# Requirements

OS

`Linux` or `OSX`

Dowload the repository

```
git clone https://github.com/zhenin/HDL.git
```

Compile Fortran functions

```
cd HDL
rm -f build_ld_ref/utils/bmult.o build_ld_ref/utils/ldscore.o \
  build_ld_ref/utils/bmult.so build_ld_ref/utils/ldscore.so
R CMD SHLIB build_ld_ref/utils/bmult.f90
R CMD SHLIB build_ld_ref/utils/ldscore.f90
```

Install R packages

```
install.packages(c('tidyr', 'dplyr', 'data.table', 'RSpectra', 'argparser'))
```

Install `HDL` (required for demo example)

```
Rscript HDL.install.R
```

# Guide

## Demo

Plink files of the demo example are generated from the data of [1000 Genomes Project](1000 Genomes Project).

```
bash build_ld_ref/run_demo.sh
```

# 1. Split chromosomes

**CAUTION** :

- `.bim` files of **ALL** chromsomes, of the LD reference panel, must be merged (`cat`) into a **SINGLE** `.bim` file.
- Variant identifiers (rsids) in the `.bim` file **MUST BE UNIQUE**.

```
Rscript build_ld_ref/1_split_chroms.R <ld_ref_path/ld_ref_name> <ALL_SNPS.bim>
--min MIN_AVG_NUM_SNPs --max MAX_AVG_NUM_SNPs
```

`--min` and `--max` options control the range of average number of variants in a segment.

## 2. Calculate LD

Prepare plink data: `bfile.bed + bfile.bim + bfile.fam`.

```
bash build_ld_ref/1_cal_ld.sh <path/to/bfile> <ld_ref_path/ld_ref_name>
[bandwidth [ld_window [chroms]]] | bash
```

Optional arguments:
`bandwidth` : bandwith (number of SNPs) for LD calculation, default= `500` .
`ld_window` : window size (kb) for LD calculation, default= `1000000` (whole segment).
`chroms` : selected chromosomes, separated by comma ( `,` ) ,
default= `1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22` .

- Or run parallely using `parallel` command

```
bash build_ld_ref/1_cal_ld.sh <path/to/bfile> <ld_ref_path/ld_ref_name> |
parallel -j n_cores
```

- Or run parallely by saving commands to a file, then spliting & submiting it to your server cluster **accordingly**

```
bash build_ld_ref/1_cal_ld.sh <path/to/bfile> <ld_ref_path/ld_ref_name> >
jobs.sh
```

## 3. Build LD reference

```
bash build_ld_ref/2_build_ld_ref.sh <ld_ref_path/ld_ref_name> | bash
```

Or run parallelly as **Step 2**.