# Family-based association tests (BURDEN and SKAT) for sequence data

Programming language

The FB-SKAT program is written in C++ language, the program was developed and tested in LINUX environment

Compilation:

Before executing, the program needs to be compiled with standard g++ compiler. The GSL - GNU Scientific Library is required. It can be freely downloaded from http://www.gnu.org/software/gsl/

In our Linux environment the program compiles with the command:

$ g++ $(gsl-config --cflags) -o FB-SKAT\_v2.1 FB-SKAT\_v2.1.cpp $(gsl-config --libs)

Compiled program for LINUX is also provided but the GSL library must be on your PATH

## Input Files

The program accepts the following input files:

**pedigreeFile** - is a standard pedigree file. The first 6 columns are:

* Family ID Individual ID
* Father ID
* Mother ID
* Gender - 1 for male/ 2 for female
* Phenotype - 1 for case/0 for control; or continuous trait

The remaining columns encode the genotypes at individual SNV positions. Each genotype is coded in an additive fashion: 0, 1 or 2 (i.e. the number of minor alleles).

Please note that family members in a trio need to be in order child, parent1, parent2.

**genesFile** - contains information about the grouping of individual SNVs into units to be tested together, such as genes. Each line in this file corresponds to one gene and contains the name of the gene, the number of the first SNV in the gene, and the number of the last SNV in the gene. These start and end positions correspond to the ordering of SNVs in the pedigree file.

**variantPassFile** - contains information about the quality of individual SNVs, one value per variant (therefore the total number of values in the file is the number of SNVs in the pedigree file). If a variant is not PASS, the value is 0. Otherwise it is > 0.

**weightsFile** - information about the weights of individual SNVs, one weight per variant (therefore the total number of weights in the file is the number of SNVs in the pedigree file). In the simplest scenario, when all variants are tested each weight is 1. If only non-synonymous variants are tested then weights are 1 for non-synonymous variants and 0 for synonymous variants.

## The output files

**resultFile** – the result file contains information that is needed by the R code to generate the P-values (see Getting the P-values section). As different tests/parameters may be submitted at a time, the parameter values will be appended to the end of the result and mendelian error file names. For example, if you submit

*resultFile=result\_,* it will become “***result\_0.05\_0.01\_0.txt***” to specify ***MAF***, ***mend\_th***, and ***ro*** parameters listed below. The same naming convention holds for the mendelian error file name.

**mendelianErrorsFile** – log file of mendelian errors occurring in trios

Other parameters:

Also, the following parameters may be changed as desired.

**npermut** - number of random permutations (default 10000)

**MAF** - maximum MAF for the variants to be tested

**mend\_th** - maximum mendelian error rate at a marker that is allowed; otherwise marker is ignored

**ro** - specifies which test to run: ro=0 SKAT; ro=1 BURDEN.

**min\_mark** – only genes with this minimum number of passing markers are analyzed (e.g. 3).

## Running C++ Program

To run the program, submit the following command at your LINUX machine:

./FB-SKAT\_v2.1 ${pedigreeFile} ${variantPassFile} ${genesFile} ${weightsFile} ${resultFile} ${mendelianErrorsFile} ${npermut} ${MAF} ${mend\_th} $ro ${min\_mark}

We provide example of the wrapper shell ***FB-SKAT\_v2.1.sh***. Please modify according to your needs.

# Getting the P-values

The P-values for each individual gene can be obtained using R and the output file, as follows.

x=read.table('results.txt');

y1=1-pchisq(x[,4],x[,5]);

y2=1-pchisq(x[,6],x[,7]);

The two vectors y1 and y2 contain the P-values for the genes in the first column of x (file 'results.txt') { there are two different ways to compute the P-values; hence y1 and y2}