Variant calls for *Medicago* HapMap accessions using the Mt5.0 assembly

25 February 2022

This document describes the SNP and indel calls for the *Medicago* HapMap population based on the Mt5.0 reference genome.

IMPORTANT NOTE:
HM241 variant calls are based on a mixture of reads from HM241 and HM022. We recommend that HM241 not be included in any analyses conducted with these data.

In data downloaded before February 2022, the variants for three lines were swapped:
HM224 calls were labeled as HM235
HM225 calls were labeled as HM224
HM235 calls were labeled as HM225

This issue has been corrected in the data currently available, so we recommend downloading the current data.

Please be aware that the coverage was quite low for most of these accessions, and that there are additional filters (*e.g.*, higher depth thresholds or masking SNPs near indels) that you may wish to apply. The raw variant call file will allow you to do your own filtering and variant processing, if you so wish. *You should also read the variant calling pipeline details so that you know how these data were created.*

Description of Files

Mt5_raw_variant_calls.2021-09-29.vcf.gz	Raw output from FreeBayes, including invariant sites (GVCF format). Please note that four accessions that are not part of the HapMap population were included in the variant calling pipeline, but have been removed from this file. The data in the INFO field was calculated with these four samples included. There is an index (.csi) and tabix (.tbi) file as well. If you want to do your own filtering, or are planning to run population genetic analyses that require invariant sites, this is the file for you. Note that this file, and all other files here, includes both SNPs and indels.
Mt5_qual30_primitives.2021-09-29.bcf	Just variant sites, transformed to aid in

	interpretation. Invariant sites were removed by retaining only rows with a variant quality score > 30. Adjacent variants that were grouped into a single row were separated. Please see Step 5 in the pipeline description for more details. This file may be useful if you want just the polymorphisms.
Mt5_qual30_primitives_imputed.2021-09-29.bcf	Imputed variants, only for *Medicago truncatula* genotypes. This file is useful for GWAS on *M. truncatula*.

Variant calling pipeline details

1. **Reads from the SRA**

   Variant-calling was performed on short-read Illumina data downloaded from the NCBI SRA. A full list of SRA accessions can be found in the SRA_accessions_list.tsv tab-delimited spreadsheet. Most of the genotypes included belong to the species *Medicago truncatula*, but there are also some other species of *Medicago*.

2. **Read cleaning with BBDuk and TrimGalore**

   The reads were cleaned by using BBDuk (from BBTools v38.86) and TrimGalore (v0.5.0). First, BBDuk was used to remove potential human, *Ensifer* (*Sinorhizobium*), and PhiX contaminants with the following command:

   ```
   bbduk.sh in1="$R1" in2="$R2" \
       ref="human.fasta,ensifer.fasta,phix.fasta" \
       k=25 hdist=0 rskip=4
   ```

   where *R1* and *R2* are the files with paired reads, *human.fasta* is the hg38 human genome sequence, *ensifer.fasta* is the Rm1021 genome sequence, and *phix.fasta* is the PhiX genome sequence.

   Then, TrimGalore was used to trim off adaptors and low quality bases:

   ```
   trim_galore --quality 30 --stringency 3 -e 0.1 --length 45 --paired \
       "$R1" "$R2"
   ```

   where *R1* and *R2* are the paired outputs from BBDuk. This command only reports reads if they are at least 45 bp long (--length 45).

3. **Read alignment with bwa mem**

Reads were aligned to the Mt5.0 reference genome available from NCBI (GCA_003473485.1 or GCA_003473485.2) by using bwa mem (v0.7.17):

```
bwa mem \
    -k 19 -w 100 -d 100 \
    -r 1.5 -c 10000 -A 1 -B 4 \
    -O 6 -E 1 -L 5 -U 9 -T 0 \
    "$REFERENCE" "$r1" "$r2"
```

where *REFERENCE* is the Mt5.0 genome assembly, and *r1* and *r2* are the filtered and trimmed reads from the previous step. Note that the only non-default option is -T, which is the minimum mapping quality allowed—setting this to zero retains all alignments.

Most of the variant calling was performed based on the alignment to GCA_003473485.1, but later, the reads were aligned to GCA_003473485.2, and variant calls for the nine extra contigs in that later assembly version were added (the two assembly versions were identical aside from the additional contigs).

Samtools fixmate and samtools markdup were then used to mark potential PCR duplicates.

4. Variant calling with FreeBayes

Variants were called by using FreeBayes (v1.3.2-46-g2c1e395) with population priors turned off (variant calls are not affected by population allele frequencies) and some options to reduce memory usage and computational time:

```
freebayes
    --region "$REGION" \
    -f "$REFERENCE" \
    -p 2 \
    -L list_of_input_bam_files.txt \
    --min-mapping-quality 30 \
    --no-population-priors \
    --use-best-n-alleles 6 \
    --gvcf
```

where *REFERENCE* is the Mt5.0 genome sequence (as above). “--use-best-n-alleles 6” reduces memory usage, at the cost of reporting no more than six alleles; in high diversity, complex regions, this may sometimes cause some true alleles to be missed; this is most likely to affect the non-*truncatula* accessions. The minimum mapping quality of 30 essentially retains only reads that align to one location. FreeBayes was run separately on 300kb genomic regions to reduce time, and then the results were combined by using vcflib (v1.0.1, see below). The output of this command is the file **Mt5_raw_variant_calls.2021-09-29.vcf.gz**, which contains the raw variant calls from FreeBayes and includes information on invariant sites. As noted above, four non-HapMap accessions were removed from this file before it was posted here.

Command for combining GVCF files from multiple regions:

```
cat $(ls "directory_with_region_gvcfs"* | sort -t: -k 1b,1 -k 2n,2) \
    | vcffirstheader \
    | vcfstreamsort -w 1000 \
    | vcfuniq \
    > merged.gvcf
```

5. Further processing and imputation with Beagle

To obtain a file with only variant sites, the raw output was filtered by the variant quality: only variants with a QUAL value > 30 were allowed. FreeBayes often combines adjacent variants into a single row (record) in its VCF files. We split these into individual records using the vcfallelicprimitives program from vcflib. The output of this program is the “primitives file.

```
vcffilter -f "QUAL > 30” merged.gvcf \
    | vcfallelicprimitives \
    | bcftools norm -O u -f "$REFERENCE" -d any \
    | bcftools annotate -O u --set-id 'freebayes-var-%CHROM-%POS-%FIRST_ALT' \
    | bcftools view -O b
```

In addition to filtering and splitting adjacent variants into one variant per site, this also removed any duplicate records introduced by the splitting step (bcftools norm -d any), added variant IDs (bcftools annotate), and converted to bcf format (bcftools view -O b).

For a subset of the accessions that are *M. truncatula*, rather than some other species of *Medicago*, we imputed using Beagle (v4.1, specifically 27Jan18.7e1).

To do this, we filtered out the non-truncatula genotypes and prepared the file with bcftools, then ran Beagle with mostly default options, except for a few changes to decrease the run-time:

```
bcftools view -Ou -S "$SAMPLES" "$INFILE" \
    | bcftools view -Ov --include 'N_PASS(GT!="mis")>0' \
    | bcftools +fixploidy -Oz > tmp.vcf.gz

# The output of the above command was then imputed:
java -jar beagle.jar gt=tmp.vcf.gz \
    chrom="$REGION" \
    out="imputed.${REGION}" \
    impute=true \
    gprobs=true \
    niterations=5 \
    ne=1000000 \
    err=0.0001 \
    modelscale=0.8 \
    maxlr=200
```

The non-default option here is maxlr (default is 5000). The imputation was run on each chromosome or contig separately (to save time), and then the results were combined by using bcftools concat, and any remaining invariant sites were also removed. The result is the imputed file.