VCF format, ~~Hardy-Weinberg Equilibrium, &~~ VCF toolkits

Applied Computational Genomics

https://github.com/quinlan-lab/applied-computational-genomics

Aaron Quinlan
Departments of Human Genetics and Biomedical Informatics
USTAR Center for Genetic Discovery
University of Utah
quinlanlab.org

Variant Calling Overview

→ FASTQ → Align (BWA) → BAM → Detect SNP/INDELs (GATK or FreeBayes) → VCF

VCF format

BIOINFORMATICS **APPLICATIONS NOTE** *Vol. 27 no. 15 2011, pages 2156–2158*
doi:10.1093/bioinformatics/btr330

Sequence analysis Advance Access publication June 7, 2011

The variant call format and VCFtools

Petr Danecek[1,†], Adam Auton[2,†], Goncalo Abecasis[3], Cornelis A. Albers[1], Eric Banks[4], Mark A. DePristo[4], Robert E. Handsaker[4], Gerton Lunter[2], Gabor T. Marth[5], Stephen T. Sherry[6], Gilean McVean[2,7], Richard Durbin[1,*] and 1000 Genomes Project Analysis Group[‡]

[1]Wellcome Trust Sanger Institute, Wellcome Trust Genome Campus, Cambridge CB10 1SA, [2]Wellcome Trust Centre for Human Genetics, University of Oxford, Oxford OX3 7BN, UK, [3]Center for Statistical Genetics, Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, [4]Program in Medical and Population Genetics, Broad Institute of MIT and Harvard, Cambridge, MA 02141, [5]Department of Biology, Boston College, MA 02467, [6]National Institutes of Health National Center for Biotechnology Information, MD 20894, USA and [7]Department of Statistics, University of Oxford, Oxford OX1 3TG, UK

Associate Editor: John Quackenbush

ABSTRACT

Summary: The variant call format (VCF) is a generic format for storing DNA polymorphism data such as SNPs, insertions, deletions and structural variants, together with rich annotations. VCF is usually stored in a compressed manner and can be indexed for fast data retrieval of variants from a range of positions on the reference genome. The format was developed for the 1000 Genomes Project, and has also been adopted by other projects such as UK10K, dbSNP and the NHLBI Exome Project. VCFtools is a software suite that implements various utilities for processing VCF files, including validation, merging, comparing and also provides a general Perl API.

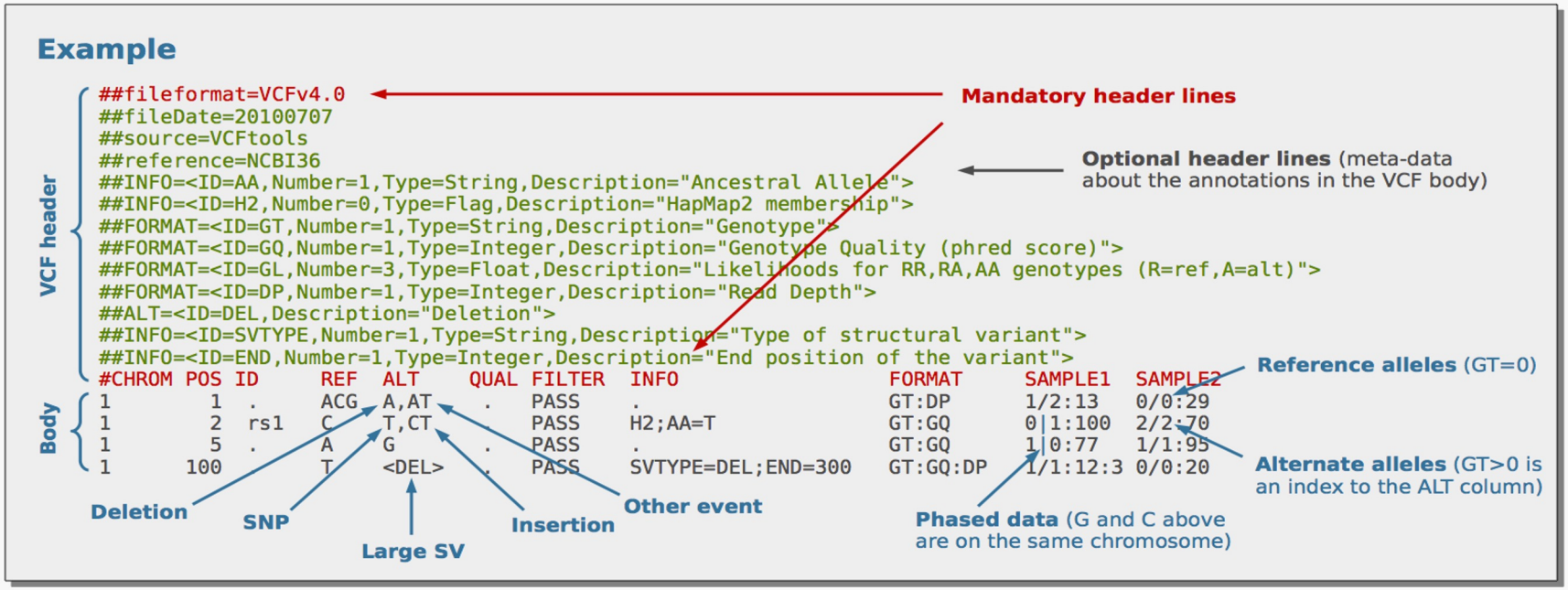
Availability: http://vcftools.sourceforge.net

Contact: rd@sanger.ac.uk

Although generic feature format (GFF) has recently been extended to standardize storage of variant information in genome variant format (GVF) (Reese *et al.*, 2010), this is not tailored for storing information across many samples. We have designed the VCF format to be scalable so as to encompass millions of sites with genotype data and annotations from thousands of samples. We have adopted a textual encoding, with complementary indexing, to allow easy generation of the files while maintaining fast data access. In this article, we present an overview of the VCF and briefly introduce the companion VCFtools software package. A detailed format specification and the complete documentation of VCFtools are available at the VCFtools web site.

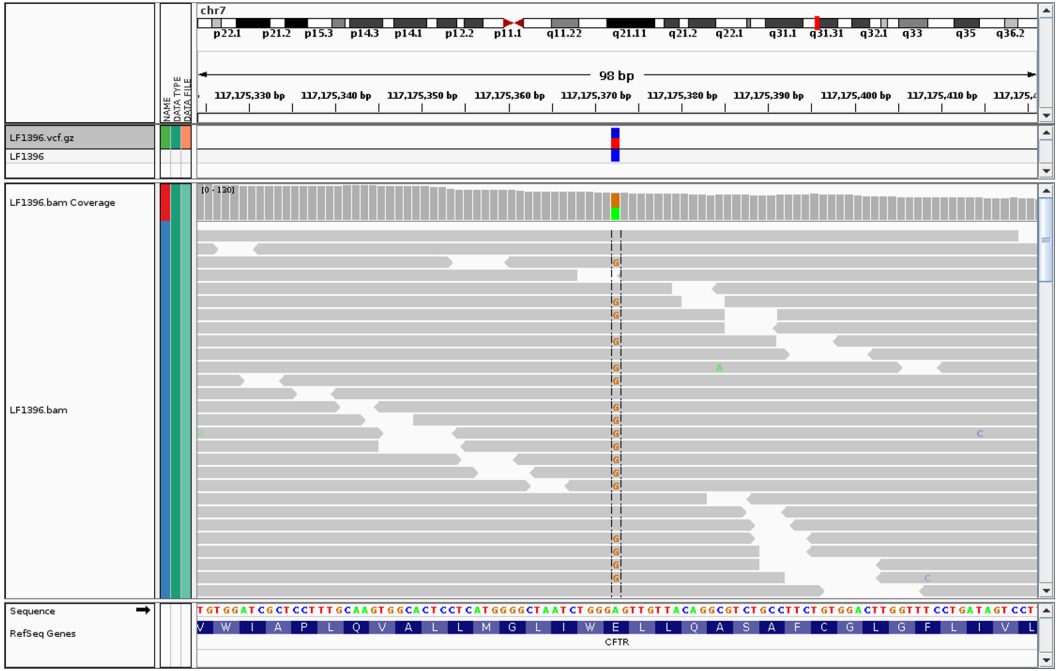
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3137218/pdf/btr330.pdf

VCF format



https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3137218/pdf/btr330.pdf

VCF format. A basic example



Heterozygous A/G. The REF allele is allele "0", ALT is allele "1"

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	LF1396
chr7	117175373	.	A	G	90	PASS			

Genotypes

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	LF1396
chr7	117175373	.	A	G	90	PASS	AF=0.0	GT	0/0
chr7	117175373	.	A	G	90	PASS	AF=0.5	GT	0/1
chr7	117175373	.	A	G	90	PASS	AF=1.0	GT	1/1
chr7	117175373	.	A	G	0	PASS	AF=0.0	GT	./.

Hom. Ref.

Het.

Hom. Alt.

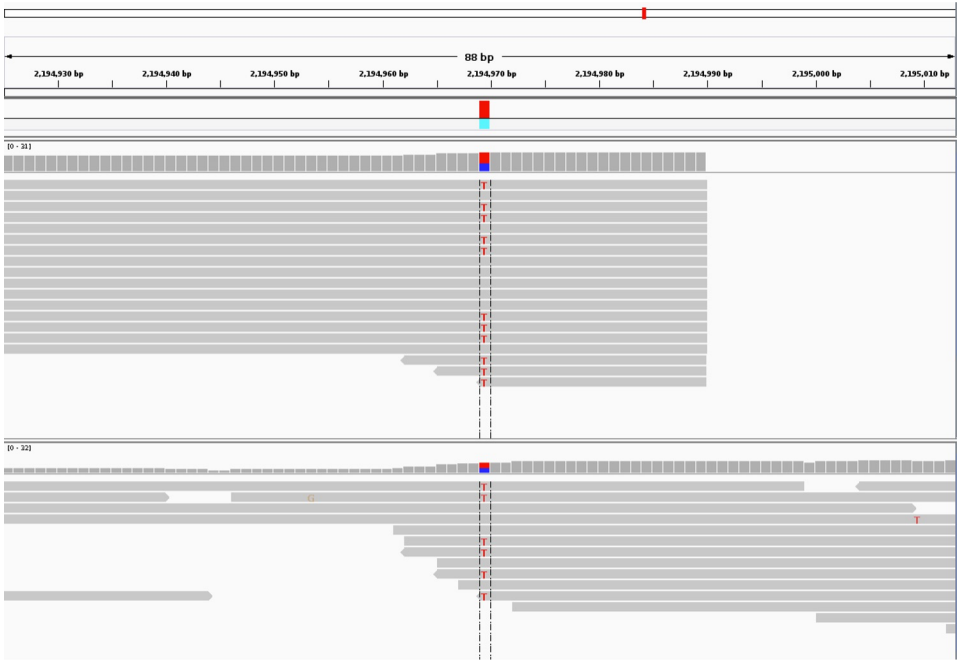
Unknown

Why would a genotype be unknown?

Multi-sample VCF

Mom

Kid



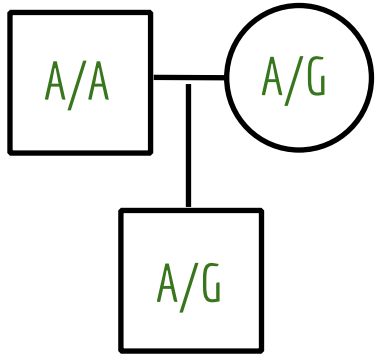
Heterozygous C/T.

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	MOM	KID
chr7	2194169	.	C	T	210	PASS				

VCF format example

```
##fileformat=VCFv4.3
##fileDate=20090805
##source=variantcallerXYZ
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##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">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
```

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	MOM	DAD	KID
20	14370	rs6054257	G	A	29	PASS	NS=3;DP=14;AF=0.5;DB;H2	GT:GQ:DP:HQ	0|0:48:1:51,51	1|0:48:8:51,51	1/1:43:5:.,.
20	17330	.	T	A	3	q10	NS=3;DP=11;AF=0.017	GT:GQ:DP:HQ	0|0:49:3:58,50	0|1:3:5:65,3	0/0:41:3
20	1110696	rs6040355	A	G,T	67	PASS	NS=2;DP=10;AF=0.333,0.667;AA=T;DB	GT:GQ:DP:HQ	1|2:21:6:23,27	2|1:2:0:18,2	2/2:35:4
20	1234567	microsat1	GTC	G,GTCT	50	PASS	NS=3;DP=9;AA=G	GT:GQ:DP	0/1:35:4	0/2:17:2	1/1:40:3



https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3137218/pdf/btr330.pdf

Let's look at a real VCF file from a mom,dad,kid "trio"

```
curl https://s3.amazonaws.com/gemini-tutorials/trio.trim.vep.vcf.gz \
| zcat > trio.vcf
```

"Phased" genotypes

#CHROM	POS		ID	REF	ALT	QUAL	FILTER	INFO
		FORMAT	MOM		KID			
chr7		2194169	.	C	T	210	PASS	
	AF=0.5	GT		0/1		0/1		

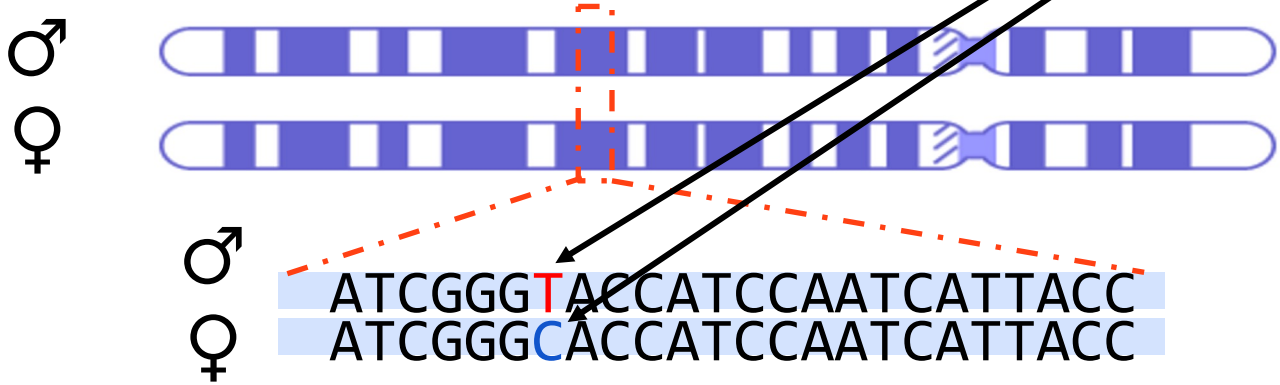
#CHROM	POS		ID	REF	ALT	QUAL	FILTER	INFO
		FORMAT	MOM		KID			
chr7		2194169	.	C	T	210	PASS	
	AF=0.5	GT		0|1		0|1		

Phased genotype

"Phased" genotypes distinguish haplotypes

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	MOM	KID
chr7	2194169	.	C	T	210	PASS	AF=0.5	GT	0|1	0|1

Phased genotype

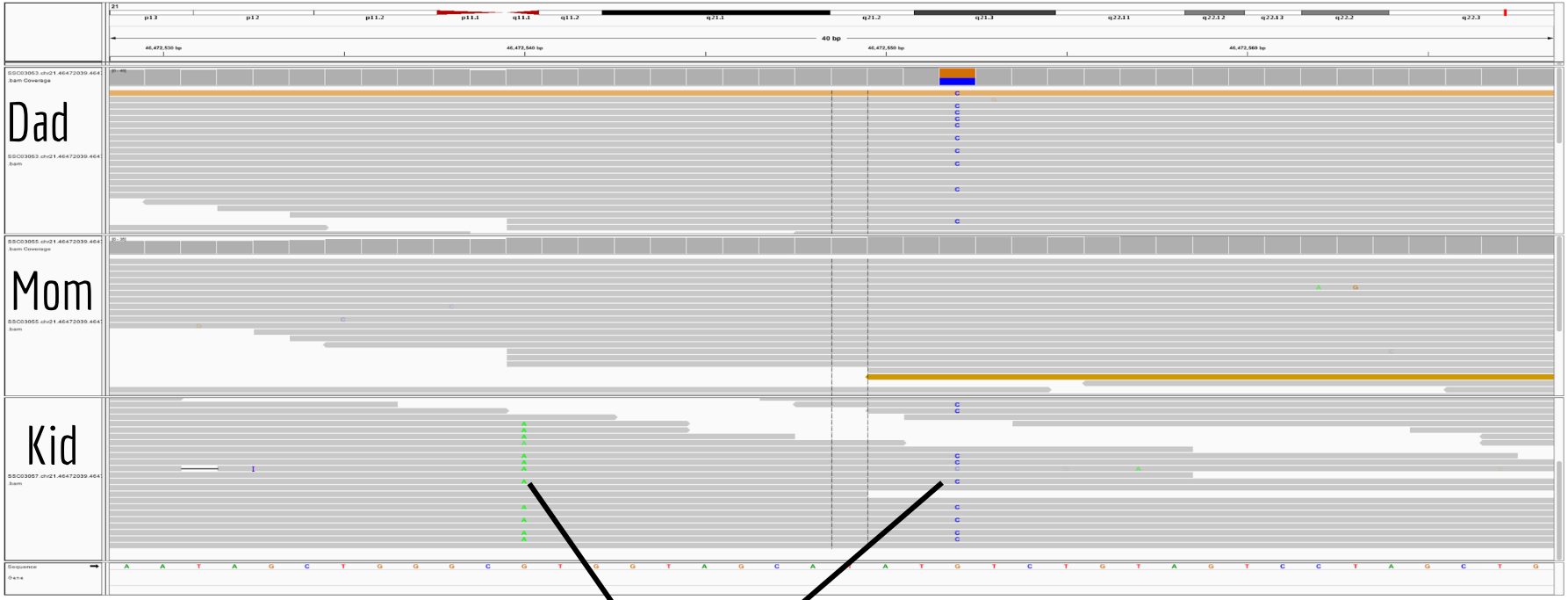


Phasing by inheritance

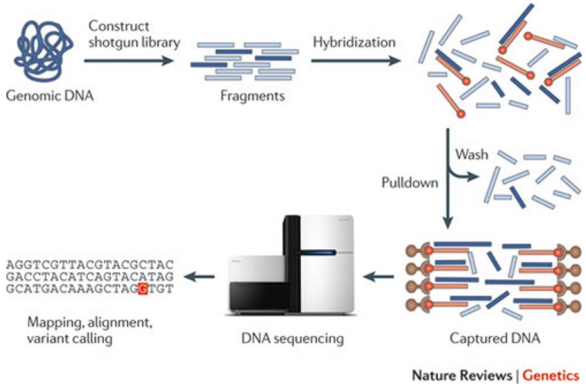
#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO
	FORMAT	MOM	DAD		KID		
chr1		100000	.	A	G	95	PASS	
	AF=0.33	GT		0/1		0/0		1|0
chr1		200000	.	T	C	99	PASS	

This is an example of a compound heterozygote. Hets where the alt allele comes from different parents at two different loci

Read-backed phasing: evidence of haplotype in read or pair



ExAC: exome sequencing of 60,706 humans



Analysis of protein-coding genetic variation in 60,706 humans

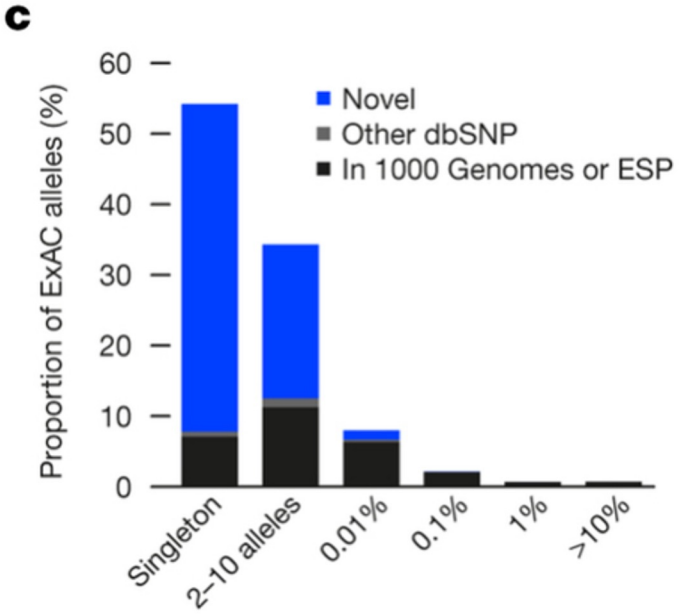
Monkol Lek[1,2,3,4], Konrad J. Karczewski[1,2]*, Eric V. Minikel[1,2,5]*, Kaitlin E. Samocha[1,2,5,6]*, Eric Banks[2], Timothy Fennell[2], Anne H. O'Donnell-Luria[1,2,7], James S. Ware[2,8,9,10,11], Andrew J. Hill[1,2,12], Beryl B. Cummings[1,2,5], Taru Tukiainen[1,2], Daniel P. Birnbaum[2], Jack A. Kosmicki[1,2,6,13], Laramie E. Duncan[1,2,6], Karol Estrada[1,2], Fengmei Zhao[1,2], James Zou[2], Emma Pierce-Hoffman[1,2], Joanne Berghout[14,15], David N. Cooper[16], Nicole Deflaux[17], Mark DePristo[18], Ron Do[19,20,21,22], Jason Flannick[2,23], Menachem Fromer[1,6,19,20,24], Laura Gauthier[18], Jackie Goldstein[1,2,6], Namrata Gupta[2], Daniel Howrigan[1,2,6], Adam Kiezun[18], Mitja I. Kurki[2,25], Ami Levy Moonshine[18], Pradeep Natarajan[2,26,27,28], Lorena Orozco[29], Gina M. Peloso[2,27,28], Ryan Poplin[18], Manuel A. Rivas[2], Valentin Ruano-Rubio[18], Samuel A. Rose[6], Douglas M. Ruderfer[19,20,24], Khalid Shakir[18], Peter D. Stenson[16], Christine Stevens[2], Brett P. Thomas[1,2], Grace Tiao[18], Maria T. Tusie-Luna[30], Ben Weisburd[2], Hong-Hee Won[31], Dongmei Yu[6,25,27,32], David M. Altshuler[2,33], Diego Ardissino[34], Michael Boehnke[35], John Danesh[36], Stacey Donnelly[2], Roberto Elosua[37], Jose C. Florez[2,26,27], Stacey B. Gabriel[2], Gad Getz[18,26,38], Stephen J. Glatt[39,40,41], Christina M. Hultman[42], Sekar Kathiresan[2,26,27,28], Markku Laakso[43], Steven McCarroll[6,8], Mark I. McCarthy[44,45,46], Dermot McGovern[47], Ruth McPherson[48], Benjamin M. Neale[1,2,6], Aarno Palotie[1,2,5,49], Shaun M. Purcell[19,20,24], Danish Saleheen[50,51,52], Jeremiah M. Scharf[2,6,25,27,32], Pamela Sklar[19,20,24,53,54], Patrick F. Sullivan[55,56], Jaakko Tuomilehto[57], Ming T. Tsuang[58], Hugh C. Watkins[44,59], James G. Wilson[60], Mark J. Daly[1,2,6], Daniel G. MacArthur[1,2] & Exome Aggregation Consortium†

Large-scale reference data sets of human genetic variation are critical for the medical and functional interpretation of DNA sequence changes. Here we describe the aggregation and analysis of high-quality exome (protein-coding region) DNA sequence data for 60,706 individuals of diverse ancestries generated as part of the Exome Aggregation Consortium (ExAC). This catalogue of human genetic diversity contains an average of one variant every eight bases of the exome, and provides direct evidence for the presence of widespread mutational recurrence. We have used this catalogue to calculate objective metrics of pathogenicity for sequence variants, and to identify genes subject to strong selection against various classes of mutation; identifying 3,230 genes with near-complete depletion of predicted protein-truncating variants, with 72% of these genes having no currently established human disease phenotype. Finally, we demonstrate that these data can be used for the efficient filtering of candidate disease-causing variants, and for the discovery of human 'knockout' variants in protein-coding genes.

1 variant every 8 bases among 60,706 humans! Also, >50% of the 9 million variants discovered were present as a heterozygote in 1 out of the 60706 individuals (a "singleton")!!!

http://www.nature.com/nature/journal/v536/n7616/pdf/nature19057.pdf

Allele frequency spectrum: most variants are rare



›50% of the 9 million variants discovered were present as a heterozygote in 1 out of the 60706 individuals (a "singleton"). ›90% present on ‹=10 chromosomes sampled

http://www.nature.com/nature/journal/v536/n7616/pdf/nature19057.pdf

Huge excess of rare variation from recent, rapid population growth

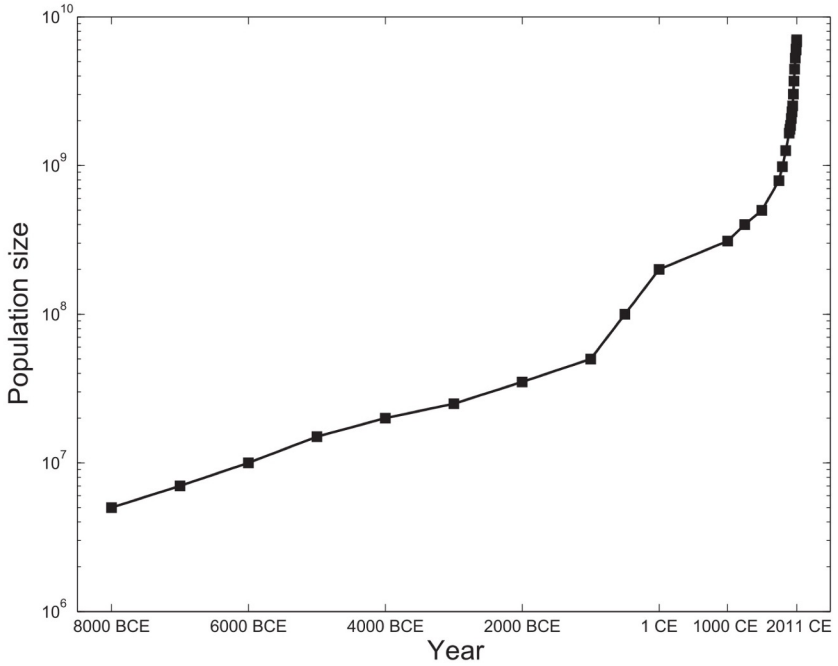
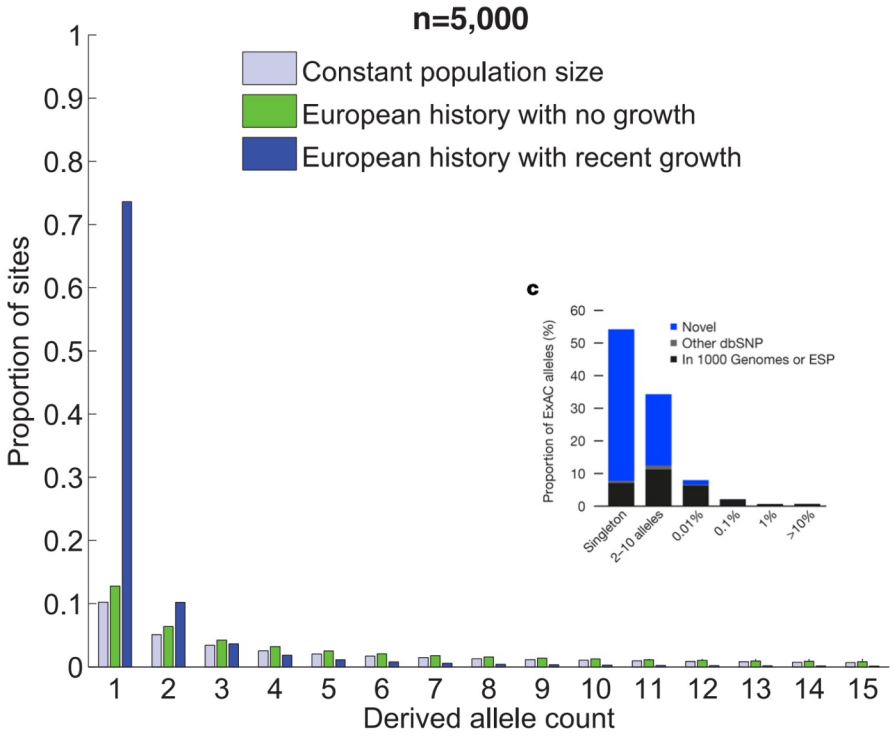
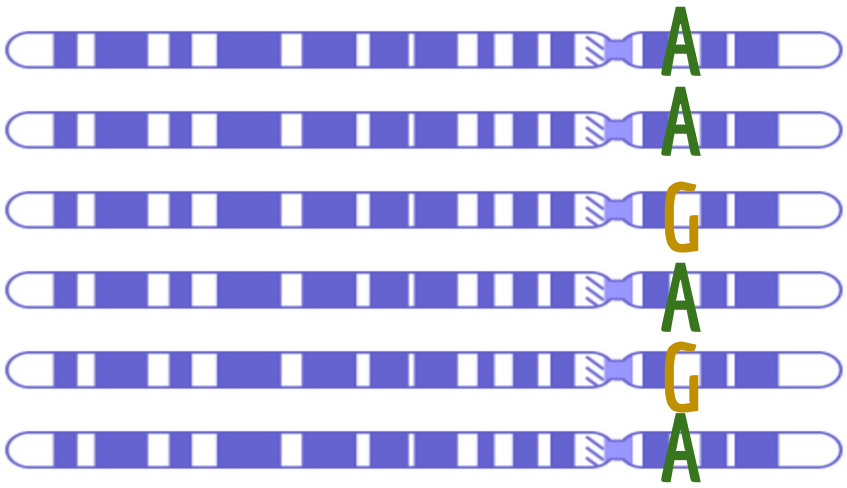


Fig. 1. Census (rather than effective) population size is presented on a logarithm scale over the past 10,000 years, from about 5 million at 8000 BCE to about 7 billion today from data in (*1*, *3*, *30*, *31*). The depicted linear increase (on the log scale) through most of the presented epoch denotes exponential growth of relatively constant percentage increase in population size per year. An acceleration of that increase starting in the Common Era is evident.



Hardy-Weinberg Equilibrium

Polymorphic loci that are biallelic (e.g., A and G alleles) have two allele frequencies, *p* and *q*.



f(A) = p = 4/6 = 0.67

f(G) = q = 2/6 = 0.33

p + q = 1

Hardy-Weinberg Equilibrium

In the absence of evolutionary forces such as selection, drift, or bottlenecks, Hardy-Weinberg equilibrium states that allele and genotype frequencies in a population will remain constant from generation to generation.
If we know the allele frequencies, p and q, we can predict the genotype frequencies that should be observed (binomial expectation).

Observed allele frequencies

$f(A) = p = 4/6 = 0.67$

$f(G) = q = 2/6 = 0.33$

Expected genotype frequencies, given the observed allele frequencies

$f(AA) = p^2 = (0.67)^2 = 0.4489$

$f(AG) = 2pq = 2(0.67)(0.33) = 0.4422$

$f(GG) = q^2 = (0.33)^2 = 0.1089$

$$p^2 + 2pq + q^2 = 1$$

https://en.wikipedia.org/wiki/Hardy%E2%80%93Weinberg_principle#Deviations_from_Hardy.E2.80.93Weinberg_equilibrium

Hardy-Weinberg Equilibrium: expected genotype freqs

$p = 0.5, q = 0.5$

$f(AA) = p^2 = (0.5)^2 = 0.25$
$f(AG) = 2pq = 2(0.5)(0.5) = 0.5$
$f(GG) = q^2 = (0.5)^2 = 0.25$

$p = 0.1, q = 0.9$

$f(AA) = p^2 = (0.1)^2 = 0.01$
$f(AG) = 2pq = 2(0.1)(0.9) = 0.18$
$f(GG) = q^2 = (0.9)^2 = 0.81$

$p = 0.01, q = 0.99$

$f(AA) = p^2 = (0.01)^2 = 0.0001$
$f(AG) = 2pq = 2(0.01)(0.99) = 0.0198$
$f(GG) = q^2 = (0.99)^2 = 0.9801$

$p = 0.001, q = 0.999$

$f(AA) = p^2 = (0.001)^2 = 0.000001$
$f(AG) = 2pq = 2(0.001)(0.999) = 0.001998$
$f(GG) = q^2 = (0.999)^2 = 0.998001$

https://en.wikipedia.org/wiki/Hardy%E2%80%93Weinberg_principle#Deviations_from_Hardy.E2.80.93Weinberg_equilibrium

Hardy-Weinberg Equilibrium

$p = 0.1, q = 0.9$

$f(AA) = p^2 = (0.1)^2 = 0.01$
$f(AG) = 2pq = 2(0.1)(0.9) = 0.18$
$f(GG) = q^2 = (0.9)^2 = 0.81$

If we sequenced 100 individuals, how many A/G heterozygotes would we expect? How many A/A homozygotes?

https://en.wikipedia.org/wiki/Hardy%E2%80%93Weinberg_principle#Deviations_from_Hardy.E2.80.93Weinberg_equilibrium

Hardy-Weinberg Equilibrium: a simulation

Hardy-Weinberg Equilibrium - example

The frequency of allele "Z" at a given locus on chromosome 7 is 0.3. What proportion of individuals do we expect to be heterozygous for the Z and Q alleles?

Deviations from Hardy-Weinberg Equilibrium

- Inbreeding
- Population bottlenecks
- Positive selection
- Purifying selection
- Random genetic drift

Example: a recessive, disease causing allele.

Expect p^2 homozygotes for the recessive allele, yet observe significantly less than p^2 * the number of individuals tested

Hardy-Weinberg Equilibrium - example

In a population that is in Hardy-Weinberg equilibrium, the frequency of the recessive homozygote genotype of a certain trait is 0.16. Calculate the percentage of individuals homozygous for the dominant allele.

http://www.houstonisd.org/cms/lib2/TX01001591/Centricity/Domain/5363/Hardy%20Weinberg%20Problem%20Set%20KEY.pdf

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

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
- **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
- **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

Bcftools examples: http://samtools.github.io/bcftools/howtos/index.html

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 powerful tool for quick querying of 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. The `\n` stands for a newline character, a notation commonly used in the world of computer programming. Any characters without a special meaning will be passed as is, so for example see this command and its output below:

```
$ bcftools query -f 'pos=%POS\n' file.bcf | head -3
pos=13380
pos=16071
pos=16141
```

Bcftools examples: http://samtools.github.io/bcftools/howtos/index.html

Filtering

Most BCFtools commands accept the `-i, --include` and `-e, --exclude` options which allow advanced filtering. In the examples below, we demonstrate the usage on the `query` command because it allows us to show the output in a very compact form using the `-f` formatting option. (For details about the format, see the Extracting information 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="."'` expression to exclude sites where FILTER is not set:

```
$ 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:

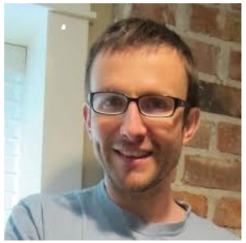
```
$ bcftools query -i'QUAL>20 && DP>10' -f'%CHROM %POS %QUAL %DP\n' file.bcf | head -2
1 14930 31.2757 13
1 17538 37.9458 12

todo:
vcf=/lustre/scratch116/vr/projects/hipsci/cnv/exome-validation/mpileup/ffdm#ffdm_3.bcf
$bt query $vcf -i'QUAL="."' -f' %CHROM %POS %QUAL\n' | head -2
Comparing string to numeric value: QUAL="."
```

FORMAT columns

When filtering FORMAT tags, the OR logic is applied with multiple samples. For example, if we want to remove sites with an uncalled genotype in any sample, the expression `-i 'GT!="."'` is not going to work:

```
$ bcftools query -i'GT!="."' -f'%CHROM %POS [ %GT]\n' file.bcf | head -2
1 30923  ./. 1/1
1 54490  ./. 1/1
```

cyvcf2: manipulate VCF files with Python



Brent Pedersen. Guru.

Genome analysis

cyvcf2: fast, flexible variant analysis with Python

Brent S. Pedersen [1]* and Aaron R. Quinlan [1]*

[1]University of Utah, Department of Human Genetics, Department of Biomedical Informatics, and USTAR Center for Genetic Discovery.

*To whom correspondence should be addressed.

Associate Editor: Dr. John Hancock

Received on XXXXX; revised on XXXXX; accepted on XXXXX

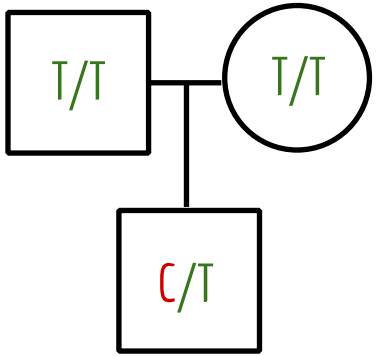
Abstract

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.
Availability: *cyvcf2* is available from https://github.com/brentp/cyvcf2 under the MIT license and from common python package managers.
Contact: bpederse@gmail.com, aaronquinlan@gmail.com
Supplementary information: Detailed documentation is available at http://brentp.github.io/cyvcf2/

Documentation: http://brentp.github.io/cyvcf2/

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

cyvcf2: find de novo mutations in a trio




```
vcf = cyvcf2.VCF(path/to/my/vcf)
PRO, MOM, DAD = range(3)
for v in vcf:
    if v.QUAL < 10:
        continue
    if np.any(v.gt_depths < 10):
        continue
    ref_depths, alt_depths = v.gt_ref_depths, v.gt_alt_depths
    if not all(v.gt_types == [vcf.HET, vcf.HOM_REF, vcf.HOM_REF]):
        continue
    if alt_depths[MOM] > 1 or alt_depths[DAD] > 1:
        continue
    print(v)
```

P(SNP) >= 10

Everyone must have depth >= 10

Parents HOM_REF, kid HET

Neither parent can have any evidence of the ALT allele