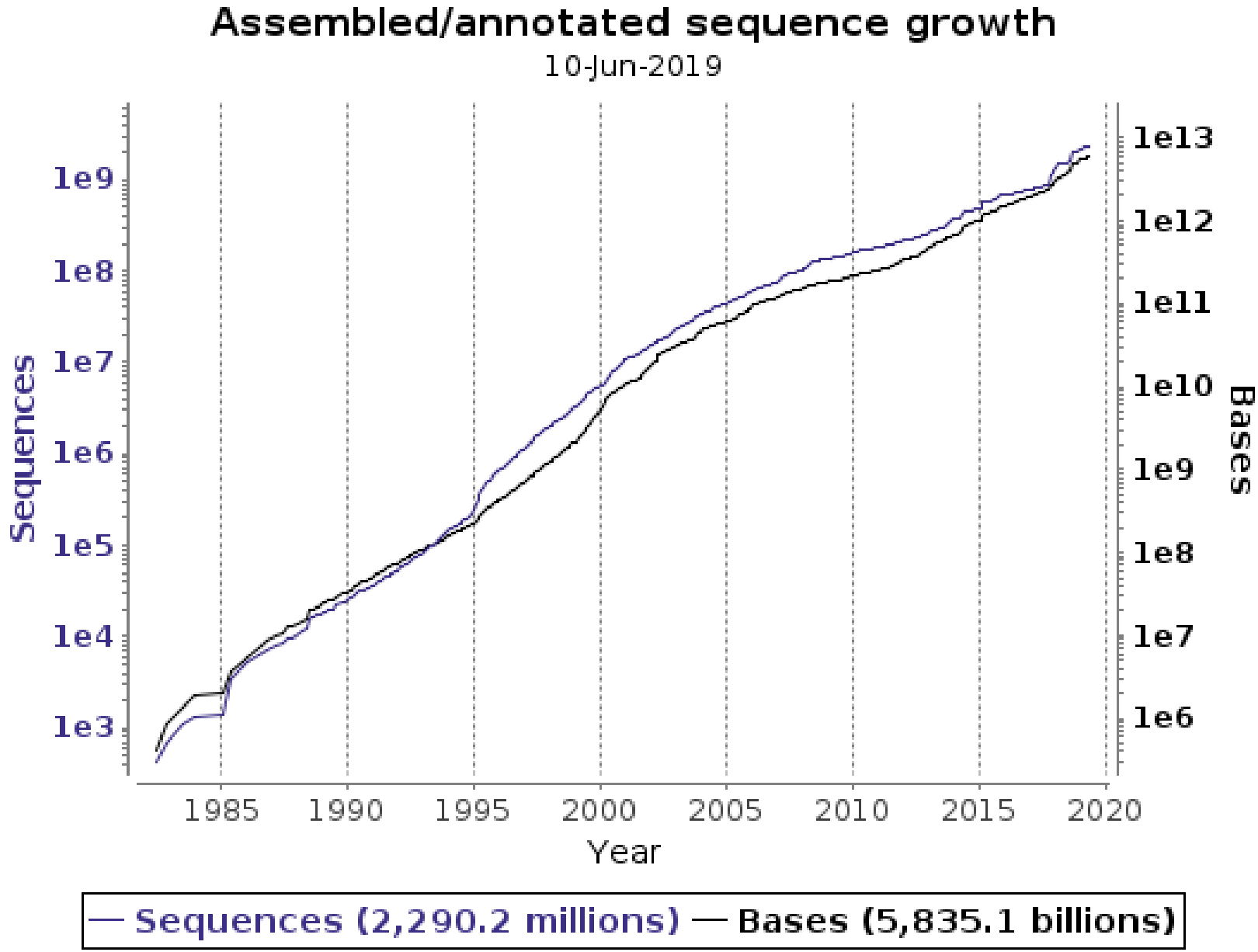
Million sequences indexing

Antoine Limasset

CNRS, Universite de Lille, CRIStAL UMR 9189, Lille

June, 2019

Biological Big Data



Indexing global archive

You are interested in a given

- Gene
- Contig
- Transcript
- Genome

What are the genomes that may involve your sequence ?

TLDL

Proposed Solution: MinEqui MinHash

- LSH Based method
- Able to locate sources of 1kb sequences
- Index 100k bacterial genomes with 32 GB

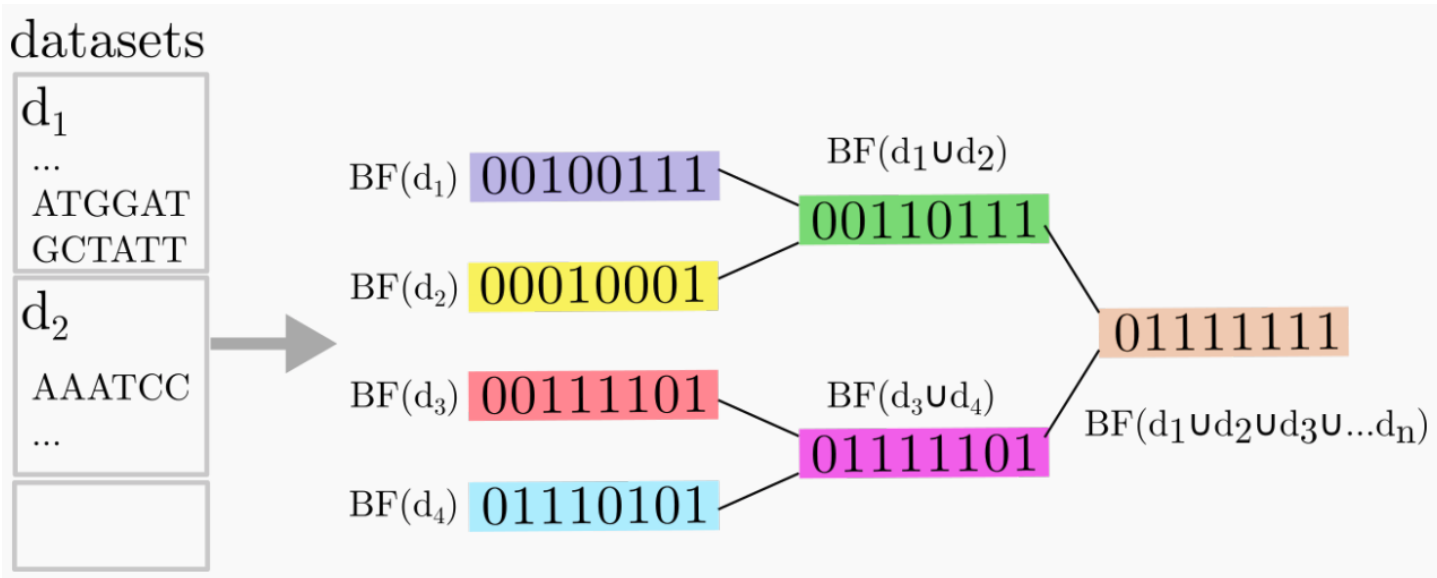
Future work

- Should be faster than know approaches
- Compressed index for faster queries
- Adaptation to genome size
- Applications (overlap detection clustering etc. . .)
- 1M genomes and beyond !

Sequence Bloom Tree

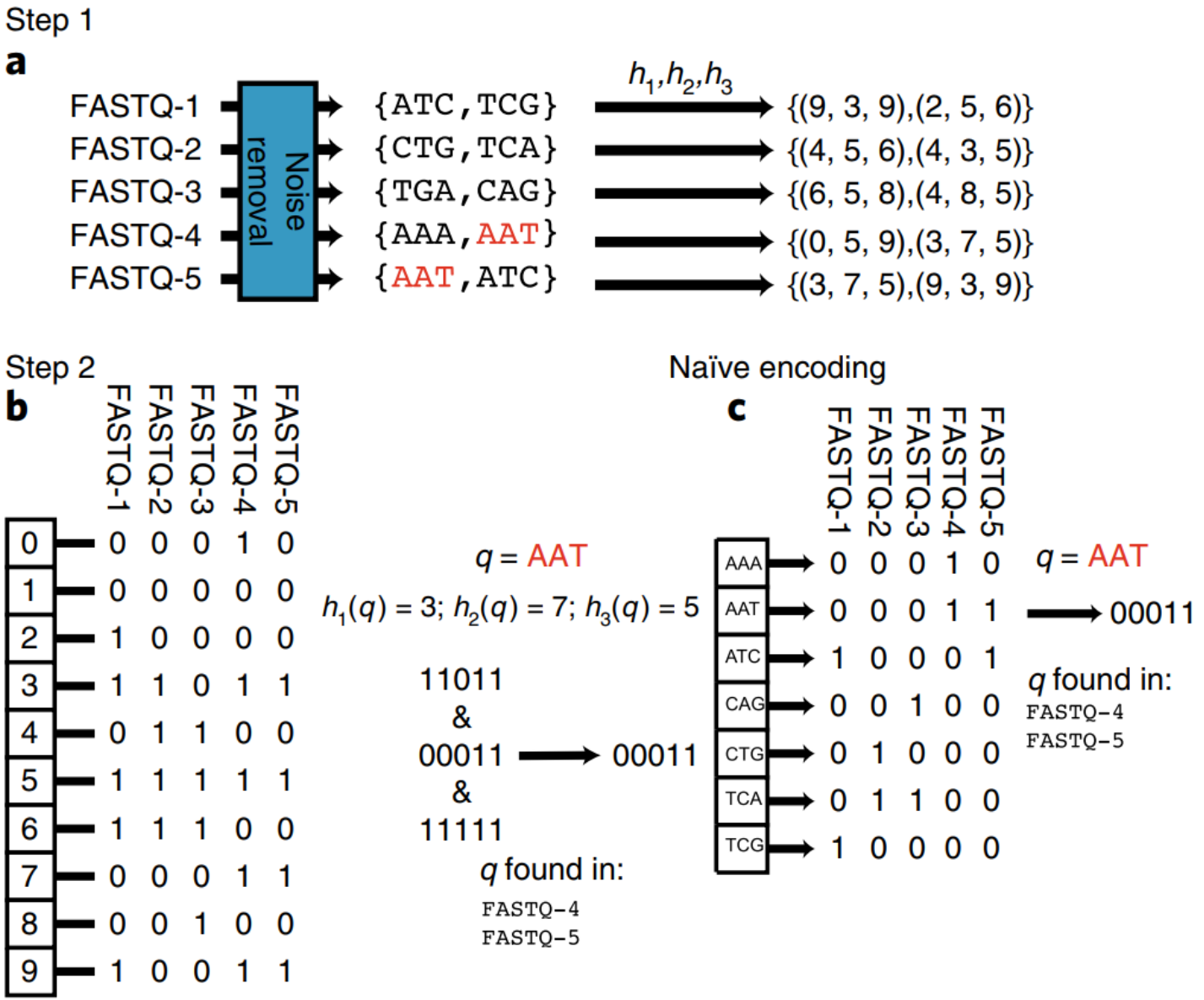
SBT problem

Given thousands sequencing datasets and a query sequence:
Which datasets contain at least 80% of the query kmers



From Marchet Camille, Bioinformatics Day Helsinki/ DSB2019

BIGSI



From Bradley2019, Nature Biotech.

Pros and cons

Higly studied problem

Treelike: SBT, SSBT, AllSomeSBT, HowDeSBT
Matrix-like: BIGSI, COBBS

Kmer level indexing

Very sensitive
All (solid) kmers are inserted in several BF

Index Construction/update

Matrix indexes are easy to construct and update
Tree indexes construction can be costly and update may lead to suboptimal performances over index reconstruction

Mash

- Use "H-min" minhash
- Works on genomes, SR and LR datasets

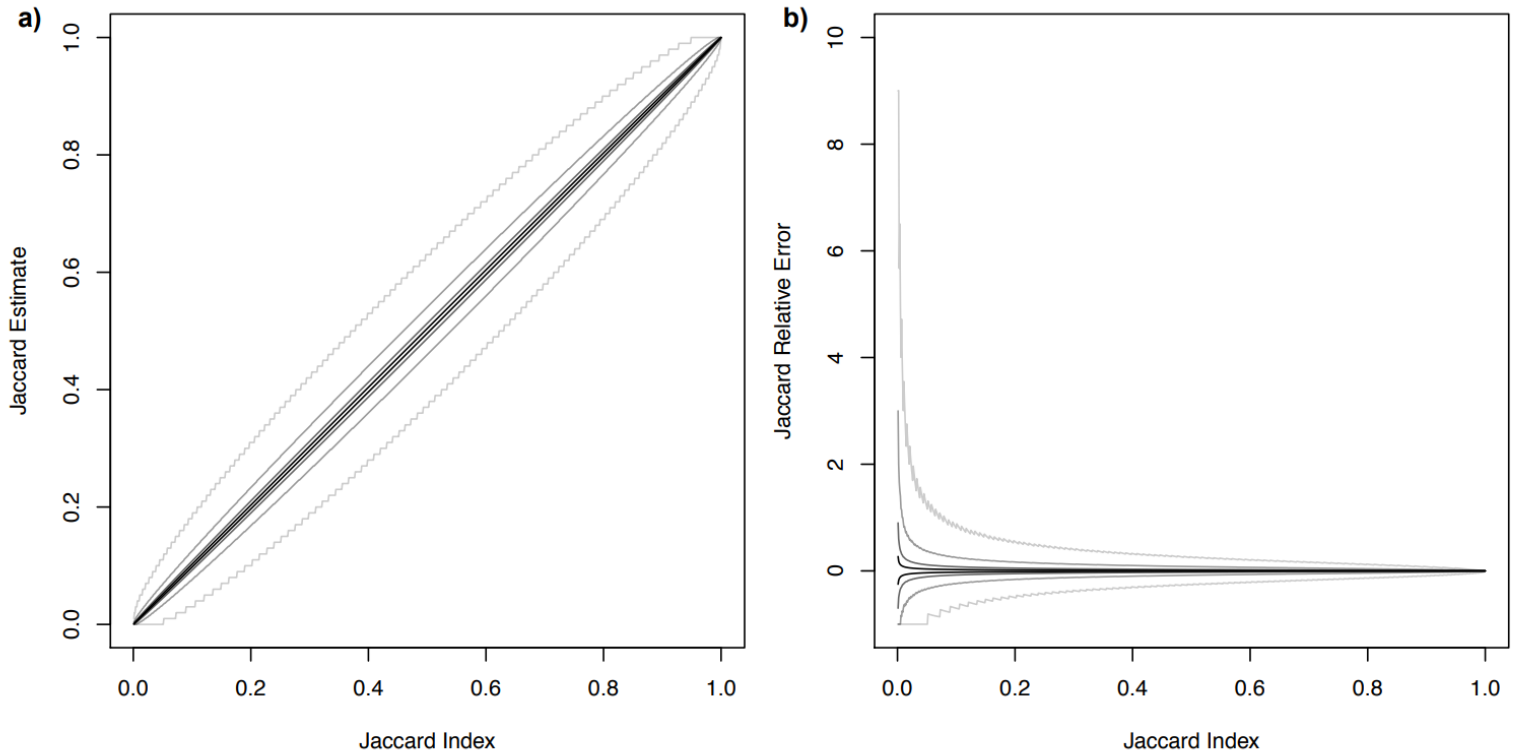


Figure S1. Absolute and relative error bounds for Mash Jaccard estimates given various sketch sizes. Increasing sketch sizes are progressively shaded from *s*=100 (light gray), *s*=1,000, *s*=10,000, and *s*=100,000 (black). Upper and lower bounds are drawn using the binomial inverse

From Supplementary data of "Mash: fast genome and metagenome distance estimation using MinHash"

Recent Work

Dashing

- Rely on cardinality estimator as Hyperloglog
- Can estimate intersection, cardinality and union

Mash Screen

- Use Bloom filter before to hit the fingerprint index
- Can approximate containment

Pro and cons

Pros

Awesome

Cons

What happens for small sequences ?

HyperMinhash

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015 1

HyperMinHash: MinHash in LogLog space

Yun William Yu, Griffin M. Weber

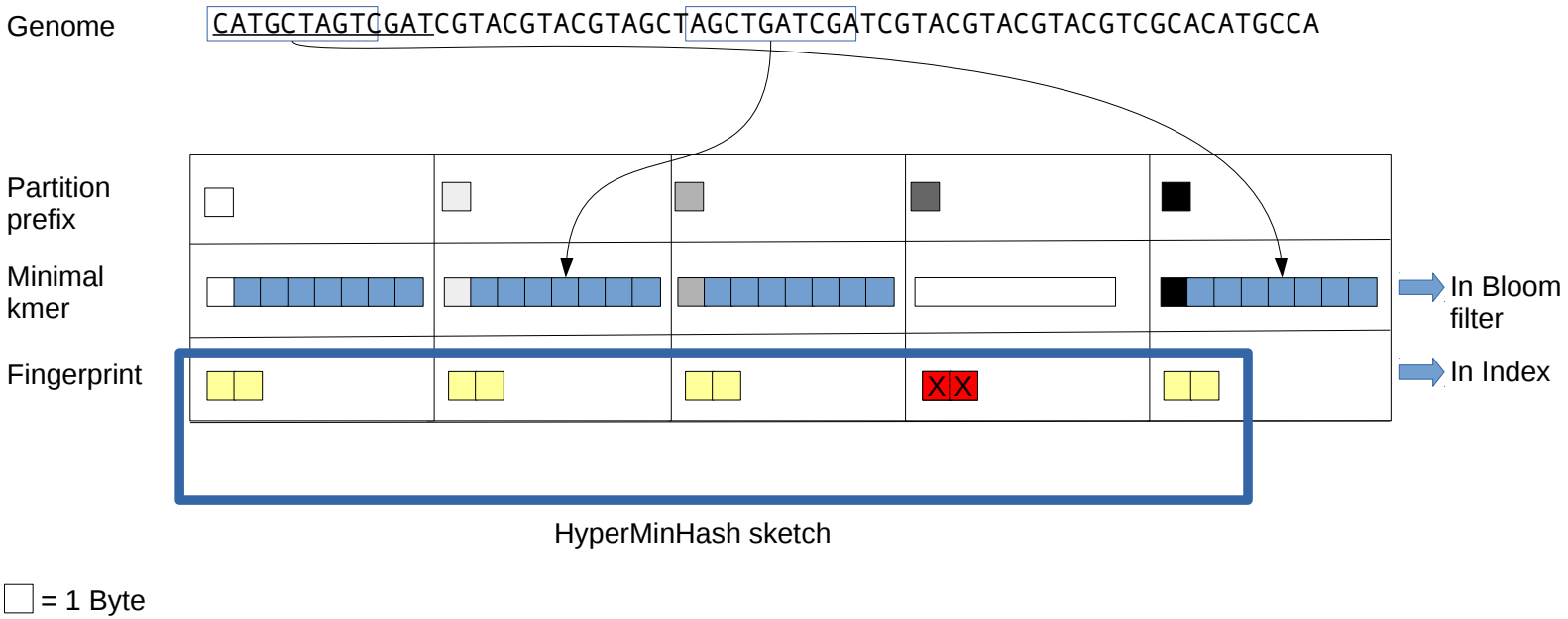
Three flavor of minhash

- H-hash
- H-min
- **H-partition**

HyperMinHash

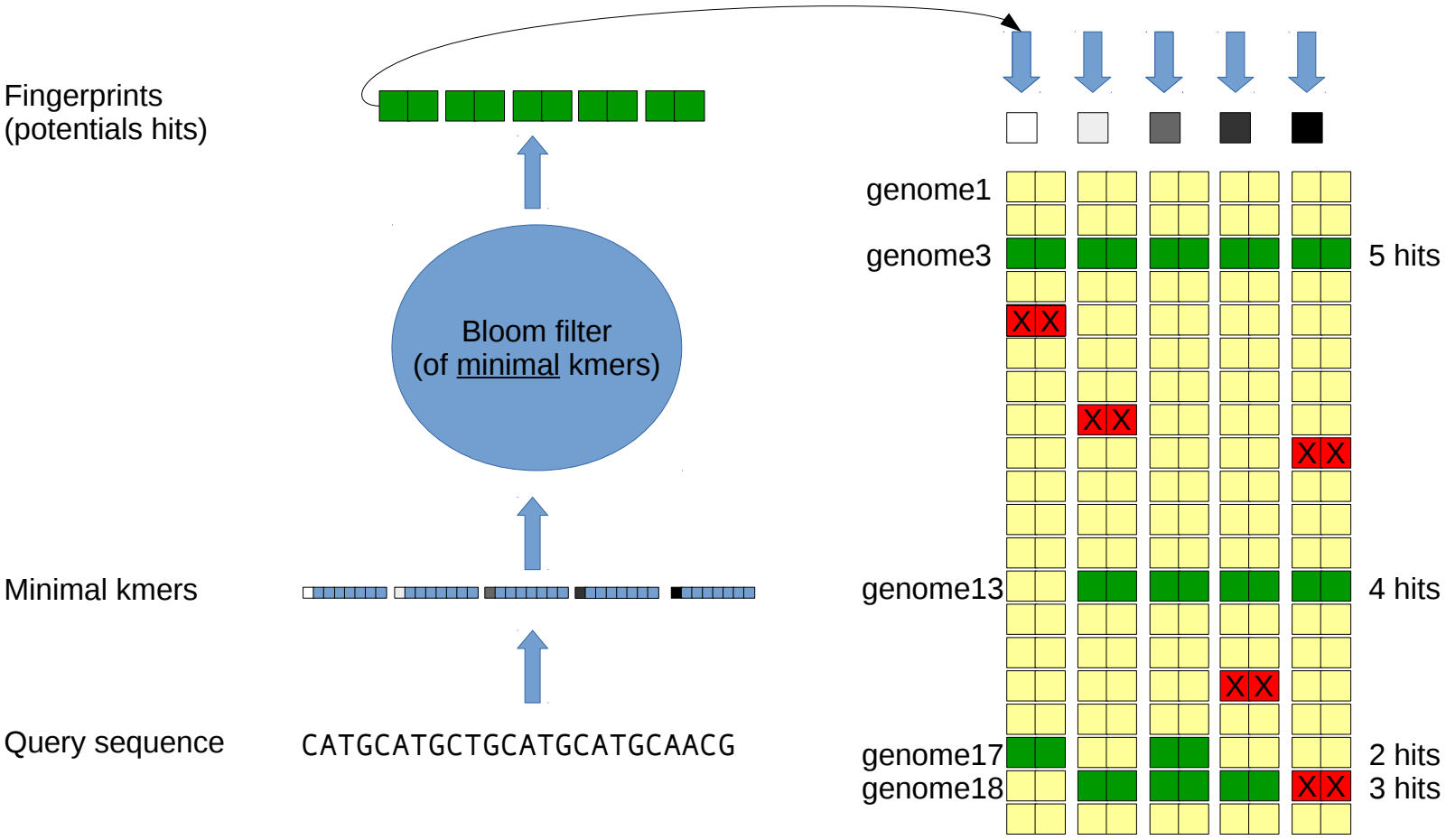
- Built-in cardinality estimator
- Better space complexity than minhash

Proposed sketch



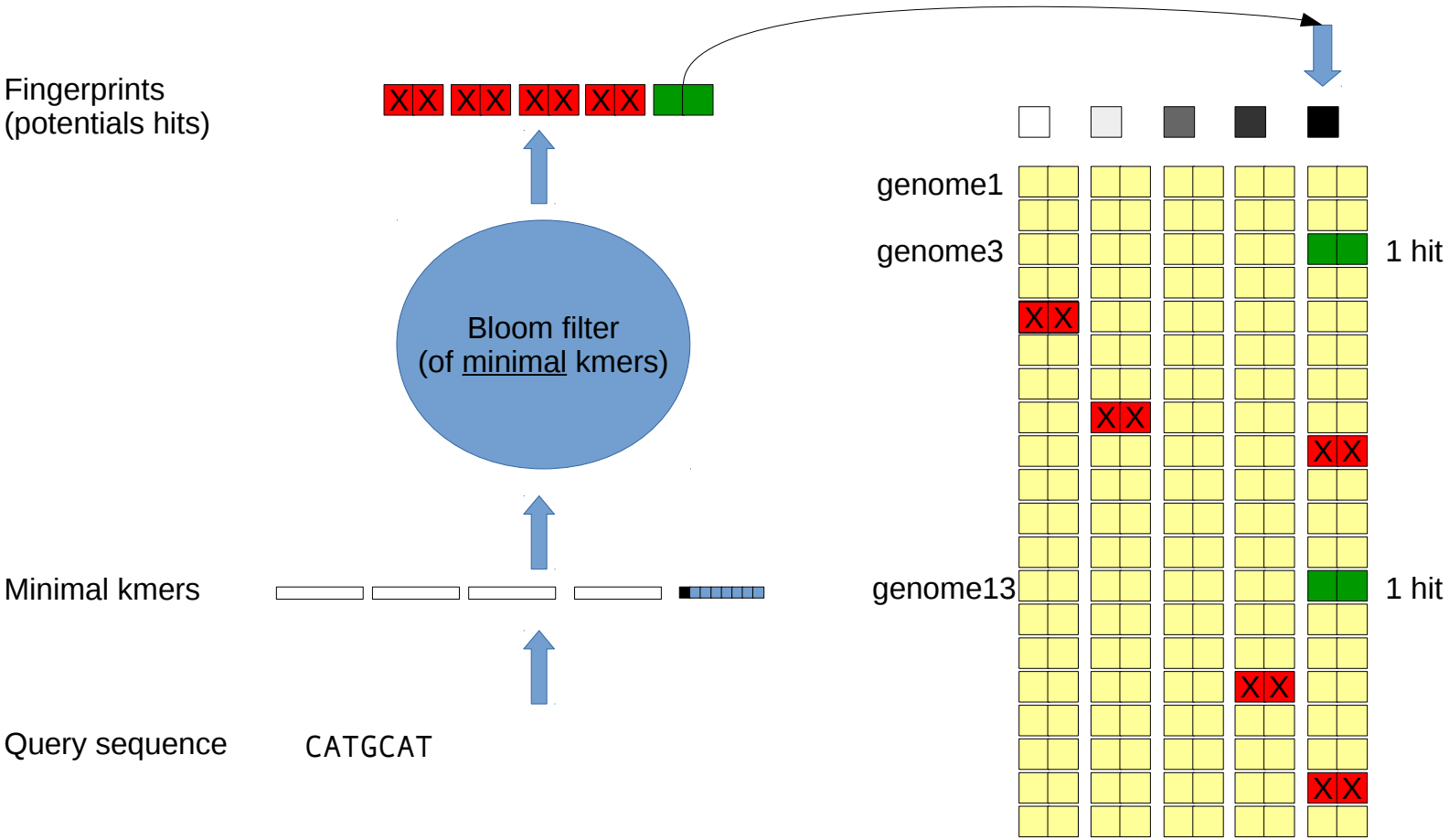
= 1 Byte

Example query

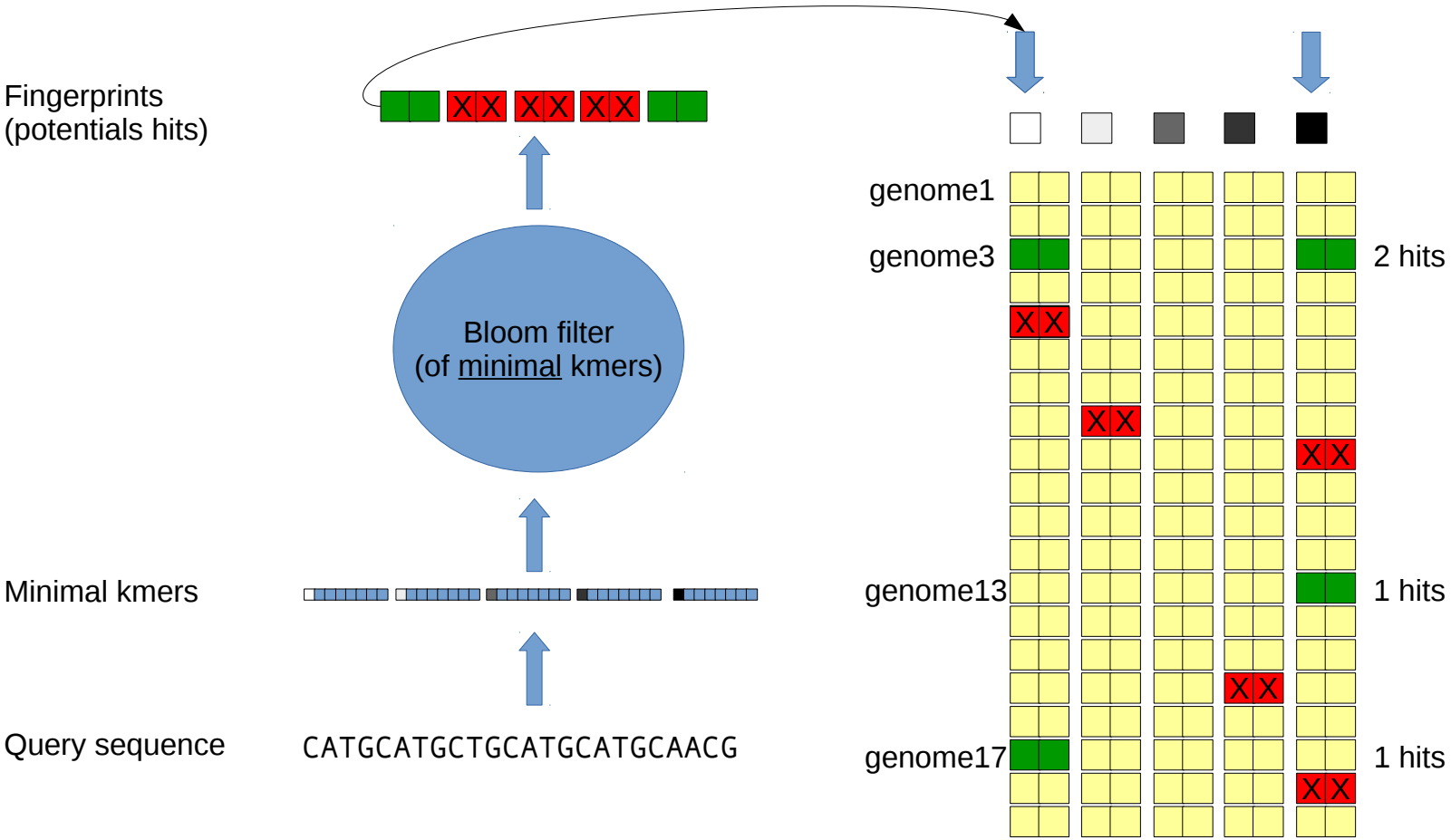


Output:
genome3: 5 hits; genome13: 4 hits; genomes18: 3 hits; genome17: 2 hits;

Small query sequence



Erreneous query sequence



Pro and cons

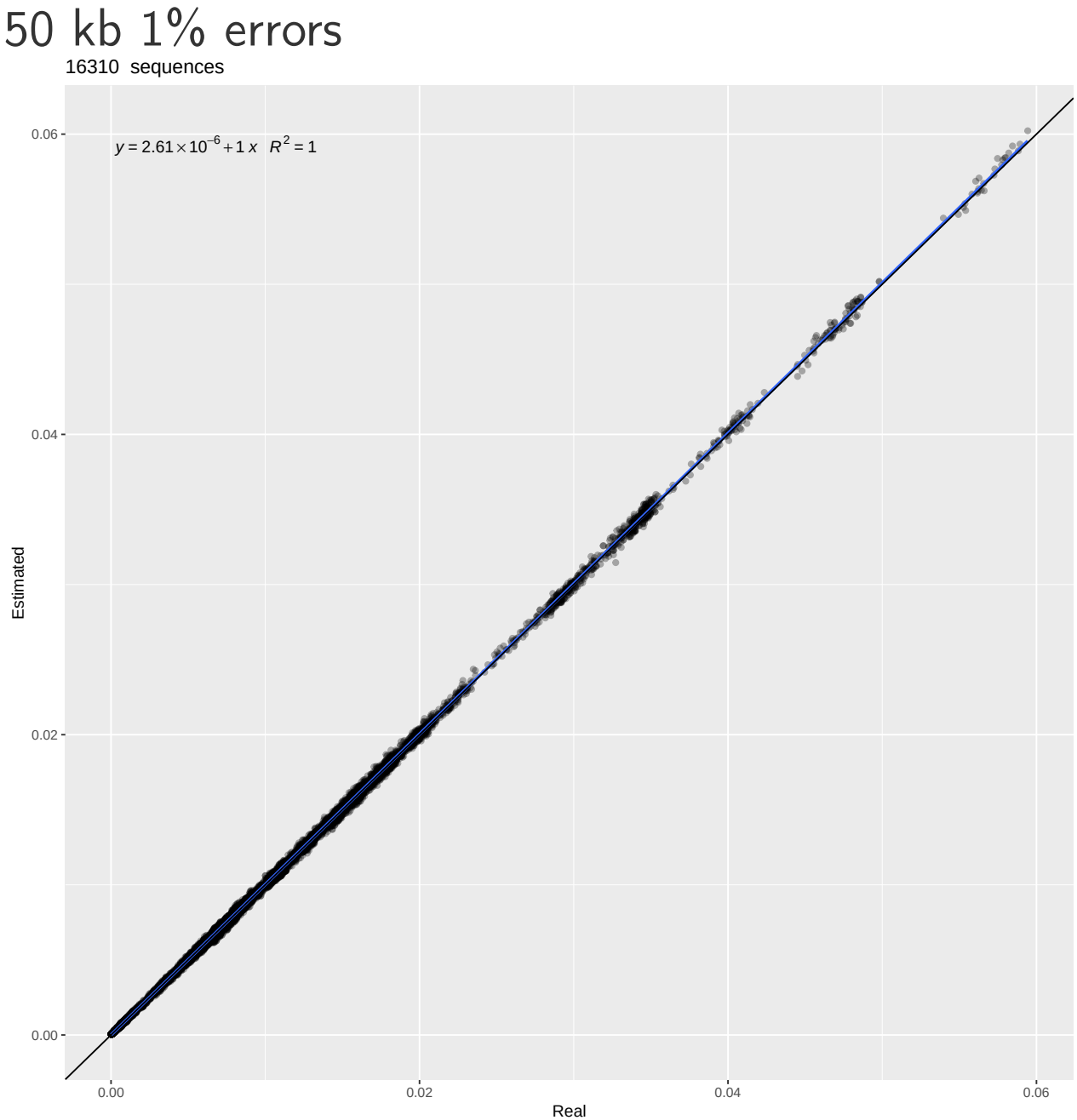
Pros

- Easy to construct and update
- Low memory usage (< 0.5 MB per genome)
- (Most) Erroneous kmers are filtered
- Only check n column (for n relevant kmers, $n < H$)

Cons

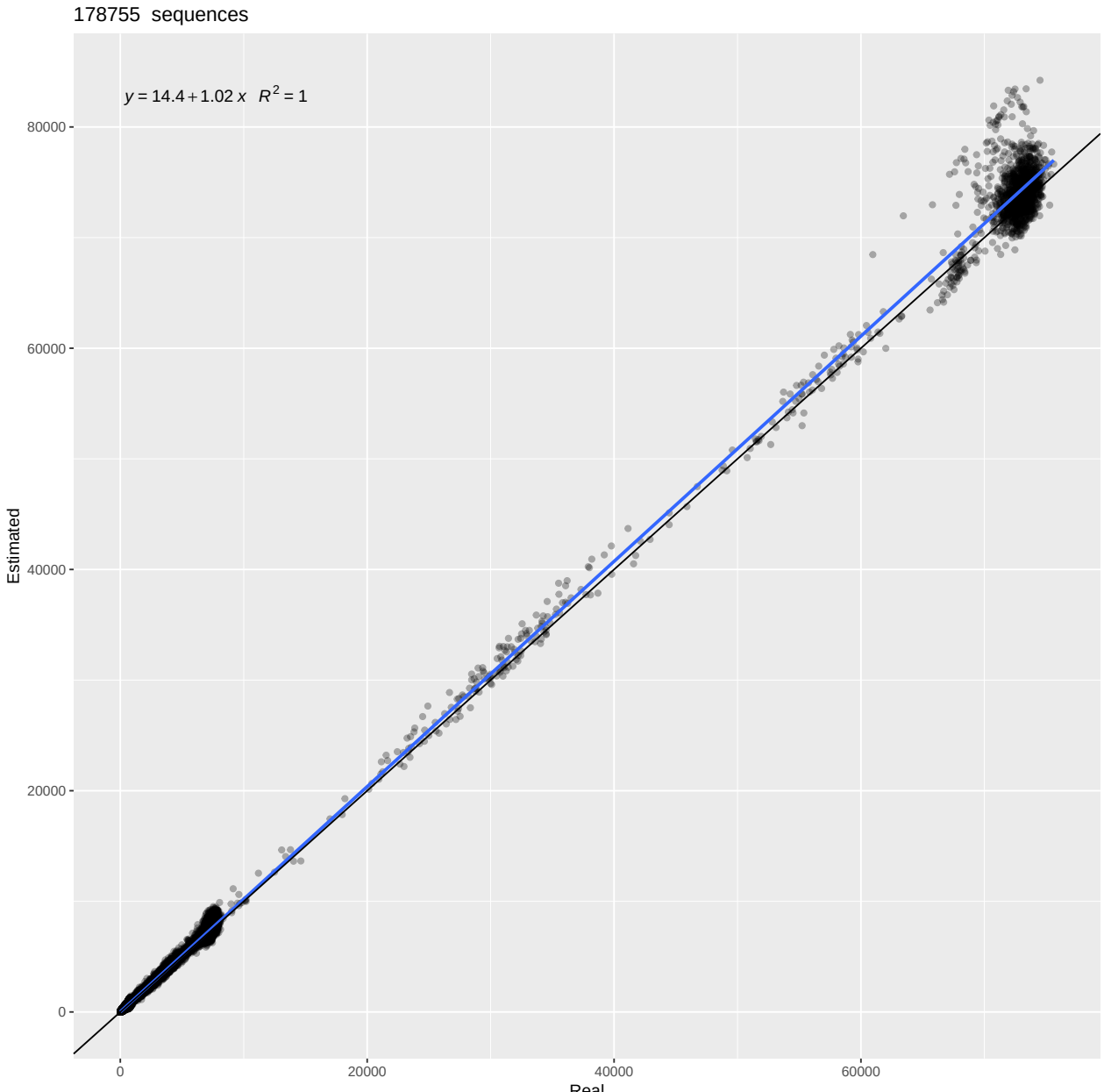
Not all kmers are indexed ! (subsampling)
May miss some matches for small sequences (<1000bp)

Shared kmer estimation: contigs

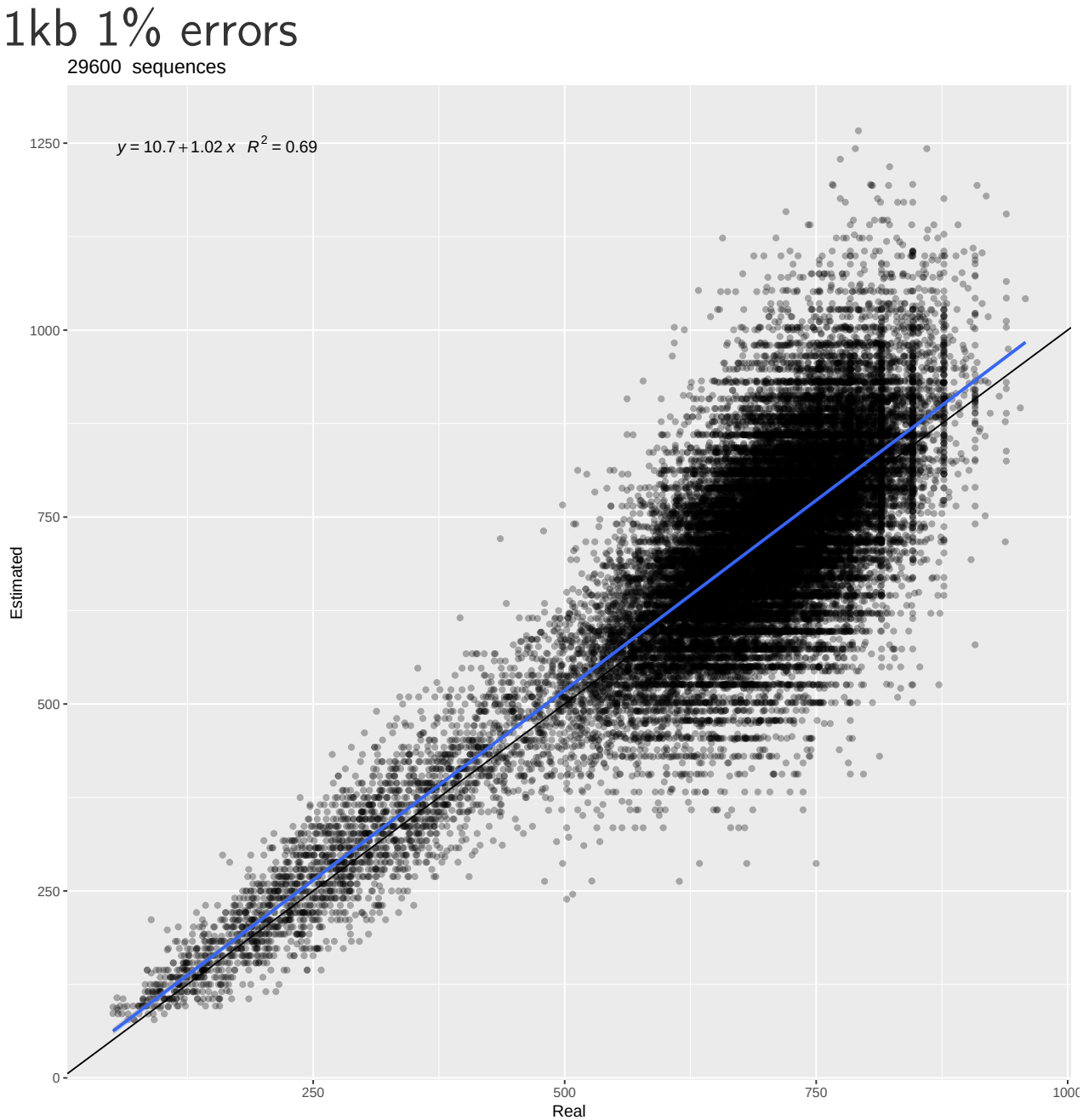


Shared kmer estimation: Varying length

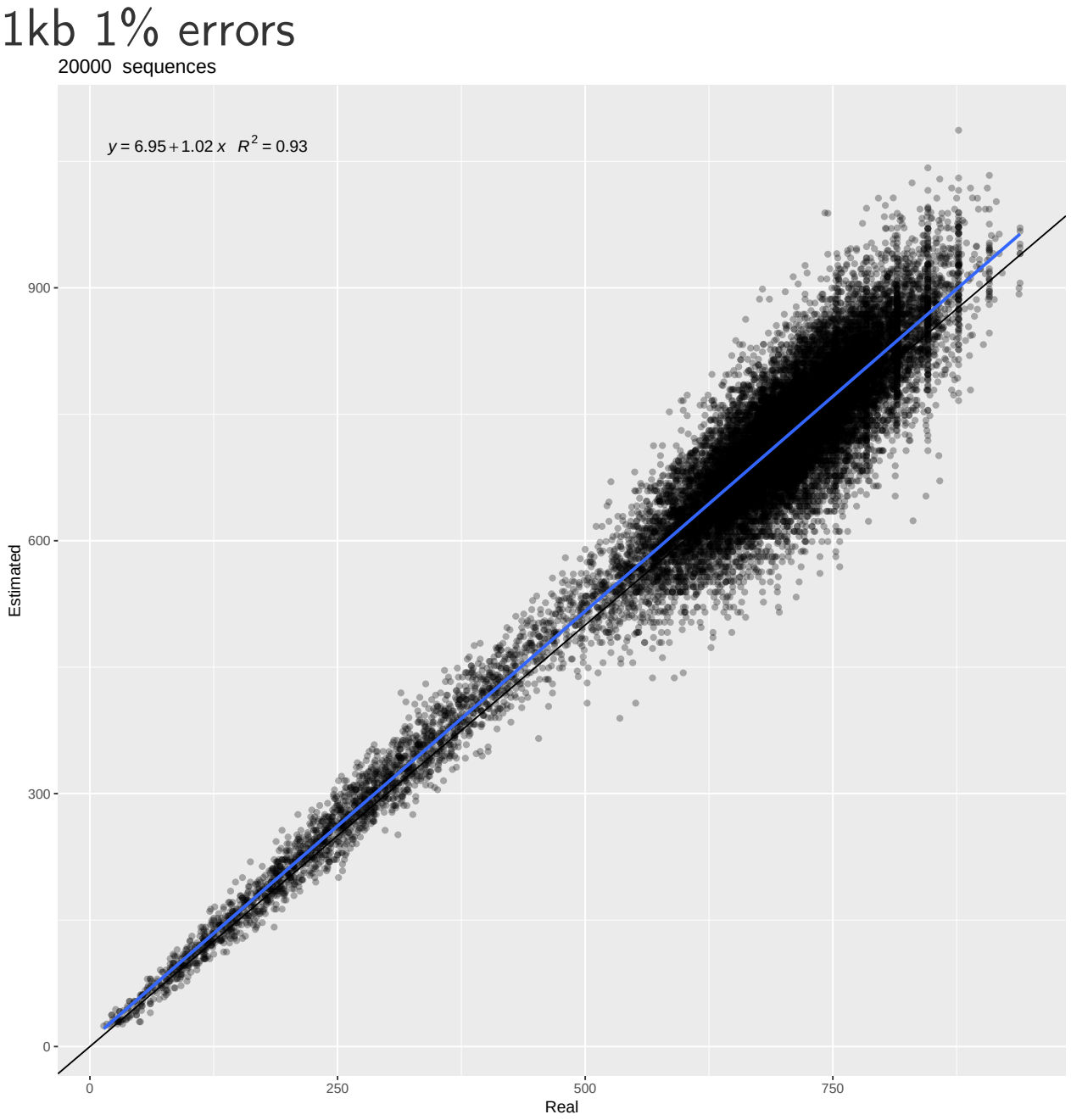
100/10/1kb 1% errors



Shared kmer estimation: Limit case

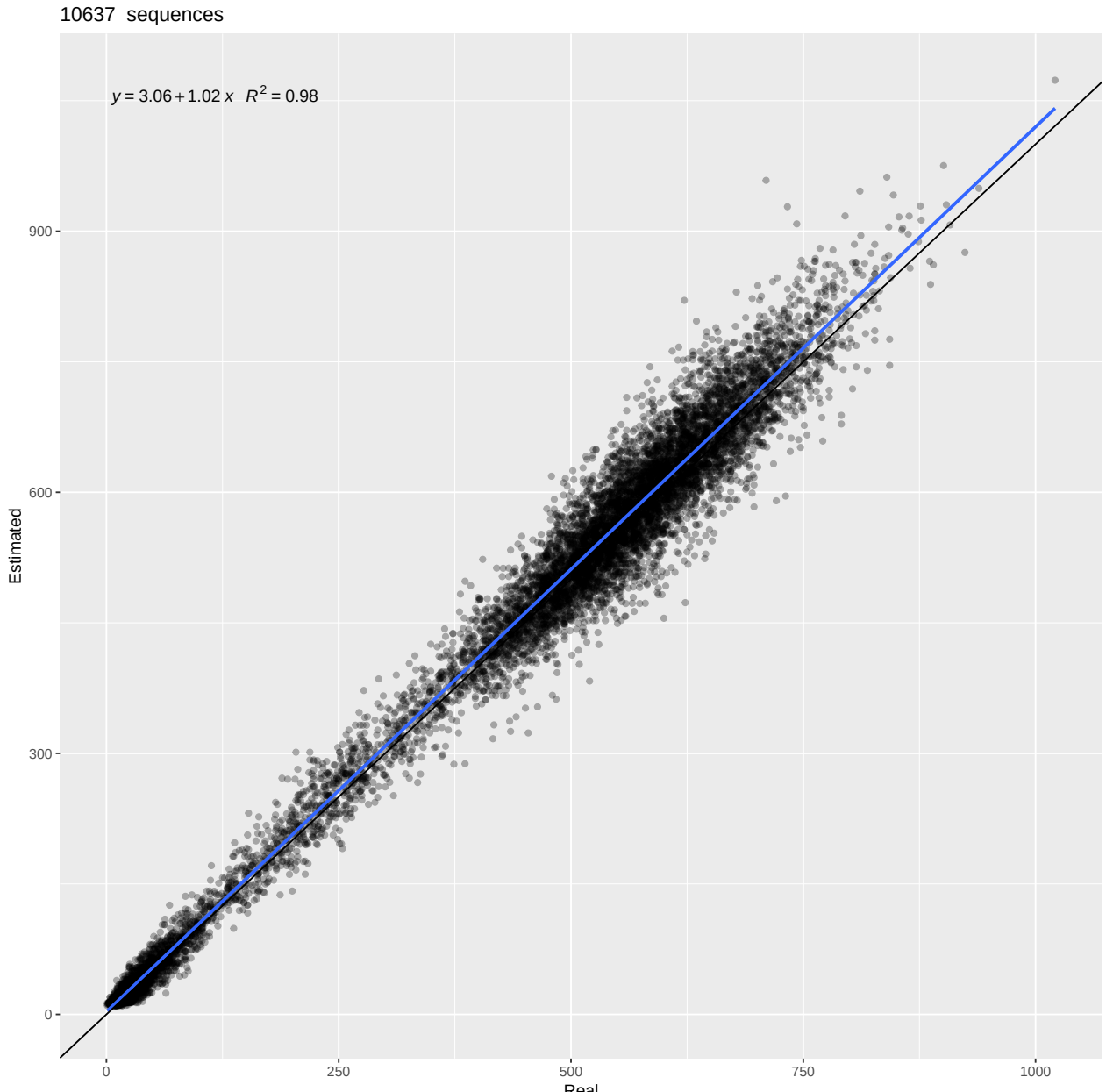


Shared kmer estimation: Limit case -> More sensitivity



Shared kmer estimation: Long reads

20kb 10% errors (21-mer)



Current limitations

- Query linear in the index size
- Naive compression
- Different genome size

Fingerprint size

More Smaller fingerprints

B-bit minhash use 1 bit fingerprint (if Jaccard similarity¿0.5)
bindash uses *approx*14 bits fingerprint (according to the genome size)

One byte fingerprint

Bounded hyperminhash fingerprint (4/5 bit for exponent and 4/3 bit of hash?)
One byte hyperloglog fingerprint (Dashing)
Double hyperloglog fingerprint (4+4) ?

Dense index

Fingerprint list

Query $\mathcal{O}(Genomes)$
Hard to compress

Genome lists per fingerprint

Query $\mathcal{O}(Hits)$
Sorted genome identifiers can be compressed using delta encoding

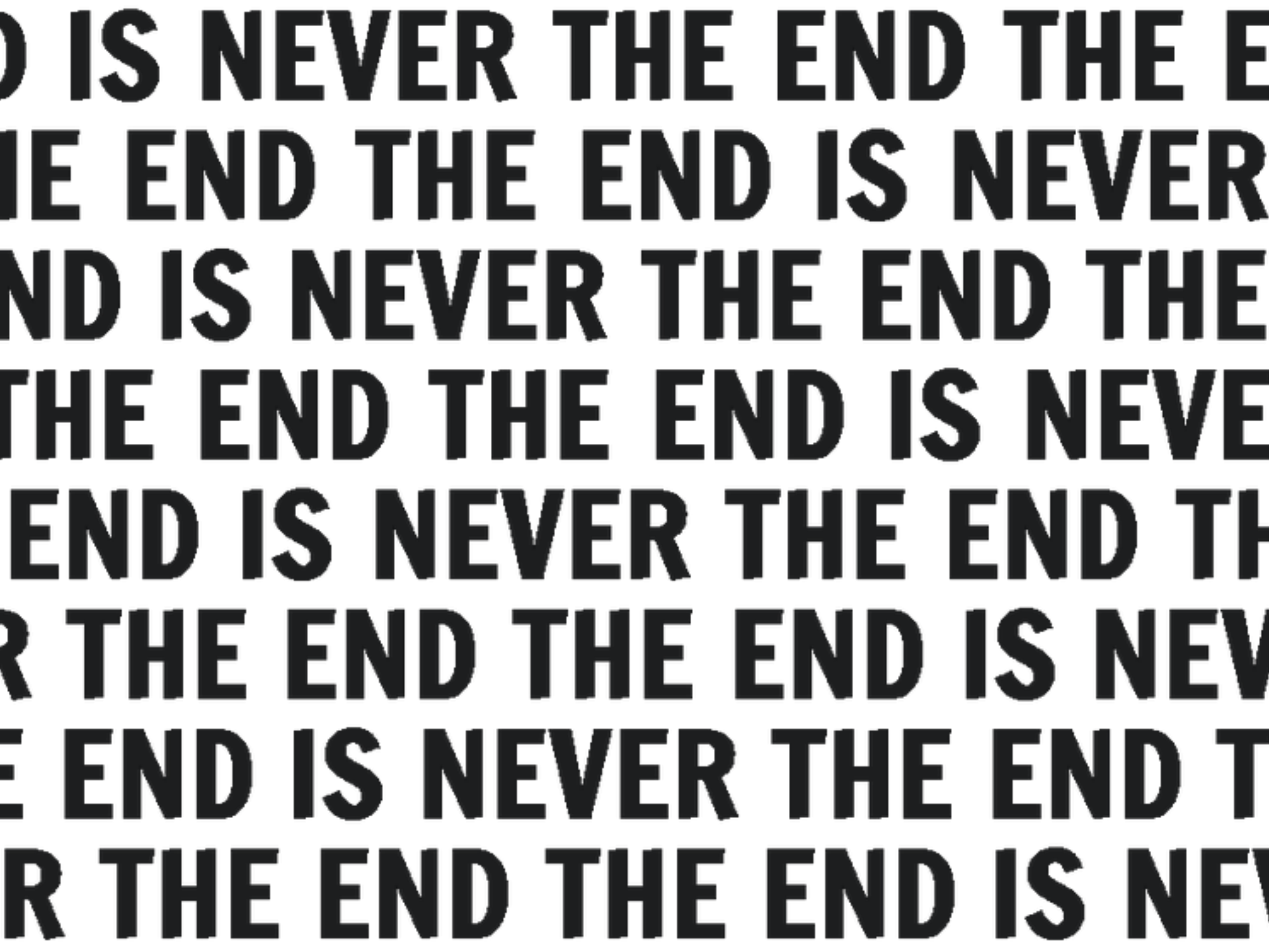
Win win scenario

We can expect sub-linear query
We can expect a high compression ratio

In development

Remaining problems

- Sensibility
- Index both gigabase and megabases genomes
- Low level optimizations


D IS NEVER THE END THE E
HE END THE END IS NEVER
ND IS NEVER THE END THE
THE END THE END IS NEVE
END IS NEVER THE END TH
R THE END THE END IS NEV
E END IS NEVER THE END T
R THE END THE END IS NE