Bis-SNP User Guide v001

For Bis-SNP v0.71

Yaping Liu, Benjamin P. Berman

19th June 2013

Contents

1 Introduction **2**

2 Prerequisites **2**

3 Quick Start **3**

3.1 Download Bis-SNP program. 3

3.2 Download input files for test 3

3.3 Run Bis-SNP program in terminal. 3

4 Step by step genotyping tutorial **4**

4.1 Add read group tag to BAM file 4

4.2 Indel realignment 4

4.2.1 Find indel region 5

4.2.2 Realign in the indel region 5

4.3 Mark duplicated reads 5

4.4 Base quality recalibration 5

4.4.1 Count Covariant 6

4.4.2 Write recalibrated base quality score into BAM file 6

4.4.3 Re-Count Covariant 6

4.4.4 Generate recalibration plot 6

4.5 Bis-SNP genotyping 7

4.6 Filter fake SNPs 8

4.7 Generate bed file or wig file for SNP/DNA methylation visualization 8

4.7.1 Converted to .wig format 8

4.7.2 Converted to .bed format 8

4.7.3 Converted to .bedgraph format 9

4.7.4 Extract cytosine coverage information to .bedgraph format 9

5 Usage Detail **9**

5.1 BisulfiteGenotyper . . . 9

5.1.1 Analysis mode options . . . 9

5.1.2 Input options . . . 10

5.1.3 Output options . . . 10

5.1.4 Threshold options . . . 11

5.1.5 Advanced options . . . 11

5.1.6 Other options for debug or still in test . . . 12

5.2 BisulfiteRealignerTargetCreator . . . 13

5.3 BisulfiteIndelRealigner . . . 13

5.4 BisulfiteCountCovariates . . . 14

5.5 BisulfiteTableRecalibration . . . 14

5.6 VCFpostprocess . . . 15

6 Interpret output files **16**

6.1 VCF file . . . 16

6.1.1 INFO column . . . 16

6.1.2 FORMAT column . . . 16

6.2 CpG reads file . . . 17

7 Additional useful script **17**

8 Bis-SNP API structure **17**

9 Build on source code **18**

10 Contact for help **18**

1 Introduction

Bis-SNP is the public available free software (GPL v3 license) for genotyping in bisulfite treated massively parallel sequencing (whole genome Bisulfite-seq(BS-seq), NOMe-seq and RRBS) on Illumina platform. It works for both of single-end and paired-end reads in Illumina directional Bisulfite-Seq library. It is implemented in Java and based on GATK map-reduce framework for the parallel computation. Copyright belongs to USC Epigenome Center.

2 Prerequisites

1. System: Linux or Mac OSX (For small BAM file with light coverage, 4G memory is the minimum requirement. For large BAM files, 10G or more memory are recommended).

2. Java: Java(TM) SE Runtime Environment 1.6 (Linux, Mac OSX). Since the whole package is built on top of GATK, which does not support OpenJDK and Java 1.7 right now(`http://www.broadinstitute.org/gsa/wiki/index.php/Prerequisites`, `http://www.broadinstitute.org/gsa/wiki/index.php/Frequently_Asked_Questions#What_versions_of_Java_does_the_GATK_support.3F`), Bis-SNP can only work under Sun JDK 1.6 currently.

3. Perl: Bis-SNP perl scripts require Perl v 5.8.8 or later.

3 Quick Start

3.1 Download Bis-SNP program.

Download jar file from `https://sourceforge.net/projects/bissnp/files/`

3.2 Download input files for test

All of the input example file could be downloaded from our website: `http://epigenome.usc.edu/publicationdata/bissnp2011/easyUsage.html` and `http://epigenome.usc.edu/publicationdata/bissnp2011/utilies.html`. All *.bz2 files need to be unzipped firstly.

1. Reference genome file: hg18_unmasked.plusContam.fa.bz2 for hg18, hg19_rCRSchrm.fa.bz2 for hg19.

2. dbSNP file: dbsnp_135.hg18.sort.vcf.bz2 for hg18, dbsnp_135.hg19.sort.vcf.bz2 for hg19.

3. Interval file: I provide whole_genome_interval_list.hg18.bed for the whole genome genotype calling in hg18 and whole_genome_interval_list.hg19.bed for the whole genome genotype calling in hg19. You can also specify the interval by such a command: `'-L chr11:7000000-7100000'`

4. BAM file: normalMerge_chr11-7M-9M.nodups.withhead.bam is BAM file for test. You also need to download normalMerge_chr11-7M-9M.nodups.withhead.bam.bai for BAM file's index or made it from SAMTOOLS by yourself. Since newest GATK no longer support BAM file without ReadGroup tag, Bis-SNP also require that input BAM file owns ReadGroup tag. If your own BAM file do not have ReadGroup tag, you should use `AddOrReplaceReadGroups.jar` in Picard.

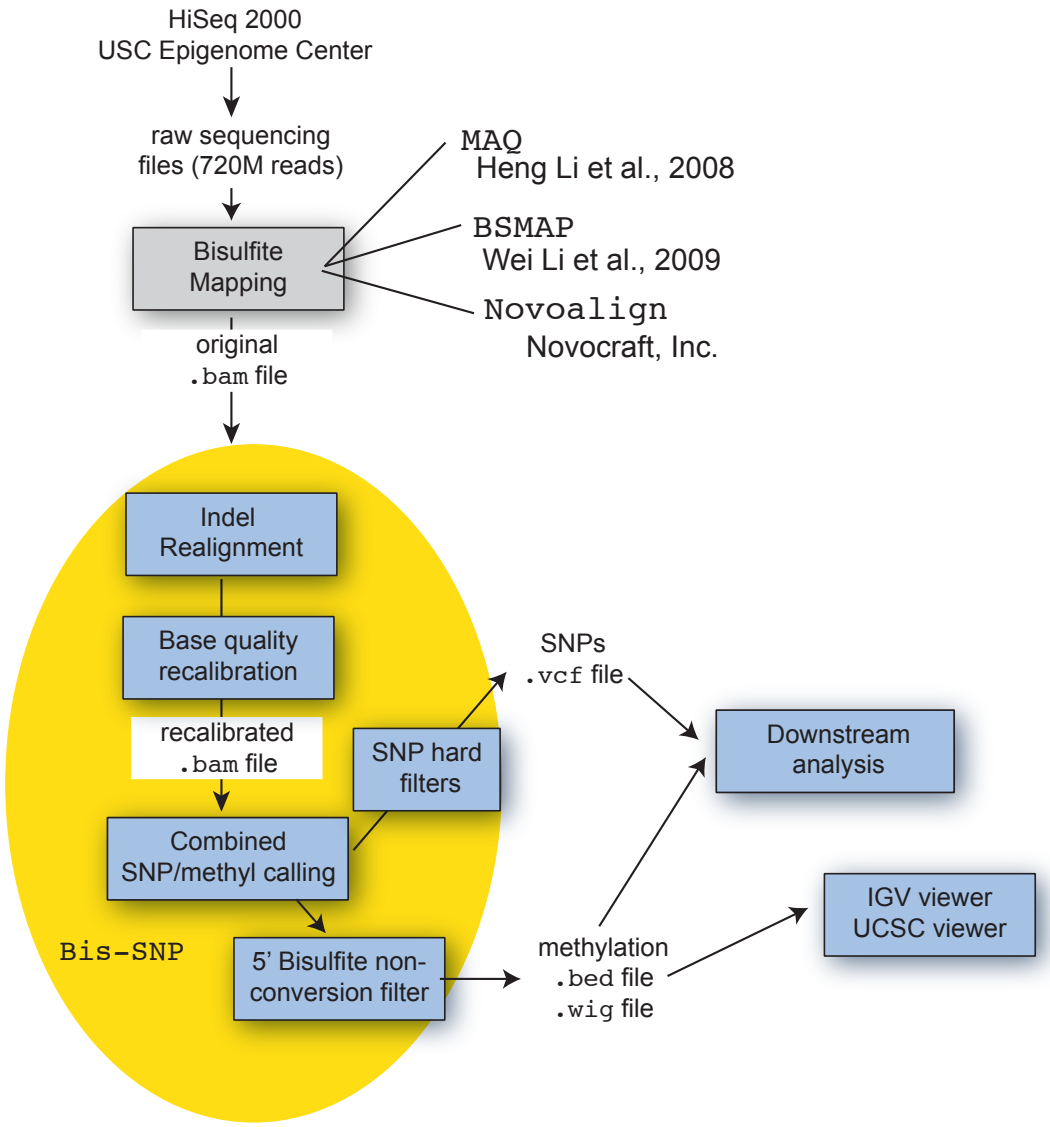
3.3 Run Bis-SNP program in terminal.

1. The following command would output all of the CpG loci in cpg.raw.vcf, all of the SNP loci in snp.raw.vcf, and generate cytosine methylation summary file (.txt):

```
java -Xmx4g -jar BisSNP-0.71.jar -R hg18_unmasked.plusContam.fa -T BisulfiteGenotyper
-I normalMerge_chr11-7M-9M.nodups.withhead.bam -D dbsnp_135.hg18.sort.vcf
-vfn1 cpg.raw.vcf -vfn2 snp.raw.vcf -L chr11:7000000-7100000
```

4 Step by step genotyping tutorial

Here is the basic pipeline by using Bis-SNP to do Bisulfite-seq genotyping and methylation calling. `-Xmx4g` is the option to specify maximum memory for JVM to use, when you have large BAM files, it will be better to specify larger memory to use. e.g. `'-Xmx10g'`



4.1 Add read group tag to BAM file

GATK engine requires BAM file to have ReadGroup tag. When your own BAM file does not contained Read group tag. Download Picard tools `http://sourceforge.net/projects/picard/files/`, then using the following command to add Read group tag to BAM file.

```
java -Xmx4g -jar AddOrReplaceReadGroups.jar I=sample.withoutRG.bam O=sample.withRG.bam ID=readGroup_name
LB=readGroup_name PL=illumina PU=run SM=sample_name CREATE_INDEX=true VALIDATION_STRINGENCY=SILENT
SORT_ORDER=coordinate
```

4.2 Indel realignment

Most of Bisulfite-seq mapping tools do not allow gap alignment yet, they could not find out indels which would cause some of false discovered SNPs. If your mapping tools allows gapped alignment, like Bismark with Bowtie2 or Novoaligner, you could omit this step. We enable Indel alignment on Bisulfite-seq by two steps. Notes: Realign in the indel region step is very slow for BAM files with very high coverages, e.g. 50X or more.

4.2.1 Find indel region

To reduce the amount of time, Bis-SNP only look at those region with known indels (provided knownIndels.vcf and indels region in your BAM files). Download known indel files from our website (transformed from dbSNP and mouse genome project) or use your own known indel vcf files (VCF format 4.0 or later). Bis-SNP would look at those regions with high mismatches and create possible indel region interval file. Replace `'-R referenceGenome.fa'` with your own reference genome file, `'-I sample.withRG.bam'` with your own input bam files. `'-T BisulfiteRealignerTargetCreator'` is to specify the type of analysis. `'-L region.bed'` specifies the region you are interested in. `'-known indel_1.vcf'` gives the known interval region. `'-o indel_target_interval.intervals'` defines the output interval file name. `'indel_target_interval.intervals'` should have ".interval" at the end, otherwise, GATK framework would not recognize it as a correct format in the next BisulfiteIndelRealigner step. `'-nt cpu_cores_number'` specifies the number of CPU threads to use.

```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -I sample.withRG.bam -T BisulfiteRealignerTargetCreator
-L region.bed -known indel_1.vcf -known indel_2.vcf -o indel_target_interval.intervals -nt cpu_cores_number
```

4.2.2 Realign in the indel region

Use the `'indel_target_interval.bed'` file generated above to do indel realignment with `'-targetIntervals indel_target_interval.intervals'`. `'-T BisulfiteIndelRealigner '` specifies the type of analysis. `'-cigar'` needs to be specified when your mapping tools do not allow gapped alignment and all CIGAR strings in BAM file are `'M'`. Original CIGAR string will be kept under `'OC'` flag in BAM file.

```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -I sample.withRG.bam -T BisulfiteIndelRealigner
-targetIntervals indel_target_interval.intervals -known indel_1.vcf -known indel_2.vcf -cigar
-o sample.withRG.realigned.bam
```

4.3 Mark duplicated reads

Use Picard tools to mark the duplicated reads which are mostly come from PCR duplication. This step could be done before indel realignment if there are too many duplicated reads which would mislead indel alignment:

```
java -Xmx10g -jar MarkDuplicates.jar I=sample.withRG.realigned.bam O=sample.withRG.realigned.mdups.bam
METRICS_FILE=sample.withRG.realigned.metric.txt CREATE_INDEX=true VALIDATION_STRINGENCY=SILENT
```

4.4 Base quality recalibration

Bis-SNP heavily depends on base quality score for the genotype probability calculation, but Illumina sequencing reads' raw base quality score could not accurately reflect the error rate of the base. We adapt GATK's base quality score recalibration for Bisulfite-seq/NOMe-seq.There are 3 steps to do base quality recalibration.

4.4.1 Count Covariant

Currently, Bis-SNP only allows recalibration on 3 covariants: ReadGroupCovariate, QualityScoreCovariate and CycleCovariate. `'-T BisulfiteCountCovariates'` specifies the type of analysis. `'-cov ReadGroupCovariate -cov ...'` specify the type of covariants to count. `'-knownSites dbsnp.vcf'` defines the known SNPs position which will not be taken into account as mismatches. `'-recalFile recalFile_before.csv'` defines the output summary table of mismatches.

```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -I sample.withRG.realigned.mdups.bam
-T BisulfiteCountCovariates -knownSites dbsnp.vcf -cov ReadGroupCovariate -cov QualityScoreCovariate
-cov CycleCovariate -recalFile recalFile_before.csv -nt cpu_cores_number
```

4.4.2 Write recalibrated base quality score into BAM file

Use `'recalFile_before.csv'` generated above for the recalibration and add recalibrated base quality score into BAM file. The original base quality score will be kept under `'OQ'` flag in output BAM file. `'-o sample.withRG.realigned.mdups.recal.bam'` specifies the output recalibrated BAM file. `'-T BisulfiteTableRecalibration'` specifies the type of analysis. `'-maxQ 40'` specifies the maximum base quality score in the original BAM file.

```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -I sample.withRG.realigned.mdups.bam
-o sample.withRG.realigned.mdups.recal.bam -T BisulfiteTableRecalibration
-recalFile recalFile_before.csv -maxQ 40
```

4.4.3 Re-Count Covariant

You could omit this step in the pipeline. This step is to validate that if the recalibration step is correct. It counts the mismatches rate with recalibrated base quality score distribution.

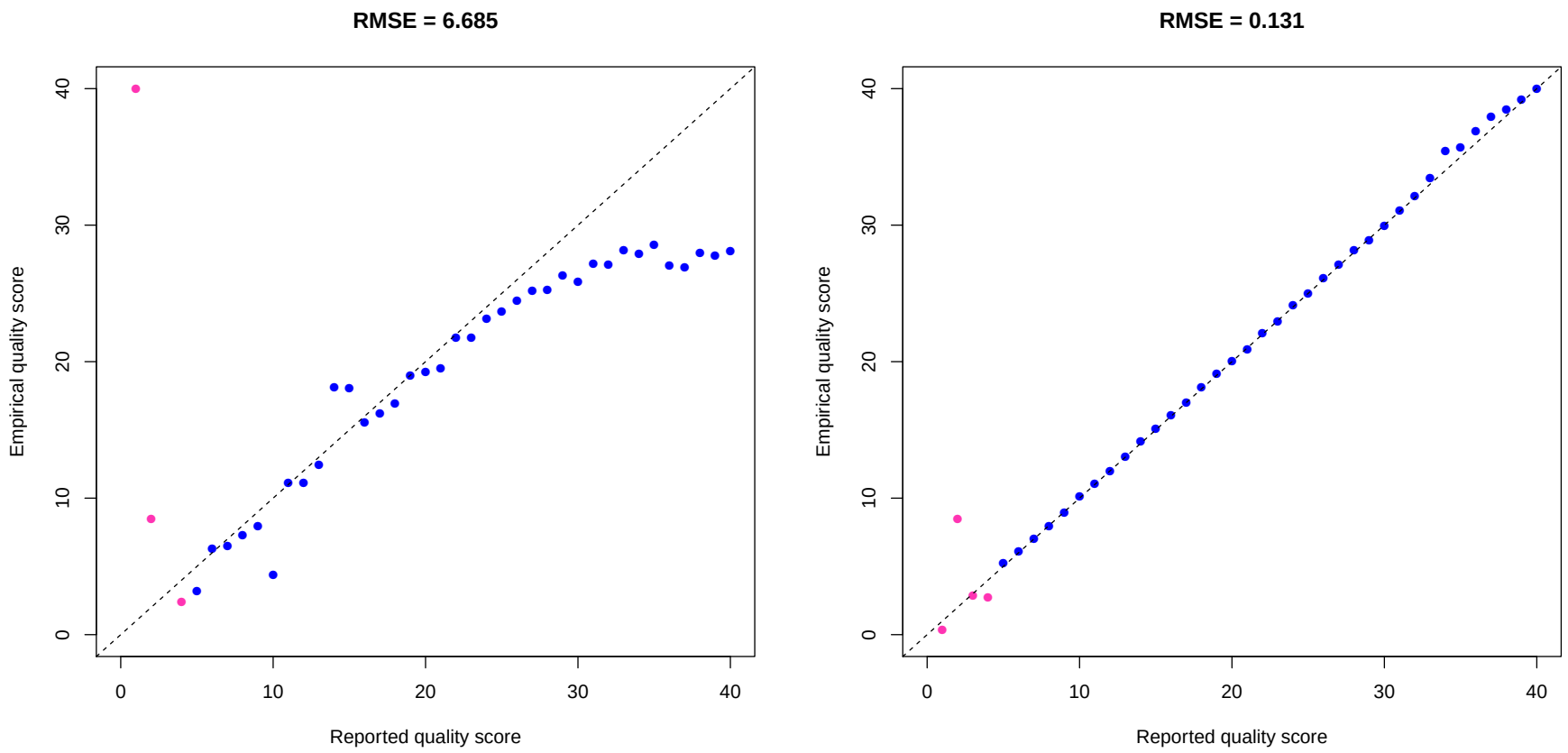
```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -I sample.withRG.realigned.mdups.recal.bam
-T BisulfiteCountCovariates -knownSites dbsnp.vcf -cov ReadGroupCovariate -cov QualityScoreCovariate
-cov CycleCovariate -recalFile recalFile_after.csv -nt cpu_cores_number
```

4.4.4 Generate recalibration plot

You could omit this step in the pipeline. Download BisulfiteAnalyzeCovariates.jar from our website: `http://epigenome.usc.edu/publicationdata/bissnp2011/utilies.html`.Then use the following command to generate plot that shows base quality score distribution before and after recalibration steps (R 2.10 or later version is required to be installed in your machine):

```
java -Xmx4g -jar BisulfiteAnalyzeCovariates.jar -recalFile recalFile_before.csv -outputDir dir_1
 -ignoreQ 5 --max_quality_score 40
java -Xmx4g -jar BisulfiteAnalyzeCovariates.jar -recalFile recalFile_after.csv -outputDir dir_2
 -ignoreQ 5 --max_quality_score 40
```

Here is an example plot that before and after base quality score recalibration:



4.5 Bis-SNP genotyping

‘`-T BisulfiteGenotyper`’ specifies the type of analysis. ‘`-D dbsnp.vcf`’ provide known SNPs information. Bis-SNP could works well without this option, but it will improve the SNPs detection accuracy especially in **low coverage, e.g. less than 10X**. The chromosome order of dbSNP VCF file should be the same as your reference genome file and input BAM file's header. Otherwise, you could use the perl script ‘`sortByRefAndCor.pl`’ we provided to sort dbSNP.vcf as your own reference genome .fasta file. ‘`-vfn1 cpg.raw.vcf`’ specifies the output CpG VCF file name. ‘`-vfn2 cpg.raw.vcf`’ specifies the output SNPs VCF file name. ‘`-stand_call_conf 20`’ defines the likelihood ratio criteria between best and second best genotype. Default value is 20 for high depth of coverage. **For multiple samples with low coverage (more than 100 samples with 4X coverage), the threshold could be defined lower than 10, or even 4. For ultra-high coverage sequencing, such as 50X, you could specify higher threshold to obtain higher accuracy.** ‘`-stand_emit_conf 0`’ specifies the emission threshold. ‘`-mmq 30`’ specifies that reads with minimum mapping quality score more than 30 would be used for genotyping. If bisulfite mapping tools do not provide correct mapping score estimation(most of them just use 255 for all of reads), this reads filter would not take effect. ‘`-mbq 0`’ specifies that the bases with minimum base quality score more than 0 are used for genotyping and methylation calling.

```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -T BisulfiteGenotyper
-I sample.withRG.realigned.mdups.recal.bam -D dbsnp.vcf -vfn1 cpg.raw.vcf -vfn2 snp.raw.vcf
-L chr11:7000000-7100000 -stand_call_conf 20 -stand_emit_conf 0 -mmq 30 -mbq 0
```

After getting the cpg.raw.vcf and snp.raw.vcf files, you should use the following script to sort vcf files by reference genome coordinates:

```
perl sortByRefAndCor.pl [--k chr_field_position] [--c coordinate_field_position] [--tmp dir]
input_files referenceGenome.fa.fai
```

chr_field_position means the chromosome name's position in the input line (1-based), for .bed file, it is 1;
coordinate_field_position means the coordinate's position in the input line (1-based), for .bed file, it is 2.

4.6 Filter fake SNPs

Because of Structual variant, indel, copy number variation mapping bias or strand bias, some of fake SNPs will be called. We use the VCFpostprocessWalker commend to filter out some fake SNPs. Right now, it still using some hard threshold to filter fake SNPs. By default, it filter out SNPs with quality score less than 20, reads coverage more than 120, strand bias more than -0.02, quality score by depth less than 1.0, mapping quality zero reads fraction more than 0.1 and 2 SNPs within the same 20 bp window. `'-T VCFpostprocess'` specifies type of analysis. `'-oldVcf snp.raw.vcf'` specifies the input old VCF files. `'-newVcf snp.filtered.vcf` specifies the output new filtered VCF file. `'-snpVcf snp.raw.vcf` specifies the SNPs used to filter SNPs cluster within defined window. `'-o snp.raw.filter.summary.txt` gives the summary methylation information in the output VCF file.

```
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -T VCFpostprocess -oldVcf snp.raw.vcf
-newVcf snp.filtered.vcf -snpVcf snp.raw.vcf -o snp.raw.filter.summary.txt
java -Xmx10g -jar BisSNP-0.71.jar -R referenceGenome.fa -T VCFpostprocess -oldVcf cpg.raw.vcf
-newVcf cpg.filtered.vcf -snpVcf snp.raw.vcf -o cpg.raw.filter.summary.txt
```

4.7 Generate bed file or wig file for SNP/DNA methylation visualization

4.7.1 Converted to .wig format

```
perl vcf2wig.pl cpg.filtered.vcf CG
```

`'CG'` here could be any other context you want to extract, like `'GCH'` for NOMe-seq (`'H'` means A,C or T in IUPAC code). If context type is not specified, then all of position in vcf file will be extracted out (including heterozygous cytosine pattern loci).

4.7.2 Converted to .bed format

```
perl vcf2bed.pl cpg.filtered.vcf CG
```

Same as above.

4.7.3 Converted to .bedgraph format

```
perl vcf2bedgraph.pl cpg.filtered.vcf CG
```

Same as above.

4.7.4 Extract cytosine coverage information to .bedgraph format

```
perl vcf2bedGraph.pl cpg.filtered.vcf CG
```

Same as above.

5 Usage Detail

5.1 BisulfiteGenotyper

```
java -Xmx10g -jar BisSNP-0.71.jar -T BisulfiteGenotyper [options]
```

5.1.1 Analysis mode options

-h or --help

Generate this help message.

-C or --cytosine_contexts_acquired < cytosine_context >

Cytosine context to be analyze.(CpG, CpH et .al/). e.g. '`-C CG,1`'. CG is the methylation pattern to check, 1 is the C's position in CG pattern. You could specify '-C' multiple times for different cytosine pattern, like: '`-C CG,1 -C CHH,1 -C CHG,1`'. Note: This set of possible context evaluated in order to identify the maximum likelihood context. You can further limit the contexts written to the output file by using the `-output_modes` option. (Default: '`-C CG,1 -C CH,1`')

-sm or --sequencing_mode < mode_of_sequencing >

Run Bis-SNP in different sequencing environment. Bisulfite-seq mode: `BM`; NOMe-seq mode: `GM` (Default: '`-sm BM`')

-nt or --num_threads < number_of_threads >

Enable genotyping in parallel mode and define how many threads should be allocated to running. (Default: '`-nt 1`')

5.1.2 Input options

-I or --input_file < input_file_name >

Input SAM or BAM file(s) for genotyping. *Required*.

-D or --dbsnp < dbsnp_file_name >

dbsnp VCF file (which should sort as the same order of chromosome in BAM file header), provide prior SNP information from dbSNP database. The chromosome contains order of dbSNP VCF file should be the same as your reference genome file and input BAM file's header. Otherwise, you could use the perl script `'sortByRefAndCor.pl'` we provided to sort dbSNP.vcf as your own reference genome fast file.You could download it from our website: dbsnp_135.hg18.sort.vcf.gz for hg18, dbsnp_135.hg19.sort.vcf.gz for hg19. These are already sorted by the chromosome order in the reference genome file we provided. *Strongly Recommend*

-R or --reference_sequence < reference_sequence_file_name >

Reference sequence fasta file (which should have the same chromosome's order as input BAM file's header). Otherwise, you could use ReorderSam.jar in picard tools to reorder BAM file's header. *Required*.

-L or --intervals < intervals_file_name >

A list of genomic intervals over which to operate. Can be explicitly specified on the command line like '-L chr11:7000000-7100000', '-L chr1' or in a bed file. I already provided whole_genome_interval_list.hg18.bed for the whole genome genotype calling in hg18 and whole_genome_interval_list.hg19.bed for the whole genome genotype calling in hg19. If not provided, program will go over all the region specified in BAM's header.

5.1.3 Output options

-vfn1 or --vcf_file_name_1 < output_vcf_file_name >

Output VCF file, when output mode is DEFAULT_FOR_TCGA, this option is used to output all CpG sites. While -vnf2 option is used to output all SNP sites. And -vnf2 option is required at that time. *Required*.

-vfn2 or --vcf_file_name_2 < output_vcf_file_name >

Output VCF file, when output mode is DEFAULT_FOR_TCGA, this option is required and used to store all SNP sites. In the other output mode, it is not required. *Conditional*

-out_modes or --output_modes < output_modes >

What kind of output file do you want.

Options (Default: `'-out_modes DEFAULT_FOR_TCGA'`):

`EMIT_ALL_SITES`: emit all of callable sites into vcf file.

`EMIT_ALL_CONFIDENT_SITES`: emit all of sites above emit confidant threshold into vcf file.

`EMIT_VARIANTS_ONLY`: emit all of SNP sites above emit threshold into vcf file.

`EMIT_ALL_CPG`: emit all of CpG sites above emit threshold into vcf file.

`EMIT_ALL_CYTOSINES`: emit all of Cytosine sites above emit threshold into vcf file.

`EMIT_HET_SNPS_ONLY`: emit all of Heterozygous SNP sites above emit threshold into vcf file.

`EMIT_VARIANT_AND_CYTOSINES`: emit all of Cytosine sites above emit threshold into vcf1 file, and all of SNP sites above emit threshold into vcf2 file.

`DEFAULT_FOR_TCGA`: emit all of CpG sites above emit threshold into vcf1 file, and all of SNP sites above emit threshold into vcf2 file.

-cpgreads,--file_name_output_cpg_reads_detail < file_name_output_detailed_cpg_reads >

Output haplotype CpG reads bed file that contain each CpG's position, methylation and reads name info. (Default: not enabled)

5.1.4 Threshold options

-stand_call_conf < standard_min_confidence_threshold_for_calling >

The minimum phred-scaled threshold for genotype calling that is confident (which is marked PASS in VCF file). (Default: `'-stand_call_conf 20'`)

-stand_emit_conf < standard_min_confidence_threshold_for_emitting >

The minimum phred-scaled threshold for genotype calling that could be emitted. (Default: `'-stand_emit_conf 0'`)

-mmq,--min_mapping_quality_score < min_mapping_quality_score >

Minimum read mapping quality required to consider a read for calling. (Default: `'-mmq 30'`)

-mbq,--min_base_quality_score < min_base_quality_score >

Minimum base mapping quality required to consider a base for calling. (Default: `'-mbq 17'`)

-minConv,--minmum_cytosine_converted < minmum_number_of_cytosine_converted >

Disregard first few cytosines in the reads which may come from incomplete bisulfite conversion in the first few cytosines of the reads, still in test yet. (Default: `'-minConv 0'`)

5.1.5 Advanced options

Notes: Don't change them unless you know the parameters' meaning!!

-hets,--heterozygosity < heterozygosity_rate >

Heterozygosity value used to compute prior likelihoods for any locus. (Default: `'-hets 0.001'`)

-bsRate,--bisulfite_conversion_rate < bisulfite_conversion_rate >

Cytosine bisulfite conversion rate used to compute raw and prior likelihoods for any locus. (Default: `'-bsRate 0.9975'`)

-overRate,--over_conversion_rate < over_conversion_rate >

Cytosine bisulfite over conversion rate used to compute raw and prior likelihoods for any locus. (Default:

‘-overRate 0.0’)

-vdh,--validateDbsnphet < validate_DbSNP_heterozygous_rate >

Heterozygous SNP rate when the loci is discovered as validate SNP in dbSNP. (Default: ‘-vdh 0.1’)

-ndh,--novelDbsnpHet < novel_DbSNP_heterozygous_rate >

Heterozygous SNP rate when the loci is discovered as not validate SNP in dbSNP. (Default: ‘-ndh 0.02’)

-toCoverage,--maximum_read_cov < maximum_read_cov >

Maximum read coverage allowed. (Default: ‘-toCoverage 250’)

-bad_mates,--use_reads_with_bad_mates

If in paired-end mode, allow bad mates that are mapped excessively far away. (Default: not enabled)

-mm40,--max_mismatches_in_40bp_window < maximum_mismatches_in_40bp_window >

Maximum number of mismatches within a 40 bp window (20bp on either side) around the target position for a read to be used for calling. (Default: ‘-mm40 3’)

-rge,--reference_genome_error < reference_genome_error_rate >

Reference genome error, the default value is human genome, in hg16 it is 99.99% accurate, in hg17/hg18/hg19, it is less than 1e-4 (USCS genome browser described); We define it here default for human genome assembly(hg18,h19) to be 1e-6 as GATK did. (Default: ‘-rge 1e-6’)

-tvt,--ti_vs_tv < Transition_rate vs. Transversion_rate >

Transition rate vs. Transversion rate used to compute prior likelihoods for any locus. (Default: ‘-tvt 2’)

-trim5,--trim_5_end_bp < number_of_5_end_bases_disregarded >

How many bases at 5’end of the reads are discarded. (Default: ‘-trim5 0’)

-trim3,--trim_3_end_bp < number_of_3_end_bases_disregarded >

How many bases at 3’end of the reads are discarded. (Default: ‘-trim3 0’)

5.1.6 Other options for debug or still in test

-loc,--test_location < test_location >

Output verbose information in likelihood calculation process, only used in debug.

-fnovd,--file_name_output_verbose_detail < file_name_output_verbose_detail >

Output file’s name that contain verbose information, if not defined, then information will go to standard error stream, for test only.

-ovd,--output_verbose_detail

Enable to output verbose information, for debug only.

-bcm,--bisulfite_conversion_only_on_one_strand

true: Directional Bisulfite-seq protocol which is often used, only bisulfite conversion strand is kept; false: Non-directional Bisulfite-seq protocol, which both of two strands are kept, still in test yet.

-useBAQ,--use_baq_for_calculation

Use BAQ for genotype calculation. Still under test. (Default: not enabled)

5.2 BisulfiteRealignerTargetCreator

```
java -Xmx10g -jar BisSNP-0.71.jar -T BisulfiteRealignerTargetCreator [options]
```

-o

The output target intervals for realignment. *Required*

-known or --known

Input VCF file(s) with known indels. *Strongly Recommend*

-maxInterval or --maxIntervalSize

Maximum interval size to created. Because the realignment algorithm is N*N, allowing too large an interval might take too long to completely realign(Default: `'-maxInterval 500'`)

-minReads or --minReadsAtLocus

Minimum reads at a locus to enable using the entropy calculation. (Default: `'-minReads 4'`)

-mismatch or --mismatchFraction

Fraction of base qualities needing to mismatch for a position to have high entropy. This feature is really only necessary when using an ungapped mapping tool(Default: `'-mismatch 0.0'`)

-window or --windowSize

Window size for calculating entropy or SNP clusters. (Default: `'-window 10'`)

5.3 BisulfiteIndelRealigner

```
java -Xmx10g -jar BisSNP-0.71.jar -T BisulfiteIndelRealigner [options]
```

-targetIntervals or --targetIntervals

Intervals file output from BisulfiteRealignerTargetCreator. *Required*

-known or --known

Input VCF file(s) with known indels. *Strongly Recommend*

-model or --consensusDeterminationModel

Determines how to compute the possible alternate consensus. (Default: `'-model USE_READS'`)

`KNOWNS_ONLY`: Uses only indels from a provided VCF of known indels

`USE_READS`: Additionally uses indels already present in the original alignments of the reads.

-cigar or --IgnoreOriginalCigar

For most of bisulfite mapping program(except Novo-aligner/Bismark with Bowtie2 as i know), they did not

output CIGAR string correctly, so enable this option to ignore the original CIGAR. (Default: not enabled)

-LOD or --LODThresholdForCleaning

LOD threshold above which the realigned will initiate indel realign. This term is equivalent to "significance" - i.e. is the improvement significant enough to merit realignment? Note that this number should be adjusted based on your particular data set. **For low coverage and/or when looking for indels with low allele frequency, this number should be smaller.**(Default: `'-LOD 5.0'`)

-entropy or --entropyThreshold

Percentage of mismatches at a locus to be considered having high entropy.The realigner will only proceed with the realignment (even above the given threshold) if it minimizes entropy among the reads (and doesn't simply push the mismatch column to another position). This parameter is just a heuristic and should be adjusted based on your particular data set.(Default: `'-entropy 0.15'`)

-indels or --indelsFileForDebugging

Output file (text) for the indels found. For debug only.

-snps or --SNPsFileForDebugging

Print out whether mismatching columns do or don't get cleaned out. For debug only.

-stats or --statisticsFileForDebugging

Print out statistics (what does or doesn't get cleaned). For debug only.

5.4 BisulfiteCountCovariates

```
java -Xmx10g -jar BisSNP-0.71.jar -T BisulfiteCountCovariates [options]
```

-recalFile or --recal_file

Filename for the output covariates table recalibration file. *Required*

-cov or --covariate

Covariates to be used in the recalibration. Each covariate is given as a separate covariate parameter. *Required*

-knownSites or --knownSites

VCF/Bed file(s) of known polymorphic sites to skip over in the recalibration algorithm. *Strongly Recommend*

5.5 BisulfiteTableRecalibration

```
java -Xmx10g -jar BisSNP-0.71.jar -T BisulfiteTableRecalibration [options]
```

-o

The output recalibrated BAM file. *Required*

-recalFile or --recal_file

Filename for the input covariates table recalibration .csv file, which is an output of BisulfiteCountCovariates.

Required

-noOQs or --doNotWriteOriginalQuals

If true, we will not write the original quality (OQ) tag for each read. (Default: `false`)

-maxQ or --max_quality_score

The integer value at which to cap the quality scores. (Default: `'-maxQ 50'`)

-pQ or --preserve_qscores_less_than

Bases with quality scores less than this threshold won't be recalibrated. In general it's unsafe to change qualities scores below ¡ 5, since base callers use these values to indicate random or bad bases. (Default: `'-pQ 5'`)

5.6 VCFpostprocess

```
java -Xmx10g -jar BisSNP-0.71.jar -T VCFpostprocess [options]
```

-o

Output summary statistics (not accurate yet, still need to be test). *Required*

-oldVcf or --old_vcf

Input vcf file to be filtered. *Required*

-newVcf or --new_vcf

Output filtered vcf file. *Required*

-snpVcf or --snp_vcf

Input raw SNP vcf file(not filtered by SNP cluster or SB yet), used to filter out SNPs cluster. *Required*

-C or --cytosine_contexts_checked

Specify the cytosine contexts to check (e.g. -C CG -C CH... You could specify '-C' multiple times for different cytosine pattern). (Default: `'-C CG -C CH'`)

-qual or --genotype_qual

Genotype quality score filter for heterozygous SNP. (Default: `'-qual 20'`)

-sb or --strand_bias

Strand bias filter for heterozygous SNP. (Default: `'-sb -0.02'`)

-minCT or --min_ct_coverage

Minimum number of CT reads for count methylation level. (Default: `'-minCT 0'`)

-maxCov or --max_coverage

Maximum coverage filter for heterozygous SNP. (Default: `'-maxCov 120'`)

-qd or --quality_by_depth

Quality by depth filter for heterozygous SNP. (Default: `'-qd 1.0'`)

-mq0 or --mapping_quality_zero

Fraction of mapping_quality_zero filter for heterozygous SNP. (Default: `'-mq0 0.1'`)

-minSNPinWind

Minimum number of SNPs in the window. (Default: `'-minSNPinWind 2'`)

-windSizeForSNPfilter

Window size for detect SNP cluster. (Default: `'-windSizeForSNPfilter 10'`, means +/- 10bp distance, no second SNP there)

-minBQ or --min_bq

Minimum base quality for both of strand. (Default: `'-minBQ 10'`, not enable this option yet)

6 Interpret output files

Bis-SNP output TCGA VCF 1.1 file which is an extension of VCF4.1 format(for VCF4.1 refer to `http://www.1000genomes.org/wiki/Analysis/Variant%20Call%20Format/vcf-variant-call-format-version-41`, for TCGA VCF 1.1 refer to `https://wiki.nci.nih.gov/display/TCGA/TCGA+Variant+Call+Format+(VCF)+1.1+Specification#TCGAVariantCallFormat%28VCF%291.1Specification-metainfo`). In the INFO and FORMAT column, there is some new items that are generated specifically in Bis-SNP:

6.1 VCF file

6.1.1 INFO column

Context: Cytosine pattern. like "CG" for homozygous CpG, "CR" for heterozygous CpG (CpA/CpG). The code rule obeys IUPAC code rule `http://www.bioinformatics.org/sms/iupac.html`. It is the summary across all of the samples.

CS: Cytosine pattern's strand.

6.1.2 FORMAT column

This is for each of sample:

CP: The Best Cytosine pattern in this locus for this sample. like "CG" for homozygous CpG, "CR" for heterozygous CpG (CpA/CpG).

CM: Number of Cytosine reads(methylated) in this Cytosine position.

CU: Number of Thymine reads(unmethylated) in this Cytosine position.

BRC6: Bisulfite read counts: 1) number of C in cytosine strand, 2) number of T in cytosine strand, 3) number of A/G/Others in cytosine strand, 4) number of G in guanine strand, 5) number of A in guanine strand, 6) number of C/T/Others in guanine strand

6.2 CpG reads file

Output each CpG in each reads. The CpG in the same reads would have the same readID. Here is the explanation of each column:

chr: chromosome name

pos: genomic coordinate.(1-based)

methyStatus: methylation status of this CpG. m: methylated. u: unmethylated

baseQ: phred-scale base quality of the Cytosine in this CpG site.

strand: strand of this CpG.

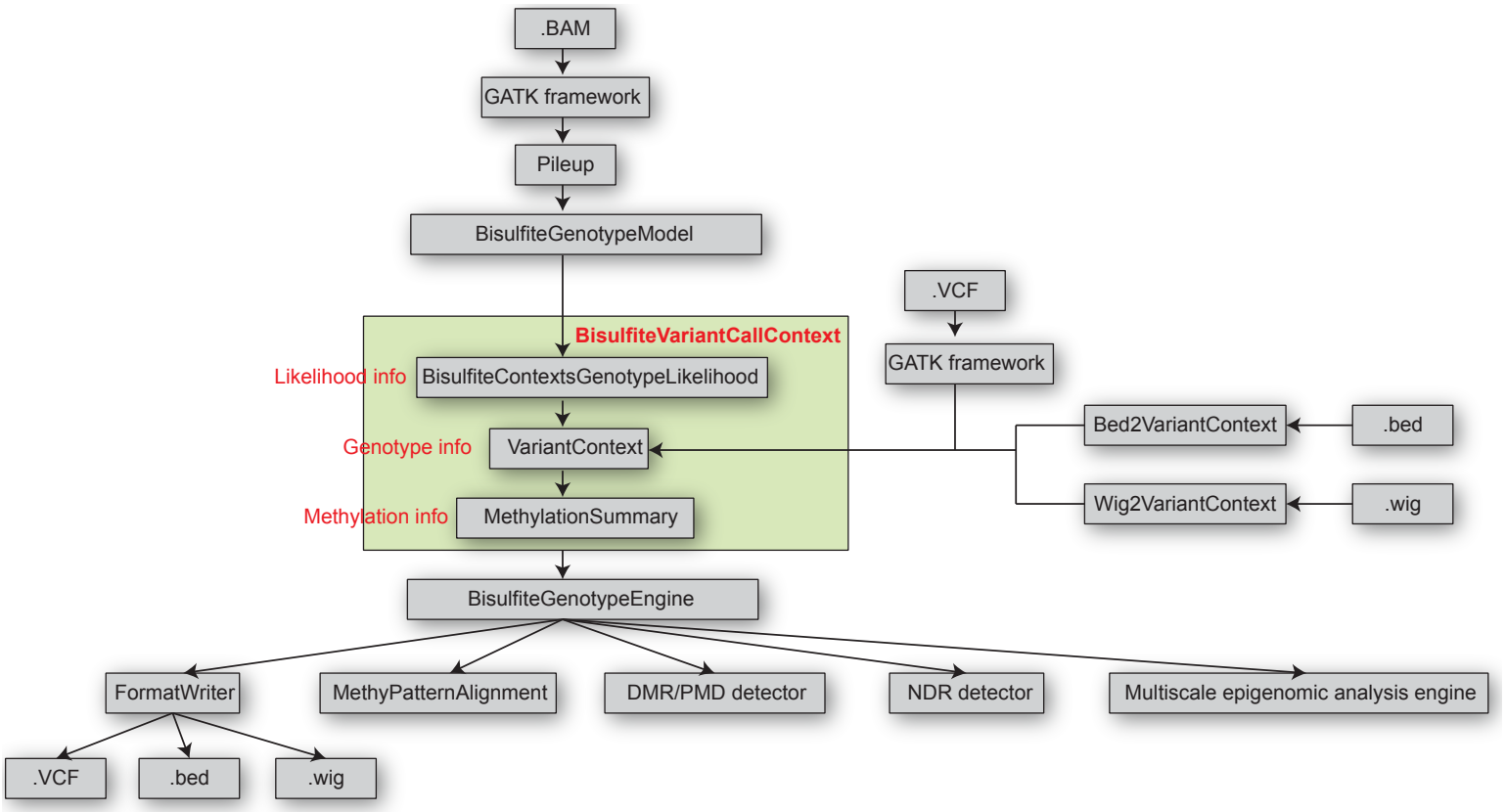
readID: encrypt readID. The same reads would have the same readID

7 Additional useful script

Some additional useful perl scripts to convert from VCF file to methylation CpG bed file or wiggle tract file for visualization. The detailed information refer to `http://epigenome.usc.edu/publicationdata/bissnp2011/utilies.html`

8 Bis-SNP API structure

This part is for other programmer to build tools on top of Bis-SNP genotype engine. Java API doc will be available soon.



9 Build on source code

Source code is available on SourceForge website, which could be checkout by command:

"svn checkout svn://svn.code.sf.net/p/bissnp/code/trunk"

All of the required libraries are available in `http://svn.code.sf.net/p/bissnp/code/trunk/lib/`.

10 Contact for help

For any of question on Bis-SNP, please send email to lyping1986@gmail.com or benbfly@gamil.com