XPA Manual

Shunkang Zhang

June 20, 2020

Contents

1 Overview **2**

2 Download and compile **2**
2.1 Download . . . 2
2.2 Compile . . . 2
2.3 Running XPA . . . 2
2.4 Help . . . 2

3 Computation requirements **3**
3.1 Operating system . . . 3
3.2 Memory . . . 3
3.3 Mutithreading . . . 3

4 XPA interface **3**
4.1 Genotypes . . . 3
4.2 Phenotypes . . . 4
4.3 Covariates . . . 4
4.4 Genotype QC . . . 4
4.5 Mom estimator . . . 4
4.6 Conjugate gradient . . . 4
4.7 Prediction with single dataset . . . 5
4.8 Cross population analysis . . . 5

5 Contact Info **5**

6 License **5**

7 Acknowledge **5**

1 Overview

XPA aims to analysis large scale genetic cross-population analysis. In this software, you can compute the variance components of the given dataset and do cross-population analysis based on the given phenotype.

2 Download and compile

2.1 Download

You can directly download the latest version of XPA software from the Github project XPA. Now we only release the initial version and we keep updating it.

2.2 Compile

If you want to compile the XPA software by your own, you can reference to my CMakeList.txt file. Please note that you should modify the path of Intel Math Kernel Library according to your own pc.

- Library dependencies:
  - Intel Math Kernel Library. In order to improve the speed of basic linear algebra involved in the XPA software, we use the Intel Math Kernel Library.
  - Boost C++ libraries. XPA uses the Boost program_options libraries to deal with the input arguments.

2.3 Running XPA

To run the XPA executable, you only need to invoke ./XPA on the Linux command line with the required parameters in the format --option optionValue. There is a running example in the following to show how to run XPA linear mixed model and cross population analysis.

- A toy example for XPA cross population analysis
  - ```
    ./XPA --bfile genotype --phenoFile phenotype
    --phenoCol diabetes --covarFile cov.txt --auxbfile auxgenotype
    --auxphenoFile auxphenotype --auxphenoCol diabetes
    --auxcovarFile auxcov.txt--predbfile predgenotype
    --precovarFile precov.txt --numThreads 8  --geneticCorr true
    --outputFile ./result
    ```

2.4 Help

To get the details description of different arguments, you can simply run:

```
./XPA -h
```
```

3 Computation requirements

3.1 Operating system

Conveniently, you can use the object file we provided. We only test our model on Linux computing environments; however, the source code and CMakeLists file are available and you can compile from the source by yourself. Please note that you should link the Intel Math Kernel Library correctly by the instructions in the official website.

3.2 Memory

For typical datasets (M, N exceeding 10,000, where M is the number of SNPs and N is the number of individuals), XPA uses about MN/4 bytes memory to store the raw genotypes. Moreover, it also needs additional memory to store the temporary result, for example the decoded SNPs vector matrix, the batch conjugate gradient result and so on.

The default max model SNPs is equal to one million and you can reset this default value by using command --maxModelSnps. Please note that in the case of cross population analysis, you probably should read three different genotype dataset. You should make sure that you can store all of them in you RAM. The maxModelSnps option only limits the max SNPs for each of the three, not the sum. Given the limitation of RAM, we split the whole matrix into many small matrix to do subsequent computation. You can specify the columns in one block by --snpsPerBlock. In default, we set the snpsPerBlock as 64 and we do not recommend you to set this value too large. Because it will cause the unnecessary cache page fault.

3.3 Mutithreading

We recommend you to execute the program on multi-core machines which can significantly reduce the program execution time. Please note that if you run with Intel HT Technology enabled, performance may be especially impacted if you run on fewer threads than physical cores. You can specify the number of threads used in model by using command --numThreads. When you deal with small datasets, we recommend you use fewer threads in case of threads conflict.

4 XPA interface

4.1 Genotypes

The XPA project takes genotype input in PLINK binary format(bed/bim/fam). If all genotypes are contained in a single bed/bim/fam file with the same prefix, you can simply used the command --bfile=prefix to pass the path of the input file. If your PLINK binary format files do not process the same prefix, you have to use the command --bed, --bim, --fam. Moreover, if there are several separate binary files, you can use the template input format (--bim=data_chr{1:22}). You can also specify the individuals and SNPs that you want to remove from the model by command --removeIndiv and --removeSnps.

In the case of missing genotype, we provide two ways to impute it – mean value imputation and zero imputation. You can specify it by using the option --imputeMethod. The default imputation method is mean value.

4.2 Phenotypes

There are two ways to input phenotypes as the following:

- --phenoUseFam: This option can use the 6th column in the fam file as the phenotypes and please note that the missing value of the phenotype should be set as -9.
- --phenoFile: This option reads the phenotype file from the disk and the first line should contain column headers and subsequent should contain records. Any number of columns may follow; the column containing the phenotype to analyze is specified with --phenoCol. Please note that the first two column must be FID and IID and value of -9 is interpreted as missing value.

In our setting, we only analysis one kind of phenotype data a time. If you provide more than one kind of phenotype, the program throws an error and exits. (In the future version, we will provide the function to deal with multiple phenotypes). Please note that if you run the XPA software to do cross population analysis, you should provide either all three datasets by -- bfile or provide them separately by -- bed, -- bim and -- fam.

4.3 Covariates

Covariate data may be specified in a file --covarFile with the same format as the phenotype file descried above. XPA default read the whole file as the covariate matrix and you can also specify the several columns by using the template input as the genotypes --covarCol pc{1:5}.

4.4 Genotype QC

The XPA project automatically filters SNPs and individuals with missing rate exceeding threshold of 0.1. You can also change the default max missing rate by using the command --maxMissingPerSnp and --maxMissingPerIndiv. If the individuals failed to pass the QC, we will automatically mask the covariate matrix accordingly and then it will exclude in the future analysis.

4.5 Mom estimator

There are two parameters related with Mom estimator. One is the iteration we used to estimate the trace. The default iteration is 10 and based on many experiments with different size of genotype datasets, it is enough to get a relatively accurate estimation by using 10 iteration when the number of individuals exceeds 10,000. If you run XPA based on small dataset, you can increase the default iteration by the command --estIteration. The other one is whether compute the exact trace or not. The Mom estimator can compute the trace approximation in O(MNB) where B is the estimation iteration. The process of computing the exact trace might take at least five times longer than mom estimator, which might provide a little bit more accurate result.

4.6 Conjugate gradient

The max iteration of the conjugate gradient method involved in the XPA is 100 and you can change the default setting by command --maxIterationConj. Besides, you can change the

convergence level of the conjugate gradient method by command --convergenceLevel. The default value is $1e^{-5}$.

4.7 Prediction with single dataset

If you just want to make prediction based on single population training dataset, you can specify the --prediction flag. In this case, you will fit parameters based on single training dataset you provide and directly make prediction based on another dataset. Please note that in this case, you should provide the prediction genotype data by --predbfile and prediction covariate file by --predcovarFile. The input format can refer to the 4.1.

4.8 Cross population analysis

You can specify the --geneticCorr to make prediction by using cross-population model. You should provide two genotype datasets. The one is the main training dataset and the other one is the auxiliary training dataset by --auxbfile. Besides, you should also provide the auxiliary covariate file and auxiliary phenotype file by --auxcovarfile and --auxphenoFile. You can also specify the selected columns in covariate file and phenotype file by --auxcovarCol and --auxphenoCol. The software will automatically match the common SNPs and flip the minor allele. What's more, you also should provide the flag as the prediction section claims.

5 Contact Info

If you have comments or questions about the XPA software, please contact Professor Can Yang, macyang@ust.hk and Shunkang Zhang, szhangcj@connect.ust.hk. Welcome more suggestions to further improve the software.

6 License

XPA is free software under the GNU General Public License v3.0 (GPLv3).

7 Acknowledge

This software partially refers to BOLT-LMM software which provides the solid baseline for our implementation. This research is supported by Professor Can Yang at The Hong Kong University of Science and Technology and Professor Xiang Wan at Shenzhen Research Institute of Big Data.