Snp HeRitability Estimation Kit (SHREK) Development Log

Sam Shing Wan, Choi
choishingwan@gmail.com

Johnny Sheung Him, Kwan
shkwan@hku.hk

Henry Chi-Ming, Leung
cmleung2@cs.hku.hk

February 1, 2016

Contents

Assumption in implementation

To make life easier, I made a number of assumption when I implement the programme and they are listed as follow

1. The intervals within each individual bed files are non-overlapping
2. SNPs filtered out when reading the p-value file will not be included in the output no matter what (avoid storing unnecessary information)
3. We will remove *all* SNPs that are in perfect LD with each-other

(a) Realistically, we can not remove *all* SNPs that are in perfect LD with each-other without first computing the LD matrix for the whole genome. However, that will incur a large speed penalty to the algorithm. To compromise, we remove SNPs that are in perfect LD with each-other *within the same window*. Take into consideration of the calculation of variance, we will need to plan forward and consider SNPs within 4 windows. That's definitely overkill but that avoid problem of crazy perfect LD interactions (to be honest, I would consider those as error, yet it is easier for me to just remove them)

4. We do not consider a SNP ambiguous unless we need to perform flipping

January 25, 2016

We are going to re-write SHREK starting from today. There are a number of main goal in this re-work

1. The whole parameter parsing
2. Better help messages
3. Allow filtering by imputation score (0-1)
4. Filtering of SNPs that have different reference/alternative allele with reference
5. Use Armadillo instead of EIGEN to improve speed
6. Filter *all* SNPs that are in perfect LD
7. Implement the advance variance calculation
8. Start documenting the codes

0.1 Finished today

1. Finished the error message for different modes
2. Finished the parsing for binary trait

3. Finished the parsing for quantitative trait
4. Disabled risk prediction, should focus on the heritability estimation first

I have not perform debugging, there can be error

January 26, 2016

Continue on where I left yesterday Assumptions we made in the programme

1. p-value file has header
2. the numbers return from Command class are index (0 based)

I really want to write a more compacted code. Will try to restructure the command class

0.2 Finished today

1. Finished refining the usage messages
2. Finished the basic parameter parsing (Still haven't finish the index based parameter though)

January 27, 2016

0.3 Aim

We want to at least finish the region parsing and SNP class update today

0.4 Finished today

1. Completed the update of the Command class (compiled without problem)
2. Completed the update of the Region class (compiled without problem)
3. Completed the update of the Snp class (compiled without problem)

I also *haven't* test run the programme, so there might be additional syntax / logic error that are not picked up at the current stage.

January 28, 2016

0.5 Finished today

1. Finished the checking of reference panel
2. Found a different algorithm for popcount, which should be platform independent
3. Installed armadillo and OpenBLAS
   (a) I have to add the OpenBLAS to the PATH and LD_LIBRARY_PATH for armadillo to detect it
4. Updated the makefile accordingly
   (a) However, I have not updated the decomposition class, therefore there will be problem when we try to compile the programme
5. Updated the r^2 calculation function such that it is using the correct sample size
   (a) The sample size when calculating the r^2 is defined as the number of samples containing *both* SNPs
   (b) Because of how complicated the genotype class is, I really don't want to modify it
   (c) The main bottleneck for the computation is in the decomposition anyway, so it should be fine

January 29, 2016

Need to at least finish the getSNP function. Must be careful with it as this is not only complicated, but also extremely important for the whole programme.

I spent whole day in annotating the document of the class to make sure I don't overlook any problem.

February 1, 2016

Start to implement the getSNP function. A number of point to note

1. First get all the SNPs that passed all filtering (e.g. MAF)
2. Then for each window, calculate the LD
3. When encountered with perfect LD, we retain the *first* SNP
   (a) The summary statistics will be the mean of all SNPs in perfect LD
   (b) All SNPs except the first are *removed*
   (c) So they will not appear in subsequent analysis