MERLIN Tutorial

These pages provide a guided tour through the main features and quirks of MERLIN. The section on input file formats is recommended for all users. The other sections will depend on the focus of your research. Merlin can be used for parametric and non−parametric linkage analysis, regression−based linkage analysis or association analysis for quantitative traits, ibd and kinship estimation, haplotyping, error detection and simulation. Linkage disequilibrium between markers can be accomodated in most analyses.

Enjoy!

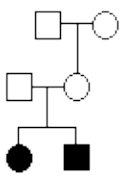
MERLIN Input Files

MERLIN performs common pedigree analyses. Input files describe relationships between individuals in your dataset, store marker genotypes, disease status and quantitative traits and provide information on marker locations and allele frequencies.

MERLIN supports input files in either QTDT or LINKAGE format. Although the two formats are very similar, in the discussion below we will focus on QTDT format.

Describing Relationships Between Individuals

Although pedigrees can become quite complex, all the information that is necessary to reconstruct individual relationships in a pedigree file can be summarized in five items: a family identifier, an individual identifier, a link to each parent (if available) and finally an indicator of each individual's sex.



As an example of how family relationships are described, we will construct a *pedigree file* for a small pedigree with two siblings, their parents and maternal grand−parents.

For this simple pedigree, the five key items take the following values:

FAMILY	PERSON	FATHER	MOTHER	SEX
example	granpa	unknown	unknown	m
example	granny	unknown	unknown	f
example	father	unknown	unknown	m
example	mother	granpa	granny	f
example	sister	father	mother	f
example	brother	father	mother	m

These key values constitute the first five columns of any pedigree file. Because of restrictions in early genetic programs, text identifiers are usually replaced by unique numeric values. After replacing each identifier with unique integer and recoding sexes as 2 (female) and 1 (male), this is what a basic space−delimited pedigree file would look like:

```
<contents of basic.ped>
1   1   0  0  1
1   2   0  0  2
1   3   0  0  1
1   4   1  2  2
1   5   3  4  2
1   6   3  4  1
<end of basic.ped>
```

A pedigree file can include multiple families. Each family can have a unique structure, independent of other families in the dataset.

Describing Phenotypes and Genotypes

Usually the five standard columns are followed by various types of genetic data, including phenotypes for discrete and quantitative traits and marker genotypes.

Disease status is usually encoded in a single column as

U or **1** for unaffecteds,
A or **2** for affecteds, and
X or **0** for missing phenotypes.

Quantitative traits are encoded as numeric values with **X** denoting missing values (it is also possible to use a peculiar numeric value to flag missing phenotypes, but the procedure is prone to error and not recommended).

Marker genotypes are encoded as two consecutive integers, one for each allele, optionally separated by a "/". To denote missing alleles, either a 0 or an X can be used. The following are all valid genotype entries *1/1* (homozygote for allele 1), *0/0* (missing genotype), and *3 4* (heterozygote for alleles 3 and 4). For the X chromosome, males should be encoded as if they had two identical alleles.

This is what the previous pedigree file might look like after adding a column for disease status, measurements for a quantitative trait and genotypes for two markers:

```
<contents of basic2.ped>
1   1   0  0  1   1      x   3 3   x x
1   2   0  0  2   1      x   4 4   x x
1   3   0  0  1   1      x   1 2   x x
1   4   1  2  2   1      x   4 3   x x
1   5   3  4  2   2  1.234   1 3   2 2
1   6   3  4  1   2  4.321   2 4   2 2
<end of basic2.ped>
```

Notice that the two siblings (individuals 5 and 6 in the last two rows) are marked as affected (value 2 in the sixth column), everyone else is marked as unaffected (value 1 in the sixth column). The quantitative trait (seventh column) takes values 1.234 and 4.321 for each sibling. Whereas everyone is genotyped at the first marker, for the second marker, only individuals 5 and 6 are genotyped.

Describing the pedigree file

Pedigree files can include any number of marker genotype, disease status and quantitative trait variables, limited only by available memory. Since each pedigree file has a unique structure (apart from the first five columns), its contents must be described in a companion *data file*.

The data file includes one row per data item in the pedigree file, indicating the data type (encoded as M − marker, A − affection status, T − Quantitative Trait and C − Covariate) and providing a one−word label for each item. A data file for the pedigree above, which has one affection status, followed by one quantitative trait and two marker genotypes might read:

```
<contents of basic2.dat>
A  some_disease
T  some_trait
M  some_marker
M  another_marker
<end of basic2.dat>
```

You can get a summary description of any pair of pedigree and data files using pedstats (included in the MERLIN distribution). To run pedstats you must provide the name of your data file (**−d** command line option) and pedigree file (**−p** command line option). In the MERLIN examples directory, try the following command:

```
prompt>pedstats −d basic2.dat −p basic2.ped
```

Genetic Maps

To analyse genetic markers, MERLIN requires information on their chromosomal location. This is usually provided in a *map file*. If you are using sex−average maps, this file has one line per marker with three columns, indicating chromosome, marker name and position (in centiMorgans). If you are using sex−specific maps, you will need two additional columns specifying the marker position along the female and male genetic maps, respectively.

The data file and map file can include different sets of markers, but markers that are absent from the map file will be ignored by MERLIN. Here is what a typical map file looks like:

```
<contents of basic2.map>
CHROMOSOME   MARKER          POSITION
24           some_marker     123.4
24           another_marker  136.2
<end of basic2.map>
```

And here is a refined version of the map file including sex−specific map positions for each marker:

```
<contents of file with sex−specific map>
CHROMOSOME   MARKER          POSITION    FEMALE_POSITION   MALE_POSITION
24           some_marker     123.4       146.8             100.0
24           another_marker  136.2       166.4             103.0
<end of sex−specific map>
```

Using separate data and map files makes for a very simple file structure and allows MERLIN to analyse multiple chromosomes in a single run.

Allele Frequency Files

LINKAGE format data files specify the number of alleles at each locus and their frequencies. When using QTDT format input files, MERLIN estimates allele frequencies by counting alleles across all individuals. If this is inappropriate for the analysis at hand you can request maximum likelihood allele frequency estimates (**−fm** command line option), specify equal allele frequencies (**−fe**), request estimates derived by counting among founders only (**−ff**) or provide a custom allele frequency file (**−f** *filename* option).

A custom allele frequency file indicates allele frequencies for all marker alleles at each marker. For each marker, a single header line naming the marker is followed by a list of allele frequencies, which can take multiple lines.

Each header line is labelled M and includes the marker name. This header is followed by a list of allele frequencies. There are two alternative formats for lines in the allele frequency list:

Classic format

Lines in the allele frequency list are labelled F and list frequencies for all alleles consecutively, starting with allele 1. This format is convenient for markers with a small number of alleles.

Extended format

Lines in the allele frequency list are labelled A and consist of a numeric allele label followed by an allele frequency. Alleles that are not specifically listed are assumed to have frequency zero.

Classic Allele Frequency Format

For example, if some_marker has four alleles with frequencies 0.1, 0.2, 0.3 and 0.4 respectively and another_marker has two alleles with frequencies 0.6 and 0.4 this is what the file might look like:

```
<contents of basic2.freq>
M some_marker
F 0.1 0.2 0.3 0.4
M another_marker
F 0.6 0.4
<end of basic2.freq>
```

An equivalent layout for the same information is:

```
<contents of basic2.freq>
M some_marker
F 0.1
F 0.2
F 0.3
F 0.4
M another_marker
F 0.6
F 0.4
<end of basic2.freq>
```

Extended allele frequency format

This format is recommended for microsatellites and other markers with large allele numbers. For example, if you are analysing a microsatellite marker with alleles of size 152, 154 and 156 base−pairs and their respective frequencies are 0.5, 0.4 and 0.1 your frequency file might read:

```
<contents of allele frequency file>
M some_microsatellite
A 152 0.5
A 154 0.4
A 156 0.1
<end of allele frequency file>
```

Well that is all you need to know about file formats to get started! You can proceed to linkage analysis, ibd and kinship estimation, haplotyping, error detection or simulation. Have fun!

MERLIN Tutorial −− Linkage Analysis

Linkage analysis tests for co−segregation of a chromosomal region and a trait of interest. In this section, we will walk through a basic non−parametric and variance components linkage analysis using MERLIN.

For this example, we will use a simulated data set that you will find in the examples subdirectory of the MERLIN distribution or in the download page.

The dataset consists of a simulated 5−cM scan of chromosome 24 in 200 affected sib−pair families and is organized into 3 files, a data file (*asp.dat*), a pedigree file (*asp.ped*) and a map file (*asp.map*). An overview of MERLIN input files is available elsewhere.

The recommended first step in any analysis is to verify that input files are being interpreted correctly. So let's start by running pedstats... Pedstats requires an input data file (**−d** parameter) and pedigree file (**−p** parameter):

```
prompt>pedstats −d asp.dat −p asp.ped
```

By examining the abbreviated pedstats output below, you should be able to confirm that there are 200 pedigrees, each with 4 individuals (two affected siblings and their parents). Among phenotyped individuals, the prevalence of the disease is 100% (there are no unaffecteds in the sample) and the pedigree also includes a quantitative trait. In addition there are no phenotyped or genotyped founders.

```
Pedigree Statistics
(c) 1999−2001 Goncalo Abecasis

The following parameters are in effect:
            QTDT Pedigree File :         asp.ped (−pname)
                QTDT Data File :         asp.dat (−dname)
            Missing Value Code :         −99.999 (−xname)

PEDIGREE STRUCTURE
==================
          Individuals: 800 (400 founders, 400 nonfounders)
             Families: 200
 Average Family Sizes: 4.00
  Average Generations: 2.00

QUANTITATIVE TRAIT STATISTICS
============================
                    [Phenotypes]      [Founders]       Mean        Var
          trait      400  50.0%        0   0.0%      0.021      1.496

AFFECTION STATISTICS
====================
                   [Diagnostics]      [Founders] Prevalence
      affection      400  50.0%        0   0.0%     100.0%
          Total      400  50.0%        0   0.0%

MARKER GENOTYPE STATISTICS
==========================
                     [Genotypes]      [Founders]     Hetero
           MRK1      400  50.0%        0   0.0%      72.8%
           MRK2      400  50.0%        0   0.0%      73.2%
        (...statistics for other markers would appear here...)
          Total     8000  50.0%        0   0.0%      74.1%
```

Everything checks out, so let's run merlin! We will need to specify an input data file (**−d** parameter), pedigree file (**−p** parameter) and map file (**−m** parameter). In addition, we need to request a non−parametric linkage analysis. In this case, we will request calculation of both the Whittemore and Halpern NPL pairs (**−−pairs**) and NPL all (**−−npl**) statistics:

```
prompt> merlin −d asp.dat −p asp.ped −m asp.map −−pairs −−npl
```

After running the command, you should first see the MERLIN banner and a summary of currently selected options:

```
MERLIN 0.8.4 − (c) 2000−2001 Goncalo Abecasis

The following parameters are in effect:
                    Data File :         asp.dat (−dname)
                Pedigree File :         asp.ped (−pname)
           Missing Value Code :         −99.999 (−xname)
                     Map File :         asp.map (−mname)
           Allele Frequencies : ALL INDIVIDUALS (−f[a|e|f|file])
           Steps Per Interval :               0 (−i9999)
                  Random Seed :          123456 (−r9999)

Data Analysis Options
         General : −−error, −−ibd, −−kinship, −−information
         Linkage : −−npl [ON], −−pairs [ON], −−qtl, −−deviates, −−vc
     Haplotyping : −−best, −−sample, −−all, −−founders
   Recombination : −−zero, −−one, −−two, −−three, −−singlepoint
          Limits : −−bits [24], −−megabytes
          Output : −−quiet, −−markerNames
      Simulation : −−simulate, −−save
      Additional : −−simwalk2, −−matrices, −−swap
```

Notice that allele frequencies were estimated by counting among all individuals (the default). Alternatively, one could calculate allele frequencies among founders only (**−ff**), request equal allele frequencies (**−fe**) or use an allele frequency file with custom frequencies.

After a few moments, you should see analysis results at each location:

```
Phenotype: affection [ALL] (200 families)
============================================================
                Pos   Zmean  pvalue    delta    LOD  pvalue
                min  −20.00     1.0   −0.707 −60.21     1.0
                max   20.00 0.00000    0.707  60.21 0.00000
              0.000    0.96     0.2    0.092   0.27    0.13
              5.268    1.39    0.08    0.126   0.54    0.06
             10.536    1.27    0.10    0.110   0.43    0.08
             15.804    1.43    0.08    0.128   0.56    0.05
             21.072    0.88     0.2    0.083   0.22     0.2
             26.340    1.37    0.08    0.130   0.55    0.06
             31.608    1.53    0.06    0.151   0.71    0.04
             36.876    2.18   0.014    0.197   1.32   0.007
             42.144    2.60   0.005    0.218   1.75   0.002
             47.412    3.00  0.0014    0.251   2.33  0.0005
             52.680    3.43  0.0003    0.286   3.05 0.00009
          (... results continue at other locations...)
```

The first two lines indicate the maximum possible scores for this dataset. These are followed by analysis results at each location (cM position, Zscore, p−value assuming normal approximation, Kong and Cox delta, KCLOD score and KCp−value). You will notice that results are identical for the NPL all and pairs statistics −− this is always the case for families with a single affected sib−pair! Linkage peaks at location 52.68 with a

Zscore of 3.43 (assymptotic p−value of 0.0003), corresponding to a Kong and Cox LOD score of 3.05 with probability 0.00009.

Commonly used linkage analysis options include requesting output with marker names, instead of cM positions (**−−markerNames** option) and requesting analysis between markers (**−−steps** *n* for *n* steps per interval) or along a grid of equally spaced locations along the chromosome (**−−grid** *n* for an n−cM grid). Try them out! For example...

```
prompt>merlin −d asp.dat −p asp.ped −m asp.map −−steps 4 −−pairs −−markerNames
```

... would calculate the NPL pairs statistic at 4 locations between consecutive markers and use marker names in the output.

TIP: The standard non−parametric linkage analysis carried out by Merlin uses the Kong and Cox (1997) *linear model* to evaluate the evidence for linkage. This model is designed to identify small increases in allele sharing spread across a large number of families −− this is what one usually expects in a complex disease. If you are searching for a large increase in allele sharing in a small number of families, you can select the Kong and Cox (1997) *exponential model* by adding the **−−exp** option to your command line, after the **−−npl** or **−−pairs** options. This alternative model is more computationally intensive and requires more memory, but provides a better linkage test if you expect a large increase in allele sharing among affected individuals.

To carry out a variance components linkage analysis on the same data set, we will use the −−vc option. If you are using a peculiar value, such as 1234 or −99.999 to represent missing values in your data, remember to use the **−x** *peculiar_value* option to tell MERLIN about it in all quantitative trait analyses. In the asp pedigree, missing values have been replaced by *x*. Let's try a variance components analysis:

```
prompt>merlin −d asp.dat −p asp.ped −m asp.map −−vc
```

In the output, you will see the estimated sample heritability for each phenotype (in this case 86%) followed by estimates of the genetic effect and LOD scores at each marker location:

```
Phenotype: trait [VC] (200 families, h2 = 86.74%)
=================================================
            Position      H2    ChiSq     LOD  pvalue
               0.000   40.95%    5.21    1.13   0.011
               5.268   51.42%    9.88    2.15  0.0008
              10.536   56.26%   13.01    2.82  0.0002
              15.804   65.40%   19.63    4.26 0.00000
              21.072   60.89%   15.36    3.34 0.00004
            (... results continue at other locations...)
```

In this case, linkage peaks at position 15.8 cM. You could identify which families are contributing the most to these linkage signals using the −−perFamily option, which generates an additional file tabulating the contribution of each family to the overall LOD score (for non−parametric analysis this partial contribution will be labelled pLOD).

Since this is a selected sample, you might want to check out the simulation section to find out how to conduct gene−dropping simulations that could be used, for example, to estimate empirical p−values. Or proceed to the error detection (improves power!), haplotyping or ibd estimation sections.

MERLIN Tutorial −− Parametric Linkage Analysis

Linkage analysis tests for co−segregation of a chromosomal region and a trait locus of interest. In parametric linkage analysis, a specific disease model is used to describe segregation of the trait locus. In this section, we will walk through a parametric linkage analysis using MERLIN.

For this example, we will use a simulated data set that you will find in the examples subdirectory of the MERLIN distribution or in the download page.

The dataset consists of a 10−cM scan of candidate chromosome in a single pedigree where a rare dominant disorder is segregating (the pedigree is picture above). Ten microsatellite markers, each with 4 equally frequent alleles, were genotyped in all pedigree members. The genotypes and phenotypes are described in 3 files, a data file (*parametric.dat*), a pedigree file (*parametric.ped*) and a map file (*parametric.map*). An overview of MERLIN input files is available elsewhere.

The recommended first step in any analysis is to verify that input files are being interpreted correctly. So let's start by running pedstats... Pedstats requires an input data file (**−d** parameter) and pedigree file (**−p** parameter):

```
prompt>pedstats −d parametric.dat −p parametric.ped
```

By examining the abbreviated pedstats output below, you should be able to confirm that there is a single pedigree, with a total of 16 individuals (8 of these individuals are affected), and that there is no missing phenotype or genotype data.

```
Pedigree Statistics − 0.5.4
(c) 1999−2005 Goncalo Abecasis, 2002−2005 Jan Wigginton

The following parameters are in effect:
                Pedigree File :  parametric.ped (−pname)
                    Data File :  parametric.dat (−dname)

PEDIGREE STRUCTURE
==================

       Individuals: 16
          Founders: 5 founders, 11 nonfounders
            Gender: 6 females, 10 males
          Families: 1

       Generations
           Average: 3.00 (3 to 3)
      Distribution: 3 (100.0%), 0 (0.0%) and 1 (0.0%)

AFFECTION STATISTICS
====================

                    [Diagnostics]      [Founders] Prevalence
VERY_RARE_DISEASE       16 100.0%        5 100.0%      50.0%
            Total       16 100.0%        5 100.0%

MARKER GENOTYPE STATISTICS
==========================

                    [Genotypes]      [Founders]     Hetero
           MRK1       16 100.0%        5 100.0%      87.5%
           MRK2       16 100.0%        5 100.0%      75.0%
```

```
        (...statistics for other markers would appear here...)
          MRK10       16 100.0%        5 100.0%      75.0%
          Total      160 100.0%       50 100.0%      78.1%
```

The pedigree and data file seem to be okay. In addition to the standard Merlin input files, parametric linkage analyses require disease locus parameters to be specified in a separate text file. This text file has one row for each of the disease models to be evaluated, and can include as many different models as available memory allows. For this analysis, the file *parametric.model* specifies a single rare dominant disease model. Here are its contents:

Affection	Disease Allele Frequency	Penetrances	Model Name
VERY_RARE_DISEASE	0.0001	0.0001,1.0,1.0	Rare_Dominant

In general, the file should be tab or space delimited, with 4 fields: affection status label (matching the data file), disease allele frequency, probability of being affected for individuals with 0, 1 and 2 copies of the disease allele (penetrances), and finally a label for the analysis model. A header line is included in the table above, for readability, but is not required. This file can also specify penetrance functions that depend on a covariate, such as age.

Okay ... let's run merlin! We will need to specify an input data file (**−d** parameter), pedigree file (**−p** parameter) and map file (**−m** parameter) as well as the file with trait model parameters (**−−model** command line option). Since parametric linkage LOD scores tend to dip at marker locations, we will request an analyses at three equally spaced locations between each consecutive pair of markers with the **−−step 3** option. With all these options, the command line will look like this:

```
prompt> merlin −d parametric.dat −p parametric.ped −m parametric.map −−model parametric.model −−s
```

After running the command, you should first see the MERLIN banner and a summary of currently selected options:

```
MERLIN DEMO−VERSION − (c) 2000−2005 Goncalo Abecasis

The following parameters are in effect:
                    Data File :  parametric.dat (−dname)
                Pedigree File :  parametric.ped (−pname)
                     Map File :  parametric.map (−mname)
           Allele Frequencies : ALL INDIVIDUALS (−f[a|e|f|file])

Data Analysis Options
          General : −−information, −−likelihood, −−model [parametric.model]
        Positions : −−steps [3], −−maxStep, −−minStep, −−grid, −−start,
                    −−stop
```

Notice that allele frequencies were estimated by counting among all individuals (the default). In this case, this does not matter because all founders are genotyped. In practice, when analysing small datasets such as this one, it might be a good idea to genotype additional unrelated individuals to obtain better estimates of allele frequencies or to use an allele frequency file with custom frequencies.

After a minute or two, you should see analysis results at each location:

```
Parametric Analysis, Model Dominant_Model
==========================================================
       POSITION        LOD      ALPHA       HLOD
            (... some results edited to save space ...)
```

```
35.000     −1.291      0.000      0.000
37.500      2.037      1.000      2.037
40.000      2.263      1.000      2.263
42.500      2.358      1.000      2.358
45.000      2.388      1.000      2.388
47.500      2.201      1.000      2.201
50.000      1.959      1.000      1.959
52.500      1.585      1.000      1.585
55.000     −9.291      0.000      0.000
   (... results continue at other locations...)
```

Each row indicates the estimated multipoint LOD score at a particular location. This is followed by the estimate proportion of linked families (since there is only one informative family in this sample, the proportion will always be 0.000 or 1.000), and the corresponding maximum heterogeneity LOD score. In this case the maximum LOD score of 2.407 is observed at position 45.000, the position of marker MRK5 in the map file.

Useful options for parametric linkage analyses options include requesting output with marker names, instead of cM positions (**−−markerNames** option), requesting analysis along a grid of equally spaced locations (**−−grid** *n* for an n−cM grid) rather than at a fixed number of steps between markers (**−−steps** *n* for n−steps between consecutive markers), or requesting a graph summarizing results (**−−pdf**). Try them out! For example...

```
prompt> merlin −d parametric.dat −p parametric.ped −m parametric.map −−model parametric.model −−g
```

... would calculate the parametric LOD scores for a 1−cM grid along the chromosome and generate a PDF file with the resulting statistics.

That is it! That is all you need to get started with parametric linkage analysis in Merlin. Remember to set your disease model carefully, as an appropriate and careful choice of disease model is essential for parametric linkage analyses.

To learn about other analyses options, you might want to check the non−parametric linkage analysis section to find out how to conduct affecteds only linkage analyses. Or you could proceed to the error detection (improves power!), haplotyping, simulation or ibd estimation sections.

MERLIN Tutorial −− QTL Regression Analysis

Quantitative trait linkage analyses examine whether a chromosomal region is responsible for some of the variation in a trait of interest. Here, we will describe how fast quantitative trait regression analyses can be carried out using MERLIN.

Data for this exercise

For this example, we will use a simulated data set that you will find in the examples subdirectory of the MERLIN distribution or in the download page.

The dataset consists of a simulated 5−cM scan of chromosome 24 in 200 sib−pair families and is organized into 3 files, a data file (*asp.dat*), a pedigree file (*asp.ped*) and a map file (*asp.map*). A quantitative trait has been scored for each offspring.

The recommended first step in any analysis is to verify that input files are being interpreted correctly. So let's start by running pedstats... Pedstats requires an input data file (**−d** parameter) and pedigree file (**−p** parameter):

```
prompt>pedstats −d asp.dat −p asp.ped
```

By examining the abbreviated pedstats output below, you should be able to confirm that there are 200 pedigrees, each with 4 individuals (two siblings and their parents). The pedigree includes a quantitative trait that has been measured on all 400 offspring but none of the founders.

```
Pedigree Statistics
(c) 1999−2001 Goncalo Abecasis

The following parameters are in effect:
            QTDT Pedigree File :         asp.ped (−pname)
                QTDT Data File :         asp.dat (−dname)
            Missing Value Code :         −99.999 (−xname)

PEDIGREE STRUCTURE
==================
          Individuals: 800 (400 founders, 400 nonfounders)
             Families: 200
 Average Family Sizes: 4.00
  Average Generations: 2.00

QUANTITATIVE TRAIT STATISTICS
=============================
                    [Phenotypes]      [Founders]       Mean        Var
          trait      400  50.0%        0   0.0%      0.021      1.496

AFFECTION STATISTICS
====================
                   [Diagnostics]      [Founders] Prevalence
      affection      400  50.0%        0   0.0%     100.0%
          Total      400  50.0%        0   0.0%

MARKER GENOTYPE STATISTICS
==========================
                     [Genotypes]      [Founders]     Hetero
           MRK1      400  50.0%        0   0.0%      72.8%
           MRK2      400  50.0%        0   0.0%      73.2%
        (...statistics for other markers would appear here...)
          Total     8000  50.0%        0   0.0%      74.1%
```

The most popular method of quantitative trait linkage is the Haseman−Elston (1972) procedure where squared trait differences for sib−pairs are regressed on IBD allele−sharing. If a gene in the region being investigate influences trait levels, sib−pairs who share more alleles are expected to show similar phenotypes and, therefore, smaller squared trait differences.

Pedigree−Wide Regression Analysis

The flexibility of the method of Haseman and Elston has lead many authors to propose enhancements and extensions. Sham et al. (2002) have recently described a regression−based procedure for linkage analysis that uses trait−squared sums and differences to predict IBD sharing between any non−inbred relative pairs. This method is implemented in the MERLIN−REGRESS program, included in the merlin distribution. The method of Sham et al. can be applied to selected samples but requires specification of the trait distribution parameters in the general population.

Analysing a single trait

To run MERLIN−REGRESS, we will need to specify the input data(**−d** parameter), pedigree (**−p** parameter) and map (**−m** parameter) file names. In addition, we will need to specify the trait distribution parameters (−−mean, −−variance and −−heritability options). In this case, we will assume that the trait of interest has mean=0.0, variance=1.5 and heritability=80% in the general population:

```
prompt> merlin−regress −d asp.dat −p asp.ped −m asp.map −−mean 0.0 −−var 1.5 −−her 0.8
```

After running the command, you should first see the familiar MERLIN banner and a summary of currently selected options:

```
MERLIN 0.9.1 − (c) 2000−2002 Goncalo Abecasis

The following parameters are in effect:
                     Data File :         asp.dat (−dname)
                 Pedigree File :         asp.ped (−pname)
            Missing Value Code :         −99.999 (−xname)
                      Map File :         asp.map (−mname)
            Allele Frequencies : ALL INDIVIDUALS (−f[a|e|f|file])
                   Random Seed :          123456 (−r9999)

Regression Analysis Options
     Trait Model : −−mean [0.00], −−variance [1.50], −−heritability [0.80]
   Recombination : −−zero, −−one, −−two, −−three, −−singlepoint
       Positions : −−steps, −−maxStep, −−minStep, −−grid, −−start, −−stop
          Limits : −−bits [24], −−megabytes, −−minutes
          Output : −−quiet, −−markerNames
          Others : −−simulate, −−swap, −−rankFamilies

Estimating allele frequencies... [using all genotypes]
   MRK1 MRK2 MRK3 MRK4 MRK5 MRK6 MRK7 MRK8 MRK9 MRK10 MRK11 MRK12 MRK13 MRK14
   MRK15 MRK16 MRK17 MRK18 MRK19 MRK20
```

After a few moments, you should see analysis results at each location:

```
Pedigree−Wide Regression Analysis (Trait: trait)
================================================
       Position      H2   Stdev    Info     LOD  pvalue
          0.000   0.406   0.192   64.8%   0.970    0.02
          5.268   0.526   0.183   71.1%   1.792   0.002
         10.536   0.598   0.182   72.1%   2.343  0.0005
         15.804   0.733   0.182   72.1%   3.520 0.00003
```

```
         21.072   0.586   0.182   72.2%   2.255  0.0006
         26.340   0.596   0.190   66.0%   2.135  0.0009
         31.608   0.535   0.189   67.0%   1.744   0.002
         36.876   0.522   0.184   70.6%   1.752   0.002
         42.144   0.414   0.181   73.0%   1.137   0.011
         47.412   0.295   0.175   77.5%   0.614    0.05
(... results continue at other locations ...)
```

Successive columns indicate position along the chromosome (in CM), estimated locus specific heritability, standard deviation for the estimate of locus specific heritability, proportion of linkage information extracted at this location (100% information corresponds to the smallest possible confidence interval for estimated effect size), LOD score and corresponding p−value. In this case, linkage peaks at position 15.8 with an estimated locus specific heritability of 73.3% and a LOD score of 3.52 (probability 0.00003).

Estimating family informativeness

Another useful option in MERLIN−REGRESS is the ability to quantify the expected amount of linkage information in each family. This can be useful when focusing genotyping efforts (for example, by genotyping the most informative families first) or identifying problematic outliers (extreme outliers will lead to some families with very large weights which can reduce effective sample size in linkage analyses).

To estimate family informativeness, specify the trait distribution in the population (by specifying it's mean, variance and heritability) and use the −−rankFamilies option. Using the example input files the command line would read:

```
prompt> merlin−regress −d asp.dat −p asp.ped −−mean 0 −−var 1.5 −−her 0.8 −−rank
```

Running this command would produce the familiar MERLIN output screen followed by a table looking like the one below:

```
Family Informativeness
======================
         Family           Trait  People  Phenos   Pairs    Info  ELOD20
              1           trait       4       2       1   0.099   0.001
              2           trait       4       2       1   0.025   0.000
              3           trait       4       2       1   1.989   0.017
              4           trait       4       2       1   0.269   0.002
              5           trait       4       2       1   0.327   0.003
(... additional rows follow for other families)
```

Each row indicates the family and trait of interest, followed by number of individuals and phenotypes in each family, the number of phenotyped relative pairs and the relative informativeness of the family. The final column indicates the expected LOD score for a region with a locus specific heritability of 20% when a fully informative marker is typed. In this case family 3 seems particularly informative (you can try and find out why by examining the phenotypes for each individual in the *asp.ped* pedigree file).

Expected LOD scores are proportional to the squared locus specific heritability. To calculate expected LOD scores for a different effect size, simply multiply the expected LOD score by (*heritability*/20)^2, where H2 denotes your desired effect size and ^2 denotes the square operator. For example, for an effect size of 40%, you should multiply each expected LOD score by 4.

Comparing trait models and analysing multiple traits

Often multiple quantitative traits may be available in a particular dataset. Each of these traits is likely to have a distinct mean, variance and heritability in the population. The **−t** *models_file* specifies the name of a text file listing analysis models, one for each trait. Using a models table allows distinct models to be specified for each phenotype in the pedigree file.

A models table includes four columns. The first column indicates the trait name and is followed by columns indicating the trait mean, variance and heritability. Optionally, a fifth column can be included with a label for each model. Here is an example:

```
<sample regression models file>
TRAIT                MEAN          VARIANCE          HERITABILITY   LABEL
Weight_Kilograms     75            10                0.63           metric_analysis
Weight_Pounds        160           40                0.63           imperial_analysis
<end of sample regression models file>
```

Where to go next?

Now that you know how to carry out a pedigree−wide regression analysis using MERLIN you might want to find out estimate empirical p−values using simulation, or perhaps explore the sections on error detection, linkage analysis, haplotyping or ibd estimation.

MERLIN Tutorial −− Simulation

When interpreting results for pedigree analysis, it is extremely helpful to know how often a similar result might arise by chance. For example, in a linkage analysis it may be helpful to know how many peaks of similar height are expected conditional on the set of phenotypes being analysed and the available marker map. When investigating suspicious genotypes, it is important to characterize the false−positive rate for error detection procedures.

MERLIN has the ability to perform gene dropping simulations which replace input data with simulated chromosomes conditional on family structure and actual marker spacings and allele frequencies, as well as missing data patterns. The procedure for generating simulated data is described in the reference section.

For this example, we will use a data set from the examples subdirectory of the MERLIN distribution as input. You can also find the example data in the download page.

Estimating false positive rates for error detection

In the error detection tutorial, we identified 7 pairs of unlikely genotypes in a 20 marker, 5−cM scan, of 200 sib−pairs, corresponding to 8,000 total genotypes. The data is organized into three files, a pedigree file summarizing genotypes and relationships (*error.ped*), a data file describing the contents of the pedigree (*error.dat*) and map file providing marker locations (*error.map*).

To review a descriptive summary of the dataset, you could run pedstats:

```
prompt> pedstats −d error.dat −p error.ped
```

To review the original set of unlikely genotypes, you could use MERLIN's automated error analysis:

```
prompt> merlin −d error.dat −m error.map −p error.ped −−error
```

To estimate false positive rates, we will request that MERLIN analyse a simulated data set with identical allele frequencies and marker spacing by using the **−−simulate** command line option. Try it out!

```
prompt> merlin −d error.dat −m error.map −p error.ped −−error −−simulate
```

You should first see the MERLIN start−up screen and summary of selected options. Note that the options *−−error* and *−−simulate* are selected. Note also that the current random seed is 123456. This seed indicates which simulated replicate will be used, and selecting a different seed produces an alternative simulated data set.

```
MERLIN 0.8.4 − (c) 2000−2001 Goncalo Abecasis

The following parameters are in effect:
                     Data File :       error.dat (−dname)
                  Pedigree File :       error.ped (−pname)
            Missing Value Code :         −99.999 (−xname)
                      Map File :         asp.map (−mname)
            Allele Frequencies : ALL INDIVIDUALS (−f[a|e|f|file])
            Steps Per Interval :               0 (−i9999)
                   Random Seed :          123456 (−r9999)

Data Analysis Options
         General : −−error [ON], −−ibd, −−kinship, −−information
         Linkage : −−npl, −−pairs, −−qtl, −−deviates, −−vc
     Haplotyping : −−best, −−sample, −−all, −−founders
```

```
   Recombination : --zero, --one, --two, --three, --singlepoint
          Limits : --bits [24], --megabytes
          Output : --quiet, --markerNames
      Simulation : --simulate [ON], --save
      Additional : --simwalk2, --matrices, --swap
```

This start−up screen should be followed by an error detection analysis for the replicate, which should indicate a single pair of unlikely genotypes:

```
Family:    38 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK6 genotype for individual 3 is unlikely [0.021855]
  MRK6 genotype for individual 4 is unlikely [0.021855]
```

NOTE: *In many newer versions of MERLIN, you may not find any unlikely genotypes in the replicate produced with the default seed. This is not a problem, and merely reflects the low false positive rate of the procedure. Continue reading to learn about how to use a different seed...*

So MERLIN flags a single pair of unlikely genotypes in this particular replicate... Is this typical of other replicates? There are two ways to investigate the issue further.

One option is to generate additional replicates, one at a time, by repeating the above procedure with a different random seed. To do this, you will need to set the **−r** command line option. The following command repeats the previous analysis but sets the random seed to 1234, thus generating a different set of simulated data:

```
prompt> merlin −d error.dat −m error.map −p error.ped −−error −−simul −r 1234
```

Another option is to request that MERLIN loop through the simulation procedure multiple times. This option is available through the **−−reruns** command line option in newer versions of MERLIN. To analyse 20 simulated datasets, try:

```
prompt> merlin −d error.dat −m error.map −p error.ped −−error −−simul −−reruns 20
```

In either way, it is straight−forward to repeat any MERLIN analysis for simulated chromosomes and estimate false−positive rates for error detection or linkage analysis (note that MERLIN does not change input phenotypes and disease status when conducting simulations).

Now that you have seen how to generate simulated replicates, you could proceed to haplotype analysis or ibd estimation. If you haven't already done so, you could try the linkage or error detection tutorials.

MERLIN Tutorial −− Modeling Marker−Marker Linkage Disequilibrium

This tutorial describe the procedures and options available for modeling marker−marker linkage disequilibrium with MERLIN. It assumes that you are relatively familiar with MERLIN and its standard command line options. If you haven't yet done so, it is a good idea to first learn about input file formats and non−parametric linkage analysis.

MERLIN can accomodate marker−marker linkage disequilibrium in nearly all available analyses, including parametric and non−parametric analysis of discrete traits, regression and variance−components based analysis of quantitative traits, haplotyping analyses and simulation. Modeling marker−marker linkage disequilibrium is especially important when analysing SNP linkage maps in datasets where some parental genotypes are missing. It has been shown that in these settings ignoring marker−marker linkage disequilibrium can result in severe biases in linkage calculations.

To model linkage disequilibrium, MERLIN organizes markers into clusters. Each cluster can include both SNP and microsatellite markers. MERLIN then uses population haplotype frequencies to assume linkage disequilibrium within each cluster. Two limitations of the model are that it assumes no recombination within clusters and no linkage disequilibrium between clusters. These approximations appear to be reasonable in many datasets.

For this example, we will use a simulated data set that you will find in the examples subdirectory of the MERLIN distribution or in the download page.

The dataset consists of a SNP linkage scan of a candidate chromosome in a set of 500 affected sibships, each with three genotyped affected siblings and one affected parent. The SNP data consists of clusters of 2−3 SNPs, all within 100kb of each other, genotyped approximately 5cM apart along a single chromosome (20 clusters and 59 SNPs in total).

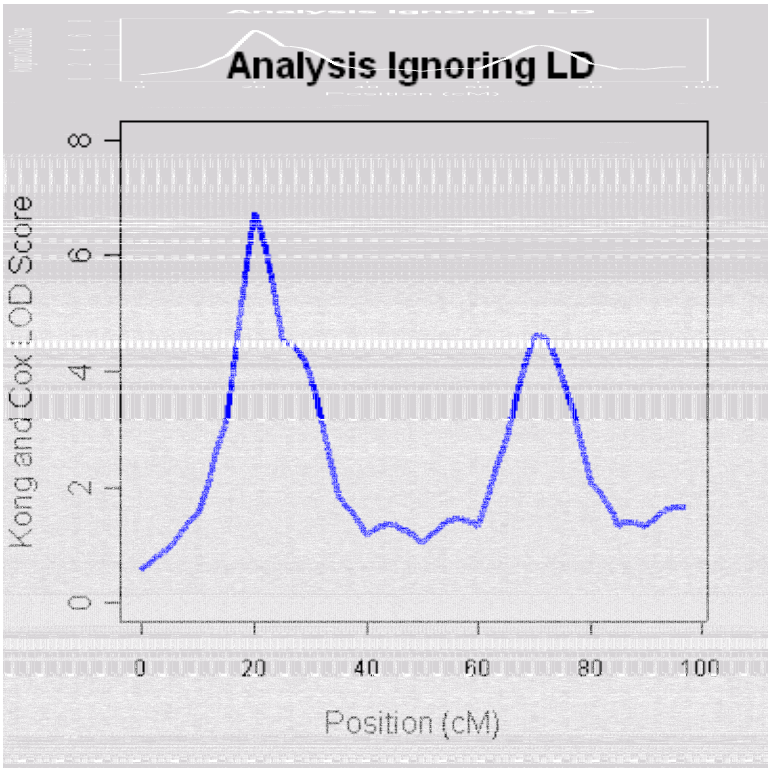
The three standard Merlin format input files are the data file *snp−scan.dat*, pedigree file *snp−scan.ped* and map file *snp−scan.map*. In the pedigree file, SNP alleles 'A', 'C', 'G' and 'T' have been coded as allele 1, 2, 3 and 4, respectively. All input files are text files, and you can check their contents using the UNIX **more** command or using the following **pedstats** command:

```
prompt>pedstats −d snp−scan.dat −p snp−scan.ped
```

We are going to evaluate the evidence for linkage is this SNP data set, and a good place to start, is to run a standard non−parametric linkage analysis (**−−npl** command option) ignoring linkage disequilibrium between markers. We will request that Merlin carry out analysis at positions spaced every 2 cM along the chromosome(**−−grid** *2* command line option). Try running the following command:

```
prompt>merlin −d snp−scan.dat −p snp−scan.ped −m snp−scan.map −−npl −−grid 2
```

After the opening banner screen, your results should be similar to the following:




```
Phenotype: DISEASE [ALL] (500 families)
=======================================================
            Pos    Zmean  pvalue    delta     LOD  pvalue
            min  -18.26     1.0   -0.408 -62.47     1.0
            max   54.77 0.00000    1.225  301.0 0.00000
          0.000    1.28    0.10    0.094   0.57    0.05
          5.000    1.85    0.03    0.109   0.96    0.02
         10.000    2.38   0.009    0.139   1.56   0.004
         15.000    3.23  0.0006    0.204   3.08 0.00008
         20.000    4.72 0.00000    0.308   6.75 0.00000
         25.000    3.97 0.00004    0.241   4.48 0.00000
         30.000    3.62 0.00014    0.234   3.98 0.00001
         35.000    2.64   0.004    0.149   1.86   0.002
         40.000    2.05    0.02    0.122   1.18   0.010
         45.000    2.27   0.012    0.126   1.35   0.006
         50.000    1.99    0.02    0.110   1.03   0.015
         55.000    2.27   0.012    0.133   1.43   0.005
         60.000    2.28   0.011    0.124   1.33   0.007
         65.000    3.21  0.0007    0.189   2.84 0.00015
         70.000    4.20 0.00001    0.236   4.62 0.00000
         75.000    3.78 0.00008    0.220   3.90 0.00001
         80.000    2.74   0.003    0.161   2.08  0.0010
         85.000    2.23   0.013    0.127   1.34   0.006
         90.000    2.20   0.014    0.127   1.33   0.007
         95.000    2.28   0.011    0.154   1.66   0.003
        100.000    1.87    0.03    0.187   1.65   0.003
```

The 4th column (labeled LOD score) is the Kong and Cox LOD score for this data. You will notice two very strong LOD score peaks, one around 20cM (LOD 6.75) and another around 70cM (LOD of 4.62). Unfortunately, ignoring marker−marker LD can lead to inflated LOD scores when some parental genotypes are missing. The results are typical of situations where marker−marker disequilibrium is not modeled appropriately and should not be taken as evidence for linkage.

To verify whether there is evidence for linkage, we will repeat the previous analysis, but modeling

marker−marker disequibrium. First, we will carry out analyses using pre−specified clusters and haplotype frequencies. Next, we will see how Merlin can automatically define clusters using the available marker map and genotype data.

We will use cluster definitions in the file *snp−scan.clusters*. This file describes clusters of SNPs in linkage disequilibrium. This file can be generated by the user, or by a previous MERLIN run. We will describe the file in detail, since it should help clarify how MERLIN models linkage disequilibrium.

The file describes a series of clusters, each consisting of a series of consecutive markers. The description of each cluster begins on a separate line with the word **CLUSTER** followed by a series of marker names, that must exactly match the data and map files. Optionally, this line can be followed by a series of entries, each on a separate line, describing the haplotypes in the cluster and their frequencies. Each of these lines begins with the word **HAPLO** followed by a haplotype frequency and a series of alleles.

For example, this is the first cluster in the *snp−scan.clusters* file:

```
CLUSTER rs556990 rs553316 rs7989953
HAPLO 0.4500   3   2   1
HAPLO 0.3167   3   2   3
HAPLO 0.2000   1   4   1
HAPLO 0.0333   1   4   3
```

The cluster includes three markers (rs556990, rs553316, rs7989953) organized into 4 distinct haplotypes. The first two markers are in complete linkage disequilibrium, such that allele 3 at rs556990 corresponds to allele 2 at rs553316, whereas allele 1 at rs556990 corresponds to allele 4 at rs553316. The last marker is in strong, but incomplete disequilibrium with the first two: allele 3 for rs7989953 nearly always occurs on the 1−4 haplotype for markers rs556990 and rs553316.

After reading the file with clustering information, MERLIN will do the following:

- Check that all markers within a cluster are contiguous. If they are not, you will get an error message.
- Check that all markers within a cluster map to the same genetic map position. If they do not, Merlin will nudge their positions to ensure the within cluster recombination rate is zero.
- If haplotype frequency estimates are not provided, they will be calculated using the available genotype data and a maximum−likelihood E−M algorithm.

So let's repeat the original analysis, but modeling of marker−marker disequilibrium enabled. To do this, use the following command−line:

```
prompt>merlin −d snp−scan.dat −p snp−scan.ped −m snp−scan.map −−npl −−grid 5 −−cluster snp−scan.c
```

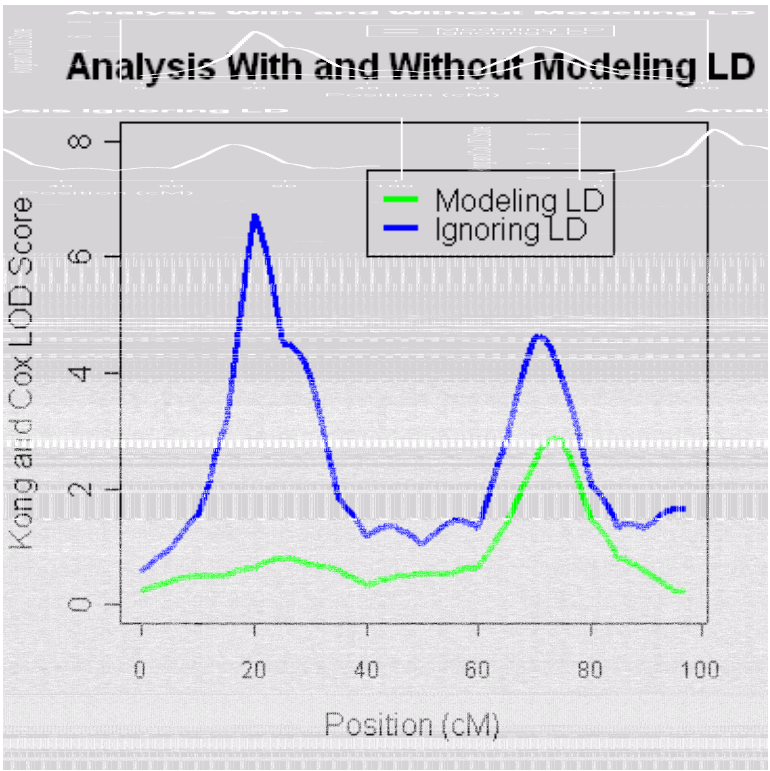
After the opening banner screen, you will first see a series of information messages:

```
MARKER CLUSTERS: Marker map changed, see [merlin−clusters.log]
MARKER CLUSTERS: User supplied file defines 20 clusters

Family:   101 − Founders: 2  − Descendants: 3  − Bits: 4
  Cluster at marker rs7334521 dropped [OBLIGATE RECOMBINANT]

Family:   287 − Founders: 2  − Descendants: 3  − Bits: 4
  Cluster at marker rs7334521 dropped [UNKNOWN HAPLOTYPE]
```

The first two lines indicate that the cluster information was successfully loaded. Since MERLIN assumes no recombination within clusters, the original genetic map was adjusted slightly −− you can examine the contents of *merlin−clusters.log* for details. In addition, MERLIN encountered two families (101 and 287) where

genotypes for one cluster did not fit with the model described in the clustering file. In family 101, the observed genotypes imply an obligate recombinant in the cluster including markers rs7334521, rs4495999 and rs9546406. In family 287, the observed genotypes imply a haplotype that is not present in the clustering file. In both families, genotypes for markers rs7334521, rs4495999 and rs9546406 will be marked as missing to allow analysis to proceed. In our experience, discarding a small proportion of the available genotypes in this manner results in no noticeable biases.

After these messages, you will find the linkage analysis results, which should be similar to the following:



Phenotype: DISEASE [ALL] (500 families)

```
==================================================
            Pos    Zmean  pvalue    delta     LOD  pvalue
            min   -18.26     1.0   -0.408 -62.47     1.0
            max    54.77 0.00000    1.225  301.0 0.00000
          0.011     0.82     0.2    0.061   0.24    0.15
          5.011     1.17    0.12    0.070   0.39    0.09
         10.011     1.32    0.09    0.078   0.49    0.07
         15.011     1.31    0.09    0.083   0.52    0.06
         20.011     1.43    0.08    0.098   0.67    0.04
         25.011     1.66    0.05    0.107   0.84    0.02
         30.011     1.54    0.06    0.102   0.75    0.03
         35.011     1.47    0.07    0.085   0.60    0.05
         40.011     1.10    0.14    0.067   0.35    0.10
         45.011     1.30    0.10    0.074   0.46    0.07
         50.011     1.41    0.08    0.079   0.53    0.06
         55.011     1.37    0.09    0.081   0.53    0.06
         60.011     1.57    0.06    0.087   0.65    0.04
         65.011     2.26   0.012    0.136   1.44   0.005
         70.011     3.05  0.0011    0.178   2.55  0.0003
         75.011     3.24  0.0006    0.192   2.92 0.00012
         80.011     2.34   0.010    0.138   1.53   0.004
         85.011     1.74    0.04    0.099   0.82    0.03
         90.011     1.45    0.07    0.084   0.58    0.05
         95.011     0.88     0.2    0.060   0.25    0.14
```

There is now a single linkage peak around 75cM (LOD of 2.92). The original peak around 20cM has completely disappeared, and was simply an artifact of linkage disequilibrium between markers. Thus, there is some good evidence for a single linkage peak in these data (at around 75cM). The analysis ignoring linkage disequilibrium, which showed an additional peak at around 20cM was quite inaccurate.

If you want to model linkage disequilibrium, but do not have a file describing preset clusters for your SNP mapping panel, MERLIN provides two options for automatically clustering markers. The **--distance** *k* option inserts a cluster breakpoint between markers that are less than *k* cM apart (that is, all consecutive markers spaced less than *k* cM are placed into a cluster). The **--rsq** *treshold* option calculates pairwise r^2 between neighboring markers and creates a cluster joining markers for which pairwise $r^2 >$ *threshold* and all intervening markers.

To explore these alternative options, try the following command lines:

```
prompt>merlin -d snp-scan.dat -p snp-scan.ped -m snp-scan.map --npl --grid 5 --rsq 0.1 --cfreq
prompt>merlin -d snp-scan.dat -p snp-scan.ped -m snp-scan.map --npl --grid 5 --dist 3 --cfreq
prompt>merlin -d snp-scan.dat -p snp-scan.ped -m snp-scan.map --npl --grid 5 --clusters snp-scan.
```

The first command-line, will search for markers for which r^2 is > 0.10 and define clusters including each identified pair and the intervening markers. The second command-line will group markers that are less than 3 cM apart into a cluster. The final command-line will use the cluster definitions in the snp-scan.clusters-only file, but estimate haplotype frequencies from the available genotype data. In each case, the **--cfreq** flag requests that the estimated clusters and their frequencies should be saved to a file.

That is it! You should be on your way to modeling linkage disequilibrium between markers in your own data, so as to make the best use of available SNP mapping panels.

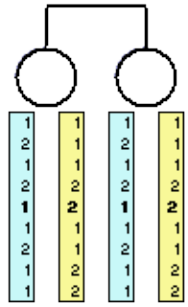
To learn about other analyses options, you might want to check the non-parametric linkage analysis or parametric linkage analysis sections, or proceed haplotyping, simulation or ibd estimation sections.

MERLIN Tutorial −− Error detection

Genotyping errors can lead to misleading inferences about gene flow in pedigrees and greatly reduce the effectiveness of pedigree analysis. In this section, we will use MERLIN to conduct a sensitivity analysis of the likelihood and identify problem genotypes.

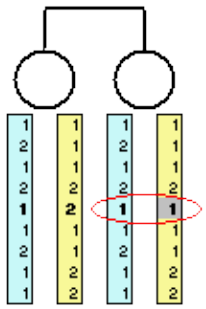
You can find the simulated data set for this section in the examples subdirectory of the MERLIN distribution or in the download page.

The dataset consists of a simulated 5−cM scan of chromosome 24 in 200 affected sib−pair families and is organized into 3 files, a data file (*error.dat*), a pedigree file (*error.ped*) and a map file (*error.map*). An overview of MERLIN input files is available elsewhere.



How does error detection work?

Before conducting the error detection analysis, we will review the basic principles behind it. Consider the simple pedigree to the left, with two siblings genotyped at several markers. Since their genotypes are identical at all markers, it seems quite likely that they share the stretch of chromosome under investigation.



Now, consider what happens if we change the genotype for a single marker (indicated by the red circle)... This marker now contradicts information provided by all others, indicating that perhaps one of the parents carried two nearly identical copies of the chromosome or two recombination events occurred.

In the first example, inference about inheritance is relatively consistent at all markers, while in the second example inference about inheritance is strongly influenced by the single genotype. Intuitively, the first outcome seems much more plausible.

MERLIN finds genotypes that provide information about gene flow in a pedigree that contradicts information provided by other available data. MERLIN considers all available data simultaneously (not just pairs of individuals) so that error detection improves in accuracy in larger pedigrees. Genotypes flagged by MERLIN are likely to be errors and are certainly worth checking!

Error detection using MERLIN

To run error detection using merlin, we need to provide an input pedigree file (**−p** command line option) and matching data and map files (**−d** and **−m** options) and request an error detection analysis (**−−error** option):

```
prompt> merlin −d error.dat −p error.ped −m error.map −−error
```

Try it out! You should see the merlin banner and a summary of selected options, followed by a list of unlikely genotypes. In this case, this is the list:

```
Family:     2 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK11 genotype for individual 3 is unlikely [0.003848]
  MRK11 genotype for individual 4 is unlikely [0.003848]

Family:    73 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK17 genotype for individual 3 is unlikely [0.008866]
  MRK17 genotype for individual 4 is unlikely [0.008866]

Family:    81 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK8 genotype for individual 3 is unlikely [0.001567]
  MRK8 genotype for individual 4 is unlikely [0.001567]

Family:    94 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK12 genotype for individual 3 is unlikely [0.002101]
  MRK12 genotype for individual 4 is unlikely [0.002101]

Family:   136 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK16 genotype for individual 3 is unlikely [0.008330]
  MRK16 genotype for individual 4 is unlikely [0.008330]

Family:   162 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK14 genotype for individual 3 is unlikely [0.003037]
  MRK14 genotype for individual 4 is unlikely [0.003037]

Family:   164 − Founders: 2  − Descendants: 2  − Bits: 2
  MRK6 genotype for individual 3 is unlikely [0.001805]
  MRK6 genotype for individual 4 is unlikely [0.001805]

Unlikely genotypes listed in file [merlin.err]
```

In this data set with 20 markers and 200 sib−pair families, MERLIN flagged 7 pairs of unlikely genotypes. Since we are dealing with sib−pairs, errors are not pinpointed to specific individuals (all that we can tell is that at least one of the siblings is likely to have an erroneous genotype in each family!).

In a real−life setting it would be worthwhile re−checking genotype assays for these individuals. In this case, we will simply run pedwipe to erase genotypes that are flagged as problematic. Run:

```
prompt> pedwipe −d error.dat −p error.ped
```

Pedwipe retrieves a list of unlikely genotypes from the *merlin.err* file and removes them from the data. A new set of data and pedigree files is created, named *wiped.dat* and *wiped.ped*. You can get a feel for the impact of these 7 problematic genotypes on linkage analysis by running a non−parametric linkage analysis before and after their removal:

```
prompt> merlin −d error.dat −p error.ped −m error.map −−npl
(...excerpt of results before removing problematic genotypes...)
Phenotype: affection [ALL] (200 families)
============================================================
                 Pos   Zmean  pvalue    delta    LOD  pvalue
```

```
              42.144    2.16    0.02    0.186   1.24   0.008
              47.412    2.39   0.008    0.204   1.51   0.004
              52.680    2.57   0.005    0.214   1.69   0.003
              57.948    1.72    0.04    0.145   0.76    0.03
              63.216    1.19    0.12    0.106   0.39    0.09

prompt> merlin −d wiped.dat −p wiped.ped −m error.map −−npl
(...excerpt of results after removing problematic genotypes...)
Phenotype: affection [ALL] (200 families)
============================================================
                 Pos   Zmean  pvalue    delta    LOD  pvalue
              42.144    2.24   0.012    0.191   1.32   0.007
              47.412    2.48   0.007    0.209   1.60   0.003
              52.680    2.87   0.002    0.237   2.10  0.0009
              57.948    2.10    0.02    0.175   1.13   0.011
              63.216    1.47    0.07    0.127   0.57    0.05
```

The seven problematic genotypes (out of 8,000 total genotypes), cause a 0.4 change in the Kong and Cox allele sharing LOD score! To learn about estimating false positive rates for error detection and linkage analysis you should proceed to the simulation section. Alternatively, you may want to learn more about linkage analysis, haplotyping or ibd estimation.

MERLIN Tutorial −− Haplotyping

Information about gene flow in a pedigree can be used to reconstruct likely haplotypes for families and individuals. In this section we will walk through some simple examples of how Merlin represents estimated haplotypes.

The sample input files used are in the examples subdirectory of the MERLIN distribution and are also available in the download page.

The first data set we will consider consists of very simple families, each with two parents and a single offspring genotyped for three SNP markers. The data is organized into three files: a pedigree file (*haplo.ped*), a data file (*haplo.dat*) and a map file (*haplo.map*).

Merlin has three haplotype estimation modes. It can either provide haplotypes corresponding to the most likely pattern of gene flow (**−−best** command line option), sample gene flow patterns according to their likelihood (**−−sample**) or provide all non−recombinant haplotypes (**−−zero −−all**). For this example, we will use the first option:

```
prompt> merlin −d haplo.dat −p haplo.ped −m haplo.map −−best
```

Estimated haplotypes are in the *merlin.chr* output file. Newer versions of Merlin will also produce a companion *merlin.flow* that summarizes the descent of estimated haplotypes through the pedigree. We will now examine these files in detail.

We will first examine the contents of the *merlin.chr* file. This file lists the two haplotypes for each individual (for non−founders the maternal haplotype is always listed first, followed by the paternal haplotype). The location of recombination events is also indicated (a **|** indicates no recombination event between the current locus and the previous informative locus, a **/** indicates a recombination event in the maternal haplotype, a **** indicates a recombination event in the paternal haplotype, a **+** indicates a recombination event in both the maternal and paternal chromosomes, and finally a **:** indicates information about recombination between the current marker and the previous marker is not available.)

By default, haplotypes are listed vertically, with multiple individuals per line (the **−−horizontal** command line flag selects an horizontal output format with a single haplotype per line and which can be more convenient for post−processing). Each family in the pedigree is listed in turn.

Let's look through the output! Notice that for the first family, father and child are heterozygous at all markers (and would have an uncertain haplotype without information on their relatives), whereas the mother is homozygous for allele '1' at all loci. Since Merlin considers all individuals jointly, all haplotypes can be resolved.

```
<−− contents of merlin.chr output file −−>

The first line names the family. In a trio family no
information on recombination is available, and this family
is labelled uninformative about recombination.
FAMILY 1 [Uninformative]

The next header line names individuals. Founders are labelled
F and non−founders are followed by their parents' names in
brackets.
       1 (F)              2 (F)             3 (2,1)

The next lines provide haplotype pairs for each individual. As noted above,
pairs are separated by a : if there is no information on recombination,
```

```
by a | if they do not recombine, or a /, \, + if they recombine
in the maternal, paternal or both chromosomes, respectively.
      2  :  1            1  :  1            1  :  2
      2  :  1            1  :  1            1  :  2
      2  :  1            1  :  1            1  :  2

<-- end of snippet -->
```

Output for the next family is similar, but you will notice that one chromosome carries an unknown allele which does not appear in any genotyped individuals. This is labelled by a **?** (question mark).

```
<-- continuation of merlin.chr output file -->
FAMILY 2 [Uninformative]

       1 (F)              2 (F)              3 (2,1)
      2  :  2            1  :  1            1  :  2
      2  :  1            1  :  1            1  :  2
      2  :  ?            1  :  1            1  :  2

<-- end of snippet -->
```

The next family presents a trickier challenge! Although all individuals are genotyped, phase is uncertain for the third marker. Either the father transmits a "2−2−2" chromosome to the child and the mother a "1−1−1" chromosome, or the father transmits a "2−2−1" chromosome and the mother transmits a "1−1−2" chromosome.

Merlin uses a special notation for ambiguous loci which can't be phased using the available information. In this case, the ambiguous phase at the third marker gives us an opportunity to examine this notation. At each locus where some ambiguity exists, each ambiguous allele is labeled with a specific uppercase letter ('A', 'B', 'C', ...) as well as two alternative allele choices. The ambiguity can be resolved by selecting either the first allele listed for all haplotypes in the set, or else by selecting the second allele for all haplotypes in the set.

This is what the output looks like:

```
<-- continuation of merlin.chr output file -->
FAMILY 3 [Uninformative]

       1 (F)              2 (F)              3 (2,1)
      2  :  2            1  :  1            1  :  2
      2  :  1            1  :  1            1  :  2
    2,1A : A1,2        1,2A : A2,1        1,2A : A2,1

<-- end of snippet -->
```

Compare to the sometimes tricky *merlin.chr* file, the *merlin.flow* file is a breeze. The file uses a unique label for each founder haplotype and helps discern descent of founder alleles through the pedigree as well as IBD relationships between individuals. In the example pedigrees, there are only 4 founder haplotypes, labeled "A", "B", "C" and "D". Here is what the Merlin output looks like:

```
<-- Contents of merlin.flow file -->
FAMILY 1 [Uninformative]

       1 (F)              2 (F)              3 (2,1)
       A : B              C : D               C : A
       A : B              C : D               C : A
       A : B              C : D               C : A

<-- end of snippet -->
```

Now that you know how to read Merlin haplotype output, you could look at more complex examples (try to haplotype the data set *gene.dat*, *gene.ped* and *gene.map*) or proceed to other sections of the tutorial. Available topics include linkage analysis, error detection, ibd estimation and simulation.

MERLIN Tutorial −− IBD and Kinship estimation

Since there a finite number of alleles at most genetic loci, individuals may exhibit the same genotype at a particular locus but, nevertheless, carry distinct chromosomes. Information on allele frequencies and neighbouring markers can be used to estimate the probability that any two individuals actually inherited the same chromosome from founders in the pedigree.

MERLIN can estimate the number of alleles shared identical−by−descent among relatives in a pedigree, and summarize this information either as probabilities that a given pair will share 0, 1 or 2 alleles IBD or as the kinship coefficient between each pair at a particular locus.

Some programs require IBD estimates as input for their analysis. For example, QTDT tests for association using all phenotypes from related individuals and requires IBD matrices to distinguish between linkage and association.

For this example, we will use a simulated data set in that you will find in the examples subdirectory of the MERLIN distribution or in the download page.

The data set includes 50 families, each with 4 siblings, genotyped for 3 SNP markers and is also used in the QTDT tutorial. We will use MERLIN to estimate IBD for this data set in a format that is ready for use by QTDT.

You should already be familiar with input file formats. The data consists of a pedigree file (*sibs.ped*), which specifies individual relationships, genotypes and phenotypes. In addition, a map file (*sibs.map*) provides marker locations and a data file (*sibs.dat*) describes the data set.

As usual, it is always a good idea to check contents of input files by running pedstats:

```
prompt> pedstats −d sibs.dat −p sibs.ped
```

To calculate pairwise IBD matrices, we will use the **−−ibd** command line option. Since MERLIN labels all results with chromosomal positions by default, we will also use the **−−markerNames** option to request that output include the marker names which are required by QTDT. So, the command:

```
prompt> merlin −d sibs.dat −p sibs.ped −m sibs.map −−markerNames −−ibd
```

Will estimate IBD coefficients for all relative pairs and produce a *merlin.ibd* file ready for use by QTDT. Each line in *merlin.ibd* begins with a family identifier followed by identifiers for two individuals. This is followed by marker names and probabilities for sharing 0, 1 and 2 alleles IBD.

Commonly used options when estimating IBD coefficients include **−−singlepoint** (which considers each marker independently) and **−−steps *n*** (which requests analysis at *n* positions between markers) or the **−−grid *k*** (which requests analysis every *k* cM along the chromosome).

Congratulations! You have reached the end of the Merlin tutorial. You may wish to review previous sections on input file formats, linkage analysis, error detection, simulation or haplotyping.

MERLIN Tutorial −− Association Analysis

Merlin can test for association between a SNP and one or more quantitative traits (if you are interested in discrete traits, you should consider the LAMP software package, which provides discrete trait association tests that integrate over missing genotypes in small pedigrees). The association test implemented in Merlin includes an integrated genotype inference feature, which can improve power when some genotypes are missing (Burdick et al. 2006). In this example, we will see how to carry out an association analysis using Merlin and how to use the integrated genotype inference feature to estimate missing genotypes.

The association tests implemented in Merlin can be used to analyze genome−wide association scans, or to study candidate regions. However, it is important to note that −− in contrast to standard family−based association tests −− the test implemented in Merlin does not control for population stratification. If population stratification is a concern, population membership should be included as a covariate or genomic control methods should be used to adjust results.

Alright ... let's walk through the analysis of an exemplar dataset. The dataset consists of 107 individuals in 9 three generation pedigrees (modelled after the CEPH pedigrees originally collected to help in linkage map construction, and which were more recently used by the HapMap Consortium to build a haplotype map of the human genotype and wre also used to study the genetics of gene expression by multiple independent groups). The data consist of genotypes for 20 SNP markers, with an average heterozigosity of about 40%. Six of the markers are genotyped for all individuals, the remaining 14 are genotyped in only 50−54 individuals. The dataset is organized into 3 files, a data file (*assoc.dat*), a pedigree file (*assoc.ped*), and a map file (*assoc.map*). All of these are available in the examples subdirectory of the Merlin distribution and, as usual, you can check their contents using **pedstats**.

To run Merlin for the association analysis, we need to specify the usual set of data (**−d** parameter), a pedigree (**−p** parameter), and a map files (**−m** parameter). In addition, we need to request one of the following association tests: a score test (**−−fastAssoc**) or a likelihood−ratio test (**−−assoc**). The score test (**−−fastAssoc**) is rapid and ideal for screening very large numbers of markers (for example, in a first pass analysis of a genome−wide association (GWA) scan), whereas the more accurate likelihood−ratio test (**−−assoc**) can be used to evaluate smaller numbers of markers (for example, in candidate regions selected for follow up analyses). In datasets that include only small pedigrees or when the effects being evaluated are small, the two tests will give very similar results.

In this example, we will first try the **−−fastAssoc** option, using the following command line:

```
prompt> merlin −d assoc.dat −p assoc.ped −m assoc.map −−fastAssoc
```

After running the command, you should first see a summary of the currently selected and available options. At the end of your output, you should see the following table of results:

```
Phenotype: mRNA [FAST−ASSOC] (9 families, h2 = 15.99%)
================================================================================
  Position        Marker  Allele  Effect      H2     LOD  pvalue
    56.077          SNP1       3   0.168   5.93%   1.186    0.02
    56.081          SNP2       2   0.168   5.71%   1.185    0.02
    56.081          SNP3       2   0.048   0.26%   0.051     0.6
    56.499          SNP4       1  −0.207   4.39%   0.906    0.04
    56.501          SNP5       3   0.172   4.11%   0.795    0.06
    56.509          SNP6       4  −0.058   0.64%   0.129     0.4
    56.938          SNP7       3   0.026   0.16%   0.032     0.7
    56.941          SNP8       4   0.026   0.17%   0.026     0.7
    56.949          SNP9       3   0.002   0.00%   0.000     1.0
    57.114         SNP10       1  −0.122   1.51%   0.205     0.3
    57.118         SNP11       3   0.497  47.02%   8.522 3.7e−10
```

```
   57.123         SNP12       2   0.315  23.60%   4.343 7.7e-06
   57.126         SNP13       4   0.315  23.60%   4.343 7.7e-06
   57.590         SNP14       2   0.115   3.14%   0.633    0.09
   57.600         SNP15       3   0.088   1.82%   0.312     0.2
   57.610         SNP16       3   0.088   1.82%   0.312     0.2
   59.410         SNP17       1   0.092   1.65%   0.344     0.2
   59.417         SNP18       4  -0.026   0.15%   0.027     0.7
   59.418         SNP19       1   0.166   4.69%   0.750    0.06
   59.784         SNP20       3   0.178   7.52%   1.432   0.010
  Peak -->        SNP11       3   0.497  47.02%   8.522 3.7e-10
```

The table summarizes the **--fastAssoc** analysis of phenotype "mRNA". The 7 columns are the position and name of the SNP being tested (markers with more than two alleles will be skipped), the allele being tested, the estimated effect of the allele, the proportion of total variance explained by the SNP, a LOD score statistic summarizing evidence for association and its corresponding p-value. The last row highlights the strongest association among all SNPs examined. In this case, it looks like every copy of allele '3' at SNP11 decreases phenotypic values by approximately 0.5 units. Overall, the SNP explains 47% of the variation in mRNA levels for the trait and is associated with a LOD score of about 8.5. Examining the detailed output, you'll see that two nearby SNPs are also strongly associated -- these are likely in linkage disequilibrium with the SNP that shows strongest association.

Since the results look interesting, it seems worthwhile to follow-up the score test with a more time-consuming maximum likelihood analysis. In large datasets, you could focus this follow-up analysis on the most promising SNPs using the **--start** and **--stop** options. If you do that, all SNPs outside the region specified by **-start** and **--stop** will still be used for inference of missing genotypes, but they won't be tested for association. In this case, there are only 20 SNPs to analyse and the maximum likleihood analysis shouldn't be too time consuming. Since we are dealing with relative large pedigrees (each pedigree has an average of >10 individuals) and a relatively large effect (the SNP explains nearly half of the variation in phenotypic values), we expect that the maximum likleihood analysis will provide us with more accurate results. To carry out the follow-up analysis, try the following command line:

```
prompt> merlin -d assoc.dat -p assoc.ped -m assoc.map --assoc
```

You should see the following results table towards the end of Merlin output:

```
Phenotype: mRNA [ASSOC] (9 families, h2 = 15.99%)
================================================================================
---- LINKAGE TEST RESULTS ----  ----------- ASSOCIATION TEST RESULTS ----------
 Position    H2    LOD  pvalue     Marker Allele  Effect      H2    LOD  pvalue
   56.077  44.1%  2.12  0.0009       SNP1      3   0.182   6.74%   1.06    0.03
   56.081  44.2%  2.12  0.0009       SNP2      2   0.182   6.50%   1.07    0.03
   56.081  44.2%  2.12  0.0009       SNP3      2   0.058   0.37%   0.05     0.6
   56.499  46.5%  2.56  0.0003       SNP4      1  -0.192   3.65%   0.67    0.08
   56.501  46.5%  2.56  0.0003       SNP5      3   0.178   4.25%   0.58    0.10
   56.509  46.5%  2.57  0.0003       SNP6      4  -0.031   0.18%   0.02     0.7
   56.938  46.8%  2.87 0.00014       SNP7      3   0.053   0.67%   0.10     0.5
   56.941  46.8%  2.87 0.00014       SNP8      4   0.061   0.87%   0.10     0.5
   56.949  46.8%  2.87 0.00014       SNP9      3   0.020   0.08%   0.01     0.8
   57.114  46.8%  2.87 0.00014      SNP10      1  -0.114   1.28%   0.14     0.4
   57.118  46.8%  2.87 0.00014      SNP11      3   0.477  42.46%   8.48 4.1e-10
   57.123  46.8%  2.87 0.00014      SNP12      2   0.283  18.66%   2.74  0.0004
   57.126  46.8%  2.87 0.00014      SNP13      4   0.283  18.65%   2.74  0.0004
   57.590  46.8%  2.87 0.00014      SNP14      2   0.098   2.24%   0.34     0.2
   57.600  46.8%  2.87 0.00014      SNP15      3   0.066   1.01%   0.13     0.4
   57.610  46.8%  2.87 0.00014      SNP16      3   0.066   1.01%   0.13     0.4
   59.410  47.0%  2.87 0.00014      SNP17      1   0.094   1.69%   0.26     0.3
   59.417  47.0%  2.87 0.00014      SNP18      4  -0.042   0.39%   0.05     0.6
   59.418  47.0%  2.87 0.00014      SNP19      1   0.153   3.92%   0.48    0.14
   59.784  47.1%  2.87 0.00014      SNP20      3   0.158   5.86%   0.88    0.04
```

```
                Peak -->      SNP11      3   0.477  42.46%   8.48 4.1e-10
```

The two commands we just walked through, **--assoc** and **--fastAssoc**, are the two you will use most often when testing for association. The commands work within Merlin for autosomal analysis, and also within Minx for the analysis of X-linked markers. You will often find it useful to combine them with the --pdf option (which generates a graphical summary of their results) and the --inverseNormal option (which automatically transforms traits so they follow a smooth normal distribution). Below, we describe how to carry out sequential association analyses (to identify multiple SNPs that are associated with the trait of interest) and how to get Merlin to output imputed genotype distirbutions for analysis in other programs. You may decide to only read about those options later, after you have tried out the **--assoc** and **-fastassoc** options on your own data.

Advanced Exercise - Sequential Association Analysis

Merlin usually tests for association one SNP at a time. After identifying the most strongly associated SNP, it is often interesting to check whether this SNP can account for the association at other neighboring SNPs and to search for other independently associated SNPs. One way to do this is to gradually refine our trait model. We might start with a model that includes only environmental covariates and search for the best associated SNP. After this SNP is identified, we might add it to the list of covariates and re-evaluate the evidence for association at all other SNPs. And so on ...

To customize the covariate list for qunatitative trait association analysis, we use the **--custom** option. This option specifies a file that describes a series of customized trait models. Each model starts with a trait name (indicated by the TRAIT keyword) and is optionally followed by a list of covariates (indicated by the COVARIATES keyword). To carry out a sequential association analysis, we start with a very simple custom model file (in this example, we will use the assoc.tbl file) and gradually refine it by including the best SNP from each round as a covariate.

To get things started, run the command:

```
prompt> merlin -d assoc.dat -p assoc.ped -m assoc.map --custom assoc.tbl --assoc
```

The *assoc.tbl* file is very simple, and includes a single line of interest:

```
< Contents of assoc.tbl file >
TRAIT mRNA
< End of assoc.tbl file >
```

Thus, it simply specifies that Merlin should run a simple analysis for the trait mRNA wiht no covariates. When you run Merlin you should see the following output:

```
CUSTOM QUANTITATIVE TRAIT MODEL #1
==================================

TRAIT: mRNA
  No covariates

Phenotype: mRNA [ASSOC] (9 families, h2 = 15.99%)
================================================================================
---- LINKAGE TEST RESULTS ----  ----------- ASSOCIATION TEST RESULTS ----------
 Position    H2    LOD  pvalue     Marker Allele  Effect      H2    LOD  pvalue
   56.077  44.1%  2.12  0.0009       SNP1      3   0.182   6.74%   1.06    0.03
   56.081  44.2%  2.12  0.0009       SNP2      2   0.182   6.50%   1.07    0.03
   56.081  44.2%  2.12  0.0009       SNP3      2   0.058   0.37%   0.05     0.6
   56.499  46.5%  2.56  0.0003       SNP4      1  -0.192   3.65%   0.67    0.08
   56.501  46.5%  2.56  0.0003       SNP5      3   0.178   4.25%   0.58    0.10
   56.509  46.5%  2.57  0.0003       SNP6      4  -0.031   0.18%   0.02     0.7
```

```
  56.938  46.8%  2.87 0.00014       SNP7      3   0.053   0.67%   0.10     0.5
  56.941  46.8%  2.87 0.00014       SNP8      4   0.061   0.87%   0.10     0.5
  56.949  46.8%  2.87 0.00014       SNP9      3   0.020   0.08%   0.01     0.8
  57.114  46.8%  2.87 0.00014      SNP10      1  -0.114   1.28%   0.14     0.4
  57.118  46.8%  2.87 0.00014      SNP11      3   0.477  42.46%   8.48 4.1e-10
  57.123  46.8%  2.87 0.00014      SNP12      2   0.283  18.66%   2.74  0.0004
  57.126  46.8%  2.87 0.00014      SNP13      4   0.283  18.65%   2.74  0.0004
  57.590  46.8%  2.87 0.00014      SNP14      2   0.098   2.24%   0.34     0.2
  57.600  46.8%  2.87 0.00014      SNP15      3   0.066   1.01%   0.13     0.4
  57.610  46.8%  2.87 0.00014      SNP16      3   0.066   1.01%   0.13     0.4
  59.410  47.0%  2.87 0.00014      SNP17      1   0.094   1.69%   0.26     0.3
  59.417  47.0%  2.87 0.00014      SNP18      4  -0.042   0.39%   0.05     0.6
  59.418  47.0%  2.87 0.00014      SNP19      1   0.153   3.92%   0.48    0.14
  59.784  47.1%  2.87 0.00014      SNP20      3   0.158   5.86%   0.88    0.04
                     Peak -->      SNP11      3   0.477  42.46%   8.48 4.1e-10
```

Refined association models stored in [merlin-assoc-covars.*]

The results should be identical to the ones from the earlier analysis, used to demonstrate the **--assoc** option. However, the key thing for us is the final line of output -- which indicates Merlin has automatically generated a set of files that will help in our sequential analysis. The set inlcudes three files. One of these, *merlin-assoc-covars.tbl*, includes a refined trait model that now includes SNP11 as a covariate. The other two, *merlin-assoc-covars.dat* and *merlin-assoc-covars.ped*, include an appropriately coded covariate which indicates the number of copies of allele '3' at SNP11 carried by each individual.

To continue our sequential analysis, we first merge the covariate into the original pedigree file and rename *merlin-assoc-covars.tbl* so that it is not overwritten when we next run Merlin. Run the following series of commands:

```
prompt> pedmerge assoc merlin-assoc-covars assoc-stage2
prompt> mv merlin-assoc-covars.tbl assoc-stage2.tbl
```

The first command combines the original pedigree file with the covariate data automatically generated by Merlin. The second command renames the trait model file generated by Merlin, so it is not overwritten on our next analysis (on windows, you should replaced the **mv** command with the **move** command). We are now ready to run the second round association analysis:

```
prompt> merlin -d assoc-stage1.dat -p assoc-stage1.ped -m assoc.map --custom assoc-stage1.tbl --a
```

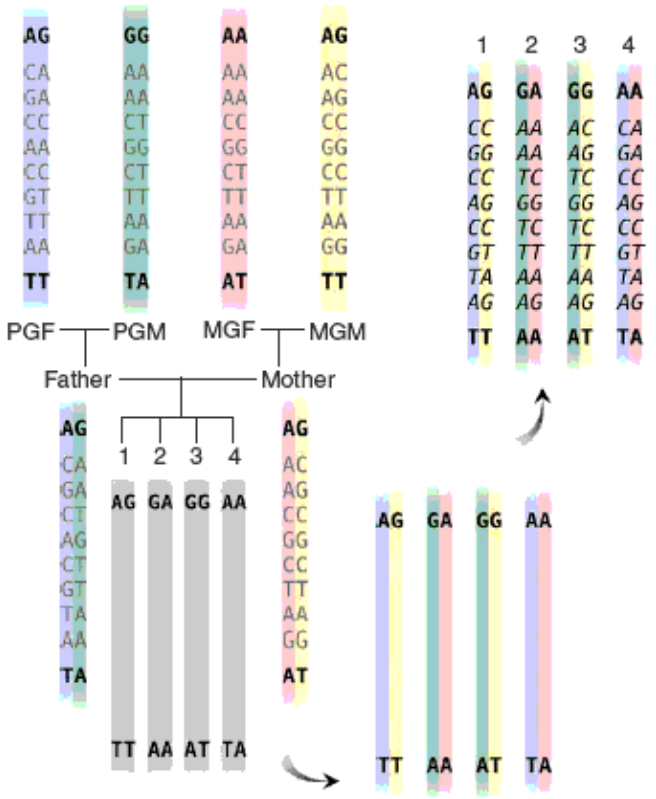
The results of this second round (pasted below), show that SNPs 12 and 13 (which showed some evidence for association in the first pass analysis) are no longer significantly associated -- their effects were likely a consequence of their closeness to SNP11. The only SNP that shows marginal evidence for association is SNP20, but this is likely not significant after adjusting for multiple testing. Thus, we stop our sequential analysis here!

```
CUSTOM QUANTITATIVE TRAIT MODEL #1
==================================

TRAIT: mRNA
  COVARIATES: SNP11

Phenotype: mRNA [ASSOC] (9 families, h2 = 0.00%)
================================================================================
---- LINKAGE TEST RESULTS ----  ----------- ASSOCIATION TEST RESULTS ----------
 Position    H2    LOD  pvalue      Marker Allele  Effect      H2    LOD  pvalue
   56.077   0.0%  0.00     0.5        SNP1      3   0.037   0.49%   0.10     0.5
   56.081   0.0%  0.00     0.5        SNP2      2   0.037   0.47%   0.10     0.5
```

```
  56.081   0.0%  0.00     0.5       SNP3      2   0.027   0.14%   0.03     0.7
  56.499   0.0%  0.00     0.5       SNP4      1  −0.103   1.78%   0.38     0.2
  56.501   0.0%  0.00     0.5       SNP5      3  −0.022   0.11%   0.02     0.8
  56.509   0.0%  0.00     0.5       SNP6      4  −0.089   2.46%   0.62    0.09
  56.938   0.0%  0.00     0.5       SNP7      3  −0.009   0.03%   0.01     0.9
  56.941   0.0%  0.00     0.5       SNP8      4  −0.009   0.03%   0.01     0.9
  56.949   0.0%  0.00     0.5       SNP9      3   0.037   0.45%   0.08     0.5
  57.114   0.0%  0.00     0.5      SNP10      1  −0.090   1.34%   0.21     0.3
  57.118   0.0%  0.00     0.5      SNP11      3       –       –      –       –
  57.123   0.0%  0.00     0.5      SNP12      2  −0.030   0.35%   0.04     0.7
  57.126   0.0%  0.00     0.5      SNP13      4  −0.030   0.35%   0.04     0.7
  57.590   0.0%  0.00     0.5      SNP14      2   0.101   3.98%   0.91    0.04
  57.600   0.0%  0.00     0.5      SNP15      3   0.091   3.20%   0.63    0.09
  57.610   0.0%  0.00     0.5      SNP16      3   0.091   3.20%   0.63    0.09
  59.410   0.0%  0.00     0.5      SNP17      1  −0.001   0.00%   0.00     1.0
  59.417   0.0%  0.00     0.5      SNP18      4   0.013   0.06%   0.01     0.8
  59.418   0.0%  0.00     0.5      SNP19      1   0.091   2.34%   0.41     0.2
  59.784   0.0%  0.00     0.5      SNP20      3   0.119   5.58%   1.13    0.02
                    Peak −−>      SNP20      3   0.119   5.58%   1.13    0.02

Refined association models stored in [merlin−assoc−covars.*]
```

Advanced Exercise − Standalone Genotype Inference

In this more detailed analysis, Merlin first evaluates the evidence for linkage at each position. The results are summarized in the first 4 columns of the summary table, which show the position of the SNP being tested, the proportion of variance that is explained by IBD sharing at that position in a variance component linkage analysis and the corresponding LOD score and p−value (the **−−vc** option provides more detailed linkage analysis results). Because we are examining a relatively small region, you will notice that the linkage signal changes only very gradually and is nearly flat for most of the region. The next set of columns summarizes results of the association test. You will see the name of the SNP and allele being tested, the estimated effect of the allele, the proportion of the trait variance it explains, and finally the LOD score and p−value evaluating the evidence for association. In this case, the **−−assoc** option found even stronger evidence for association (as expected, since the SNP being tested is very close to the gene encoding the mRNA levels we measured).



One unique feature of association tests in MERLIN is that missing genotypes are imputed and incorporated in the association test. A cartoon illustration of the procedure is provided in the figure above (adapted from Burdick et al. 2006). In the figure, all missing genotypes can be inferred by using information from flanking markers. Even when this is not the case, missing genotypes can often be estimated using information at other markers and family relationships. In principle, genotype inference can substantially improve power of association tests. In fact, we expect that when a genome−wide association scan follows a linkage study, only a porportion of individuals may need to be genotyped and genotypes of the remaining family members can be estimated computationally.

Merlin allows genotype inference to be decoupled from association tests. To estimate missing genotypes, use the **−−infer** parameter. Details of estimated genotypes will be stored in a pair of files, *merlin−infer.dat* and *merlin−infer.ped*. Try the following command line:

```
prompt> merlin −d assoc.dat −p assoc.ped −m assoc.map −−infer
```

In the inferred pedigree file (saved as *merlin−infer.ped* here), each genotype is described in 5 columns: the most likely genotype, the expected number of copies for the tested allele (0, 1, or 2 if the genotype is observed; but fractional counts may occur for missing genotyeps), and the posterior probabilities for the three alternative genotypes. This information can be useful to other programs that can use imputed genotype distributions to test for association, but that are not themselves equipped with an integrated genotype inference feature.

This tutorial was written by Weimin Chen and Goncalo Abecasis. Hope you enjoyed it!