Journal of
open research software

Petrowski, P F, et al. 2019 An R Framework for the Partitioning of Linkage Disequilibrium between and Within Populations. *Journal of Open Research Software,* 7: 15. DOI: https://doi.org/10.5334/jors.250

SOFTWARE METAPAPER

An R Framework for the Partitioning of Linkage Disequilibrium between and Within Populations

Paul F. Petrowski[1], Elizabeth G. King[1] and Timothy M. Beissinger[2]

[1] University of Missouri, Division of Biological Sciences, Columbia Missouri, US

[2] Georg-August-Universität Göttingen, Center for Integrated Breeding Research, Göttingen, DE

Corresponding author: Paul F. Petrowski (pfpetrowski@mail.missouri.edu)

Patterns of linkage disequilibrium (LD) across the genome result from a myriad of contributing factors including selection and genetic drift. Natural selection can increase LD near individually selected loci, or it can influence LD between epistatically selected groups of loci. Statistics have previously been derived which compare levels of linkage disequilibrium in subpopulations relative to the total population. These statistics may be leveraged to identify loci that may be under selection or epistatic selection. This is a powerful approach, but to date no framework exists to support its use on a genome-wide scale. We present `ohtadstats`, an R package designed to facilitate the implementation of Ohta's D statistics in a variety of use cases. Statistics calculated by this package can be used to determine whether a locus is under selection or not, and can provide insight into the nature of the selection that is taking place (hard sweep or epistatic selection). This package is available on the Comprehensive R Archive Network (CRAN).

Keywords: Linkage disequilibrium; selective sweep; population structure; epistasis; genome scan
Funding statement: This research was supported by funding from the USDA Agricultural Research Service. PFP is funded by the University of Missouri Life Sciences Fellowship and a training grant from the National Institute of Health (T32GM008396).

(1) Overview

Introduction

Pronounced signatures are left in the genomes of species undergoing selection. These telltale signals may reveal selected loci and details regarding the selection pressures that have been applied [1]. Notably, selection modifies the correlation of alleles at sites near the selected locus. This correlation between alleles is known as linkage disequilibrium (LD). As a rule of thumb, selection reduces genetic variability and increases LD [2]. Tomoka Ohta [3] derived a series of statistics (from here on referred to as Ohta's D statistics) designed to partition LD into between and within population components in a manner analogous to Sewall Wright's F statistics, a measure of inbreeding [4]. She posited that for a pair of loci, deviations from expected levels of LD in a given subpopulation may be indicative of an epistatic selection event. More recently, Beissinger et al. (2016) demonstrated that Ohta's D Statistics, particularly the D'^{2}_{IS} statistic, can identify traditional hard selective sweeps [2] in addition to epistatic selection, and they developed a null-distribution that enables the genome-wide identification of selection candidates.

Several studies have been conducted that utilize Ohta's D statistics to test for epistatic selection [5–7], which is selection acting on a favorable combination of loci, rather than a single locus independently. However, software suites for measuring Ohta's D statistics are limited. Programs that evaluate the statistics on a locus-by-locus basis exist [8–10], but there is currently no framework available to facilitate the implementation of Ohta's D statistics on a genome-wide basis. Furthermore, the web-based platform supplied by [9] and its predecessor [8] have two significantly limiting attributes that are resolved by our implementation. First, they require that input data be limited to no more than 80 SNPs in 30 or fewer subpopulations. Second, they treat sample size as population size, an approach that will only occasionally reflect reality.

Ohta's D statistics are computed in a pairwise fashion between markers, so evaluating even a relatively small marker set of a few hundred or thousands of SNPs requires an efficient implementation. Therefore, we have developed `ohtadstats`, a freely available R package with convenient, flexible, and powerful tools to perform the computation of Ohta's D statistics in a variety of use cases. By leveraging the R statistical software platform [11], `ohtadstats` is fast, scalable, and most importantly adaptable to an endless array of system architectures and high-throughput computing systems. Here, we describe the capabilities of the `ohtadstats` package and demonstrate its applicability to datasets both small scale and genome wide.

A Review of Ohta's D Statistics

Ohta's D statistics are a set of five statistics, termed D^2_{it}, D^2_{is}, D^2_{st}, D'^2_{is}, and D'^2_{st}. The specific forms of these statistics have been covered in depth by Ohta [3] so we will not go in depth here. Briefly, from Beissinger et al. [6]:

- D^2_{it} is the correlation of two alleles occurring on the same gamete in a subpopulation compared to the expectation of them occurring together in the total population
- D^2_{is} is the expected variance of LD for subpopulations
- D^2_{st} is the correlation of alleles in a subpopulation relative to their expected correlation in the total population
- D'^2_{is} is the correlation of the appearance of two alleles on the same gamete in a subpopulation relative to that of the total population
- D'^2_{st} is the variance of LD in the total population

Consider a comparison between two loci, A and B. Here, $x_{i,k}$ and $y_{j,k}$ are the frequencies of the i^{th} and j^{th} alleles at loci A and B in the k^{th} subpopulation, $g_{ij,k}$ is the frequency of gametes A_iB_j in the k^{th} subpopulation. Averages of these values are denoted with bars. These statistics may be calculated as follows:

$$D^2_{IT} = E\left\{\sum_{i,j}\left(g_{ij,k} - \bar{x}_i\,\bar{y}_j\right)^2\right\}$$

$$D^2_{IS} = E\left\{\sum_{i,j}\left(g_{ij,k} - x_{i,k}\,y_{j,k}\right)^2\right\}$$

$$D^2_{ST} = E\left\{\sum_{i,j}\left(\mathrm{x}_{i,k}\,y_{j,k} - \bar{\mathrm{x}}_i\,\bar{y}_j\right)^2\right\}$$

$$D'^2_{IS} = E\left\{\sum_{i,j}\left(g_{ij,k} - \bar{g}_{ij}\right)^2\right\}$$

$$D'^2_{ST} = E\left\{\sum_{i,j}\left(\bar{g}_{ij} - \bar{x}_i\,\bar{y}_j\right)^2\right\}$$

Implementation and architecture

The ohtadtats package includes five functions: `dstat`, `dwrapper`, `dheatmap`, `dparallel`, and `dfilter`. Detailed descriptions and example code can be found at https://github.com/pfpetrowski/OhtaDStats, as well as in the documentation of the R package.

The first of these functions, `dstat`, is the workhorse of the package. This function computes each of Ohta's D statistics for a given pair of loci, and returns these results in a vector. The `dstat` function also returns the number of subpopulations included in the analysis. This number may be less than the total number of subpopulations as a result of filtering. To avoid spurious associations between alleles that are not truly in LD, `dstat` has an initial allele frequency filtering step designed to remove loci that fall below a specified minor allele frequency threshold. During the filtering step, `dstat` also removes any subpopulations which have a minor allele frequency below a given threshold. This feature prevents subpopulations which are fixed or nearly fixed at a given locus from appearing to be under selection when in reality the effect is due to small sample size. Both of these minor allele frequency thresholds are modifiable arguments with can be changed by the user on demand. The `dstat` function returns a vector containing the number of populations included in the calculation and each of Ohta's five D statistics for the specified pair of markers. See **Figure 1** for an illustration of the dstat function.

It is important to note that the `dstat` function returns raw D statistic values. To assess statistical significance, `dstat` can be used to generate an empirical null distribution, as was used in [6]. Null distributions will vary by organism and model system, but the tools provided in `ohtadstats` can be used for their creation. This is achieved by implementing the function on a large number of pairs of physically unlinked SNPs.

The `dwrapper` function computes Ohta's D statistics for all possible pairs of loci in a matrix of genotypes (**Figure 2**). The result is returned as a list of matrices, with one matrix for each of Ohta's D statistics along with a matrix specifying the number of populations used for each comparison. This output format simplifies the process of looking up a D statistic for a specific pair of loci. It is important to note that because `dwrapper` evaluates all pairwise combinations of loci, it scales on the order of n^2, where n is the number of genetic markers represented in the dataset. This means that the number of pairwise comparisons to be made scales exponentially with the number of markers being evaluated. This is not a problem for small datasets. Indeed, we successfully executed the `dwrapper` function on a dataset that included 100 SNPs from 1417 individuals using a 1.3GHz, 8GB RAM MacBook Air in only two minutes. This dataset is a subset of the data used by Beissinger et al. (2016) [6] and is included in

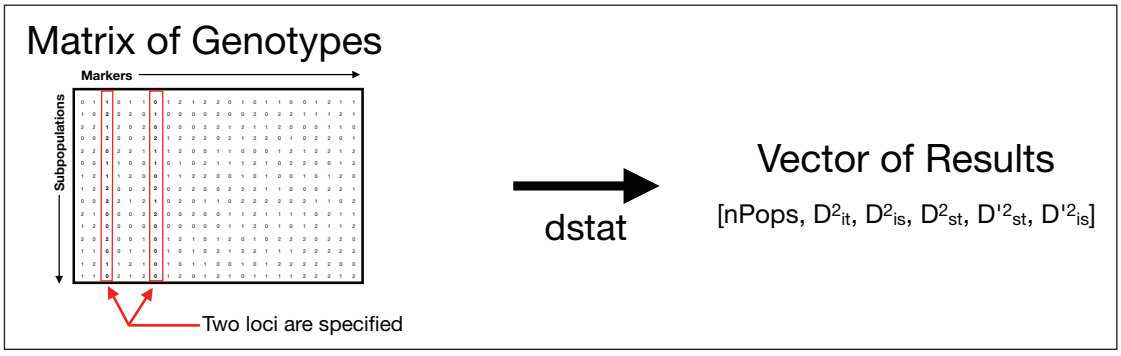

Figure 1: The `dstat` function calculates Ohta's D statistics for two specified loci.

the package as beissinger_data.rda. For large datasets (ie thousands of markers), we suggest parallelization via the `dparallel` function.

The `dheatmap` function provides a convenient tool for data visualization. This function, which is based on the levelplot function from the lattice R package [12] takes any of the matrix outputs provided by the `dwrapper` function and returns a colorized heat map, making patterns of LD visible. Options are provided which allow the user to modify the colors used and how those colors are scaled. The `dheatmap` function provides three modes for the scaling of the colors. The first mode, "linear", is appropriate for most use cases. In this mode, the values in the matrix are distributed continuously across the color spectrum. The "truncated" mode is provided for use on the ratio matrices, where division by small numbers may cause certain values to be many orders of magnitude greater than others. In these cases, using "linear" is not ideal because large values will drive mid-magnitude values towards the extreme low end of the color spectrum. The "truncated" mode corrects for this issue by changing values greater than one to a value just higher than one. Colors are then scaled across this new spectrum of values. The final mode, "binned", operates similarly to "truncated", except that colors are not scaled across the new spectrum of values. Instead, values are placed into one of five bins, and colored accordingly.

The `dparallel` function is designed to facilitate parallelization of `dstat` on commercial high throughput computing platforms. By pairing a simple R script with a scheduler such as slurm [13], massive datasets on the scale commonly seen today can be analyzed in a reasonable amount of time. This function works by generating a virtual table of locus pairs to compare and executing dstat on each. Using this method, a number of equally sized jobs limited only by time and computational resources can be performed. The `dparallel` function is not meant to be used directly in the R environment. Instead it is designed to be set up in an R script, multiple instances of which are then spawned using a slurm scheduler array, or equivalent (**Figure 3**). Example scripts of this setup are available in

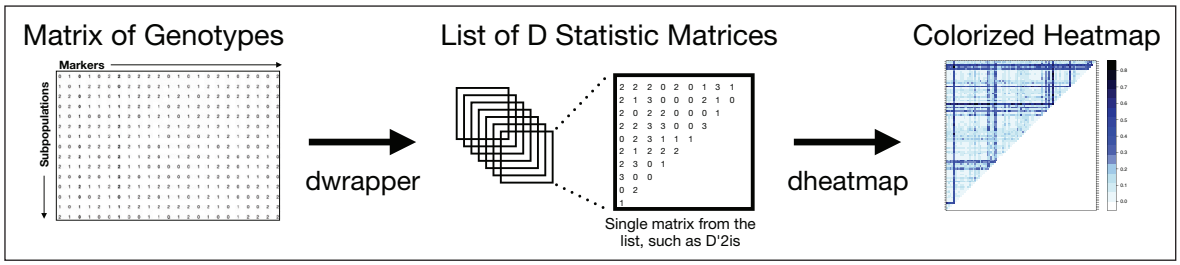

Figure 2: Given a single matrix of genotypes, `dwrapper` will produce a matrix for each of Ohta's D statistics with pairwise comparisons of each locus. `Dheatmap` will produce a colorized map for visualization of the values in a matrix.

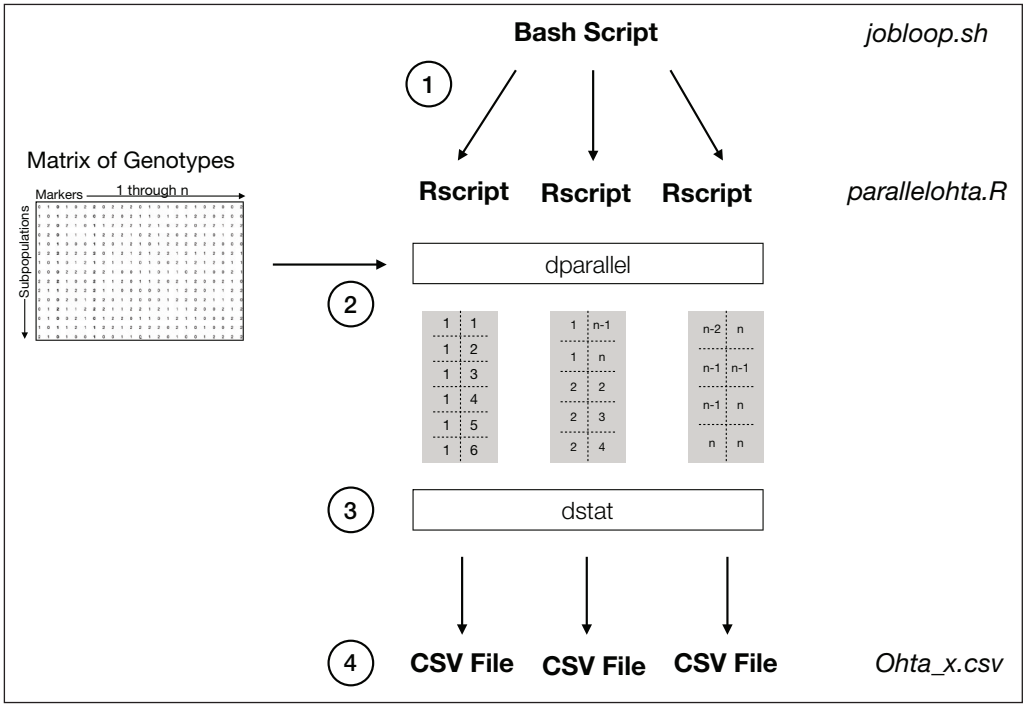

Figure 3: The `dparallel` workflow. Specific files associated with the general steps are in italics to the right of the diagram. 1) A bash script initializes a specified number of R processes, each of which executes the `dparallel` function on the specified dataset. 2) Each `dparallel` process infers a unique set of locus pairs for which to compute Ohta's D statistics. 3) Each R process calls `dstat` on each row (pair of loci) in its unique set. 4) Results are returned in csv files.

the OhtaDStats GitHub repository (https://github.com/pfpetrowski/OhtaDStats).

Lastly, the `dfilter` function is provided to perform the basic data preprocessing step of removing any subpopulations from the dataset if that subpopulation is too small. The `dfilter` function works by taking a dataset and an integer value for the minimum number of samples, and returning that dataset with only subpopulations that meet the threshold. Similar to the minor allele frequency filtering that is performed within the `dstat` function, this mitigates the danger of small sample sizes leading to spuriously large values of D.

Quality Control

To ensure that this package accurately calculates Ohta's D statistics, We simulated a small dataset containing 18 individuals across 3 subpopulations and three loci. We evaluated this data set using an implementation of LinkDOS [9] available at Genepop on the Web [14], and also using the `ohtadstats` package to ensure that results were equivalent. The sample dataset and code are available in the GitHub repository. In addition, examples included in this package are tested daily on the CRAN servers across Windows, MacOS, and Unix operating systems.

(2) Availability

Operating system

The `ohtadstats` package is designed for use with R versions 3.0.0 or later. R is supported on Windows, MacOS, and major Linux distributions. Minimum operating system versions are as follows:

Windows: Windows 7
MacOS: MacOS 10.9 (Mavericks)
Ubuntu: 14.04 (Trusty)

Programming language

R

Additional system requirements

R requires that 150 MB of disk space be available for installation.

Dependencies

Requires the "lattice" and "grDevices" R packages. We also require R version 3.0.0 or later for this package.

List of contributors

Paul F. Petrowski, Timothy M. Beissinger, Elizabeth G. King

Software location

Archive

Name: ohtadstats
Persistent identifier: https://doi.org/10.5281/zenodo.1406484
Licence: MIT
Publisher: Paul Petrowski
Version published: 2.1.1
Date published: 20/03/19

Code repository

Name: OhtaDStats
Identifier: https://github.com/pfpetrowski/OhtaDStats
Licence: MIT
Date published: 18/03/19

Language

English

(3) Reuse potential

Ohta's D statistics are useful quantities for assessing linkage disequilibrium in genomic data sets. As such, this package may be useful to anyone looking to quantify linkage disequilibrium in their system of study. This includes any individual investigating the fields of population, quantitative, or evolutionary genetics. A typical use case may involve looking across a number of subpopulations of a species in an effort to detect evidence of selection. Other methods of using LD as a measurement of selection have been previously described, including the integrated haplotype score (iHS) [15] and extended haplotype homozygosity (EHH) [16]. These commonly-applied methods are designed to identify hard sweeps in a single population, while Ohta's D statistics are best applied to data including multiple populations, and have the potential to additionally identify epistatic selection. Therefore, these approaches may be complementary – researchers may find different selected loci based on Ohta's D stats than by applying iHS or EHH, and vice versa.

Acknowledgements

We would like to thank Jake Gotberg from the Mizzou Research Computing Support Service for contributing his time and expertise in setting up an efficient parallelization workflow. Computation for this work was performed on the high performance computing infrastructure provided by Research Computing Support Services and in part by the National Science Foundation under grant number CNS-1429294 at the University of Missouri, Columbia MO.

Competing Interests

The authors have no competing interests to declare.

References

1. **Vitti, J J, Grossman, S R** and **Sabeti, P C** 2013 Detecting Natural Selection in Genomic Data. *Annual Review of Genetics*, 47(1): 97–120. DOI: https://doi.org/10.1146/annurev-genet-111212-133526
2. **Maynard, J** and **Haigh, J** 1974 The hitch-hiking effect of a favourable gene. *Genetics Research*, 89(5–6): 391–403. DOI: https://doi.org/10.1017/S0016672308009579
3. **Ohta, T** 1982 Linkage disequilibrium due to random genetic drift in finite subdivided populations. *Proceedings of the National Academy of Sciences of the United States of America*, 79(6): 1940–1944. DOI: https://doi.org/10.1073/pnas.79.6.1940

4. **Wright, S** 1922 Coefficients of Inbreeding and Relationship. *The American Naturalist*, 56(645): 330–338. DOI: https://doi.org/10.1086/279872
5. **Miyashita, N T, Aguadé, M** and **Langley, C H** 1993 Linkage disequilibrium in the white locus region of Drosophila melanogaster. *Genetical Research*, 62(2): 101. DOI: https://doi.org/10.1017/S0016672300031694
6. **Beissinger, T M, Gholami, M, Erbe, M,** et al. 2016 Using the variability of linkage disequilibrium between subpopulations to infer sweeps and epistatic selection in a diverse panel of chickens. *Heredity*, 116(2): 158–166. DOI: https://doi.org/10.1038/hdy.2015.81
7. **Song, B-H, Windsor, A J, Schmid, K J,** et al. 2009 Multilocus Patterns of Nucleotide Diversity, Population Structure and Linkage Disequilibrium in Boechera stricta, a Wild Relative of Arabidopsis. *Genetics*, 181(3): 1021–1033. DOI: https://doi.org/10.1534/genetics.108.095364
8. **Black, W C** and **Krafsur, E S** 1985 A FORTRAN program for the calculation and analysis of two-locus linkage disequilibrium coefficients. *Theoretical and Applied Genetics*, 70(5): 491–496. DOI: https://doi.org/10.1007/BF00305981
9. **Garnier-Gere, P** and **Dillmann, C** 1992 A Computer Program for Testing Pairwise Linkage Disequilibria in Subdivided Populations. *Journal of Heredity*, 83(3): 239–239. DOI: https://doi.org/10.1093/oxfordjournals.jhered.a111204
10. **Yeh, F** 1997 Population genetic analysis of co-dominant and dominant marker and quantitative traits. *Belgian J Bot*, 130: 129–157.
11. **R Core Team** 2018 R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing.
12. **Sarkar, D** 2008 Lattice: Multivariate Data Visualization with R. New York. Springer. DOI: https://doi.org/10.1007/978-0-387-75969-2
13. **Jette, M A, Yoo, A B** and **Grondona, M** 2002 SLURM: Simple Linux Utility for Resource Management. In: *Lect. Notes Comput. Sci. Proc. Job Sched. Strateg. Parallel Process*, 44–60. JSSPP 2003. Springer-Verlag.
14. **Raymond, M** and **Rousset, F** 1995 GENEPOP (Version 1.2): Population Genetics Software for Exact Tests and Ecumenicism. *Journal of Heredity*, 86(3): 248–249. DOI: https://doi.org/10.1093/oxfordjournals.jhered.a111573
15. **Voight, B F, Kudaravalli, S, Wen, X** and **Pritchard, J K** 2006 A Map of Recent Positive Selection in the Human Genome. *PLOS Biology*, 4(3): e72. DOI: https://doi.org/10.1371/journal.pbio.0040072
16. **Sabeti, P C, Reich, D E, Higgins, J M,** et al. 2002 Detecting recent positive selection in the human genome from haplotype structure. *Nature*, 419(6909): 832–837. DOI: https://doi.org/10.1038/nature01140

How to cite this article: Petrowski, P F, King, E G and Beissinger, T M 2019 An R Framework for the Partitioning of Linkage Disequilibrium between and Within Populations. *Journal of Open Research Software,* 7: 15. DOI: https://doi.org/10.5334/jors.250

Submitted: 24 October 2018 **Accepted:** 02 April 2019 **Published:** 30 April 2019