PRECISION Medicine 2019

Can AI Accelerate Precision Medicine?

Analysis of scalable genomic variant data with HAIL using clusters in cloud environments

Instructors: Ines Krissaane, Lindsay Leek and Carlos De Niz

HARVARD
MEDICAL SCHOOL

Contents

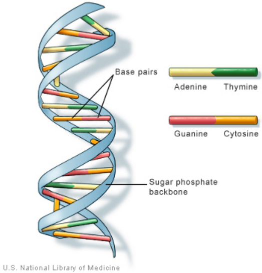
What are variants and how to represent them?

- Variants in the Human Genome
- The Variant Call Format (VCF)

Scalable genomic data analysis

- Scalability
- Hail, a genomic sariant store
- Scalability and cost-effectiveness analysis of whole genome wide association studies on Google Cloud Platform and Amazon Web Services

Variants in the Human Genome

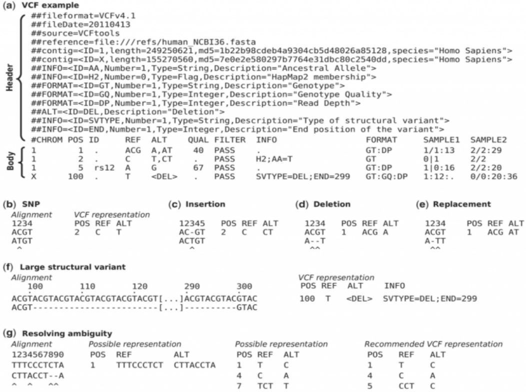


U.S. National Library of Medicine

- The human genome is made of 3.2 billion (3×10^9) base pairs, distributed across 20k genes and comprised in 23 pairs of chromosomes
- Only ~1.5% of the genome codes for protein
- DNA-wise, humans are 99.9% the same. The remaining 0.1% of these genetic variations, or **variants**, may give place to differences such as distinct skin, eye or hair color, but some of them may also be linked to or causative of disease
- Human mutation rate is estimated to be 2.5×10^{-8} mutations/bp/generation, which leads to 100-200 new variants in each newborn
- It is crucial to study and understand the possible effects of variants towards increasing risk for disease

The Variant Call Format (VCF)

Aim: to score the similarity of alleles (SNPs, indels) to reference genome in scalable data sets ▸ Click HERE for additional information ▸ VCF 4.1 Spec


Header
Body

(a) VCF example

```
##fileformat=VCFv4.1
##fileDate=20110413
##source=VCFtools
##reference=file:///refs/human_NCBI36.fasta
##contig=<ID=1,length=249250621,md5=1b22b98cdeb4a9304cb5d48026a85128,species="Homo Sapiens">
##contig=<ID=X,length=155270560,md5=7e0e2e580297b7764e31dbc80c2540dd,species="Homo Sapiens">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##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">
##ALT=<ID=DEL,Description="Deletion">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant">
#CHROM POS ID     REF ALT    QUAL FILTER INFO                FORMAT     SAMPLE1   SAMPLE2
1        1 .      ACG A,AT     40 PASS   .                   GT:DP      1/1:13    2/2:29
1        2 .      C   T,CT      . PASS   H2;AA=T             GT         0|1       2/2
1        5 rs12   A   G        67 PASS   .                   GT:DP      1|0:16    2/2:20
X      100 .      T   <DEL>     . PASS   SVTYPE=DEL;END=299  GT:GQ:DP   1:12:.    0/0:20:36
```

(b) SNP

Alignment *VCF representation*

```
1234      POS REF ALT
ACGT      2   C   T
ATGT
 ^
```

(c) Insertion

```
12345   POS REF ALT
AC-GT   2   C   CT
ACTGT
  ^
```

(d) Deletion

```
1234    POS REF ALT
ACGT    1   ACG A
A--T
 ^^
```

(e) Replacement

```
1234    POS REF ALT
ACGT    1   ACG AT
A-TT
 ^^
```

(f) Large structural variant

Alignment *VCF representation*

```
   100       110       120        290       300         POS REF  ALT    INFO
    .         .         .          .         .
ACGTACGTACGTACGTACGTACGTACGT[...]ACGTACGTACGTAC         100  T   <DEL>  SVTYPE=DEL;END=299
ACGT------------------------[...]----------GTAC
```

(g) Resolving ambiguity

Alignment *Possible representation* *Possible representation* *Recommended VCF representation*

```
1234567890    POS  REF       ALT        POS REF ALT      POS  REF  ALT
TTTCCCTCTA    1    TTTCCCTCT CTTACCTA   1   T   C        1    T    C
CTTACCT--A                              4   C   A        4    C    A
^  ^    ^^                              7   TCT T        5    CCT  C
```

Sections of a VCF file

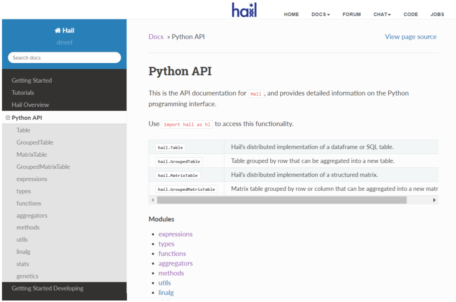
Scalability

- Genomic data grows fast. Analyzing it requires the use of clusters as often single machines can no longer process or store all these large data
- Clusters are flexible, as one can increase the number of working machines (cores) to easily boost computation power
- Data and system scalability involves virtual cluster deployment, monitoring and management in cloud environments
- An example of data scalability is the 1,000 Genome project, which has evolved ever since it was created:

1000 Genome Release	**Variants**	**Individuals**	**Populations**
Phase 3	84.4 million	2504	26
Phase 1	37.9 million	1092	14
Pilot 1	14.8 million	179	4

Hail, a genomic variant store

- Open-source, modular, scalable platform for statistical genetics in continuous development by the Neale lab at the Broad Institute
- It is exposed through Python and backed by distributed algorithms built on top of Apache Spark



Python API

This is the API documentation for `Hail`, and provides detailed information on the Python programming interface.

Use `import hail as hl` to access this functionality.

`hail.Table`	Hail's distributed implementation of a dataframe or SQL table.
`hail.GroupedTable`	Table grouped by row that can be aggregated into a new table.
`hail.MatrixTable`	Hail's distributed implementation of a structured matrix.
`hail.GroupedMatrixTable`	Matrix table grouped by row or column that can be aggregated into a new matr

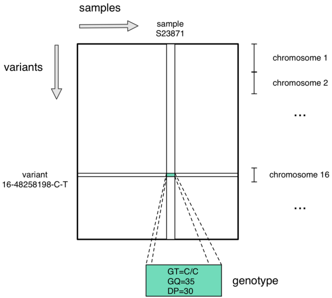
Modules

- expressions
- types
- functions
- aggregators
- methods
- utils
- linalg

Python API for Hail

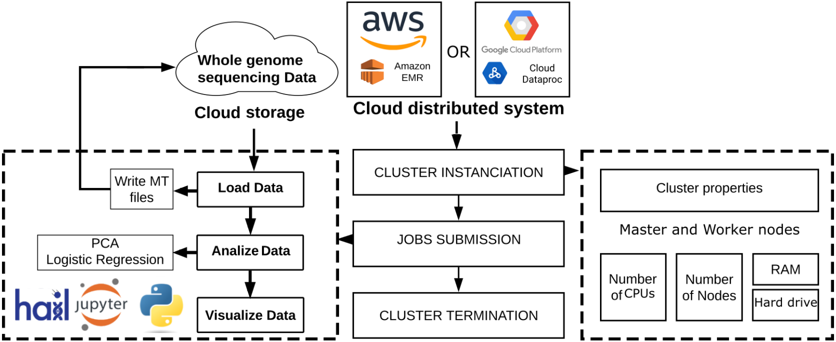
From VCF to MT file

A **Matrix Table (MT)** is a Hail specific data representation made by three tables: rows, columns and entries. Variant data is transformed into .mt format for processing in Hail. Samples are found in the columns table, alleles in the rows and format information from the VCF file in the entries table.



Hail Matrix Table Format

A distributed computational framework for large scale genomic analysis.



Additional Information

Documentation and support:

- Documentation, tutorials and sample code:
  `hail.is`
- Forum and chat:
  `discuss.hail.is`

Spinning clusters in cloud services:

- Hail Deployment in Google Cloud:
  `github.com/hms-dbmi/Hail-on-Google-Cloud`
- Hail Deployment in Amazon Web Services:
  `github.com/hms-dbmi/hail-on-AWS-spot-instances`