USER MANUAL OF FIPSA v0.1

NAME

FIPSA – Fine population structure analysis using genotypes of large number of individuals

SYNOPSIS

FIPSA -K num_sub_pop -i inp.vcf -o vcf.out
FIPSA -K num_sub_pop -a inp.ped -b inp.map -o ped.out

DESCRIPTION

FIPSA is an individual-based non-parametric method for exploring the fine population structure using genotypes of large number of individuals. Taking genotypes of SNVs across individuals as input, FIPSA outputs the grouping of these individuals under given number of groups. FIPSA is able to work on datasets with as many as 10e10 SNVs*individuals. For example, 100K individuals at 100K SNVs.

The core function "FIPSA" is written in C, and it calculates the best grouping under given number of groups (K). It is recommended for users to run **replicates** for the same K, as some of the replications might not reach global optima. We suggest running 20 times for the same K and taking the grouping with the highest score (LR value) as the best grouping.

We provide a statically built FIPSA on a 64-bit RedHatLinux system (FIPSA_RedHatLinux), and a statically built FIPSA on a 64-bit Ubuntu 12.04 system (FIPSA_Ubuntu64_12.04). Both files should work well on any 64-bit Linux system. Therefore, no compilation is required. Please report any bug to xuedongpan.tz@gmail.com.

Commands and Options

Input is vcf format

FIPSA -K num_sub_pop -i inp.vcf -o vcf.out

Input is ped format (there should be a map file accompanying the ped file)

FIPSA -K num_sub_pop -a inp.ped -b inp.map -o ped.out

INPUT

-i

VCF format as input

The standard VCF format.

Here is how the file would be processed in detail:

1) Lines starting with "##" are ignored.

2) The first line starting with "#" (but not "##") is regarded as the line containing column names. Individual ids are extract from this line.
3) Any line after the column name line is regarded as line of genotypes for one locus.

-a

Ped format as input

Should be used together with –b inp.map. Should not use –i.

The standard ped format. Please refer to the manual of the software plink (http://zzz.bwh.harvard.edu/plink/index.shtml) for details.

-b

Map format accompanying the ped file

Should also specify –a inp.ped. Should not use –i.

The standard map format. Please refer to the manual of the software plink (http://zzz.bwh.harvard.edu/plink/index.shtml) for details.

-K

Number of subpopulations

The number of subpopulations (K) should be specified. It is recommended that users specify K based on biological knowledge. The default value of K is 2.

Optional parameter

-M

Number of group assignment updates per individual during the current replicate

This value specifies the duration of the current replicate. This value is proportional to the time consumption of the current replicate. Default value is 100, which means each individual's group assignment will be updated 100 times during the current replicate.

OUTPUT

Grouping file

The output file contains two columns. The first column is the individual id. The second column is the population group index.

1) The first line contains summary statistics from the run. The first value (e.g. 33850/41351) refers to the iteration where the maximum LR is reached. The second value (e.g. 195603.7) is the maximum LR value. The maximum LR is an important value. When we run replicates, the maximum LR values may differ. Replicate with the highest LR corresponds to the best grouping.
2) Starting from the second line, each row represents the population grouping of one individual. The first value (e.g. HG00096) is the id of the individual. The second value (e.g. 1) is the grouping index of this individual. Individuals with the same grouping index are in the same group. If we specified K subpopulations as input, then the grouping index should be

0,1,...,K-1.

Note: it is highly recommended to run 20 replications per K, because it is always possible for any replicate to be stuck in the local optima. To save time, users could submit each replicate to one core, and run the replicates in parallel.

EXAMPLE

Below is an example showing how to use FIPSA on a linux system.

```
#!/bin/bash

cwd=`pwd`

cd $cwd

# Multiple replicates are necessary. Replicates are independent of each other. Therefore, they could be submitted to different cores.
# For the test dataset, one replicate takes less than one minute. If we run 20 replicates on 20 cores respectively, then we get 20 results within one minute.

# Here is the example of running replicate 1 for K=5
ped=$cwd/inp_test_1KG_P3.ped   # test ped file
map=$cwd/inp_test_1KG_P3.map # test map file
K=5
rep=1
# using ped and map as input
$cwd/ FIPSA_RedHatLinux -K ${K} -a $ped -b $map -o $cwd/ped.rep${rep}.out

vcf=$cwd/inp_test_1KG_P3.vcf
rep=1
# using vcf as input
$cwd/ FIPSA_RedHatLinux -K ${K} -i $vcf -o $cwd/vcf.rep${rep}.out
```

Illustration of one output

33850/41351	195603.7
HG00096	3
HG00097	3
HG00099	3
HG00100	3
HG00101	3
HG00102	3
HG00103	3
HG00105	3
HG00106	3

The first line contains summary statistics from the run. “33850/41351” corresponds to the number of iteration at which LR reaches the maximum. “195603.7” is the maximum LR value. This value is very important (shown in red). For different replicates, this value may differ. Replicate with the highest maximum LR gives the best grouping.
The first column is the individual id. The second column is the grouping index. If we specified K subpopulations, then the grouping index should be 0,1,...,K-1.
In this example, the individual "HG00096" belongs to group "3", so do "HG00097", "HG00099", "HG00100", "HG00101"…

Then we get the outputs from different replications. The best grouping is the one with the highest maximum LR value which locates in the first row second column of each output file (shown in red). That means we should select the replicate with the highest value in the first row second column of the output.