Scalable solutions for genomic data analysis

Tomasz Gambin

Institute of Computer Science, Warsaw University of Technology


http://biodatageeks.org


http://github.com/ZSI-Bio


http://twitter.com/biodatageeks

Warsaw University of Technology

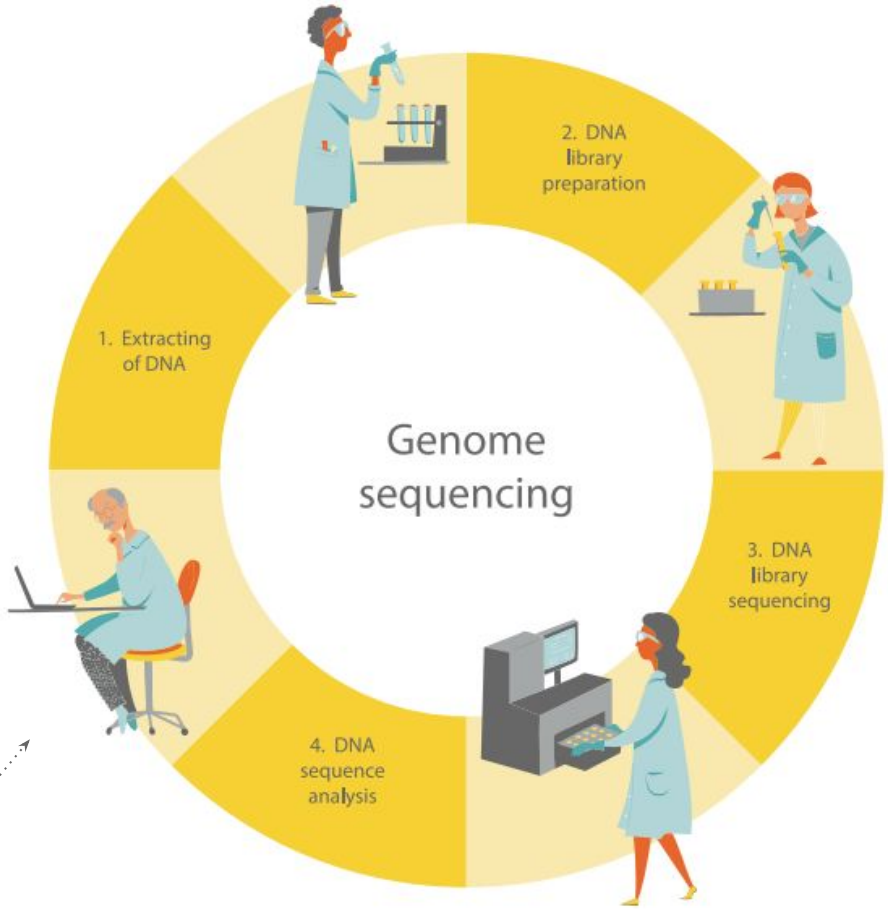
About us

Areas of interest:

- Genomic data analysis
- Distributed solutions for genomics

Applications:

- Distributed pipelines for variant calling
- Depth of coverage analysis, quality control
- Distributed range joins implementation
- Efficient variant data warehousing solutions
- Interpretation tools



That's us

>300 novel disease genes

Chong et al. *AJHG* , 2015

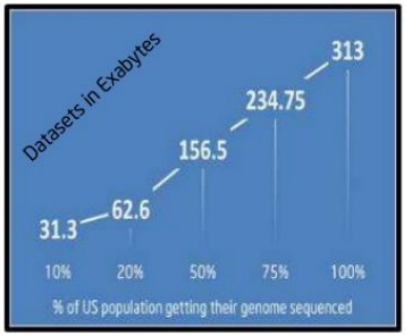
James Lupski

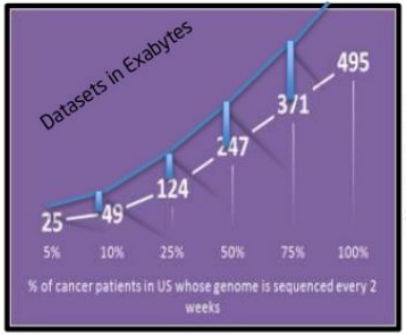
Eric Boerwinkle

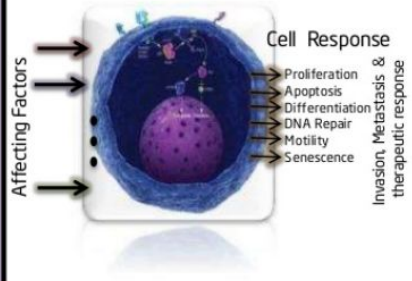
Richard Gibbs

Genomics - Big Data Problem

The day when every newborn gets their DNA sequenced is not far away: http://www.nih.gov/news/health/sep2013/nhgri-04.htm.







313 *Exabytes*
if everyone in the US has their genes sequenced

495 *Exabytes*
if every cancer patient in the US has their genes sequenced every 2 weeks.

Images, Assays and Drug response data will push it further up as shown in Blue line

Complex interaction of *varied & changing* intrinsic and extrinsic factors determine cell response

With Genomic Data growing rapidly, hospitals and research centers need to access the local data (the ones not shared) and the centralized public/private data for various analysis and analytics for Genomic Research/Development/Medicine.

Compute has to be done "where data is" and need to be consistent locally and in the cloud.
Energy, Total Cost of Operation are key

Source: Knights Cancer Institute, Oregon Health Sciences University & Intel

Figure: Genomics as big data problem, Source: Knights Cancer Institute, Oregon Health Sciences University & Intel

Challenges

Performance

Computation intensive steps in pipelines and painful long lasting operations in data analysis.

Multi-sample analysis

Data standardization and merging as additional overhead

Data size

Heavy files, inefficient data access, temporary files generation. High storage cost

Data security

'All or nothing' approach is not enough for research projects or clinics

Our solutions

Distributed calculations

Reimplementation of algorithms using Apache Spark and other Big Data tools

Optimized algorithms

Ensuring scalability to handle population-scale analysis

Unified data model

Providing ANSI SQL-compliant interfaces
Table-oriented processing

Standard technologies

Fine-grained access control

SeQuiLa (range joins)- scalable intersection of interval sets

e.g: What variants (snps) occur WITHIN genes

chr	name	start	end
11	Gene1	100	150
11	Gene2	200	250
11	Gene3	300	350
12	Gene4	100	150

chr	start	end	ref	alt	af
11	101	101	C	T	0.5
11	104	105	CT	C	0.01
11	134	135	AA	TT	0.01
11	201	201	A	G	0.05
12	102	102	T	G	0.05
13	1004	1005	TA	CG	0.04
13	2004	2005	TA	CG	0.04

chr	name	start	end	start	end	ref	alt	af
11	Gene1	100	150	101	101	C	T	0.5
11	Gene1	100	150	104	105	CT	C	0.01
11	Gene1	100	150	134	135	AA	TT	0.01
11	Gene2	200	250	201	201	A	G	0.05
12	Gene4	100	150	102	102	T	G	0.05

```
SELECT g.chr, g.name, g.start, g.end, s.start,
s.end, s.af
FROM genes g JOIN snps s ON (
  g.chr = s.chr AND s.start>= g.start AND s.end
<= g.end)
```

- counting overlaps
- additional criterias on overlap (maxGap, minOverlap)

real genomic example: $(160 * 10^6) \times (200 * 10^3)$ or even: $(2{,}6 * 10^9) \times (200 * 10^3)$

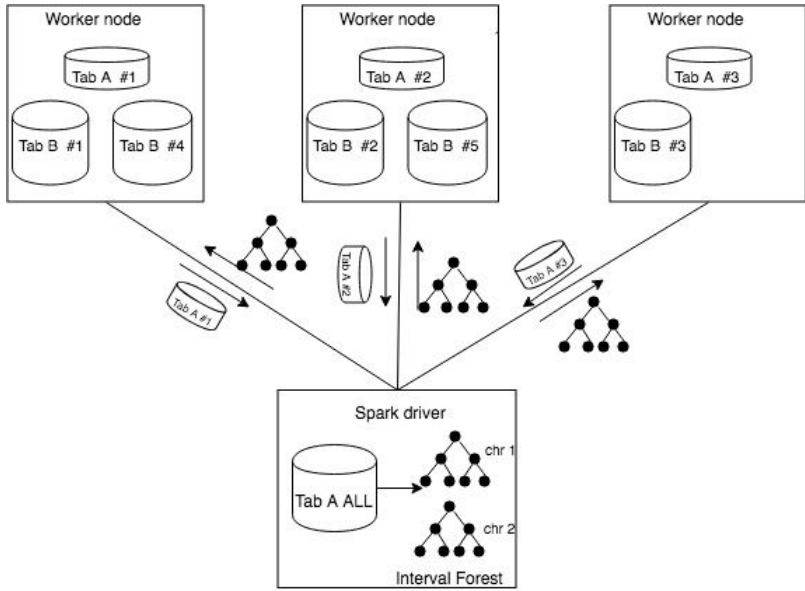
SeQuiLa (range joins): methods

Extension of Catalyst (SparkSQL component)

1. **IntervalTree** structure is used for efficient overlaps search
   a. Interval Forest (one tree for each chromosome)
2. augmenting IntervalTree with table data if possible

Algorithm for range join table A (small) with table B (big):

1. Send to driver node all table A partitions
2. Build Interval Forest in driver node
3. Broadcast Interval Forest to all worker nodes
4. Perform interval search
5. Join search results with table A if necessary

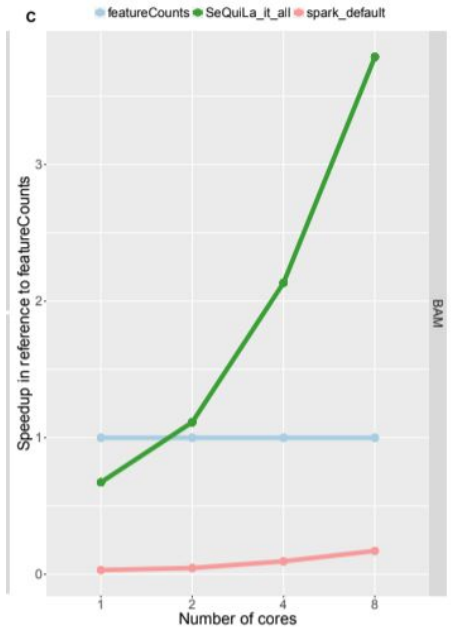


SeQuiLa (range joins)

SeQuiLa: an elastic, fast and scalable SQL-oriented solution for processing and querying genomic intervals

Marek Wiewiórka, Anna Leśniewska, Agnieszka Szmurło, Kacper Stępień, Mateusz Borowiak, Michał Okoniewski, Tomasz Gambin ✉

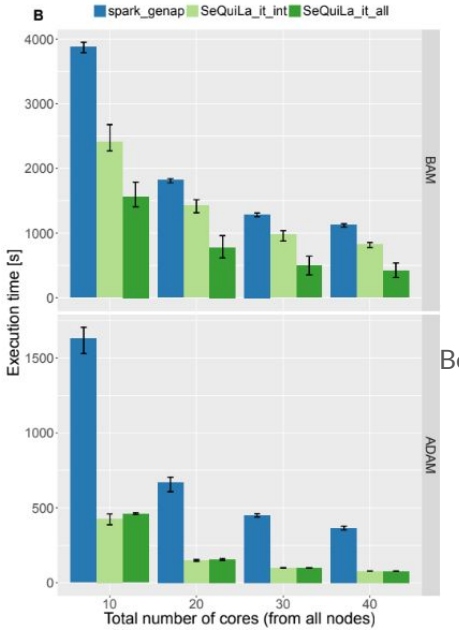
Bioinformatics, bty940, https://doi.org/10.1093/bioinformatics/bty940
Published: 14 November 2018 **Article history** ▾



- single node
- data: WES (17 GB
- reads (160 *10^6) x targets (200 *10^3))

Benchmark against:

- featureCounts
- SparkGenap
- spark default



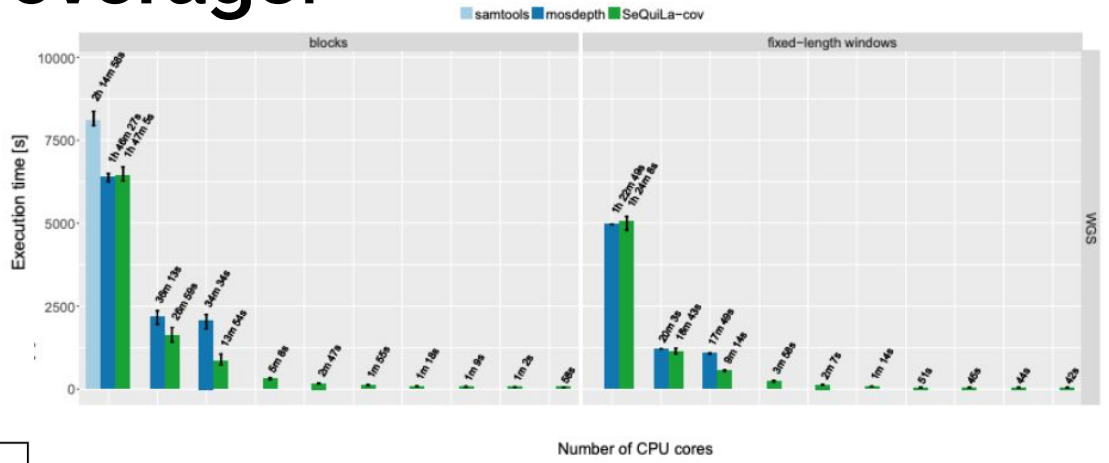
- cluster (4 worker nodes)
- data: WGS (270 GB)
- reads (2,6 *10^9) x targets (200 *10^3)

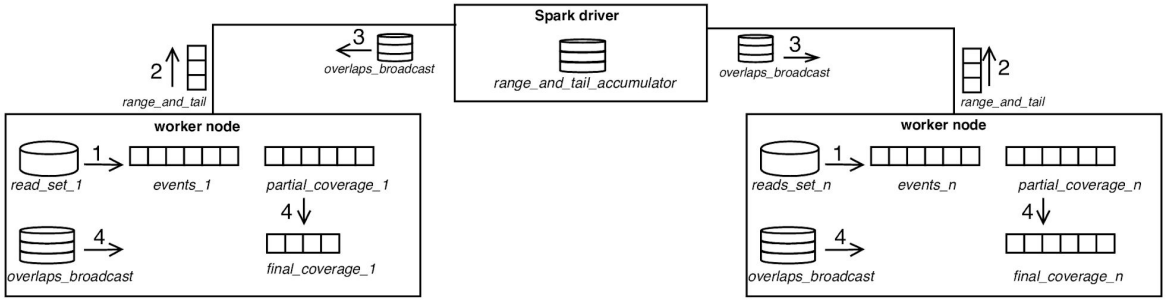
Benchmark against:

- SparkGenap

Distributed Depth of Coverage:

- Distributed calculations
- Simple event-based algorithm
- Low level optimizations
- Standalone version





2019

SeQuiLa-cov: A fast and scalable library for depth of coverage calculations

Marek Wiewiórka[1,*], Agnieszka Szmurło[1,*], Wiktor Kuśmirek[1] and Tomasz Gambin[1,†]

[1]Institute of Computer Science, Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, Poland,

OXFORD

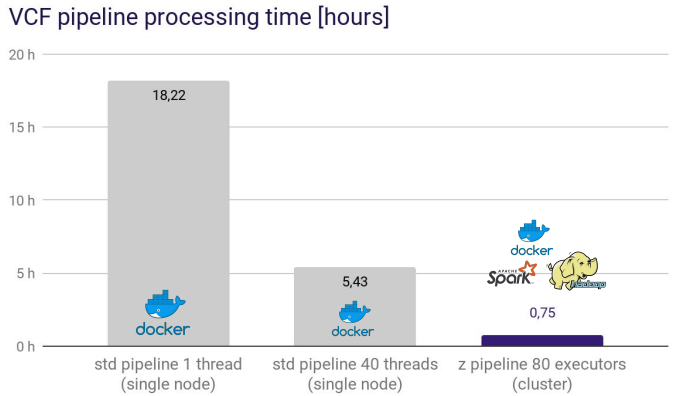
(GIGA)n SCIENCE

http://biodatageeks.org/sequila/

Distributed variant annotation pipeline:

- Automatic execution
- Customizations
- Monitoring of task execution
- Distributed calculations

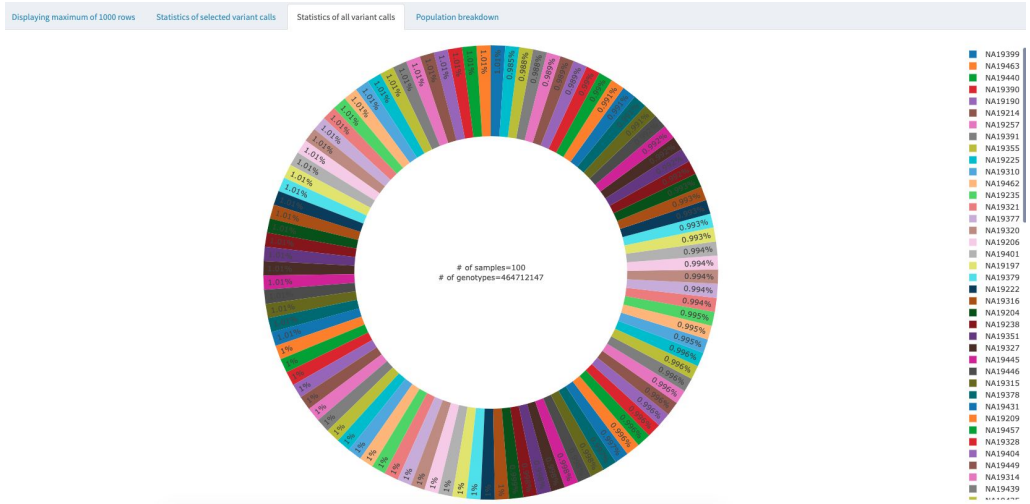
Benchmark:

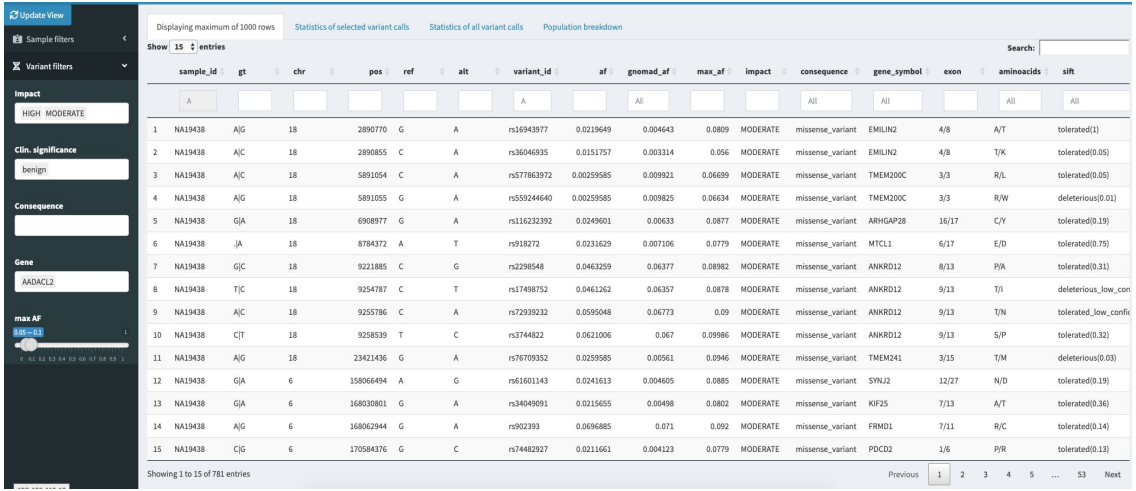


	Off	cleanup		zettagene				
	On	cov_only	None	zettagene	2	2019-03-18 13:02	1	
	On	coverage_cleanup	None	zettagene	4	2019-03-19 09:37	8 9	
	On	coverage_pipeline	None	zettagene	5	2019-03-20 13:05	3 9	
	On	vcf_cleanup	None	zettagene				
	On	vcf_pipeline	None	zettagene	7	2019-03-08 19:10	50 79	

Interpretation tools:

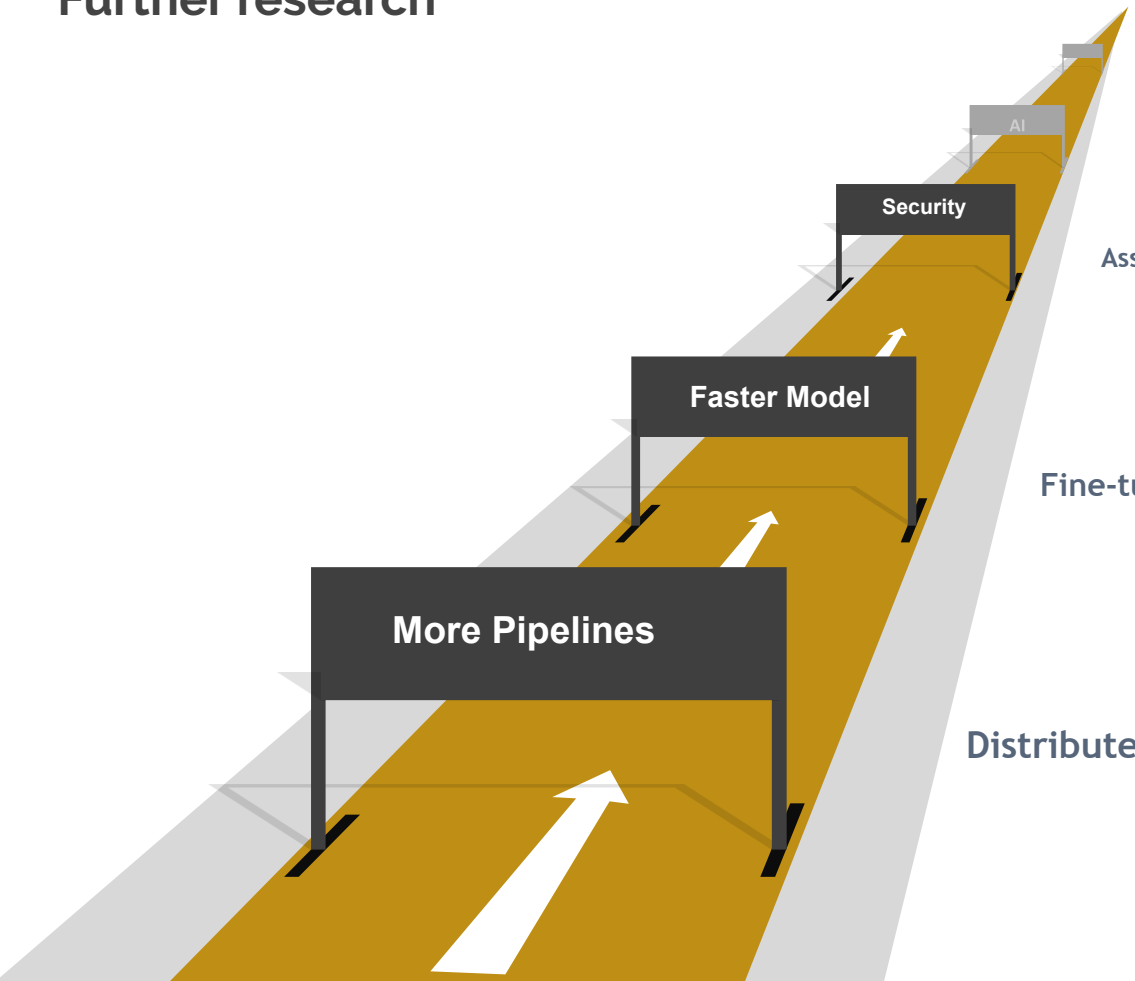
- Tabular view on variants and genotypes
- Charts, statistics, breakdowns
- Fine grained access control
- IGV view of variants and aligned reads






Bioinformatics pipelines
NGS variant calling
TAA
CGTC
sequence alignment
variant calling
decompose
normalize
annotate
Coverage QC
BAM
store reads
calculate coverage
calculate aggregates
RNA-Seq
TAA
CGTC
transcript alignment
calculate coverage
differential analysis
Data analysis
docker
Interpretation tools
Shiny
Superset
Custom analyses
RStudio
jupyter
Spark
Data
data store
HIVE
reads
coverage
variants
aggregates
mviews
genotypes
execution environment
Spark
SeQuiLa-join
SeQuiLa-cov
SparkSQL
distributed platform
resource manager (YARN)
distributed storage (HDFS)
Security
user management (FreeIPA)
security policies (Ranger)
Authentication (Kerberos)

Further research



Distributed implementations of algorithms for downstream analysis

Assuring data security for in-memory cache solutions

Fine-tuning of data model to increase query performance

Distributed pipelines for CNV and RNA-seq

Thanks!

Any questions?

You can find us at **biodatageeks.org**