INDELible

Glen Morrison & Chris Fiscus

Rationale and Project Summary

Indels, short for insertions or deletions, are mutations

Data

For this project we used Illumina whole genome sequencing reads from Cao, J. *et al.* Whole-genome sequencing of multiple Arabidopsis thaliana populations. Nat. Genet. 43, 956-963 (2011) representing 80 samples of *Arabidopsis thaliana* sequenced across 175 sequencing runs. We downloaded this dataset from the European Nucleotide Archive (ENA) using NCBI Short Read Archive accession SRA029270. We used the ENA website interface to create a file consisting of the sample accession, secondary sample accession, run accession, and FTP addresses for each sequencing run. This file is included in the repository under data/file_list.txt.

Pipeline

Core Scripts

0_setup_ref.sh
Slurm script that downloads the *Arabidopsis thaliana* reference genome sequence from EnsemblGenomes and references it with the *index* command from bwa v. 0.7.12.

1_dl_align.sh
Slurm script that downloads the sequencing reads from each sequencing run listed in the file list, quality trims using sickle v. 1.33, aligns to the reference genome using *bwa mem*, then uses samtools v. 1.4.1 to convert the resulting sequence alignment map (SAM) file into a sorted binary alignment map (BAM) file with duplicates removed. For computational efficiency, this script runs as an array job, launching an instance for each sequencing run listed in the file list and allowing each run to be processed simultaneously. The default parameters were used when running *sickle* and *bwa mem*. The read groups were set by *bwa mem* upon aligning the reads to the reference genome. The value of the SLURM_ARRAY_TASK_ID variable was used as the read group ID while the secondary sample accession was used as the read group sample (SM).

2_call_snps.sh
Slurm script that uses freebayes v. 1.1.0 to identify variants in the population of 80 *A. thaliana* individuals. Prior to running this script, a list of BAM files to process was generated by running the following command:

```
ls results/*.bam > ./data/bam_list.txt
```

For computational efficiency, only 4 alleles at each site were considered when running *freebayes*, which was set using the *–use-best-n-alleles 4* argument.

3_filter_vcf.sh
Slurm script that uses the *vcffilter* command from vcflib to filter the vcf file produced by *freebayes*. We used a hard filter, keeping only indels that had at least 10 reads and a quality score of 20.

4_indel_size_and_position.py

Shiny Application

Additional Scripts

dl_results.sh
To facilitate sharing the filtered vcf file produced by script 3 above and to make this project reproducible independent of the high performance computing center (i.e. biocluster) at UC Riverside, the data was uploaded to figshare. This script creates a results folder and downloads the filtered vcf file to this directory.

Results

Acknowledgements

The authors would like to thank Professor Jason Stajich for guidance and suggestions throughout the course of this project. All code is hosted on github at this link. Our Shiny application was deployed to the cloud with Shinyapps.io and can be accessed here.