LinDen User Guide

Tyler Cowman[1], Mehmet Koyutürk[1,2]

(1) Dept. of Electrical Engineering & Computer Science,
(2) Center for Proteomics & Bioinformatics
Case Western Reserve University, Cleveland, OH 44106, USA.
e-mail: tyler.cowman@case.edu

1 Compiling

Currently, LinDen is only available for the Linux operating system. In order to compile LinDen, navigate to its directory through a terminal and type *make*. If this fails, make sure you have an up to date version of the gcc compiler.

2 Input Format

LinDen utilizes three plain text files as input. The locations of each must be specified in the appropriate fields of a valid configuration file. The number of rows in each file should be equal, as each row corresponds to single locus. In the control and case files, each column represents a sample.

2.1 Info File

This should be a three column tab delimited text file. The first column is the name/identifier for the loci. The second and third columns are the chromosome number and base-pair locations respectively. Currently the chromosome number must be an integer, ex: sex chromosomes need to be recoded from x and y if they are labeled as such.

2.2 Controls File

This should consist of a grid {0, 1, 2} characters, representing homozygous major, heterozygous and homozygous minor genotypes respectively for all control samples. Rows correspond to loci and there should be no delimiter between columns.

2.3 Cases File

The formatting for this file should be identical to the controls file except it should contain all case samples.

3 Configuration File

LinDen uses a configuration file rather than passing a series of command line arguments. This file is parsed one line at a time. Only lines that begin with #file or #parameter are parsed, thus all other lines are treated as comments. There is an example configuration file for the toy input included with the source code download.

Below us an example of the configuration file line specifying the maximum number of threads that LinDen will create. The *-np* stands for normal priority and only makes a difference when multiple values are provided for more than one parameter. It controls the order in which the parameter permutations are run. This is explained more in the example configuration file but this shouldn't make much of a difference. However, the *maxUnknownFraction* parameter should generally be set to *-lp* to ensure that the correct ground truth file are used when a series of runs including $maxUnknownFraction = 0.0$ is used. Notice that the space between the semicolon and the last parameter value is necessary.

```
#parameter -np maxThreadUsage 4 ;
```

- **infoFilePath** The location of the input info file.
- **controlsFilePath** The location of the input control samples file.
- **casesFilePath** The location of the input case samples file.
- **outputFileName** Prefix for the three output files. If the output directory does not contain the files with this prefix they will be created. Otherwise, the results will be appended onto the existing files.
- **snpRange** This specifies a subset of the total input to parse. The syntax consists of { from to } where from and to refer to the loci in the input set (rows). In most cases this should be set to { -1 -1 } which has been coded as the entire set of input loci. Also, notice the syntax must be exact with a space delimiting each component.
- **permuteSamples** This can be set to 0 or 1 meaning permute and don't permute respectively. Permuting the samples consists of randomly assigning each sample (column) as a case or control. This is useful for permutation testing to obtain a measure of pairwise significances of loci in the input dataset under a null distribution.
- **noTrees** This parameter can also be set to 0 or 1. Setting it to 1 will essentially treat all LD-Trees as flat groupings. A test between two groups consists of randomly choosing one locus from each as a representative. This option was mostly used for testing and there is not really a reason to set this to 1.
- **maxThreadUsage** Sets the maximum number of threads that will be utilized by LinDen. This should be set to the number of precessing units LinDen is intended to be run on. The runtime scaling with the number of processing units is roughly linear.
- **maxUnknownFraction**
- **minimumMinorAlleleFrequency** This is used to filter out loci in which the minor allele frequency is too low for effective analysis. A common threshold is 0.05
- **maxMarginalSignificance** This parameter takes an integer value from 1-6, representing the maximum marginal significance of loci for consideration in pairwise testing as a -$\log_{10}$ p-value. For example, a value of 3 will filter out all loci with a marginal significance less than 0.001

4 Running LinDen

LinDen is a command line program with two modes of operation {run, formatout}. To run linden, a valid configuration file within the conf directory must be specified, as well as the desired mode. When running LinDen, all runs under the same output name are appended to the same file by row. Thus we provide a tool to convert the row based output to an easier to read and parse format.

```
$ ./linden (file.conf) (mode)
```

If the formatout mode is chosen, a third argument for the max number of top k pairs in each row of the raw input file to include must be provided.

```
$ ./linden (file.conf) (mode) (topk)
```

5 Output

After running LinDen, three files will be either created, or appended to, in the output directory.

5.1 Summary

The summary file contains several statistics on a run of LinDen, such as the parameter values used, statistical tests performed, and measures of precision and recall if a ground truth file had been generated prior to the run. Each row corresponds to a separate run.

5.2 Cutoff Pairs

Contains all detected pairs above the dynamic significance threshold at the conclusion of a run of LinDen.

5.3 Reciprocal Pairs

Contains a subset of the pairs in the cutoff pairs set such that both loci in a pairing had the other locus selected as its most significant interaction.

5.4 Formatted

After formatting either a cutoff pairs or reciprocal pairs file this will produce an eight column output with : row, chi-squared significance, locus 1 and 2, chromosome 1 and 2, base pair 1 and 2. SO each row represents a single locus pairing.