Methodology report

Teddy ALEXANDRE, Arthur BABIN, Group 9, Bio project

1. Introduction

After presenting the dataset used for the project and describing the explanatory variables as well as the target variables, we will now focus on the methodology we adopted for the project. We will describe our goals, how to reach them and the first results we obtained.

2. Methods

2.1. Classification model based on Manhattan plots

A Manhattan plot, which plots the association statistical significance as $-log10(p_{value})$ in the y-axis against chromosomes in the x-axis, is a good way of displaying millions of genetic variants in one figure. One can easily spot regions of the genome that cross a particular significance threshold. As we can see in the Figure 1, each dot represent a SNP and dots above the separation are considered as significant regarding the Bonferonni correction ($p_{value} < \frac{0.01}{\#SNPs}$)

Our goal in this method is to train a KNN model with the different loci in the genome that we identified as significant for the trait. For example in Figure 1 we can observe a significant line for the chromosome n°2, chances are that this line corresponds to a locus coding for the seed number number per panicle. Indeed in Figure 2a and 2b both SNP n°8102 and n°8152 analysis show that being homozygous dominant increase on average by 0.3 wich is very interesting.

Having identified the different loci coding for the trait we divide our population in two categories: those who have a majority of advantageous SNPs and those who don't. Then we just have to use a KNN model in order to build a model that can predict if a genome will result in an interesting value for the trait. Moreover with this method, based on methods used in GWAS, it is easy to explain the link between the input and the output and it can conveniently be used by scientists.

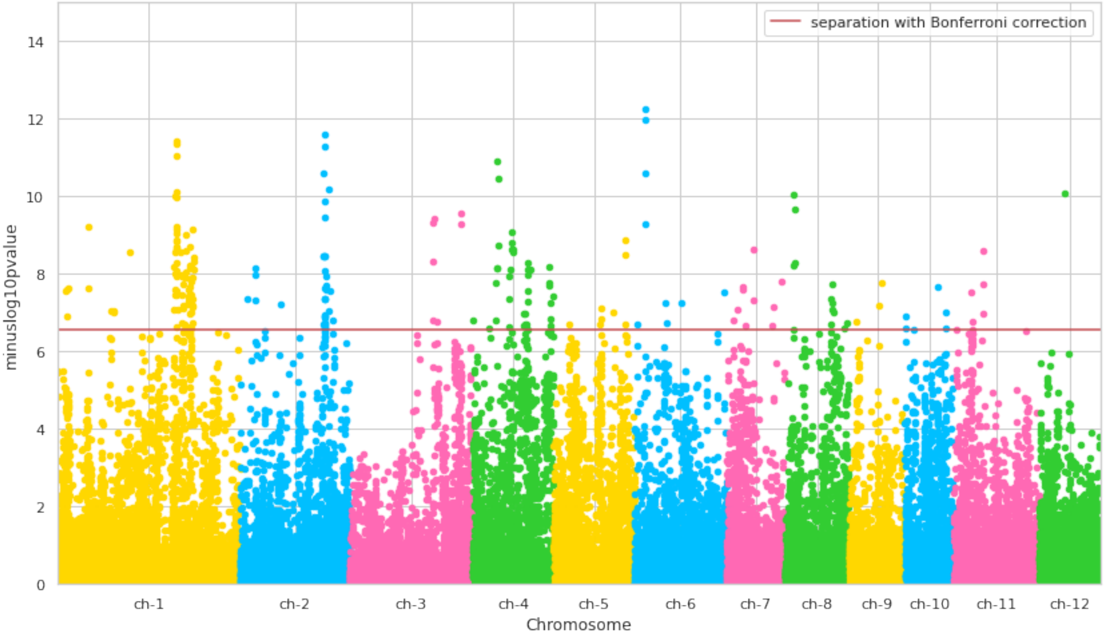
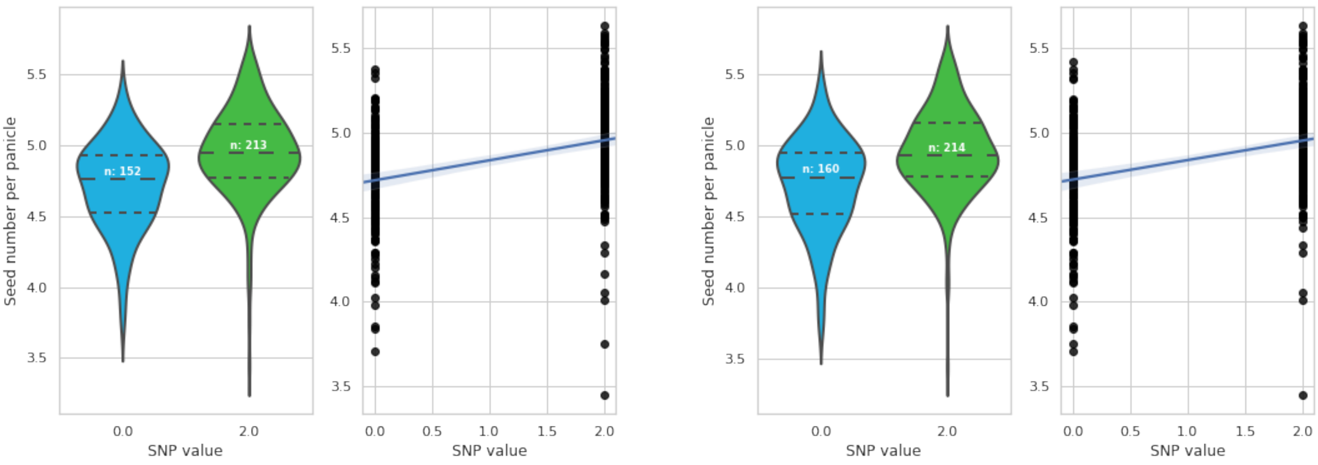


Figure 1. Manhattan plot for Seed.number.per.panicle

2.2. Regression model

In order to precede our methodology about the regression we chose to do, we first realized a few violinplots on several SNPs to detect a tendancy. With the two figures below, we observe a distribution that feels recurrent on some SNPs, which is coherent with what we want to evidence.



(a) SNP n°8102 analysis

(b) SNP n°8152 analysis

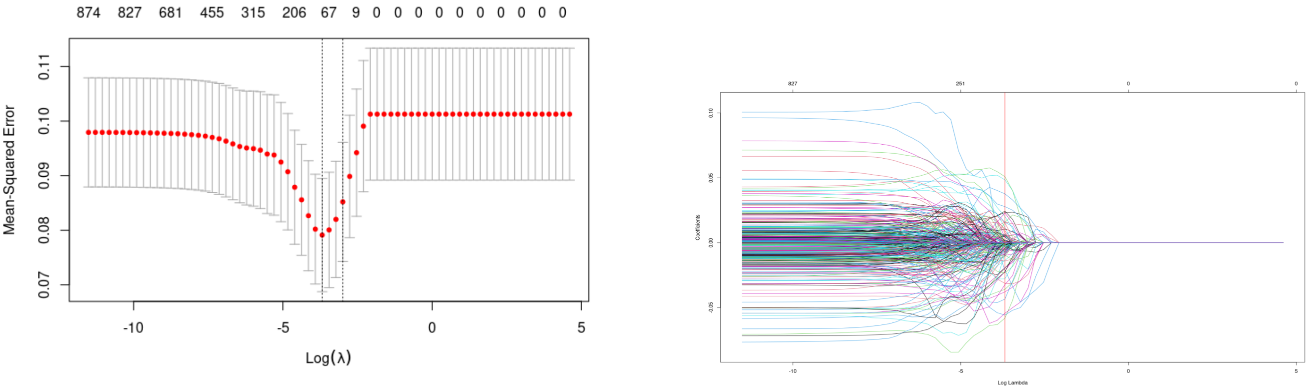
Figure 2. Violin and regression plots of 2 SNP showing their impact on the seed number per panicle

3. Result and Discussion

As suggested by the articles provided with the project, we have thought about implementing a **Lasso**-penalized regression to complete our classification model. We will then optimize our program (with a K-Fold procedure for example). Many reasons pushed us to do so :

- The l^1-penalization method induce "sparse" models, which is here quite helpful since we have a significant number of explanatory variables. Thus, it performs feature selection as well in a way;
- It executes quite fast in terms of fitting.

Thus, we implemented a basic Lasso regression in R, with the function *glmnet*. We provide a few results and plots to appreciate the performance of the algorithm, in Figures 3a and 3b.

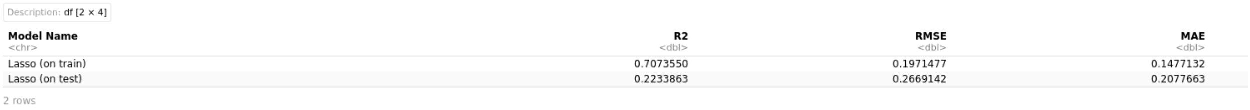


(a) MSE in function of log(λ)

(b) Regularization path of the Lasso

Figure 3. First results obtained with a Lasso regression

From these plots, we can evidence the lambda that optimizes the problem, ie which corresponds the smallest Mean Squared Error. The regularization path shows the good performance of the Lasso, which does a "soft-thresholding" on the coefficients. When λ grows, the coefficients are shrunked to zero. We will then discuss the metrics that can measure the performance of the Lasso in Figure 4. The values obtained on the train phase are satisfying, but the R^2 coefficient on the test phase can be optimized. The RMSE and MAE are slightly higher for the test phase, which is to look on afterwards. The predictive power of the model will be evaluated and presented during the presentation.



Description: df [2 × 4]

Model Name <chr>	R2 <dbl>	RMSE <dbl>	MAE <dbl>
Lasso (on train)	0.7073550	0.1971477	0.1477132
Lasso (on test)	0.2233863	0.2669142	0.2077663

2 rows

Figure 4. Value of the metrics, on the train phase and the test phase