the assembly

Many different software packages. Ex. SGA SOAP REPTILE One adapted to RNA-Seq reads = Seecer.

The challenge is to separate errors from rare polymorphisms in an efficient manner.

!!! MacManes, M. D., & Eisen, M. B. (2013). Improving transcriptome assembly through error correction of high-throughput sequence reads. PeerJ, 1, e113.

NGS reads normalization (by Trinity)

- Context:
  - By definition RNAseq display a wide range of expressions Very low expressed → Very highly expressed transcripts
  - The information given by reads from high expression transcripts is redundant, and very high coverage also brings more sequencing errors
  - De-novo assemblers do not benefit from coverage increase beyond a certain point, and fewer data means quicker assemblies

→How to decrease coverage of highly expressed transcripts without decreasing that of low expressed transcripts ?

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

```
>
CAGTCGATCA

>
CGATCAGTCG
```

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	1
GATCA	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	2
GATCA	1

CNRS UPMC
Station Biologique
Roscoff

Abims 4

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	2
GATCA	2

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	2
GATCA	2
ATCAG	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

> CAGTCGATCA

> CGATCAGTCG

CAGTC	1
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	2
GATCA	2
ATCAG	1
TCAGT	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	2
AGTCG	1
GTCGA	1
TCGAT	1
CGATC	2
GATCA	2
ATCAG	1
TCAGT	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	2
AGTCG	2
GTCGA	1
TCGAT	1
CGATC	2
GATCA	2
ATCAG	1
TCAGT	1

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):

e.g. for k = 5

>
CAGTCGATCA

>
CGATCAGTCG

CAGTC	2
AGTCG	2
GTCGA	1
TCGAT	1
CGATC	2
GATCA	2
ATCAG	1
TCAGT	1
…	

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):
   - with k = 25

2. For each read, compute the median, average and stdev kmers coverage

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):
   - with k = 25

2. For each read, compute the median, average and stdev kmers coverage

3. Accept a read with a probability of:

NGS reads normalization (by Trinity)

3. Accept a read with a probability of:

e.g. with $maxcoverage = 30$

Read_A: $median\ coverage = 60 \rightarrow max\_coverage/median = 0.5$

➔ Read_A has a 50% chance of being kept

Read_B: $median\ coverage = 10 \rightarrow max\_coverage/median = 3$

➔ Read_B has a 300% chance of being kept ;-)

➔ Read_B will be kept

NGS reads normalization (by Trinity)

3. Accept a read with a probability of:

Read_A comes from a highly expressed transcript and is 2 times more covered than the threshold. We know its information is also contained by other reads.

➔ So it has less chance to be kept.

Read_B comes from a low expressed transcript, way below the threshold. Its information is not very redondant, we will need it for the assembly.

➔ So it will absolutly be kept

NGS reads normalization (by Trinity)

1. Count kmers in all the data (Jellyfish):
    - with k = 25

2. For each read, compute the median, average and stdev kmers coverage

3. Accept a read with a probability of:

4. Remove a read if: (100%)

NGS reads normalization (by Trinity)

4. Remove a read if: (100%)

is also known as the coefficient of variation (CV)

The CV measures the dispersion of the values

Applied to NGS reads the CV is an indication of the variability in the kmer coverage of a read

A high variability in a read kmer coverage means there is probably a lot of sequencing errors in this read

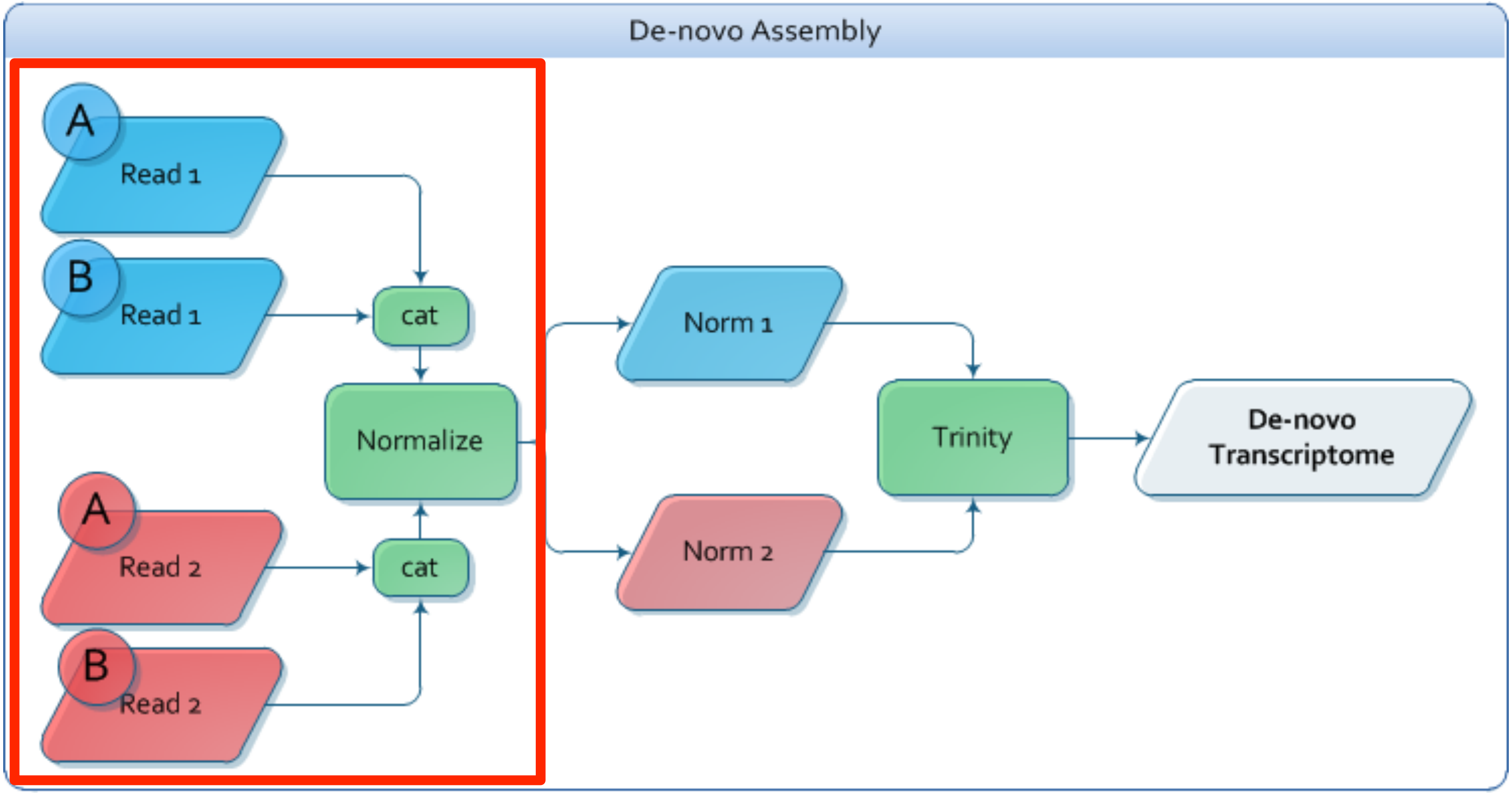
NGS reads normalization (by Trinity)

- Pros:
  - Reduce the data to be assembled
    → faster assemblies
    → RAM requirement highly reduced
  - Remove reads with potentially lots of sequencing errors
    → better assemblies ?

- Cons:
  - Small loss of information → slightly worse assemblies ?
  - Stringent filter on kmer coverage variability
    → loss of low expressed alternative transcripts (splice junctions) ?

Normalization

TP

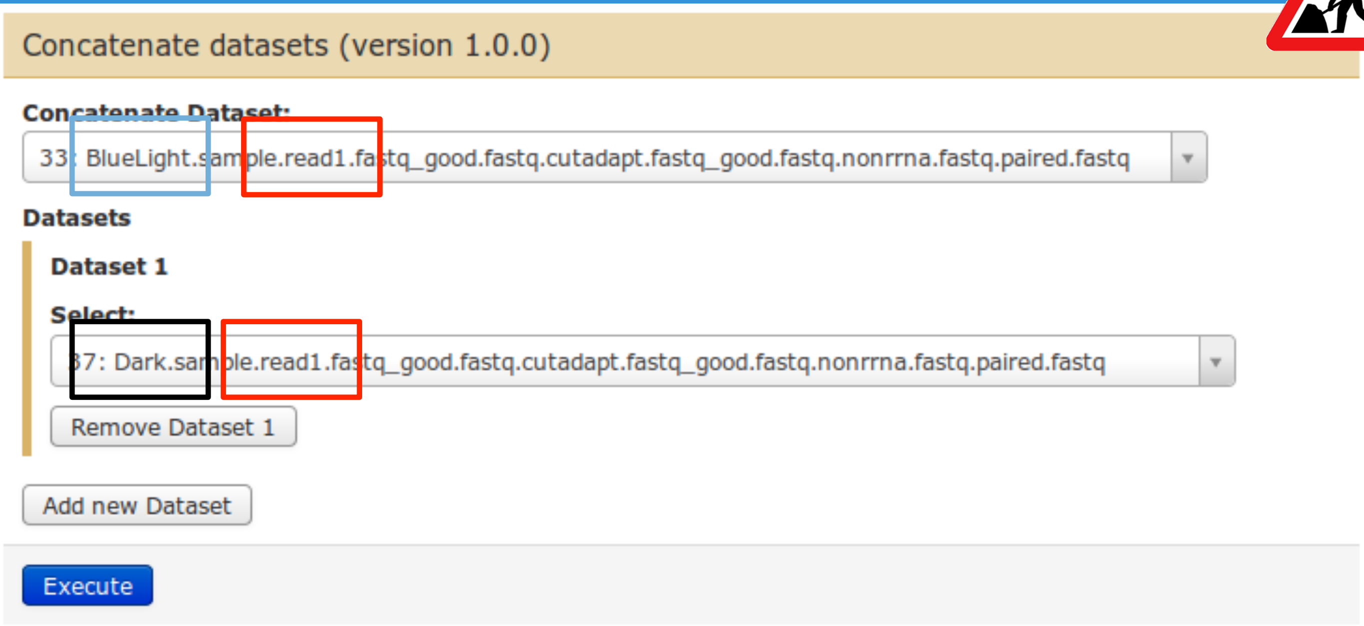


NGS reads normalization (by Trinity)

TP

- Concatenate left reads from all conditions
  → all.read1.fastq
- Concatenate right reads from all conditions
  → all.read2.fastq

- Normalize by kmer coverage:
  - Paired: all.read1.fastq & all.read2.fastq
  - pairs together
  - max coverage = 30
  - max pct stdev = 100



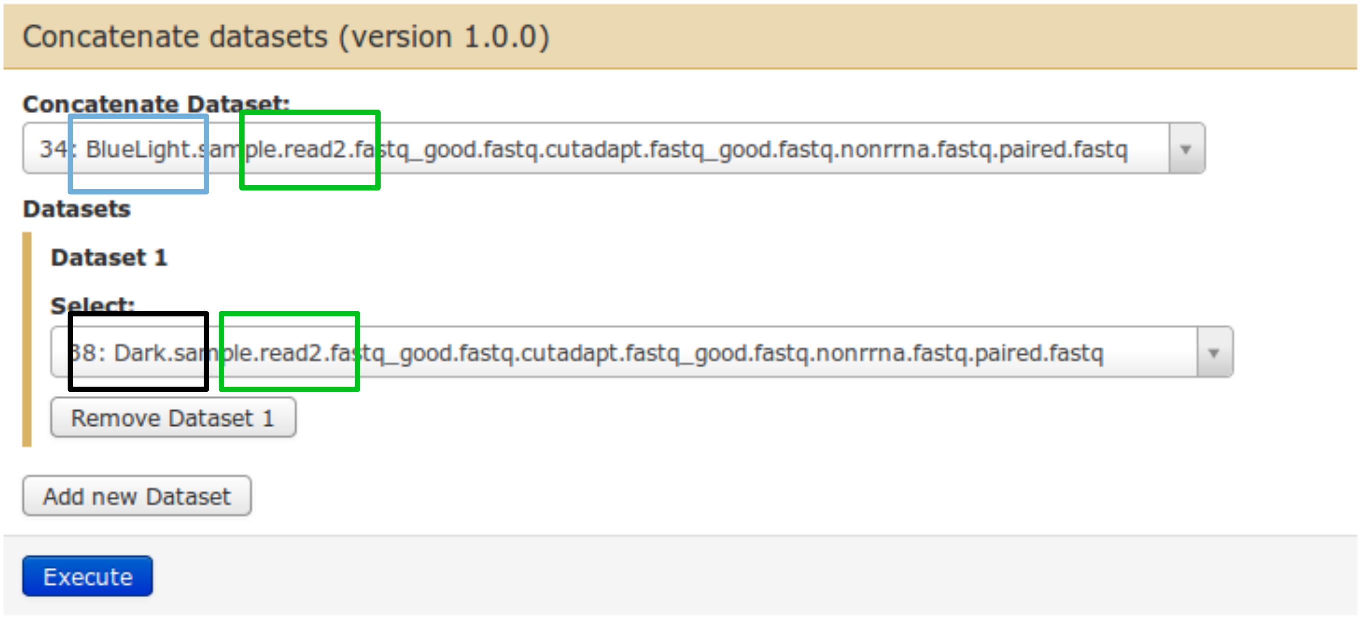
Concatenate datasets (version 1.0.0)

Concatenate Dataset:
33: BlueLight.sample.read1.fastq_good.fastq.cutadapt.fastq_good.fastq.nonrrna.fastq.paired.fastq

Datasets

Dataset 1

Select:
37: Dark.sample.read1.fastq_good.fastq.cutadapt.fastq_good.fastq.nonrrna.fastq.paired.fastq



Concatenate datasets (version 1.0.0)

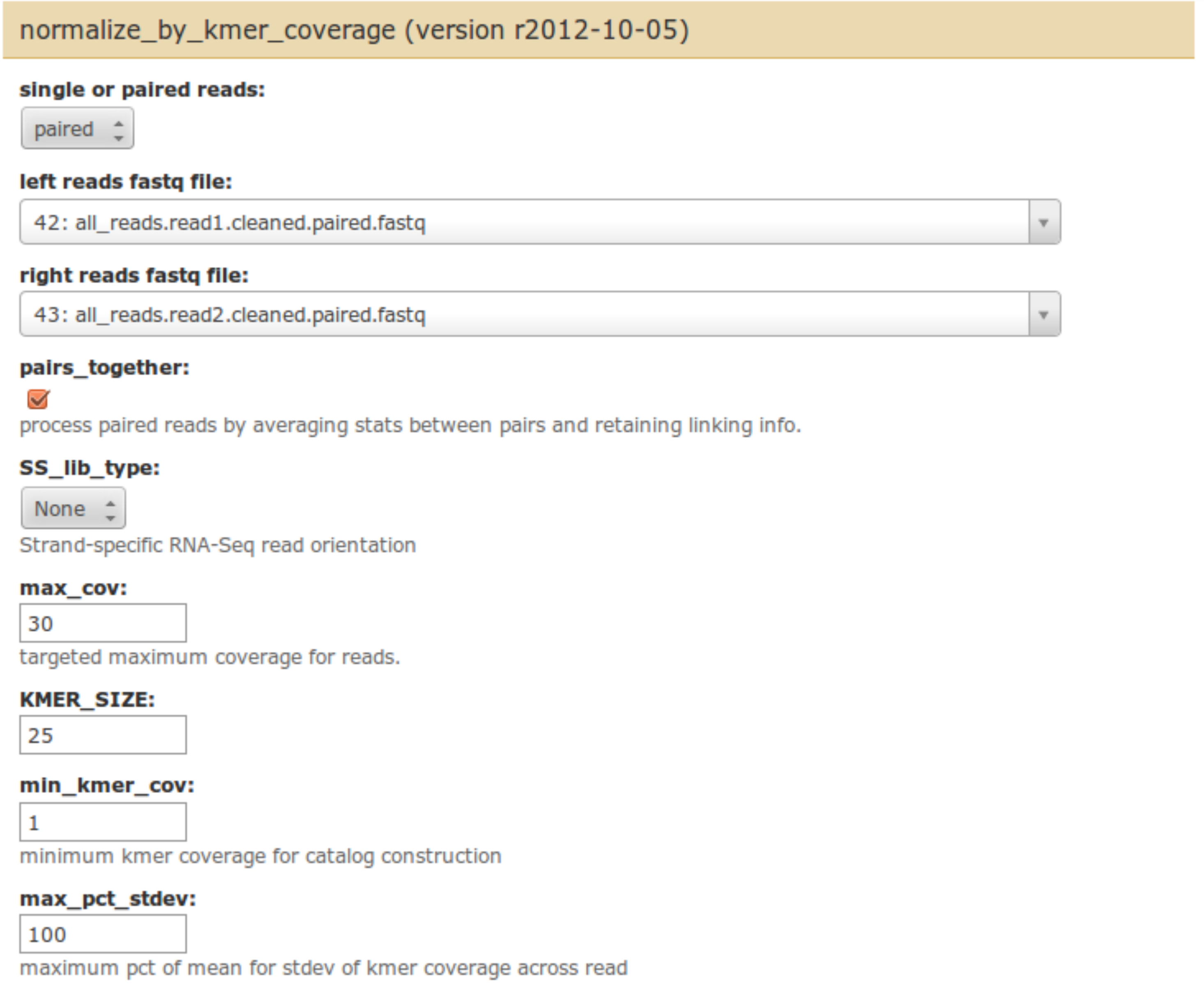
Concatenate Dataset:
34: BlueLight.sample.read2.fastq_good.fastq.cutadapt.fastq_good.fastq.nonrrna.fastq.paired.fastq

Datasets

Dataset 1

Select:
38: Dark.sample.read2.fastq_good.fastq.cutadapt.fastq_good.fastq.nonrrna.fastq.paired.fastq

TP



normalize_by_kmer_coverage (version r2012-10-05)

single or paired reads:

paired

left reads fastq file:

42: all_reads.read1.cleaned.paired.fastq

right reads fastq file:

43: all_reads.read2.cleaned.paired.fastq

pairs_together:

☑

process paired reads by averaging stats between pairs and retaining linking info.

SS_lib_type:

None

Strand-specific RNA-Seq read orientation

max_cov:

30

targeted maximum coverage for reads.

KMER_SIZE:

25

min_kmer_cov:

1

minimum kmer coverage for catalog construction

max_pct_stdev:

100

maximum pct of mean for stdev of kmer coverage across read