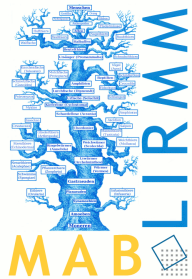
High Throughput Sequencing & bioinformatics analysis

Eric Rivals

LIRMM & IBC, Montpellier

http://www.lirmm.fr/~rivals



High Throughput Sequencing or Next Generation Sequencing

Human Genome Project

1988 - 2004

16 years et 3 billion $

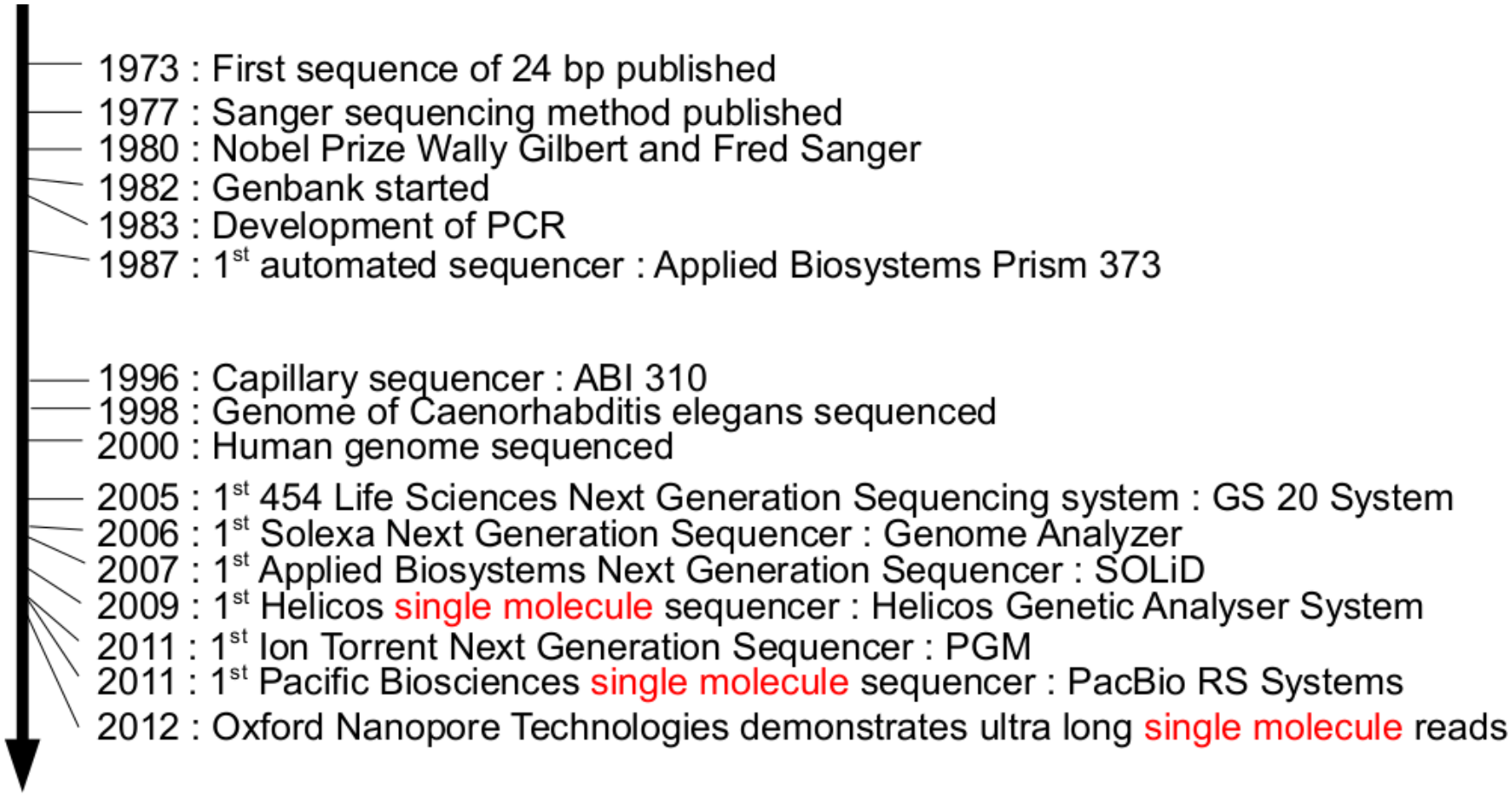
1 dollar per base

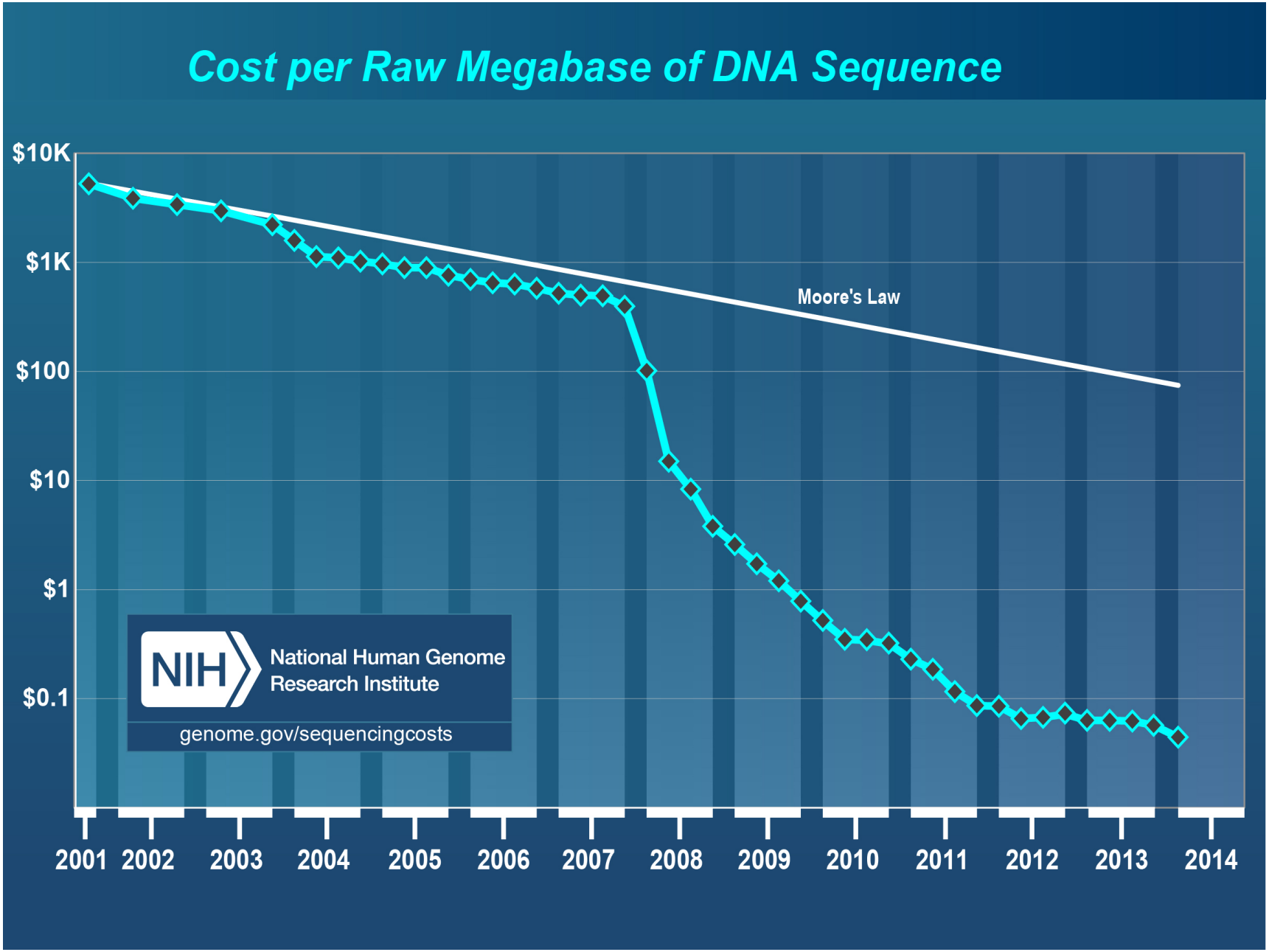
Sequencing macromolecules

Only the chemical bases are variable along the DNA helix
4 possible bases

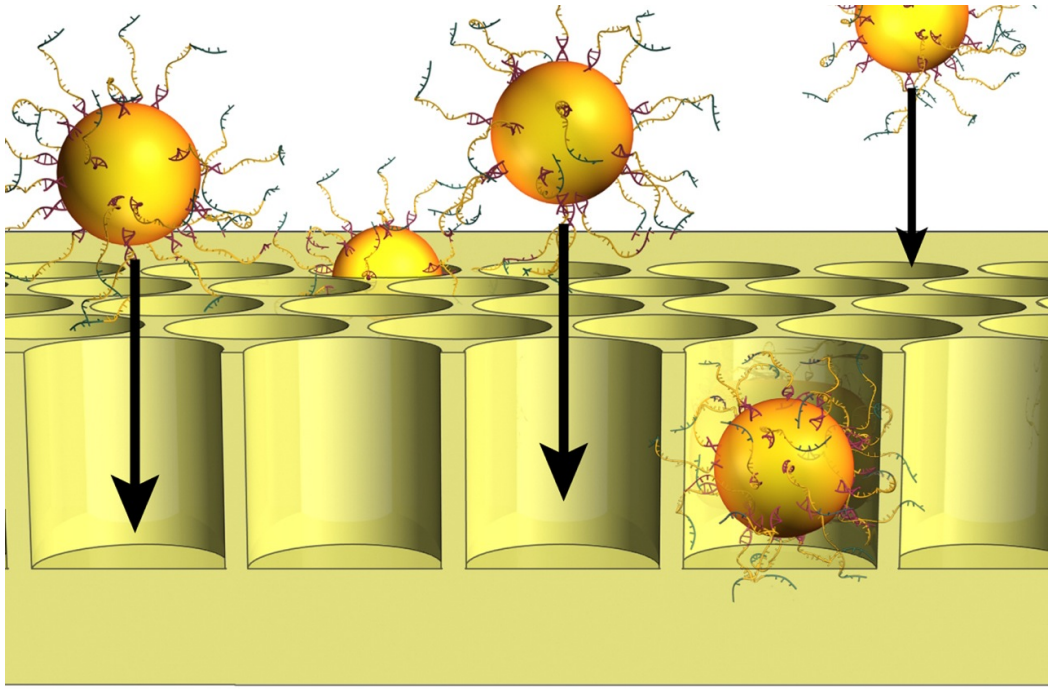
Sequencing: getting the sequence of bases along the molecule
for DNA or RNA molecules

Sequencing: technological evolution




Cost per Raw Megabase of DNA Sequence
$10K
$1K
$100
$10
$1
$0.1
Moore's Law
NIH
National Human Genome Research Institute
genome.gov/sequencingcosts
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014

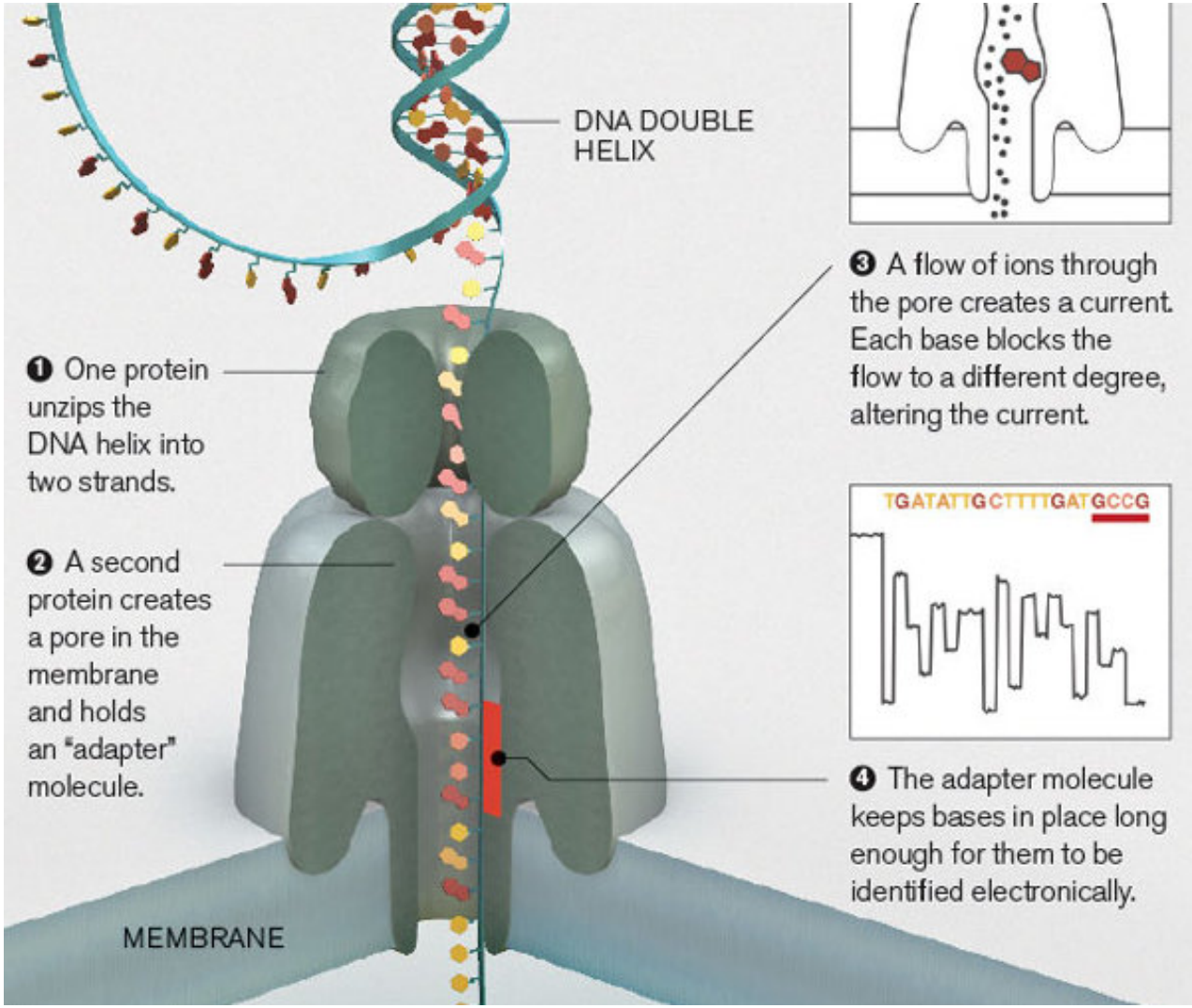
Optical and parallel sequencing



©Roche - Life Sciences

requires an exponential multiplication (amplification) of DNA/RNA molecules $\Rightarrow$ bias

Single Molecule Sequencing



©Oxford Nanopore

Overview of techniques

Name	Read Lg	Time	Gb/run	pros / cons
454 GS Flex	700	23 h	0.7	long
Illumina HiSeq	2*100	48 h	120	short/cost
SOLID (LifeSc)	85	8 d	150	long time
Ion Proton	200	2 h	100	new
PacBio Sciences	3-15000	0.3	3	high error rate

HTS output: an example

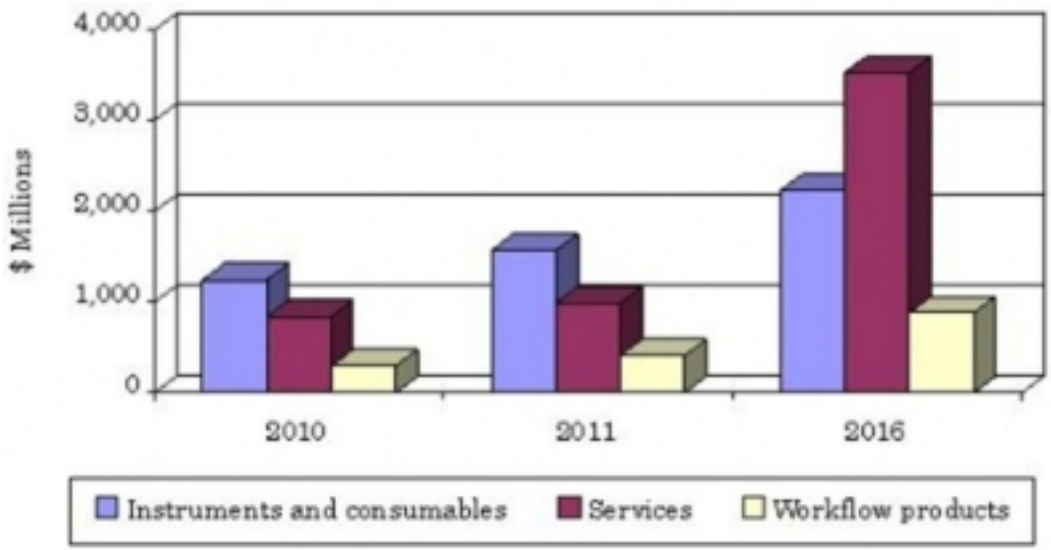
one Human RNA library
75 million reads
of 100 bp each
representing $> 140,000$ splice events on $16,000$ expressed genes

HTS output: an example

one Human RNA library
75 million reads
of 100 bp each
representing $> 140,000$ splice events on $16,000$ expressed genes

Bottleneck: Bioinformatics read analysis

Sequencing Market



Source: http://www.prnewswire.com/

What can life scientists do with NGS assays?

Domains of applications

- bio-molecular research
- biotechnology (e.g. bio-fuels)
- biodiversity monitoring
- personalised medicine
- epidemiology surveillance
- pharmacogenomics
- personal genomics
- forensic
- agronomy (animal and plant research)

NGS change ...

- the scale,
- sensitivity,
- accessibility and
- cost of sequence based assays.

Major economical and social impact

For instance, pharmacogenomics and personalized medicine impact on social security costs

Biological questions

- sample genomic variations in a population of individuals
- detect genotype differences related to a disease
- measure variations in gene expression & identify RNA variants
- study replication, transcription or translation processes
- interrogate protein binding sites on the whole genome or RNAs
- assess epigenetic modifications on the genome (3D structure)
- estimate the fitness contribution of each gene in bacteria
- identify genes involved in pathogenicity or adaptation
- study gene interactions and their role in regulatory or metabolic pathways
- survey the species or assess the biodiversity of an environment
- list the bio-molecular functions or processes active in an environmental sample

Diverse applications of High Throughput Sequencing

Investigate key biological questions at genome scale, with huge depth

- Genome sequencing or resequencing, Exome sequencing
- Transcriptomics: find known or novel RNAs in a cell line or tissue
  RNA-Seq, DGE
- Variation identification: SNPs, somatic mutations, Copy Number Variations (CNVs), genomic rearrangements
- Epigenomics: locate on DNA protein binding or chromatin modification sites
  ChIP-seq
- Mutagenesis sequencing: test phenotypes
- Meta-genomics: survey biodiversity in a sample

Diverse applications of High Throughput Sequencing

Investigate key biological questions at genome scale, with huge depth

- Genome sequencing or resequencing, Exome sequencing
- Transcriptomics: find known or novel RNAs in a cell line or tissue
  RNA-Seq, DGE
- Variation identification: SNPs, somatic mutations, Copy Number Variations (CNVs), genomic rearrangements
- Epigenomics: locate on DNA protein binding or chromatin modification sites
  ChIP-seq
- Mutagenesis sequencing: test phenotypes
- Meta-genomics: survey biodiversity in a sample

Bottleneck: Bioinformatics read analysis

Remarks on seq based assays

- This type of questions and assays pre-existed to NGS
- but NGS made them cheaper, high-throughput, and genome-wide
- Genome wide is the major qualitative change:

  no predefined target, no knowledge required,

  potentially all sites are scrutinized

Remarks on seq based assays

- This type of questions and assays pre-existed to NGS
- but NGS made them cheaper, high-throughput, and genome-wide
- Genome wide is the major qualitative change:

  no predefined target, no knowledge required,

  potentially all sites are scrutinized

- Nonetheless, biological assays and NGS suffer from limitations
  e.g. many species cannot be "grown" in the lab, hence difficult to get DNA

Computational tasks

- Mapping, multiple pattern matching
- Finding sequence similarities, clustering, mappability
- Text Indexing
- Error correction
- Splicing junctions and mutations detection
- Genetic haplotype inference
- Statistical estimation of measure like gene expression levels

Personalized Medicine

Personalised Medicine

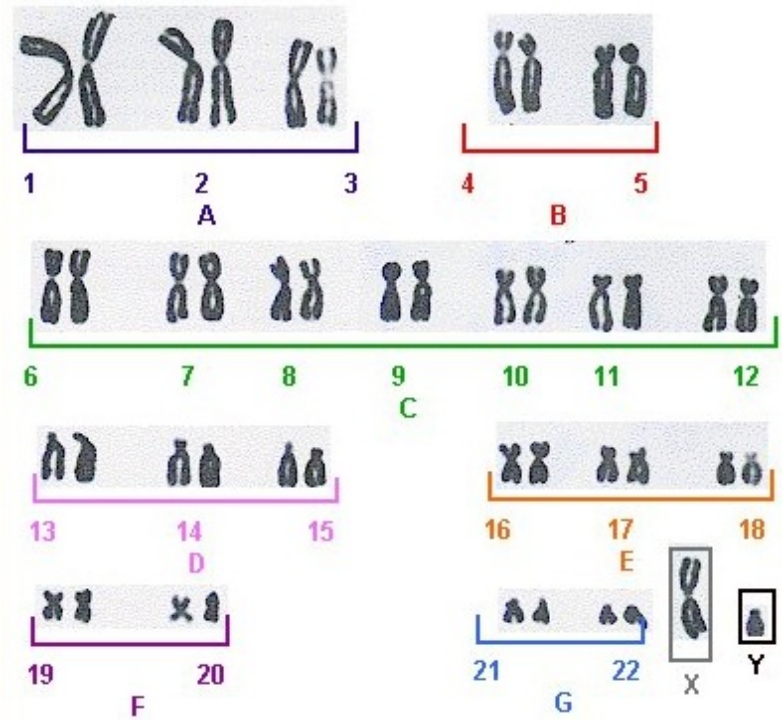
Wikipedia

emphasizes the systematic use of information about an individual patient to select or optimize that patient's preventative and therapeutic care.

US Congress definition

the application of genomic and molecular data to better target the delivery of health care, facilitate the discovery and clinical testing of new products, and help determine a person's predisposition to a particular disease or condition.

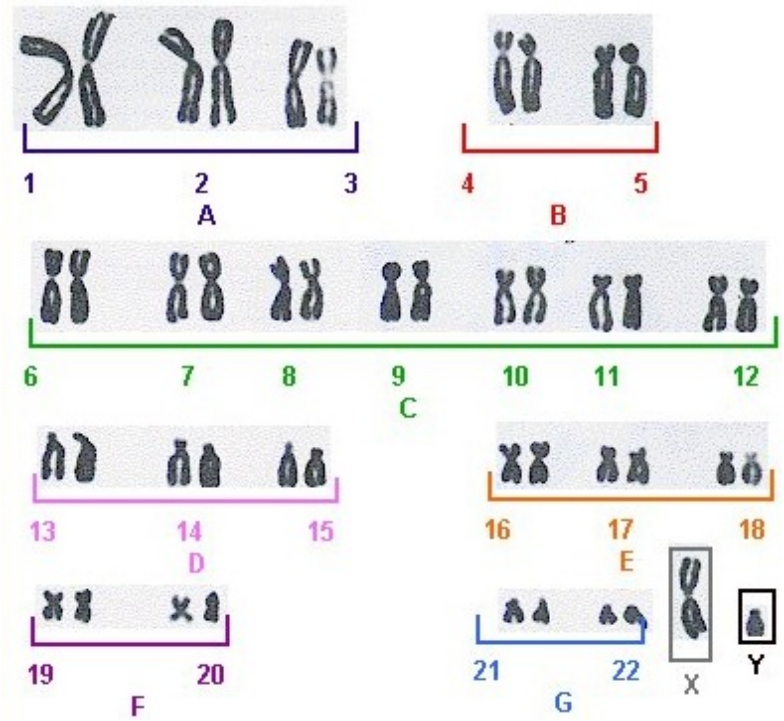
Abnormal chromosome pool in cancer



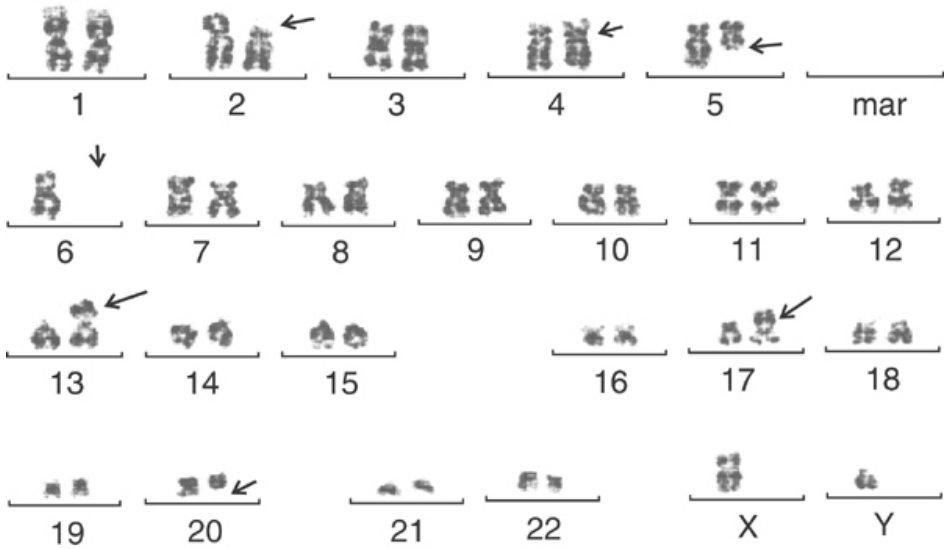
Normal human karyotype

Blood cancer karyotype (leukemia)

Abnormal chromosome pool in cancer



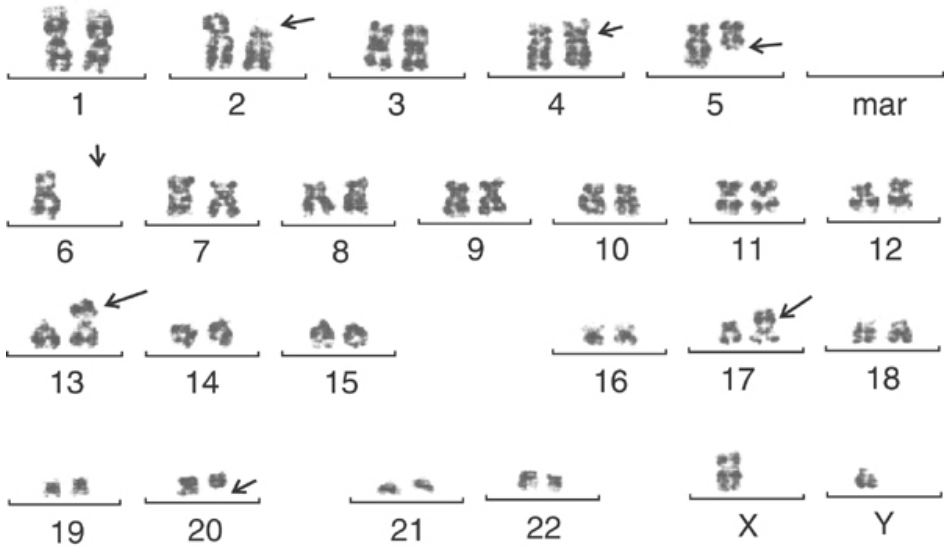
Normal human karyotype



Blood cancer karyotype (leukemia)

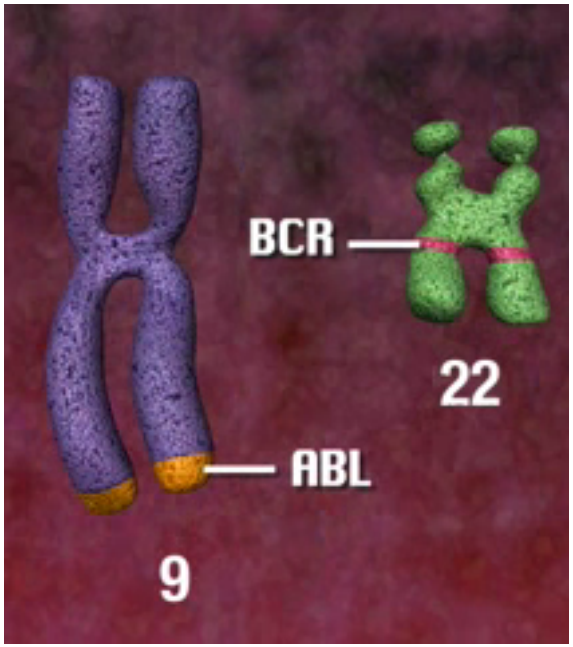
Abnormal chromosome pool in cancer

- Diagnosis of chronic myelogenous leukemia (CML)
- Prognosis in myelodysplastic syndrome

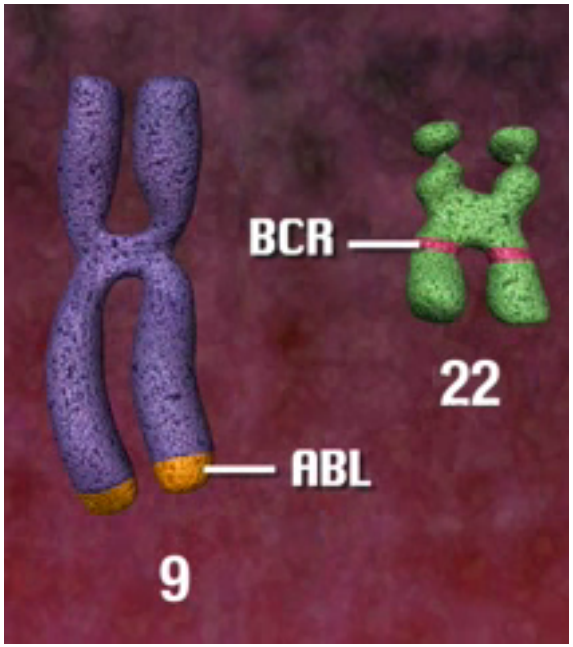


Blood cancer karyotype (leukemia)

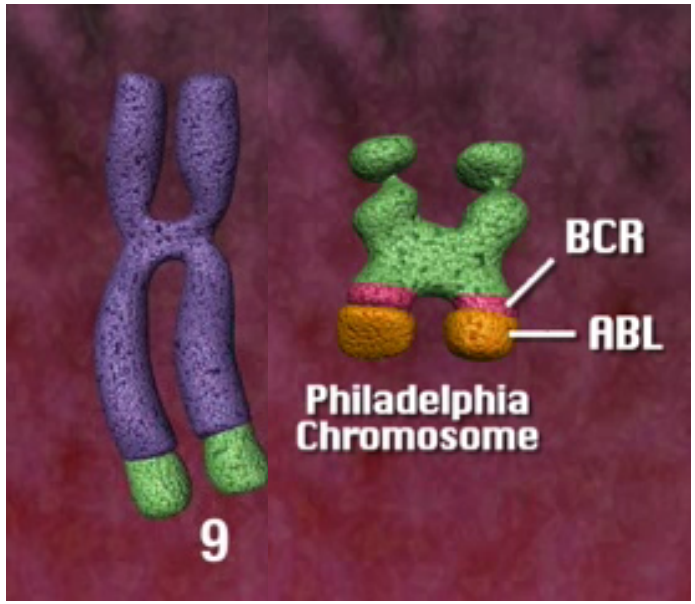
Leukemia with gene fusion



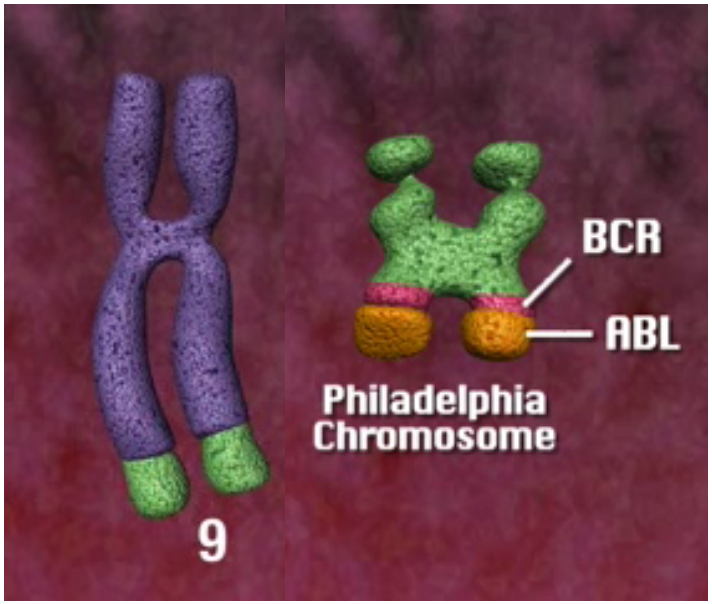
Leukemia with gene fusion

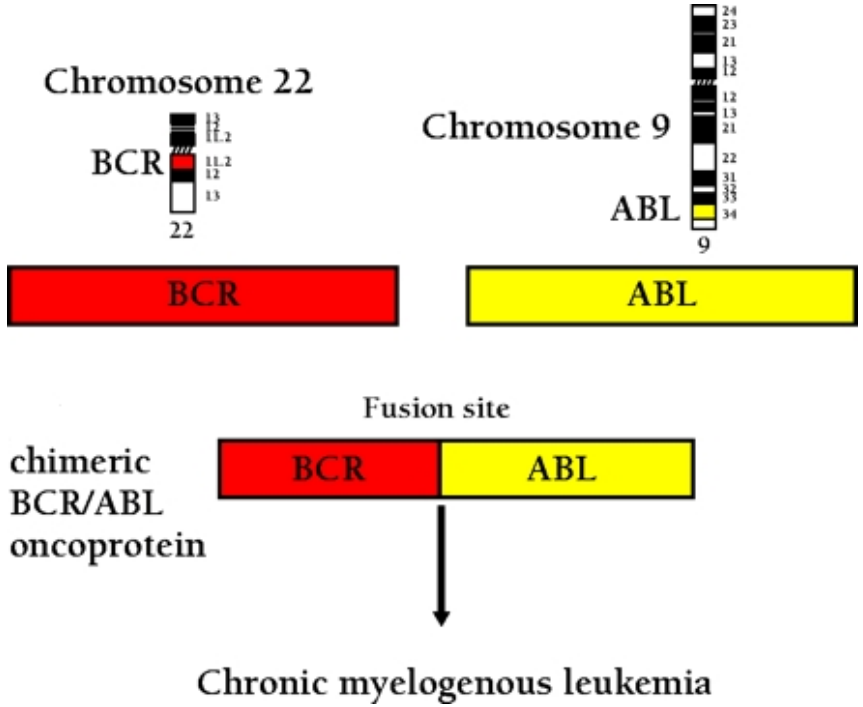


translocation



Translocated gene to fusion RNA





Personalised Medicine for Chronic Myelogenous Leukemia

Test in the bone marrow: presence of BCR-ABL t(9;22) fusion RNA?

1. diagnosis
2. monitoring disease recurrence
3. treatment follow up

Personalised Medicine for Chronic Myelogenous Leukemia

Test in the bone marrow: presence of BCR-ABL t(9;22) fusion RNA?

1. diagnosis
2. monitoring disease recurrence
3. treatment follow up

What if

- test goes wrong because of human genetic variability?
- another form of BCR-ABL fusion is produced?
- other, still unknow, aberrant RNA are involved in this cancer?

What we need ...

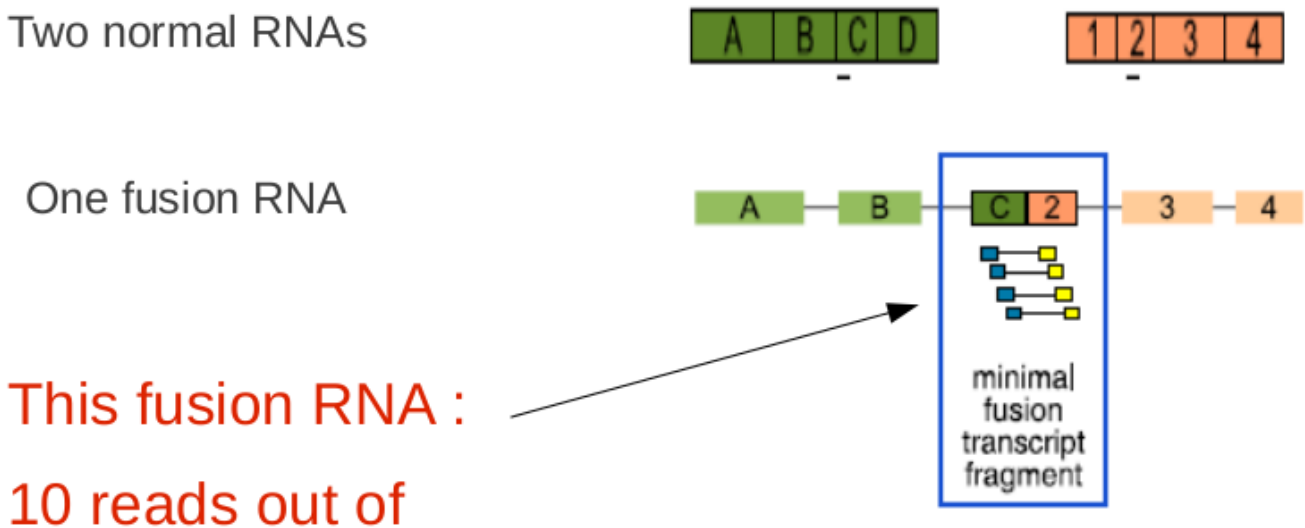
- Monitoring all active genes, *i.e.* RNAs, in a cell
- very fast
- at low cost, and
- limited cell material.

High-Throughput Transcriptomics

RNA-seq

determining which genomic regions are transcribed and activated in a cell, at which activation/expression level

Detection needs sensitivity and specificity



Mutations, transcriptomics and metagenomics

Mutations in cancer (Leukemia [Mardis et al NJEM, 2010])

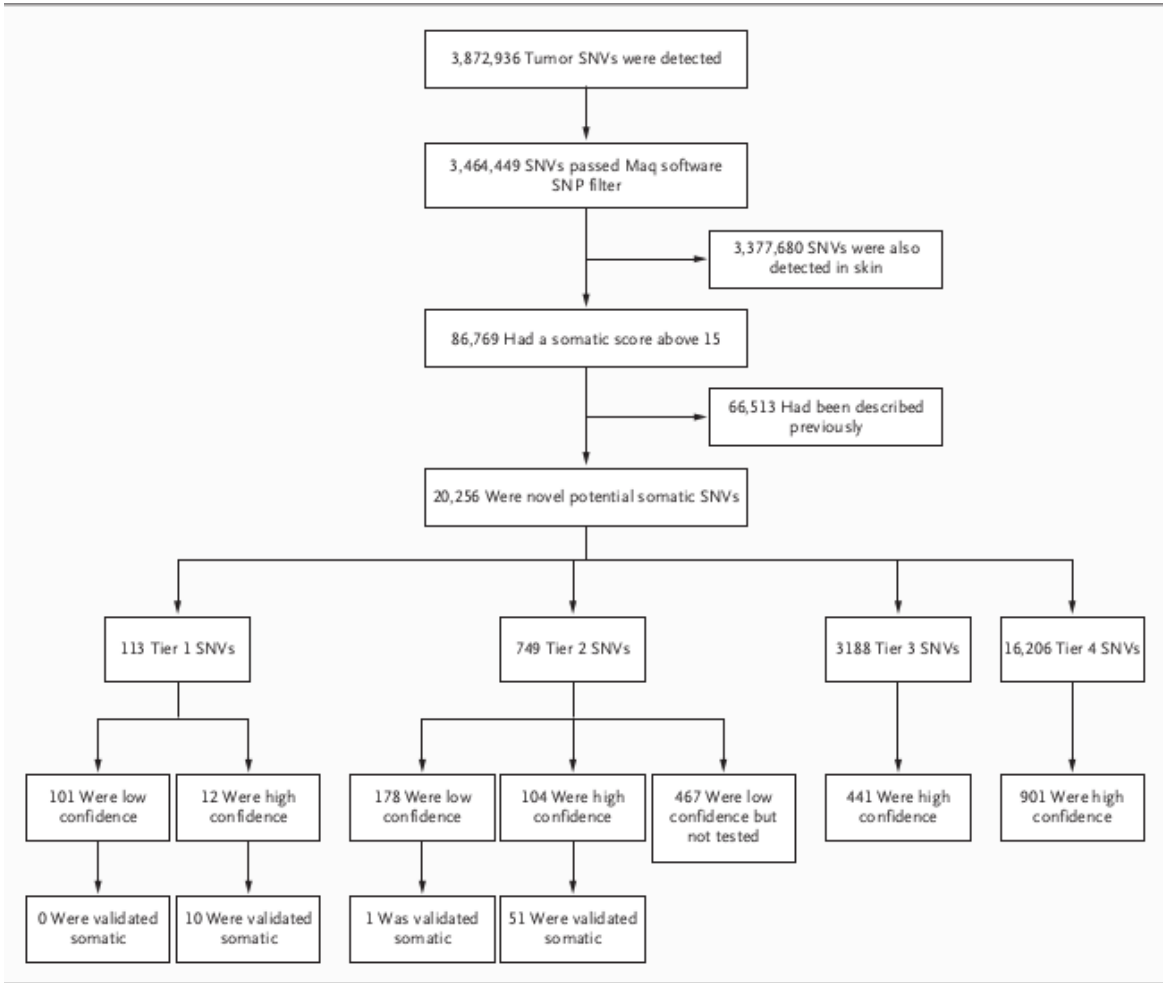
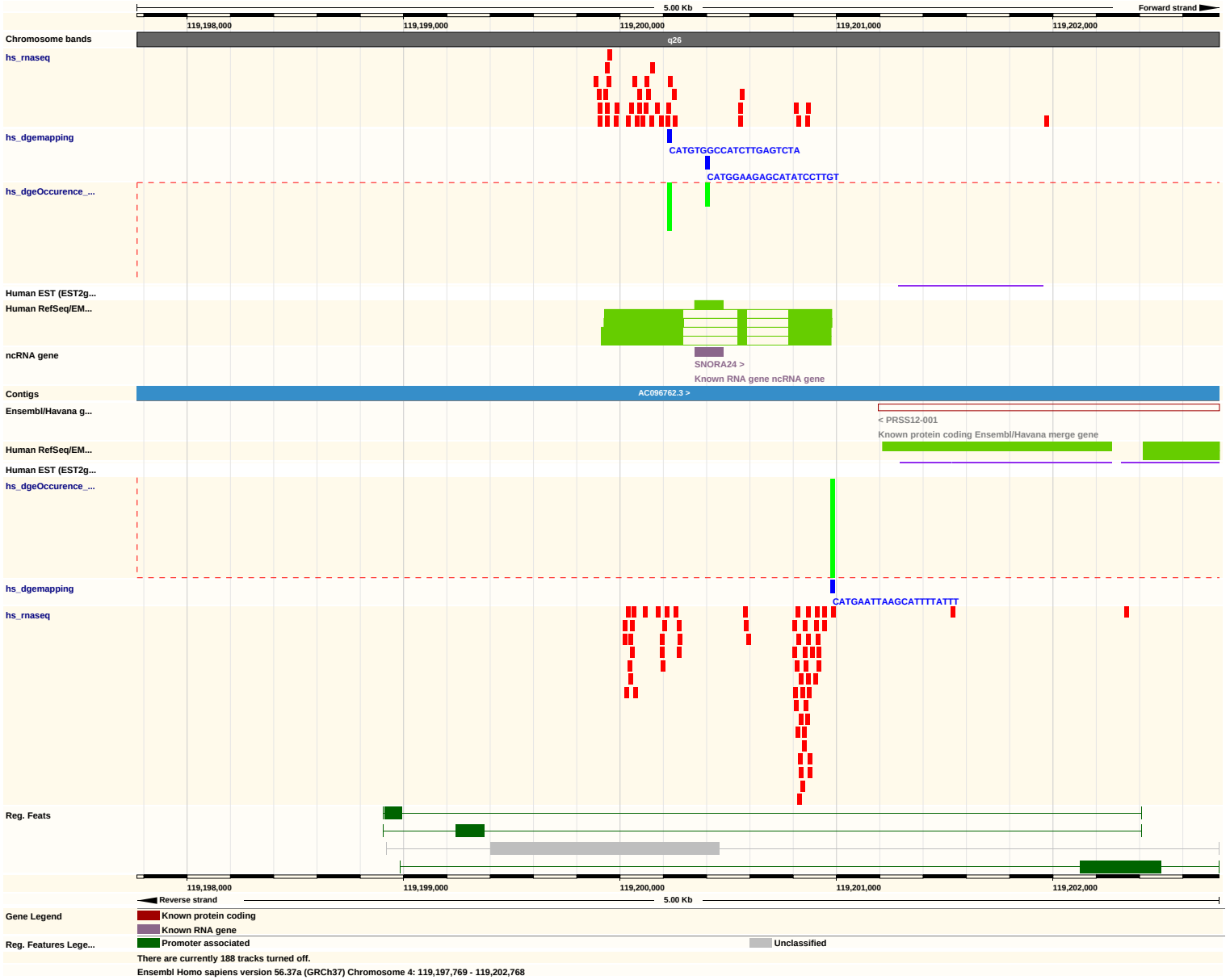


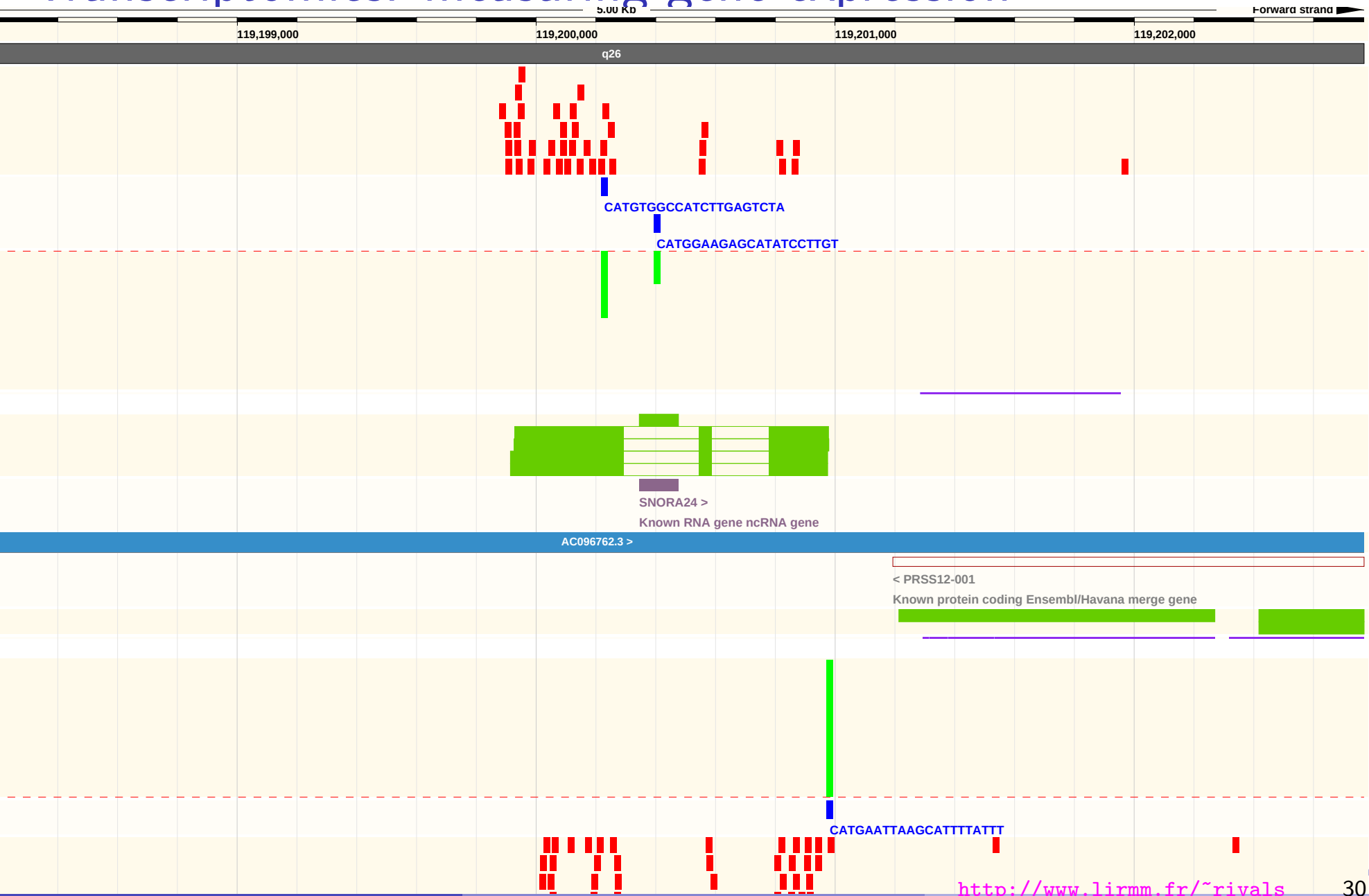
Figure 1. Flow Chart for Identification of Somatic Point Mutations in the Acute Myeloid Leukemia Genome.
Maq denotes Mapping and Assembly with Quality, SNP single-nucleotide polymorphism, and SNV single-nucleotide variant.

From 64 billion bp sequenced, and 45,000 high quality mutations.

Transcriptomics: measuring gene expression



Transcriptomics: measuring gene expression



Applications in "meta-genomics"

Why meta?

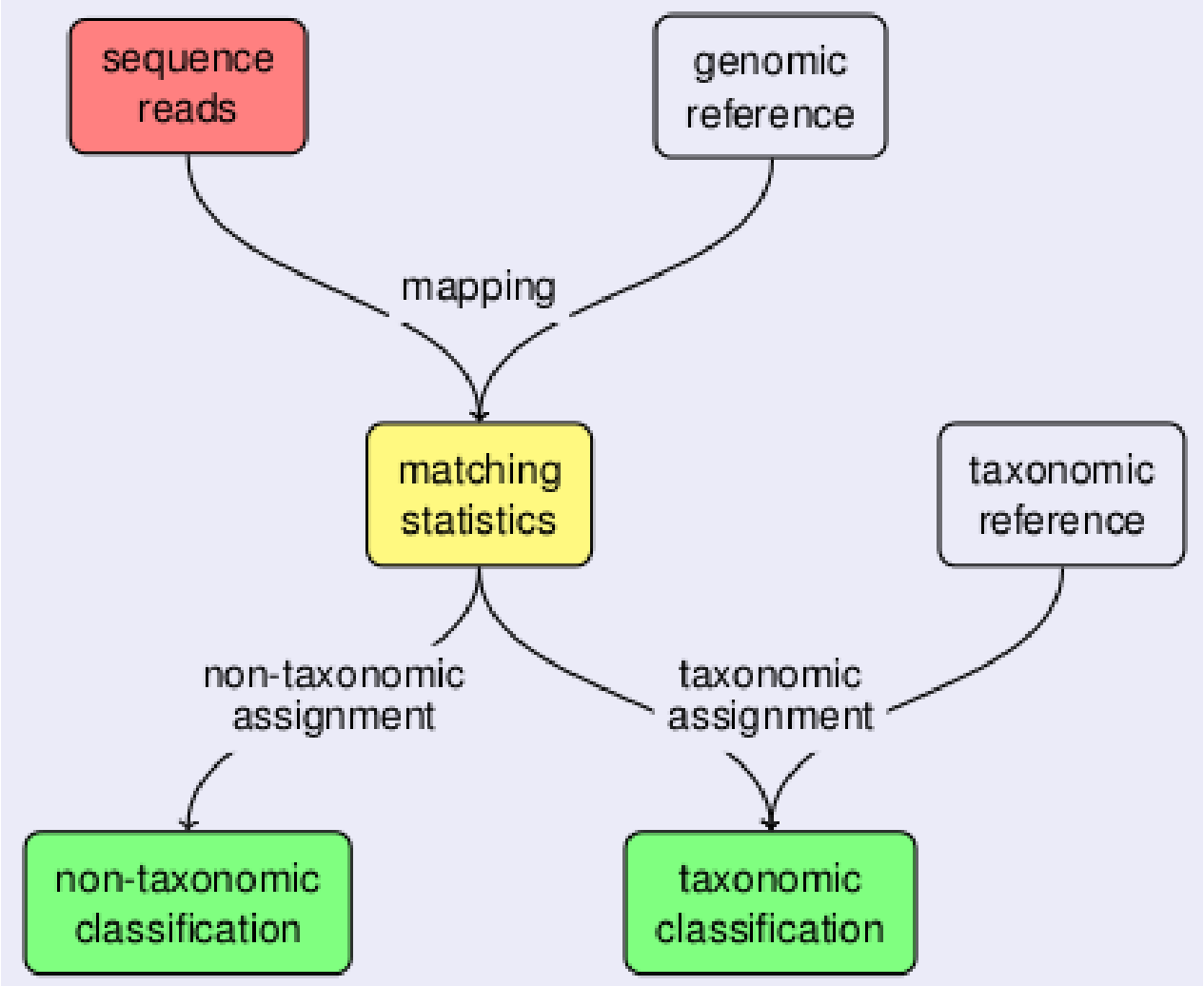
In an ecosystem, species are rarely isolated.

In an environmental sample, one finds numerous DNAs/RNAs of distinct origins, which reflect the species diversity and their interactions.

With meta-genomics, one sequence a set, mixture of various genomes.

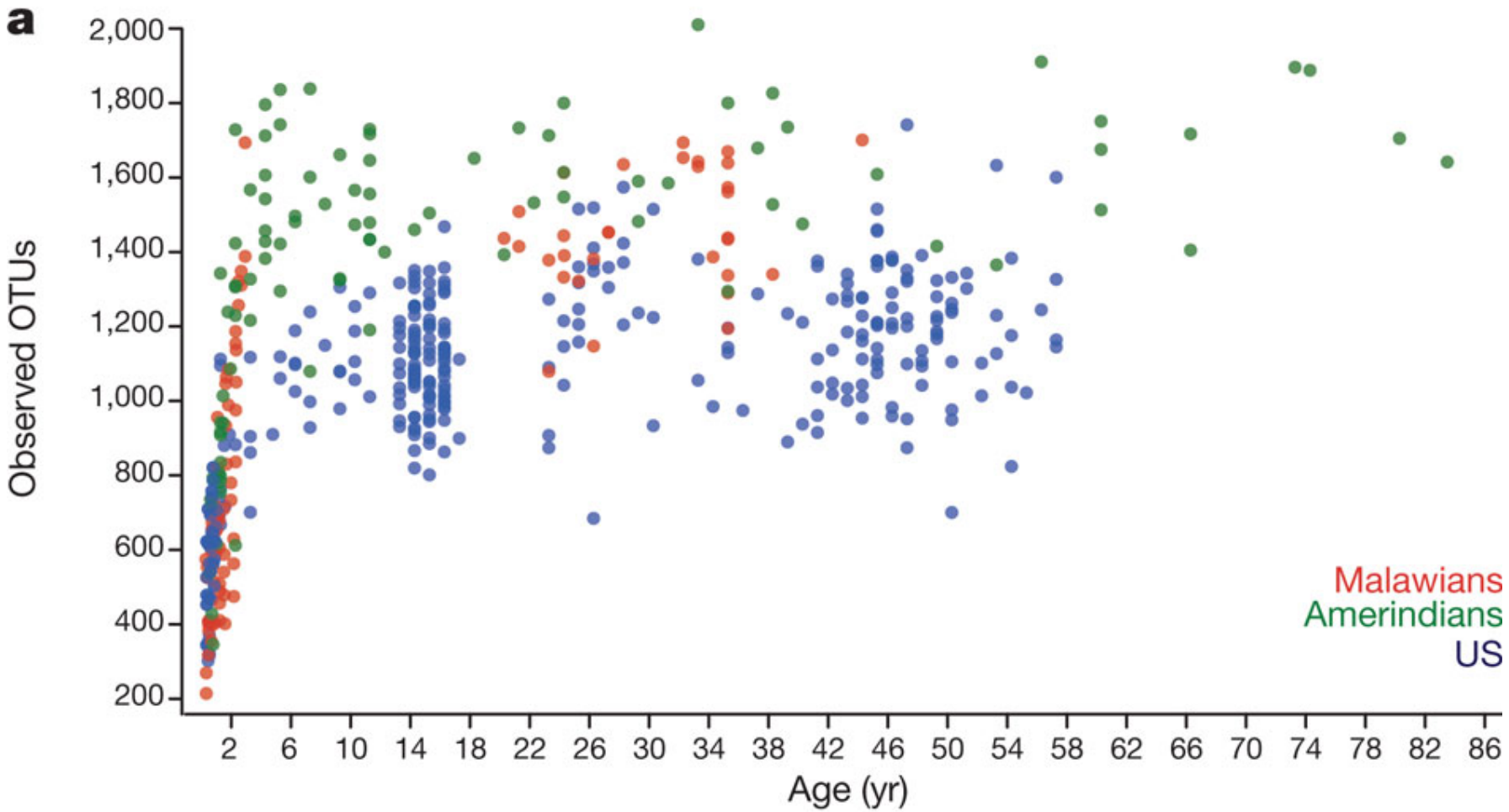
- Biodiversity: identify the species living in a sample
  mines, ocean, gut, etc.
- or their biochemical activities and interactions
- Ex: Project TARA oceans http://oceans.taraexpeditions.org

Taxonomic assignment of metagenomics reads



Pipeline for reads assignment to species [from G. Valiente]

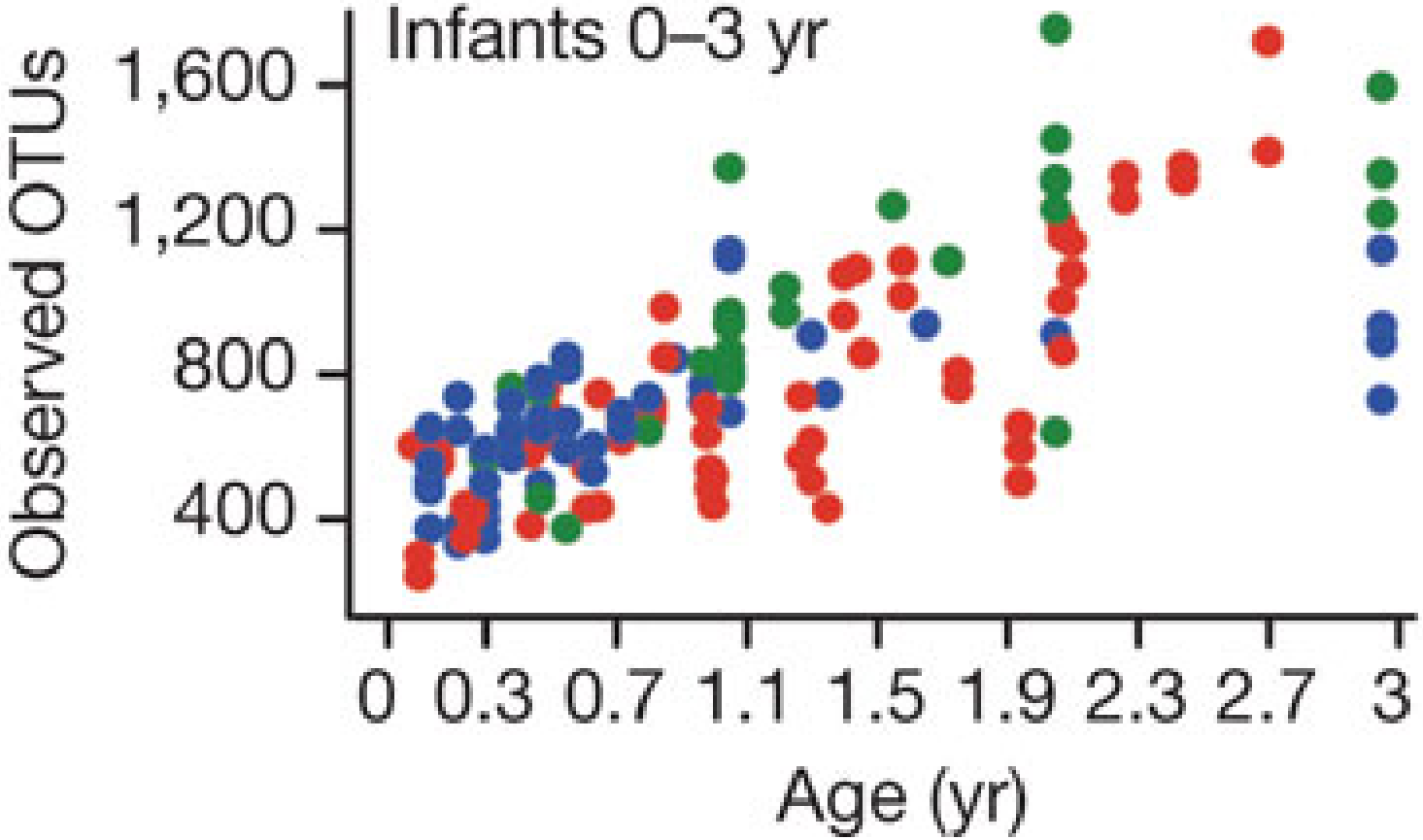
Metagenomics of the human gut



Diversity of species found with respect to origin and age

[Yatsunenko et al., Nature, 2012]

Metagenomics of the human gut



Diversity of species found with respect to origin and age
[Yatsunenko et al., Nature, 2012]

Two situations in "genomics"

1. a reference genome is available

   $\Rightarrow$ map reads on the genome

Two situations in "genomics"

1. a reference genome is available

   $\Rightarrow$ map reads on the genome

2. without a reference genome

   $\Rightarrow$ assemble the reads to get the genome

   or comparative analysis of several read sets

Locating read on a reference sequence "Mapping"

A definition of mapping

Locating or "mapping" reads

for each read, find its location of origin on the reference genome

A definition of mapping

Locating or "mapping" reads

for each read, find its location of origin on the reference genome

How?

use the sequence similarity between the read and the reference

$\Rightarrow$ approximate pattern matching or alignment

A definition of mapping

Locating or "mapping" reads

for each read, find its location of origin on the reference genome

How?

use the sequence similarity between the read and the reference

$\Rightarrow$ approximate pattern matching or alignment

Differences in sequence come from

1. sequencing errors
2. genetic variability at intra- and inter-individual
3. splicing of RNA compared to DNA sequence

Mapping for genomics, transcriptomics, or epigenomics

- Find for each read all genomic positions at which the read match either exactly or approximately on the genome (+/− strands)
- Results: is a read located? once or more than once?

  unmapped : not found

  uniquely mapped : mapped at a single genomic location

  mutiply mapped : mapped at several genomic locations

Bottleneck of mapping

- Data volume, typically:
  - 3 Giga bp of the Human genome sequence
  - 50 million reads, each 100 bases long par read

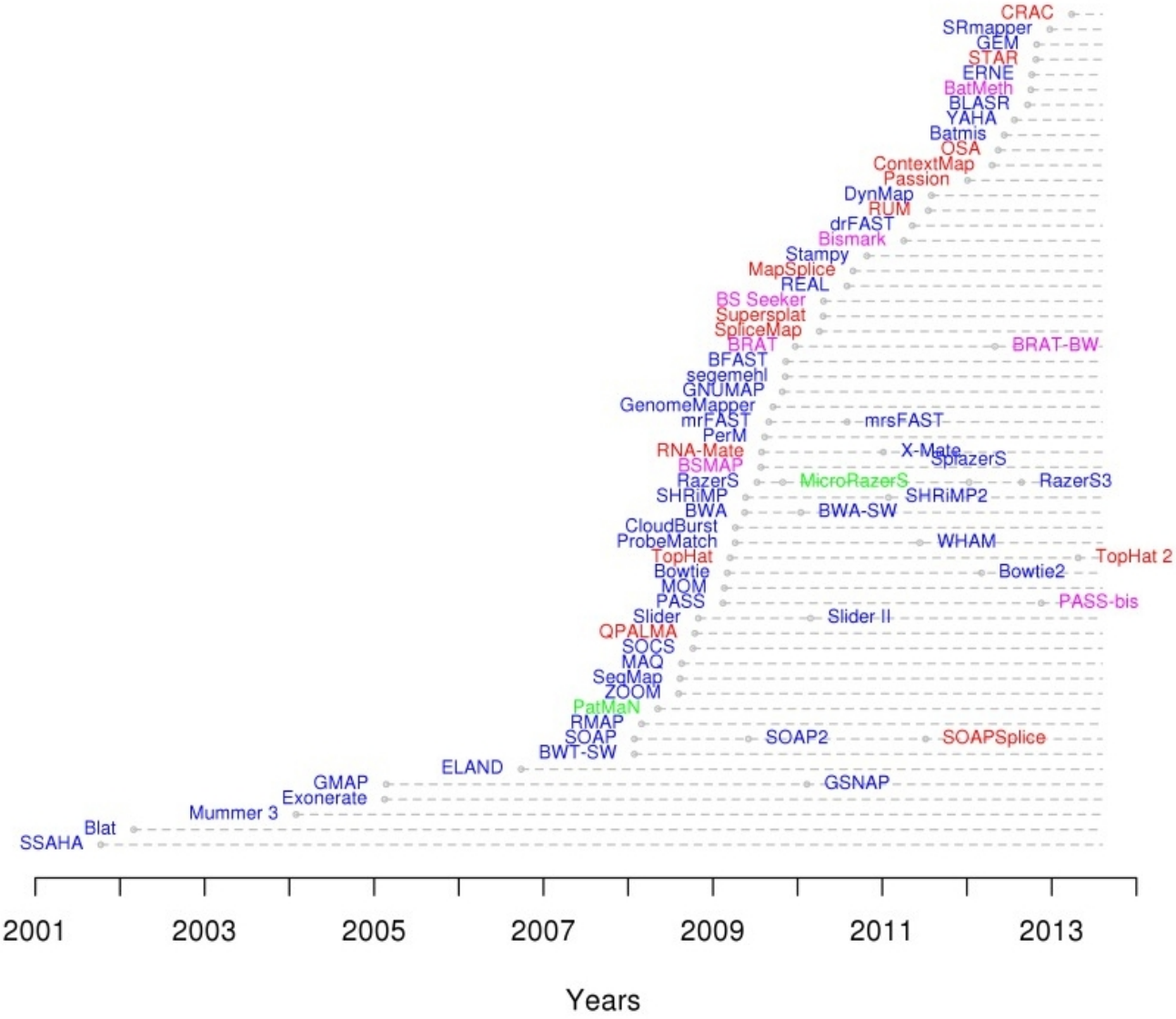
Bottleneck of mapping

- Data volume, typically:
  - 3 Giga bp of the Human genome sequence
  - 50 million reads, each 100 bases long par read
- Main issue: Scalability in terms of memory and time
  data flow especially in sequencing centers

How?

- indexing the genome sequence for answering pattern matching queries
- filtration algorithms for fast alignment

Mapping programs http://wwwdev.ebi.ac.uk/fg/hts_mappers



Is mapping difficult?

Major issues

- approximate matching: which definition? which threshold?
- which sequence length?
- probability of mapping in a random sequence
- sequence errors, model

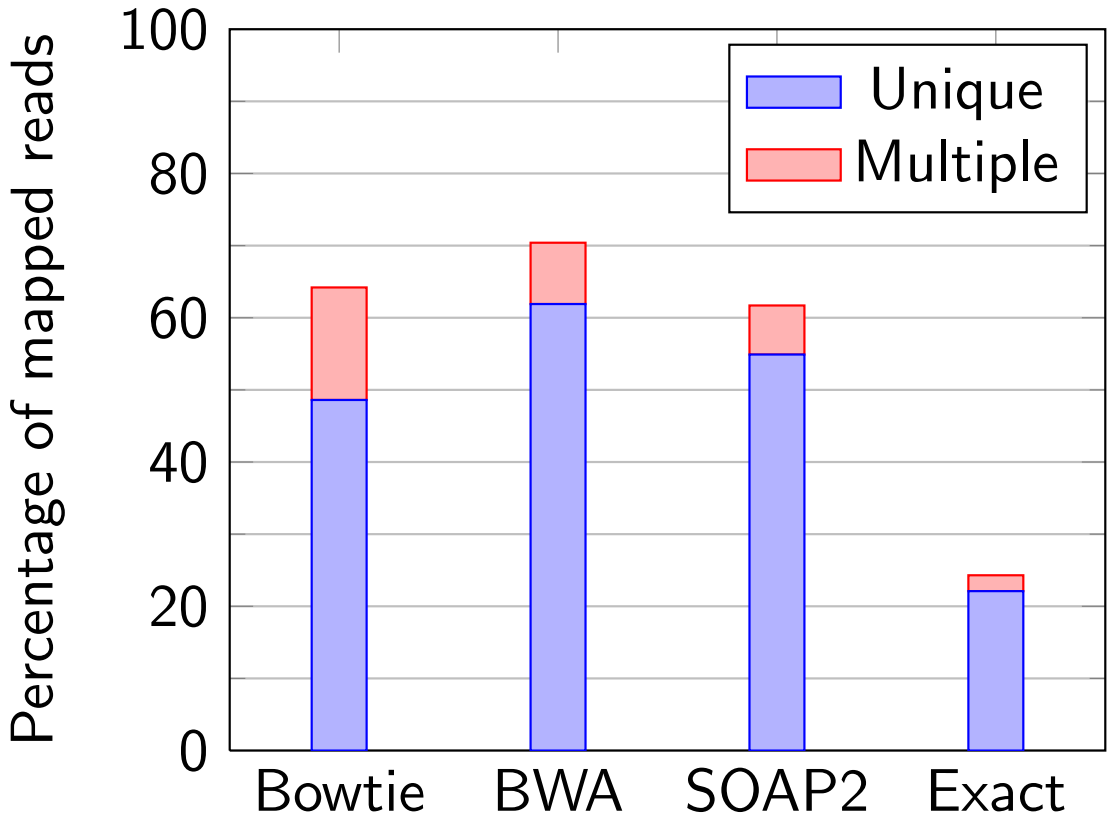
Other issues

- efficiency
- how to account for sequence quality?
- result guarantee?
- pratical issues: genome indexing, memory/disk occuppation

Mapping comparison

Data

- Human K562 cancer cell line RNA-Seq library
- 12 millions reads, 75 bp long



A pattern matching primer

Outline

1. The problem
2. Text indexing approach
3. Filtration approach

Pattern Matching

1. a *text* T of length n
2. a *pattern* M of length m
3. generally $m << n$.

Example: $M := tgtg$

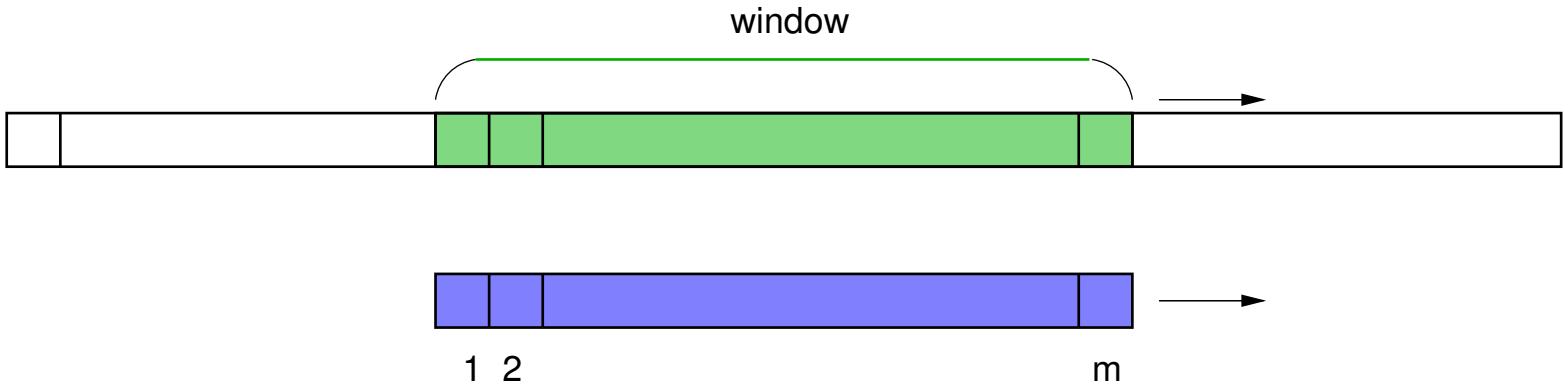
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
T:	c	t	g	t	g	t	g	t	a	c	a	t	g	t	g
		t	g	t	g										
				t	g	t	g								
												t	g	t	g

Solution: $\{2, 4, 12\}$

Pattern Matching

1. a *text* T of length n
2. a *pattern* M of length m
3. generally $m << n$.

For one read:



How to do it for millions of reads?

Naive and involved algorithms

- Naive algorithm:
  for each window m pairwise symbol comparisons
  $(n - m)$ windows
  $O(n * m)$ time complexity

- Linear time solutions:
  Idea: exploit results on a window to ease that of overlapping windows
  Boyer-Moore or Knuth Morris Pratt algorithms
  $O(n + m)$ time complexity

Naive and involved algorithms

- Naive algorithm:
  for each window m pairwise symbol comparisons
  $(n - m)$ windows
  $O(n * m)$ time complexity

- Linear time solutions:
  Idea: exploit results on a window to ease that of overlapping windows
  Boyer-Moore or Knuth Morris Pratt algorithms
  $O(n + m)$ time complexity

Limitations: single query and exact match

Naive and involved algorithms

- Naive algorithm:
  for each window m pairwise symbol comparisons
  $(n - m)$ windows
  $O(n * m)$ time complexity

- Linear time solutions:
  Idea: exploit results on a window to ease that of overlapping windows
  Boyer-Moore or Knuth Morris Pratt algorithms
  $O(n + m)$ time complexity

Limitations: single query and exact match

Answers: indexing text and filtration approaches

Multiple PM with a text index

Matching in two steps:

1. preprocessing the text T in $O(n)$ time
   build and store a data structure: an index
   enables exact search query

2. search for each pattern in the index in $O(m)$ time (optimal)

Text indexing data structures

For a text of length n, a good index:

1. occupancy memory in $O(n)$
2. construction time in $O(n)$ units
3. enables exact motif search in $O(m)$ time for a motif of length m

Three historical structures:

1. **compact suffix tree** [Wiener 73, McCreight 76, Ukkonen 92]
2. **suffix array**: construction in in $O(n \log(n))$ [Mamber-Myers 90], in $O(n)$ [Kärkkäinen & Sanders 03]
3. **DAWG** (Directed Acyclic Word Graph) [Blumer et al. 85]

Some index structures: Suffix array & BWT

Text example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- Alphabet: aeiln\$ with order : $\$ < a < e < i < l < n$
- \$ is a terminator symbol (no suffix is prefix of another)
- T a text of length 16
- T_i denotes the suffix starting at position i
  $T_{12} =$ eani\$

Ex: Suffix array (not yet sorted)

i	$SA[i]$	T_i
1		annilannealeani$
2		nnilannealeani$
3		nilannealeani$
4		ilannealeani$
5		lannealeani$
6		annealeani$
7		nnealeani$
8		nealeani$
9		ealeani$
10		aleani$
11		leani$
12		eani$
13		ani$
14		ni$
15		i$
16		$

Ex: Suffix array (sorted)

i	$SA[i]$	$T_{\mathrm{SA[i]}}$
1	16	$
2	10	aleani$
3	13	ani$
4	6	annealeani$
5	1	annilannealeani$
6	9	ealeani$
7	12	eani$
8	15	i$
9	4	ilannealeani$
10	5	lannealeani$
11	11	leani$
12	8	nealeani$
13	14	ni$
14	3	nilannealeani$
15	7	nnealeani$
16	2	nnilannealeani$

Ex: Suffix array (not yet sorted)

c	C[c]	#c
$	0	1
a	1	4
e	5	2
i	7	2
l	9	2
n	11	5
lg	16	-

i	$SA[i]$	$T_{\mathrm{SA[i]}}$
1	16	$
2	10	aleani$
3	13	ani$
4	6	annealeani$
5	1	annilannealeani$
6	9	ealeani$
7	12	eani$
8	15	i$
9	4	ilannealeani$
10	5	lannealeani$
11	11	leani$
12	8	nealeani$
13	14	ni$
14	3	nilannealeani$
15	7	nnealeani$
16	2	nnilannealeani$

Ex: Suffix array (not yet sorted)

c	C[c]	#c
$	0	1
a	1	4
e	5	2
i	7	2
l	9	2
n	11	5
lg	16	-

i	$SA[i]$	$T_{\mathrm{SA[i]}}$
1	16	$
2	10	aleani$
3	13	ani$
4	6	annealeani$
5	1	annilannealeani$
6	9	ealeani$
7	12	eani$
8	15	i$
9	4	ilannealeani$
10	5	lannealeani$
11	11	leani$
12	8	nealeani$
13	14	ni$
14	3	nilannealeani$
15	7	nnealeani$
16	2	nnilannealeani$

Ex: Suffix array (not yet sorted)

c	C[c]	#c
$	0	1
a	1	4
e	5	2
i	7	2
l	9	2
n	11	5
lg	16	-

i	$SA[i]$	$T_{\mathrm{SA[i]}}$
1	16	$
2	10	aleani$
3	13	ani$
4	6	annealeani$
5	1	annilannealeani$
6	9	ealeani$
7	12	eani$
8	15	i$
9	4	ilannealeani$
10	5	lannealeani$
11	11	leani$
12	8	nealeani$
13	14	ni$
14	3	nilannealeani$
15	7	nnealeani$
16	2	nnilannealeani$

Ex: Suffix array (not yet sorted)

c	C[c]	#c
$	0	1
a	1	4
e	5	2
i	7	2
l	9	2
n	11	5
lg	16	-

i	$SA[i]$	$T_{\mathrm{SA[i]}}$
1	16	$
2	10	aleani$
3	13	ani$
4	6	annealeani$
5	1	annilannealeani$
6	9	ealeani$
7	12	eani$
8	15	i$
9	4	ilannealeani$
10	5	lannealeani$
11	11	leani$
12	8	nealeani$
13	14	ni$
14	3	nilannealeani$
15	7	nnealeani$
16	2	nnilannealeani$

Ex: Suffix array (not yet sorted)

c	C[c]	#c
$	0	1
a	1	4
e	5	2
i	7	2
l	9	2
n	11	5
lg	16	-

i	$SA[i]$	$T_{\mathrm{SA[i]}}$
1	16	$
2	10	aleani$
3	13	ani$
4	6	annealeani$
5	1	annilannealeani$
6	9	ealeani$
7	12	eani$
8	15	i$
9	4	ilannealeani$
10	5	lannealeani$
11	11	leani$
12	8	nealeani$
13	14	ni$
14	3	nilannealeani$
15	7	nnealeani$
16	2	nnilannealeani$

Burrows-Wheeler Transform (BWT) of a text

Definition

Burrows-Wheeler Transform of a text T BWT is a string of length $|T|$ such that $BWT[i]$ is the symbol in T preceding the suffix $T_{\mathrm{SA[i]}}$; in other words

$$BWT[i] := \begin{cases} T[SA[i]-1] & \text{if } SA[i] > 1 \\ \$ & \text{otherwise} \end{cases}$$

Ex: BWT

i	$SA[i]$	$BWT[i]$	T_i	T_i^c
1	16	i	$	annilannealeani
2	10	e	aleani$	annilanne
3	13	e	ani$	annilanneale
4	6	l	annealeani$	annil
5	1	$	annilannealeani$	ϵ
6	9	n	ealeani$	annilann
7	12	l	eani$	annilanneal
8	15	n	i$	annilannealean
9	4	n	ilannealeani$	ann
10	5	i	lannealeani$	anni
11	11	a	leani$	annilannea
12	8	n	nealeani$	annilan
13	14	a	ni$	annilannealea
14	3	n	nilannealeani$	an
15	7	a	nnealeani$	annila
16	2	a	nnilannealeani$	a

Ex: BWT

i	$SA[i]$	$BWT[i]$	T_i
1	16	i	$
2	10	e	aleani$
3	13	e	ani$
4	6	l	annealeani$
5	1	$	annilannealeani$
6	9	n	ealeani$
7	12	l	eani$
8	15	n	i$
9	4	n	ilannealeani$
10	5	i	lannealeani$
11	11	a	leani$
12	8	n	nealeani$
13	14	a	ni$
14	3	n	nilannealeani$
15	7	a	nnealeani$
16	2	a	nnilannealeani$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ `an`

1. `n` and full-interval $[1, 16]$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$
2. a and n-interval $[12, 16]$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$
2. a and n-interval $[12, 16]$ $\longrightarrow$ an-interval: $[3, 5]$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$
2. a and n-interval $[12, 16]$ $\longrightarrow$ an-interval: $[3, 5]$

$$BS(a, [12, 16]) = [C[a] + Occ(a, 12-1) + 1, C[a] + Occ(a, 16)]$$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$
2. a and n-interval $[12, 16]$ $\longrightarrow$ an-interval: $[3, 5]$

$$\begin{aligned} BS(a, [12, 16]) &= [C[a] + Occ(a, 12-1) + 1, C[a] + Occ(a, 16)] \\ &= [1 + Occ(a, 11) + 1, 1 + Occ(a, 16)] \end{aligned}$$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$
2. a and n-interval $[12, 16]$ $\longrightarrow$ an-interval: $[3, 5]$

$$\begin{aligned} BS(a, [12, 16]) &= [C[a] + Occ(a, 12-1) + 1, C[a] + Occ(a, 16)] \\ &= [1 + Occ(a, 11) + 1, 1 + Occ(a, 16)] \\ &= [1 + 1 + 1, 1 + 4] \end{aligned}$$

Backward Search example

T:	a	n	n	i	l	a	n	n	e	a	l	e	a	n	i	\$
i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Backward Search:

$BS(c, [i, j]) = [C[c] + Occ(c, i-1) + 1, C[c] + Occ(c, j)]$

One seeks $v :=$ an

1. n and full-interval $[1, 16]$ $\longrightarrow$ n-interval: $[12, 16]$
2. a and n-interval $[12, 16]$ $\longrightarrow$ an-interval: $[3, 5]$

$$\begin{aligned} BS(a, [12, 16]) &= [C[a] + Occ(a, 12-1) + 1, C[a] + Occ(a, 16)] \\ &= [1 + Occ(a, 11) + 1, 1 + Occ(a, 16)] \\ &= [1 + 1 + 1, 1 + 4] \\ &= [3, 5] \end{aligned}$$

Ex: BWT

Symbol counts table

c	C[c]	Occ(c,16)
$	0	1
a	1	4
e	5	2
i	7	2
l	9	2
n	11	5

i	$SA[i]$	$BWT[i]$	T_i
1	16	i	$
2	10	e	aleani$
3	13	e	ani$
4	6	l	annealeani$
5	1	$	annilannealeani$
6	9	n	ealeani$
7	12	l	eani$
8	15	n	i$
9	4	n	ilannealeani$
10	5	i	lannealeani$
11	11	a	leani$
12	8	n	nealeani$
13	14	a	ni$
14	3	n	nilannealeani$
15	7	a	nnealeani$
16	2	a	nnilannealeani$

Filtration for approximate pattern matching

Filtration for approximate matching

Algorithm in 2 phases: **filtration** & **verification**
based on a necessary condition for a match

Filtration : find all substrings M' satisfying the condition
M' is a *potentiel match*

Verification : check whether M' is a true occurrence of M
dynamic programming in $O(nm)$ time

Pros: if the condition is easy to test and potential matches are rare, few substrings of T will be considered for verification
$\Rightarrow$ gain in computing time

k-mer distance

Definition

a *k-mer* is a string of length q over an alphabet Σ

Idea

Let d be the max nb of errors allowed (an integer), and $k \leq \lfloor \frac{m}{d+1} \rfloor$.

Count the nb of k-mers equal between M & window M'

Each difference affect at most k k-mers.

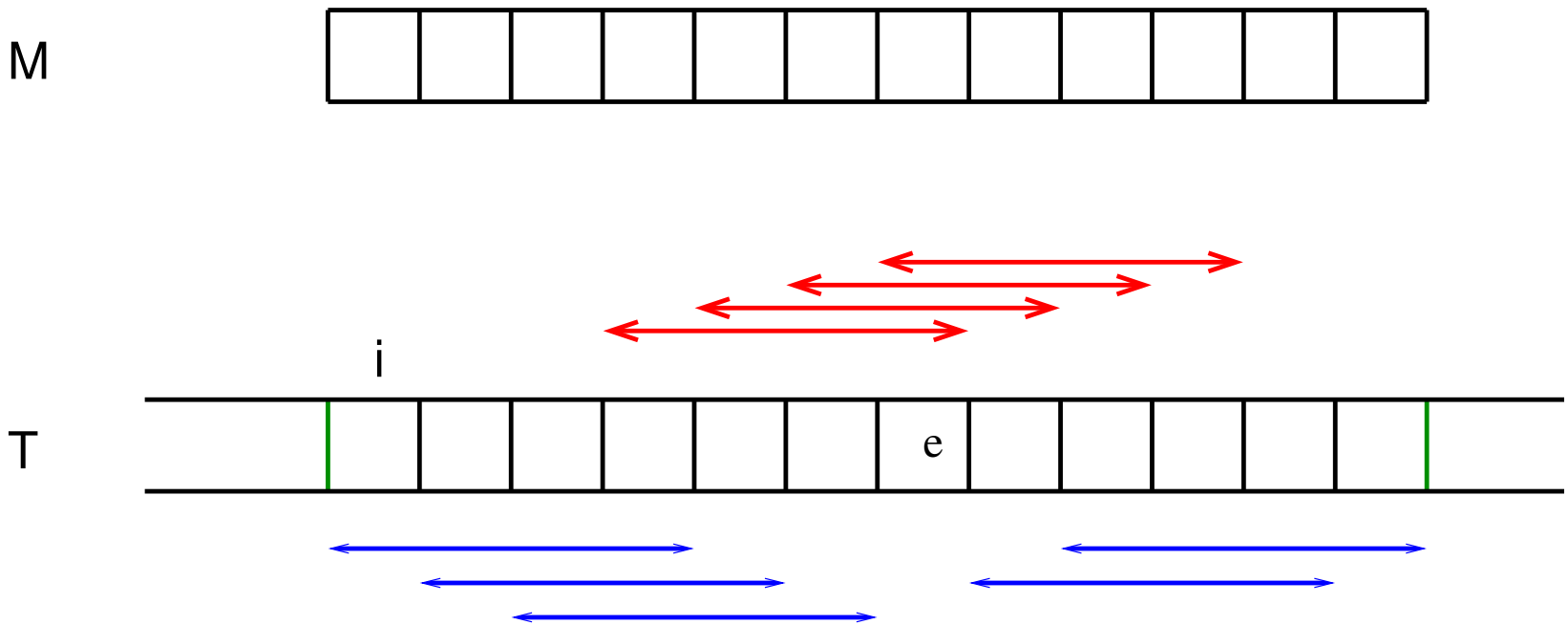
Worst case: if the edit distance $e(M, M') \leq d$,
$m - (d+1)k + 1$ k-mers match between M & M'.

k-mer filter

length of M: 12; $q := 4$

$\longleftrightarrow$: equal k-mers

$\longleftrightarrow$: different k-mers



[Owolabi, McGregor, 88]

Mapping tools

Fast mapping tools

Bowtie (v1 & 2), BWA, BWA-SW, SOAP2

- k-mer filtration
- use a compressed genome index: Burrows-Wheeler Transform (BWT)
- search for continous alignments
- differing by at most a few mismatch/indels

GASSST [Rizk, Lavenier, 2010]

- spaced seed based filtration
- Hash tables: dedicated index
- allows longer indels

Tools for RNA-seq

To detect splice junctions

- TopHat (v1 & 2) [Trapnell et al., 2009]
- MapSplice [Wang et al., 2010]
- GSNAP [Wu et Nacu, 2010]
- CRAC [Philippe *et al.* 2013]

To detect fusion RNAs splice junctions

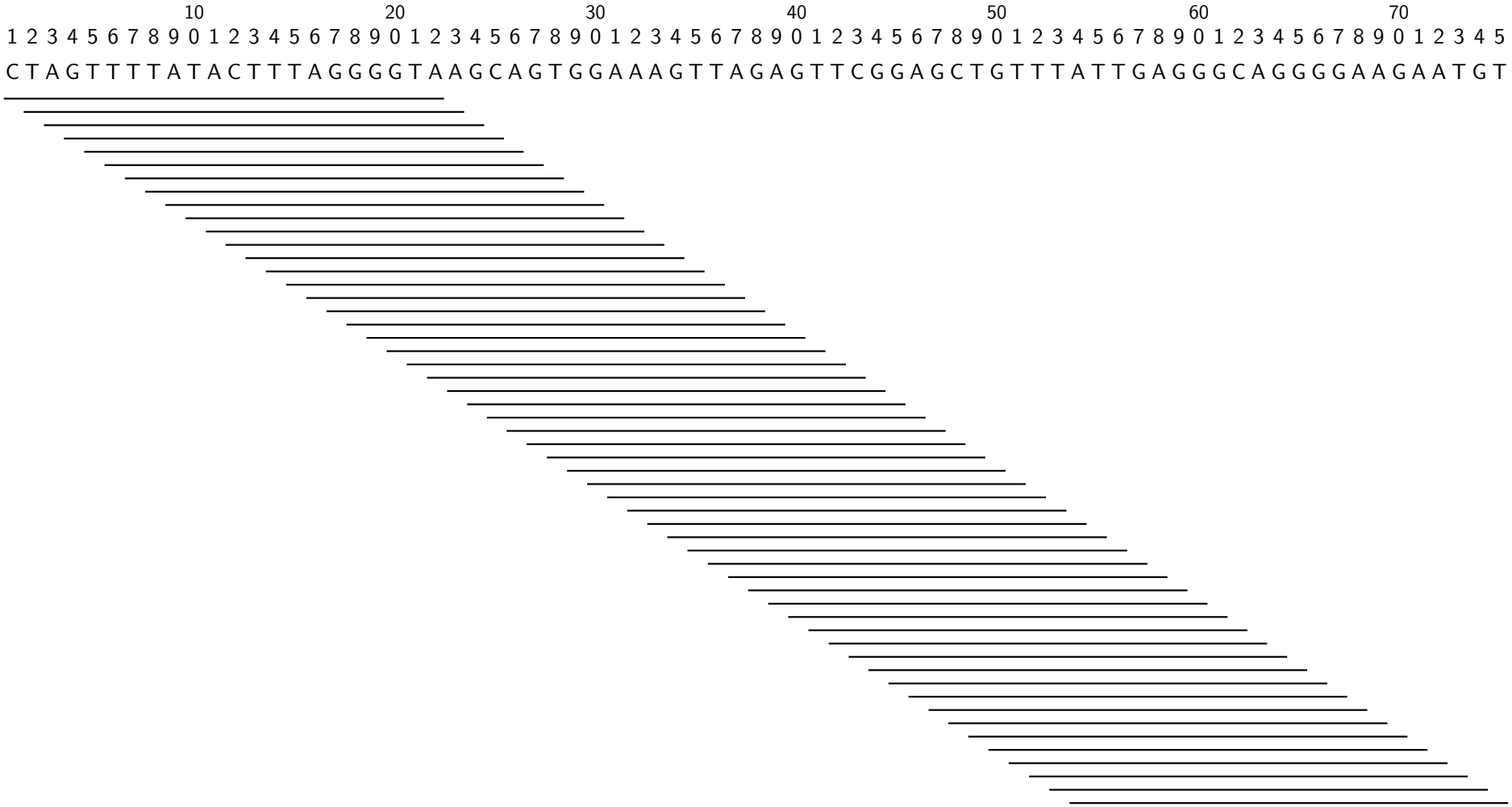
- MapSplice [Wang et al., 2010] single reads
- TopHat fusion [McPherson et al., 2011] single reads
- FusionSeq [Sboner et al., 2010] paired reads
- FusionHunter [Li et al., 2011] paired reads
- CRAC [Philippe *et al.* 2013] single & paired

CRAC

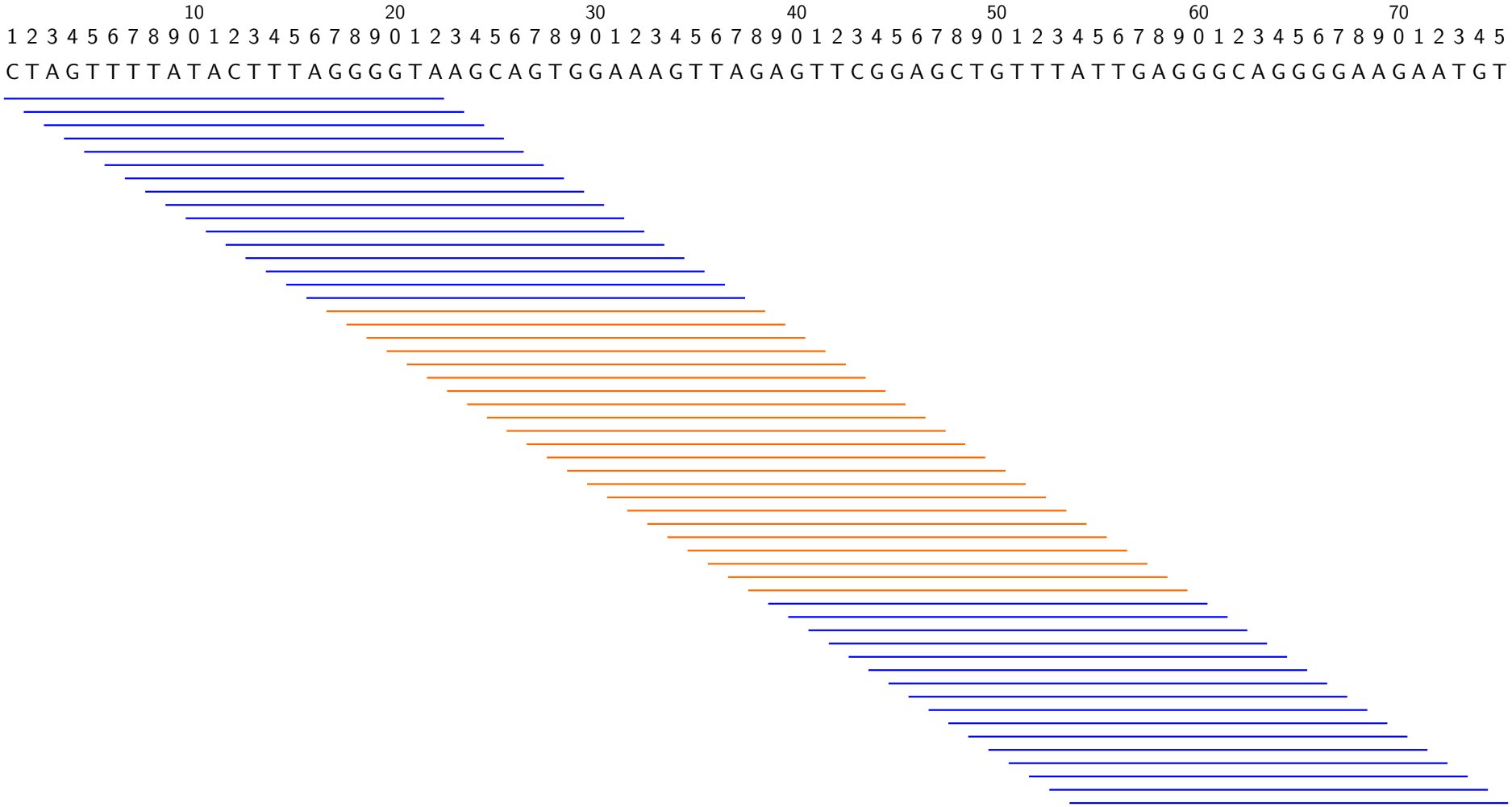
CRAC: *k*-mer profiling

```
          10        20        30        40        50        60        70
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
C T A G T T T T A T A C T T T A G G G G T A A G C A G T G G A A A G T T A G A G T T C G G A G C T G T T T A T T G A G G G C A G G G G A A G A A T G T
```

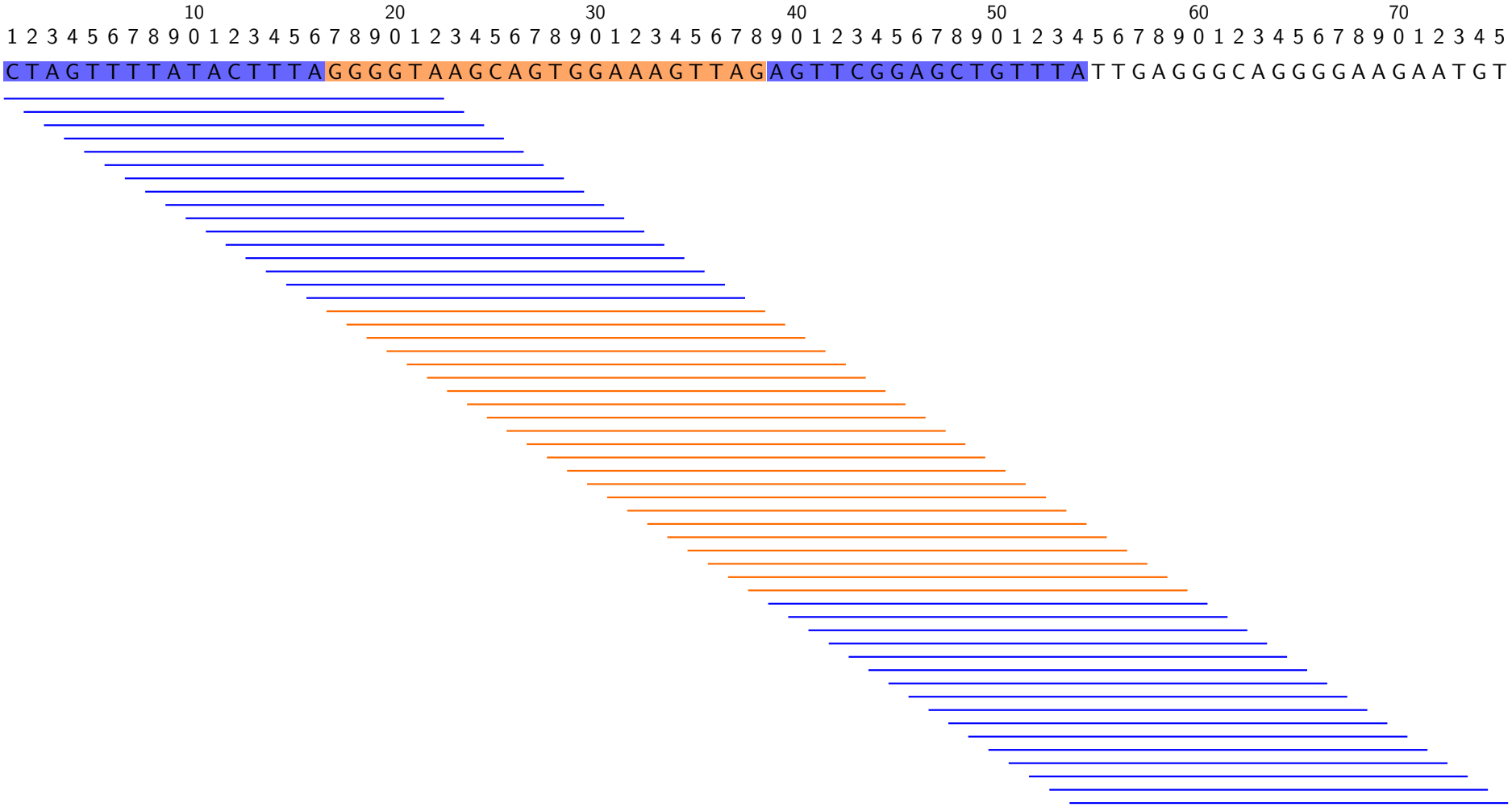
CRAC: *k*-mer profiling



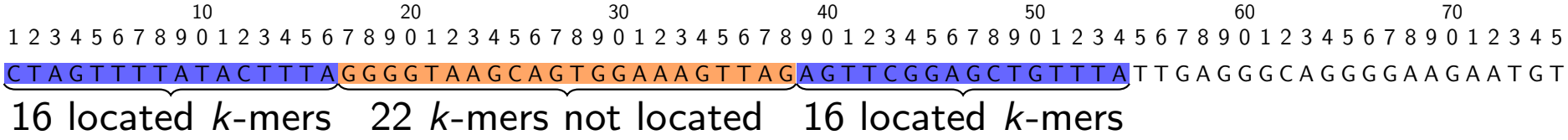
CRAC: *k*-mer profiling



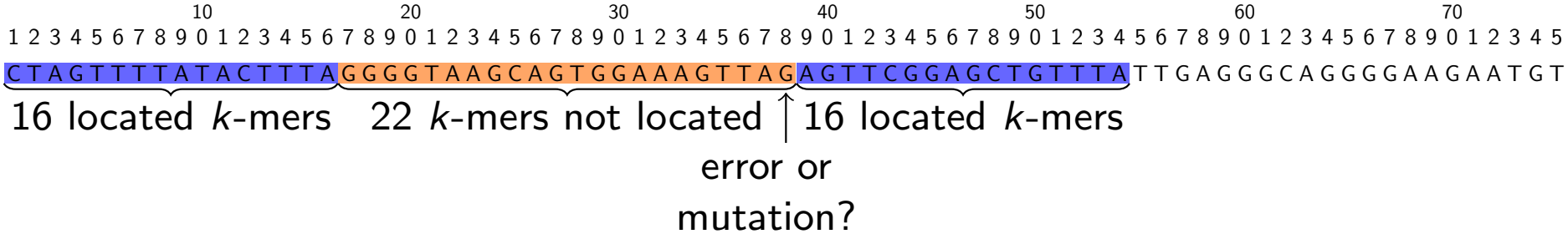
CRAC: *k*-mer profiling



CRAC: *k*-mer profiling



CRAC: *k*-mer profiling

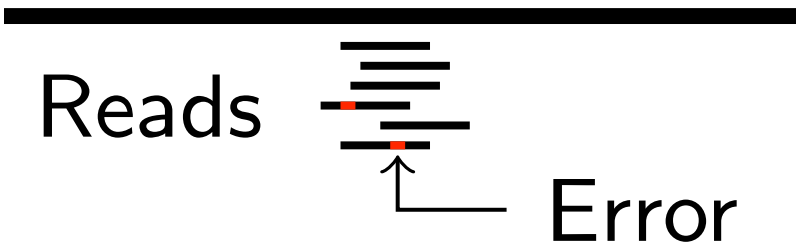


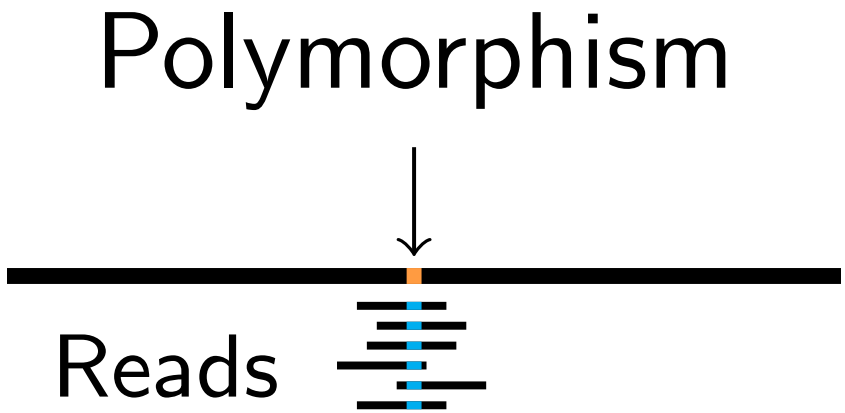
Principle II: genetic variation

A genetic variation affect all reads covering its position

while a sequence error occurs in a read, affects only that read

Error or mutation?





CRAC: idea

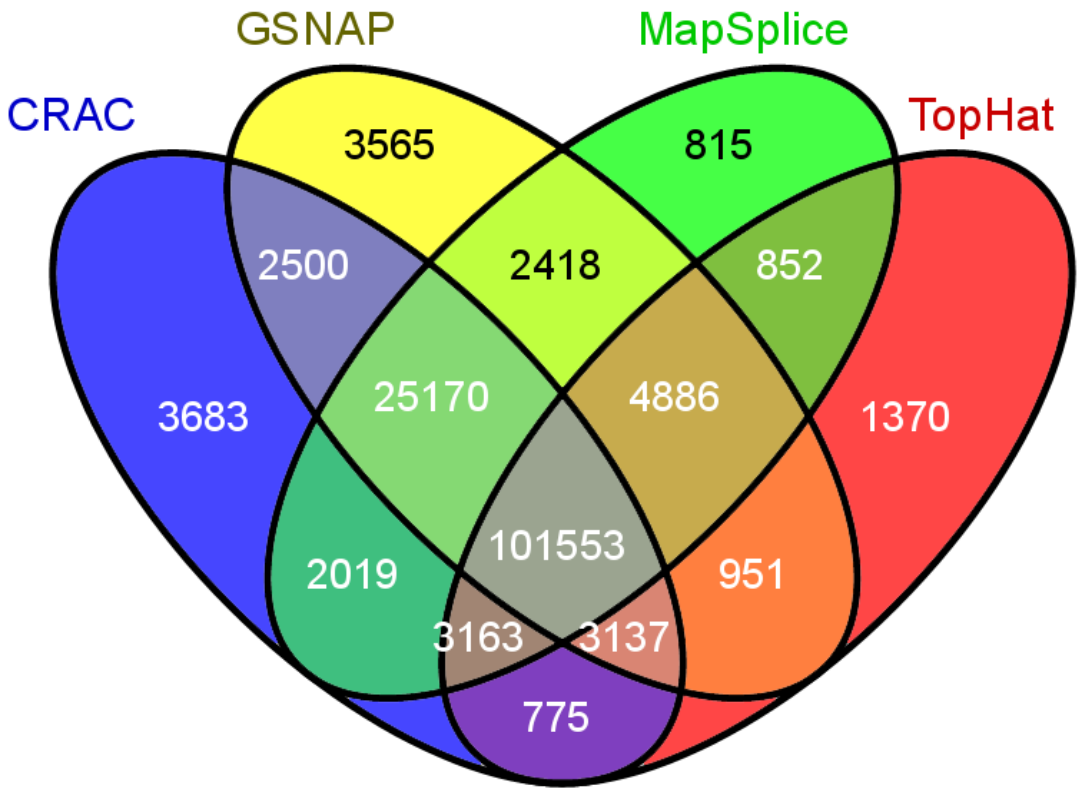
For each read, it analyzes jointly two signals for each k-mer

- the location of the k-mer on the genome
  i.e. its matching locations and their number,
- the support: the number of reads sharing this k-mer

How? on the fly using indexes:

- a compressed Burrows-Wheeler Transform of the genome
- a generalized k-factor table built on all reads [Philippe et al., 2011]

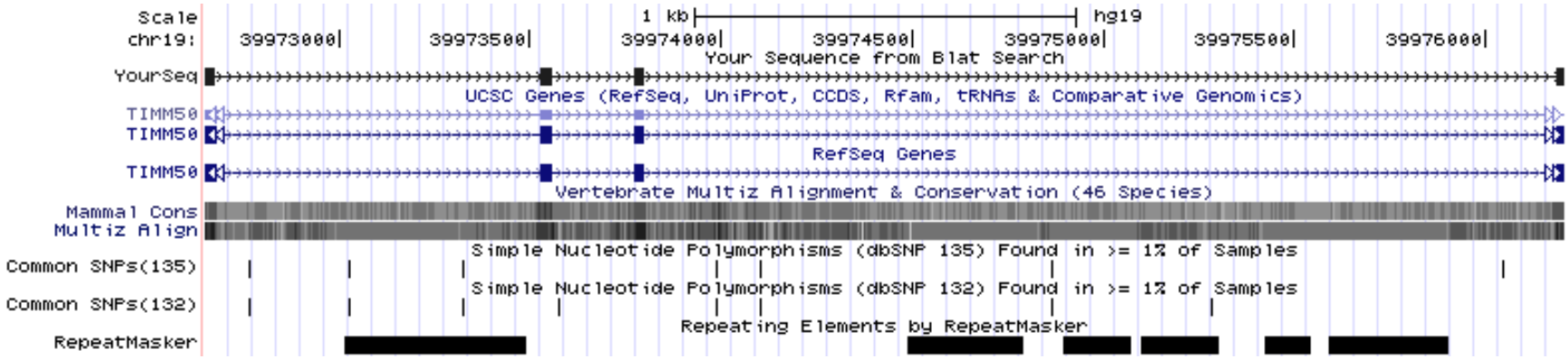
Splice junction detection on real data (Human)



Agreement between tools on known RefSeq splice junctions

Reads spanning several exons and junctions

a read overlapping exons 2-5 of TIMM50 gene



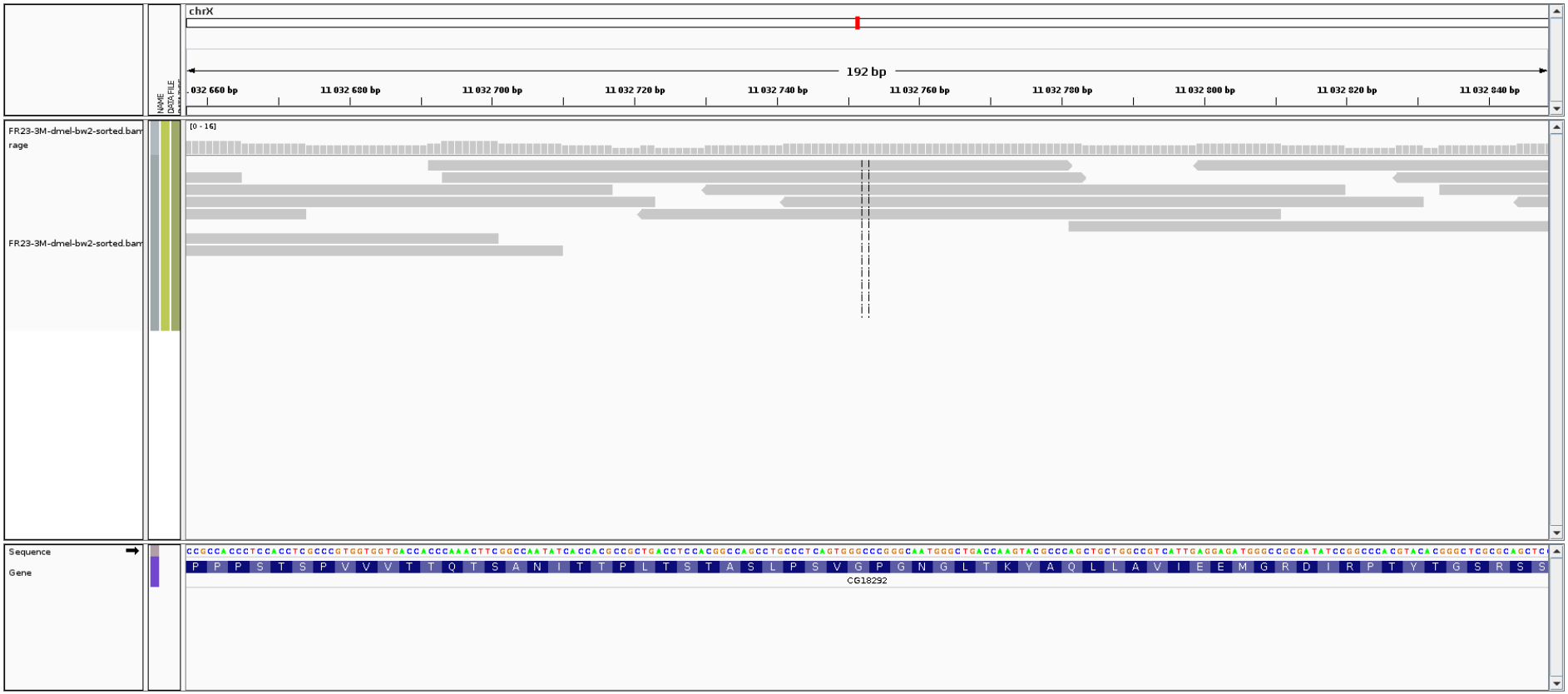
Candidate fusion RNAs in four Breast cancer libraries

Edgren et al. : 4 cancer cell lines, RNA-seq, 50 millions reads of 50 nt

Edgren libraries	CRAC after GSNAP chRNA	CRAC after GSNAP reads	TopHat-fusion after GSNAP chRNA	TopHat-fusion after GSNAP reads
BT-474	153	460	81 327	189 523
KPL-4	60	199	23 075	53 165
MCF-7	90	180	27 267	57 676
SK-BR-3	152	577	61 494	130 682

Visualisation & Integrative selection of candidate

Visualisation : "Integrative Genomics Viewer (IGV)"



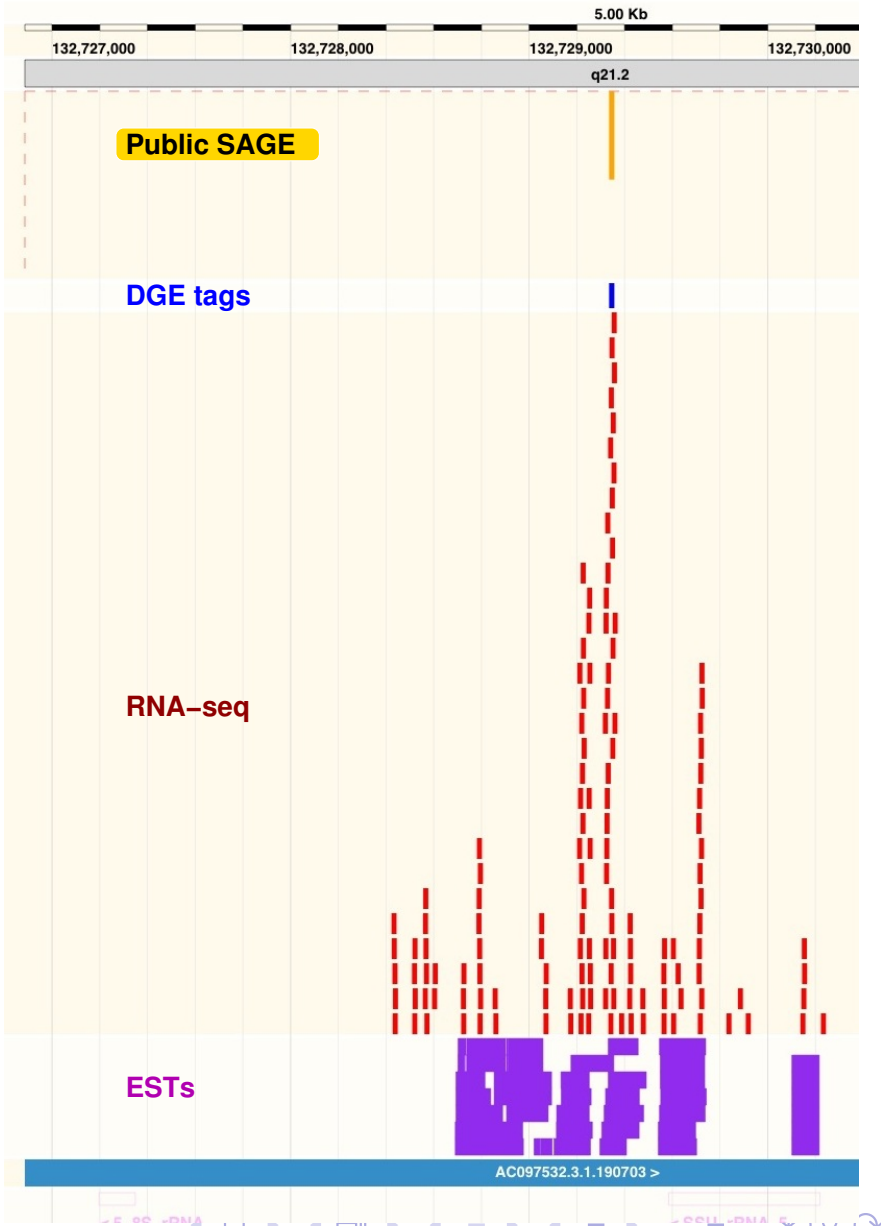
mapped reads on Drosophila genome *D. melanogaster*

Integrative selection of candidate novel RNAs

Criteria:

- Distal intergenic reads
- DGE expression level > 2
- Public SAGE expr. level > 5
- RNA-seq in $5'$ vicinity > 3

[Philippe et al., Human Genome Meeting, 2010]



Conclusion

- NGS assays pervade many domains of biology
  and are exploited for numerous and divers studies
- Bioinformatics analysis is the current bottleneck
  growing demand, lot of development and research
- The scalability challenge is solved up to now . . .
  with text indexing algorithms
- Data integration for prioritizing candidates
- Opportunities for Computer Science developments
- Needed in research and industry

Funding and acknowledgments

MAB team and in particular B. Cazaux, M. Hébrard, V. Maillol, V. Lefort

MASTODONS SePhHaDe project

Thanks for your attention

Questions?

A few references

- CRAC: an integrated approach to the analysis of RNA-seq reads: N. Philippe, M. Salson, T. Commes, E. Rivals. Genome Biology 14:R30, 2013.
- Filtration:
  S. Burkhardt, A. Crauser, P. Ferragina, H.-P. Lenhof, E. Rivals, M. Vingron,
  q-gram Based Database Searching Using a Suffix Array (QUASAR),
  Proc. of the *3rd International Conference on Computational Molecular Biology (RECOMB99)*, ACM Press.
- Index data structures:
  D. Gusfield's book, OUP, 1997.
  V. Mäkinen, G. Navarro: Compressed Text Indexing. Encyclopedia of Algorithms. Springer-Verlag, 2008.
  N. Välimäki, E. Rivals, *Scalable and Versatile k-mer Indexing for High-Throughput Sequencing Data*, *ISBRA*, LNBI 7875, 2013.