

Filtering read alignments in BAM format

Tonatiuh Peña-Centeno
University of Greifswald

December 18, 2011

Abstract

This is a description of the implementation of filterBam, a software that filters out SAM alignment reads by the following criteria: coverage, percentage identity and, if requested, insert length.

1 NOTES:

This document makes reference to the SAM/BAM format specification of Li et al. [2009].

2 Bamtools

Bamtools is a C++ wrapper API of the more well-known Samtools software. The latest version of Bamtools is 2.0 and is available on the website

<https://github.com/pezmaster31/bamtools/downloads>

3 Compilation

Make sure to link with the “-lz” and “-libbamtools.a” flags on; where -lz refers to the ZLIB library, and libbamtools.a to the static bamtools library included in the software distribution. An example of how to compile and link follows:

```
g++ -I$BAMTOOLS/include -g -std=c++0x -c filterBam.cc -o filterBam.o
g++ -g -std=c++0x filterBam.o -o filterBam $BAMTOOLS/lib/libbamtools.a -lz
```

where **\$BAMTOOLS** is the path where Bamtools was installed.

Note that the flag “-std=c++0x” has been used given that some of the functionalities of the filter require some of the newest features of GNU’s g++ compiler. This and future versions of the software have been tested on Ubuntu’s g++ version 4.4.3.

4 How to run

A run that will let pass most, if not all, readings:

```
./filterBam input.bam output.bam -minCover 0 -minId 0 -insertLimit 10000000
-nointrons
```

Note: that all options are provided at the very end.

5 Coverage, percent of identity and insert length

The coverage is computed as the sum of the alignment matches (sequence matches or mismatches) and the insertions to the reference. Both figures, alignment matches and insertions to the reference, correspond to CIGAR string operations M and I , respectively. Thus the following is done

$$\text{coverage} = \frac{\sum \text{CIGAR}(M, I)}{qLength} \quad (1)$$

An approximation to the percentage of identity is given by computing the query length and subtracting the so-called edit distance to the reference (tag “NM” in SAM jargon), i.e.

$$\text{percId} = \frac{qLength - \text{Tag}(NM)}{qLength} \quad (2)$$

The length of inserts is estimated by summing CIGAR operations “M” and “I”, which correspond to alignment matches and deletions from the reference. In other words, we do the following

$$\text{InsertSize} = \frac{\sum \text{CIGAR}(D, I)}{qLength} \quad (3)$$

References

- H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Math, G. Abecasis, R. Durbin, and . G. P. D. P. Subgroup. The sequence alignment/map format and samtools. *Bioinformatics Applications Note*, 25(16):2078–2079, 2009.