Data compression and Its Application in Biological Data Management

M. Oğuzhan Külekci
kulekci@itu.edu.tr

Informatics Institute, Istanbul Technical University, Turkey

2018 International Workshop on String Algorithms in Bioinformatics (StringBio), October 25, 2018
University of Central Florida, Orlando, FL

Outline

Bioinformatics - Big Picture and the Sequencing Pipeline

Data Compression

Compression of Sequencing Data

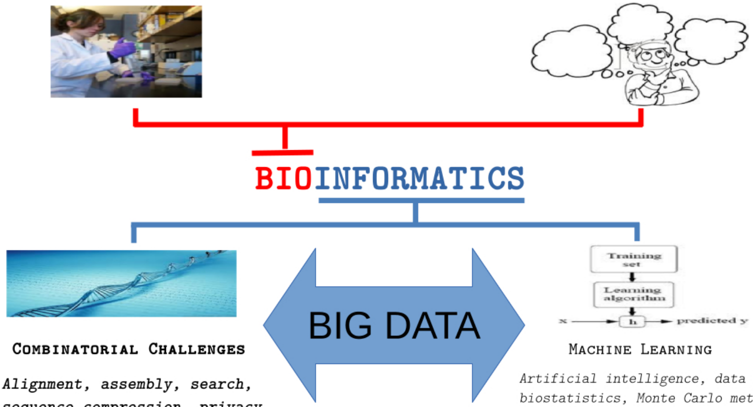
Compression Beyond Space Efficiency

Future Research Avenues

Section 1

Bioinformatics - Big Picture and the Sequencing Pipeline

Bioinformatics - *a subjective view*



Wet Lab.

Biological question,
hypothesis generation and testing

BIOINFORMATICS

BIG DATA

COMBINATORIAL CHALLENGES

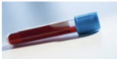
*Alignment, assembly, search,
sequence compression, privacy,
archival,structural variations....*

MACHINE LEARNING

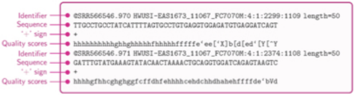
*Artificial intelligence, data mining,
biostatistics, Monte Carlo methods,
HMM/bayesian inference, prediction...*

DNA Sequencing Pipeline

Target DNA sequence:



FASTQ FILE

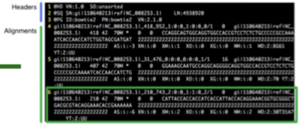


```
@SRR566546.970 HWUSI-EAS1673_11067_FC7070M:4:1:2299:1109 length=50
TTGCCTGCCTATCATTTTAGTGCCTGTGAGGTGGAGATGTGAGGATCAGT
+
hhhhhhhhhhghhghhhhhfhhhhhfffffe'ee['X]b[d[ed'[Y["Y
@SRR566546.971 HWUSI-EAS1673_11067_FC7070M:4:1:2374:1108 length=50
GATTTGTATGAAAGTATACAACTAAAACTGCAGGTGGATCAGAGTAAGTC
+
hhhhgfhhcghghggfcffdhfehhhhcehdchhdhahehffffde'bVd
```

High Computing Power

Read mapping

SAM/BAM FILE

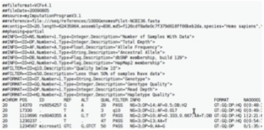


```
@HD VN:1.0  SO:unsorted
@SQ SN:gi|110640213|ref|NC_008253.1|    LN:4938920
@PG ID:bowtie2  PN:bowtie2  VN:2.1.0
gi|110640213|ref|NC_008253.1|_418_952_1:0:0_1:0:0_0/1   0   gi|110640213|ref|NC_
008253.1|   418 42  70M *   0   0   CCAGGCAGTGGCAGGTGGCCACCGTCCTCTCTGCCCCCGCCAAA
ATCACCAACCATCTGGTAGCGATGAT  2222222222222222222222222222222222222222222222222
2222222222222222222  AS:i:-3 XN:i:0  XM:i:1  XO:i:0  XG:i:0  NM:i:1  MD:Z:8G61
YT:Z:UU
gi|110640213|ref|NC_008253.1|_31_476_0:0:0_0:0:0_1/1    16  gi|110640213|ref|NC_
008253.1|   407 42  70M *   0   0   GGAAAGCAATGCCAGGCAGGGGCAGGTGGCCACCGTCCTCTCTG
CCCCCGCCAAAATCACCAACCATCTG  2222222222222222222222222222222222222222222222222
2222222222222222222  AS:i:0  XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:70 YT:Z
:UU
gi|110640213|ref|NC_008253.1|_210_743_2:0:0_1:1:0_2/1   0   gi|110640213|ref|NC_
008253.1|   210 42  70M *   0   0   CATTACCACCACCATCACCATTACCACAGGAAACGGTGCGGGCT
GACGCGTACAGGAAACACCGAAAAAA  2222222222222222222222222222222222222222222222222
2222222222222222222  AS:i:-6 XN:i:0  XM:i:2  XO:i:0  XG:i:0  NM:i:2  MD:Z:30T31A7
   YT:Z:UU
```

Each row describes a single alignment of a raw read against the reference genome.
Each alignment has 11 mandatory fields, followed by any number of optional fields.

Variation Detection

VCF FILE

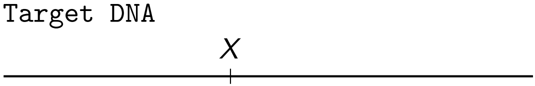


```
##fileformat=VCFv4.1
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS     ID        REF    ALT     QUAL FILTER INFO                              FORMAT      NA00001
20     14370   rs6054257 G      A       29   PASS   NS=3;DP=14;AF=0.5;DB;H2           GT:GQ:DP:HQ 0|0:48:
20     17330   .         T      A       3    q10    NS=3;DP=11;AF=0.017               GT:GQ:DP:HQ 0|0:49:
20     1110696 rs6040355 A      G,T     67   PASS   NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:
20     1230237 .         T      .       47   PASS   NS=3;DP=13;AA=T                   GT:GQ:DP:HQ 0|0:54:
20     1234567 microsat1 GTC    G,GTCT  50   PASS   NS=3;DP=9;AA=G                    GT:GQ:DP    0/1:35:
```

A description from a computational point of view

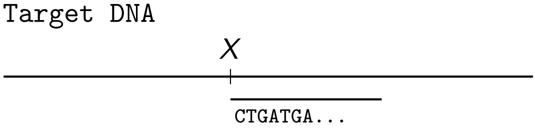
Target DNA

A description from a computational point of view



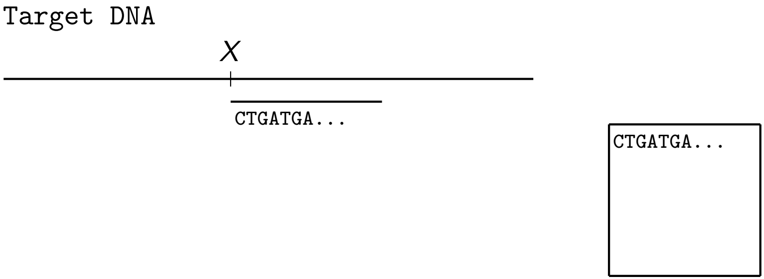
- Pick a random point X

A description from a computational point of view



- Pick a random point X
- Read the next k (fixed/variable) bases
  Important notice: The explicit value of X is not available!

A description from a computational point of view


Target DNA

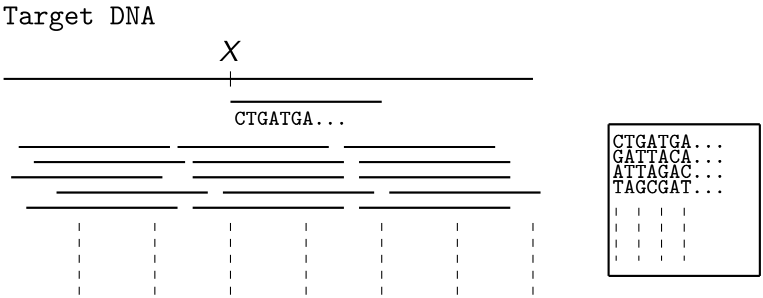
X

CTGATGA...

CTGATGA...

- Pick a random point X
- Read the next k (fixed/variable) bases
  Important notice: The explicit value of X is not available!
- Record them into a text file with *supplementary* info

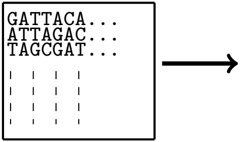
A description from a computational point of view

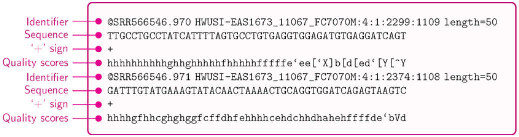


- Pick a random point X
- Read the next k (fixed/variable) bases
  Important notice: The explicit value of X is not available!
- Record them into a text file with *supplementary* info
- Repeat the same procedure hundreds of millions time.

FASTQ to SAM/BAM

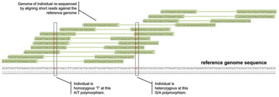
The output of the sequencing machine is the FASTQ file.





```
@SRR566546.970 HWUSI-EAS1673_11067_FC7070M:4:1:2299:1109 length=50
TTGCCTGCCTATCATTTTAGTGCCTGTGAGGTGGAGATGTGAGGATCAGT
+
hhhhhhhhhhghhghhhhhfhhhhhfffffe`ee[`X]b[d[ed`[Y[^Y
@SRR566546.971 HWUSI-EAS1673_11067_FC7070M:4:1:2374:1108 length=50
GATTTGTATGAAAGTATACAACTAAAACTGCAGGTGGATCAGAGTAAGTC
+
hhhhgfhhcghghggfcffdhfehhhhcehdchhdhahehffffde`bVd
```

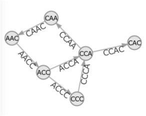
Alignment



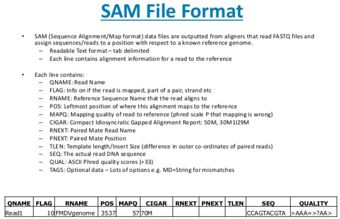
Genome of individual re-sequenced by aligning short reads against the reference genome
reference genome sequence
Individual is homozygous 'T' at this A/T polymorphism.
Individual is heterozygous at this G/A polymorphism.

When the reference genome is available, map billions of reads onto it, which generates the SAM file (seq. alignment map)

Assembly



When there is no reference, then it is akin to solving a puzzle with billions of pieces.



SAM File Format

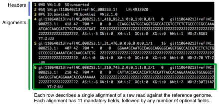
- SAM (Sequence Alignment/Map format) data files are outputted from aligners that read FASTQ files and assign sequences/reads to a position with respect to a known reference genome.
  - Readable Text format – tab delimited
  - Each line contains alignment information for a read to the reference
- Each line contains:
  - QNAME: Read Name
  - FLAG: Info on if the read is mapped, part of a pair, strand etc
  - RNAME: Reference Sequence Name that the read aligns to
  - POS: Leftmost position of where this alignment maps to the reference
  - MAPQ: Mapping quality of read to reference (phred scale P that mapping is wrong)
  - CIGAR: Compact Idiosyncratic Gapped Alignment Report: 50M, 30M1I29M
  - RNEXT: Paired Mate Read Name
  - PNEXT: Paired Mate Position
  - TLEN: Template length/Insert Size (difference in outer co-ordinates of paired reads)
  - SEQ: The actual read DNA sequence
  - QUAL: ASCII Phred quality scores (+33)
  - TAGS: Optional data – Lots of options e.g. MD=String for mismatches

QNAME	FLAG	RNAME	POS	MAPQ	CIGAR	RNEXT	PNEXT	TLEN	SEQ	QUALITY
Read1	10	FMDVgenome	3537	57	70M				CCAGTACGTA	>AAA=>?AA>

SAM/BAM to VCF

- Variations between the sequenced genome and the reference.
- Single-nucleotide-polymorphisim (SNP), copy-number-variation (CNV), structural variations, deletions/insertions/block transforms, etc...
- Validation against previously reported mutations that are collected in some number of databases

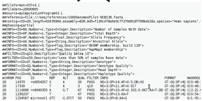
SAM/BAM File



```
@HD VN:1.0  SO:unsorted
@SQ SN:gi|110640213|ref|NC_008253.1|    LN:4938920
@PG ID:bowtie2  PN:bowtie2  VN:2.1.0
gi|110640213|ref|NC_008253.1|_418_952_1:0:0_1:0:0_0/1   0   gi|110640213|ref|NC_
008253.1|   418 42  70M *   0   0   CCAGGCAGTGGCAGGTGGCCACCGTCCTCTCTGCCCCCGCCAAA
ATCACCAACCATCTGGTAGCGATGAT  2222222222222222222222222222222222222222222222222222
222222222222222222  AS:i:-3 XN:i:0  XM:i:1  XO:i:0  XG:i:0  NM:i:1  MD:Z:8G61
YT:Z:UU
gi|110640213|ref|NC_008253.1|_31_476_0:0:0_0:0:0_1/1    16  gi|110640213|ref|NC_
008253.1|   407 42  70M *   0   0   GGAAAGCAATGCCAGGCAGGGGCAGGTGGCCACCGTCCTCTCTG
CCCCCGCCAAAATCACCAACCATCTG  2222222222222222222222222222222222222222222222222222
222222222222222222  AS:i:0  XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:70 YT:Z
:UU
gi|110640213|ref|NC_008253.1|_210_743_2:0:0_1:1:0_2/1   0   gi|110640213|ref|NC_
008253.1|   210 42  70M *   0   0   CATTACCACCACCATCACCATTACCACAGGAAACGGTGCGGGCT
GACGCGTACAGGAAACACCGAAAAAA  2222222222222222222222222222222222222222222222222222
222222222222222222  AS:i:-6 XN:i:0  XM:i:2  XO:i:0  XG:i:0  NM:i:2  MD:Z:30T31A7
    YT:Z:UU
```

Each row describes a single alignment of a raw read against the reference genome.
Each alignment has 11 mandatory fields, followed by any number of optional fields.

Variation DBs (dbSNP, dbVar, dbClinVar, etc...)

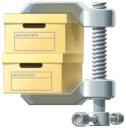


```
##fileformat=VCFv4.1
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS     ID        REF    ALT     QUAL FILTER INFO                              FORMAT      NA00001
20     14370   rs6054257 G      A       29   PASS   NS=3;DP=14;AF=0.5;DB;H2           GT:GQ:DP:HQ 0|0:48:
20     17330   .         T      A       3    q10    NS=3;DP=11;AF=0.017               GT:GQ:DP:HQ 0|0:49:
20     1110696 rs6040355 A      G,T     67   PASS   NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:
20     1230237 .         T      .       47   PASS   NS=3;DP=13;AA=T                   GT:GQ:DP:HQ 0|0:54:
20     1234567 microsat1 GTC    G,GTCT  50   PASS   NS=3;DP=9;AA=G                    GT:GQ:DP    0/1:35:
```

VCF (variation call format) file

Challenges in Managing the Sequencing Data

- Efficient storage, *the classical problem ?*
- Retrieval/search of *relevant* data from huge repositories
- Big data processing issues in the bioinformatics tools
- Distribution of the files, particularly the raw data, over internet (download data from EBI, NCBI, UCSC, etc...)
- When using a cloud service for post-processing, necessity to transmit over the lines (or by a regular courier ?)
- Privacy/security issues



may help in many cases

(and maybe beyond ?)

Section 2

Data Compression

Data Compression

Really, a very old issue...

Data size always increased proportional to the available resource!
It seems there is always hunger for more space!

- **Remove the redundancy, and squeeze the data down to its entropy, which is analogous to a vacuum storage bag.**

Main Methodology in Data Compression

Modeling

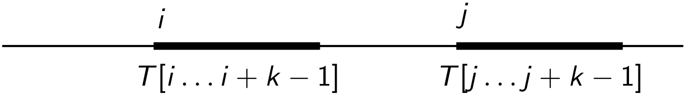
- Find a good way to describe your data, which helps to make implicit redundancies explicit.
- Very important since we can compress the data as much as we understand it!

Entropy Coding

- Encode your *transformed* data with a chosen entropy coder (e.g., Huffman or arithmetic coding).
- The effect of different entropy coder preference on performance is much less significant when compared to the effect of different **models**!

Repeat-detection based modeling (LZ77)

Text $T[1 \ldots n]$:



i j

$T[i \ldots i+k-1]$ $T[j \ldots j+k-1]$

Point j is **described** as $\langle i, k, T[j] \rangle$.

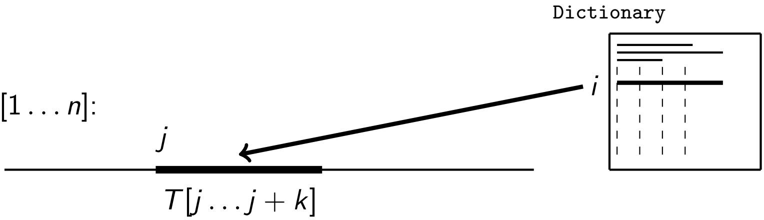
Copy k symbols starting at $T[i]$ ($i < j$) and append the symbol $T[j+k]$.

- Continue for the point $j+k+1$ with the same operation until the text is covered.
- At the end we will be left with the 3-dim vectors, which will be send to the entropy encoder.

Attention: For point j, we have to find the previous point i with the largest k !

Dictionary-based modeling (LZ78)

Text $T[1 \ldots n]$:



Point j is **described** as $\langle i, T[j+k] \rangle$.

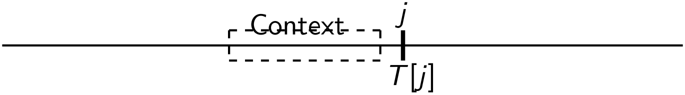
Copy the ith entry, which is of length k, from the dictionary and append the symbol $T[j+k]$.

- Continue for the point $j+k+1$ with the same operation until the text is covered.
- At the end we will be left with the 2-dim vectors, which will be send to entropy encoder.
- **Dictionary creation and maintenance ?** Different strategies that mostly process on-the-fly.

Attention: Dictionary maintenance since it can get quite large!

Statistical-bias based modeling (PPM-type)

Text $T[1 \dots n]$:

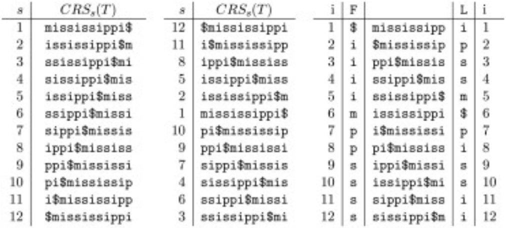


Describe point j according to its **context** - the preceding symbols-

- Among the observed symbols succeeding this context so far, $T[j]$ is the kth one, or $T[j]$ appeared with probability p. Example: What you expect to see after context que? e,r,l,..
- Send this probability (or rank) to the entropy encoder. Notice that skewed probabilities help better compression!
- Update the context, and proceed with the next position $T[j+1]$.

Attention: What would be a good context length, context modeling. Maintenance of the statistics, particularly on large context length.

The Burrows–Wheeler Transform for modeling



s	$CRS_s(T)$
1	mississippi$
2	ississippi$m
3	ssissippi$mi
4	sissippi$mis
5	issippi$miss
6	ssippi$missi
7	sippi$missis
8	ippi$mississ
9	ppi$mississi
10	pi$mississip
11	i$mississipp
12	$mississippi

a Cyclic–right–shifts

s	$CRS_s(T)$
12	$mississippi
11	i$mississipp
8	ippi$mississ
5	issippi$miss
2	ississippi$m
1	mississippi$
10	pi$mississip
9	ppi$mississi
7	sippi$missis
4	sissippi$mis
6	ssippi$missi
3	ssissippi$mi

b Sorted CRS

i	F		L	i
1	$	mississipp	i	1
2	i	$mississip	p	2
3	i	ppi$missis	s	3
4	i	ssippi$mis	s	4
5	i	ssissippi$	m	5
6	m	ississippi	$	6
7	p	i$mississi	p	7
8	p	pi$mississ	i	8
9	s	ippi$missi	s	9
10	s	issippi$mi	s	10
11	s	sippi$miss	i	11
12	s	sissippi$m	i	12

c $BWT(T) = L$

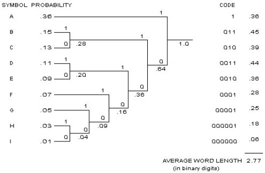
- Sort of reordering the symbols such that those sharing the same context become subsequent.
- Though initially proposed (1993) to enhance compression, it is now used as the backbone of many full-text indexing schemes.
- Extremely important, particularly for bioinformatics tools. *De facto* standard aligners are all making use of this beautiful transform.

Attention: BWT computation requires significant memory that may become inhibiting on large data sets.

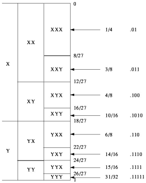
Entropy Coding Phase

Whatever model we use to describe our data, we encode this description via an entropy coder at the end.

Huffman Code



Arithmetic Code



- Surely, many variants exist
- Other alternatives, such as universal coding schemes, may also help in some situations

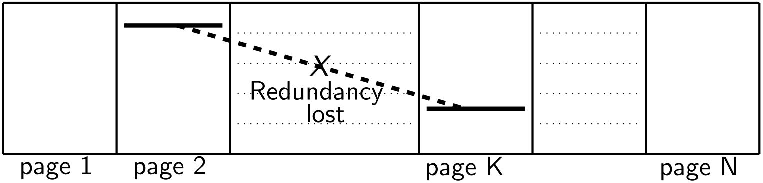
However, the effect of modeling phase seems superior to entropy coding phase in the overall performance of a compressor.

Main Challenge in Compressing Massive Data

Modeling big data is hard!

- Searching $T[1..j-1]$ to find the longest match starting at $T[j]$ for large j ?
- Maintaining a single dictionary for whole *big* data ?
- Maintaining the *relevant* statistics ?
- Huge resource requirement in computing the BWT of a large volume ?
- Others ...

Thus, almost every compressor processes data page-by-page.



Drawback: When there is redundancy between data in different pages, it cannot be detected, and thus, cannot be removed!

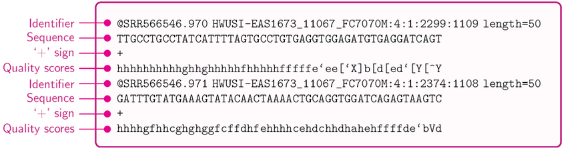
Section 3

Compression of Sequencing Data

FASTQ file compression

HTS machinery gave rise to huge increase in data generation!

Remembering the FASTQ file structure :



```
@SRR566546.970 HWUSI-EAS1673_11067_FC7070M:4:1:2299:1109 length=50
TTGCCTGCCTATCATTTTAGTGCCTGTGAGGTGGAGATGTGAGGATCAGT
+
hhhhhhhhhghhghhhhhfhhhhhfffffe‘ee[‘X]b[d[ed‘[Y[^Y
@SRR566546.971 HWUSI-EAS1673_11067_FC7070M:4:1:2374:1108 length=50
GATTTGTATGAAAGTATACAACTAAAACTGCAGGTGGATCAGAGTAAGTC
+
hhhhgfhhcghghggfcffdhfehhhhcehdchhdhahehffffde‘bVd
```

We need to represent

- Base sequence (A,T,C,G,N)
- Quality Scores
- Read labels *(actually not an issue)*

as compact as possible.

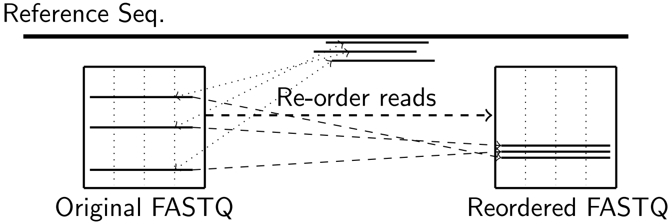
The Pistoia Alliance
Sequence Squeeze
COMPETITION
Pistoia Alliance

2011 - Sequence Squeeze Competition

See "Numanagic et. al., Comparison of high-throughput sequencing data compression tools, Nature Methods, 13(12), 2016" for an excellent survey of available tools.

Base-sequence Compression

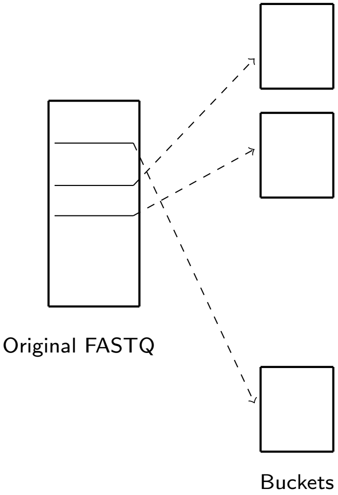
Reads are randomly selected segments from the target genome.
To cope with the challenge that highly similar, but distant reads cannot be compressed well!



Reference based methods (e.g., Fqzcomp, Fastqz, ...)

- Subject to availability of a reference sequence
- Map reads to the reference and store the locations and *differences*
- Perform a *light* alignment omitting deep inspection, for speed-up
- Compression ratio is better than reference-free approaches

Base-sequence Compression



Classify reads into bins according to their anchor signatures ?

Reference-free methods

- No need for a reference!
- Split reads into bins according to representative *anchors*
    - Longest core substring (SCALCE)
    - Minimizers (Orcom, Mince)
    - you can try another ?
- Each bin contains reads with high overlaps
- Compress each bin seperately
- How to cluster reads, how many bins, managing the bin buffers ...

Base-sequence Compression

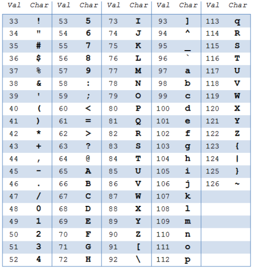
Other approaches

- Assembler based compressors (Quip, Leon, KIC)
  Create a reference genome by assembling some number of reads in the file, and then represent reads according to this reference
- Simply use PPM type compression by using longer context length, e.g., 12 or more. (DSRC, DSRC2, ...)
- BWT-based solutions (BEETL)
- SAMtools, CRAMtools suites *de facto* standard in industry

Quality Score Compression

- The accuracy of the sequencing machine when calling a base
- Quality score is $Q = -10 \log_{10} P$
  P: the probability that the base-call is wrong

Probability of error	Q-score	Printed Symbol
0.1	10	+
0.01	20	5
0.001	30	?
0.0001	40	I
0.00001	50	S
0.000001	60	]



Val	Char	Val	Char	Val	Char	Val	Char	Val	Char
33	!	53	5	73	I	93	]	113	q
34	"	54	6	74	J	94	^	114	R
35	#	55	7	75	K	95	_	115	S
36	$	56	8	76	L	96	`	116	T
37	%	57	9	77	M	97	a	117	U
38	&	58	:	78	N	98	b	118	V
39	'	59	;	79	O	99	c	119	W
40	(	60	<	80	P	100	d	120	X
41	)	61	=	81	Q	101	e	121	Y
42	*	62	>	82	R	102	f	122	Z
43	+	63	?	83	S	103	g	123	{
44	,	64	@	84	T	104	h	124	|
45	-	65	A	85	U	105	i	125	}
46	.	66	B	86	V	106	j	126	~
47	/	67	C	87	W	107	k		
48	0	68	D	88	X	108	l		
49	1	69	E	89	Y	109	m		
50	2	70	F	90	Z	110	n		
51	3	71	G	91	[	111	o		
52	4	72	H	92	\	112	p		

- Very important in variant-calling phase (SAM-to-VCF)!

Base-sequence compression has to be **lossless**, but quality score compression may be a **lossy** one.

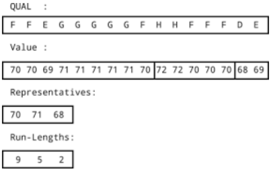
Quality Score Compression

Lossless Scenario:

- Make use of universal codes, e.g., Golomb/Rice, run–length
- Achieve a PPM-type compression according to a **model**

Lossy Scenario:

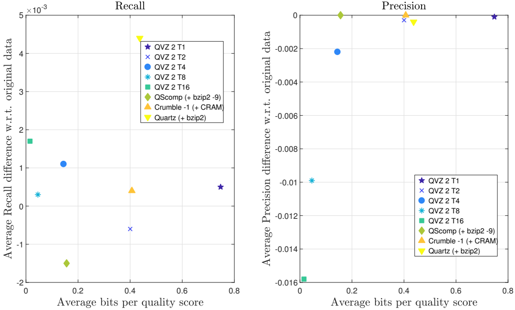
- Compress the **quantized** QS



```
QUAL  :
F  F  E  G  G  G  G  G  F  H  H  F  F  F  D  E
Value :
70 70 69 71 71 71 71 71 70 72 72 70 70 70 68 69
Representatives:
70  71  68
Run-Lengths:
9  5  2
```

Canovas et. al., "Lossy compression of quality scores in genomic data ", Bioinformatics, 30(15), 2014

- To measure the effect of this quantization, re-run the VCF creation with the quantized values and observe the difference



Voges et. al., "A two level scheme for quality score compression", JCB, 25(10), 2018

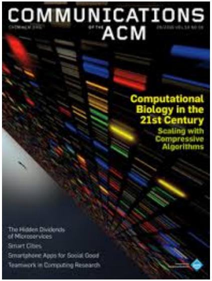
Section 4

Compression Beyond Space Efficiency

Compressive Genomics

- In-memory data processing is a lot more efficient than external memory.
- However, generally data size is larger than the available memory.
- Capability to operate on the compressed data helps to process more data in one shot, and thus, improves I/O efficiency.

”Algorithms that compute directly on compressed genomic data allow analyses to keep pace with data generation.”
Compressive genomics, Loh & Baym & Berger, Nature Biotechnology, 2012



COMMUNICATIONS
OF THE ACM

Computational Biology in the 21st Century
Scaling with Compressive Algorithms

The Hidden Dividends of Microservices
Smart Cities
Smartphone Apps for Social Good
Teamwork in Computing Research

Directly Operable Compressed Data

- What if we need to extract just one item from the zipped vacuum bag? Inflate and deflate?
- Better to find ways to act directly on zip-bag.
- Compressed indexing and *compressed data structures*!

Compressed data structures

- Represent the data structure in space as small as possible without a loss in its functionality.
  G. Jacobson, Succinct Static Data Structures, PhD thesis, CMU, 1989.
  D. Clark: Compact Pat Trees, PhD thesis, University of Waterloo, Canada, 1996
- Compressed arrays, lists, trees, ...
- Very active area since 2000 especially in data management and information retrieval.
  J. Vitter, "Compressed Data Structures with Relevance", CIKM'12

Compressed Self-text Indexes

- Classical Index: Index + Data (inverted-index)
- Self-Index: Index revealing data without need to the data (BWT)
- Compressed self-index: Self-index in size close to the compressed size of the data (BWT + compressed data structures)

Today, aligners (BWA, Bowtie,...) use genome indexes heavily built with compressed data indexes.

Burrows-Wheeler Aligner

BOW TIE

Pioneering works in compressed self text indexing:

- FOCS'2000, Ferragina and Manzini, "Opportunistic data structures..."
- STOC'2000, Grossi and Vitter, "Compressed suffix arrays...."

For practical implementations, you can refer to SDSL-Lite
`https://github.com/simongog/sdsl-lite`

Compression as a classification/clustering tool

- Compression to measure the information distance:

$$NCD(x, y) = \frac{|C(xy)| - min\{|C(x)|, |C(y)|\}}{max\{|C(x)|, |C(y)|\}}$$

- If two sequence are syntactically close, the compression size of their concatenation is expected to be close to their individual compression sizes.
  Cilibrasi and Vitanyi, "The Google Similarity Distance," IEEE Trans. on Knowledge & Data Engineering, 2007.
- Can make sense in biological data classification and clustering processes as well.
  Cilibrasi and Vitanyi, "Clustering by compression", IEEE Transactions on Information Theory, 51(4),2005 See `complearn.org`

Section 5

Future Research Avenues

Privacy/security Aspects

In the bag?
If so, where ?

What if you don't want others to see what is inside the zipped bag? Use a non-transparent bag! If so, how would you search items in the non-transparent bags?

Privacy-preserving

- ...data compression
- ...pattern matching
- ...text–indexing

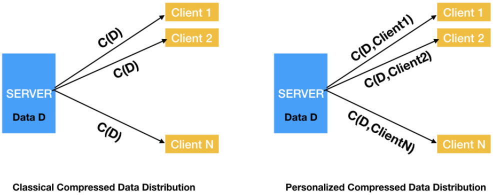
Privacy/security aspects of text processing algorithms and data structures, e.g., secure BWT/suffix array/suffix tree, etc...

Interdisciplinary research between text algorithms, security/privacy, and bioinformatics communities for management of highly sensitive personal biological data !

! Contact me (kulekci@itu.edu.tr) if you are interested !

Personalized Data Compression

- A data server doesn't care who is asking for the data, but only what the requested data is.

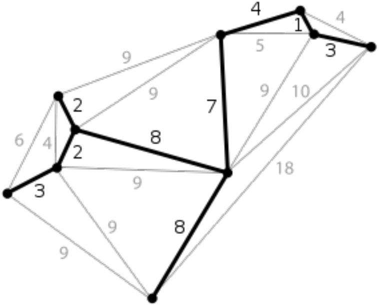


- However, subsequent downloads of an individual from the repositories are expected to be highly correlated!
- Can we improve the transmission via reference-based compression schemes (a.k.a. compression by side information).
- A data file compressed differently per each individual! Pros and cons?

! Contact me (kulekci@itu.edu.tr) if you are interested !

Compressed File Nets

- Better compression due to the existence of a similar reference file. *Kuruppu, et al., "Relative Lempel-Ziv compression of genomes for large-scale storage and retrieval." SPIRE'10*
- FASTQ, VCF, SAM/BAM, and etc. files are highly similar
- **Create a graph** $G(V, E)$ **such that vertices denote the files and the edge from** v_i **to** v_j **represents their similarity.**
- Find minimum spanning tree of this graph and compress the files with reference to their ancestors.
- Sample study on the VCF file repository of 1000 Genomes project ?



! Contact me (kulekci@itu.edu.tr) if you are interested !

Final Remarks

- When high-throughput sequencing becomes a daily practice in medicine, we will experience real data explosion.
- Disruptive solutions to enhance the S-o-A, probably with *acceptable assumptions*
- Privacy/security aspects, referential coding and relative data structures for enhanced data management
- Compression aiming not only to improve space efficiency, but also to improve operational capacity on compressed data

Final Remarks

- When high-throughput sequencing becomes a daily practice in medicine, we will experience real data explosion.
- Disruptive solutions to enhance the S-o-A, probably with *acceptable assumptions*
- Privacy/security aspects, referential coding and relative data structures for enhanced data management
- Compression aiming not only to improve space efficiency, but also to improve operational capacity on compressed data

WE UNDERSTAND THE DATA
AS MUCH AS
WE CAN COMPRESS IT!

Thanks !