

NAME

lnkg2lmm – converts data and parameters from LINKAGE format to lm_markers format

SYNOPSIS

lnkg2lmm [OPTIONS]

OPTIONS*Input files*

--pf=FILE, --pedfile=FILE

Specify the name of the **LINKAGE**-format pedigree file (*FILE*). Default is *pedfile.dat*.

--df=FILE, --datafile=FILE

Specify the name of the **LINKAGE**-format “datafile” (*FILE*). Default is *datafile.dat*.

--alleles=FILE

Specify the name of the file containing the allele-code conversions. Default behavior is to leave alleles unconverted.

Output files

--out_ped=FILE

Specify the name of the **lm_markers** pedigree file (*FILE*) that will be created. Default is *lmm_ped*.

--out_par=FILE

Specify the name of the **lm_markers** parameter file (*FILE*) that will be created.. Default is *lmm_par*.

--out_mrks=FILE

Specify the name of the **lm_markers** marker-data file (*FILE*) that will be created. Default is *lmm_mrks*.

--out_seeds=FILE

Specify the name of the **lm_markers** seed file (*FILE*) that will be created. Default is *lmm_seeds*.

Other options

--quant_miss=STRING

Specify the *string* that is used as the missing data code for a quantitative trait in the input pedigree file. Default is “0”.

--id_sep=STRING

Specify the *string* that will be used to delimit the family and individual IDs when creating unique IDs. Default is “_”. It is best to use a character that does not exist in the family and individual IDs.

--iters=N

The number of MCMC iterations (N) that will be specified in the **lm_markers** parameter file. Default is to leave this blank and let the user fill it in later.

--help, --?

Display brief documentation.

--man

Display complete documentation in manpage format (or use pod2text, pod2man, or another pod utility).

For all of the input and output options, “-” can be used in place of a file name to print to stdout, or read from stdin. Use “/dev/null” in place of an output file name if you don’t want a particular output file.

DESCRIPTION

Converts data and parameters from **LINKAGE** format into **lm_markers** format. Input must include the so-called “datafile” (usually called *datafile.dat*), which contains the parameter values for the markers and trait, and the pedigree file (which contains the actual data and is usually called *pedfile.dat*). The pedigree file should be in pre-makeped format.

The “datafile” must meet fairly strict criteria: It must be in either **MLINK** or **LINKMAP** format; the trait must be the first locus; the markers must be listed in the same order as they occur on the genetic map; and the trait must be either a binary or quantitative trait. Multiple liability classes are not allowed. If your input files do not meet these criteria, then you might receive a cryptic, non-descriptive perl-ish error message (or *many* such error messages).

If the trait is quantitative, then the input missing-data code is, by default, assumed to be the *string* “0” (without quotes), *not* the numerical value 0 (i.e. “0.0” will not be treated as missing). Note that this is probably not quite the same assumption that the **LINKAGE** programs make. You can use the `--quant_miss` option to specify a different string as the missing-data code, but the missing-ness will still be based on string values, not numeric values.

Four files will be created: a pedigree file, containing the pedigree structure and trait values; a marker data file; a seed file; and an `lm_markers` parameter file, which defines the model and other things that are necessary for running `lm_markers`. In order to run `lm_markers`, you will need to choose the number of MCMC iterations, either by using the `--iters` option, or by editing the parameter file after it is created. In either case, I strongly suggest that you examine the parameter file before running `lm_markers`. It is also a good idea to make sure that the pedigree and marker files look sensible.

Optionally, an allele-conversion file can be given as input, if the marker alleles in the pedfile are not consecutive integers (1, 2, 3, ...). In this case, each allele can be any string that does not contain whitespace. The allele-conversion file should have one line for each marker, with each line containing the input codes for alleles 1, 2, ... For example, if you have two microsatellites and one SNP, your “alleles” file might look like this:

```
150 152 154 156 160 162 166
94 98 100 102 106
A T
```

In this example, allele “154” at the first locus will be recoded as “3”, “102” at the second locus will be recoded as “4”, “A” at the third locus will be recoded as “1”, etc. Make sure that the allele frequencies given in the datafile correspond to this recoding. Note that “0” (without quotes) is always assumed to be the missing-data code for marker data, in accordance with the **LINKAGE** format.

EXAMPLES

```
lnkg2lmm --pf=myped --df=mydat
```

Read pedigree data from *myped*, and read parameter values from *mydat*.

```
lnkg2lmm --quant_miss=-99 --df mydat --pf myped
```

Same as above, but assume that missing data code in the input is -99.

```
lnkg2lmm --alleles allele_codes
```

Read from default file names (*datafile.dat* and *pedfile.dat*), and convert the marker data based on the conversion given in the file *allele_codes*.

SEE ALSO

MORGAN, including **lm_markers**, is available at <http://www.stat.washington.edu/thompson/Genepi/MORGAN/Morgan.shtml>.

The **MORGAN** tutorial is available at <http://www.stat.washington.edu/thompson/Genepi/MORGAN/Morgan.shtml#tut>.

The **LINKAGE** input file format is (at least partially) described at <http://linkage.rockefeller.edu/soft/linkage/>.

AUTHOR

Joe Rothstein <joe419@u.washington.edu>