ruby-htslib

kojix2 Naohisa Goto

24 May 2022

Summary

Ruby-htslib is the Ruby bindings to HTSlib (Bonfield et al. 2021), a C library for processing high throughput sequencing (HTS) data. It will provide APIs to read and write file formats such as SAM/BAM and VCF/BCF.

- Code of ruby-htslib : https://github.com/kojix2/ruby-htslib

Statement of need

The Ruby language is an object-oriented programming language. It is a general-purpose programming language used primarily in the field of web application development. Ruby has also been used in the bioinformatics field, and the BioRuby project (Goto et al. 2010) provides access to many file formats.

In recent years, the volume of biological data generated by sequencing technologies has increased. file formats such as SAM, BAM, CRAM, VCF, and BCF have become widely used with the spread of next-generation sequencers. SAM, BAM, and CRAM are file formats for alignments. VCF and BCF are file formats for variants. The specifications for these file formats are defined in hts-specs. Samtools and Bcftools (Danecek et al. 2021) were created to manipulate HTS files. And the core part of samtools became a library called HTSlib. We can use HTSLib (Bonfield et al. 2021) to read, write and query HTS files.

However, the ways to manipulate HTS files from the Ruby language have been limited. BioRuby does not have a module to work with HTS files. Bio-Samtools (Etherington, Ramirez-Gonzalez, and MacLean 2015) was originally developed as a samtools binding. However, when samtools separated htslib, the bindings stopped working. Now it uses open3 to call samtools directly from standard streams.

Ruby-htslib is a binding for htslib. It provides access to comprehensive HTS files from the Ruby language. This allows the Ruby language to analyze genomes and create applications.

Implementation

ruby-htslib was implemented using Ruby-FFI. Reduced memory usage by the method used in the hts-nim (Pedersen and Quinlan 2018).

Benchmark

Examples

Reading bam file.

```ruby
require 'htslib'

bam = HTS::Bam.open("test/fixtures/moo.bam")
```

```ruby
bam.each do |r|
  pp name: r.qname,
     flag: r.flag,
     chrm: r.chrom,
     strt: r.pos + 1,
     mapq: r.mapq,
     cigr: r.cigar.to_s,
     mchr: r.mate_chrom,
     mpos: r.mpos + 1,
     isiz: r.isize,
     seqs: r.seq,
     qual: r.qual_string,
     MC:   r.aux("MC")
end

bam.close
```

Reading Bcf file.

```ruby
bcf = HTS::Bcf.open("b.bcf")

bcf.each do |r|
  p chrom:  r.chrom,
    pos:    r.pos,
    id:     r.id,
    qual:   r.qual.round(2),
    ref:    r.ref,
    alt:    r.alt,
    filter: r.filter,
    info:   r.info.to_h,
    format: r.format.to_h
end

bcf.close
```

htsgrid

We present a very simple genome browser as an example of the use of Ruby-htslib.

bam-filter and bcf-filter

Using eval allows for very flexible sorting.

Reference

Bonfield, James K., John Marshall, Petr Danecek, Heng Li, Valeriu Ohan, Andrew Whitwham, Thomas Keane, and Robert M. Davies. 2021. "HTSlib: C Library for Reading/Writing High-Throughput Sequencing Data." *GigaScience* 10 (2): giab007. https://doi.org/10.1093/gigascience/giab007.

Danecek, Petr, James K. Bonfield, Jennifer Liddle, John Marshall, Valeriu Ohan, Martin O. Pollard, Andrew Whitwham, et al. 2021. "Twelve Years of SAMtools and BCFtools." *GigaScience* 10 (2): giab008. https://doi.org/10.1093/gigascience/giab008.

Etherington, Graham J., Ricardo H. Ramirez-Gonzalez, and Dan MacLean. 2015. "Bio-Samtools 2: A Package for Analysis and Visualization of Sequence and Alignment Data with SAMtools in Ruby: Fig. 1."

Bioinformatics 31 (15): 2565–67. https://doi.org/10.1093/bioinformatics/btv178.

Goto, Naohisa, Pjotr Prins, Mitsuteru Nakao, Raoul Bonnal, Jan Aerts, and Toshiaki Katayama. 2010. "BioRuby: Bioinformatics Software for the Ruby Programming Language." *Bioinformatics* 26 (20): 2617–19. https://doi.org/10.1093/bioinformatics/btq475.

Pedersen, Brent S, and Aaron R Quinlan. 2018. "Hts-Nim: Scripting High-Performance Genomic Analyses." Edited by Inanc Birol. *Bioinformatics* 34 (19): 3387–89. https://doi.org/10.1093/bioinformatics/bty358.