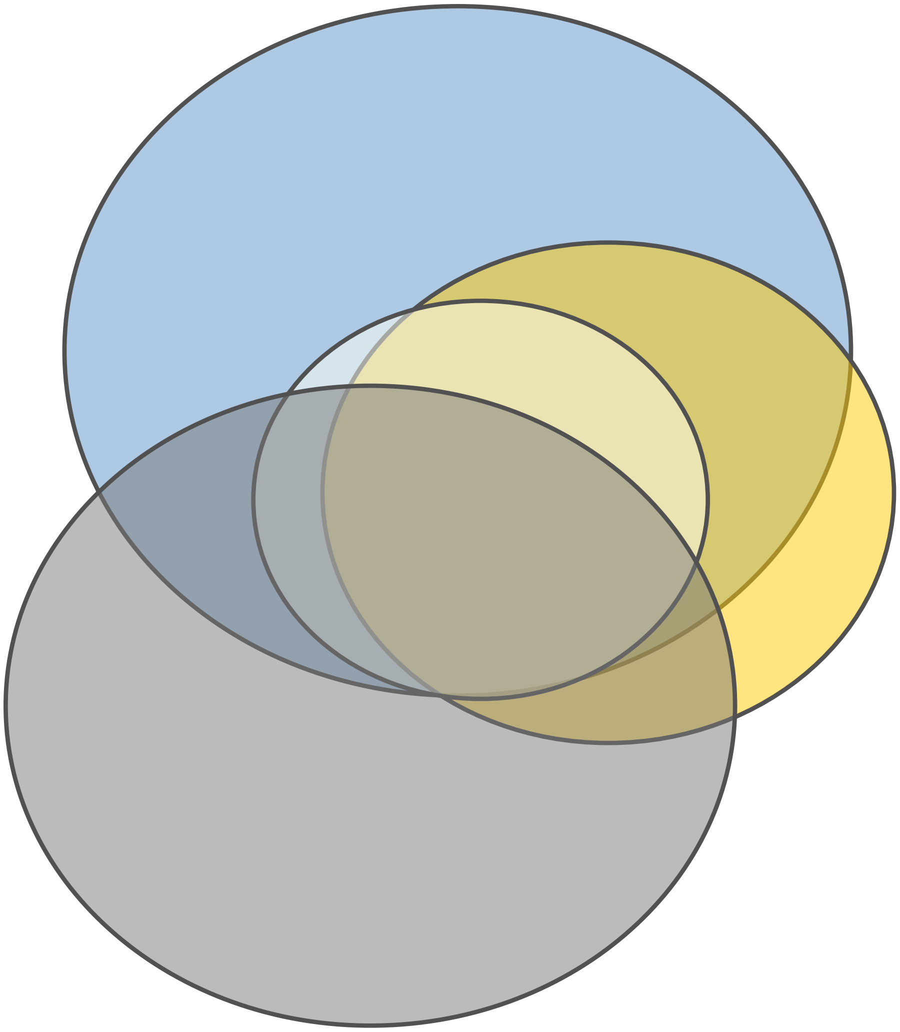
LD SPLINE

Reference Manual rev. 1.0.1



ldspline - Contiguous LD Representation

http://chgr.mc.vanderbilt.edu/ritchielab/method.php?method=ldspline

Table of Contents

`ldspline import-lomap ldspline-filename new-build-number` 7

Introduction

Purpose of this manual

Contained within this manual are details for configuring and running the application, ldspline. If this is your first time to use the software, we highly recommend that you take a few minutes to download and work through one or more tutorials. Then, once familiar with the capabilities of the software, users can refer to this guide when making changes to the basic configuration settings.

Conventions Used

There are two conventions used throughout this document. These text conventions are intended to help distinguish examples from configuration parameters.

Example commands

```
ldspline64 report-bounds ceu dp 1.0 19 120323 120500
```

The application itself will be named differently according to the platform it was compiled for. On Linux systems, it will be ldspline or ldspline64 depending on whether it runs on 32bit or 64bit systems. Windows and OS X will have be named slightly differently according to their platform as well.

Program Output

```
Program output will be in a gray box. Some lines will
often be removed when the output is lengthy.
```

Configuration details are listed first in bold- left aligned with the rest of the text.

The first word(s) are the keywords which specify what is being changed. Each keyword (or phrase) has some number of parameters. These are listed in the order they should appear in the configuration line. In some cases, parameters can be repeated or are optional. Those are denoted inside []s.

Configuration details are generally followed immediately by an example line:

```
This is an example
```

Examples show how an actual entry would look and are followed by some descriptive information to help the user understand how the example would affect the ldspline application runtime.

Common Parameters

There are a number of parameters which are used commonly across multiple configuration settings. In order to simplify the descriptions of the various properties of each command, we'll describe those properties here, and just refer to them as if they were a type.

Integer

Parameters specified in this way just simply refer to a whole number. In general, these values should be equal to or greater than 0, except when specified otherwise.

Float

Values specified as float are decimal values.

Index

If a parameter is listed as an index, it refers to the index, starting at 1 the user wishes to select.

max

This is generally an integer value representing the upper bound of some value. In some cases, such as minor allele frequency, it might represent a floating point value.

min

This is generally an integer value representing the lower bound of some value. In some cases, such as minor allele frequency, it is possible that it represents a floating point value.

On/Off

These parameters accept a boolean, Yes/No type setting. Users can use ON/OFF or YES/NO to set them.

filename

When a configuration refers to a file for input or output, the filename is generally used. This can be either a fully qualified path (such as /home/torstees/wga) or it can specified as a path relative to the directory where the application was run (such as ../data/goodfilename). It can also be just a plain filename as long as the file itself is available from the directory in which the application was run.

label

A label refers to a parameter whose value can be any text string without whitespace. These labels are generally used for reporting but in many cases are used to determine filenames. As a result, users should avoid using unusual characters in the string that could possibly cause problems with filenames. Because spaces and tabs are used to separate each parameter on a given line, labels can not contain spaces.

description

A description is a chunk of text that can contain spaces. It will always be at the very end of a line and is generally optional.

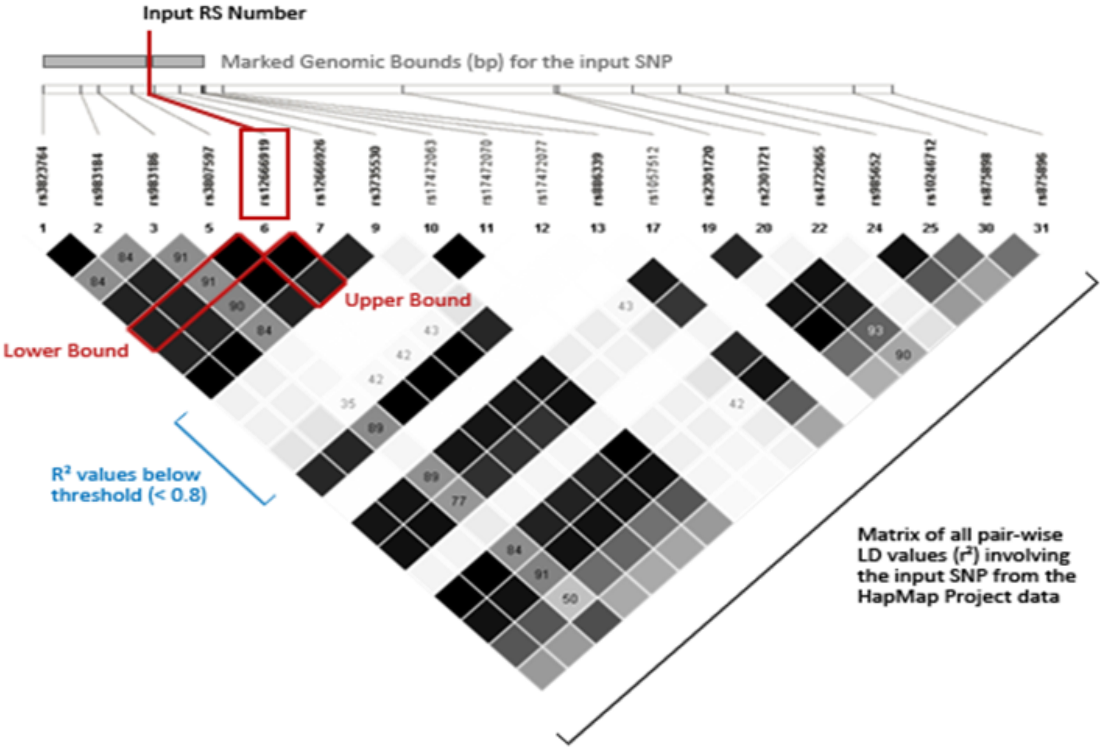
Using the ldspline Application

Unlike most of the applications developed by the ritchie lab, ldspline does not use a configuration file. Instead, all variations on program execution are done at the command line. While many of these commands take similar sets of parameters, some do vary, which will be covered in the following document.

In order to more clearly describe the various parameters required, this document will present each function independently, and the appropriate parameters that are used.

In general, the first parameter is the command (summarize, list, report-bounds, etc) and the second is the spline filename. This filename can be a fully qualified filename with or without the extension .ldspline. This extension should not be changed and is assumed.

The term spline below refers to the stretch of pairwise LD both up and downstream from a single SNP where the LD remains above the given threshold. Splines can represent either RSquared or DPrime.



Summarize

The summarize report is mostly used for debugging, but it can be helpful to users to identify exactly which SNPs are found inside a given Spline file. Summary reports are based on a single chromosome.

ldspline summarize spline-filename chromosome

ldspline summarize ceu X

This would generate a report for the chromosome X based on the spline file ceu.ldspline that exists in the user's current directory. Below is a small portion of the output you might see.

```
3370    X       rs5961354       5675878 9994064572      192     157
3371    X       rs5961355       5675932 9994067372      193     156
3372    X       rs5961356       5676132 9994070172      194     155
3373    X       rs1032040       5677733 9994072972      194     156
3374    X       rs1482826       5679969 9994075780      190     156
3375    X       rs5961828       5680521 9994078556      191     155
```

List Splines

This will generate a comprehensive list of all splines for each SNP listed. SNPs are listed by base-pair location.

ldspline list spline-filename DP/RS float chrom bp [bp...]

ldspline list ceu-b37 DP 0.8 3 122287289

This will generate a listing of a single spline associated with the locus 3:122287289 with a threshold of DPrime of 0.8.

In this example, we are using build 37 for the CEU population.

```
[torstees@queso data] ldspline list ceu-b37 DP 0.8 3 122287289 122265739
chrom   primary rs      primary pos     secondary rs    secondary pos   DP       dir
3       rs9839782       122287289       rs9917729       122285701       0.8070   down
3       rs9839782       122287289       rs12486285      122286160       1.0000   down
3       rs9839782       122287289       rs2332285       122288210       0.8010   up
3       rs7629285       122265739       rs13070816      122234028       1.0000   down
3       rs7629285       122265739       rs4485669       122234615       1.0000   down
                                     (truncated)
3       rs7629285       122265739       rs4677957       122257767       1.0000   down
3       rs7629285       122265739       rs28365795      122259640       1.0000   down
3       rs7629285       122265739       rs6780181       122264619       1.0000   down
3       rs7629285       122265739       rs17267388      122267546       1.0000   up
3       rs7629285       122265739       rs9289196       122268506       1.0000   up
                                     (truncated)
```

Reporting Boundaries

Users can query for upper and lower bounds for splines, similar to the **list** function except the application will report only the upper and lower bounds.

ldspline report-bounds spline-filename LD/RS float bp [bp][...]

ldspline report-bounds ceu-b37 DP 0.8 3 122287289 122265739

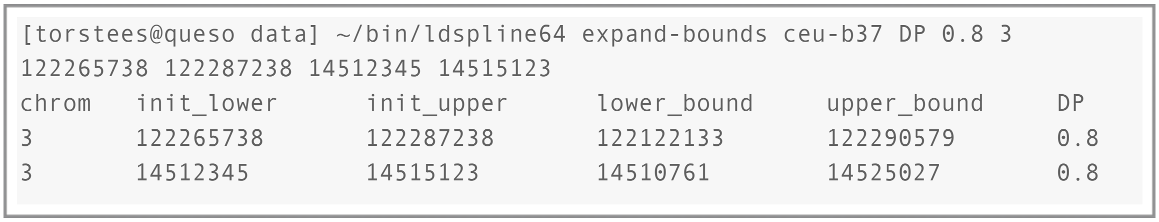
Executing this command will return the boundaries for two SNPs.

Expanding Boundaries

Users can query for boundary expansion using splines. The returned boundaries is the upper and lower bound for all SNPs (splines) found within the bounded region. This allows users to quickly identify the maximum region covered by LD for a given bounded region.

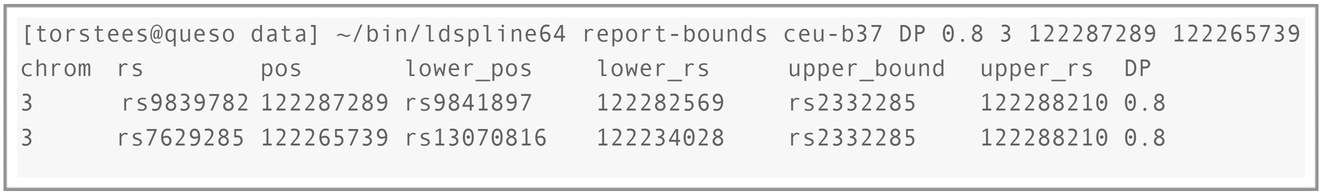
ldspline expand-bounds spline-filename DP/RS float chrom bp-lower bp-upper [bp-lower bp-upper][..]

ldspline expand-bounds ceu-b37 DP 0.8 3 122265738 122287238 14512345 14515123



```
[torstees@queso data] ~/bin/ldspline64 expand-bounds ceu-b37 DP 0.8 3
122265738 122287238 14512345 14515123
chrom   init_lower      init_upper      lower_bound     upper_bound     DP
3       122265738       122287238       122122133       122290579       0.8
3       14512345        14515123        14510761        14525027        0.8
```

Load



```
[torstees@queso data] ~/bin/ldspline64 report-bounds ceu-b37 DP 0.8 3 122287289 122265739
chrom  rs        pos       lower_pos     lower_rs     upper_bound   upper_rs  DP
3      rs9839782 122287289 rs9841897     122282569    rs2332285     122288210 0.8
3      rs7629285 122265739 rs13070816    122234028    rs2332285     122288210 0.8
```

Converting hapmap ld data into a binary spline is possible using the LOAD command. Users should download the Hapmap data and have it extracted. A "configuration" file is then created with two columns, the first naming the chromosome and the second indicating the file containing the ld data. Below is an example of the portion of our chb configuration:

```
1       ld_chr1_CHB.txt
2       ld_chr2_CHB.txt
3       ld_chr3_CHB.txt
4       ld_chr4_CHB.txt
5       ld_chr5_CHB.txt
6       ld_chr6_CHB.txt
7       ld_chr7_CHB.txt
8       ld_chr8_CHB.txt
9       ld_chr9_CHB.txt
10      ld_chr10_CHB.txt
(from the file, chb)
```

ldspline load filename [filename]...

ldspline load chb

This would parse the file, chb, converting the contents into an LD Spline binary file.

Converting Splines to a Different Genome Build

Users can import new base location data into splines using files generated by liftOver (http://hgdownload.cse.ucsc.edu/admin/exe/) In order to update a spline to a new build, users must perform 3 tasks:

ldspline export-lomap ldspline-filename

ldspline export-lomap ceu

This will generate a bim file compatible with the liftOver application. This will be named the same as the population with the extension of .bim added.

Users should run liftOver and specify the following parameters:

- population.new - as the new file
- population.unmapped - as the unMapped files

Where population is the name of the spline without the extension (i.e. ceu or ceu-b36)

Once the two new files have been generated, importing requires one more command:

ldspline import-lomap ldspline-filename new-build-number

```
ldspline import-lomap ceu 37
```

The previous example will import the new position information from the files, ceu.new and ceu.unmapped and name the new spline file, ceu-b37.ldspline.