CS 121 - Genetic Variant Caller

Genetic Variant Caller Project Report

Introduction

This project involved the development of a SNP (Single Nucleotide Polymorphism) caller using BAM files, which contain aligned DNA sequences. The primary goal was to identify genetic variations at specific genomic positions and compute the likelihood of different genotypes based on the data observed. The relevant SNP data, including chromosomal positions, reference alleles, and alternate alleles, were extracted from a `putative_snps.tsv` file.

Tools and Libraries Used

For this project, PySAM, a Python library, was utilized for its efficiency in handling large genomic datasets. It facilitated the reading and manipulation of the BAM file. Other libraries included:

- **pandas**: For data manipulation and analysis.

- **NumPy**: For numerical operations.
- **sys**: To execute the Python script with command-line arguments.

Methodology

Pseudocode Overview

Before coding, I developed a pseudocode to outline the steps involved in the project:

1. Import necessary libraries (pandas, NumPy, PySAM, sys).
2. Open the BAM file and the SNP data file.
3. Define a function to retrieve read counts and quality scores.
4. Define a function to calculate posterior probabilities using a Bayesian model.
    - need to calculate the priors, the probability of the data, and the probability of the genotype given the data.
5. Process each SNP
6. compute posterior probabilities for each genotype
7. output the results.

Calculating Posterior Probabilities Function

The calculation of posterior probabilities was based on Hardy-Weinberg equilibrium, which considers the minor allele frequency (maf). The formulas used were:

- **P(AA) =** $(1 - maf)^2$
- **P(AB) =** $2 * (1 - maf) * maf$
- **P(BB) =** $(maf)^2$

The Bayesian model required:

- The probability of the data given the genotype.
- The prior probabilities of each genotype.

- The error rate.

The likelihoods of observing the data given the genotypes were calculated as follows:

- For AA: **(1 - error_rate) ^ n_ref * (error_rate) ^ n_alt**
- For AB: **0.5 ^ (n_ref + n_alt)**
- For BB: **(1 - error_rate) ^ n_alt * (error_rate) ^ n_ref**

The posterior probability of each genotype given the data was computed using Bayes' theorem.

Extracting Read Counts and Quality Scores Function

The function for extracting genetic data was designed to pinpoint specific SNP locations for analysis. Using PySAM's pileup function, it distinguished reads supporting the reference versus alternate alleles, considering genomic indexing conventions. Additional conditions in the code filtered out deletions or skips in the reads, which could skew allele counting.

The quality of each read was assessed to ensure the reliability of the SNP calling. The average error probability at each SNP position was calculated based on the quality scores, ensuring precision in the posterior probability calculations.

Implementing -

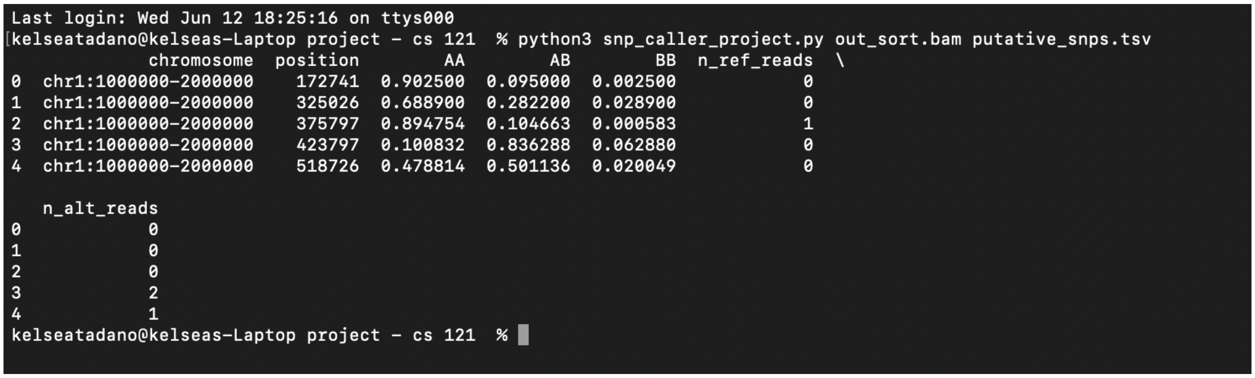
To run the program in the folder with the putative_snps, bam file, and my python script -

```
python3 snp_caller_project.py out_sort.bam putative_snps.tsv
```

Results

The script processes each SNP from the `putative_snps.tsv` file, utilizing the functions defined to extract data and calculate probabilities. The output includes the posterior probabilities for each genotype, the count of reference and alternate alleles at each position, and the averaged error rate.

A reference of what the results look like in terminal -



```
Last login: Wed Jun 12 18:25:16 on ttys000
[kelseatadano@kelseas-Laptop project - cs 121  % python3 snp_caller_project.py out_sort.bam putative_snps.tsv
             chromosome  position        AA        AB        BB  n_ref_reads  \
0  chr1:1000000-2000000    172741  0.902500  0.095000  0.002500            0
1  chr1:1000000-2000000    325026  0.688900  0.282200  0.028900            0
2  chr1:1000000-2000000    375797  0.894754  0.104663  0.000583            1
3  chr1:1000000-2000000    423797  0.100832  0.836288  0.062880            0
4  chr1:1000000-2000000    518726  0.478814  0.501136  0.020049            0

   n_alt_reads
0            0
1            0
2            0
3            2
4            1
kelseatadano@kelseas-Laptop project - cs 121  % 
```

Conclusion

This project demonstrates the application of bioinformatics tools and statistical models to identify and analyze SNPs from genomic data. By integrating concepts from population genetics and applying computational methods, this project demonstrates a robust approach to understanding genetic variations.

Sources

- Hardy-Weinberg Equilibrium Calculations
- Namipashaki, A., Razaghi-Moghadam, Z., & Ansari-Pour, N. (2015). The Essentiality of Reporting Hardy-Weinberg Equilibrium Calculations in Population-Based Genetic Association Studies. *Cell Journal*, 17(2), 187–192. https://doi.org/10.22074/cellj.2016.3711