The Well-Dressed Bioinformatician

(NGS Computational Methods)

I got some sequence data. What now?

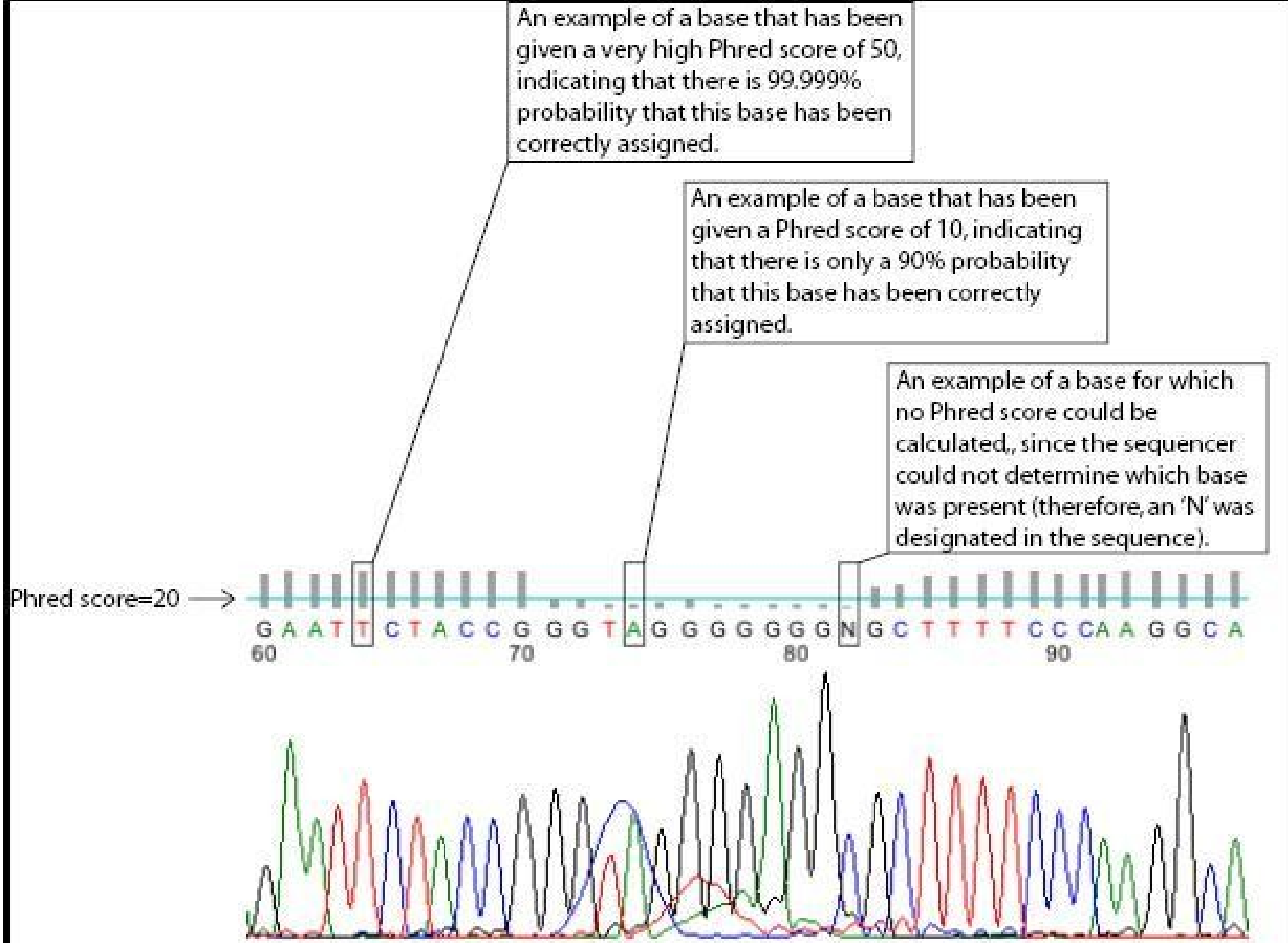
I. **File Formats**
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

Some common file formats

- FASTA
  - Just the sequence, ma'am
- FASTQ
  - FASTA + “Quality” score for each base.
  - Quality is ASCII (33 + Phred score)

```
@SEQ_ID

GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```



An example of a base that has been given a very high Phred score of 50, indicating that there is 99.999% probability that this base has been correctly assigned.

An example of a base that has been given a Phred score of 10, indicating that there is only a 90% probability that this base has been correctly assigned.

An example of a base for which no Phred score could be calculated,, since the sequencer could not determine which base was present (therefore, an 'N' was designated in the sequence).

Figure 1. An example of a DNA sequence tracing and the Phred score (grey bars) corresponding to each colored peak. The colored peaks on the trace correspond to each DNA letter. For example 'T' bases are represented in red, and this sequence has four 'T' bases on a row, as viewed by the four red peaks in the sequence. The aqua horizontal line placed across the grey bars represents a Phred score of 20 which is considered an acceptable level of accuracy. As indicated in Table 1, a Phred score of 20 corresponds to a 99% accuracy in the base call. Therefore, bars above this line indicate base calls that have a higher than 99% probability of being correct. Those below have less than a 99% probability of being correct. Sequence tracing program is courtesy of FinchTV (www.geospiza.com).

Phred scores

Q = Phred score

P = error probability

$$Q = -10 \log_{10} P$$

$$P = 10^{\frac{-Q}{10}}$$

More file formats

- SRA
  - native format for Sequence Read Archive (repository of NGS data)
  - Aligned sequences can be compressed by genome reference
  - Convert to FASTQ with **sra-toolkit** (fastq-dump)

More file formats

- SAM
  - **S**equence **A**lignment / **M**ap
  - generic format for aligned sequences
- BAM
  - binary version of SAM
  - smaller
  - most software wants this
- **Samtools**
  - Convert BAM to SAM
  - Sort and index BAMs
  - Count number of reads at a position (pileup)
  - pysam: a python wrapper
  -

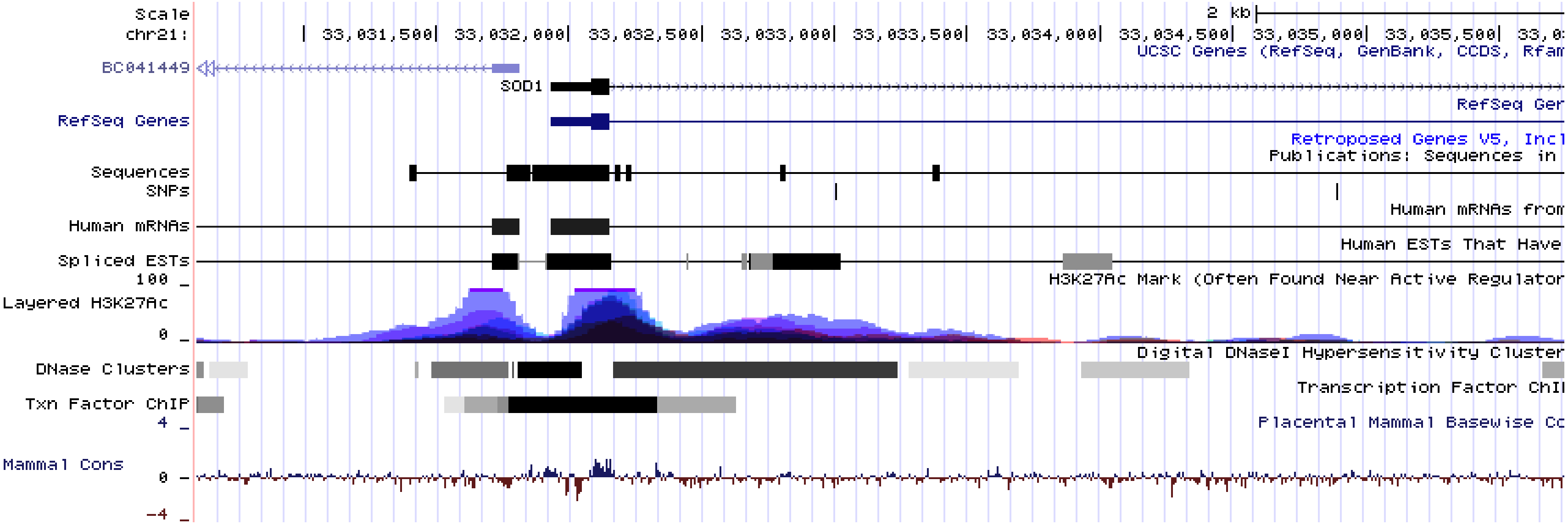
More file formats

- BED
  - Discrete genome annotations (regions)
    - Gene positions
    - Exons / Introns
    - Transcription start sites
    - Protein binding sites
    - Histone locations
    - etc.
  - Contains:
    - Chromosome
    - Start position
    - Stop position
    - Description fields

- **Bedtools**
- GFF and GTF are similar formats.
- All are tab-separated, human-readable

More file formats

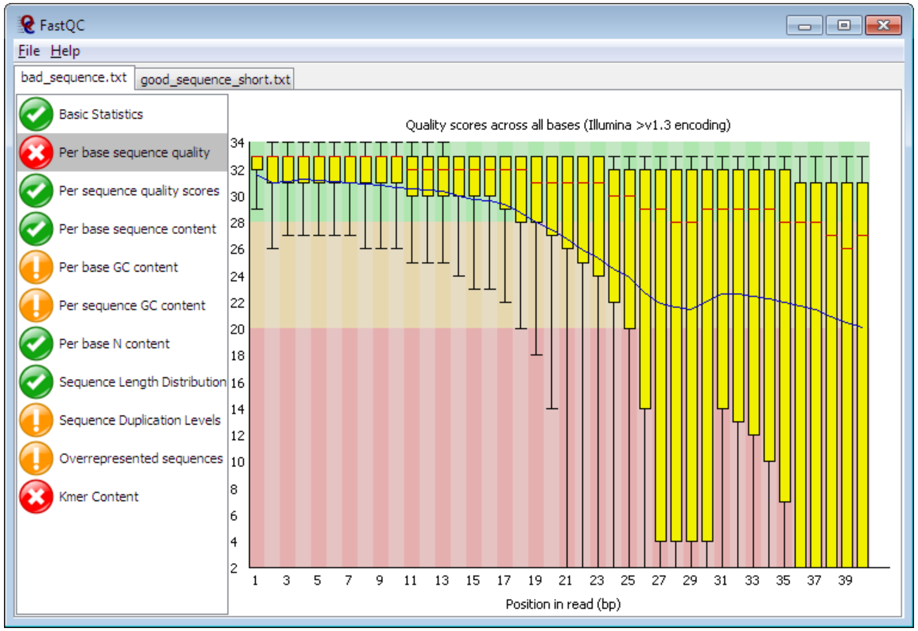
- wig and bigWig
  - Wiggle tracks
  - Continuous-valued genome annotations



I. File Formats
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

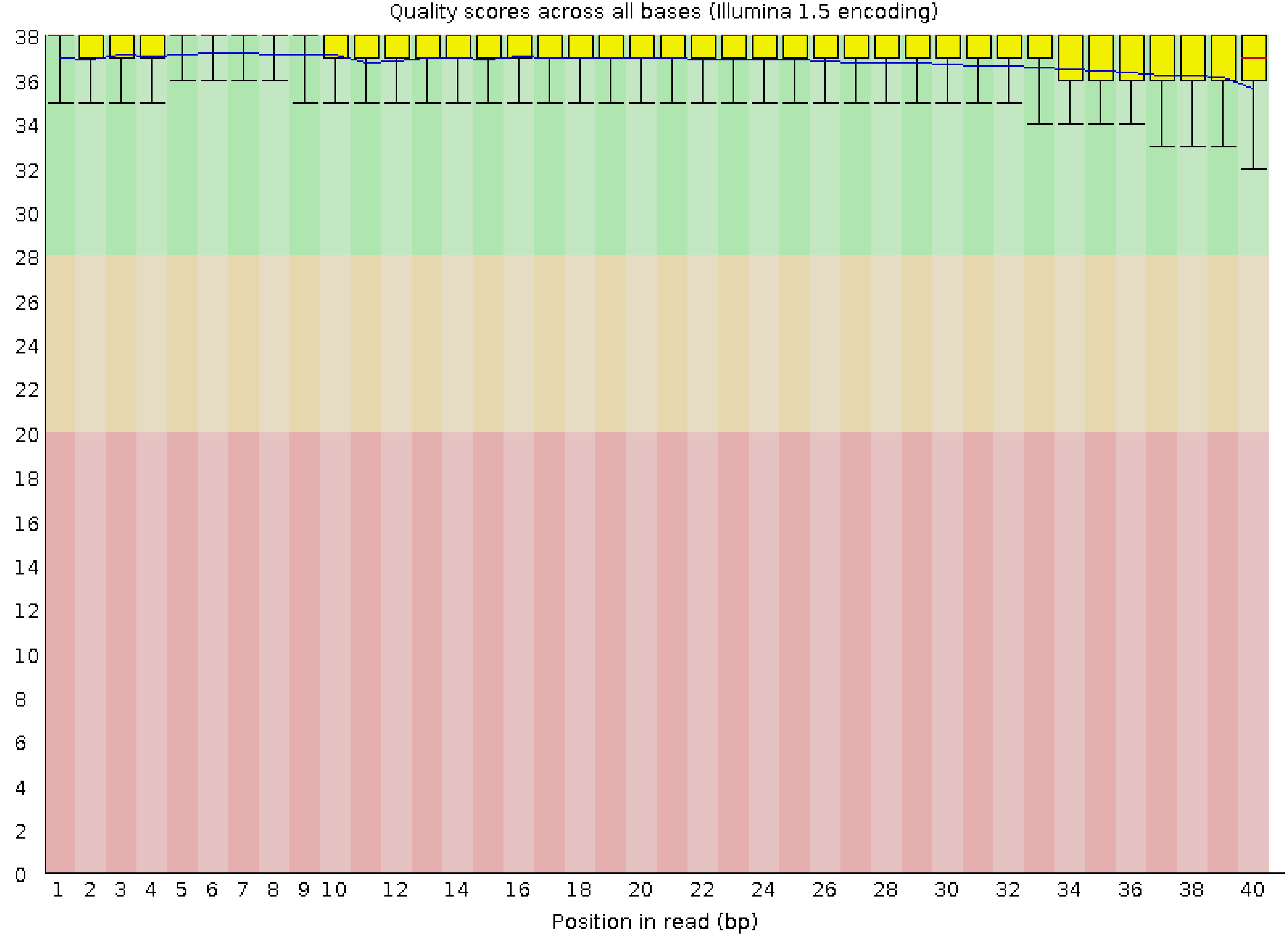
FASTQC

- Graphical tool to evaluate NGS quality
- Can be run in interactive or command line mode
- Generates HTML reports
- Evaluates results:

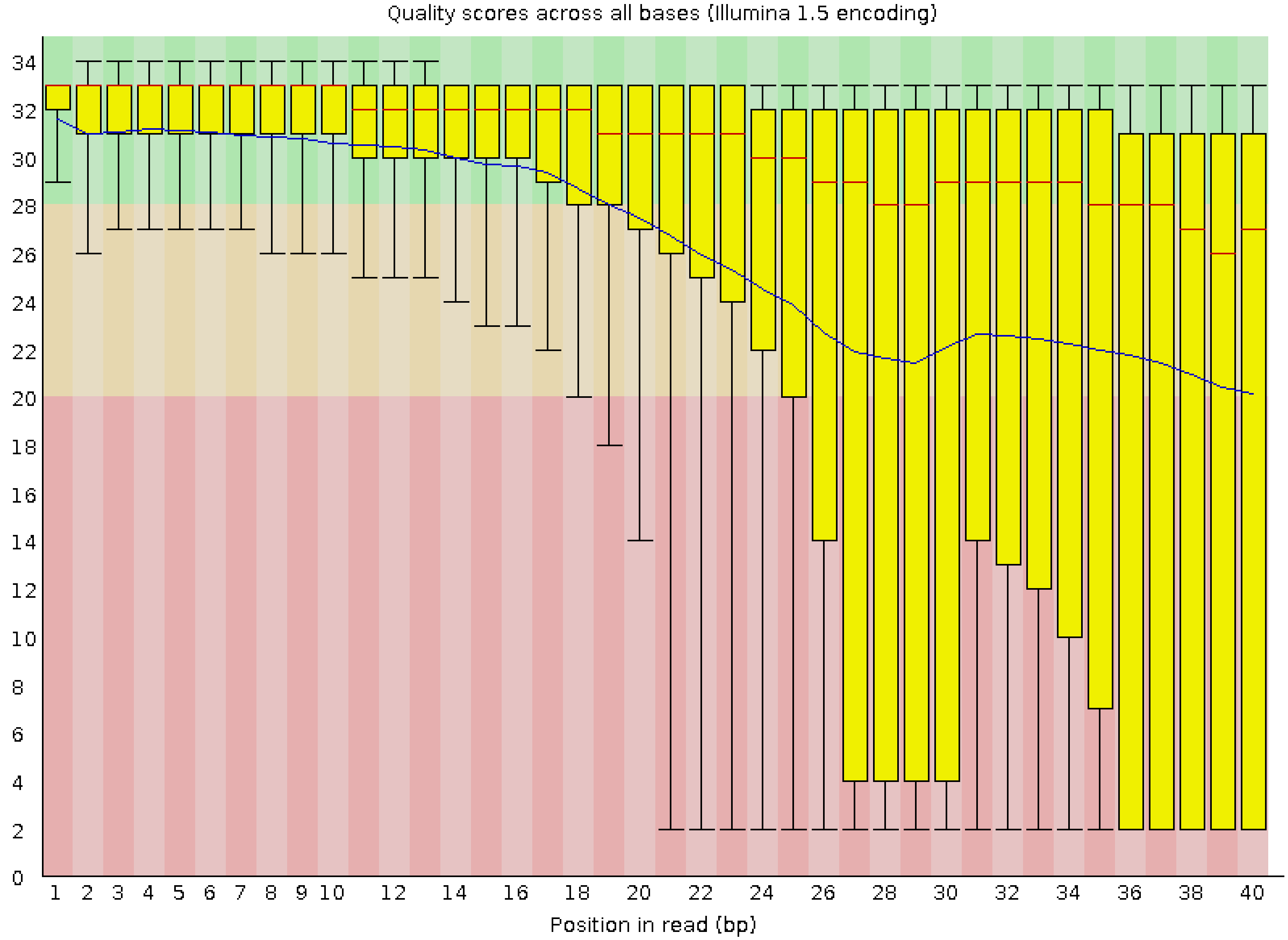

FastQC
File Help
bad_sequence.txt
good_sequence_short.txt
✓ Basic Statistics
✗ Per base sequence quality
✓ Per sequence quality scores
✓ Per base sequence content
! Per base GC content
! Per sequence GC content
✓ Per base N content
✓ Sequence Length Distribution
! Sequence Duplication Levels
! Overrepresented sequences
✗ Kmer Content
Quality scores across all bases (Illumina >v1.3 encoding)
34
32
30
28
26
24
22
20
18
16
14
12
10
8
6
4
2
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39
Position in read (bp)

Note: Depending on your experiment, it may give false alarms

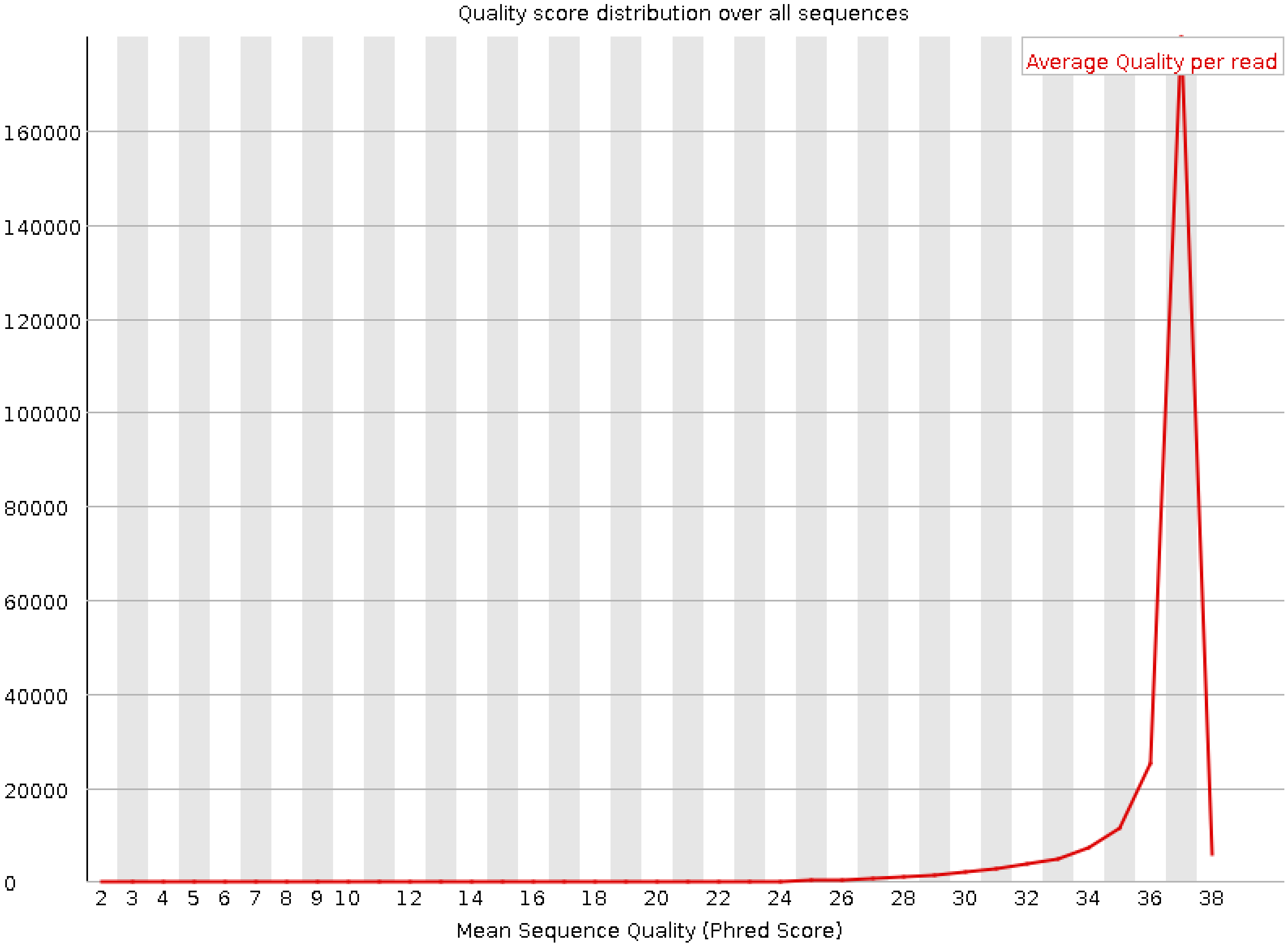
Per base sequence quality: good



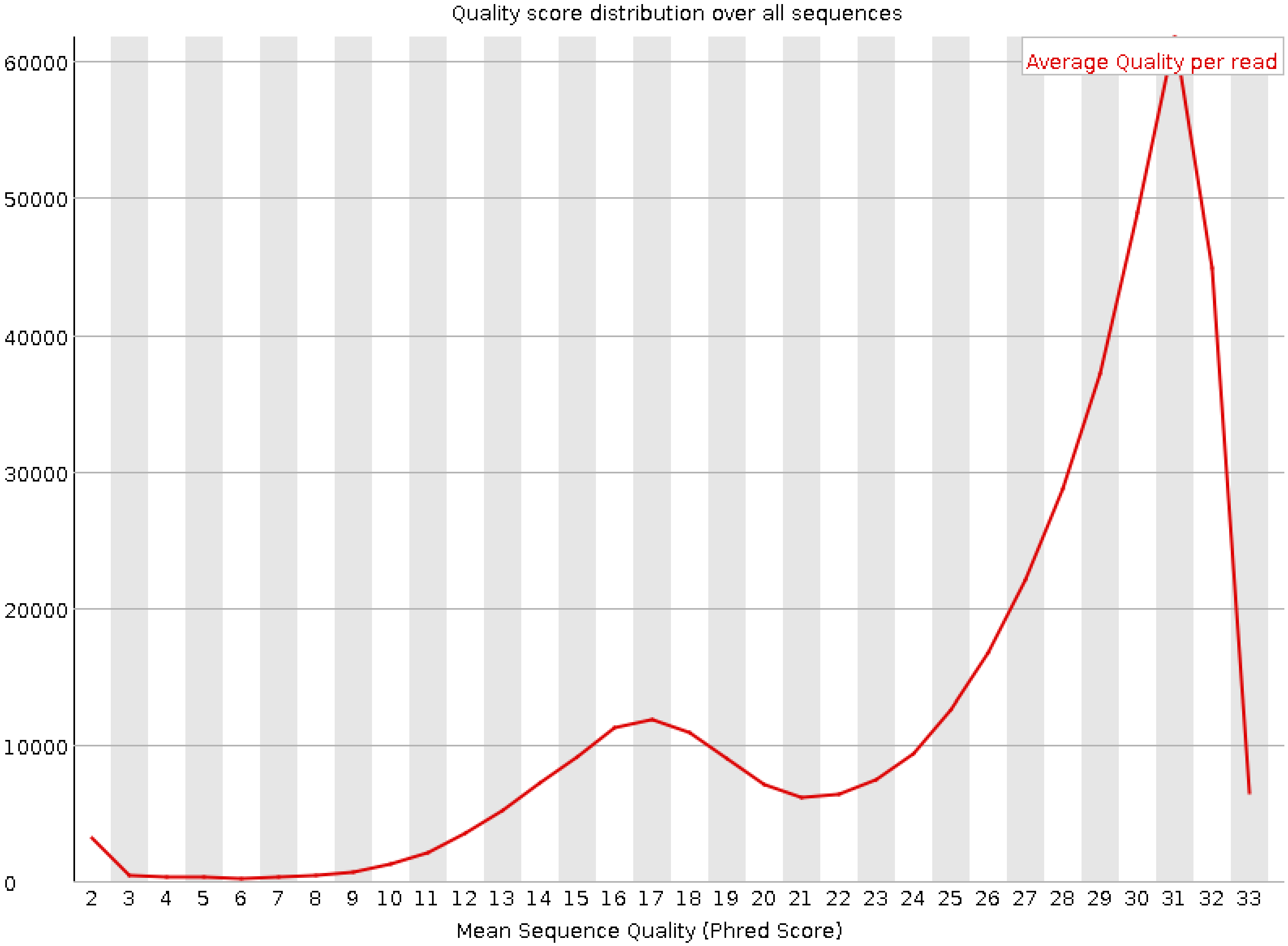
Per base sequence quality: bad



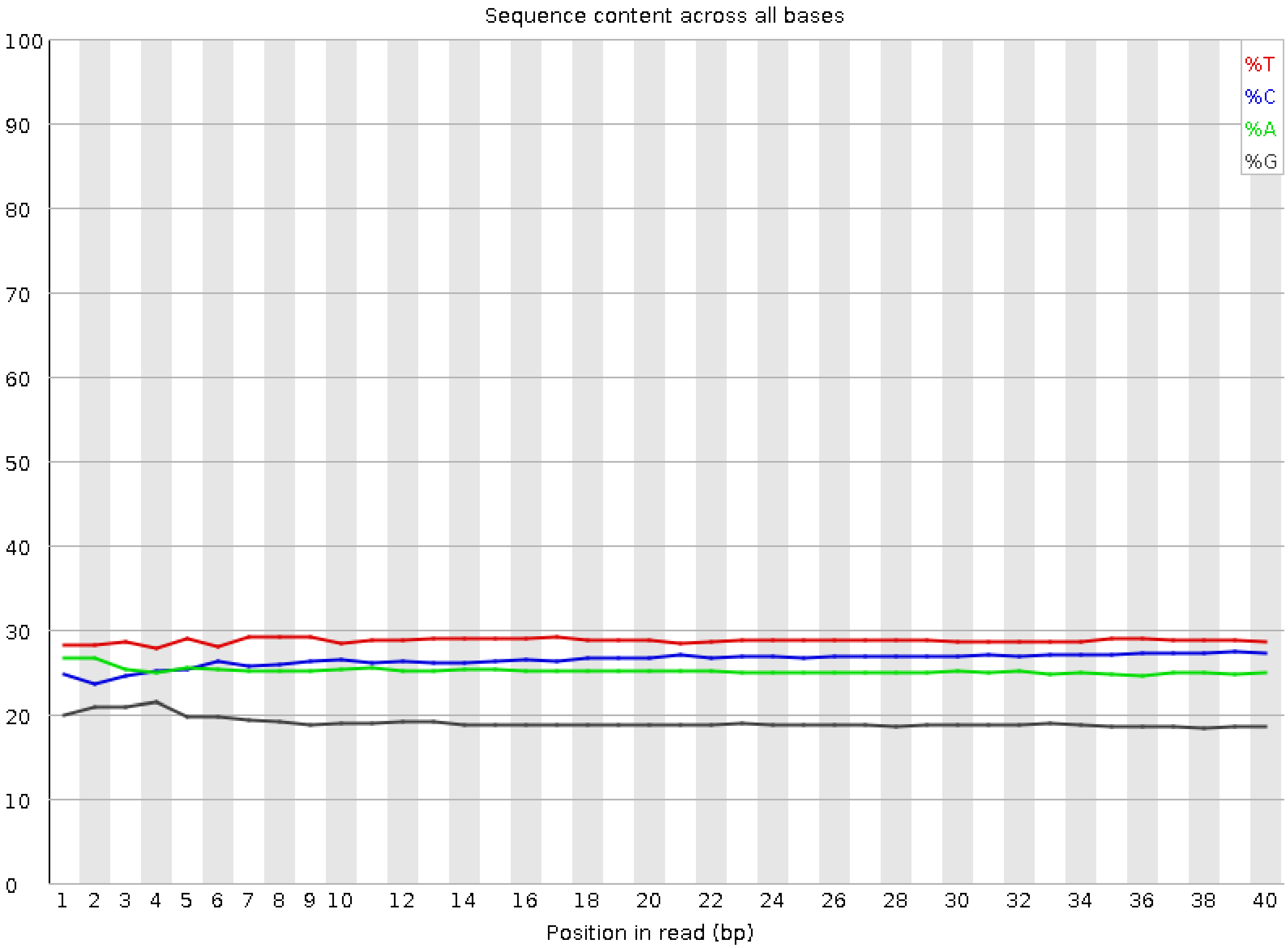
Per sequence quality scores: good



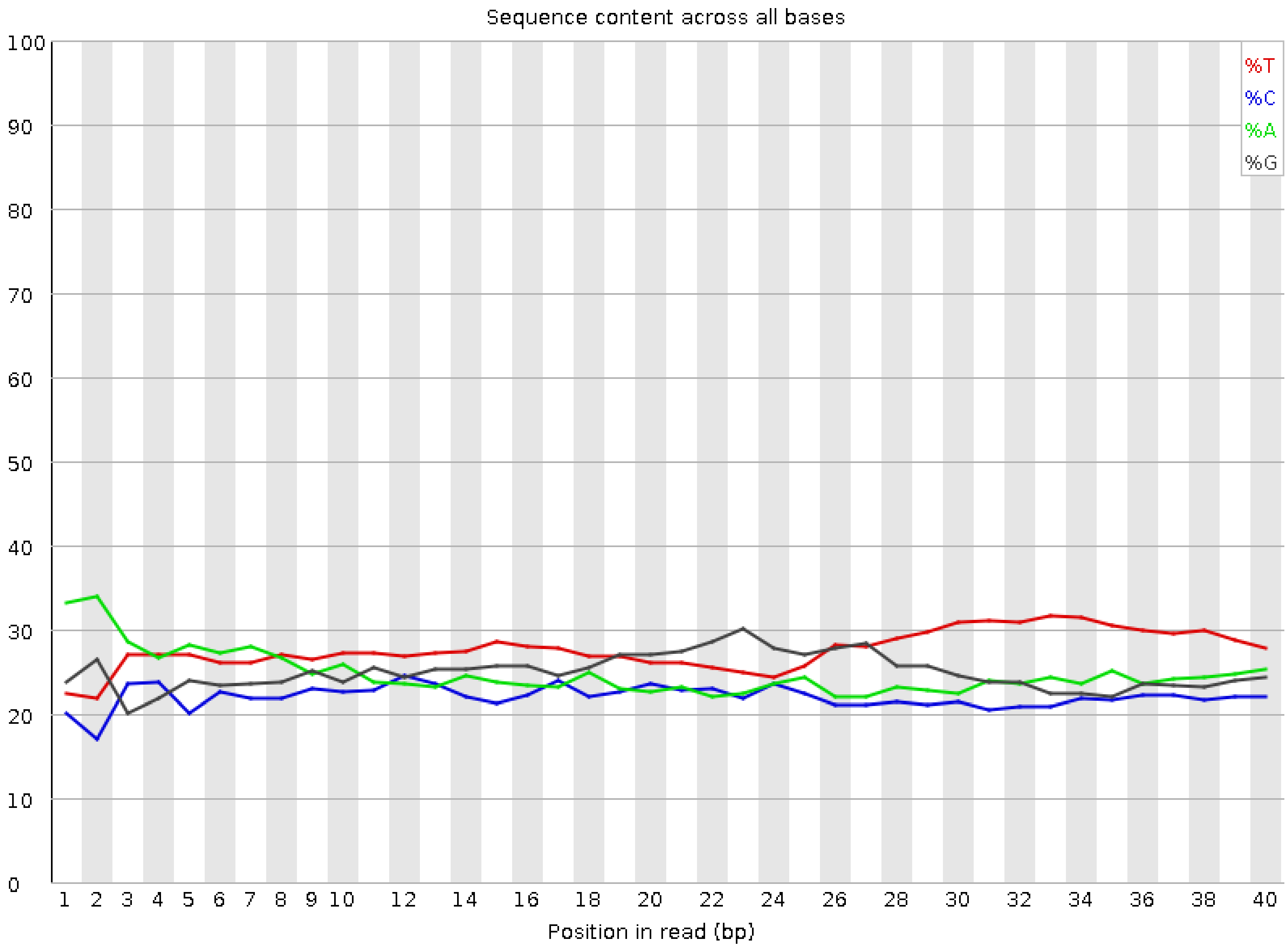
Per sequence quality scores: bad



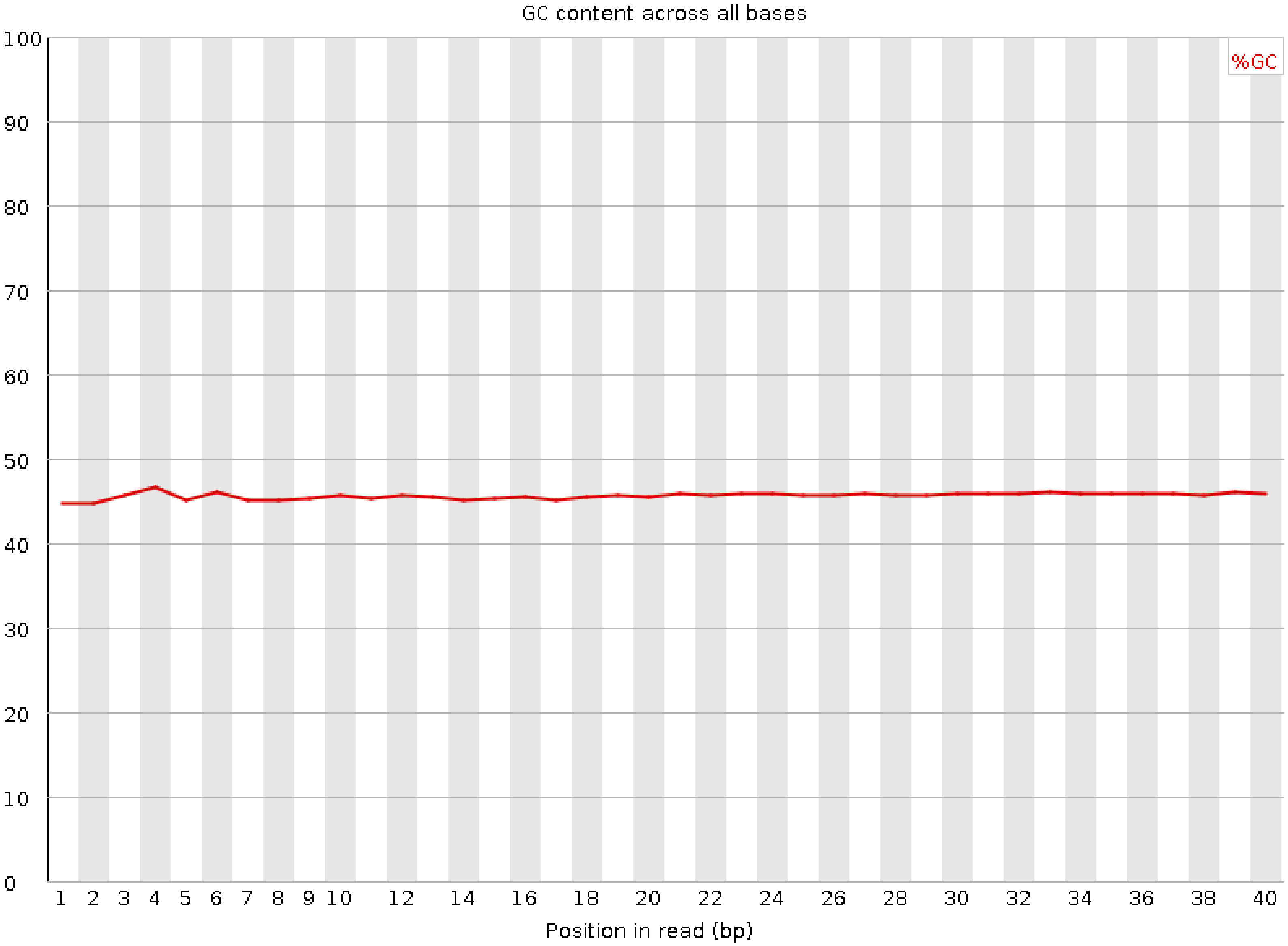
Per base sequence content: good



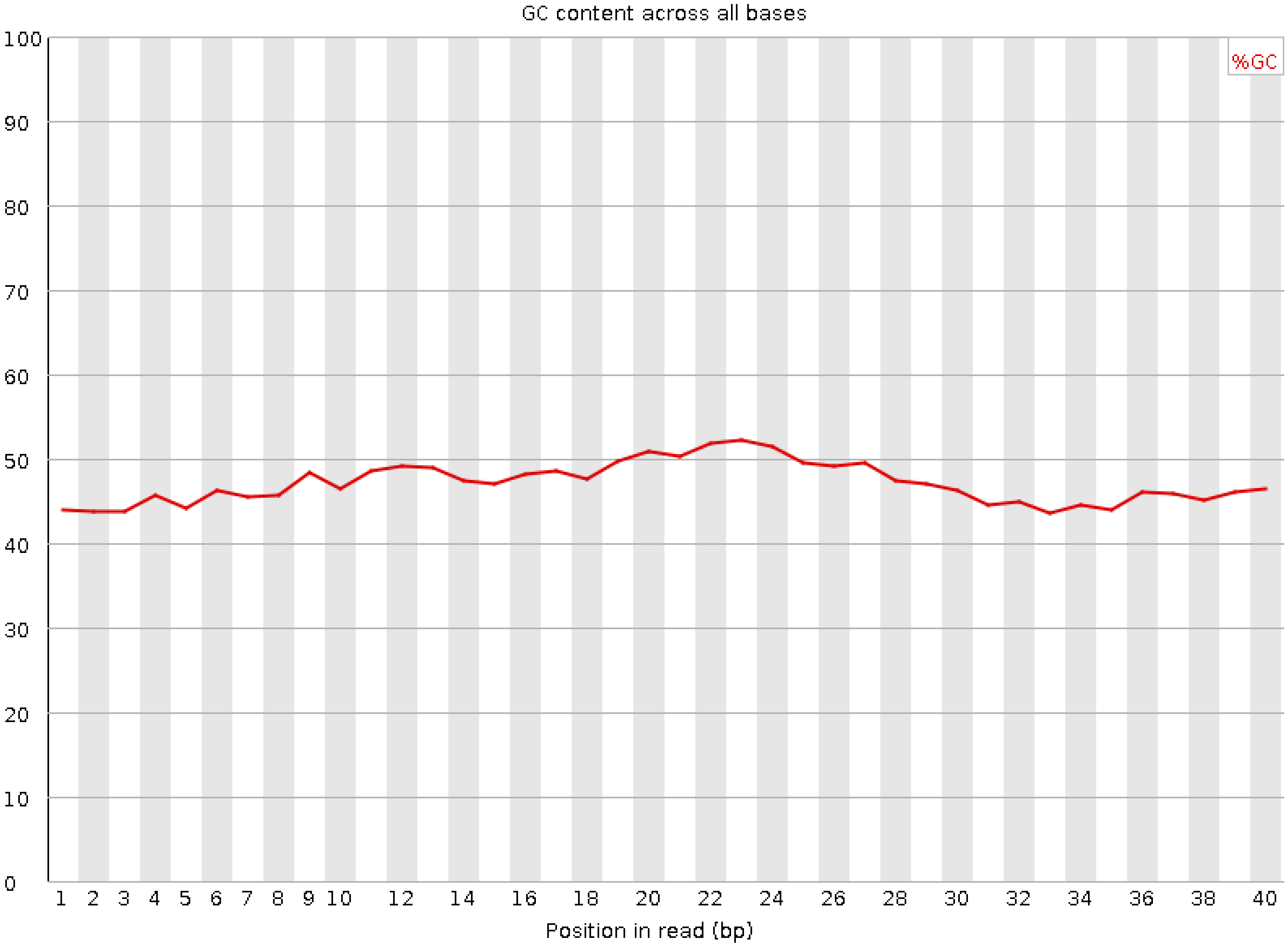
Per base sequence content: bad



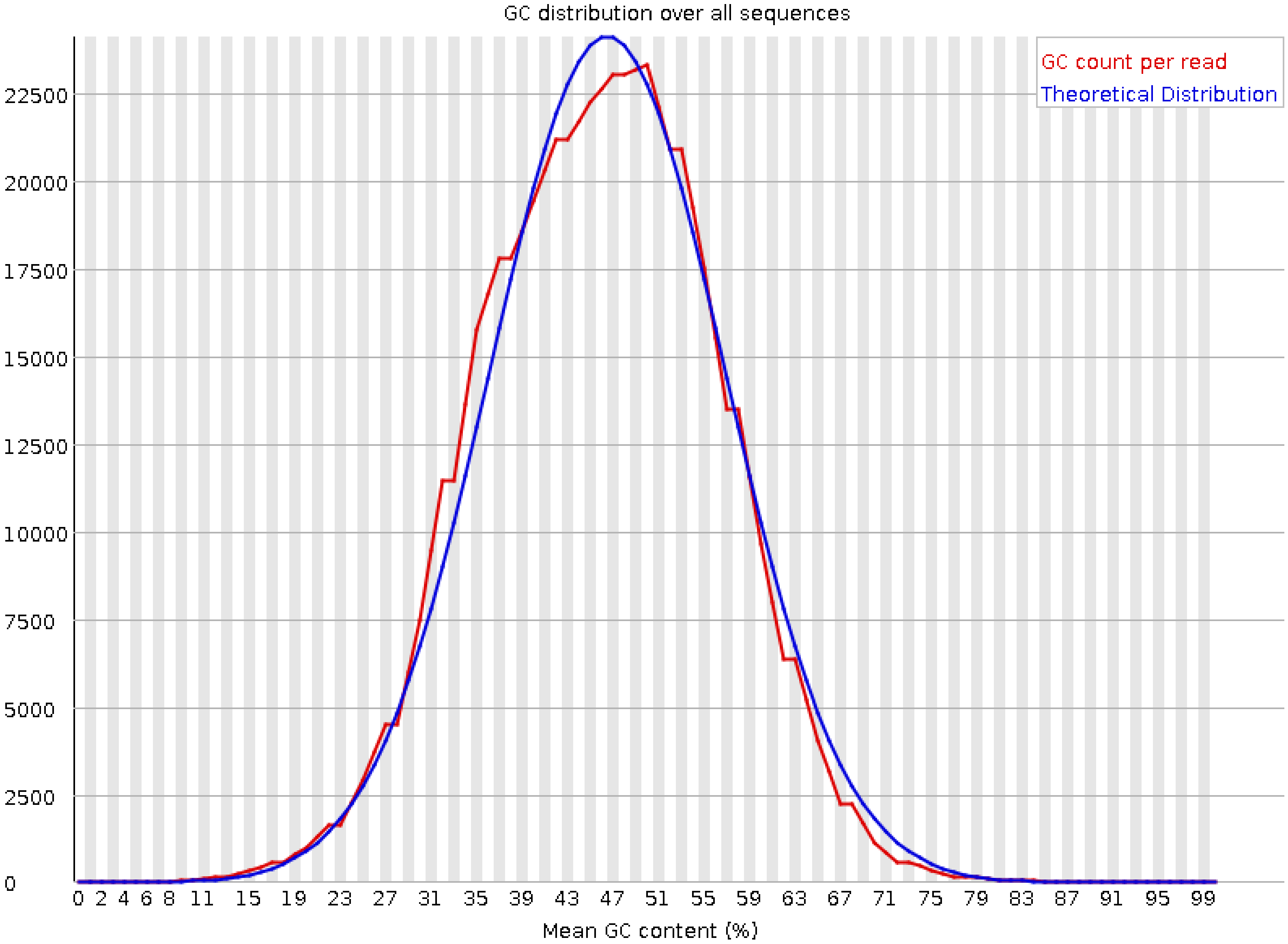
Per base GC content: good



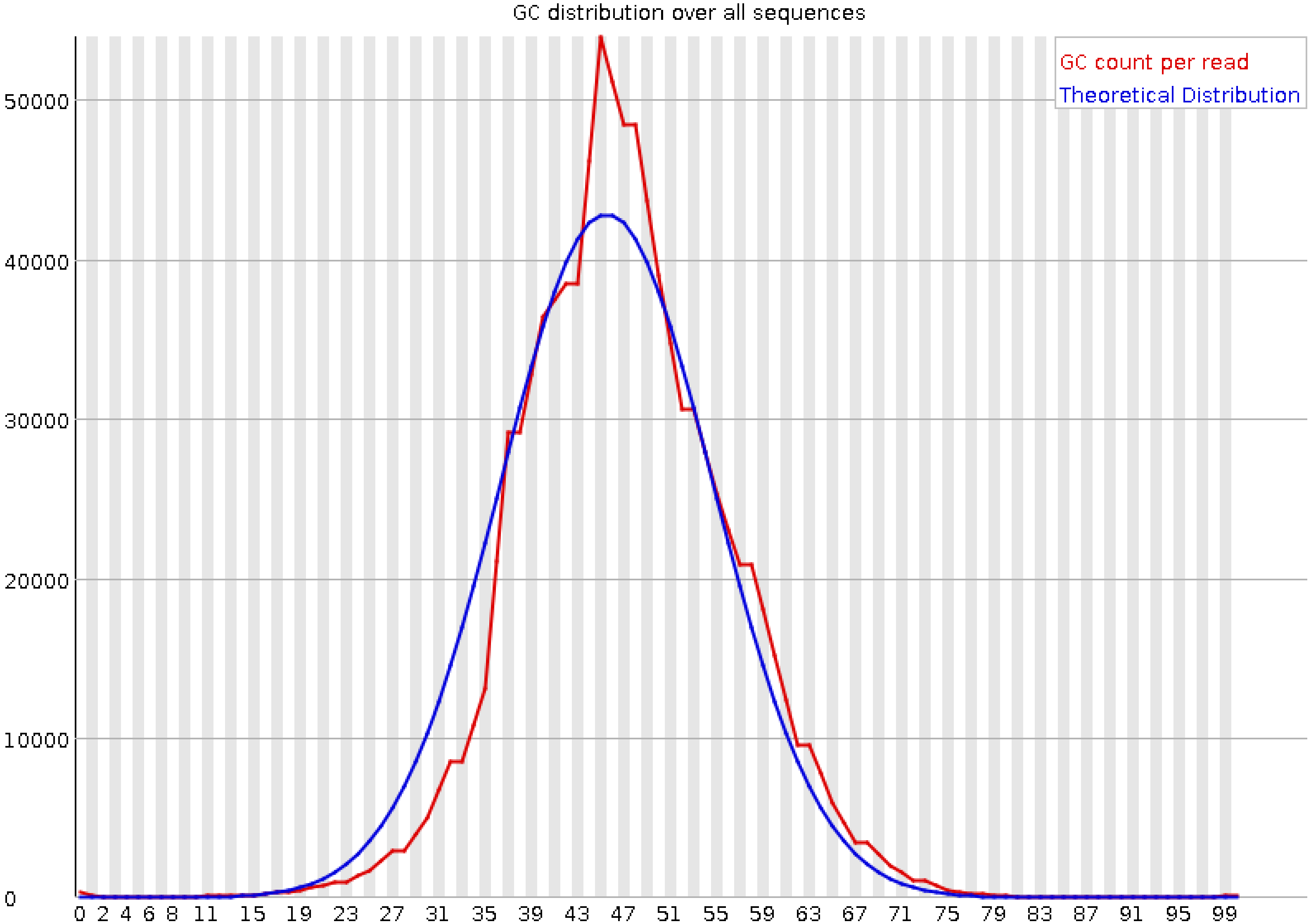
Per base GC content: bad



Per sequence GC content: good



Per sequence GC content: bad



I. File Formats
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

Sequence Alignment

- Alignment with exact match

```
ABCTUV
ABUV

ABCTUV
AB--UV
```

Sequence Alignment

- Inexact

  ATTCCATTCATAA
  ATTCTTAATAA

  ATTCCATTCATAA
  ATTC--TTAATAA

Sequence Alignment

- Global alignment

  CATGCTATGGTCA
  ATGCGGTCGACAC

- Local alignment

  CATGCTATGGTCA
   ATGC---GGTC

Pairwise Alignment Algorithms

- Exact (returns optimal solution)
  - Needleman – Wunsch (1970)
    - Global
  - Smith-Waterman (1981)
    - Local
- Heuristic (trade exactness for speed)
  - FASTA (1985)
  - BLAST (1990)
  - MUMmer (1999)
  - Vmatch
  - Many more!

Multiple Alignment Algorithms

- Optimal solution:
  - $O(\text{length}^{\text{num\_sequences}})$
- Approximations:
  - ClustalW, T-coffee
    - Progressively combining pairwise alignments
  - SAM, HMMER
    - Hidden Markov models
  - SAGA, RAGA
    - Genetic algorithms
  - MSASA
    - Simulated annealing

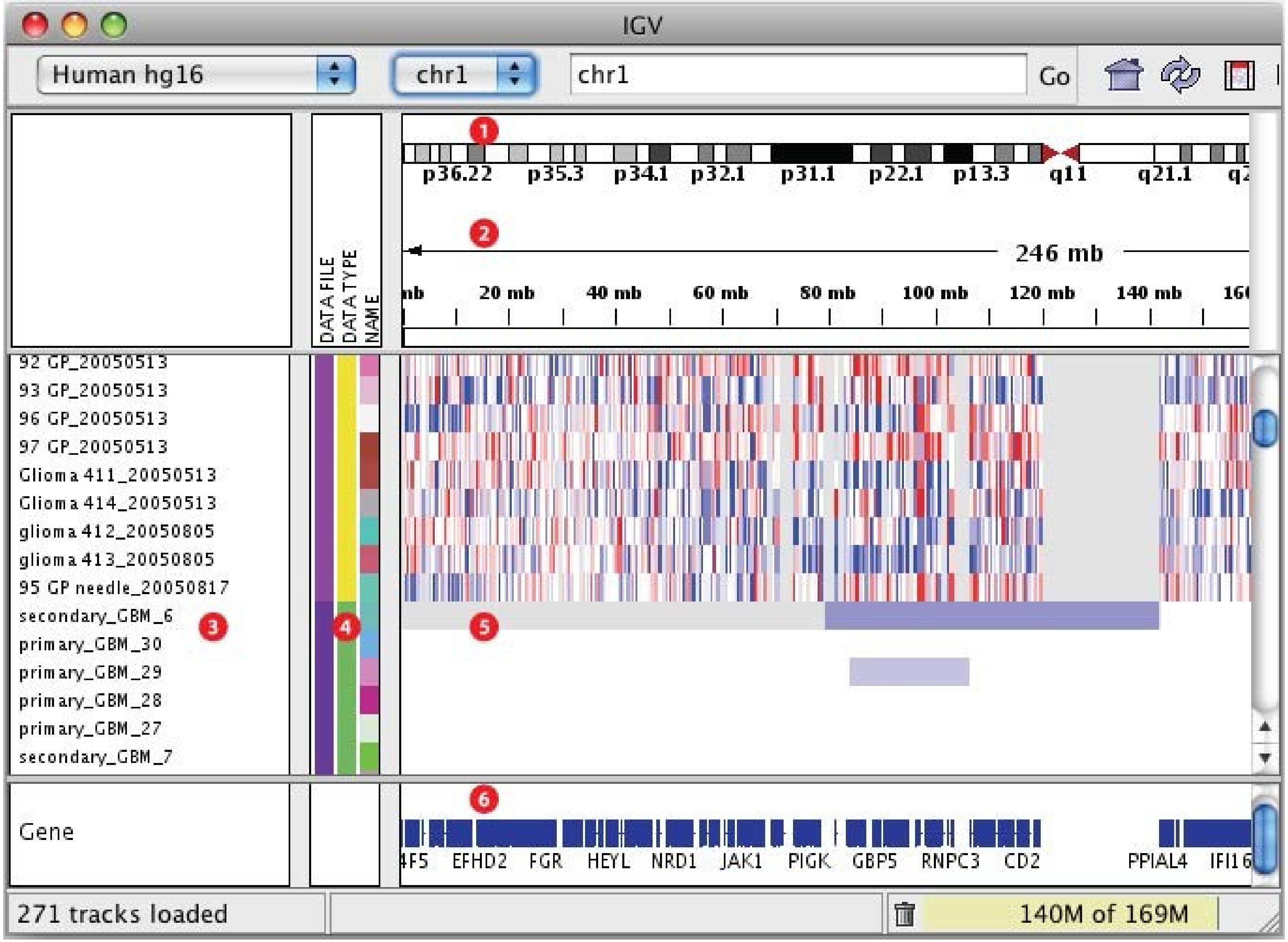
Short Read Alignment

- Some ways to improve performance:
    - Compress and index the genome
        - k-mer hashing
        - Suffix trees
        - Enhanced suffix arrays
        - Burrows-Wheeler transform (BWT)
    - Seed local alignments with exact matches

Short Read Alignment

- BFAST
  - Explicit time vs. accuracy tradeoff
  - Handles indels
- BWA
  - Handles indels
- Bowtie
  - Doesn't handle indels
- Bowtie2
  - Handles indels
  - Supports local alignment
  - Handles paired end sequencing better

IGV



IGV
Human hg16
chr1
chr1
Go
1
p36.22
p35.3
p34.1
p32.1
p31.1
p22.1
p13.3
q11
q21.1
q2
2
246 mb
mb
20 mb
40 mb
60 mb
80 mb
100 mb
120 mb
140 mb
16(
DATA FILE
DATA TYPE
NAME
92 GP_20050513
93 GP_20050513
96 GP_20050513
97 GP_20050513
Glioma 411_20050513
Glioma 414_20050513
glioma 412_20050805
glioma 413_20050805
95 GP needle_20050817
secondary_GBM_6
primary_GBM_30
primary_GBM_29
primary_GBM_28
primary_GBM_27
secondary_GBM_7
3
4
5
6
Gene
4F5
EFHD2
FGR
HEYL
NRD1
JAK1
PIGK
GBP5
RNPC3
CD2
PPIAL4
IFI16
271 tracks loaded
140M of 169M

I. File Formats
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

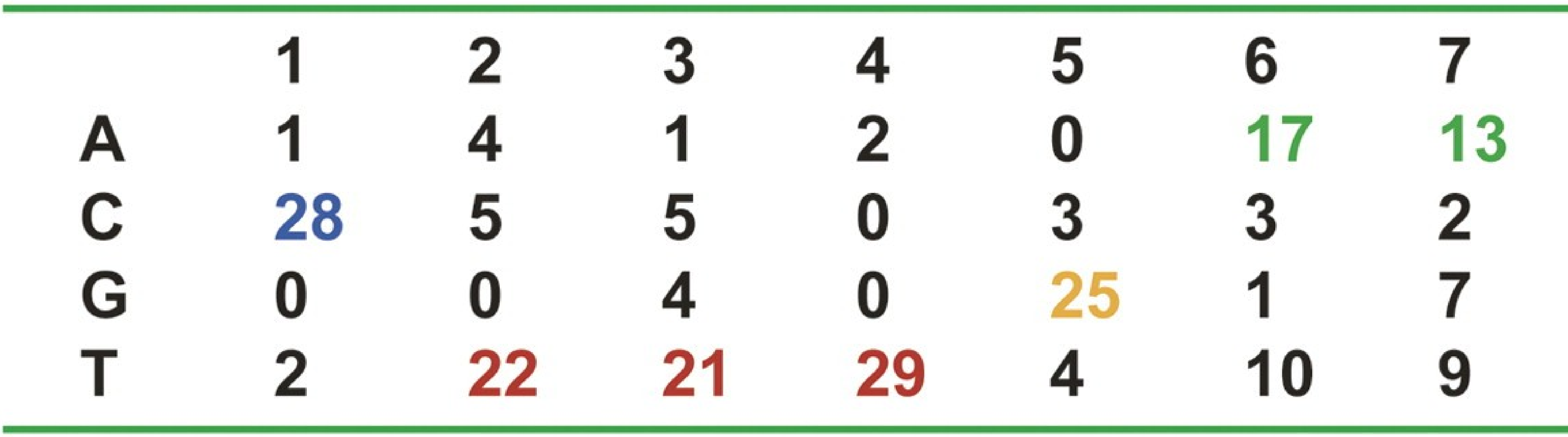
Motif discovery (*de novo*)

MEME

Motif discovery (*de novo*)

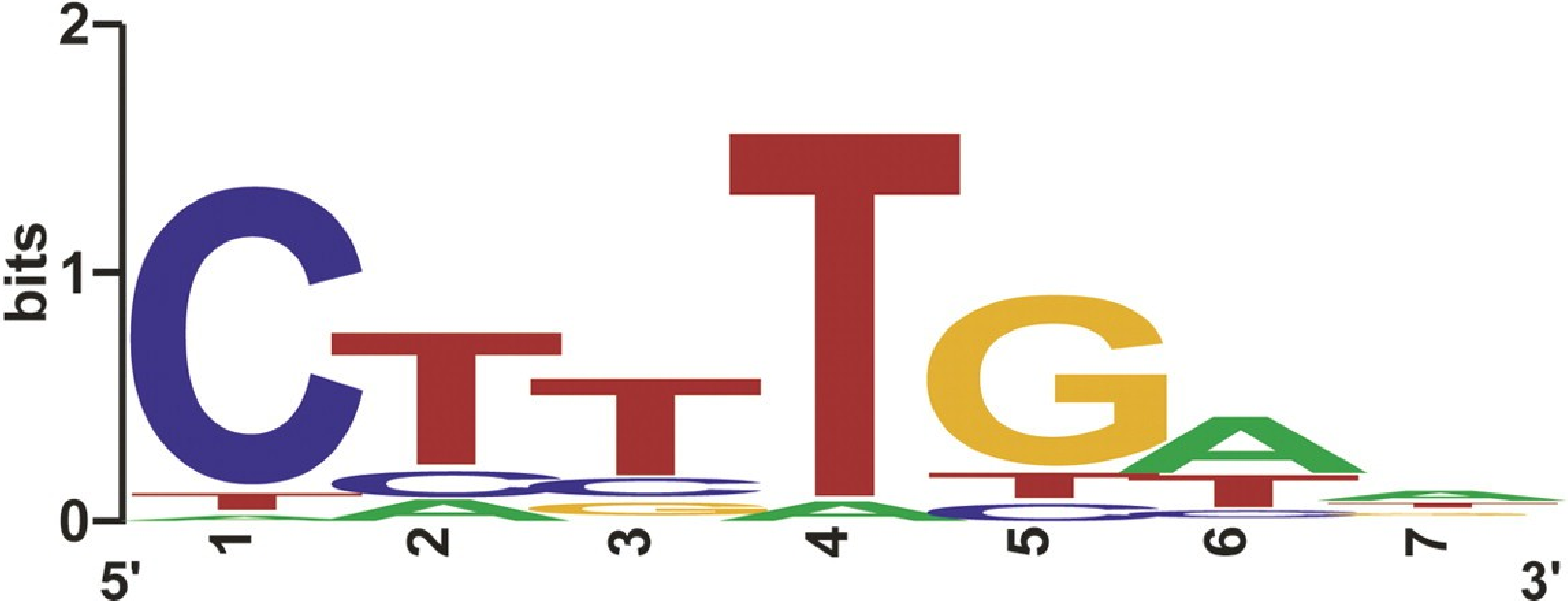
- HOMER
  - **H**ypergeometric **O**ptimization of **M**otif **E**n**R**ichment
  - Motif analysis & more!
  - Chris Brenner (Glass lab)

A



	1	2	3	4	5	6	7
A	1	4	1	2	0	17	13
C	28	5	5	0	3	3	2
G	0	0	4	0	25	1	7
T	2	22	21	29	4	10	9

B



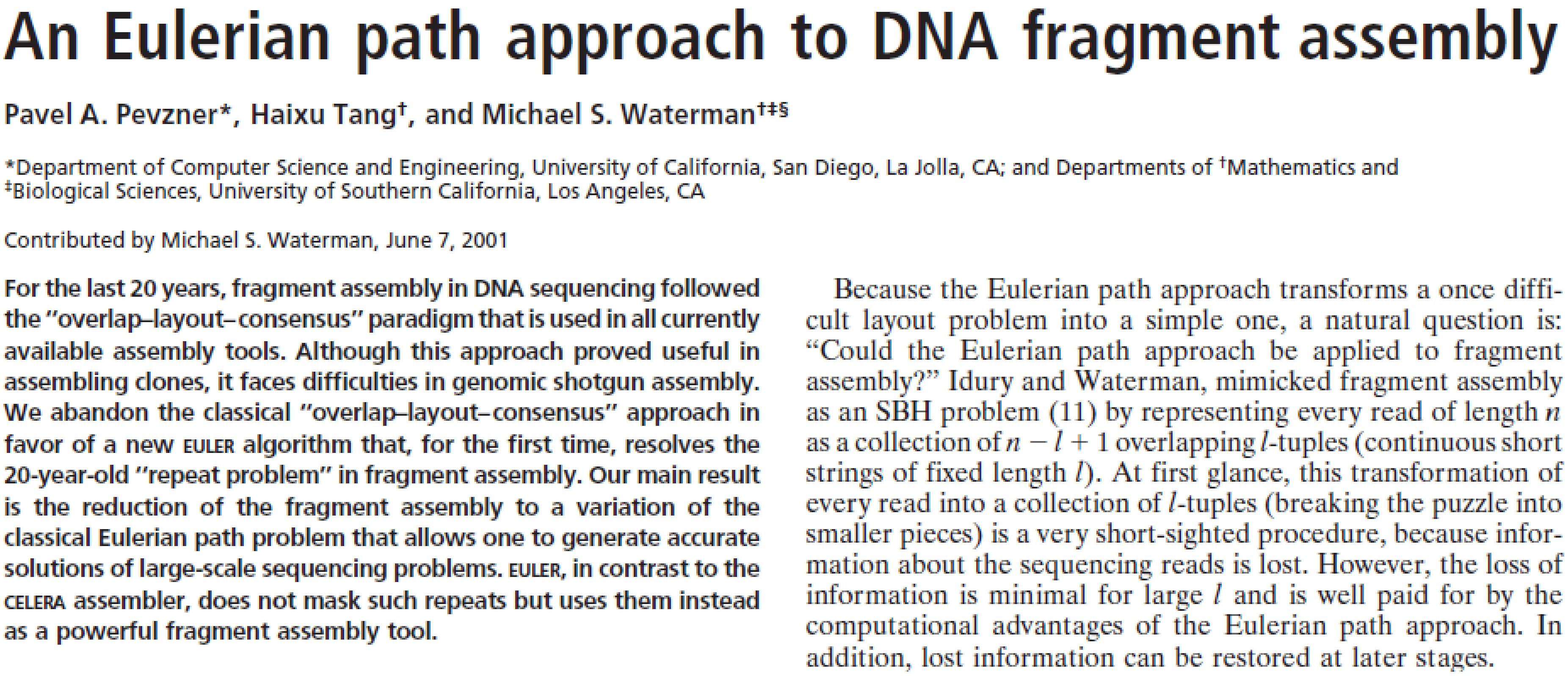
I. File Formats
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

De novo genome assembly

- Traditionally done with Sanger sequencing
  - Long reads ~1000s of bp
  - Tools (overlap-layout-consensus)
    - Atlas
    - ARACHNE
    - Celera
    - PCAP
    - *phrap*
    - Phusion

De novo genome assembly

- Overlap-layout-consensus approach
  - Does OK with small numbers of long reads
  - poorly suited to millions of short NGS reads
    - 10s to 100s of bp
  - Doesn't handle repeats well – so generally just ignores them



An Eulerian path approach to DNA fragment assembly

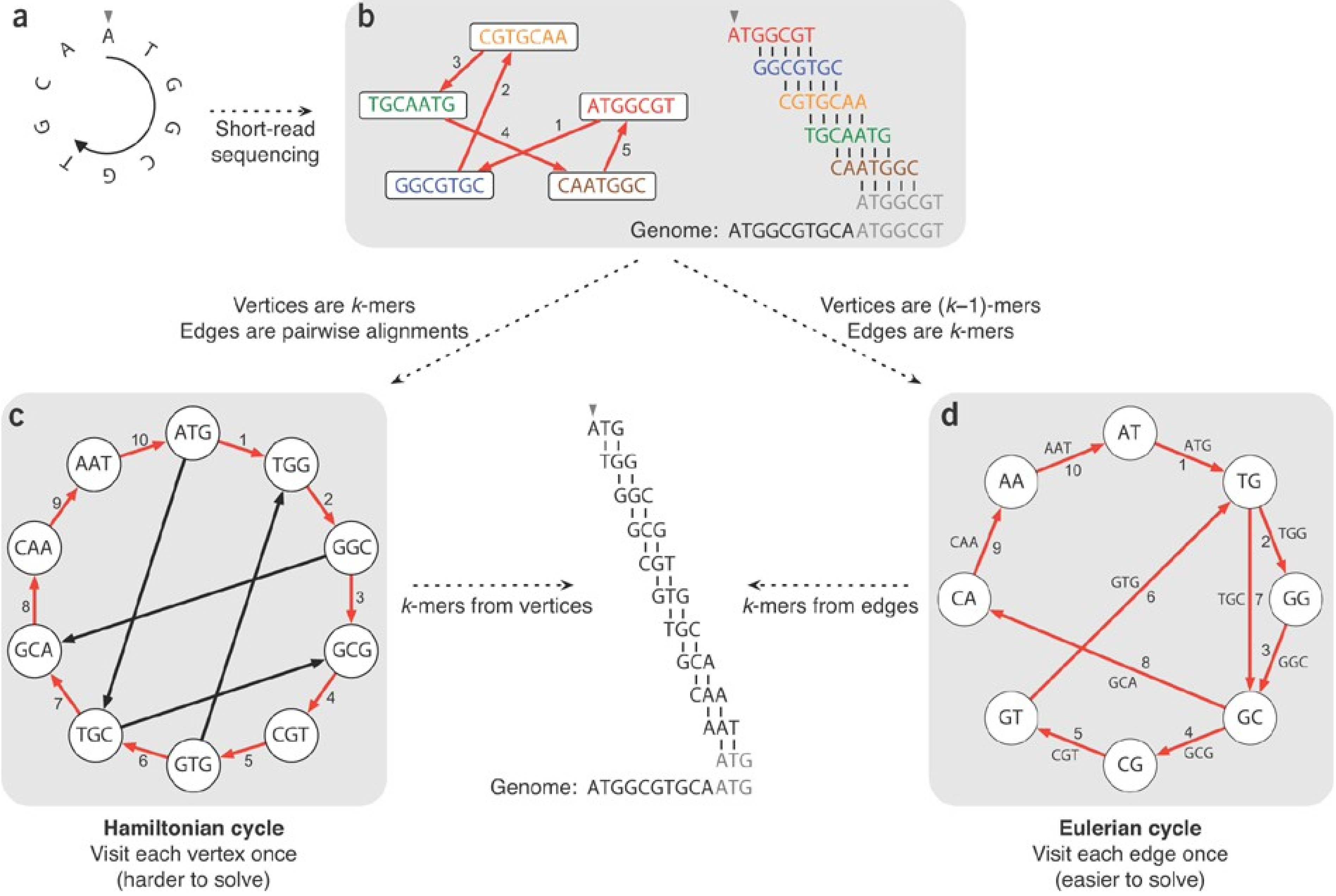
Pavel A. Pevzner*, Haixu Tang†, and Michael S. Waterman†‡§

*Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA; and Departments of †Mathematics and ‡Biological Sciences, University of Southern California, Los Angeles, CA

Contributed by Michael S. Waterman, June 7, 2001

For the last 20 years, fragment assembly in DNA sequencing followed the ''overlap–layout–consensus'' paradigm that is used in all currently available assembly tools. Although this approach proved useful in assembling clones, it faces difficulties in genomic shotgun assembly. We abandon the classical ''overlap–layout–consensus'' approach in favor of a new EULER algorithm that, for the first time, resolves the 20-year-old ''repeat problem'' in fragment assembly. Our main result is the reduction of the fragment assembly to a variation of the classical Eulerian path problem that allows one to generate accurate solutions of large-scale sequencing problems. EULER, in contrast to the CELERA assembler, does not mask such repeats but uses them instead as a powerful fragment assembly tool.

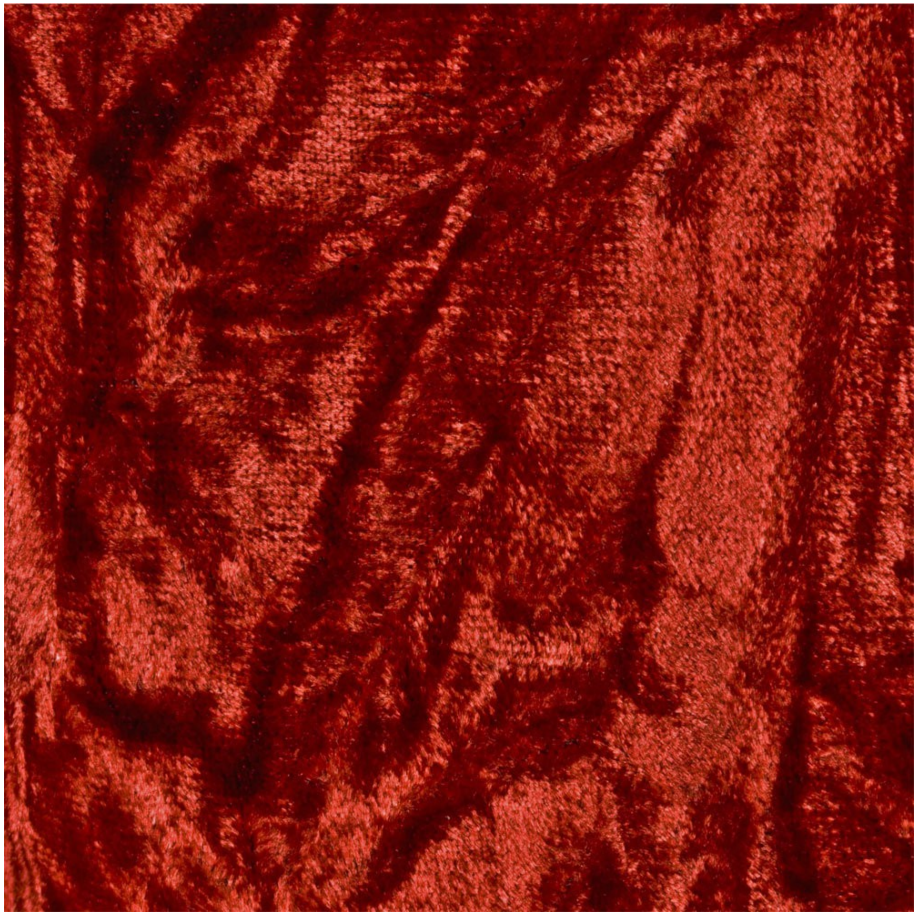
Because the Eulerian path approach transforms a once difficult layout problem into a simple one, a natural question is: "Could the Eulerian path approach be applied to fragment assembly?" Idury and Waterman, mimicked fragment assembly as an SBH problem (11) by representing every read of length n as a collection of $n - l + 1$ overlapping l-tuples (continuous short strings of fixed length l). At first glance, this transformation of every read into a collection of l-tuples (breaking the puzzle into smaller pieces) is a very short-sighted procedure, because information about the sequencing reads is lost. However, the loss of information is minimal for large l and is well paid for by the computational advantages of the Eulerian path approach. In addition, lost information can be restored at later stages.

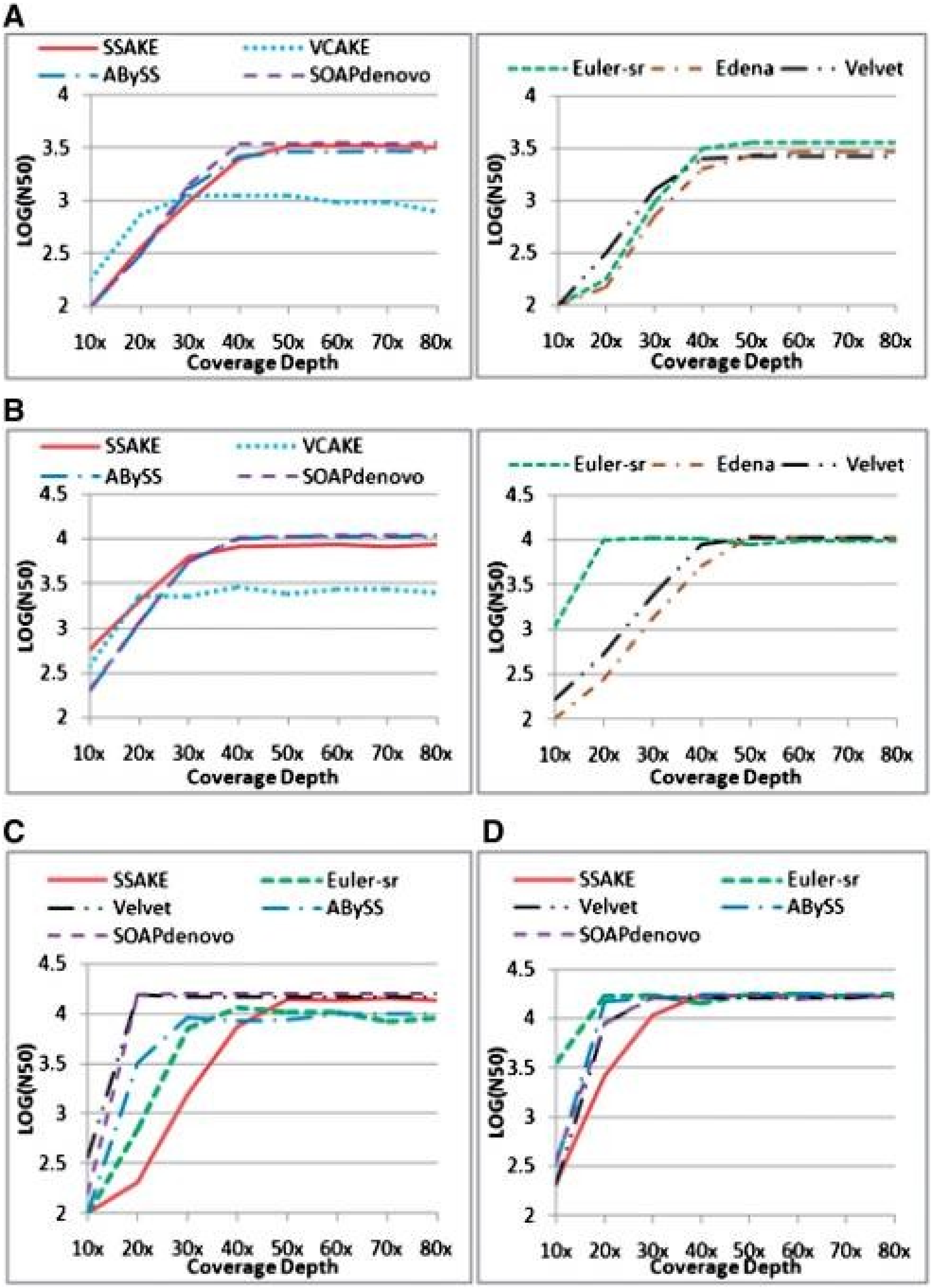


Compeau, P.E.C., Pevzner, P.A., and Tesler, G. (2011). How to apply de Bruijn graphs to genome assembly. Nature Biotechnology 29, 987–991.

Velvet

- velveth
  - Creates hashtable for velvetg
  - May want to play with k-mer length (hash length)
- velvetg
  - Assembles using de Bruijn graphs
- VelvetOptimizer
  - optimizes parameters



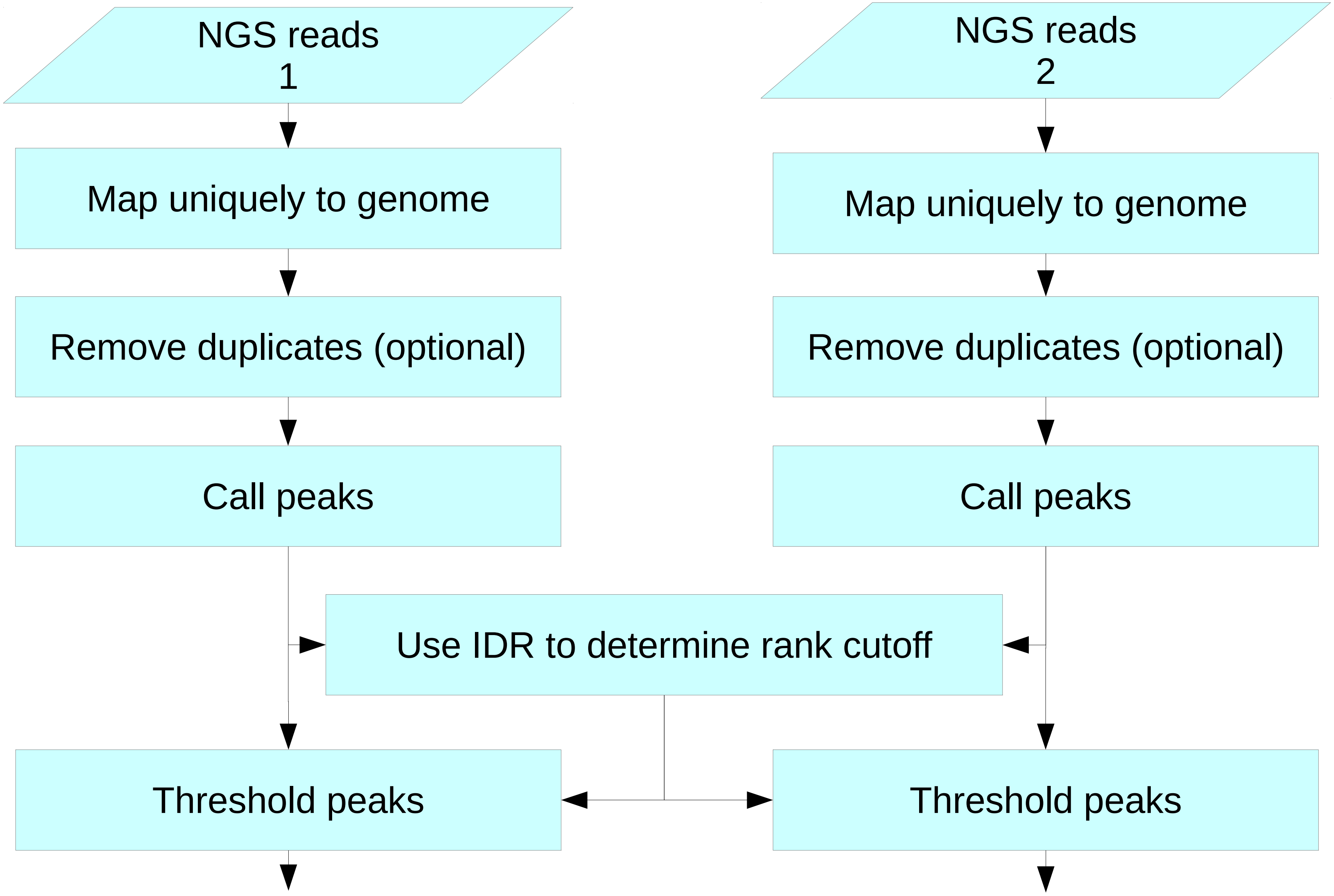


Comparison of the effect of various coverage depths on N50 length in T.bru assembly when BCER was 0.6%. (A) Single-end reads assembly, read length (RL)=35 bp; (B) single-end assembly, RL=75 bp; (C) paired-end reads assembly, RL=35 bp; (D) paired-end assembly, RL=75 bp.

Lin, Y., Li, J., Shen, H., Zhang, L., Papasian, C.J., and Deng, H.-W. (2011). Comparative studies of de novo assembly tools for next-generation sequencing technologies. Bioinformatics 27, 2031–2037.

I. File Formats
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

ChIP-seq data analysis



NGS reads 1

Map uniquely to genome

Remove duplicates (optional)

Call peaks

NGS reads 2

Map uniquely to genome

Remove duplicates (optional)

Call peaks

Use IDR to determine rank cutoff

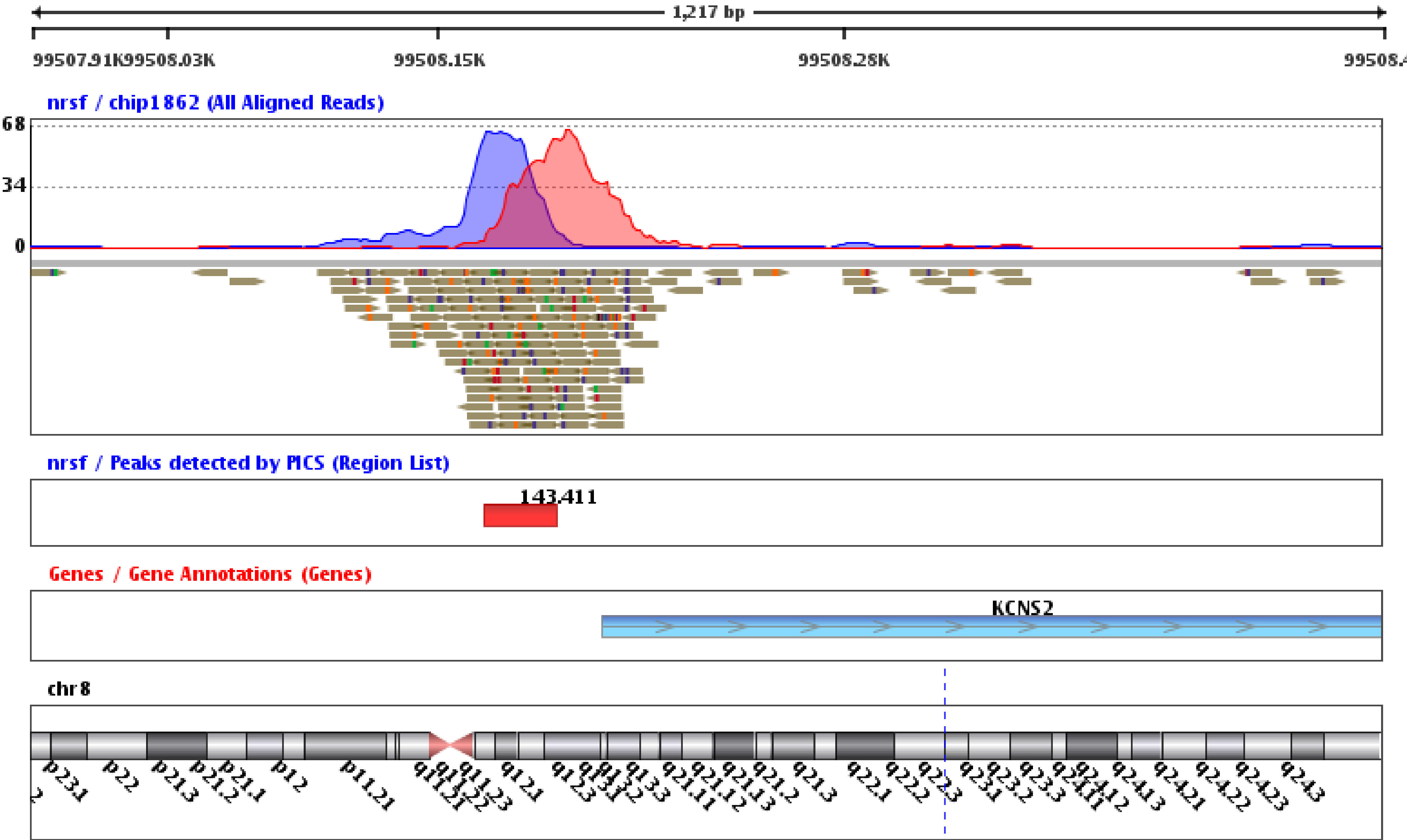
Threshold peaks

Threshold peaks

Duplicate removal

- PCR amplification step produces multiple copies of same fragment
  - sequence composition, size, etc. biases distribution
  - shows up as multiple reads with same 5' position
- However, expect some proportion of “real” duplicate reads based on coverage
- Benefits of removal for various applications debated but most favor removal

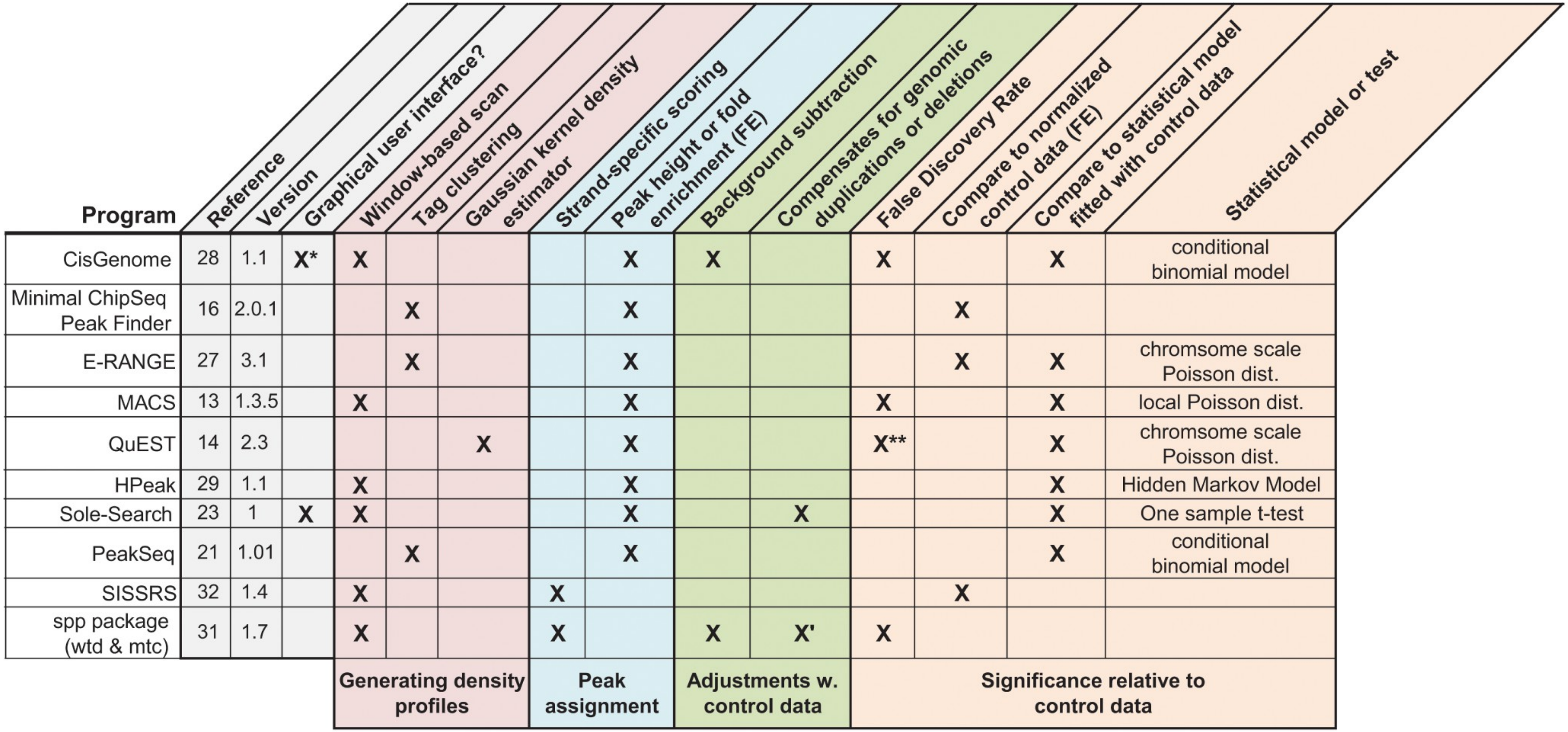
Peak Calling



Peak Calling

- Large (40+) and expanding panoply of algorithms
- “It is challenging to rigorously assess subtle improvements due to the scarcity and unreliability of verified binding sites for any ChIP-seq dataset.”
  - Wilbanks and Facciotti (2010)

ChIP-seq peak calling programs selected for evaluation in Wilbanks and Facciotti (2010)



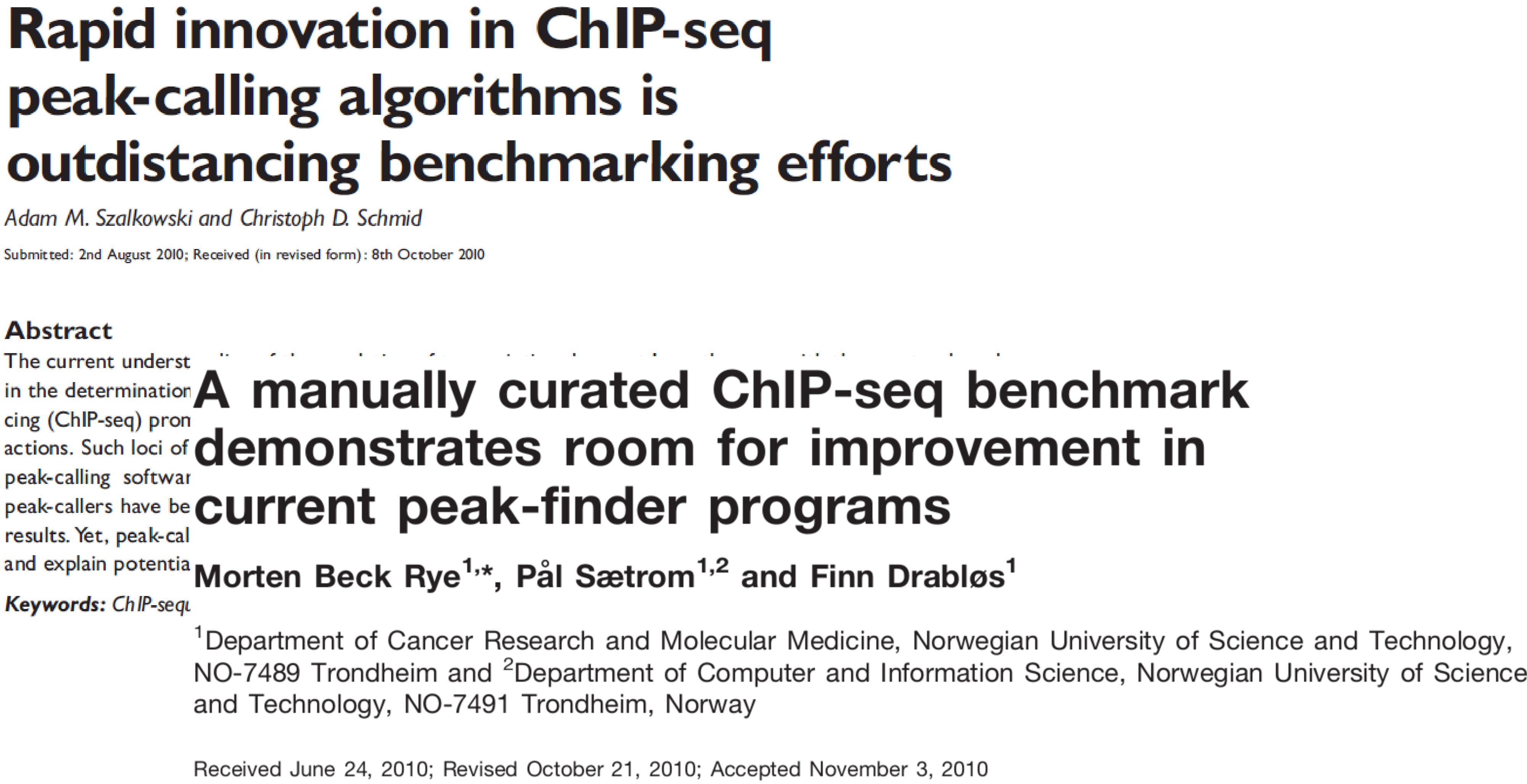
Program	Reference	Version	Graphical user interface?	Window-based scan	Tag clustering	Gaussian kernel density estimator	Strand-specific scoring	Peak height or fold enrichment (FE)	Background subtraction	Compensates for genomic duplications or deletions	False Discovery Rate	Compare to normalized control data (FE)	Compare to statistical model fitted with control data	Statistical model or test
CisGenome	28	1.1	X*	X				X	X		X		X	conditional binomial model
Minimal ChipSeq Peak Finder	16	2.0.1			X			X				X		
E-RANGE	27	3.1			X			X				X	X	chromsome scale Poisson dist.
MACS	13	1.3.5		X				X			X		X	local Poisson dist.
QuEST	14	2.3				X		X			X**		X	chromsome scale Poisson dist.
HPeak	29	1.1		X				X					X	Hidden Markov Model
Sole-Search	23	1	X	X				X		X			X	One sample t-test
PeakSeq	21	1.01			X			X					X	conditional binomial model
SISSRS	32	1.4		X			X					X		
spp package (wtd & mtc)	31	1.7		X			X		X	X'	X			
				Generating density profiles			Peak assignment		Adjustments w. control data		Significance relative to control data			

X* = Windows-only GUI or cross-platform command line interface

X** = optional if sufficient data is available to split control data

X' = method exludes putative duplicated regions, no treatment of deletions

Wilbanks EG, Facciotti MT (2010) Evaluation of Algorithm Performance in ChIP-Seq Peak Detection. PLoS ONE 5(7): e11471. doi:10.1371/journal.pone.0011471
http://www.plosone.org/article/info:doi/10.1371/journal.pone.0011471



Rapid innovation in ChIP-seq peak-calling algorithms is outdistancing benchmarking efforts

Adam M. Szalkowski and Christoph D. Schmid

Submitted: 2nd August 2010; Received (in revised form): 8th October 2010

Abstract

The current underst
in the determination
cing (ChIP-seq) pron
actions. Such loci of
peak-calling softwar
peak-callers have be
results. Yet, peak-cal
and explain potentia

Keywords: *ChIP-sequ*

A manually curated ChIP-seq benchmark demonstrates room for improvement in current peak-finder programs

Morten Beck Rye[1,*], Pål Sætrom[1,2] and Finn Drabløs[1]

[1]Department of Cancer Research and Molecular Medicine, Norwegian University of Science and Technology, NO-7489 Trondheim and [2]Department of Computer and Information Science, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway

Received June 24, 2010; Revised October 21, 2010; Accepted November 3, 2010

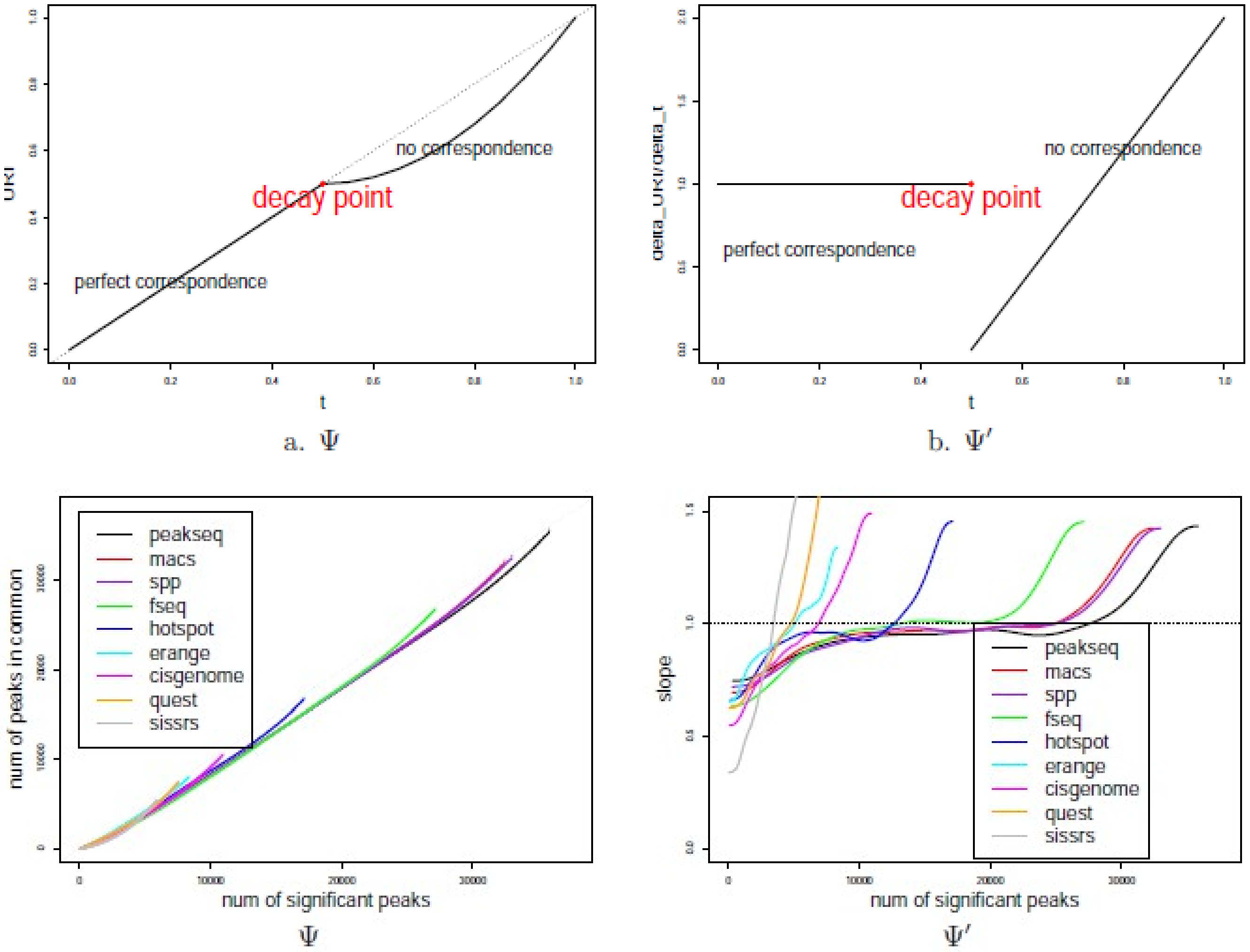
The MACS methodology initially identifies a set of high-confidence peaks which exhibit between 20-100 fold enrichment in reads and then uses these to model the average distance between reads on the two DNA strands.

It then uses a binomial distribution to remove duplicate tags which are represented at statistically unlikely levels within the sample.

Next it generates a full set of candidate peaks by comparing the enrichment within windows to a Poisson distribution generated from the control (input DNA or mock IP)

Irreproducible discovery rate (IDR)

Li, Q., Brown, J.B., Huang, H., and Bickel, P.J. (2011). Measuring reproducibility of high-throughput experiments. The Annals of Applied Statistics 5, 1752–1779.



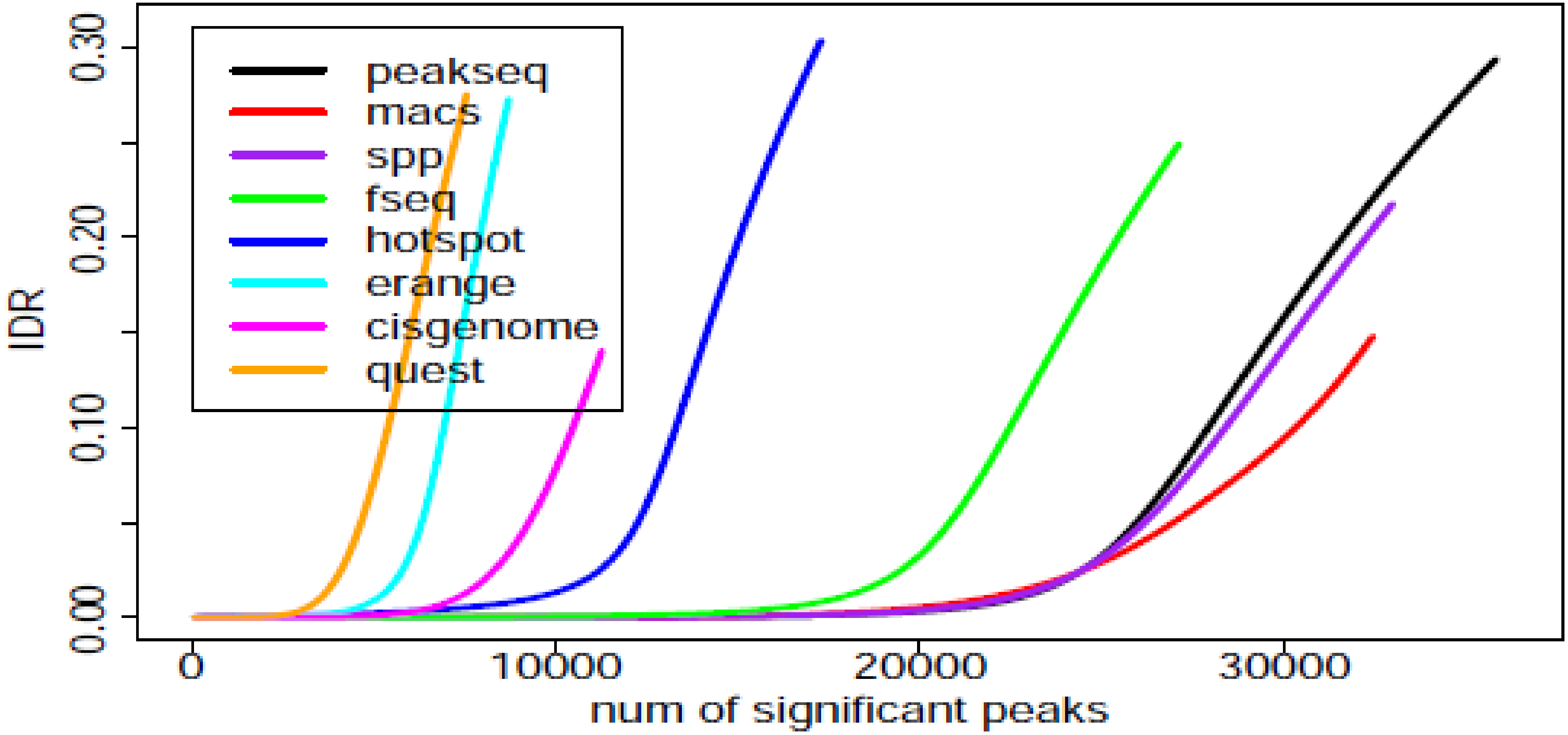
a. Ψ

b. Ψ'

Ψ

Ψ'

Irreproducible discovery rate (IDR)



I. File Formats
II. Quality Control
III. Sequence Alignment
IV. Motif Discovery
V. De Novo Genome Assembly
VI. ChIP-seq
VII. RNA-seq

RNA-seq

- Cufflinks
  - Analyze transcriptomes (at least 2 conditions) for differential expression & significance

RNA-seq

- CummeRbund
  - R package
  - Visualizing results from cuffdiff
  - Create SQLite database of results & relationships

RNA-seq

- TopHat
  - Uses BowTie to map reads to genome
  - Analyzes mapping for evidence of splice junctions
  - Huge diversity of splice isoforms in higher eukaryotes