Introduction of each R scripts

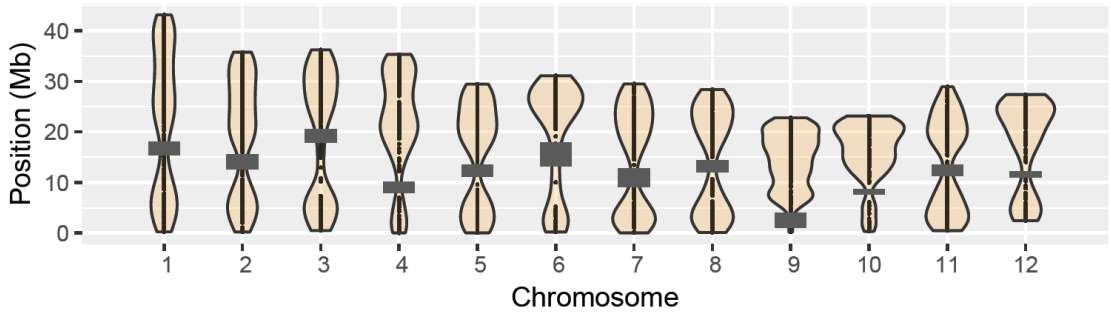
demo_breakpoint site calculation.R

This script is designed to identify the recombination breakpoint sites along the rice chromosome using ABH-form csv files. The recombination frequency is important for QTL mapping, as the higher frequency creates much diversity within a population. Therefore, identifying the recombination breakpoints as well as their amount within a population is a crucial step before conducting QTL mapping.

- Input: ABH form csv file
  - Column：ID of SNP(chromosome number_physical position)
  - Row: ID of individual (first row is the chromosome number)

- Output: dataframe with chromosome, breakpoint position, left/right-flanking markers
  - Column: chromosome number, breakpoint physical position, left and right flanking marker positions of the breakpoints.
  - Row: no. for each position

- Visualization:



Demo_fasta file generation.R:

This code is made for the generation of fasta file for given intervals using the genotype information provided by ABH file. At first, you have vcf file that gives you

the position of each SNP marker. Then after the genotyping step, the ABH file reveals the most reasonable genotyping results. Combining the position information from vcf file and the genotype information from the ABH file, we can generate the most correct sequence of a given interval.

- Input data:
  - ABH form file (described above)
  - Reference fasta file

- Function: fas_generate (chr,position,one_side_length ,vcf_name =NA,ref_data, vcf_reffas_path = NA)

- Arguments:
  - Chr: chromosome number of the given interval
  - Position: the center physical position of the interval
  - On_side_length: the half length of the interval, you can determine the interval length by adjusting this value, and the total length of the interval will be twice the parameter value.
  - vcf_name: name of the vcf file, you must divide the vcf file according to the chromosome number. For example, 12 chromosome results in 12 vcf file, and chromosome one will have the name of your_vcf_name.chr1.vcf.
  - ref_data: name of reference fasta file
  - vcf_reffas_path: the direct path that contains both vcf file and the reference data
- Output: fasta file for given positions

demo_gene id searching in RAPDB.R

The aim of this script is to automate the searching progress for genes in RAPDB (rice database). After QTL mapping, we might get intervals with candidate genes located within. The problem is, if the range is too wide, there might be thousands of genes, making it hard to identify the target genes. By utilizing the web crawler technics, the script can automatically download the gene information for multiple given intervals, combining them into a single dataframe, making it much easier to investigate or classify these genes. Besides, it's important to note that you need to download chromedriver.exe and java before executing the script.

- Input:
  - Interval csv file: 3 columns (chromosome number, start position and end

position)

- Output:
  A txt file containing all genes from given intervals

demo_snp density.R

The script is made to investigate the SNP density in certain window size (here, 0.1Mb is used). Generally, common SNP density-related package such as CMplot (Yin et al., 2021) will produce SNP density heatmap along the chromosome, making it easy to have a overview of SNP distribution. However, the actual amount of SNP within each window remains unknown. Therefore, this script is designed to count the SNP number, and show the position where the SNP amount is the most. Also, the distance between each SNP can be counted to observe the biggest gap in the whole genome.

- Input: vcf file (described above)

- Output: SNP amount within each window and the physical distance between each two SNPs

Reference

Yin, L., Zhang, H., Tang, Z., Xu, J., Yin, D., Zhang, Z., Yuan, X., Zhu, M., Zhao, S., & Li, X. (2021). rMVP: a memory-efficient, visualization-enhanced, and parallel-accelerated tool for genome-wide association study. *Genomics, proteomics & bioinformatics*, *19*(4), 619-628.