Name: Kevin Muirhead

MDSC 679: Machine Learning Project 1

Abstract

Introduction

*Arabadopsis thaliana* is an important model organism which has a relatively short life cycle, a small genome (135 megabase pairs (Mbp)) and highly curated genomic database TAIR10 (). Having these features allows for a multitude of possible research interests of economic and commercial value for agriculture and subsistence farming. *A. thaliana* is useful for studying multifactorial traits such as flowering time (FT), which has been extensively studied as a model for this trait. The flowering time trait is regulated by the interaction of multiple genes on various chromosomes and stimulus from the environment.

The goal of this particular study is to find the relationship of genotype to phenotype for the flowering time (FT) trait. Leveraging this knowledge I hope to. Since this is a

In this study only homozygous genotypes were observed. Making it quite difficult to use the fischer exact test for Hardy-Weinberg Equilibrium (HWE) for each Single Nucleotide Polymorphism (SNP) variant because of the significant derivations to HWE.

Methods

In order to study the genetic causation of flowering time (FT), a phenotype to genotype mapping study was conducted to assess the underlying mechanism of the trait. A genome wide association study (GWAS) was conducted using 251 individuals. A panel of 214553 variant SNPs were genotyped and analyzed using a custom pipeline implemented in the python and the R programming languages.

## The pipeline workflow is captured in the following figure x. In the first step, the quality control workflow was developed in python for filtering the data for further analysis. The quality\_control.py script performs the following tasks Quality filtering, Association mapping, quality control and formatting input files for the machine learning models.

## Quality Filtering

Flowering time phenotypes are filtered for “NA” values, which indicated that those particular genotypes had missing data and were removed. A total of 238 phenotypes were retained for further processing of individual genotypes. Genotypes were first filtered for biallelic SNPs. Genotypes with flowering time phenotype data were assessed for minor and major allele frequency. Alleles of each variant were observed and counted, and minor and major alleles were determined. After filtering by phenotypes and biallelic SNPs, the Minor Allele Frequency (MAF) of each variant was calculated. Variants with a MAF of greater than or equal to 0.01 or 1% were retained and considered for further analysis.

Association Mapping

A total of 230 genotypes and 214219 variants were retained before association testing of SNP variants using the rMVP Memory-efficient, Visualization-enhanced, and Parallel-accelerated Tool for Genome-Wide Association Study R package version v.1.0.0 (). A pedigree format file (PED) and a genotypes map file was generated for input into Plink version 1.90b6.21 (blank). Plink was ran with the recode vcf parameter option for converting the PED and genotypes MAP file to Variant Call Format (VCF) file using the os.system call. A phenotypes file was generated using the list of filtered genotype ids in python for input into the rMVP.Data function of the rMVP R package. All variants were subjected to Generalized Linear Model (GLM) (), Mixed Linear Model (MLM) and FarmCPU associations tests in the rMVP R Package and compared. The density of genotyped SNPs is shown in figure x giving a layout of concentration of variants that the three models, GLM, MLM and FarmCPU assessed for tests of association with the flowering time (FT) phenotype. The type I error rate was assessed for each of the three models using a Quantile-Quantile Plot (QQ-Plot) shown in figure x. The QQ-Plot indicates that the type I error rate for the MLM model is under control and was selected as the association testing model conducted for further analysis. A Manhattan plot (−log10(P) genome-wide association plot) was generated to visualize significant SNP variants across the entire genome in the locations that were genotyped. As shown in figure x the Manhattan plot of the MLM model, the P-values are at a reasonable value.

Quality Control

P-values of tested variants were subjected to adjusted p-value calculations for Bonefferoni correction and q-values using the calculate\_adjusted\_pvalues.R script which uses the pvalue.adjust function and the qvalue library in R. It was determined that an alpha value less than or equal to 0.05 were retained for further processing as this indicated that they were significant with a False Discovery Rate of 5%. A total of xxx individual genotypes and XXX variants were retained for further processing.

File Formatting For Machine Learning

Genotypes at each SNP variant that passed the filtering procedure were encoded 0 for the major allele and 1 for the minor allele. Genotypes and corresponding phenotypes were written to a file for input into the machine learning models using python.

Machine Learning Models

The second step of the workflow executes the machine learning models, polygenetic linear regression (paper), LASSO L1 Regularization (SVM) (blank), RIDGE L2 Regularization (blank).

The dataset criteria for input into the three models were the following. Genotypes were filtered using MAF >= 0.01, adjusted p-value (q-value blank) alpha <= 0.05 for an FDR of 5%, and the major allele (reference) was encoded as 0 and the minor allele (alternative) was encoded as 1 and Phenotypes were filtered by missing values and filtered genotype ids in the quality control step. There were 230 genotyped individuals and xxx SNP variants used as input for each of the three models. All machine learning models were implemented in the python scikit-learn package version 0.24.1 (blank). The encoded genotype files were parsed using the python pandas version 1.2.3 (blank) and loaded into a numpy array using the python numpy package version 1.20.2 (blank) for each model. Then a cross-validation approach ( which one ) was used in order to validate the hyperparameter (look up more information). Then a test set was used for prediction.

## Need to structure the artificial neural network (paper), Support Vector Machine (SVM) (blank), and Random Forest (blank) methods. Then talk about it in the results/discussion.

Computation Resources

Program implementation, development and analyses were performed on a HPC cluster running Slurm as the job scheduler (arc.ucalgary.ca) in interactive mode using the cpu2019 partition with 38 GB RAM on one CPU per node.

A search for *Arabadopsis thaliana* genes known to be associated with flowering time was conducted using the TAIR website (<https://www.arabidopsis.org/>). A total of 79 genes were found using the search term “flowering time” and the accession numbers were downloaded. The 79 genes found in the TAIR database were compared to the genes with variants to assess whether or not these “flowering time” associated

genotypes.csv

Annotations (gene\_model.gff)

Results

Discussion

Conclusion

References:

1. Reference Dr. Long’s paper.
2. Tanya Z. Berardini, Leonore Reiser, Donghui Li, Yarik Mezheritsky, Robert Muller, Emily Strait and Eva Huala. The Arabidopsis Information Resource: Making and mining the "gold standard" annotated reference plant genome. genesis 2015 doi: 10.1002/dvg.22877
3. Salomé PA, Bomblies K, Laitinen RA, Yant L, Mott R, Weigel D. “Genetic architecture of flowering-time variation in Arabidopsis thaliana.” Genetics. 2011 Jun;188(2):421-33. doi: 10.1534/genetics.111.126607. Epub 2011 Mar 15. PMID: 21406681; PMCID: PMC3122318.
4. Lilin Yin, Haohao Zhang, Zhenshuang Tang, Jingya Xu, Dong Yin, Zhiwu Zhang, Xiaohui Yuan, Mengjin Zhu, Shuhong Zhao, Xinyun Li, Xiaolei Liu, “rMVP: A Memory-efficient, Visualization-enhanced, and Parallel-accelerated tool for Genome-Wide Association Study”, Genomics, Proteomics & Bioinformatics, ISSN 1672-0229, <https://doi.org/10.1016/j.gpb.2020.10.007>. (2021)
5. F. Bouché, G. Lobet, P. Tocquin, C. Périlleux, “FLOR-ID: An interactive database of flowering-time gene networks in Arabidopsis thaliana.” Nucleic Acids Res. **44**, D1167–D1171 (2016).
6. Pedregosa et al., “Scikit-learn: Machine Learning in Python”, JMLR 12, pp. 2825-2830 (2011).
7. The pandas development team, “Pandas”, Zenodo, version, doi: 10.5281/zenodo.3509134 (2020)

Look at references in here https://www.pnas.org/content/116/36/17890

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2894516/