tgggattatagttatgagccactatgctcggccagtgtgtctgaattctgggcatctgtg
tggccagctgaacctcagggcattccagtactgaagggaggaggggaggatggccactga
gggctaggtagggatctctgcctcactgcgccgcaggtccaggtggccagtgggctccag
agagaagggtaggggtgatggctgcttcatctccttttctcttatatccaccccagatct
cgacatggcaggagctgaggcagctgcaggagcagatccggagcctggaggaagagaagg
cagctgtgactgaggcagtgcgggccctgctggtgagcatggtgccctgagtaggtgggg

From a Myriad of Short Strings to Biological Discovery

Broňa Brejová

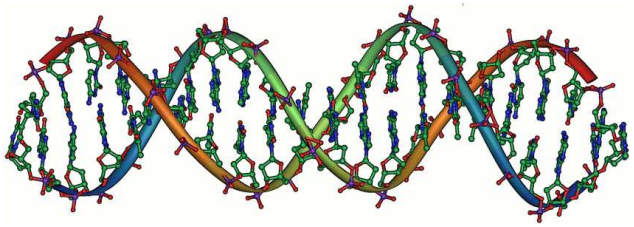
Katedra informatiky FMFI UK

brejova@fmph.uniba.sk

taccggcaacttcactgcactgatctttcctcagtttcccatcagggtttccagatggct
gctgtagtccagcatctcctcacatgactgtgccaggggaaggagacagagacttctctg
gcttgagttccttttcaggagcaagtacacctttcctgaagcatccagcaaagtcccct
cgtgtcccctgggcccagacccccaccacactccagttgtcagccggaggaatggagttc
cggcagccggcataggctgatcagaggccatcctccacctgggcccacttcccctgaact
acaggctaccagggaagggtgaactgcgtcagagttctgtggggaaggaggaagaaggga
tctttattttgttttttatttttttgagaccgagtctcactctgttacccaggctggagtg
cagtggcatgatctgggctcactgcaccctctgccccctgggttcaagtgattcttctgc
ctcagcctccccagtagctgggattacagacatgcgccaccacgcccggctaattttgta

Basics: genome, DNA

Genome: genetic information stored in each cell as molecules of DNA

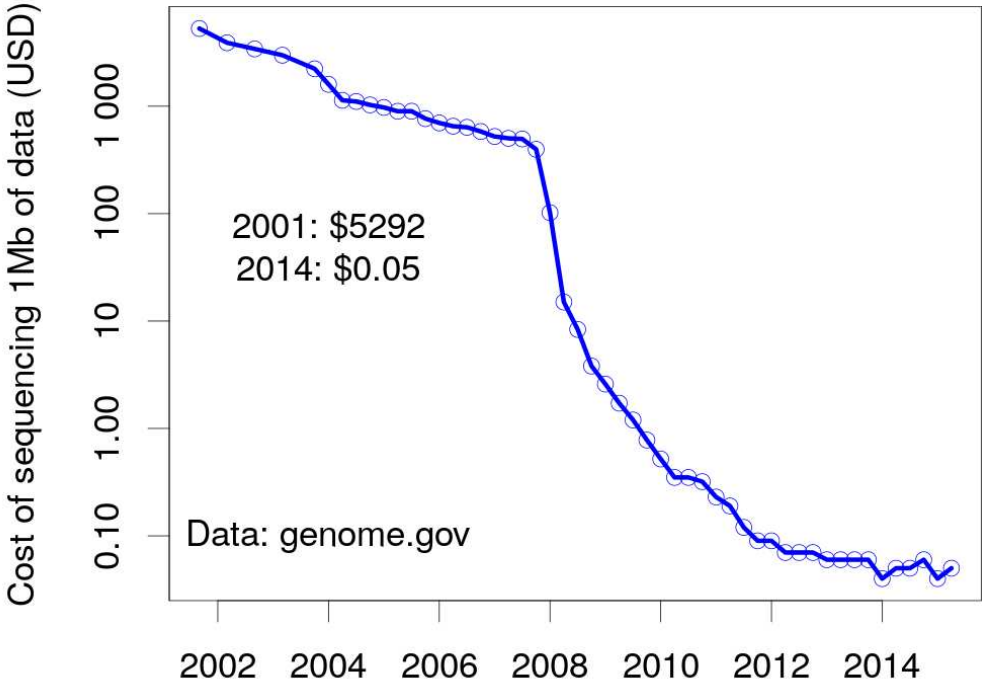


Human genome: Text with 3 billions of letters A, C, G, T

```
TCACTTGAACCCAGGAGGCGGAGGTTGCAGGAGCCAAGATCATGCCACTG
CACTCCAGCCTGGGCAACAGAATGAGACTCCATCTTAAAAAAAAAAAAAA
AAAAAAAAGGTAAAGACCAACACAAAAACCTCAGACTTCCTATACAAGTT
AAGCAGAGGAACCCGGAAAATGTCTGAAGAAAACGATTACCAATTTTTTT
TGTTTTTTTTGTTTTTTGGTGACCGGGTCTCACTCTGTTGCCCAGGTGGG
AGTGCAGTGGTGCGATCACAGCTTCCTGCAGCCTTGACTTCCTGGGCTCA
AGTGATCCTCCCATCTCAGCATCCTGGGTAGGTGGGACCATAGGTGTGCG
CCACCACGACTGGCTAATTTTTTGTATTTTTAGTAGAGATGGAGTTTTGC
CATGTCACCCAGGCTGGTCTTGAACTCCTGGGCTCAAGCAATCCTCTTGC
CTCAGCCTCCTAAAGTGTTGGGATTACAGGTGTGAGCCAGCGCACCCAGC
CACACATTATTTTAAAATTTGTTTACATAATACAAAAATAAGTATTTTGC
CCCAAATTTTCTCTTTAAAAATAT...
```

Genome sequencing

- Human genome: 3 billions of A, C, G, T
- Human genome sequencing: 1988-2004, 3 billions dollars
- Today can be done in several days, costs thousands of dollars
- Next generation sequencing: starting in 2004, still developing



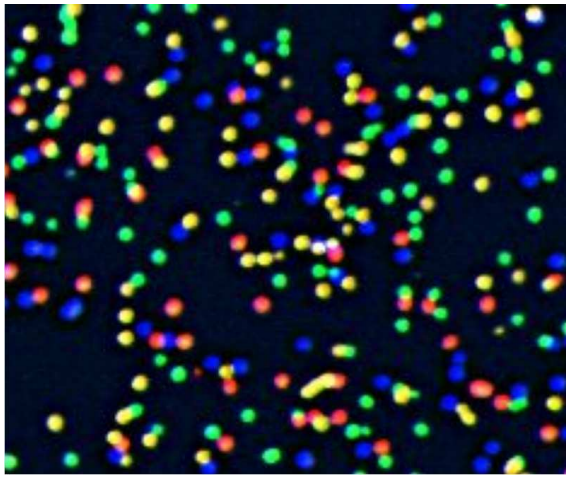
Next generation sequencing (NGS)

ABI Sanger sequencer in 2001:

96 reads of length 500 in 10 hours, 115kb/day

Illumina HiSeq 3000 in 2015:

$4 \cdot 10^9$ reads of length 150 in 3 days, 200Gb/day



Many parallel sequencing reactions in a tiny area

Next generation sequencing (NGS)

Applications of DNA sequencing

- Sequencing genomes of animals, plants, fungi, bacteria, viruses, . . .
- Sequencing many individuals of the same species, cancer tumors, . . .
- Interrogating cell processes

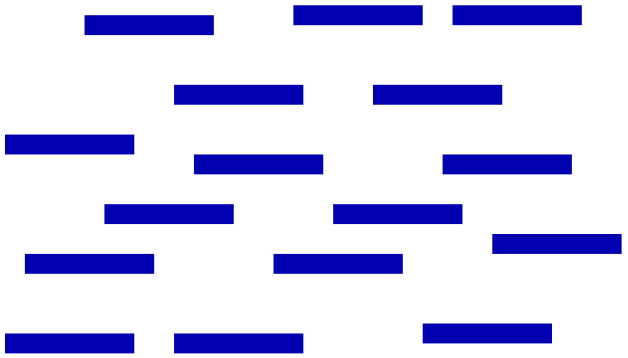
Challenges for computer science

- Sequencing machines read only short fragments of DNA (**reads**)
- Large volumes of data require fast algorithms

Genome assembly

Sequencing produces short reads from random locations in DNA

Position of individual reads on the target DNA is not known



Goal of genome assembly: reconstruct original genome from reads

Simple but unrealistic formulation

Shortest common superstring problem.

We are given several strings $S_1, \ldots, S_k$ (reads),

find the shortest string S containing each S_i as a (contiguous) substring

Motivation: to use overlaps between reads as much as possible

Example:

Input: GCCAAC,CCTGCC,ACCTTC

Output: CCTGCCAACCTTC (reads connected in order S_2, S_1, S_3)

NP hard problem [Gallant et al. 1980]

no known polynomial-time algorithm can find optimal answer for each input

Heuristics for Shortest Common Superstring

- Repeatedly find and connect two reads with longest overlap
- Example: CATATAT, TATATA, ATATATC

  Optimum: CATATATATC, length 10

  Heuristics: CATATATCTATATA, length 14
- This heuristics is an approximation algorithm:

  It finds a string which is at most $3.5\times$ longer than optimal superstring

  [Kaplan and Shafrir 2005]
- **Conjecture:** it is in fact a 2-approximation algorithm
- There is a different 2.48-approximation algorithm [Mucha 2013]
- Related Overlap-Layout-Consensus approach used in practice

  [Batzoglou et al. 2002]

Real data are more complex

Factors not considered in Shortest Common Superstring:

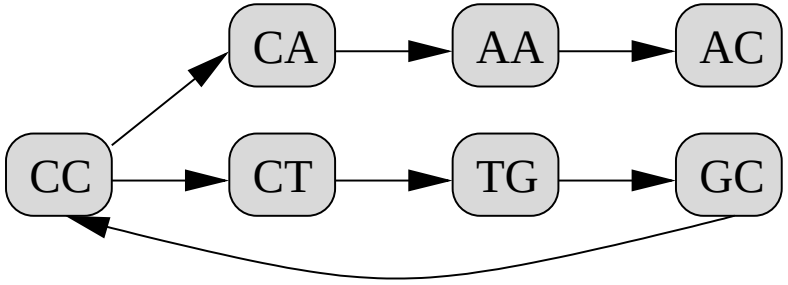
- Sequencing errors
- Sequence repeats
- Multiple chromosomes
- Other types of experimental data

Simplification:

- We connect only parts bridged by multiple reads
- The result is a set of contigs
- Conservative approach: sacrifice completeness for accuracy

Assembling NGS data: de Bruijn graphs [Pevzner et al. 2001]

- Split reads to overlapping windows of length k
- **Vertices:** substrings of length k from all reads
- **Directed edges:** connect k-mers consecutive in at least one read (overlapping by $k-1$ bases)
- **Example:** $k=2$, reads: CCTGCC, GCCAAC



- **Ideal situation:** single chromosome, no repeats, no errors

  $\longrightarrow$ get a single path of vertices

Assembly using de Bruijn graphs [Zerbino & Birney 2008]

- Create graph from input reads (data structures)
- Remove parts likely corresponding to errors (low coverage)
- Paths without branching will be contigs
- Possibly connect contigs using information from
  - original reads
  - additional longer reads with high error rate
  - approximate distances between pairs of reads
- Practical heuristics, typically not nicely formulated

Our approach based on probabilistic models

- Probabilistic model of sequencing [Godsi et al. 2013]
- If read r occurs n_r times in a genome A
  $\Pr(r|A) = n_r/|A|$
- Also add random independent errors
  $\Pr(r|A) = \sum_{i=1}^{|A|-|r|+1} \Pr(A_{i,\ldots,i+|r|-1} \text{ read as } r)/|A|$
- Individual reads $R = (r_1, \ldots, r_k)$ independent
  $\Pr(R|A) = \prod_{i=1}^{k} \Pr(r_i|A)$
- Our goal: find assembly A maximizing likelihood $\Pr(R|A)$

[Boža, B., Vinař, WABI 2014, AMB 2015]

GAML: genome assembly by maximum likelihood

- Our goal: find assembly A maximizing likelihood $\Pr(R|A)$
- Complex optimization problem
- Create de Bruijn graph by existing method, look for a set of walks
- Simulated annealing with local assembly changes
- Efficient data structures to evaluate likelihood fast after small assembly changes
- **Main advantage:** probabilistic model can incorporate many different technologies and data sources by changing $\Pr(A_{i,\ldots,i+|r|-1}$ read as $r)$

[Boža, B., Vinař, WABI 2014, AMB 2015]

Summary (I)

- DNA sequencing have become cheap and fast
- It produces many short reads which need to be assembled
- Theoretical formulation: shortest common superstring
- Many practical tools, interesting data structure and engineering issues
- Probabilistic models capture sequencing process and allow us to formulate the problem with complicated real-life factors

Next: can we sometimes avoid assembly altogether?

Assembly-free genome analysis

- Genome assembly is complicated process,

  may introduces biases and artefacts,

  also computationally intensive

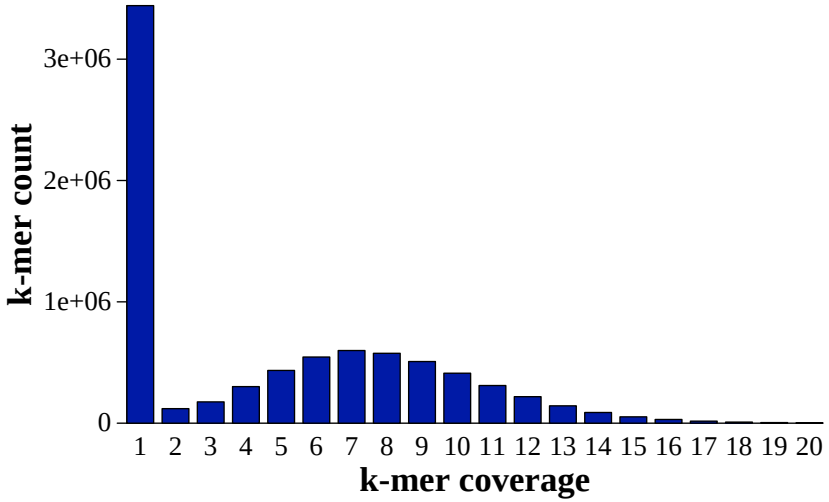
- Our goal: estimate genome size and coverage from reads directly

- Faster computation, simpler model, works at lower coverage

[Hozza, Vinař, B., SPIRE 2015]

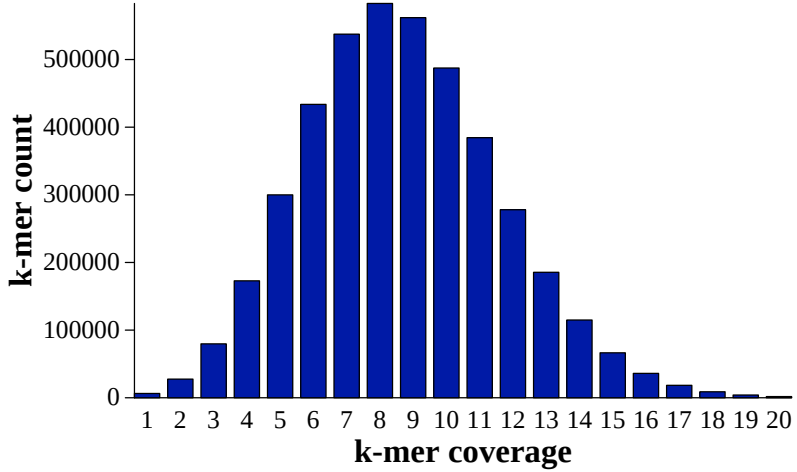
k-mer abundance spectrum

- For each k-mer, count its occurrences in the set of reads
- Summarize in a histogram of k-mer abundances using existing efficient software, $k = 21$
- Infer coverage from spectrum using probabilistic models
- Unlike previous work, explicitly model sequencing errors [Li and Waterman 2003, Williams et al 2013]

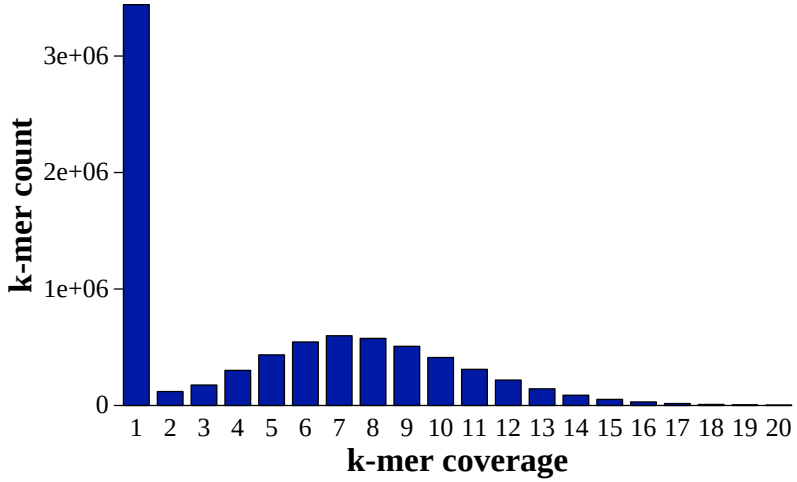


Histogram for $10\times$ coverage of Escherichia coli

Under ideal circumstances we expect truncated Poisson distribution:



Real histogram (many unique k-mers due to errors):

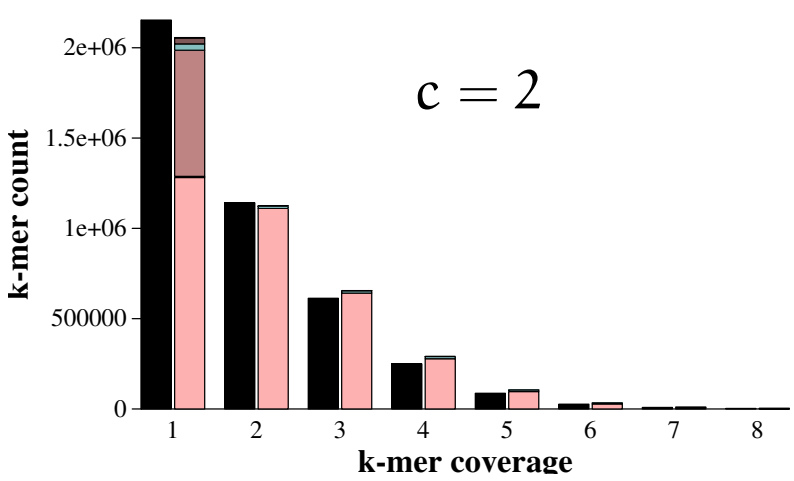


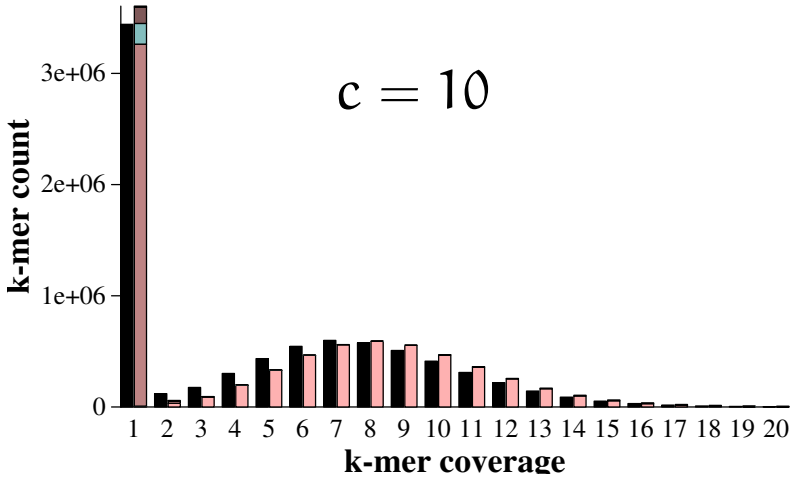
Our model works even for low coverage

Real E. coli genome size 4.64 Mbp

Comparison to KSA [Williams et al. 2013]

Method	$c = 0.5$	$c = 1$	$c = 2$	$c = 4$	$c = 10$	$c = 30$	$c = 50$
Ours	4.16	4.70	4.58	4.63	4.71	4.69	4.68
KSA	N/A	N/A	N/A	6.03	4.61	4.59	4.58





Summary (II)

- Some questions can be answered directly from reads, without assembly
- We have considered genome size and coverage

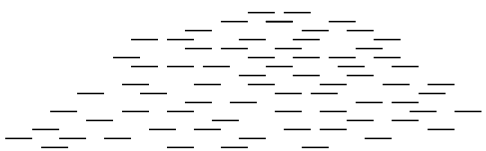
  Future work: estimate the number of occurrences of a repeat
- Again natural application of probabilistic models
- Many extensions of the model possible

Next: sometimes we sequence genomes for which assembly is already known

Read mapping

Example of NGS application: sequencing individuals

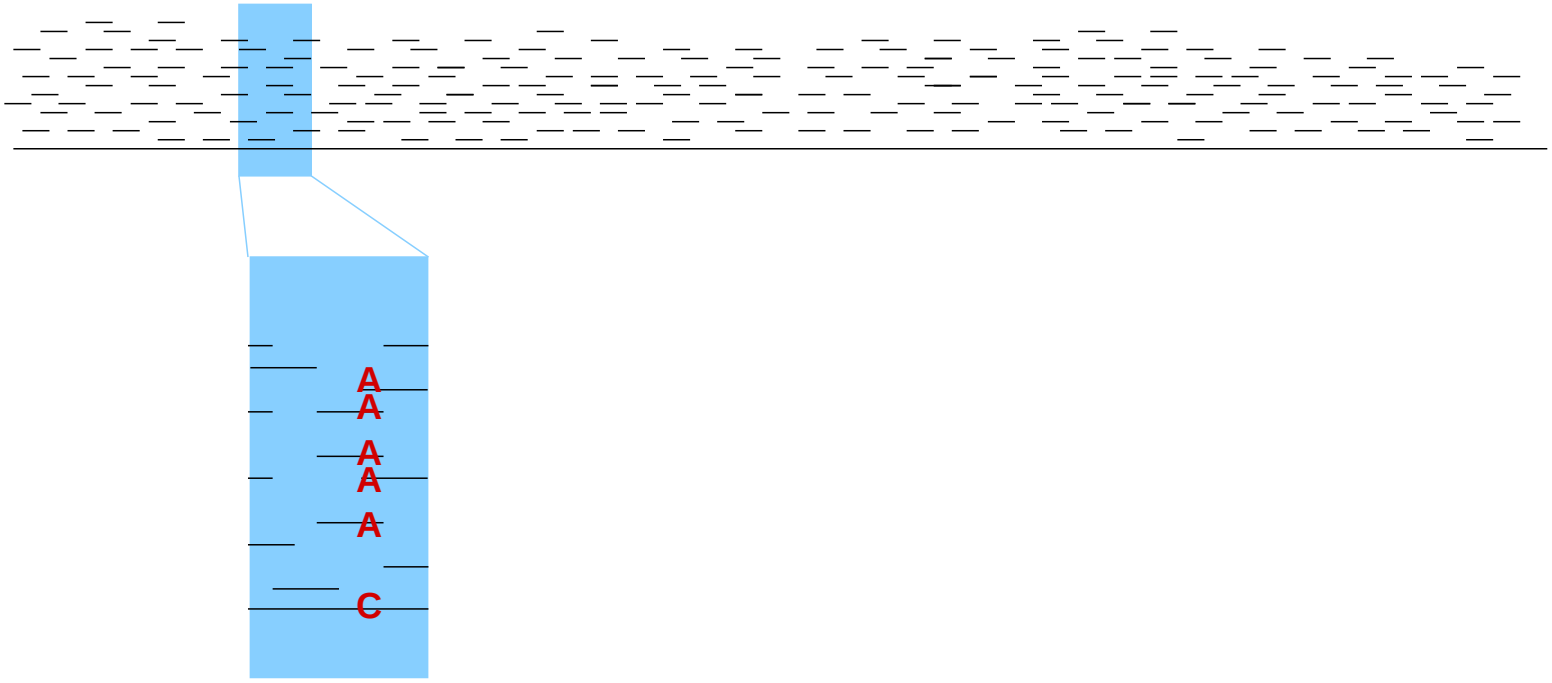
300M reads of length 100



Reference genome (3Gbp)

Read mapping

Example of NGS application: sequencing individuals



Map reads,

then find differences between individual's and reference genomes

Sequence alignment

```
GAAGAGAAGGCAGCTGTGACTGAGGCAGTGCGGGCCCTGCTG----GTGAGCAT
|.|||||||..||||||.|.||||||||||||||||||||    ||||||||
GCAGAGAAGGAGGCTGTGGCAGAGGCAGTGCGGGCCCTGCTGGTGAGTGAGCAT
```

- Aligning sequences = finding their similar regions
- One of basic tools in bioinformatics
- Read mapping a special case
- Best alignment can be found by dynamic programming
- Dynamic programming too slow for read mapping
- Heuristic filtering: find promising parts of genomes

  apply dynamic programming only there

Heuristic filtering

- For example BLAST [Altschul et al. 1990]
- Find all exact matches of length w (**seeds**)
- Use slower algorithms on regions around seeds
- Exact matches can be found fast
- Some good alignments do not contain a seed, those are not found

Small w: many random seed hits, slow

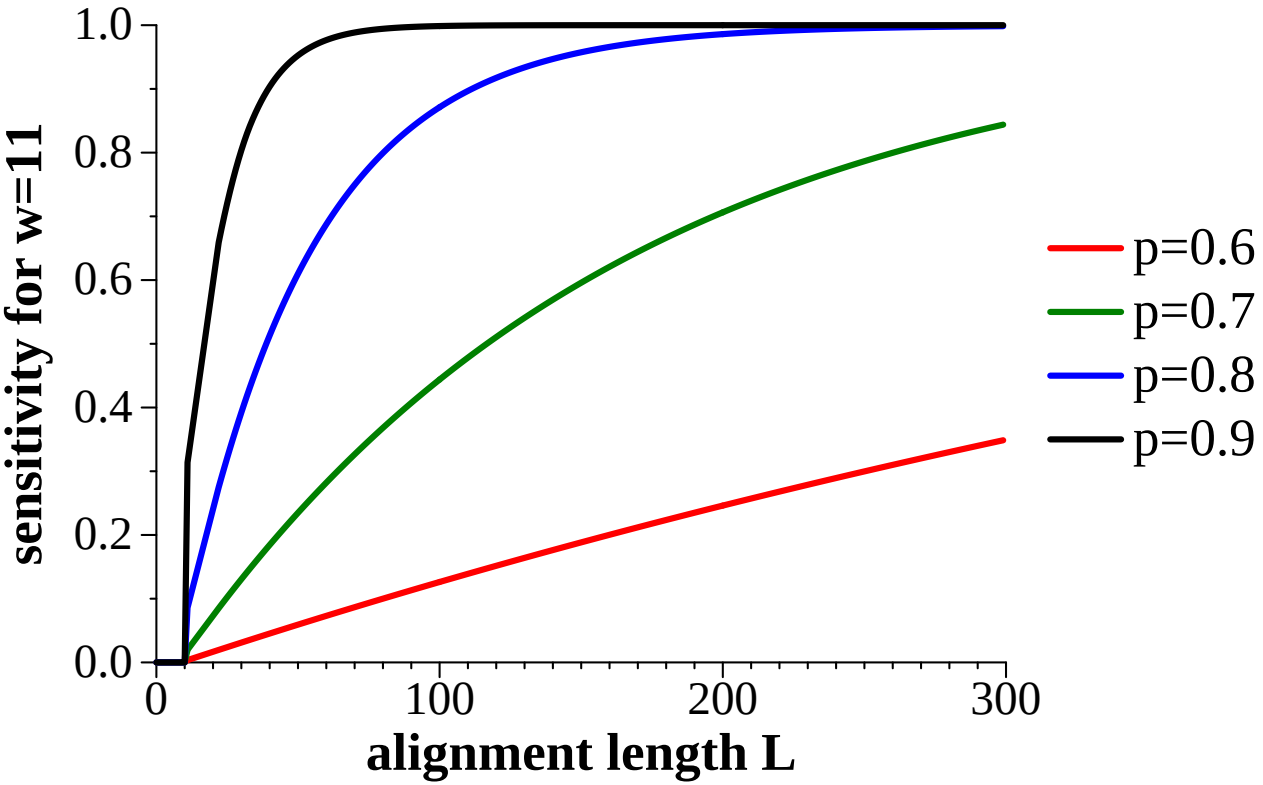
Large w: many alignments not found, but fast

Estimating sensitivity

Assume random alignment of length L

Every position match with probability p

Sensitivity $f(L, p) = \Pr(\text{alignment contains } w \text{ consecutive matches})$



Spaced seeds [Ma, Tromp, Li 2002]

Spaced seed: required configuration of matches in a hit

Example:

"match—match—don't care—match" denoted as `1101`

```
GTGGTGCTCTCTGACAAAGCC
  |  ||  ||    |||  ||||
ATTGTTCTTAATGAGAAAGAA
   1101      1101
                1101
```

BLAST: 11 consecutive matches

equivalent to seed `11111111111`

Spaced seed `111010010100110111`

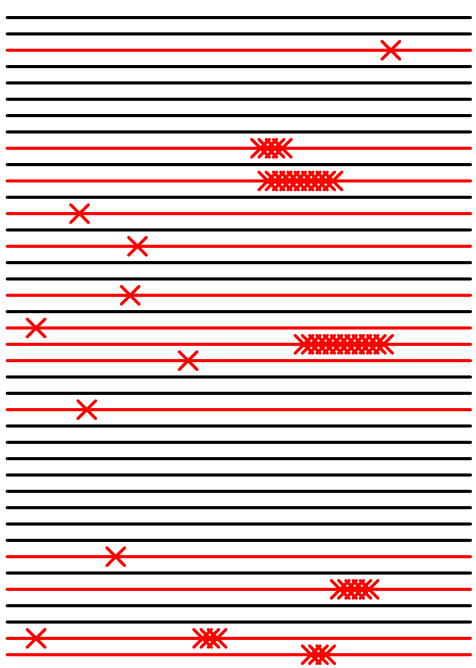
also 11 matches, but more sensitive

Why are spaced seeds more sensitive?

Assume: alignment length 64,

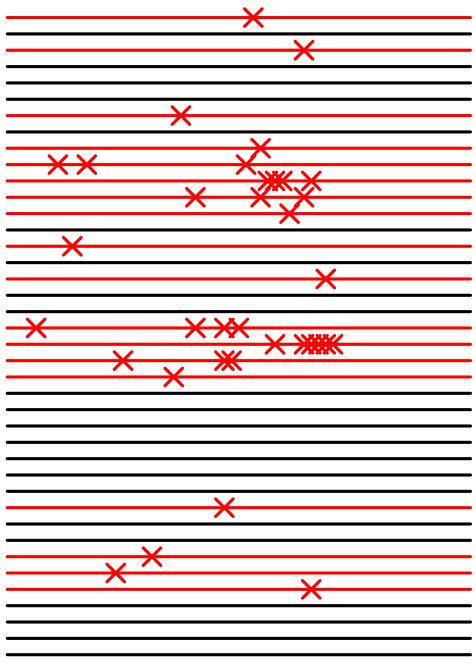
probability of match 0.7 at every position

Generate 40 random alignments, find seed hits



11111111111

Sn.: 14/40, hits: 46



1110100101001101111

Sn.: 18/40, hits: 35

Why are spaced seeds more sensitive?

Assume: alignment length 64, probability of match 0.7 at every position

Consecutive matches	**Spaced seed**
`11111111111`	`1110100101001101111`
Expected number of hits in alignment:	
$54 \cdot 0.7^{11} = 1.1$	$47 \cdot 0.7^{11} = 0.9$
Probability of hit at position $i+1$ if hit at position i:	
0.7	$0.7^6 = 0.12$
`1111111111` `1111111111`	`1110100101001101111` `1110100101001101111`
Hits clustered together	Hits more “independent”
Sensitivity (probability of at least one hit):	
0.30	0.47

Alignment models do matter

Inside genes, each amino acid encoded by a triple (codon)

Probability of match varies within codon:

Position within codon:	first	second	third
Probability of a match:	0.67	0.77	0.40

Sensitivity on a testing set of protein coding sequences:

Seed		Human vs. Drosophila	Human vs. mouse
Optimal for data	**110 110 000 110 110 11**	86%	92%
Opt. for codon model	**110 110 010 110 010 11**	86%	91%
WABA [Kent, Zahler 2000]	**110 110 110 110 11**	80%	90%
Optimal for i.i.d. model	**1110010010010111**	60%	86%
BLAST	**1111111111**	43%	81%
Worst	**1010101010101011**	39%	79%

[B., Brown, Vinař 2004]

Summary (III)

- Sequence alignment aims to find similar sequences
- Used for read mapping, genome comparison, ...
- Exact algorithms too slow, prefiltering often used
- Spaced seeds better than contiguous
- Optimize seeds wrt realistic alignment models
- Also many interesting combinatorial problems
  (e.g. connection to Golomb rulers)

Next:

Genome assembly and sequence alignment are basic infrastructure.

What about real biological discovery?

Genome sequencing projects

- Potentially interesting genome selected for sequencing
- Sequencing followed by genome assembly, annotation, comparison to related genomes, search for interesting features
- In big projects, consortium of experts in different areas
- Today, genome sequencing feasible for a small team,
  but analysis remains a challenge

Genome of common marmoset (Callithrix jacchus)

- Consortium of 110 authors from 8 countries,

  led by Baylor College of Medicine and Washington University St. Louis

- Tiny primate, 250g, 18cm

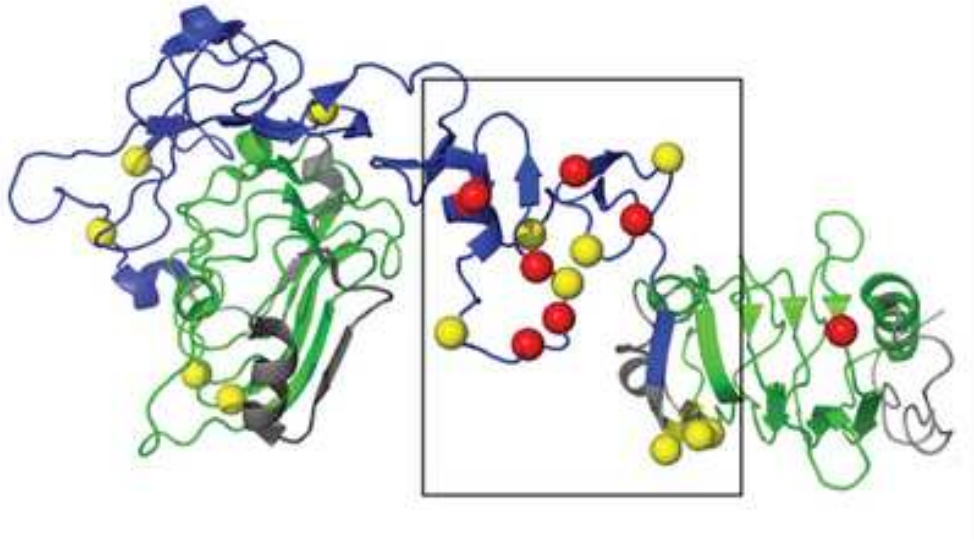
- Which genetic changes are related to the small size?

[Nature Genetics 2014]

IGF1R: Insulin-like growth factor 1 receptor

Responsible for cell growth and survival

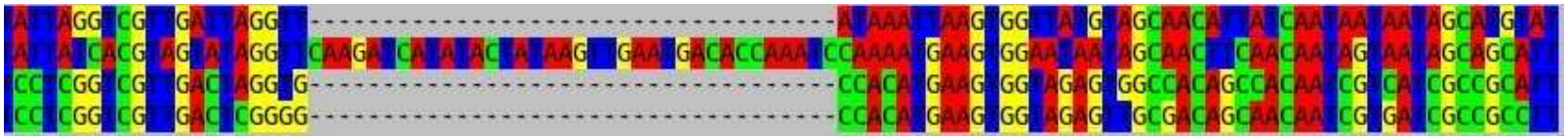
```
human     RDFCANILSAESSDSEGFVIHDGECMQECPSGFIRNGSQSMYCIPCEGPCPKVC-EEEKKTK
chimp     RDFCANILSAESSDSEGFVIHDGECMQECPSGFIRNGSQSMYCIPCEGPCPKVC-EEEKKTK
orang     RDFCANILSAESSDSEGFVIHDGECMQECPSGFIRNGSQSMYCIPCEGPCPKVC-EEEKKTK
macaque   RDFCANILSAESSDSEGFVIHDGECMQECPSGFIRNGSQSMYCIPCEGPCPKVC-EEEKKTK
marmoset  RQFCASIVSSENSENNKFVIHDGECMQDCPSGFIRDTTHSMQCIPCKGPCPKVC-D-EQMAK
mouse     RDFCANIPNAESSDSDGFVIHDDECMQECPSGFIRNSTQSMYCIPCEGPCPKVCGDEEKKTK
rat       RDFCANIPNAESSDSDGFVIHDGECMQECPSGFIRNSTQSMYCIPCEGPCPKVCGDEEKKTK
dog       RDFCANIPSAESSDSEGFVIHDGECMQECPSGFIRNGSQSMYCIPCEGPCPKVC-EEEKKTK
```



[Nature Genetics 2014]

Small genome, big surprise

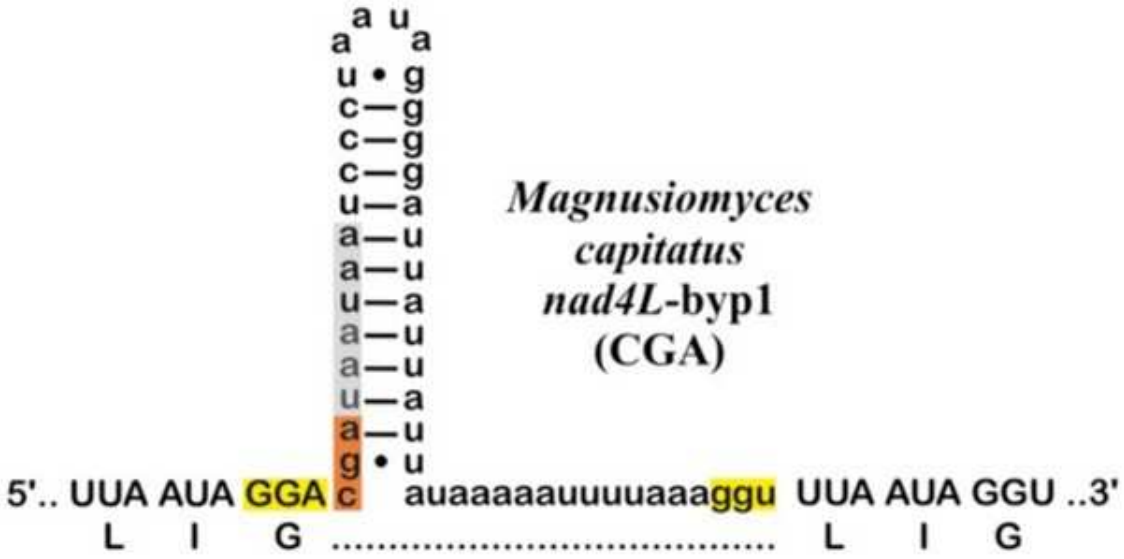
- Collaboration with the group of J. Nosek from Faculty of Natural Sciences, later joined by a group from U. of Montreal.
- Mitochondrial genome of yeast *Magnusiomyces capitatus*, length 43kb
- Many genes interrupted by insertions, overall 81 insertions of length 30-55bp



[Lang et al, PNAS 2014]

Ribosome hopping

- Experiments confirmed that these insertions are skipped when the gene is translated to protein
- Previously only one occurrence of such a phenomenon was known, in bacteriophage T4 [Huang et al 1988]



Different aspects of bioinformatics

- **Engineering:** make better (faster, more accurate, easy to use) tools for analyzing biological data
- **Biology:** find interesting phenomena in real data using these tools
- **CS theory/math:** study underlying problems