Bioinformatics, 2017, 1–2
doi: 10.1093/bioinformatics/btx011
Advance Access Publication Date: 16 January 2017
Applications Note

OXFORD

Genetics and population analysis

VCF-kit: assorted utilities for the variant call format

Daniel E. Cook[1,2] and Erik C. Andersen[2,*]

[1]Interdisciplinary Biological Sciences Program and [2]Department of Molecular Biosciences, Northwestern University, Evanston, IL 60208, USA

*To whom correspondence should be addressed.

Associate Editor: Oliver Stegle

Received on December 7, 2016; revised on January 2, 2017; editorial decision on January 8, 2017; accepted on January 10, 2017

Abstract

Summary: The variant call format (VCF) is a popular standard for storing genetic variation data. As a result, a large collection of tools has been developed that perform diverse analyses using VCF files. However, some tasks common to statistical and population geneticists have not been created yet. To streamline these types of analyses, we created novel tools that analyze or annotate VCF files and organized these tools into a command-line based utility named VCF-kit. VCF-kit adds essential utilities to process and analyze VCF files, including primer generation for variant validation, dendrogram production, genotype imputation from sequence data in linkage studies, and additional tools.
Availability and Implementation: https://github.com/AndersenLab/VCF-kit
Contact: erik.andersen@northwestern.edu

1 Introduction

Population and quantitative genetics investigate how individuals within a population differ. The identification of these differences enables a variety of analyses to be performed. For example, genetic variation can be used to identify the basis of phenotypes, to answer evolutionary questions, or to facilitate forensics. The development of the variant call format (VCF) (Danecek *et al.*, 2011) as a standard for representing genetic variation across a group of individuals or samples has fueled the development of a large number of tools for genetic analyses using variant data. Examples include genotype imputation (Browning and Browning, 2007), variant annotation (Pedersen *et al.*, 2016), variant prediction (Cingolani *et al.*, 2012) and population genetic analysis (Pfeifer *et al.*, 2014). Despite the available tools, we found our studies required a unique set of tools that could interface directly with a VCF file. As a result, we have assembled a collection of tools into a command-line based program written in Python, VCF-kit, which functions directly on VCF files and performs a variety of unique analyses.

2 Implementation and design

VCF-kit is invoked using a command-line interface written in Python and requires a Unix-based operating system. Additionally, VCF-kit requires BWA (Li and Durbin, 2009), BLAST (Altschul *et al.*, 1990), MUSCLE (Edgar, 2004), Samtools (Li *et al.*, 2009), bcftools and PRIMER3 (Untergasser *et al.*, 2012) for certain types of analyses.

3 Usage

3.1 Reference genome management

Several of the utilities included in VCF-kit require a reference genome that has been either indexed by BWA (Li and Durbin, 2009) and Samtools (Li and Durbin, 2009) or used to generate a BLAST database. The `genome` command retrieves and processes reference genomes from the National Center for Biotechnology (NCBI) genomes database (Pruitt and Maglott, 2001). Reference genomes are processed with other available utilities.

3.2 Phylogenetic tree generation

VCF-kit can be used to produce a tree from a VCF using the `phylo` command. Variants are concatenated from each sample and then combined into a single FASTA file, where each line represents one sample. This file effectively represents a multiple sequence alignment that only incorporates variable sites from the VCF samples and can be used to calculate a difference matrix using MUSCLE (Edgar,

2004). MUSCLE outputs the tree in the Newick format, which the user may plot.

3.3 NIL and RIL calling from low coverage sequence data

Recombinant inbred lines (RILs) and near-isogenic lines (NILs) are powerful tools for understanding quantitative traits. RILs are a mixture of genotypes from two or more strains generated by various crossing schemes. NILs are nearly identical to a parental strain except for a single region that has been introgressed from the other parental genome. Both RILs and NILs are used for quantitative genetic mappings. However, genotyping all markers in many RIL or NIL strains can be expensive.

To save on genotyping costs, researchers can barcode and mix samples for pooled sequencing to generate low-coverage genotype data. Low-coverage sequencing, alignment and variant calling produces VCF files with sparsely defined genotypes. If the parental strains are sequenced at higher coverages, they can be used to impute the missing genotypes of RIL and NIL strains using a Hidden-Markov Model (HMM). Because recombination events are rare between tightly linked alleles, parental genotypes can be imputed from linked alleles in RIL and NIL strains. These linked alleles might possess errors because of the nature of low-coverage sequence data. An HMM can be used to identify stretches of genotype calls and infer missing genotypes to classify regions of RIL and NIL genomes inherited from a specific parental strain.

The `hmm` command implements the process described above. A VCF with high-quality genotype data from at least one parent and the genotypes from a population of RIL or NIL samples must be supplied. We manually set the parameters of the HMM based on the level of concordance observed with existing genotype data. Documentation includes examples detailing how to plot genotypes, which is useful to assess imputation results.

3.4 Generating primers for variant validation

The primer command generates primer sequences to validate variants using Sanger sequencing and to genotype restriction fragment length polymorphisms (RFLP) or insertion/deletion (indel) variants. When invoked, sequences flanking the desired variant from the reference genome are retrieved. Generated primers are filtered if they target multiple locations in the reference genome as determined by BLAST (Altschul *et al.*, 1990).

When using the `primer` command for Sanger sequencing validation, a pair of primers are generated for PCR template amplification of the region with the variant. The left primer can also be used to initiate sequencing. For RFLP genotyping, the primer command calculates the expected product sizes given the restriction enzyme and size of PCR amplification product. Users are provided with the product sizes for each restriction fragment, primer sequences, restriction site locations and required restriction enzymes. Finally, primers and product sizes in the presence or absence of an indel variant are output for indel genotyping.

3.5 Call variants from sanger sequences

VCF-kit provides the `call` command for comparing SNVs within a VCF against Sanger sequencing for verifying variants. Users should take care when using the `call` command as it is not a substitute for the manual examination of chromatograms to validate variants. The `call` command takes a FASTA, FASTQ, or AB1 file with Sanger sequences annotated by sample and a VCF file as input. Sequences are compared by BLAST (Altschul *et al.*, 1990) against the specified reference genome and the genotypes corresponding to variant positions within the VCF are output. If the input sequence data is annotated with sample names, output variants can be classified as true positives, true negatives, false positives, or false negatives as compared with Sanger sequencing results.

3.6 Additional tools

The `rename` command can be used to prepend, append, or substitute strings on sample names. The `vcf2tsv` command can convert a SnpEff annotated VCF to a TSV. The `calc` command can be used to count the number of homozygous variants per sample shared with other samples (i.e. the number of singletons, doubletons, tripletons, etc. per sample). VCF-kit documentation features the full list of tools and subcommands.

4 Conclusion

VCF-kit was developed to centralize a collection of tools and scripts we have developed to streamline analyses of genetic variation. VCF-kit is open-source software. We welcome community contributions and feedback. Documentation is available at vcf-kit.readthedocs.io.

Funding

National Institutes of Health [R01GM107227] and American Cancer Society Research Scholar Award to E.C.A.; The National Science Foundation Graduate Research Fellowship [DGE-1324585] to D.E.C.

Conflict of Interest: none declared.

References

Altschul,S.F., *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Browning,S.R. and Browning,B.L. (2007) Rapid and accurate haplotype phasing and missing-data inference for whole-genome association studies by use of localized haplotype clustering. *Am. J. Hum. Genet.*, **81**, 1084–1097.

Cingolani,P., *et al.* (2012) A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of *Drosophila melanogaster* strain W 1118; Iso-2; Iso-3. *Fly*, **6**, 80–92.

Danecek,P., *et al.* (2011) The variant call format and VCFtools. *Bioinformatics*, June. doi:10.1093/bioinformatics/btr330.

Edgar,R.C. (2004) MUSCLE: Multiple Sequence Alignment with High Accuracy and High Throughput. *Nucleic Acids Res.*, **32**, 1792–1797.

Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with Burrows–Wheeler Transform. *Bioinformatics*, **25**, 1754–1760.

Li,H., *et al.* (2009) The sequence alignment/map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

Pedersen,B.S., *et al.* (2016) Vcfanno: Fast, Flexible Annotation of Genetic Variants. *Genome Biol.*, **17**, 118. genomebiology.biomedcentral.com.

Pfeifer,B., *et al.* (2014) PopGenome: an efficient Swiss Army Knife for population genomic analyses in R. *Mol. Biol. Evol.*, **31**, 1929–1936. SMBE.

Pruitt,K.D. and Maglott,D.R. (2001) RefSeq and LocusLink: NCBI Gene-Centered Resources. *Nucleic Acids Res.*, **29**, 137–140.

Untergasser,A., *et al.* (2012) *Primer3—new Capabilities and Interfaces. Nucleic Acids Research*. Oxford University Press: New York, NY, **40**, e115–e115.