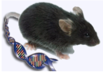
# Single Nucleotide Polymorphisms

Mikhail Dozmorov

Spring 2018

# How much do we differ? (number of aligned DNA base differences)

| | | |
|---|---|---|
| • Identical twins | | 0 |
| • Unrelated humans | | 1/1,000 |
| • Human vs. chimp | | 1/100 |
| • Human vs. mouse |  | 1/6 - 1/3 |

- 3 billion DNA bases → 3 million differences (single nucleotide variants [SNVs]) between each pair of haploid human DNA sequences

Human mutation rate is 1.0 – 1.5 x 10-8 per bp per generation: we transmit ~30 new DNA variants with each gamete (J. Roach et al., 2010, Science; D. Conrad et al., 2011, Nature Genetics)

# Genome diversity

- **SNPs** (Single Nucleotide Polymorphisms) - base substitutions
- In humans, occur approx. once per 1,000 bases ($\sim 3x10^6$ per genome)
- Most polymorphisms (~90%) take the form of SNPs: variations that involve just one nucleotide
- InDels (insertion/deletion, frameshifts) - occur in 1 in every 300 bp (human)

Average number of SNVs per individual

Orangutans 9.3 million > Gorillas 6.5 million > Chimpanzees 5.7 million > **Humans 3-4 million**

As a species, humans have relatively low diversity

(Prado-Martinez *et al.*, 2013, *Nature*)

# Functional Consequences

| Type | Consequence |
|---|---|
| SNPs in coding area that alter aa sequence | Cause of most monogenic disorders, e.g:<br>Cystic fibrosis (CFTR)<br>Hemophilia (F8) |
| SNPs in coding areas that don't alter amino acid sequence | May affect splicing |
| SNPs in promoter or regulatory regions | May affect the level, location or timing of gene expression |
| SNPs in other regions | No direct known impact on phenotype<br>Useful as markers |

# Studying variation – why?

- SNPs can cause disease
  - SNP in clotting factor IX codes for a stop codon: haemophilia
- SNPs can increase disease risk
  - SNP in LDL receptor reduces efficiency: high cholesterol
- SNPs can affect drug response
  - SNP in CYP2D8, a gene in the drug breakdown pathway in the liver, disrputs breakdown of debrisoquine, a treatment for high blood pressure

# Studying variation – why?

- Determine disease risk
- Individualised medicine (pharmacogenomics)
- Forensic studies
- Biological markers
- Hybridisation studies, marker-assisted breeding
- Understanding evolution

# Reference Sequence

- The Human Genome Project gave the “average” DNA sequence of a small number of people.
- This helps us find out how a human develops and works
- Does not show us the DNA differences between different humans
- Does not reflect the major alleles

# 1000 genomes

- Delivering 20TB of sequence data
- First Pilot. 60 HapMap samples sequenced (low coverage)
- Second Pilot. Two trios of European and African descent (high coverage)
- Third Pilot. Sequence 1,000 genes in 1,000 individuals (high coverage)

# Haplotypes

- Adjacent SNPs are often highly correlated, occurring together in individuals of similar ancestry
- These combinations of adjacent SNPs are termed **haplotypes**
- A **haplotype** is a set of SNPs (on average ~25 kb) found to be statistically associated on a single chromatid and which therefore tend to be inherited together over time.
- The International HapMap (haplotype mapping) project was launched in 2002 and provided critical insight regarding differences in the SNP frequencies and genome-wide haplotypes of different ethnic groups worldwide
- Used for grouping subjects by haplotypes.

www.hapmap.org

# HapMap (phase I & II)

Samples from populations with African, Asian and European ancestry.

- 270 DNA samples from 4 populations:
- 30 trios (two parents and an adult child) from the Yoruba people of Ibadan, Nigeria
- 45 unrelated Japanese from the Tokyo area
- 45 unrelated Han Chinese from Beijing
- 30 trios from Utah with Northern and Western European ancestry (CEPH)

# HapMap (phase III)

Genotypes from 1115 individual from 11 populations:

- ASW African ancestry in Southwest USA (71)
- CEU Utah residents with Northern and Western European ancestry from the CEPH collection (162)
- CHB Han Chinese in Beijing, China (70)
- CHD Chinese in Metropolitan Denver, Colorado (70)
- GIH Gujarati Indians in Houston, Texas (83)
- JPT Japanese in Tokyo, Japan (82)
- LWK Luhya in Webuye, Kenya (83)
- MEX Mexican ancestry in Los Angeles, California (71)
- MKK Maasai in Kinyawa, Kenya (171)
- TSI Toscani in Italia (77)
- YRI Yoruba in Ibadan, Nigeria (163)

# dbSNP

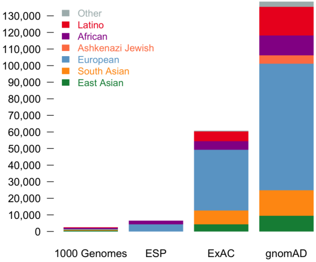
Central repository for simple genetic polymorphisms:

- single-base nucleotide substitutions
- small-scale multi-base deletions or insertions
- retroposable element insertions and microsatellite repeat variations
- For human (dbSNP build 151)
  - 907.2 Million submissions (submitter SNPs, ss#'s)
  - 325.7 Million unique submitted SNPs (reference SNPs, rs#'s)

http://www.ncbi.nlm.nih.gov/SNP/

# Other SNP resources

- **ExAC** - the Exome Aggregation Consortium (ExAC), a coalition of investigators seeking to aggregate and harmonize exome sequencing data from a wide variety of large-scale sequencing projects, and to make summary data available for the wider scientific community
- **gnomAD** - the Genome Aggregation Database (gnomAD), adds whole genome variants



http://exac.broadinstitute.org/, ftp://ftp.broadinstitute.org/pub/ExAC_release

# SNP types

- **Non-synonymous** - In coding sequence, resulting in an aa change
- **Synonymous** - In coding sequence, not resulting in an aa change
- **Frameshift** - In coding sequence, resulting in a frameshift
- **Stop lost** - In coding sequence, resulting in the loss of a stop codon
- **Stop gained** - In coding sequence, resulting in the gain of a stop codon

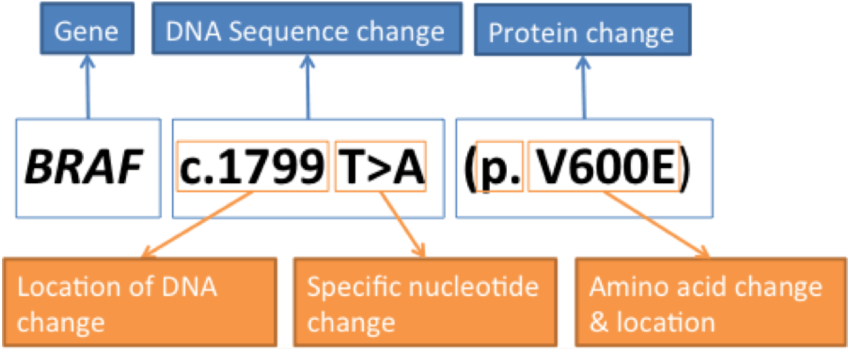
How ClinVar defines its clinical significance values. https://www.ncbi.nlm.nih.gov/clinvar/docs/clinsig/

# SNP types

- **Essential splice site** - In the first 2 or the last 2 basepairs of an intron
- **Splice site** - 1-3 bps into an exon or 3-8 bps into an intron
- **Upstream** - Within 5 kb upstream of the 5'-end of a transcript
- **Regulatory region** - In regulatory region annotated by Ensembl
- **5' UTR** - In 5' UTR
- **Intronic** - In intron
- **3' UTR** - In 3' UTR
- **Downstream** - Within 5 kb downstream of the 3'-end of a transcript
- **Intergenic** - More than 5 kb away from a transcript

# Sequence Variant Nomenclature

- Human Genome Variation Society nomenclature.
  - Example: NM_004006.1:c.[145C>T;147C>G] - two substitutions replacing codon CGC (position c.145 to c.147) by TGG

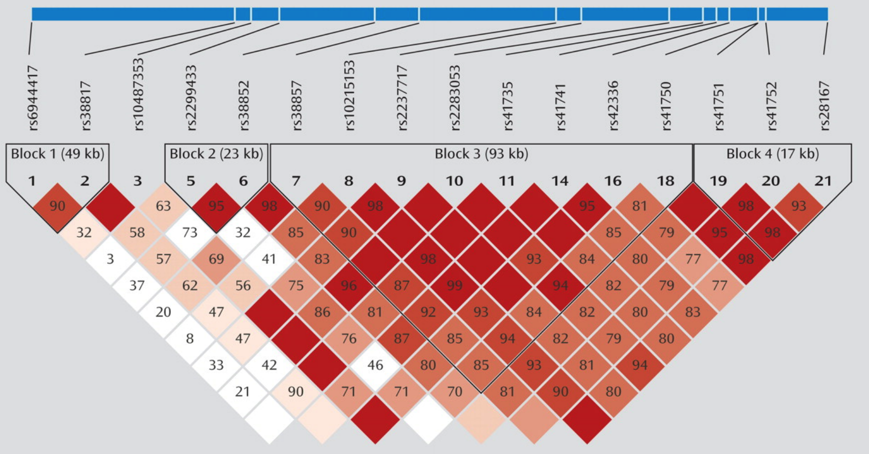


http://varnomen.hgvs.org/

# Linkage Disequilibrium

- **LINKAGE DISEQUILIBRIUM** - Correlation between nearby variants such that the alleles at neighbouring markers (observed on the same chromosome) are associated within a population more often than if they were unlinked.
- LD is the deviation from equilibrium, or random association. (i.e. in a population, two alleles are always inherited together, though they should undergo recombination some of the time.)

# Linkage disequilibrium



LD values between two variants are displayed by means of inverted coloured triangles going from white (low LD) to red (high LD)

# Measures of LD

$D = P(AB) - P(A)P(B)$

- $D$ ranges from – 0.25 to $+$ 0.25
- $D = 0$ indicates linkage equilibrium
- dependent on allele frequencies, therefore of little use

$D' = D/maximum\ possible\ value$

- $D' = 1$ indicates perfect LD
- estimates of D' strongly inflated in small samples

$r^2 = D^2/P(A)P(B)P(a)P(b)$

- $r^2 = 1$ indicates perfect LD
- measure of choice

# Tag SNPs

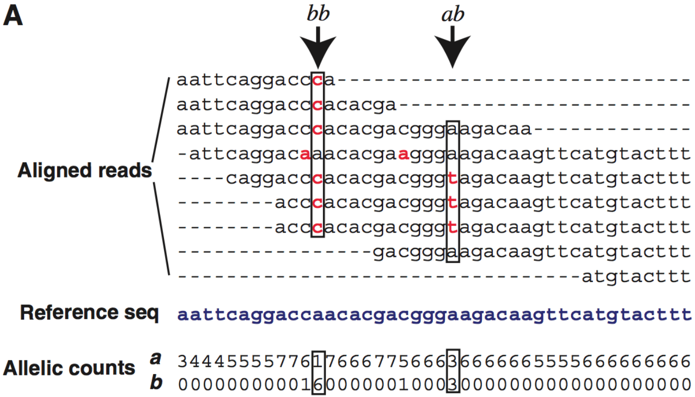
- **HAPLOTYPE** - A sequential set of genetic markers that are present on the same chromosome.
- **TAG SNPs** - Single nucleotide polymorphisms that are correlated with, and therefore can serve as a proxy for, much of the known remaining common variation in a region.
    - Tag SNPs define the minimum SNP set to identify a haplotype

$r^2 = 1$ between two SNPs means one would be ‘redundant’ in the haplotype.

https://estrip.org/articles/read/tinypliny/44920/Linkage_Disequilibrium_Blocks_Triangles.html

# Genotypes

- **Homozygote** - a SNP having **two identical alleles** of a particular gene or genes
- **Heterozygote** - a SNP having **two different alleles** of a particular gene or genes



```
aattcaggaccca-----------------------------
aattcaggacccacacga------------------------
aattcaggacccacacgacgggaagacaa-------------
-attcaggacaaacacgaagggaagacaagttcatgtacttt
----caggacccacacgacgggtagacaagttcatgtacttt
--------acccacacgacgggtagacaagttcatgtacttt
--------acccacacgacgggtagacaagttcatgtacttt
----------------gacgggaagacaagttcatgtacttt
---------------------------------atgtacttt

aattcaggaccaacacgacgggaagacaagttcatgtacttt

a 344455557761766677566636666665555666666666
b 000000000016000000100030000000000000000000
```

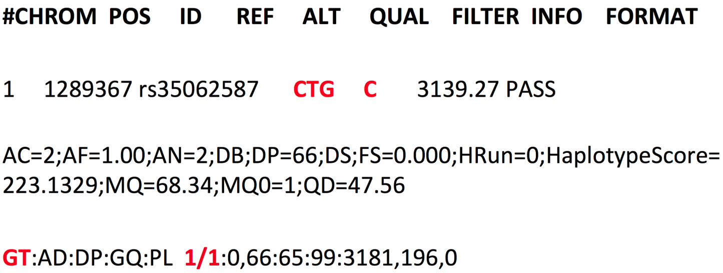
# Homo

```
#CHROM  POS     ID      REF   ALT   QUAL    FILTER  INFO   FORMAT

1   801943  rs7516866    C     T     9787.34 PASS

AC=2;AF=1.00;AN=2;BaseQRankSum=1.009;DB;DP=556;DS;Dels=0.00;
FS=18.302;HRun=1;HaplotypeScore=4.6410;MQ=44.04;MQ0=38;MQR
ankSum=5.122;QD=17.60;ReadPosRankSum=3.375

GT:AD:DP:GQ:PL  1/1:37,518:556:99:9787,685,0
```

# Hetero

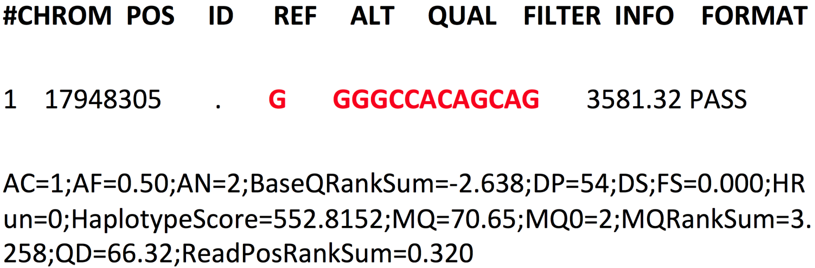
```
#CHROM  POS    ID     REF   ALT   QUAL   FILTER  INFO   FORMAT

1   1918488 rs4350140     A     G      233.10  PASS

AC=1;AF=0.50;AN=2;BaseQRankSum=1.349;DB;DP=33;DS;Dels=0.00;
FS=0.000;HRun=0;HaplotypeScore=0.0000;MQ=68.18;MQ0=1;MQRa
nkSum=0.436;QD=7.06;ReadPosRankSum=1.547

GT:AD:DP:GQ:PL  0/1:21,12:33:99:263,0,620
```

# HomDel



```
#CHROM  POS     ID     REF    ALT    QUAL    FILTER  INFO    FORMAT

1    1289367 rs35062587    CTG    C      3139.27 PASS

AC=2;AF=1.00;AN=2;DB;DP=66;DS;FS=0.000;HRun=0;HaplotypeScore=
223.1329;MQ=68.34;MQ0=1;QD=47.56

GT:AD:DP:GQ:PL  1/1:0,66:65:99:3181,196,0
```

# HetetroIns

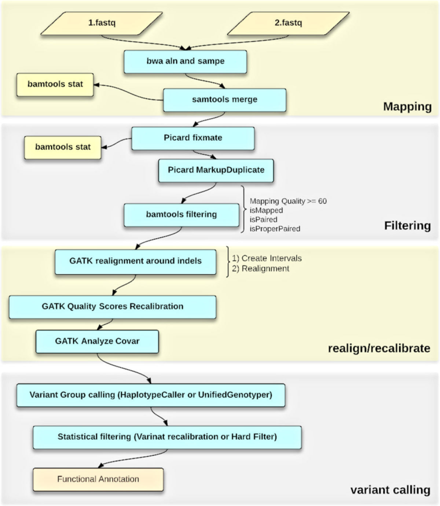


```
#CHROM  POS       ID    REF   ALT           QUAL      FILTER  INFO   FORMAT

1   17948305      .     G     GGGCCACAGCAG       3581.32 PASS

AC=1;AF=0.50;AN=2;BaseQRankSum=-2.638;DP=54;DS;FS=0.000;HR
un=0;HaplotypeScore=552.8152;MQ=70.65;MQ0=2;MQRankSum=3.
258;QD=66.32;ReadPosRankSum=0.320
```

# Genotypes summary

```
#CHROM  POS        ID  REF  ALT  QUAL FILTER  INFO     FORMAT  LF1396
chr7    117175373  .   A    G    90   PASS    AF=0.0   GT      0/0      Hom. Ref.

chr7    117175373  .   A    G    90   PASS    AF=0.5   GT      0/1      Het.

chr7    117175373  .   A    G    90   PASS    AF=1.0   GT      1/1      Hom. Alt.

chr7    117175373  .   A    G    90   PASS    AF=0.0   GT      ./.      Unknown
```

# SNP calling pipeline



https://humgenomics.biomedcentral.com/articles/10.1186/1479-7364-8-14

# SNP calling

- Algorithms should have high power to detect a wide range of variation, including single- and multiple-nucleotide variants (SNVs and MNVs) and structural variation including indels, sequence replacements and mobile element insertions
- Must have low false discovery rates (FDRs) to minimize costly validation experiments
- Should be able to cope with challenging loci, including highly repetitive sequence and reference errors, and be robust to high levels of local diversity to access clinically interesting regions such as the human leukocyte antigen (HLA) loci
- Should have low resource requirements and run on commodity hardware while achieving fast turnaround times

# SNP calling

- The most common approach is to map reads to a reference genome and either scan for systematic differences with the reference or identify haplotypes that are well supported by the data
  - **Strengths:** Highly sensitive, use common reference, use paired-end information, low on computations
  - **Weaknesses:** Focus on single-base variants, fail in highly divergent regions, e.g., Human Leukocyte Antigen region, require realignment around known indels, computationally high

# SNP calling

- A complementary approach is reference-free sequence assembly - de Bruijn or overlap graphs
- Search this data structure for evidence of polymorphisms
    - **Strengths:** By not relying on a reference genome, this approach is variant agnostic, copes well with highly divergent regions, naturally works on the local haplotype level rather than on the level of individual variants and avoids the need for an initial mapping and alignment step
    - **Weaknesses:** high computational requirements, lower sensitivity than mapping-based approaches, limited by repetitive sequence, as contiguity information is lost when the reads are broken up into their consecutive k-mers during graph construction

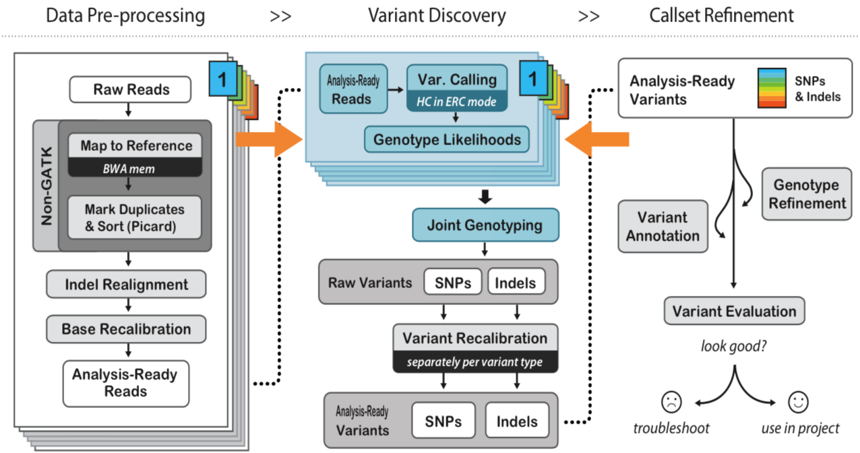
# Genome Analysis Toolkit

- A single framework and the associated tools capable of discovering high-quality variation and genotyping individual samples using diverse sequencing machines and experimental designs
  - Initial read mapping;
  - Local realignment around indels;
  - Base quality score recalibration;
  - SNP discovery and genotyping to find all potential variants;
  - Machine learning to separate true segregating variation from machine artifacts common to next-generation sequencing technologies.

https://software.broadinstitute.org/gatk/

DePristo, Mark A., Eric Banks, Ryan Poplin, Kiran V. Garimella, Jared R. Maguire, Christopher Hartl, Anthony A. Philippakis, et al. "A Framework for Variation Discovery and Genotyping Using Next-Generation DNA Sequencing Data." Nature Genetics 43, no. 5 (May 2011): 491–98. https://doi.org/10.1038/ng.806.

# GATK Variant Calling Best Practices



Data Pre-processing >> Variant Discovery >> Callset Refinement

Raw Reads

Non-GATK

Map to Reference

*BWA mem*

Mark Duplicates & Sort (Picard)

Indel Realignment

Base Recalibration

Analysis-Ready Reads

1

Analysis-Ready Reads

Var. Calling

*HC in ERC mode*

1

Genotype Likelihoods

Joint Genotyping

Raw Variants

SNPs

Indels

Variant Recalibration

*separately per variant type*

Analysis-Ready Variants

SNPs

Indels

Analysis-Ready Variants

SNPs & Indels

Genotype Refinement

Variant Annotation

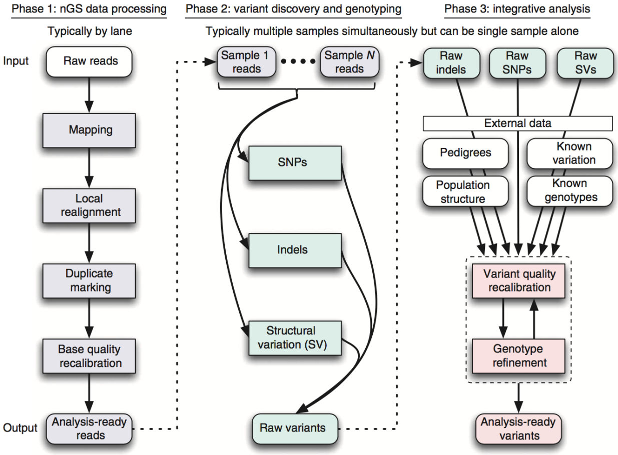
Variant Evaluation

*look good?*

☹ *troubleshoot*

☺ *use in project*

# Genome Analysis Toolkit



https://software.broadinstitute.org/gatk/

DePristo, Mark A., Eric Banks, Ryan Poplin, Kiran V. Garimella, Jared R. Maguire, Christopher Hartl, Anthony A. Philippakis, et al. "A Framework for Variation Discovery and Genotyping Using Next-Generation DNA Sequencing Data." Nature Genetics 43, no. 5 (May 2011): 491–98. https://doi.org/10.1038/ng.806.

# Genome Analysis Toolkit

- Excellent documentation, tutorials, best practices guidelines
- Cloud-ready and parallelizable
- Current version - GATK4
    - Uses Mutect2 algorithm
    - Adapted for better CNV/SV detection
    - Neural network for variant filtering

https://software.broadinstitute.org/gatk/gatk4

# GATK HaplotypeCaller

- Jointly calling variants on multiple samples
- Better detects insertions and deletions
- Produces square matrix with samples vs. variants calls
- Algorithm:
  - defining "Active regions" with high coverage
  - local reassembly using de Bruijn graph
  - hidden Markov Model to identify match, insertion, or deletion
  - haplotype calling based on CIGAR information using Bayesian model

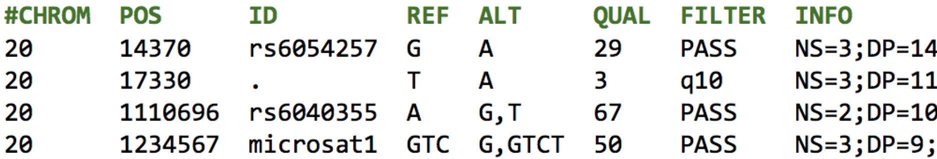
Poplin, Ryan, Valentin Ruano-Rubio, Mark A. DePristo, Tim J. Fennell, Mauricio O. Carneiro, Geraldine A. Van der Auwera, David E. Kling, et al. "Scaling Accurate Genetic Variant Discovery to Tens of Thousands of Samples." BioRxiv, January 1, 2017. https://doi.org/10.1101/201178.

# Filtering

The rationale for filtering

- To eliminate False Positive variants from variant list
- What causes errors in variant calling?
  - **Sequencing errors** - should be accounted for by base quality + recalibration + marking of duplicates
  - **Incorrect alignment** - Re-alignment step should have reduced this problem but not eliminated it
- Thus although QUAL (which depends on Mapping Quality of reads and Base qualities) is a useful measure, there will still be FP with high QUAL

# VCF annotation



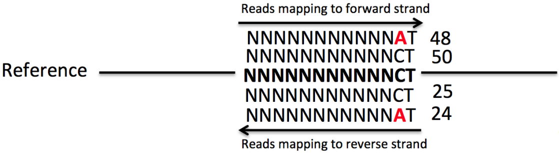
```
#CHROM  POS      ID         REF  ALT    QUAL  FILTER  INFO
20      14370    rs6054257  G    A      29    PASS    NS=3;DP=14
20      17330    .          T    A      3     q10     NS=3;DP=11
20      1110696  rs6040355  A    G,T    67    PASS    NS=2;DP=10
20      1234567  microsat1  GTC  G,GTCT 50    PASS    NS=3;DP=9;
```

# INFO fields – important for filtering

- QD: variant quality score over depth
    - Confidence in the site being variant should increase with increasing depth
- MQ: Root Mean Square of MAPQ of all reads at locus
    - Regions of excessively low mapping quality are ambiguously mapped and variants called within are suspicious
- MQ0: number of MAPQ 0 reads at locus
- MQRankSum: Mapping quality rank sum test
    - If the alternate bases are more likely to be found on reads with lower MAPQ than reference bases then the site is likely mismapped
- Haplotype score: Probability that the reads in a window around the variant can be explained by at most two haplotypes
- FS: fisher exact test of read strand
    - If the reference-carrying reads are balanced between forward and reverse strands then the alternate-carrying reads should be as well
- ReadPosRankSum: Read position rank sum test
    - If the alternate bases are biased towards the beginning or end of the reads then the site is likely a mapping artifact

# Strand bias (assume heterozygote)

Strand bias **is NOT** about more reads mapping to one of the strands than the other



Reads mapping to forward strand

NNNNNNNNNNNAT 48
NNNNNNNNNNNCT 50
Reference **NNNNNNNNNNNCT**
NNNNNNNNNNNCT 25
NNNNNNNNNNNAT 24

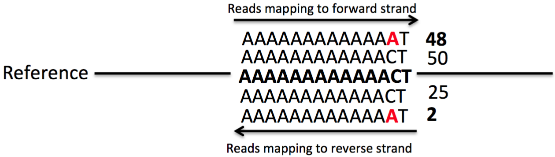
Reads mapping to reverse strand

| | | Fw | Rev |
|---|---|---|---|
| **Ref** | C | 50 | 25 |
| **Alt** | A | 48 | 24 |

Clearly more reads mapping to FW than Rev

**But**, Fw/Rev ratio is same for Ref allele and Alt: 50/25 = 48/24

This **IS** strand bias



Reads mapping to forward strand

AAAAAAAAAAAAT **48**
AAAAAAAAAAAACT 50
Reference **AAAAAAAAAAAACT**
AAAAAAAAAAAACT 25
AAAAAAAAAAAAAT **2**

Reads mapping to reverse strand

| | | Fw | Rev |
|---|---|---|---|
| **Ref** | C | 50 | 25 |
| **Alt** | A | 48 | 2 |

**50/25 != 48/2**

# Hard vs. soft filtering

- Can set thresholds for the relevant INFO fields and request that all thresholds are passed for a variant to be considered valid
- Which fields to you use and where do you set the thresholds? – use datasets of known SNPs and compare their INFO fields to those likely FP variants
- Disadvantage of hard filtering – loosely justified hard cut-offs
- Variant Quality Score Recalibration (GATK) or soft filtering

# VCF files: normalization

- The VCF format is quite precise but still leaves room for representing one variant in multiple ways - normalization (harmonization) of variant representation is needed
- **Parsimony**
  - Pos: 5, Ref: `ATC`, Alt: `AT`
  - **Or** Pos: 6, Ref: `TC`, Alt: `T` >> most parsimonious
- Left alignment, suppose context: pos 8, ref: `ATTTT`, `T` deletion
  - Pos: 10, Ref: `TT`, Alt: `T`
  - **Or** Pos: 8, Ref: `AT`, Alt: `A` >> left aligned
- **MNP on separate lines**
  - 150 `TCT` `CCC` - Can be decomposed into two records: 150 `T` `C` AND 152 `T` `C`
- One should also ensure that the same reference naming is used in both comparison files and that both files have the same sort order

https://github.com/chapmanb/bcbio.variation/wiki/Normalized-variant-representation

http://genome.sph.umich.edu/wiki/Variant_Normalization

# Other VCF issues

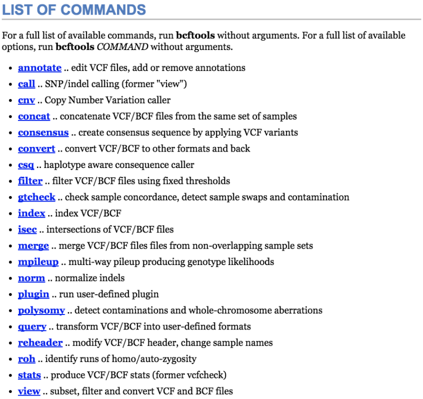
- Chromosome labeling: chr1, chr2 . . . vs. 1, 2, X, Y, M
- Chromosome ordering: 1, 2, 3, 4 . . . vs. 1, 10, 11, . . .
- GATK enforcement of “X, Y, MT” sorting vs. “MT, X, Y”

# vcflib - a simple C++ library for parsing and manipulating VCF files, + many command-line utilities

- Comparison: intersection, overlay-merge, combine, validate
- Format conversion: to tab-separated, BED formats
- Filtering: using the INFO and sample fields, random sampling, select by criteria
- Annotation: one VCF with INFO fields from another VCF, from BED, annotate by distance
- Samples: extract sample names, remove samples
- Ordering: sort, remove duplicates
- Variant representation: complex variants harmonization
- Statistics and EDA: summary stats, entropy, heterozygosity rate, classify variants

https://github.com/vcflib/vcflib

# bcftools — utilities for variant calling and manipulating VCFs and BCFs



## LIST OF COMMANDS

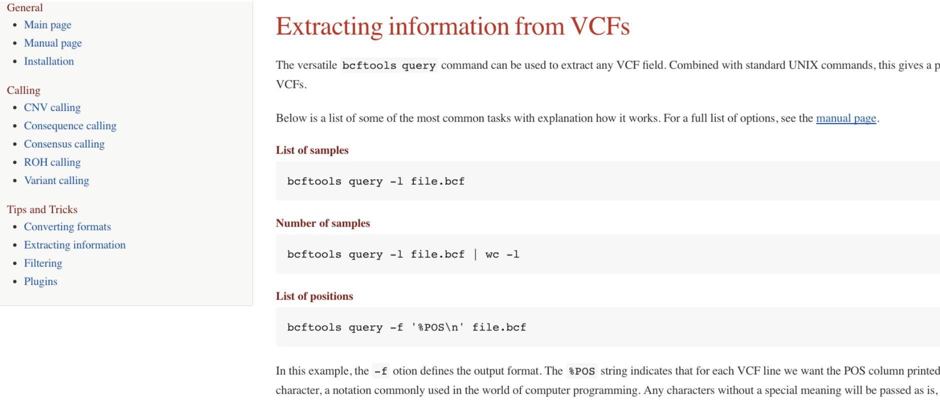
For a full list of available commands, run **bcftools** without arguments. For a full list of available options, run **bcftools** *COMMAND* without arguments.

- **annotate** .. edit VCF files, add or remove annotations
- **call** .. SNP/indel calling (former "view")
- **cnv** .. Copy Number Variation caller
- **concat** .. concatenate VCF/BCF files from the same set of samples
- **consensus** .. create consensus sequence by applying VCF variants
- **convert** .. convert VCF/BCF to other formats and back
- **csq** .. haplotype aware consequence caller
- **filter** .. filter VCF/BCF files using fixed thresholds
- **gtcheck** .. check sample concordance, detect sample swaps and contamination
- **index** .. index VCF/BCF
- **isec** .. intersections of VCF/BCF files
- **merge** .. merge VCF/BCF files files from non-overlapping sample sets
- **mpileup** .. multi-way pileup producing genotype likelihoods
- **norm** .. normalize indels
- **plugin** .. run user-defined plugin
- **polysomy** .. detect contaminations and whole-chromosome aberrations
- **query** .. transform VCF/BCF into user-defined formats
- **reheader** .. modify VCF/BCF header, change sample names
- **roh** .. identify runs of homo/auto-zygosity
- **stats** .. produce VCF/BCF stats (former vcfcheck)
- **view** .. subset, filter and convert VCF and BCF files

http://www.htslib.org/doc/bcftools.html

Li, Heng. "A Statistical Framework for SNP Calling, Mutation Discovery, Association Mapping and Population Genetical Parameter Estimation from Sequencing Data." Bioinformatics (Oxford, England) 27, no. 21 (November 1, 2011): 2987–93. https://doi.org/10.1093/bioinformatics/btr509.

# bcftools examples



General

- Main page
- Manual page
- Installation

Calling

- CNV calling
- Consequence calling
- Consensus calling
- ROH calling
- Variant calling

Tips and Tricks

- Converting formats
- Extracting information
- Filtering
- Plugins

## Extracting information from VCFs

The versatile `bcftools query` command can be used to extract any VCF field. Combined with standard UNIX commands, this gives a p VCFs.

Below is a list of some of the most common tasks with explanation how it works. For a full list of options, see the manual page.

**List of samples**

```
bcftools query -l file.bcf
```

**Number of samples**

```
bcftools query -l file.bcf | wc -l
```

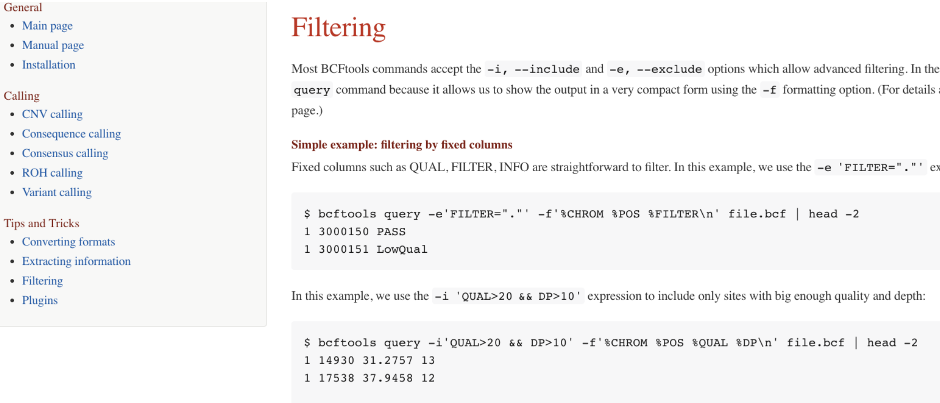
**List of positions**

```
bcftools query -f '%POS\n' file.bcf
```

In this example, the `-f` otion defines the output format. The `%POS` string indicates that for each VCF line we want the POS column printed character, a notation commonly used in the world of computer programming. Any characters without a special meaning will be passed as is,

https://samtools.github.io/bcftools/howtos/index.html

# bcftools examples



General

- Main page
- Manual page
- Installation

Calling

- CNV calling
- Consequence calling
- Consensus calling
- ROH calling
- Variant calling

Tips and Tricks

- Converting formats
- Extracting information
- Filtering
- Plugins

## Filtering

Most BCFtools commands accept the `-i, --include` and `-e, --exclude` options which allow advanced filtering. In the `query` command because it allows us to show the output in a very compact form using the `-f` formatting option. (For details page.)

### Simple example: filtering by fixed columns

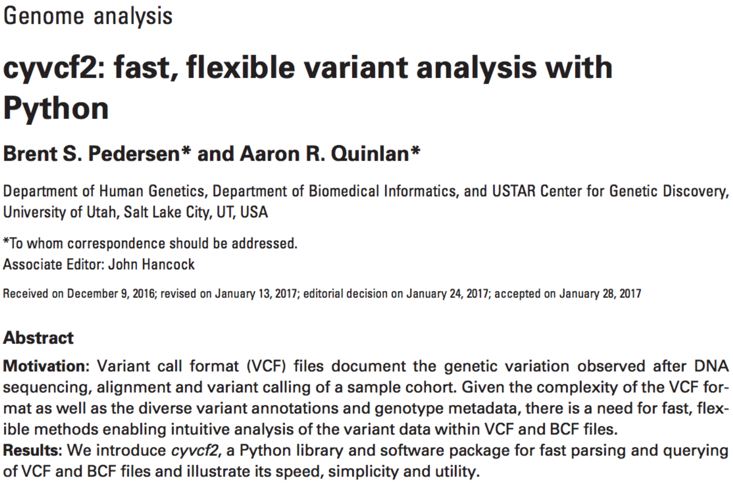
Fixed columns such as QUAL, FILTER, INFO are straightforward to filter. In this example, we use the `-e 'FILTER="."'` ex

```
$ bcftools query -e'FILTER="."' -f'%CHROM %POS %FILTER\n' file.bcf | head -2
1 3000150 PASS
1 3000151 LowQual
```

In this example, we use the `-i 'QUAL>20 && DP>10'` expression to include only sites with big enough quality and depth:

```
$ bcftools query -i'QUAL>20 && DP>10' -f'%CHROM %POS %QUAL %DP\n' file.bcf | head -2
1 14930 31.2757 13
1 17538 37.9458 12
```

# cyvcf2



Genome analysis

## cyvcf2: fast, flexible variant analysis with Python

**Brent S. Pedersen* and Aaron R. Quinlan***

Department of Human Genetics, Department of Biomedical Informatics, and USTAR Center for Genetic Discovery, University of Utah, Salt Lake City, UT, USA

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on December 9, 2016; revised on January 13, 2017; editorial decision on January 24, 2017; accepted on January 28, 2017

### Abstract

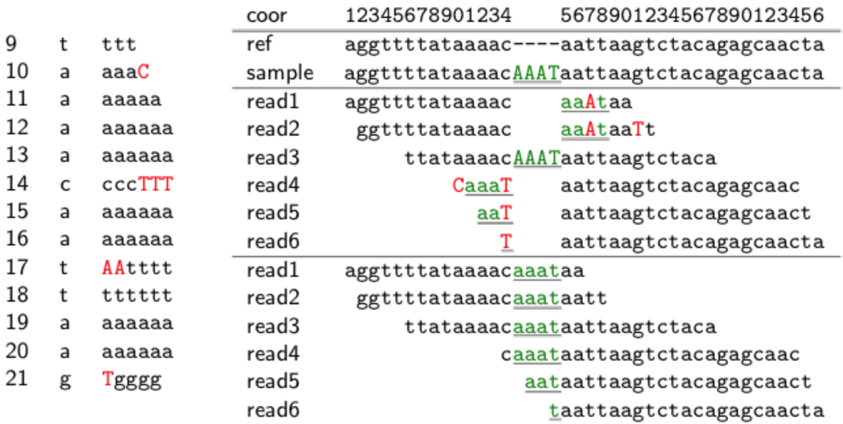
**Motivation:** Variant call format (VCF) files document the genetic variation observed after DNA sequencing, alignment and variant calling of a sample cohort. Given the complexity of the VCF format as well as the diverse variant annotations and genotype metadata, there is a need for fast, flexible methods enabling intuitive analysis of the variant data within VCF and BCF files.

**Results:** We introduce *cyvcf2*, a Python library and software package for fast parsing and querying of VCF and BCF files and illustrate its speed, simplicity and utility.

https://academic.oup.com/bioinformatics/article/2971439/

https://brentp.github.io/cyvcf2/

# Alignment errors during mapping require fix

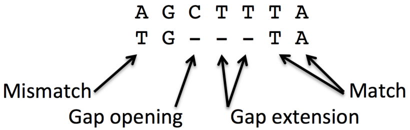


| | | |
|---|---|---|
| 9 | t | ttt |
| 10 | a | aaaC |
| 11 | a | aaaaa |
| 12 | a | aaaaaa |
| 13 | a | aaaaaa |
| 14 | c | cccTTT |
| 15 | a | aaaaaa |
| 16 | a | aaaaaa |
| 17 | t | AAtttt |
| 18 | t | tttttt |
| 19 | a | aaaaaa |
| 20 | a | aaaaaa |
| 21 | g | Tgggg |

```
coor     12345678901234    5678901234567890123456
--------------------------------------------------
ref      aggttttataaaac----aattaagtctacagagcaacta
sample   aggttttataaaacAAATaattaagtctacagagcaacta
--------------------------------------------------
read1    aggttttataaaac    aaAtaa
read2     ggttttataaaac    aaAtaaTt
read3         ttataaaacAAATaattaagtctaca
read4             CaaaT    aattaagtctacagagcaac
read5               aaT    aattaagtctacagagcaact
read6                 T    aattaagtctacagagcaacta
--------------------------------------------------
read1    aggttttataaaacaaataa
read2     ggttttataaaacaaataatt
read3         ttataaaacaaataattaagtctaca
read4                 caaataattaagtctacagagcaac
read5                   aataattaagtctacagagcaact
read6                     taattaagtctacagagcaacta
```

# Alignment

- Key component of alignment algorithm is the scoring
  - negative contribution to score
    - opening a gap
    - extending a gap
    - mismatches
  - positive contribution to score
    - matches



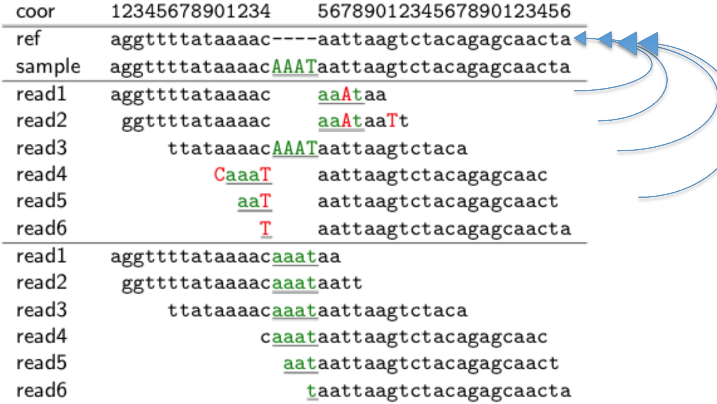
- When aligning two sequences there **is only one set of differences** to consider

- In a multiple sequence alignment, **one has to consider all pairs of differences** in the scoring algorithm

# Few mismatches when considering one-to-one

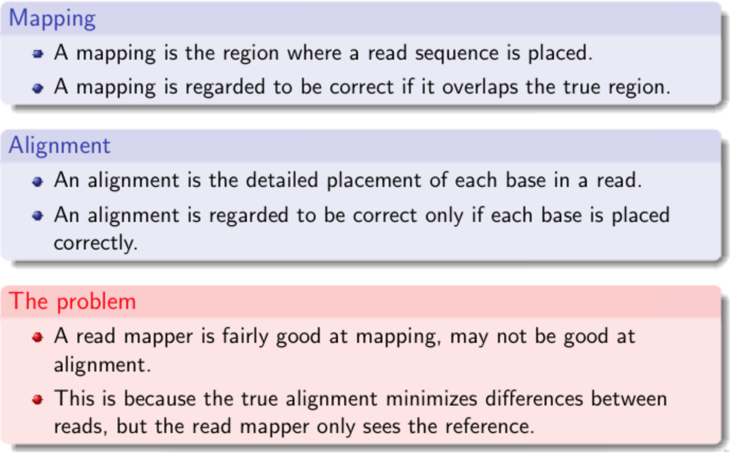
**Base stacks**

| | | |
|---|---|---|
| 9 | t | ttt |
| 10 | a | aaaC |
| 11 | a | aaaaa |
| 12 | a | aaaaaa |
| 13 | a | aaaaaa |
| 14 | c | cccTTT |
| 15 | a | aaaaaa |
| 16 | a | aaaaaa |
| 17 | t | AAtttt |
| 18 | t | tttttt |
| 19 | a | aaaaaa |
| 20 | a | aaaaaa |
| 21 | g | Tgggg |



```
coor   12345678901234    5678901234567890123456
ref    aggttttataaaac----aattaagtctacagagcaacta
sample aggttttataaaacAAATaattaagtctacagagcaacta
read1  aggttttataaaac    aaAtaa
read2   ggttttataaaac    aaAtaaTt
read3       ttataaaacAAATaattaagtctaca
read4           CaaaT    aattaagtctacagagcaac
read5             aaT    aattaagtctacagagcaact
read6               T    aattaagtctacagagcaacta
read1  aggttttataaaacaaataa
read2   ggttttataaaacaaataatt
read3       ttataaaacaaataattaagtctaca
read4               caaataattaagtctacagagcaac
read5                 aataattaagtctacagagcaact
read6                   taattaagtctacagagcaacta
```

# Mapping vs. alignment



### Mapping

- A mapping is the region where a read sequence is placed.
- A mapping is regarded to be correct if it overlaps the true region.

### Alignment

- An alignment is the detailed placement of each base in a read.
- An alignment is regarded to be correct only if each base is placed correctly.

### The problem

- A read mapper is fairly good at mapping, may not be good at alignment.
- This is because the true alignment minimizes differences between reads, but the read mapper only sees the reference.

# Local realignment around indels

Sequence aligners are often unable to perfectly map reads containing insertions or deletions (indels)

- Indel-containing reads can be either left unmapped or arranged in gapless alignments
- Mismatches in a particular read can interfere with the gap, esp. in low-complexity regions
- Single-read alignments are "correct" in a sense that they do provide the best guess given the (limited!) information and constrains.

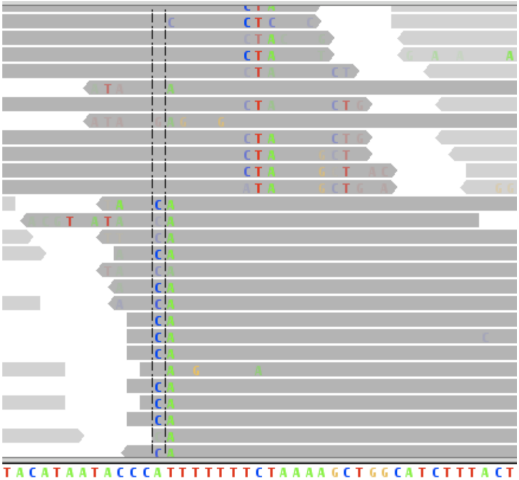
Major issues:

- Indel detection becomes difficult with so many missing reads
- Indels can be overlooked or misplaced in individual reads
- Artifacts introduced by the gapless alignments cause the appearance of false positive SNPs (usually in clusters)

https://www.broadinstitute.org/files/shared/mpg/nextgen2010/nextgen_sivachenko.pdf

# Example: SNP clusters are really a hidden indel

Before MSA realignment:


T A C A T A A T A C C C A T T T T T T T C T A A A A G C T G G C A T C T T T A C T

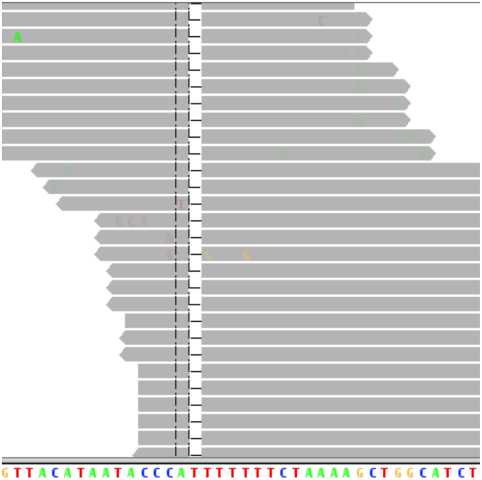
- Notice that the “SNP”s are all found in clusters

- Notice that the “SNP”s change depending on which end of the read span them

- Most likely what you’re looking at is a 1bp deletion (see next slide); the aligner is unable to accurately align the reads here

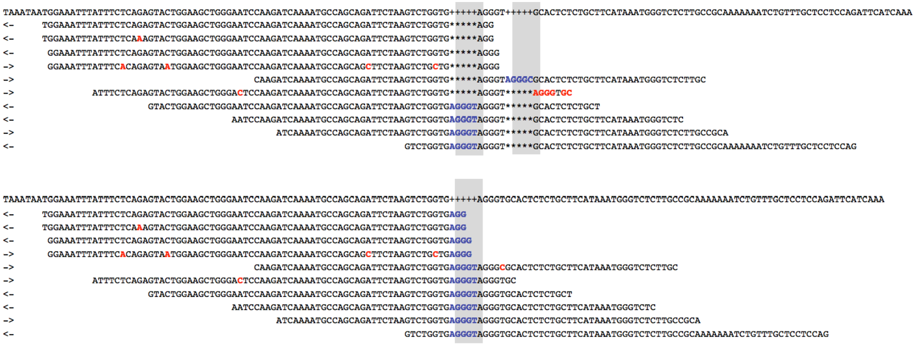
# Example: SNP clusters are really a hidden indel

After MSA realignment:



- SNP clusters disappear when it is run through our MSA realigner…

# Example : Indel “scatter”



```
TAAATAATGGAAATTTATTTCTCAGAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG+++++AGGGT+++++GCACTCTCTGCTTCATAAATGGGTCTCTTGCCGCAAAAAAATCTGTTTGCTCCTCCAGATTCATCAAA
<-     TGGAAATTTATTTCTCAGAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG*****AGG
<-     TGGAAATTTATTTCTCAAAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG*****AGG
<-      GGAAATTTATTTCTCAGAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG*****AGGG
->      GGAAATTTATTTCACAGAGTAATGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGCTTCTAAGTCTGCTG*****AGGG
->                                           CAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG*****AGGGTAGGGCGCACTCTCTGCTTCATAAATGGGTCTCTTGC
->              ATTTCTCAGAGTACTGGAAGCTGGGACTCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG*****AGGGT*****AGGGTGC
<-                        GTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGT*****GCACTCTCTGCT
<-                                       AATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGT*****GCACTCTCTGCTTCATAAATGGGTCTC
->                                               ATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGT*****GCACTCTCTGCTTCATAAATGGGTCTCTTGCCGCA
<-                                                                      GTCTGGTGAGGGTAGGGT*****GCACTCTCTGCTTCATAAATGGGTCTCTTGCCGCAAAAAAATCTGTTTGCTCCTCCAG

TAAATAATGGAAATTTATTTCTCAGAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTG+++++AGGGTGCACTCTCTGCTTCATAAATGGGTCTCTTGCCGCAAAAAAATCTGTTTGCTCCTCCAGATTCATCAAA
<-     TGGAAATTTATTTCTCAGAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGG
<-     TGGAAATTTATTTCTCAAAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGG
<-      GGAAATTTATTTCTCAGAGTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGG
->      GGAAATTTATTTCACAGAGTAATGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGCTTCTAAGTCTGCTGAGGG
->                                           CAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGCGCACTCTCTGCTTCATAAATGGGTCTCTTGC
->              ATTTCTCAGAGTACTGGAAGCTGGGACTCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGTGC
<-                        GTACTGGAAGCTGGGAATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGTGCACTCTCTGCT
<-                                       AATCCAAGATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGTGCACTCTCTGCTTCATAAATGGGTCTC
->                                               ATCAAAATGCCAGCAGATTCTAAGTCTGGTGAGGGTAGGGTGCACTCTCTGCTTCATAAATGGGTCTCTTGCCGCA
<-                                                                      GTCTGGTGAGGGTAGGGTGCACTCTCTGCTTCATAAATGGGTCTCTTGCCGCAAAAAAATCTGTTTGCTCCTCCAG
```

- A (heterogeneous) insertion + adjacent insertion may be clean homogeneous (?) insertion
- Even when aligner detects indels in individual reads successfully, they can be scattered around (e.g. due to additional mismtaches in the read)

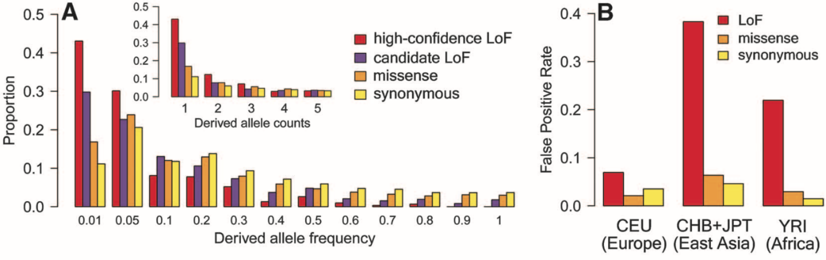
# What is annotation?

- Adding information about the variants
- Two broad categories of annotations
- annotations that depend on gene models
    - coding/non-coding
    - if coding: synonymous / non-synonymous
    - if non-synonymous - what is the impact on protein structure (Polyphen, SIFT, etc)
- annotations that do not depend on gene models
    - variant frequency in different databases / different populations
    - degree of conservation across species
- Considerable complications caused by different gene models
- Two approaches to problem
    - decide ex-ante what which transcript to use for each gene
    - annotate with all transcript for a given gene and pick the highest impact effect

# Loss of function (LoF) SNPs

- Genetic variants predicted to severely disrupt protein-coding genes, collectively known as loss-of-function (LoF) variants
- Typically rare
- Human genomes typically contain ~100 genuine LoF variants with ~20 genes completely inactivated

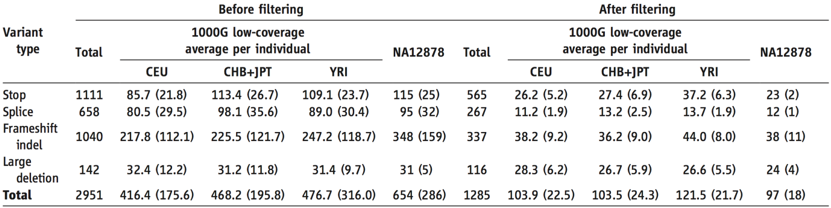
# Frequency of loss of function SNPs



http://science.sciencemag.org/content/335/6070/823

# Types of LoF SNPs

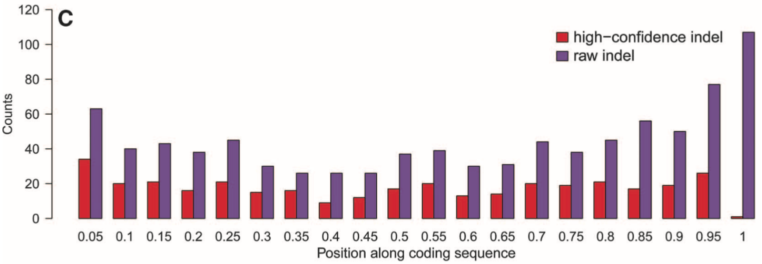
- Stop codon–introducing (nonsense) or splice site–disrupting single-nucleotide variants (SNVs)
- Insertion/deletion (indel) variants predicted to disrupt a transcript's reading frame
- Larger deletions removing either the first exon or more than 50% of the protein-coding sequence of the affected transcript



| | Before filtering | | | | | After filtering | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Variant type | Total | 1000G low-coverage average per individual | | | NA12878 | Total | 1000G low-coverage average per individual | | | NA12878 |
| | | CEU | CHB+JPT | YRI | | | CEU | CHB+JPT | YRI | |
| Stop | 1111 | 85.7 (21.8) | 113.4 (26.7) | 109.1 (23.7) | 115 (25) | 565 | 26.2 (5.2) | 27.4 (6.9) | 37.2 (6.3) | 23 (2) |
| Splice | 658 | 80.5 (29.5) | 98.1 (35.6) | 89.0 (30.4) | 95 (32) | 267 | 11.2 (1.9) | 13.2 (2.5) | 13.7 (1.9) | 12 (1) |
| Frameshift indel | 1040 | 217.8 (112.1) | 225.5 (121.7) | 247.2 (118.7) | 348 (159) | 337 | 38.2 (9.2) | 36.2 (9.0) | 44.0 (8.0) | 38 (11) |
| Large deletion | 142 | 32.4 (12.2) | 31.2 (11.8) | 31.4 (9.7) | 31 (5) | 116 | 28.3 (6.2) | 26.7 (5.9) | 26.6 (5.5) | 24 (4) |
| **Total** | 2951 | 416.4 (175.6) | 468.2 (195.8) | 476.7 (316.0) | 654 (286) | 1285 | 103.9 (22.5) | 103.5 (24.3) | 121.5 (21.7) | 97 (18) |

# Location of LoF SNPs

Both nonsense SNVs and frameshift indels are enriched toward the 3' end of the affected gene, consistent with a greater tolerance to truncation close to the end of the coding sequence



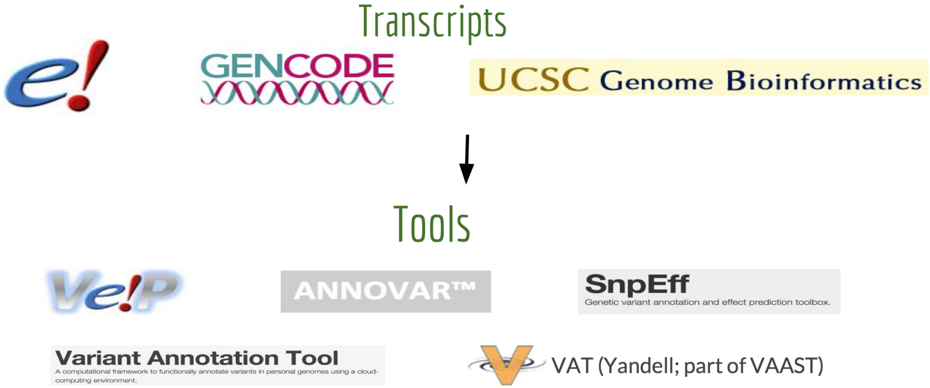
Distribution of frameshift indels along the coding region of affected genes, before and after filtering

http://science.sciencemag.org/content/335/6070/823

# False positives LoF SNPs

- Predicted functional effect of a nonsense or frameshift variant can be altered by other nearby variants on the same chromosome
- Predicted splice-disrupting SNVs and indels can be rescued by nearby alternative splice sites

# Many tools + many transcript annotations = many answers



Transcripts

e!
GENCODE
UCSC Genome Bioinformatics

Tools

Ve!P
ANNOVAR™
SnpEff
Genetic variant annotation and effect prediction toolbox.
Variant Annotation Tool
A computational framework to functionally annotate variants in personal genomes using a cloud-computing environment.
V

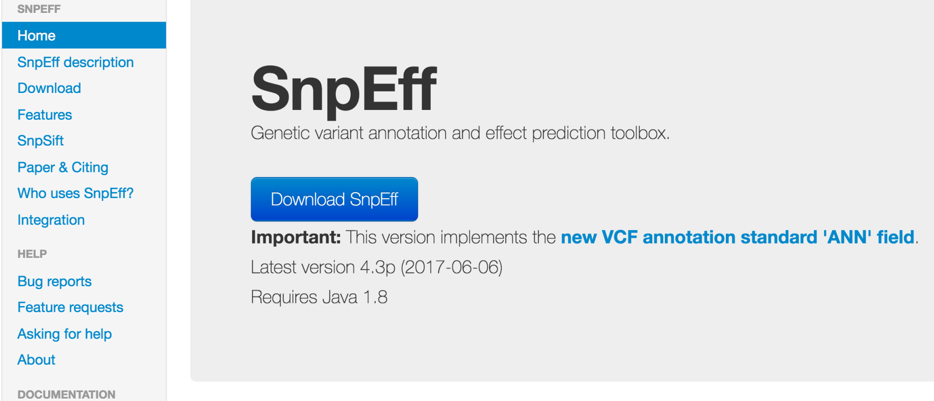
VAT (Yandell; part of VAAST)

# Annotation software

Two sets of software

- Annovar
  - provides a wide range of annotations that can be applied with one tool
- SNPEff and dbNSFP (non-synoymous functional prediction)
- GATK recommends snpEff, but with strict requirements
  - snpEff version 2.0.5 (not 2.0.5d)
  - db should be GRCh37.64 (which is the ensembl database version 64)
  - should use the option -onlyCoding true (using false can cause erroneous annotation)
- GATKs VariantAnnotator to pick the highest impact.
- Finally, also annotate with dbNSFP, which contains:
  - variant frequencies
  - conservation scores
  - protein function effect

# snpEff annotation get placed into INFO field



SNPEFF

- Home
- SnpEff description
- Download
- Features
- SnpSift
- Paper & Citing
- Who uses SnpEff?
- Integration

HELP

- Bug reports
- Feature requests
- Asking for help
- About

DOCUMENTATION

SnpEff

Genetic variant annotation and effect prediction toolbox.

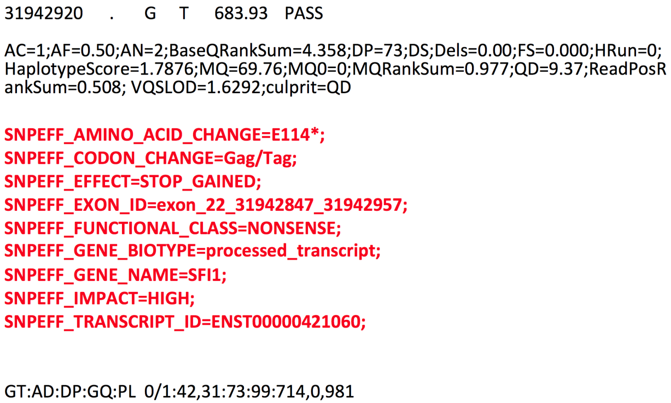
**Important:** This version implements the **new VCF annotation standard 'ANN' field**.

Latest version 4.3p (2017-06-06)

Requires Java 1.8

http://snpeff.sourceforge.net/

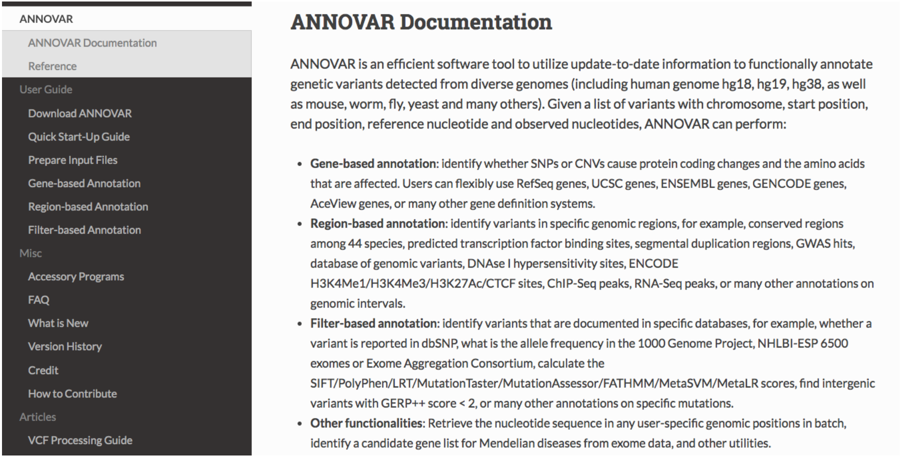
# snpEff annotation get placed into INFO field



```
31942920    .    G    T    683.93    PASS

AC=1;AF=0.50;AN=2;BaseQRankSum=4.358;DP=73;DS;Dels=0.00;FS=0.000;HRun=0;
HaplotypeScore=1.7876;MQ=69.76;MQ0=0;MQRankSum=0.977;QD=9.37;ReadPosR
ankSum=0.508; VQSLOD=1.6292;culprit=QD

SNPEFF_AMINO_ACID_CHANGE=E114*;
SNPEFF_CODON_CHANGE=Gag/Tag;
SNPEFF_EFFECT=STOP_GAINED;
SNPEFF_EXON_ID=exon_22_31942847_31942957;
SNPEFF_FUNCTIONAL_CLASS=NONSENSE;
SNPEFF_GENE_BIOTYPE=processed_transcript;
SNPEFF_GENE_NAME=SFI1;
SNPEFF_IMPACT=HIGH;
SNPEFF_TRANSCRIPT_ID=ENST00000421060;


GT:AD:DP:GQ:PL  0/1:42,31:73:99:714,0,981
```

http://snpeff.sourceforge.net/

# Annovar



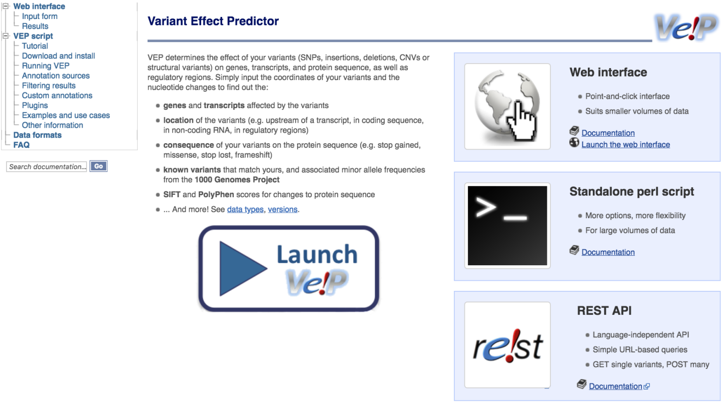
## ANNOVAR Documentation

ANNOVAR is an efficient software tool to utilize update-to-date information to functionally annotate genetic variants detected from diverse genomes (including human genome hg18, hg19, hg38, as well as mouse, worm, fly, yeast and many others). Given a list of variants with chromosome, start position, end position, reference nucleotide and observed nucleotides, ANNOVAR can perform:

- **Gene-based annotation:** identify whether SNPs or CNVs cause protein coding changes and the amino acids that are affected. Users can flexibly use RefSeq genes, UCSC genes, ENSEMBL genes, GENCODE genes, AceView genes, or many other gene definition systems.
- **Region-based annotation:** identify variants in specific genomic regions, for example, conserved regions among 44 species, predicted transcription factor binding sites, segmental duplication regions, GWAS hits, database of genomic variants, DNAse I hypersensitivity sites, ENCODE H3K4Me1/H3K4Me3/H3K27Ac/CTCF sites, ChIP-Seq peaks, RNA-Seq peaks, or many other annotations on genomic intervals.
- **Filter-based annotation:** identify variants that are documented in specific databases, for example, whether a variant is reported in dbSNP, what is the allele frequency in the 1000 Genome Project, NHLBI-ESP 6500 exomes or Exome Aggregation Consortium, calculate the SIFT/PolyPhen/LRT/MutationTaster/MutationAssessor/FATHMM/MetaSVM/MetaLR scores, find intergenic variants with GERP++ score < 2, or many other annotations on specific mutations.
- **Other functionalities:** Retrieve the nucleotide sequence in any user-specific genomic positions in batch, identify a candidate gene list for Mendelian diseases from exome data, and other utilities.

http://annovar.openbioinformatics.org/en/latest/

# VEP - Variant Effect Predictor



- Web interface
  - Input form
  - Results
- VEP script
  - Tutorial
  - Download and install
  - Running VEP
  - Annotation sources
  - Filtering results
  - Custom annotations
  - Plugins
  - Examples and use cases
  - Other information
- Data formats
- FAQ

Search documentation... Go

## Variant Effect Predictor

VEP determines the effect of your variants (SNPs, insertions, deletions, CNVs or structural variants) on genes, transcripts, and protein sequence, as well as regulatory regions. Simply input the coordinates of your variants and the nucleotide changes to find out the:

- **genes** and **transcripts** affected by the variants
- **location** of the variants (e.g. upstream of a transcript, in coding sequence, in non-coding RNA, in regulatory regions)
- **consequence** of your variants on the protein sequence (e.g. stop gained, missense, stop lost, frameshift)
- **known variants** that match yours, and associated minor allele frequencies from the **1000 Genomes Project**
- **SIFT** and **PolyPhen** scores for changes to protein sequence
- ... And more! See data types, versions.

Launch VeP

### Web interface

- Point-and-click interface
- Suits smaller volumes of data

Documentation
Launch the web interface

### Standalone perl script

- More options, more flexibility
- For large volumes of data

Documentation

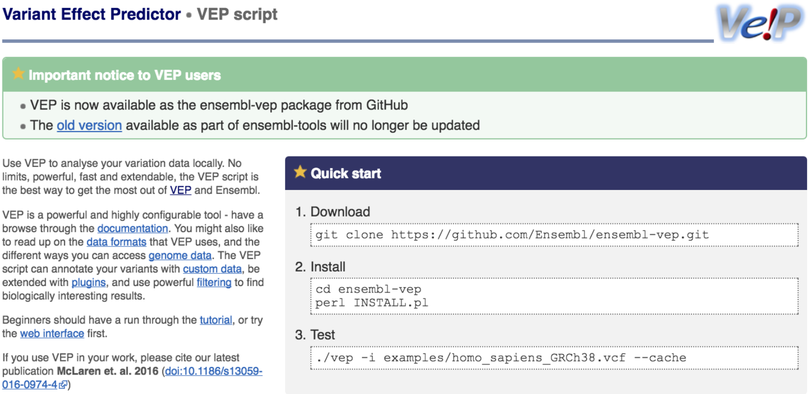
### REST API

- Language-independent API
- Simple URL-based queries
- GET single variants, POST many

Documentation

http://www.ensembl.org/info/docs/tools/vep/index.html

# VEP script



Variant Effect Predictor • VEP script

★ Important notice to VEP users

- VEP is now available as the ensembl-vep package from GitHub
- The old version available as part of ensembl-tools will no longer be updated

Use VEP to analyse your variation data locally. No limits, powerful, fast and extendable, the VEP script is the best way to get the most out of VEP and Ensembl.

VEP is a powerful and highly configurable tool - have a browse through the documentation. You might also like to read up on the data formats that VEP uses, and the different ways you can access genome data. The VEP script can annotate your variants with custom data, be extended with plugins, and use powerful filtering to find biologically interesting results.

Beginners should have a run through the tutorial, or try the web interface first.

If you use VEP in your work, please cite our latest publication **McLaren et. al. 2016** (doi:10.1186/s13059-016-0974-4)

★ Quick start

1. Download

```
git clone https://github.com/Ensembl/ensembl-vep.git
```

2. Install

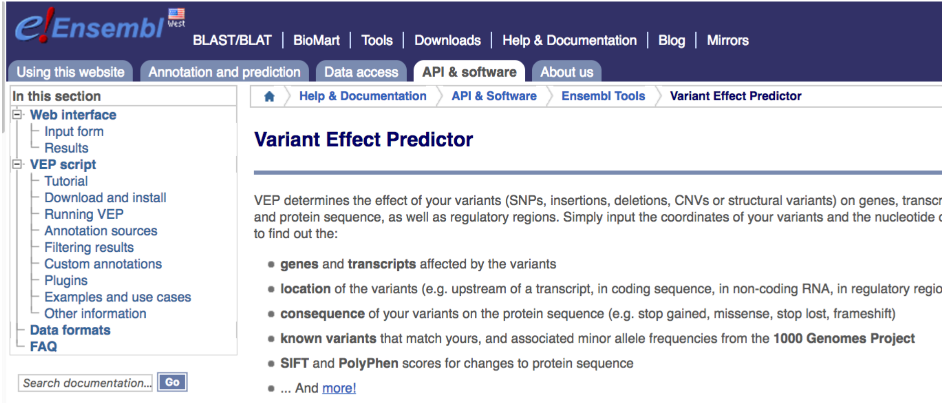
```
cd ensembl-vep
perl INSTALL.pl
```

3. Test

```
./vep -i examples/homo_sapiens_GRCh38.vcf --cache
```

http://uswest.ensembl.org/info/docs/tools/vep/script/index.html

# VEP script



## Variant Effect Predictor

VEP determines the effect of your variants (SNPs, insertions, deletions, CNVs or structural variants) on genes, transcr and protein sequence, as well as regulatory regions. Simply input the coordinates of your variants and the nucleotide c to find out the:

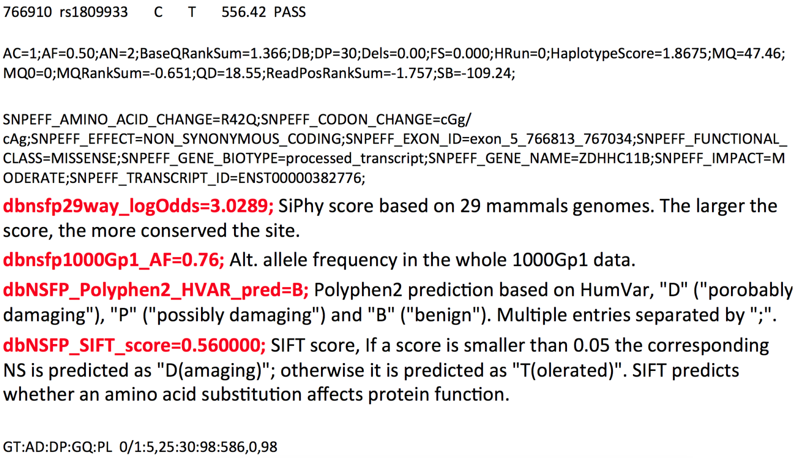
- **genes** and **transcripts** affected by the variants
- **location** of the variants (e.g. upstream of a transcript, in coding sequence, in non-coding RNA, in regulatory regio
- **consequence** of your variants on the protein sequence (e.g. stop gained, missense, stop lost, frameshift)
- **known variants** that match yours, and associated minor allele frequencies from the **1000 Genomes Project**
- **SIFT** and **PolyPhen** scores for changes to protein sequence
- ... And more!

http://uswest.ensembl.org/info/docs/tools/vep/script/index.html

# A second source of functional annotation: dbNSFP

- NSFP = Non-synonymous functional prediction
- Limited to non-synonymous variants
- Has many data fields. We use only:
  - dbnsfpSIFT_score
  - dbnsfpPolyphen2_HVAR_pred
  - dbnsfp29way_logOdds
  - dbnsfp1000Gp1_AF

# Example of annotation with dbNSFP



766910 rs1809933 C T 556.42 PASS

AC=1;AF=0.50;AN=2;BaseQRankSum=1.366;DB;DP=30;Dels=0.00;FS=0.000;HRun=0;HaplotypeScore=1.8675;MQ=47.46;
MQ0=0;MQRankSum=-0.651;QD=18.55;ReadPosRankSum=-1.757;SB=-109.24;

SNPEFF_AMINO_ACID_CHANGE=R42Q;SNPEFF_CODON_CHANGE=cGg/
cAg;SNPEFF_EFFECT=NON_SYNONYMOUS_CODING;SNPEFF_EXON_ID=exon_5_766813_767034;SNPEFF_FUNCTIONAL_
CLASS=MISSENSE;SNPEFF_GENE_BIOTYPE=processed_transcript;SNPEFF_GENE_NAME=ZDHHC11B;SNPEFF_IMPACT=M
ODERATE;SNPEFF_TRANSCRIPT_ID=ENST00000382776;

**dbnsfp29way_logOdds=3.0289;** SiPhy score based on 29 mammals genomes. The larger the score, the more conserved the site.

**dbnsfp1000Gp1_AF=0.76;** Alt. allele frequency in the whole 1000Gp1 data.

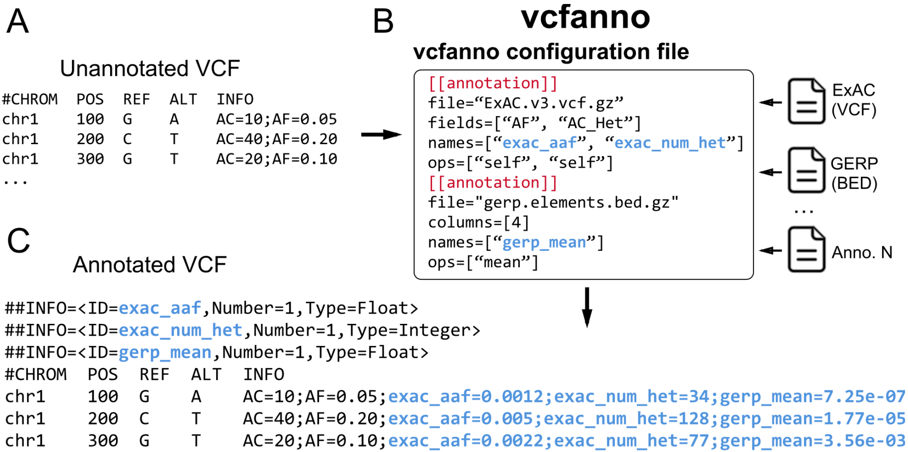
**dbNSFP_Polyphen2_HVAR_pred=B;** Polyphen2 prediction based on HumVar, "D" ("porobably damaging"), "P" ("possibly damaging") and "B" ("benign"). Multiple entries separated by ";".

**dbNSFP_SIFT_score=0.560000;** SIFT score, If a score is smaller than 0.05 the corresponding NS is predicted as "D(amaging)"; otherwise it is predicted as "T(olerated)". SIFT predicts whether an amino acid substitution affects protein function.

GT:AD:DP:GQ:PL 0/1:5,25:30:98:586,0,98

# vcfanno

annotate a VCF with other VCFs/BEDs/tabixed files



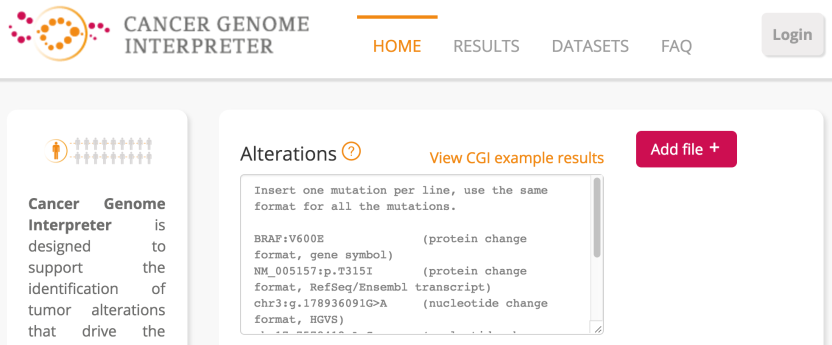
```
#CHROM   POS   REF   ALT   INFO
chr1     100   G     A     AC=10;AF=0.05
chr1     200   C     T     AC=40;AF=0.20
chr1     300   G     T     AC=20;AF=0.10
...
```

```
[[annotation]]
file=“ExAC.v3.vcf.gz”
fields=[“AF”, “AC_Het”]
names=[“exac_aaf”, “exac_num_het”]
ops=[“self”, “self”]
[[annotation]]
file="gerp.elements.bed.gz"
columns=[4]
names=[“gerp_mean”]
ops=[“mean”]
```

```
##INFO=<ID=exac_aaf,Number=1,Type=Float>
##INFO=<ID=exac_num_het,Number=1,Type=Integer>
##INFO=<ID=gerp_mean,Number=1,Type=Float>
#CHROM   POS   REF   ALT   INFO
chr1     100   G     A     AC=10;AF=0.05;exac_aaf=0.0012;exac_num_het=34;gerp_mean=7.25e-07
chr1     200   C     T     AC=40;AF=0.20;exac_aaf=0.005;exac_num_het=128;gerp_mean=1.77e-05
chr1     300   G     T     AC=20;AF=0.10;exac_aaf=0.0022;exac_num_het=77;gerp_mean=3.56e-03
```

# Cancer Genome Interpreter (CGI)

Designed to support the identification of tumor alterations that drive the disease and detect those that may be therapeutically actionable. CGI relies on existing knowledge collected from several resources and on computational methods that annotate the alterations in a tumor according to distinct levels of evidence.

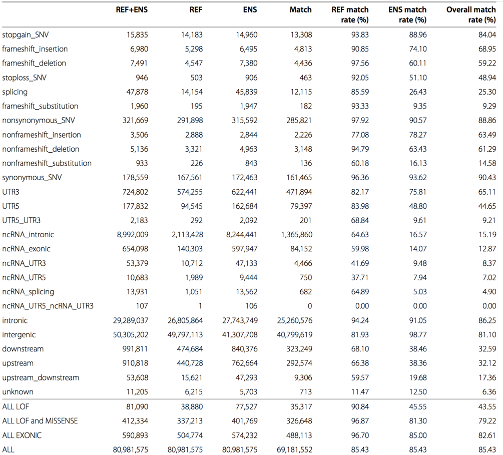


**Cancer Genome Interpreter** is designed to support the identification of tumor alterations that drive the

```
Insert one mutation per line, use the same
format for all the mutations.

BRAF:V600E             (protein change
format, gene symbol)
NM_005157:p.T315I      (protein change
format, RefSeq/Ensembl transcript)
chr3:g.178936091G>A    (nucleotide change
format, HGVS)
```

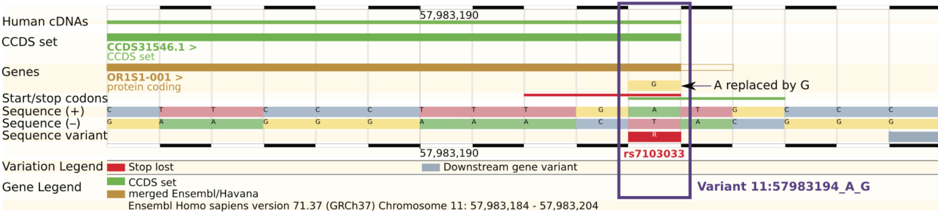
# Annotation problems

Ambiguity - one variant may be annotated differently depending on the choice of transcripts and software



| | REF+ENS | REF | ENS | Match | REF match rate (%) | ENS match rate (%) | Overall match rate (%) |
|---|---|---|---|---|---|---|---|
| stopgain_SNV | 15,835 | 14,183 | 14,960 | 13,308 | 93.83 | 88.96 | 84.04 |
| frameshift_insertion | 6,980 | 5,298 | 6,495 | 4,813 | 90.85 | 74.10 | 68.95 |
| frameshift_deletion | 7,491 | 4,547 | 7,380 | 4,436 | 97.56 | 60.11 | 59.22 |
| stoploss_SNV | 946 | 503 | 906 | 463 | 92.05 | 51.10 | 48.94 |
| splicing | 47,878 | 14,154 | 45,839 | 12,115 | 85.59 | 26.43 | 25.30 |
| frameshift_substitution | 1,960 | 195 | 1,947 | 182 | 93.33 | 9.35 | 9.29 |
| nonsynonymous_SNV | 321,669 | 291,898 | 315,592 | 285,821 | 97.92 | 90.57 | 88.86 |
| nonframeshift_insertion | 3,506 | 2,888 | 2,844 | 2,226 | 77.08 | 78.27 | 63.49 |
| nonframeshift_deletion | 5,136 | 3,321 | 4,963 | 3,148 | 94.79 | 63.43 | 61.29 |
| nonframeshift_substitution | 933 | 226 | 843 | 136 | 60.18 | 16.13 | 14.58 |
| synonymous_SNV | 178,559 | 167,561 | 172,463 | 161,465 | 96.36 | 93.62 | 90.43 |
| UTR3 | 724,802 | 574,255 | 622,441 | 471,894 | 82.17 | 75.81 | 65.11 |
| UTR5 | 177,832 | 94,545 | 162,684 | 79,397 | 83.98 | 48.80 | 44.65 |
| UTR5_UTR3 | 2,183 | 292 | 2,092 | 201 | 68.84 | 9.61 | 9.21 |
| ncRNA_intronic | 8,992,009 | 2,113,428 | 8,244,441 | 1,365,860 | 64.63 | 16.57 | 15.19 |
| ncRNA_exonic | 654,098 | 140,303 | 597,947 | 84,152 | 59.98 | 14.07 | 12.87 |
| ncRNA_UTR3 | 53,379 | 10,712 | 47,133 | 4,466 | 41.69 | 9.48 | 8.37 |
| ncRNA_UTR5 | 10,683 | 1,989 | 9,444 | 750 | 37.71 | 7.94 | 7.02 |
| ncRNA_splicing | 13,931 | 1,051 | 13,562 | 682 | 64.89 | 5.03 | 4.90 |
| ncRNA_UTR5_ncRNA_UTR3 | 107 | 1 | 106 | 0 | 0.00 | 0.00 | 0.00 |
| intronic | 29,289,037 | 26,805,864 | 27,743,749 | 25,260,576 | 94.24 | 91.05 | 86.25 |
| intergenic | 50,305,202 | 49,797,113 | 41,307,708 | 40,799,619 | 81.93 | 98.77 | 81.10 |
| downstream | 991,811 | 474,684 | 840,376 | 323,249 | 68.10 | 38.46 | 32.59 |
| upstream | 910,818 | 440,728 | 762,664 | 292,574 | 66.38 | 38.36 | 32.12 |
| upstream_downstream | 53,608 | 15,621 | 47,293 | 9,306 | 59.57 | 19.68 | 17.36 |
| unknown | 11,205 | 6,215 | 5,703 | 713 | 11.47 | 12.50 | 6.36 |
| ALL LOF | 81,090 | 38,880 | 77,527 | 35,317 | 90.84 | 45.55 | 43.55 |
| ALL LOF and MISSENSE | 412,334 | 337,213 | 401,769 | 326,648 | 96.87 | 81.30 | 79.22 |
| ALL EXONIC | 590,893 | 504,774 | 574,232 | 488,113 | 96.70 | 85.00 | 82.61 |
| ALL | 80,981,575 | 80,981,575 | 80,981,575 | 69,181,552 | 85.43 | 85.43 | 85.43 |

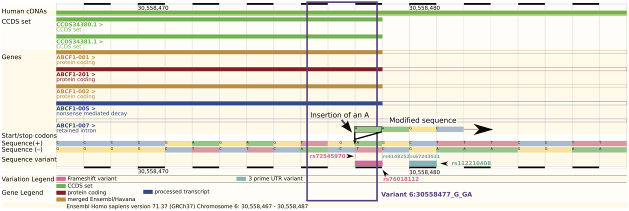
# Straightforward annotation



The variant NC_000011.9:g.57983194A>G (rs7103033) is relatively straightforward to annotate. It is the final base of the final exon in both transcripts at this position (a CCDS transcript (green) and a ‘merged’ ENSEMBL/Havana (GENCODE) transcript (gold)). The final codon has changed from TGA (stop codon) to TGG (tryptophan), so this is unambiguously a stop-loss variant. Using the ENSEMBL transcript set, both ANNOVAR and VEP correctly annotate this variant as stop-loss.

https://genomemedicine.biomedcentral.com/articles/10.1186/gm543

# Ambigious annotation



The variant NC_000006.11:g.30558477_30558478insA (rs72545970) is more difficult to annotate. It is the penultimate base of the exon for all but one of the transcripts shown. It is a single-base insertion, so could be annotated as a frameshift variant. Then again, it is an insertion in a stop codon, so could be a stop-loss variant. In fact, the final codon, TGA (stop codon), remains TGA with this variant (insertion of a single base A), so it is actually a synonymous variant.

https://genomemedicine.biomedcentral.com/articles/10.1186/gm543

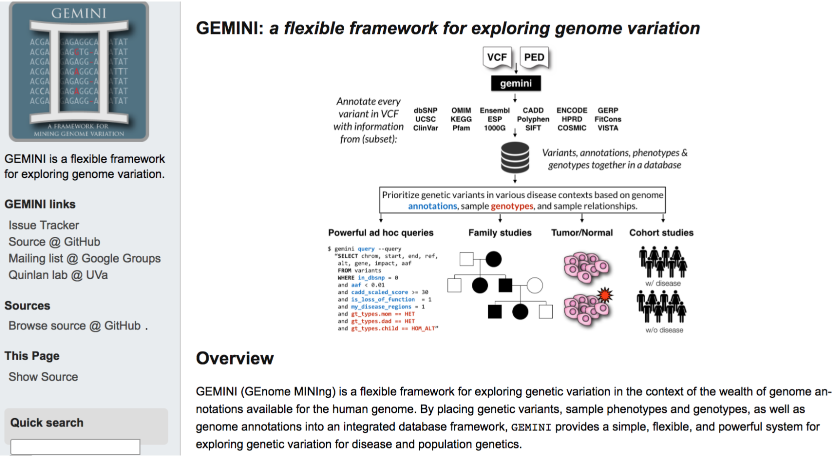
# Allele frequencies differ in different populations

exac.broadinstitute.org gnomad.broadinstitute.org

- Always filter by frequency separately in every available population
  - do not filter for frequency in only one population
  - do not filter on average worldwide frequency
- If variant causes severe phenotype, should *always* be rare in every population

ExAC reports the allele frequency from diverse ancestries

# SNP exploration



GEMINI
A FRAMEWORK FOR MINING GENOME VARIATION

GEMINI is a flexible framework for exploring genome variation.

**GEMINI links**

Issue Tracker
Source @ GitHub
Mailing list @ Google Groups
Quinlan lab @ UVa

**Sources**

Browse source @ GitHub .

**This Page**

Show Source

**Quick search**

**GEMINI: *a flexible framework for exploring genome variation***

VCF
PED
gemini
Annotate every variant in VCF with information from (subset):
dbSNP UCSC ClinVar
OMIM KEGG Pfam
Ensembl ESP 1000G
CADD Polyphen SIFT
ENCODE HPRD COSMIC
GERP FitCons VISTA
Variants, annotations, phenotypes & genotypes together in a database
Prioritize genetic variants in various disease contexts based on genome annotations, sample genotypes, and sample relationships.
Powerful ad hoc queries
Family studies
Tumor/Normal
Cohort studies
w/ disease
w/o disease

```
$ gemini query --query
 “SELECT chrom, start, end, ref,
  alt, gene, impact, aaf
  FROM variants
  WHERE in_dbsnp = 0
  and aaf < 0.01
  and cadd_scaled_score >= 30
  and is_loss_of_function  = 1
  and my_disease_regions = 1
  and gt_types.mom == HET
  and gt_types.dad == HET
  and gt_types.child == HOM_ALT”
```

## Overview

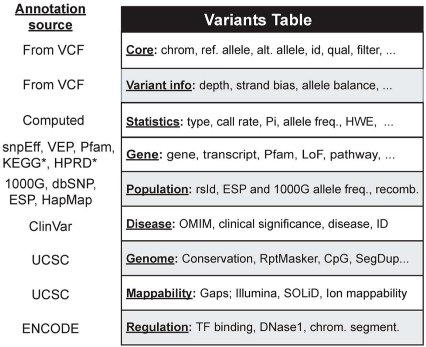
GEMINI (GEnome MINIng) is a flexible framework for exploring genetic variation in the context of the wealth of genome annotations available for the human genome. By placing genetic variants, sample phenotypes and genotypes, as well as genome annotations into an integrated database framework, `GEMINI` provides a simple, flexible, and powerful system for exploring genetic variation for disease and population genetics.

https://gemini.readthedocs.io/en/latest/

https://github.com/arq5x/gemini

# GEMINI annotations

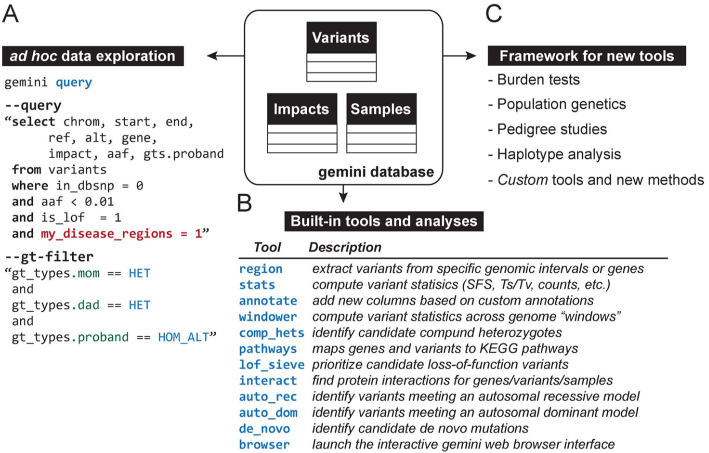
- GEMINI (GEnome MINIng), a flexible software package for exploring all forms of human genetic variation.
- Integrates genetic variation with a diverse and adaptable set of genome annotations (e.g., dbSNP, ENCODE, UCSC, ClinVar, KEGG) into a unified database to facilitate interpretation and data exploration.



| **Annotation source** | **Variants Table** |
|---|---|
| From VCF | **Core**: chrom, ref. allele, alt. allele, id, qual, filter, ... |
| From VCF | **Variant info**: depth, strand bias, allele balance, ... |
| Computed | **Statistics**: type, call rate, Pi, allele freq., HWE, ... |
| snpEff, VEP, Pfam, KEGG*, HPRD* | **Gene**: gene, transcript, Pfam, LoF, pathway, ... |
| 1000G, dbSNP, ESP, HapMap | **Population**: rsId, ESP and 1000G allele freq., recomb. |
| ClinVar | **Disease**: OMIM, clinical significance, disease, ID |
| UCSC | **Genome**: Conservation, RptMasker, CpG, SegDup... |
| UCSC | **Mappability**: Gaps; Illumina, SOLiD, Ion mappability |
| ENCODE | **Regulation**: TF binding, DNase1, chrom. segment. |

# GEMINI variant mining framework

- Structured Query Language (SQL), SQLite database with SNP annotations.



A

**ad hoc data exploration**

```
gemini query
--query
"select chrom, start, end,
      ref, alt, gene,
      impact, aaf, gts.proband
 from variants
 where in_dbsnp = 0
 and aaf < 0.01
 and is_lof  = 1
 and my_disease_regions = 1"
--gt-filter
"gt_types.mom == HET
 and
 gt_types.dad == HET
 and
 gt_types.proband == HOM_ALT"
```

C

**Framework for new tools**

- Burden tests
- Population genetics
- Pedigree studies
- Haplotype analysis
- *Custom* tools and new methods

B

**Built-in tools and analyses**

| *Tool* | *Description* |
|---|---|
| region | *extract variants from specific genomic intervals or genes* |
| stats | *compute variant statisics (SFS, Ts/Tv, counts, etc.)* |
| annotate | *add new columns based on custom annotations* |
| windower | *compute variant statistics across genome "windows"* |
| comp_hets | *identify candidate compund heterozygotes* |
| pathways | *maps genes and variants to KEGG pathways* |
| lof_sieve | *prioritize candidate loss-of-function variants* |
| interact | *find protein interactions for genes/variants/samples* |
| auto_rec | *identify variants meeting an autosomal recessive model* |
| auto_dom | *identify variants meeting an autosomal dominant model* |
| de_novo | *identify candidate de novo mutations* |
| browser | *launch the interactive gemini web browser interface* |

# Gemini howto

- Getting started with GEMINI
- Summary plots from GEMINI
- Incidental findings using GEMINI

https://davetang.org/muse/2016/01/13/getting-started-with-gemini/

https://davetang.org/muse/2017/06/18/summary-plots-gemini/

https://davetang.org/muse/2017/06/21/incidental-findings-using-gemini/

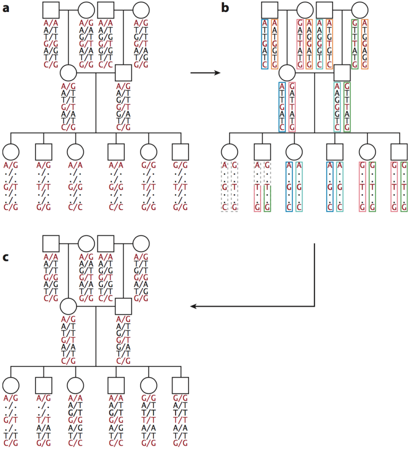
Paila, Umadevi, Brad A. Chapman, Rory Kirchner, and Aaron R. Quinlan. "GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations." PLoS Computational Biology 9, no. 7 (2013): e1003153. doi:10.1371/journal.pcbi.1003153. http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003153

# Genotype imputation

- Generally, a subset of all genetic markers in the genome can be directly genotyped (SNP arrays, exome sequencing)
- Imputation allows evaluating genetic markers that are not directly genotyped for association with a phenotype
- Particularly useful in GWAS meta-analysis

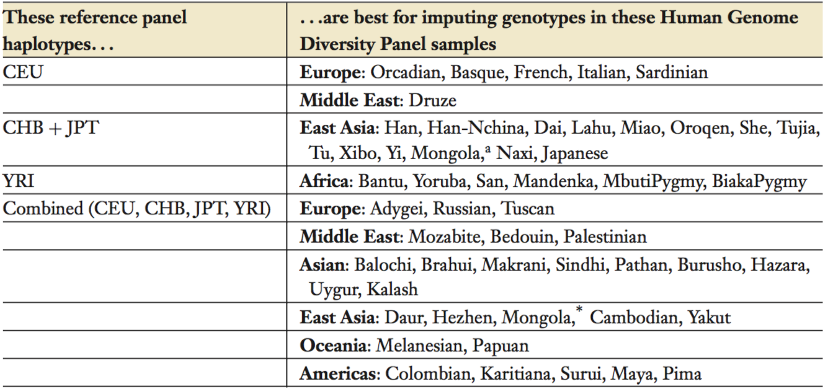
# Genotype imputation in related individuals

Family samples are the most intuitive and simple to genotype - using stretches of shared haplotypes - “identity-by-descent” (IBD) blocks


a
A/A A/T T/T G/G A/G T/T C/G
A/G A/A G/T G/T A/A T/T G/G
A/A A/T G/G G/T G/G T/T C/C
A/G T/T G/T G/T A/A T/G G/G
A/G A/T T/T G/T A/A T/T C/G
A/G A/T G/T G/T G/A T/T C/G
A/G ./. ./. G/T ./. ./. C/G
A/G ./. ./. T/T ./. ./. G/G
A/A ./. ./. G/G ./. ./. C/C
A/A ./. ./. G/G ./. ./. C/C
G/G ./. ./. T/T ./. ./. G/G
G/G ./. ./. T/T ./. ./. G/G
b
ATTGATC AATGGTG
GATTATG AAGGATG
AAGGGTC ATGTGTC
GTTTATG ATGGAGG
ATTGATC GATTATG
AAGGGTC GTTTATG
A··G··C G··T··G
A··T··G G··T··G
A··G··C A··G··C
A··G··C A··G··C
G··T··G G··T··G
G··T··G G··T··G
c
A/A A/T T/T G/G A/G T/T C/G
A/G A/A G/T G/T A/A T/T G/G
A/A A/T G/G G/T G/G T/T C/C
A/G T/T G/T G/G A/A T/G G/G
A/G A/T T/T G/T A/A T/T C/G
A/G A/T G/T G/T G/A T/T C/G
A/G ./. ./. G/T ./. T/T C/G
A/G ./. ./. T/T A/A T/T G/G
A/A A/T G/T G/G A/G T/T C/C
A/A A/T G/T G/G A/G T/T C/C
G/G A/T T/T T/T A/A T/T G/G
G/G A/T T/T T/T A/A T/T G/G

# Genotype imputation in unrelated individuals

Using haploblocks from haplotype reference panels, e.g., HapMap, 1000 genomes



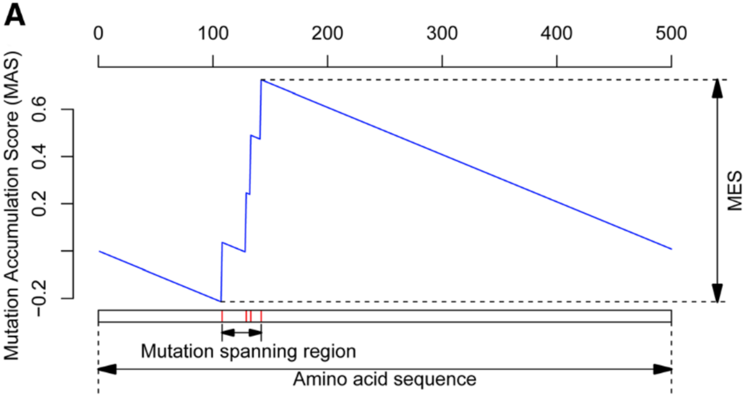
| These reference panel haplotypes... | ...are best for imputing genotypes in these Human Genome Diversity Panel samples |
|---|---|
| CEU | **Europe**: Orcadian, Basque, French, Italian, Sardinian |
| | **Middle East**: Druze |
| CHB + JPT | **East Asia**: Han, Han-Nchina, Dai, Lahu, Miao, Oroqen, She, Tujia, Tu, Xibo, Yi, Mongola,[a] Naxi, Japanese |
| YRI | **Africa**: Bantu, Yoruba, San, Mandenka, MbutiPygmy, BiakaPygmy |
| Combined (CEU, CHB, JPT, YRI) | **Europe**: Adygei, Russian, Tuscan |
| | **Middle East**: Mozabite, Bedouin, Palestinian |
| | **Asian**: Balochi, Brahui, Makrani, Sindhi, Pathan, Burusho, Hazara, Uygur, Kalash |
| | **East Asia**: Daur, Hezhen, Mongola,* Cambodian, Yakut |
| | **Oceania**: Melanesian, Papuan |
| | **Americas**: Colombian, Karitiana, Surui, Maya, Pima |

# Genotype imputation software

- Genotype imputation tools typically fall into two categories:
  - computationally intensive tools such as `IMPUTE`, `MACH` and `fastPHASE/BIMBAM` that take into account all observed genotypes when imputing each missing genotype
  - computationally more efficient tools such as `PLINK`, `TUNA`, `WHAP` and `BEAGLE` that typically focus on genotypes for a small number of nearby markers when imputing each missing genotype

# SNP clustering

- `MSEA-clust` - Kolmogorov-Smirnov adaptation to test whether the distribution of mutations along the genes is significantly different from a random distribution.



# MuSiC

- Mutational Significance in Cancer (MuSiC) Mutation analysis pipeline:
  1. significantly mutated genes,
  2. significantly mutated pathways,
  3. mutation correlation test (pairwise gene test for mutation correlation/exclusion),
  4. clinical correlation test,
  5. proximity analysis (clustering of mutations),
  6. COSMIC/OMIM matching,
  7. Pfam protein domain mutation analysis.

http://gmt.genome.wustl.edu/

https://github.com/ding-lab/MuSiC2

Dees, Nathan D., Qunyuan Zhang, Cyriac Kandoth, Michael C. Wendl, William Schierding, Daniel C. Koboldt, Thomas B. Mooney, et al. "MuSiC: Identifying Mutational Significance in Cancer Genomes." Genome Research 22, no. 8 (August 2012): 1589–98. https://doi.org/10.1101/gr.134635.111.

# OncodriveCLUST

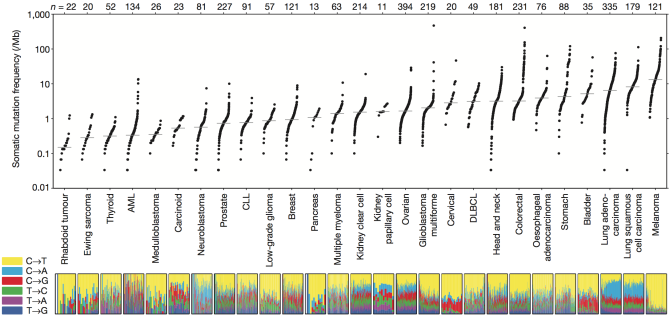
- Gene-centric protein-affecting mutation clustering.
- Significant mutations defined vs. background rate accounting for gene length and the overal number of gene' mutations (binomial test)
- Clusters within 5 amino-acid residues.

http://bg.upf.edu/group/projects/oncodrive-clust.php

Tamborero, David, Abel Gonzalez-Perez, and Nuria Lopez-Bigas. "OncodriveCLUST: Exploiting the Positional Clustering of Somatic Mutations to Identify Cancer Genes." Bioinformatics (Oxford, England) 29, no. 18 (September 15, 2013): 2238–44. https://doi.org/10.1093/bioinformatics/btt395.

# MutSigCV

- Mutational heterogeneity (among patients and cancers) leads to many false positive detection. Need to account for:
  1. regional heterogeneity (among patients, considering mutation spectrum),
  2. gene expression (highly expressed genes mutate more frequently),
  3. replication timing (higher at later replicating regions)



http://archive.broadinstitute.org/cancer/cga/mutsig

# Other tools

- `Platypus` - SNP caller combining haplotype-based, multi-sample, local sequence assembly algorithms in a Bayesian framework

http://www.well.ox.ac.uk/platypus

Rimmer, Andy, Hang Phan, Iain Mathieson, Zamin Iqbal, Stephen R. F. Twigg, WGS500 Consortium, Andrew O. M. Wilkie, Gil McVean, and Gerton Lunter. "Integrating Mapping-, Assembly- and Haplotype-Based Approaches for Calling Variants in Clinical Sequencing Applications." Nature Genetics 46, no. 8 (August 2014): 912–18. https://doi.org/10.1038/ng.3036.