hogwash: Three Methods for Genome-Wide Association Studies in Bacteria

Authors
Katie Saund[1] (0000-0002-6214-6713) and Evan S. Snitkin[1,2] (0000-0001-8409-278X)

Affiliations
[1]Department of Microbiology and Immunology
[2]Department of Internal Medicine/Division of Infectious Diseases
University of Michigan, Ann Arbor, Michigan

Corresponding Author
Evan S. Snitkin, esnitkin@med.umich.edu

Keywords
GWAS, bacterial genomics, convergent evolution, software

ABSTRACT
Bacterial genome-wide association studies (bGWAS) capture associations between genomic variation and phenotypic variation. Convergence based bGWAS methods identify genomic mutations that arise more often in the presence of phenotypic variation than is expected by chance. This work introduces hogwash, an open source R package that implements three algorithms for convergence based bGWAS. Hogwash additionally contains a novel grouping tool to perform gene- or pathway-analysis to improve power and increase convergence detection for related but weakly penetrant genotypes. To identify optimal use cases we applied hogwash to data simulated with a variety of phylogenetic signals and convergence distributions. These simulated data are publicly available and contain the relevant metadata regarding convergence and phylogenetic signal for each phenotype and genotype. Hogwash is available for download from GitHub.

DATA SUMMARY
1. hogwash is available from GitHub under the MIT license (https://github.com/katiesaund/hogwash) and can be installed using the R command `devtools::install_github("katiesaund/hogwash")`
2. The simulated data used in this manuscript and the code to generate it are available from GitHub (https://github.com/katiesaund/simulate_data_for_convergence_based_bGWAS)

IMPACT STATEMENT
We introduce hogwash, an R package with three methods for bacterial genome-wide association studies. There are two methods for handling binary phenotypes, including an implementation of PhyC(1), as well as one method for handling continuous phenotypes. We formulate two novel indices quantifying the relationship between phenotype convergence and genotype convergence on a phylogenetic tree, one for binary phenotypes and one for continuous phenotypes. These indices shape an intuitive understanding for the ability of hogwash to detect significant intersections of phenotype convergence and genotype convergence and how to interpret hogwash outputs.

INTRODUCTION
Bacterial Genome-Wide Association Studies
Bacterial genome-wide association studies (bGWAS) infer statistical associations between genotypes and phenotypes. Seminal bGWAS papers identified novel variants associated with

antibiotic resistance in *M. tuberculosis* and host specificity in *Campylobacter*(1,2). Since then, there have been numerous applications of bGWAS that have further highlighted the potential of this approach to identify genetic pathways underlying phenotypic variation and provide insights into the evolution of phenotypes of interest. Association studies can use various genetic data types including single nucleotide polymorphisms (SNPs), k-mers, copy number variants, accessory genes, insertions, and deletions. To improve the power and interpretability of bGWAS inclusion criteria or weighting can be applied to these variants based on predicted functional impact, membership in pathways of interest, or other user preferences(3,4). Differences between human and bacterial GWAS have been reviewed extensively by Power *et al.*(5). Of note, clonality and horizontal gene transfer complicate the application of human GWAS methodology to bacteria. However, certain bGWAS approaches can leverage unique features of bacterial evolution, including frequent phenotypic convergence and genotypic convergence, to identify phenotype-genotype correlations.

bGWAS Software

bGWAS methods can be classified into non-exclusive groups based on some critical features: A) methods for SNPs, accessory genes(6), or k-mers(7), B) methods using regression(7,8) or phylogenetic convergence(1,9), and C) methods designed for humans(10) or specifically for bacteria(7,9). Differences between regression based and convergence based bGWAS were expertly reviewed by Chen and Shapiro(11). Convergence based methods identify multiple independent events where a genomic mutation arises more often in the presence of the phenotype of interest. Convergence based methods can yield higher significance with a smaller sample size, but may fail to identify some statistical associations that traditional GWAS approaches would identify when the population is clonal(11). Additionally, convergence based methods are limited to smaller data sets because of their large memory requirements and computational time relative to traditional methods(12), but can surmount issues of clonality and take advantage of horizontal gene transfer.

Objective

This work describes hogwash, an R package that implements three different convergence based bGWAS approaches available on GitHub. Two approaches, PhyC and the Synchronous Test, handle binary phenotypes while the third approach, the Continuous Test, handles continuous phenotypes. PhyC is an algorithm introduced by Farhat *et al.*(1) that we implement. The Synchronous Test is a stringent variation of PhyC, requiring a tighter relationship between the genotype and phenotype. We describe the algorithms and evaluate them on a set of simulated data.

Alternative Approach to Grouped Genotype Analysis

Pathway analysis is a common post-GWAS approach that groups loci into meaningful groups, such as mapping SNPS to pathways(13,14). Analyzing aggregated loci can improve both the interpretability of GWAS results and improve power to detect associations(13,14). However, post-GWAS pathway analysis may not surmount the high stringency of convergence based bGWAS approaches. Hogwash implements a novel grouping tool prior to performing the bGWAS that may avoid this potential loss of information and improve convergence detection for related but weakly penetrant genotypes.

Data Simulation

We evaluate hogwash results on simulated data generated to capture aspects of bacterial evolution pertinent to these bGWAS approaches. We simulated data with a range of phylogenetic signals and convergence distributions to highlight the critical impact of these features on bGWAS results. The simulated data are publicly available and could be used to

compare the impact of convergence patterns within phenotypes, genotypes, and their intersection when benchmarking various convergence based bGWAS methods.

PACKAGE DESCRIPTION

We developed hogwash to allow users to perform three bGWAS methods, including an open source implementation of the previously described PhyC algorithm(1), and aggregate genotypes by user-defined groups. The hogwash function minimally requires a phenotype, a phylogenetic tree, and a binary genotype matrix. An optional argument may be supplied to facilitate grouping genotypes. The genotype matrix and tree can be prepared from a multiVCF file by the variant preprocessing tool prewas(15). Hogwash assumes that the genotype is encoded such that 0 refers to wild type and 1 refers to a mutation and that binary phenotypes are encoded such that 0 refers to absence and 1 refers to presence.

In brief, the hogwash workflow (Figure 1A) begins with the user supplying a phenotype, a set of genotypes, and a tree. Hogwash performs ancestral state reconstruction for the phenotype and genotypes. If the user supplies a key to group together genotypes, hogwash groups them after the genotype ancestral state reconstructions. The convergence of each phenotype and each genotype are recorded as the edges where they intersect on the tree (Figure 1B); the definition of convergence and intersection is unique for each of the three association tests. Then the genotype is permuted, and its intersection with the phenotype is recorded as a null distribution. Significance is calculated with correction for multiple testing.

PhyC

PhyC is a convergence based bGWAS method introduced by Farhat *et al.*(1) that identified novel antibiotic resistance-conferring mutations in *M. tuberculosis*. To our knowledge, the original PhyC code is not publicly available, but the algorithm is well described in the original paper. The algorithm addresses the following question: Does the genotype transition from wild type, 0, to mutant, 1, occur more often than expected by chance on tree edges where the phenotype is present, 1, than where the phenotype is absent, 0? By requiring the overlap of the phenotype with the genotype transition, instead of genotype presence, associations are not inflated by clonal sampling and thus this approach controls for population structure. We implement the PhyC algorithm as described in Farhat *et al.*(1).

For each test we formulate the following terms: $\beta_{genotype}$ and $\beta_{phenotype}$. In PhyC (Figure 2) $\beta_{genotype}$ is a non-negative integer that records the number of tree edges where the genotype arises (mutation appears). This is encoded as a tree edge where the parent node is evaluated as 0 by ancestral state reconstruction and child node is evaluated as 1. These edges are called genotype transitions. In PhyC $\beta_{phenotype}$ is a non-negative integer that records the number of tree edges where the phenotype is present. This is encoded as a tree edge where the child node is evaluated as 1 by ancestral state reconstruction. The number of edges on the tree where both a genotype arises and the phenotype is present is calculated as $\beta_{genotype} \cap \beta_{phenotype}$.

For the permutation the genotype mutations ($\beta_{genotype}$) are randomized on the tree. The number of edges where the permuted genotype mutation intersects with phenotype presence edges is recorded for each permutation; these permuted $\beta_{genotype} \cap \beta_{phenotype}$ values create a null distribution. An empirical *P*-value is calculated based on the observed $\beta_{genotype} \cap \beta_{phenotype}$ as compared to the null distribution.

Our PhyC implementation has several important differences from the original paper. First, multiple test correction in hogwash is performed with False Discovery Rate instead of the more stringent Bonferroni correction. Second, hogwash reduces the multiple testing burden by testing only those genotype-phenotype pairs for which convergence is detectable; genotypes with $\beta_{genotype} < 2$ are excluded and genotype-phenotype pairs with $\beta_{genotype} \cap \beta_{phenotype} < 2$ are assigned a *P*-value of 1. Third, ancestral state reconstruction for genotypes and phenotypes was performed using only maximum likelihood. Finally, users only supply one phylogenetic tree instead of three.

Synchronous Test

This test (Figure 2) is an extension of PhyC but requires more stringent association between the genotype and phenotype. The Synchronous Test addresses the question: Do genotype transitions occur more often than expected by chance on phenotype transition edges than on phenotype non-transition edges? Both PhyC and the Synchronous Test are only appropriate for binary phenotypes.

The Synchronous Test $\beta_{genotype}$ is a non-negative integer that records the number of tree edges where the genotype changes (mutation appears or disappears). This is encoded as a tree edge where the parent node value as inferred from ancestral state reconstruction is different than the child node. These edges are called genotype transitions. The Synchronous Test $\beta_{phenotype}$ is a non-negative integer that records the number of tree edges where the phenotype changes. This is encoded as a tree edge where the phenotype parent node is different than the child node as inferred from ancestral state reconstruction. The number of edges on the tree where both a genotype transitions and the phenotype transitions is calculated as $\beta_{genotype} \cap \beta_{phenotype}$. As in PhyC, the genotypes with $\beta_{genotype} < 2$ are removed, genotype-phenotype pairs with $\beta_{genotype} \cap \beta_{phenotype} < 2$ are assigned a *P*-value of 1,and the remaining genotypes are permuted and a null distribution of the $\beta_{genotype} \cap \beta_{phenotype}$ is calculated to determine the significance of each genotype.

This test is similar to the Simultaneous Score in treeWAS(9). The Simultaneous Score is derived from the number of edges on the tree where the genotype and phenotype transition in the same direction (both have an inferred parent node of 0 and an inferred child node of 1 or parent node of 0 and child node of 1). In contrast, the Synchronous Test in hogwash allows for the phenotype and genotype transition directions to mismatch, thus allowing for the detection of genotypes with inconsistent effect directions.

Continuous Test

The Continuous Test (Figure 2) is an application of a convergence based GWAS method to continuous phenotypes. The Continuous Test addresses the question: Does the phenotype change more than expected by chance on genotype transition edges than on genotype non-transition edges?

The Continuous Test $\beta_{genotype}$ is a non-negative integer that records the number of tree edges where the genotype changes (mutation appears or disappears). This is encoded as a tree edge where the parent node value as inferred from ancestral state reconstruction is different than the child node (Figure 1B). These edges are called genotype transitions.

Formula 1.

$$\Delta_{edge} = |phenotype_{parent\ node} - phenotype_{child\ node}|$$

A Δ_{edge} value is calculated for each tree edge and is scaled from 0 to 1. The Continuous Test $\beta_{phenotype}$ is the sum of all Δ_{edge} values that occur on high confidence edges. $\beta_{genotype} \cap \beta_{phenotype}$ is the multiplicative sum of the Δ_{edge} and $\beta_{genotype}$. As above, the genotypes with $\beta_{genotype} < 2$ are removed; the remaining genotypes are permuted and a null distribution of the $\beta_{genotype} \cap \beta_{phenotype}$ is calculated to determine the significance of each genotype.

User inputs

The user must provide a phylogenetic tree, genotype matrix, and a phenotype. The user may optionally provide a key that maps individual genomic loci into groups in order to use hogwash's grouping feature. For a detailed description of the user inputs please see the Supplementary Package Description.

Hogwash outputs

The package produces two files per test: data (.rda) and plots (.pdf). The data file contains many pieces of information, including *P*-values for each tested genotype. The plots are described below in the results section.

Grouping feature

To identify an association between a genomic variant and a phenotype, hogwash requires that a variant occur in multiple different lineages. Hogwash may classify some causal variants as independent of a phenotype if they are weakly penetrant, an issue common to convergence based methods. To surmount this issue, related genomic variants may be aggregated to capture larger trends at the grouped level. For example, a user may apply this method to group only nonsynonymous SNPs by gene to use hogwash to detect associations between the mutated gene and the phenotype. Grouping related variants can improve power through a reduction in the multiple testing correction penalty. However, the power benefits are dependent on grouping variants with similar effect directions.

Hogwash implements the grouping features by first performing ancestral state reconstruction for each individual locus (Figure 3). If the user supplies a key that maps individual loci to groups, then the edges contributing to $\beta_{genotype}$ for individual loci are joined together into the indicated group. Grouped loci with $\beta_{genotype} < 2$ are excluded from analysis. After this point hogwash runs as previously described for non-grouped genotypes.

Users may supply hogwash with data that was previously grouped (for example, using the group SNPs by gene functionality in prewas(15)) but this approach may mask some genotype transitions. In this case, the user does not need to provide a key and the hogwash grouping step is skipped.

METHODS

Data simulation

Trees

We simulated four random coalescent phylogenetic trees with 100 tips each.

Phenotypes

For each tree we simulated phenotypes that model either Brownian motion or white noise. A phenotype modeled well with white noise may suggest a role for horizontal gene transfer, gene loss, or convergent evolution(16). A white noise phenotype may be better suited to the hogwash algorithms than a phenotype modeled by Brownian motion given the requirement for phylogenetic convergence. A continuous phenotype that is modeled well by Brownian motion

has a phylogenetic signal, λ, near 1 while a white noise phenotype has a phylogenetic signal near 0(17). In contrast, a binary phenotype that is modeled well by Brownian motion has a phylogenetic signal, D statistic, near 0 while a white noise phenotype has a phylogenetic signal near 1(18). For each tree we simulated sixteen phenotypes: eight phenotypes with a phylogenetic signal fitting a Brownian motion model (four binary and four continuous), and eight phenotypes with a phylogenetic signal fitting a white noise model (four binary and four continuous). For phenotypes modeling Brownian motion, binary phenotypes were restricted to $-0.05 < D < 0.05$ and continuous phenotypes to $0.95 < \lambda < 1.05$. For phenotypes modeling white noise, binary phenotypes were restricted to $0.95 < D < 1.05$ and continuous phenotypes to $-0.05 < \lambda < 0.05$.

Genotypes

For each simulated tree a set of unique binary genotypes were generated. We generated genotypes that span a range of phylogenetic signals, degree of similarity to the phenotype, and prevalence.

<u>Genotypes to be used in discrete hogwash tests</u>

First, 25,000 binary genotypes were generated using ape::rTraitDisc; these genotypes have a range of phylogenetic signals(19). Second, these genotypes were duplicated and randomized with the following approach: one quarter had 10% of tips changed, one quarter had 25% of tips changed, one quarter had 40% of tips changed, and one quarter were entirely redistributed. Third, we removed any simulated genotypes present in 0, 1, $N-1$, or N samples. Fourth, we subset the genotypes to keep only unique presence/absence patterns. Fifth, we subset genotypes to only those within a range of $-1.5 < D < 1.5$. These filtering steps result in a reduction in the data set size (range 2214-2334).

<u>Genotypes to be in used in the Continuous Test</u>

In addition to the five steps above we added two more data generation steps. First, we made all possible genotypes based on the rank of the continuous phenotype. Second, we made genotypes based on which edges of the tree had high Δ_{edge}. The filtering steps reduced the data set size (range 1234-1310).

Hogwash on simulated data

We ran each hogwash test for each of the tree-phenotype-genotype sets. In addition to generating *P*-values for each tested genotype, hogwash also reports convergence information. We ran hogwash with the following settings: permutations = 50,000; false discovery rate = 0.0005 (discrete), 0.05 (continuous); bootstrap value = 0.70; no genotype grouping key was provided.

Calculation of $\beta_{genotype}$, $\beta_{phenotype}$ and ε

Using the ancestral state reconstruction data, hogwash identified convergence within each genotype ($\beta_{genotype}$), phenotype ($\beta_{phenotype}$), and their weighted intersection (ε).

Formula 2.

$$\varepsilon_{binary} = \frac{2 \times (\beta_{genotype} \cap \beta_{phenotype})}{\beta_{genotype} + \beta_{phenotype}}$$

Where $0 \leq \beta_{genotype} \leq Number\ tree\ tips$ and $0 \leq \beta_{phenotype} \leq Number\ tree\ tips$; both $\beta_{genotype}$ and $\beta_{phenotype}$ are integers. Therefore, $0 \leq \varepsilon \leq 1$.

Formula 3.

$$\varepsilon_{continuous} = \frac{\beta_{genotype} \cap \beta_{phenotype}}{\beta_{genotype} + \beta_{phenotype} - \beta_{genotype} \cap \beta_{phenotype}}$$

Where $0 \leq \beta_{genotype} \leq Number\ tree\ tips$ and $0 \leq \beta_{phenotype} < Number\ tree\ tips$. $\beta_{genotype} \cap \beta_{phenotype}$ is the multiplicative sum of the Δ_{edge} and $\beta_{genotype}$. The denominator is $\beta_{genotype} \cup \beta_{phenotype}$. Therefore, $0 \leq \varepsilon \leq 1$.

Data analysis

Statistical analyses were conducted in R v3.6.2(20). The R packages used can be found in the simulate_data.yaml file on GitHub(19,21–25) and can be installed using miniconda(26).

RESULTS

Hogwash output for simulated data

Hogwash outputs two sets of results: a data file and a PDF file with plots. Each run of PhyC produces at least three plots: the phenotype reconstruction (Figure 4A), a Manhattan plot (Figure 4D), and a heatmap of all tested genotypes (Figure 4E). The phenotype reconstruction, also referred to as $\beta_{phenotype}$, is highlighted on the tree (Figure 4A). The Manhattan plot shows the distribution of *P*-values from the hogwash run (Figure 4D). Lastly, the heatmap shows the genotype reconstruction ($\beta_{genotype}$) and phenotype reconstruction ($\beta_{phenotype}$) for each tree edge (rows) and genotype (columns) (Figure 4E). The genotypes are clustered by the $\beta_{genotype}$ presence/absence pattern. Two additional plots are produced for each genotype that is significantly associated with the phenotype: a phylogenetic tree showing the genotype transition edges (Figure 4B) and the null distribution of $\beta_{genotype} \cap \beta_{phenotype}$ (Figure 4C).

The Synchronous Test and Continuous Test output plots that reflect their test-specific $\beta_{genotype}$ and $\beta_{phenotype}$ definitions (Figure S1, S2). Running hogwash on 100 samples required <3 hours and <2 GB of memory for binary data and <5 hours and <2 GB of memory for continuous data (Figure S3).

Hogwash evaluation on simulated data

To help users identify optimal use cases and also interpret hogwash results we describe the behavior of hogwash on simulated data. We note that this assessment is not meant to convey performance in the sense of calculating sensitivity and specificity, but rather evaluate whether hogwash can robustly detect the association between phenotypic and genotypic convergence. To guide our assessment, we compared the relationship between the *P*-value and ε values produced by hogwash on sets of simulated data constructed using different evolutionary models (Figure 5). ε is a quantification of the relationship between phenotype convergence and genotype convergence; we define ε for the discrete and continuous tests in Formulae 2 and 3, respectively. Low ε values indicate little to no intersection of phenotype convergence and genotype convergence, while higher ε values indicate their increased intersection.

For discrete phenotypes, we observe an overall strong positive association between -log(*P*-value) and ε, demonstrating that as the intersection of phenotype convergence and genotype convergence increase hogwash predicts that it is less likely that they intersect due to chance (Table 1). In other words, below a certain ε_{binary} threshold, hogwash attributes the association between the genotype convergence and phenotype convergence to chance; from Figure 5 the user can get a sense for the range of this ε_{binary} threshold under different evolutionary regimes.

For the simulated continuous data an $\varepsilon_{continuous}$ threshold that separates meaningful genotype-phenotype associations from associations by chance is less apparent. Higher ε, low significance values demonstrate that some overlap of $\beta_{genotype}$ and $\beta_{phenotype}$ is likely by chance given the data. Low ε, high significance values demonstrate that some values with even small amounts of $\beta_{genotype}$ and $\beta_{phenotype}$ overlap are unlikely, however that does not necessarily suggest that these hits are the best candidates for *in vitro* follow up. We suspect that these associations are largely driven by poor exploration of the sampling space, despite running many permutations, because of the edge-length based sampling probability of the permutation method. Therefore, it is essential *P*-values be interpreted within the context of ε. Notably, the Continuous Test was only able to detect significant genotype-phenotype associations for phenotypes modeled by white noise, suggesting this method is particularly sensitive to the phenotype's evolutionary model.

We observe for both the discrete and continuous tests that ε is more tightly correlated with -log(*P*-value) for phenotypes characterized by white noise than by Brownian motion (Table 1), indicating that hogwash performs better under a white noise model. Thus, to help the user in assessing the appropriateness of hogwash and in interpreting their results we allow users to check if their phenotype is better modeled by white noise than Brownian motion by using the report_phylogenetic_signal function.

DISCUSSION

We have developed hogwash, an open-source R package that implements three different approaches to bGWAS and includes the previously described PhyC algorithm(1). Hogwash also introduces a novel grouping feature to aggregate related genomic variants to increase detection of convergence for weakly penetrant genotypes. Hogwash is best used for datasets comprising binary and/or continuous phenotypes, phenotypes fitting white noise models, situations where convergence may occur at the level of genes or pathways and with datasets whose size can be accommodated given the time and memory constraints of convergence methods.

The results of running hogwash on simulated data suggest that after a certain ε threshold, it unlikely that the intersection between phenotype convergence and genotype convergence occurs by chance, particularly for white noise phenotypes. Given the variability in results within each method, as shown in Figure 5, users may want to contextualize the statistical significance of the tested genetic loci with the amount of convergence possible for any one particular data set; to facilitate this the hogwash output includes both *P*-values and ε.

The simulated data set presented here is published to serve as a resource or template for future work focused on benchmarking convergence based bGWAS software as such a dataset has not yet, as far as we are aware, been made available(27). The simulated data set is available on GitHub and includes convergence information for each phenotype, genotype, and their intersection.

AUTHOR CONTRIBUTIONS

KS and ESS conceptualized the project and edited the manuscript. KS designed and implemented the software, performed the analysis, prepared the original draft, and visualized the data. ESS supervised the project.

CONFLICTS OF INTEREST

The authors declare that there are no conflicts of interest.

FUNDING

KS was supported by the National Institutes of Health (T32GM007544). ESS and KS were supported by the National Institutes of Health (1U01AI124255).

ACKNOWLEDGEMENTS

We thank Brad Saund for his help formalizing the continuous algorithm ε definition.

REFERENCES

1. Farhat MR, Shapiro BJ, Kieser KJ, Sultana R, Jacobson KR, Victor TC, et al. Genomic analysis identifies targets of convergent positive selection in drug-resistant Mycobacterium tuberculosis. Nat Genet. 2013;45(10):1183–9.
2. Sheppard SK, Didelot X, Meric G, Torralbo A, Jolley KA, Kelly DJ, et al. Genome-wide association study identifies vitamin B 5 biosynthesis as a host specificity factor in Campylobacter. PNAS. 2013;110(29):11923–7.
3. Sveinbjornsson G, Albrechtsen A, Zink F, Gudjonsson SA, Oddson A, Másson G, et al. Weighting sequence variants based on their annotation increases power of whole-genome association studies. Nat Genet. 2016 Mar 1;48(3):314–7.
4. Hendricks AE, Bochukova EG, Marenne G, Keogh JM, Atanassova N, Bounds R, et al. Rare Variant Analysis of Human and Rodent Obesity Genes in Individuals with Severe Childhood Obesity. Sci Rep. 2017 Dec 1;7(1):1–14.
5. Power RA, Parkhill J, de Oliveira T. Microbial genome-wide association studies: lessons from human GWAS. Nat Rev Genet. 2016;
6. Brynildsrud O, Bohlin J, Scheffer L, Eldholm V. Rapid scoring of genes in microbial pan-genome-wide association studies with Scoary. Genome Biol. 2016;17.
7. Lees JA, Galardini M, Bentley SD, Weiser JN, Corander J. pyseer: a comprehensive tool for microbial pangenome-wide association studies. Bioinformatics [Internet]. 2018 [cited 2018 Dec 19]; Available from: http://pyseer.readthedocs.io.
8. Earle SG, Wu C-H, Charlesworth J, Stoesser N, Gordon NC, Walker TM, et al. Identifying lineage effects when controlling for population structure improves power in bacterial association studies. Nat Microbiol. 2016;1.
9. Collins C, Didelot X. A phylogenetic method to perform genome-wide association studies in microbes that accounts for population structure and recombination. PLoS Comput Biol. 2018;
10. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MAR, Bender D, et al. PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses. Am J Hum Genet [Internet]. 2007 [cited 2017 Mar 22];81(3):559–75. Available from: www.ajhg.org
11. Chen PE, Shapiro BJ. The advent of genome-wide association studies for bacteria. Vol. 25, Current Opinion in Microbiology. 2015. p. 17–24.
12. Corander J, Croucher NJ, Harris SR, Lees JA, Tonkin□Hill G. Bacterial Population Genomics. In: Handbook of Statistical Genomics. Wiley; 2019. p. 997–1020.
13. Mooney MA, Wilmot B. Gene set analysis: A step-by-step guide. Am J Med Genet Part B Neuropsychiatr Genet. 2015 Oct 1;168(7):517–27.
14. White MJ, Yaspan BL, Veatch OJ, Goddard P, Risse-Adams OS, Contreras MG. Strategies for Pathway Analysis Using GWAS and WGS Data. Curr Protoc Hum Genet. 2019 Jan 1;100(1):e79.
15. Saund K, Lapp Z, Thiede SN, Pirani A, Snitkin ES. prewas: Data pre-processing for more informative bacterial GWAS. bioRxiv. 2019 Dec 20;2019.12.20.873158.
16. van Assche A, Alvarez-Perez S, de Breij A, de Brabanter J, Willems KA, Dijkshoorn L, et al. Phylogenetic signal in phenotypic traits related to carbon source assimilation and

chemical sensitivity in Acinetobacter species. Appl Microbiol Biotechnol. 2016;101:367–79.
17. Pagel M. Inferring the historical patterns of biological evolution. Nature. 1999;401(6756):877–84.
18. Fritz SA, Purvis A. Selectivity in mammalian extinction risk and threat types: A new measure of phylogenetic signal strength in binary traits. Conserv Biol. 2010;24(4):1042–51.
19. Paradis E, Schliep K. Phylogenetics ape 5.0: an environment for modern phylogenetics and evolutionary analyses in R.
20. R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.; 2018.
21. Orme D. The caper package□: comparative analysis of phylogenetics and evolution in R. R Packag version 05, 2. 2013;1–36.
22. Revell LJ. phytools: An R package for phylogenetic comparative biology (and other things). Methods Ecol Evol. 2012;3(2):217–23.
23. Wickham H. tidyverse: Easily Install and Load the “Tidyverse.” 2017.
24. Wickham H, Seidel D. scales: Scale Functions for Visualization. 2019.
25. Auguie B. gridExtra: Miscellaneous Functions for “Grid” Graphics. [Internet]. 2017. Available from: https://cran.r-project.org/package=gridExtra
26. Anaconda [Internet]. [cited 2020 Feb 21]. Available from: https://www.anaconda.com/
27. Saber MM, Shapiro BJ. Benchmarking bacterial genome-wide association study methods using simulated genomes and phenotypes. Microb genomics. 2020;6(3).

FIGURES

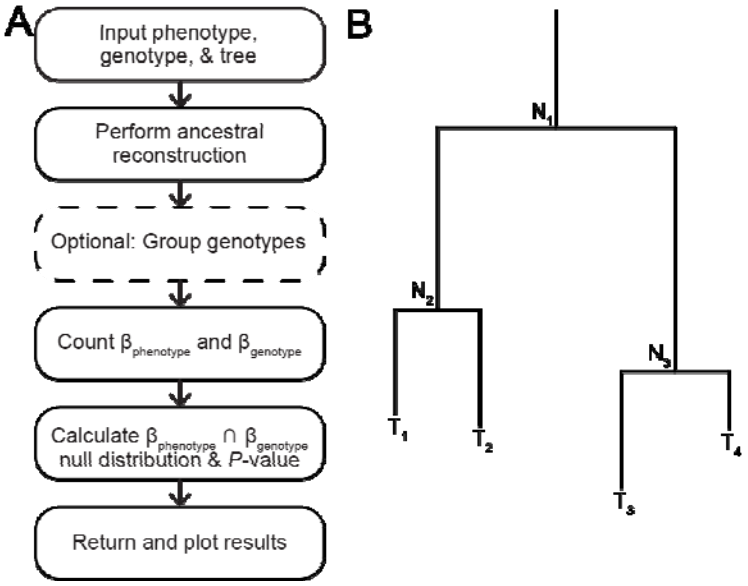


Figure 1. Hogwash workflow and tree nomenclature.
A) Software workflow. B) In this example phylogenetic tree N_1 is the root. Tree nodes are labeled N_1 – N_3. Tree tips are labeled T_1 – T_4. N_1 is a parent node to N_2 and N_3. N_2 is a child of N_1 and a parent to T_1 and T_2. Edges are lines connecting a parent node to a child node or a parent node to a tip.

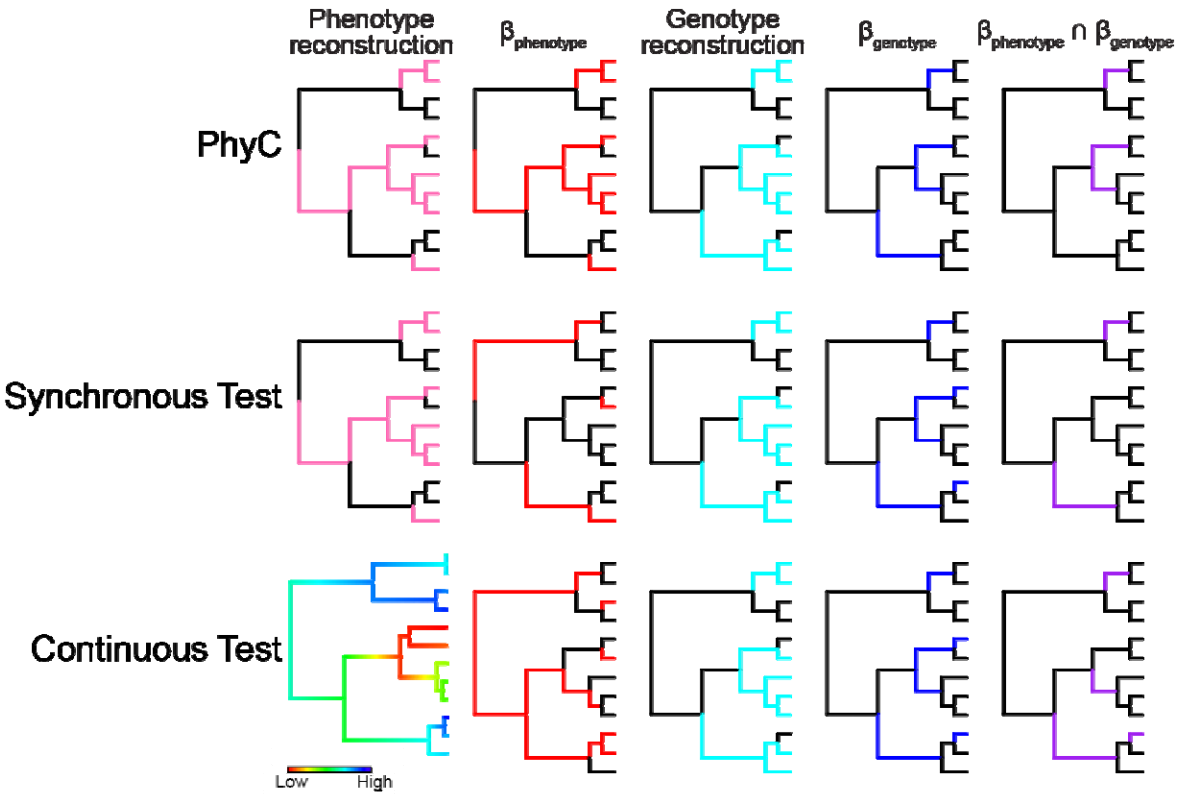


Figure 2. Schematic of PhyC, Synchronous, and Continuous Tests. Tree edges indicate: binary phenotype presence (pink), continuous phenotype value (rainbow), (red), genotype presence (light blue), (dark blue), and (purple).

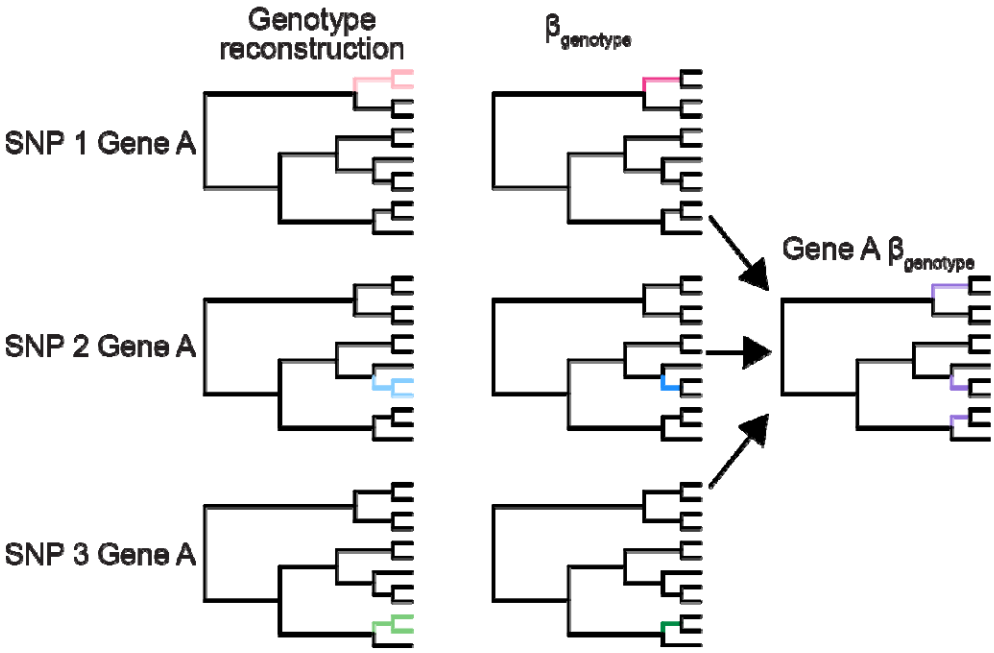


Figure 3. Example of hogwash grouping feature. In this case, three SNPs are found in the same gene (Gene A). No individual SNP is convergent on the tree. Hogwash performs ancestral state reconstruction on each SNP. The edges where SNP presence is inferred are colored.

Next, hogwash identifies the for each SNP (colored edges). Finally, hogwash combines the three SNP together to create the Gene A (purple edges). When the SNPS are grouped into Gene A the genotype converges on the tree.

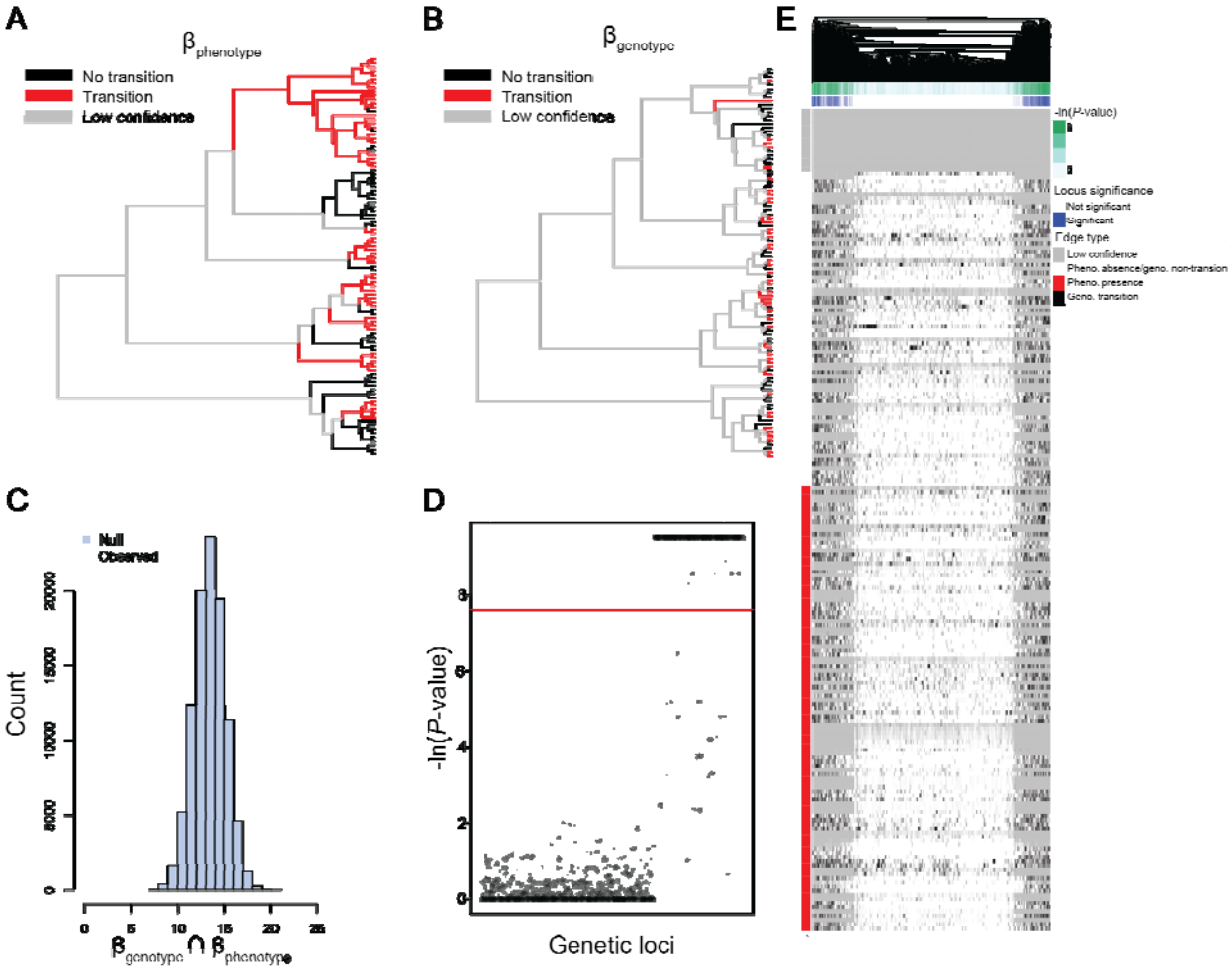


Figure 4. Example output from hogwash PhyC results from simulated data. A) Edges with: phenotype presence ( ) in red; phenotype absent in black; low confidence in tree or low confidence phenotype ancestral state reconstruction in gray. B) Edges with: genotype mutations that arose ( ) in red; genotype mutation did not arise in black; low confidence in tree or low confidence genotype ancestral state reconstruction in gray. C) Null distribution of . D) Manhattan plot for all tested genotypes. E) Heatmap with tree edges in the rows and genotypes in the columns. The genotypes are hierarchically clustered. The genotypes are classified as being a transition edge in black or non-transition edge in white. The column annotations pertain to loci significance; green indicates the *P*-value while blue indicates that the *P*-value is more significant than the user-defined threshold. The row annotation classifies the phenotype at each edge; red indicates phenotype presence and white indicates phenotype absence. Gray indicates a low confidence tree edge; low confidence can be due to low phenotype ancestral state reconstruction likelihood, low genotype ancestral state reconstruction likelihood, low tree bootstrap value, or long edge length.

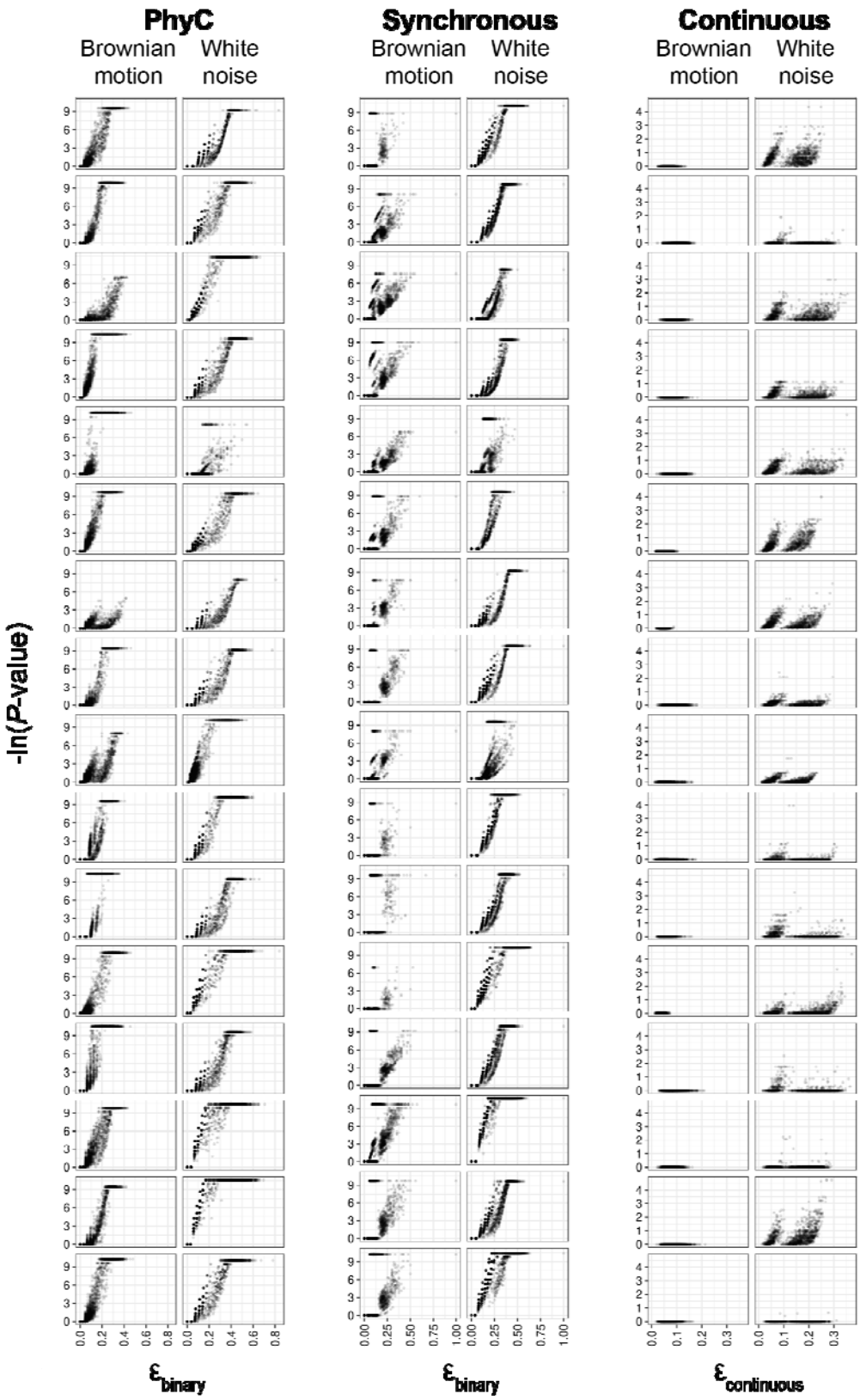
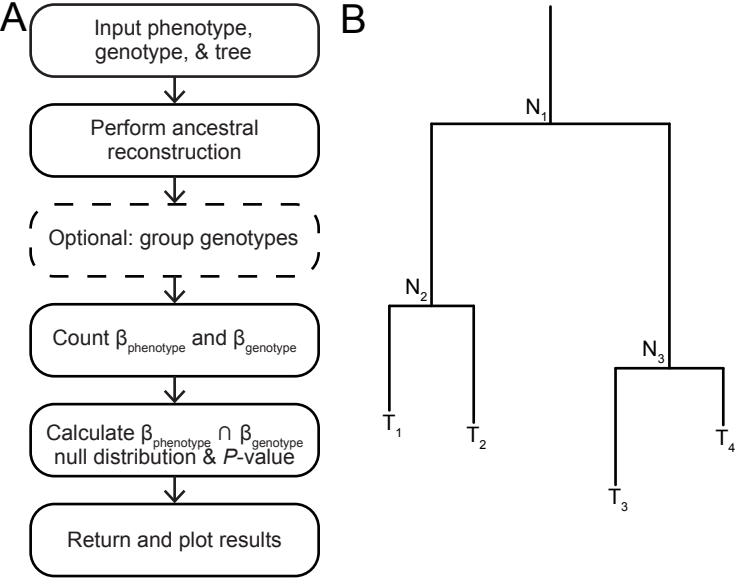


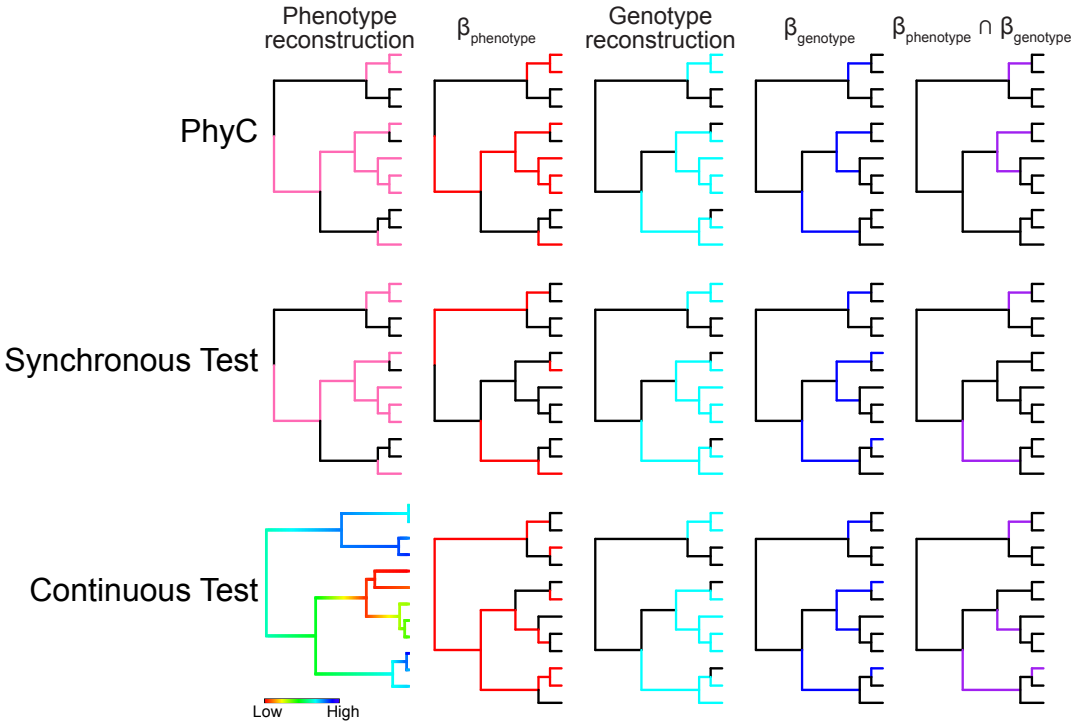
Figure 5. High values correlate with increased significance. Each plot is a tree-phenotype pair. Each point represents one genotype-phenotype pair.

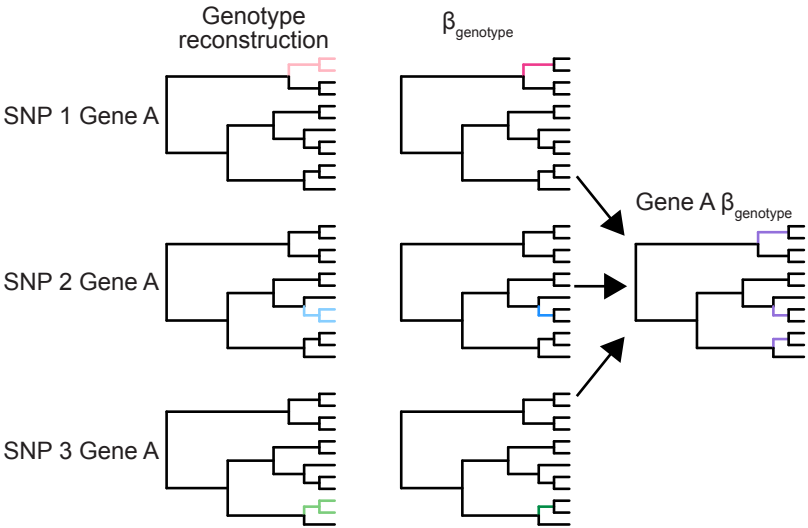
TABLES

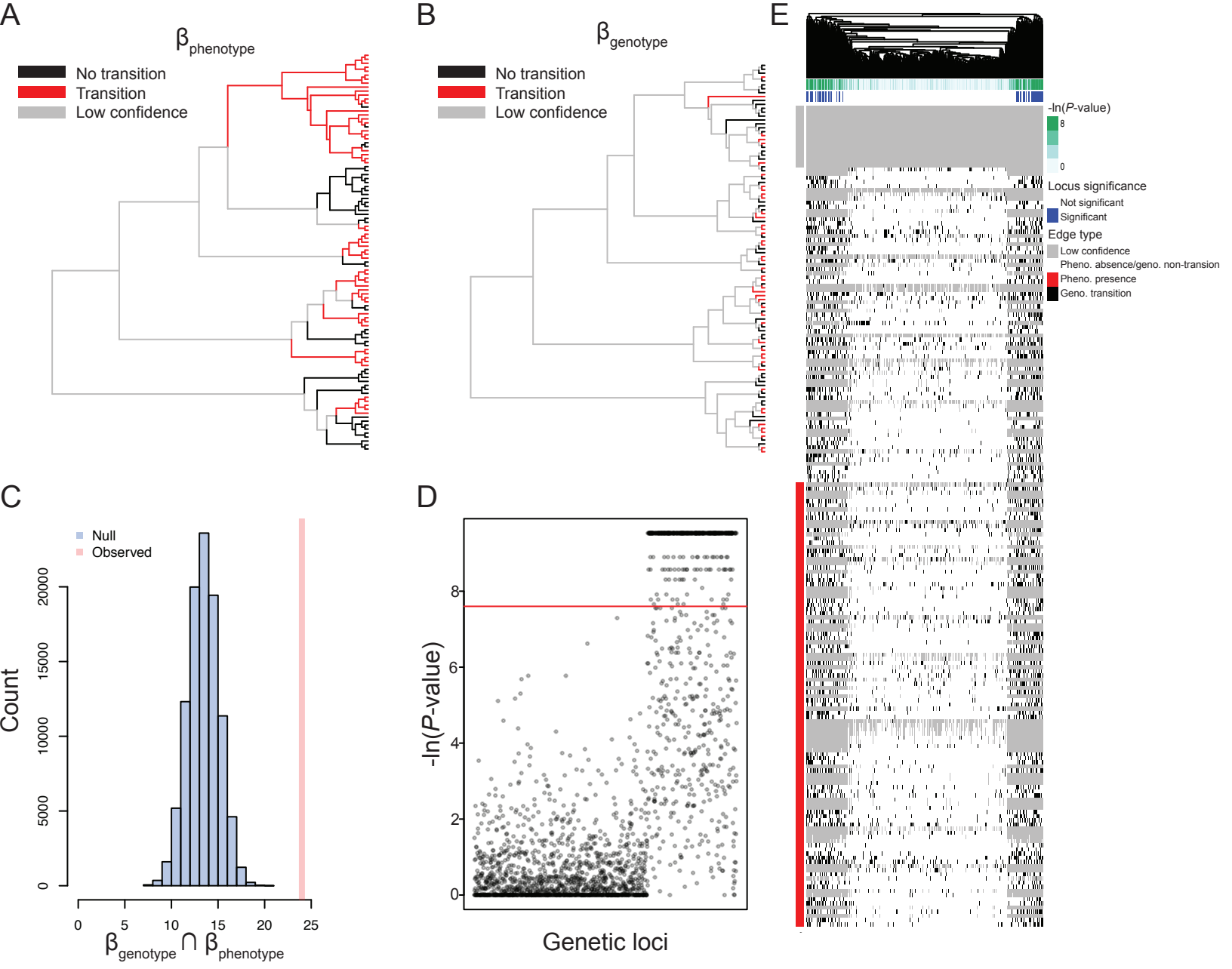
	Phenotype	
	Brownian motion	White noise
PhyC	0.91	0.93
Synchronous Test	0.60	0.94
Continuous Test	NA	0.08

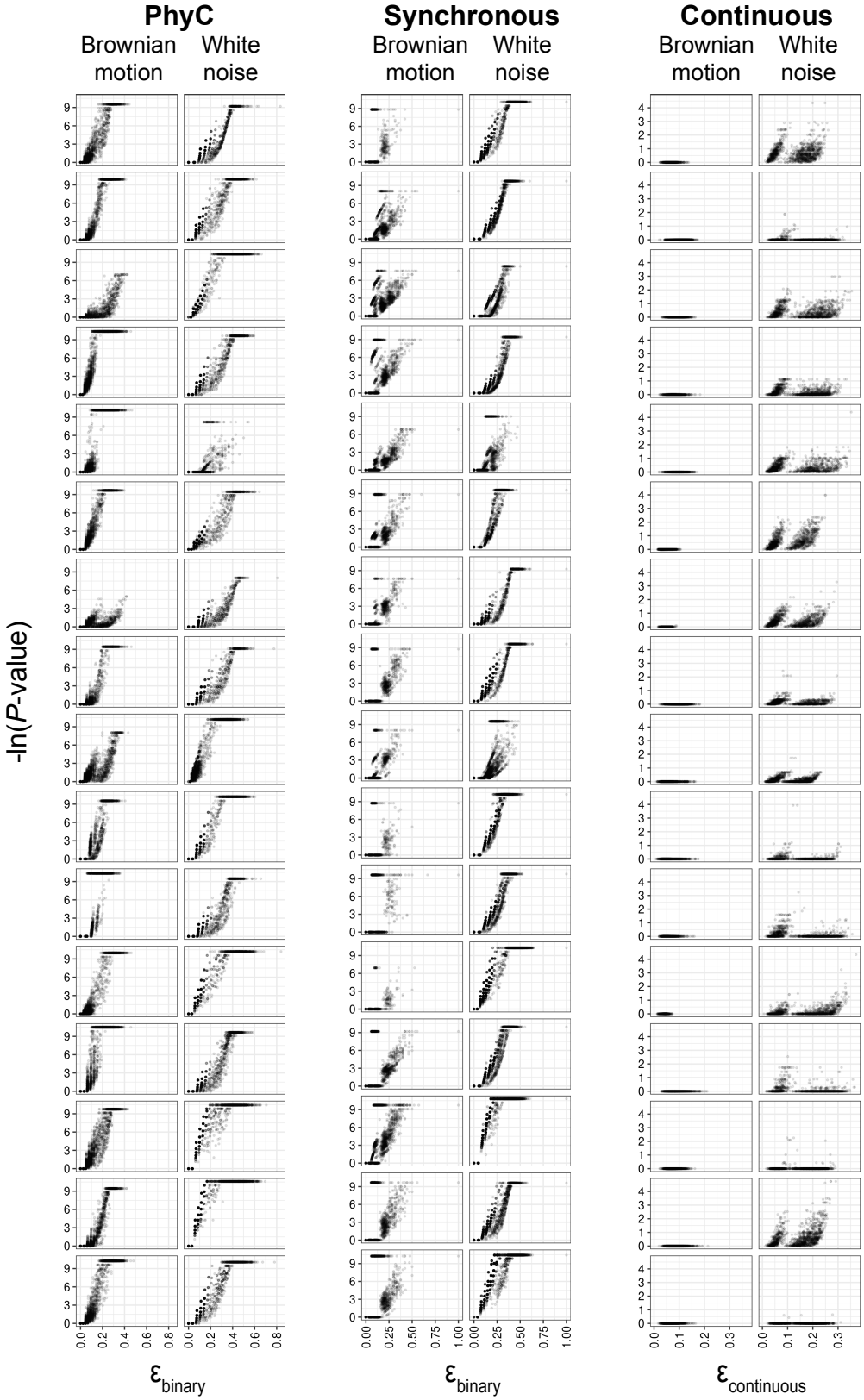
Table 1. Mean Spearman's rank correlation coefficient for -ln(*P*-value) versus from hogwash run on simulated data. The ρ could not be calculated for the results from the Continuous Test on the Brownian motion phenotypes because, after multiple testing correction, all *P*-values are identical.


A
Input phenotype, genotype, & tree
Perform ancestral reconstruction
Optional: group genotypes
Count $\beta_{phenotype}$ and $\beta_{genotype}$
Calculate $\beta_{phenotype} \cap \beta_{genotype}$ null distribution & *P*-value
Return and plot results
B
N_1
N_2
N_3
T_1
T_2
T_3
T_4


Phenotype reconstruction
$\beta_{phenotype}$
Genotype reconstruction
$\beta_{genotype}$
$\beta_{phenotype} \cap \beta_{genotype}$
PhyC
Synchronous Test
Continuous Test
Low
High


Genotype reconstruction
$\beta_{genotype}$
SNP 1 Gene A
SNP 2 Gene A
SNP 3 Gene A
Gene A $\beta_{genotype}$


A
$\beta_{phenotype}$
No transition
Transition
Low confidence
B
$\beta_{genotype}$
No transition
Transition
Low confidence
E
-ln(P-value)
8
0
Locus significance
Not significant
Significant
Edge type
Low confidence
Pheno. absence/geno. non-transion
Pheno. presence
Geno. transition
C
Null
Observed
Count
0
5000
10000
15000
20000
0
5
10
15
20
25
$\beta_{genotype} \cap \beta_{phenotype}$
D
-ln(P-value)
0
2
4
6
8
Genetic loci


PhyC
Brownian motion
White noise
Synchronous
Brownian motion
White noise
Continuous
Brownian motion
White noise
-ln(P-value)
ε binary
ε binary
ε continuous

	Phenotype	
	Brownian motion	White noise
PhyC	0.91	0.93
Synchronous Test	0.60	0.94
Continuous Test	NA	0.08

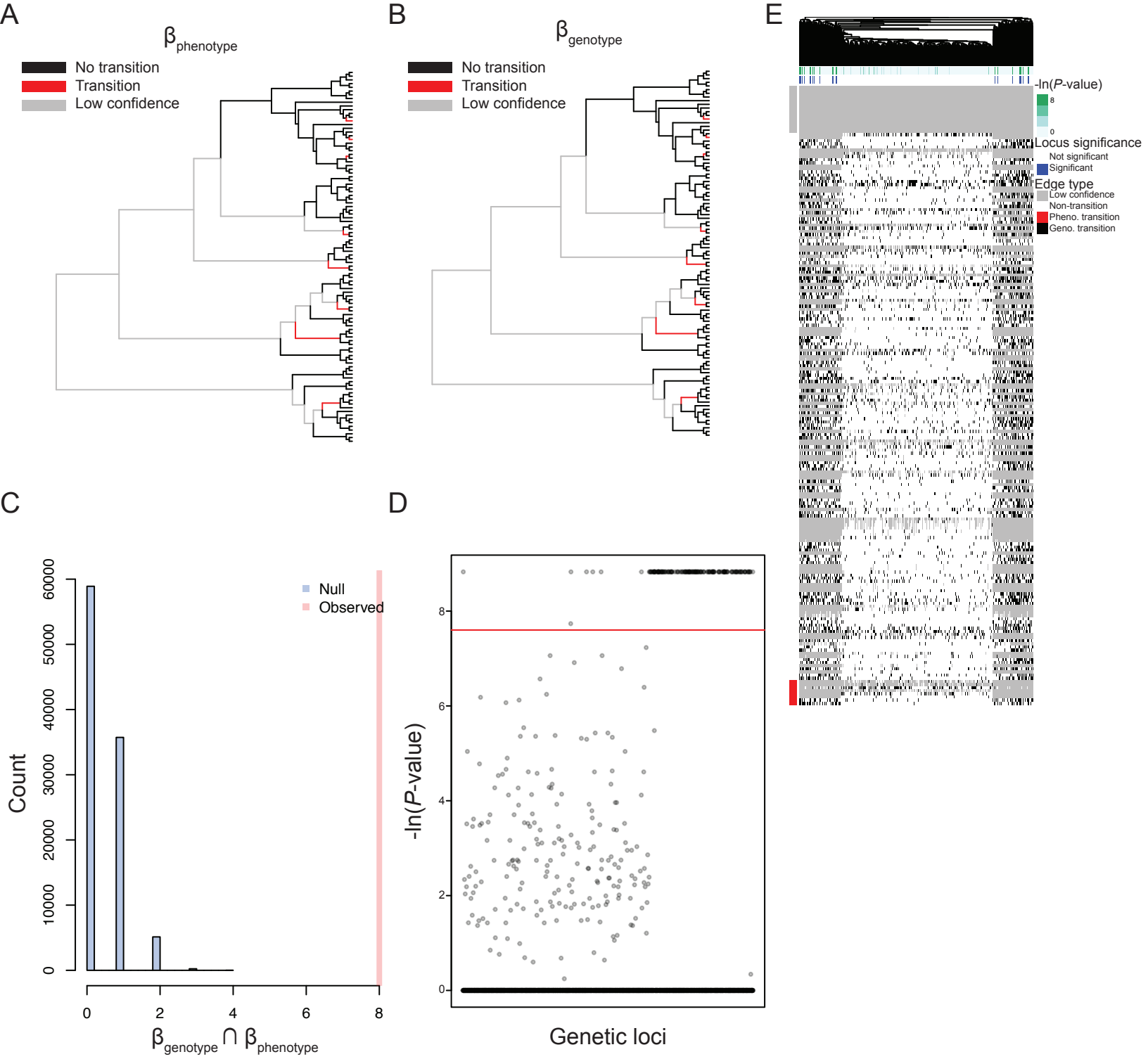


Figure S1. Example output from hogwash Synchronous Test results on simulated data
A) Ancestral reconstruction of phenotype. Phenotype transition edges ($\beta_{phenotype}$) in red, phenotype non-transitions edges in black, low confidence edges in gray. B) Ancestral reconstruction of simulated genotype. Genotype transition edges ($\beta_{genotype}$) in red; non-transition edges in black; low confidence edges in gray. C) Null distribution of $\beta_{genotype} \cap \beta_{phenotype}$; observed value in red. D) Manhattan plot for all tested genotypes. Significance threshold indicated in red. E) Heatmap with tree edges in the rows and genotypes in the columns. The genotypes are hierarchically clustered. The genotypes are classified as being a transition edge in black or non-transition edge in white. The column annotations pertain to the *P*-value; green is the *P*-value and blue indicates that the *P*-value is more significant that the user-defined threshold. The row annotation classifies the phenotype edge type. Phenotype transition edges are in red and non-transition edges are in white. Gray indicates a low confidence tree edge; low confidence can be due to low genotype ancestral state reconstruction likelihood, low tree bootstrap value, or long edge length.

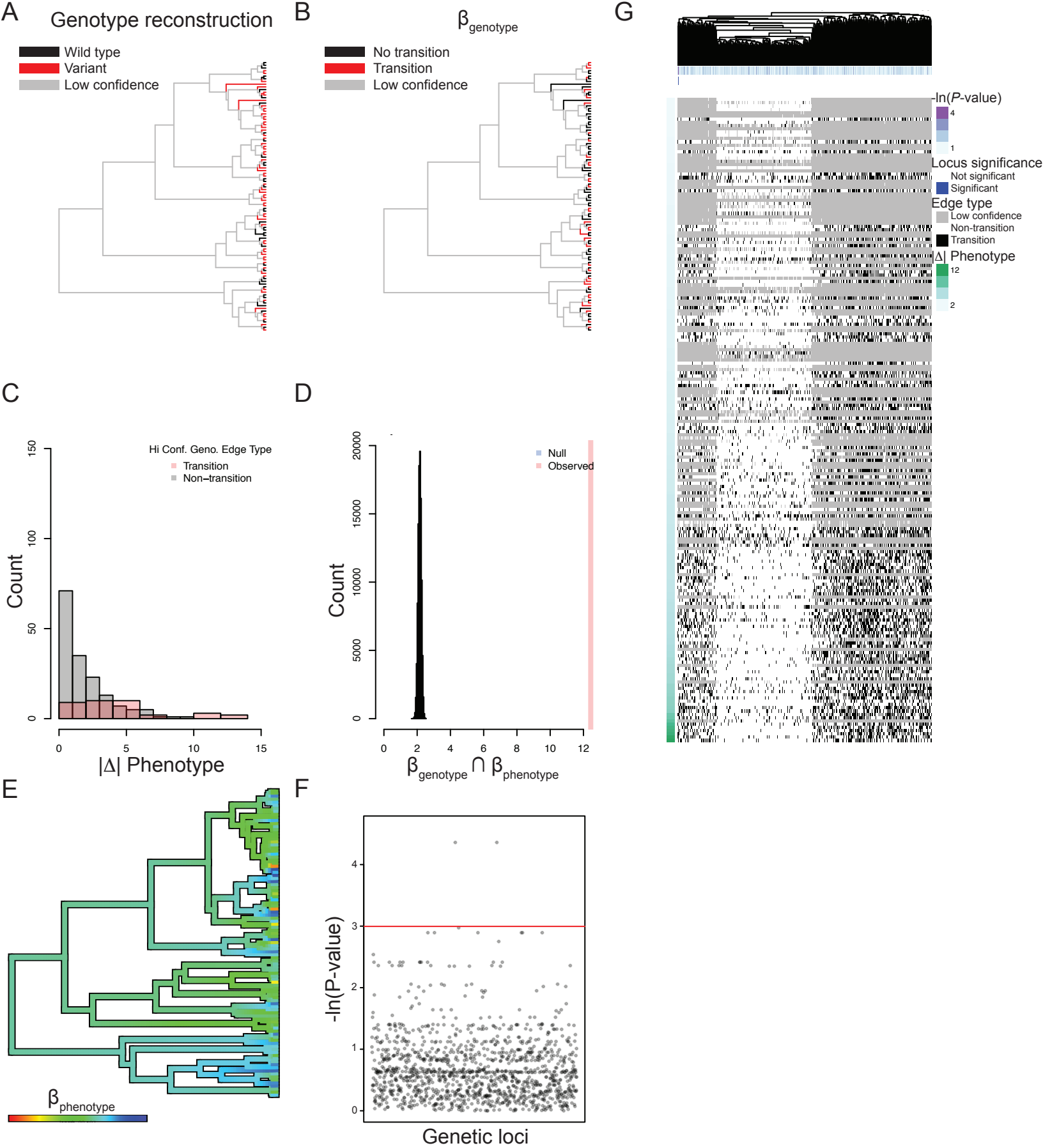


Figure S2. Example output from the Continuous Test run on simulated data.
A) Reconstruction for a simulated genotype. Wild type in black, variant presence in red, and low confidence edges in gray. B) Genotype transition edges ($\beta_{genotype}$) in red; non-transition edges in black; low confidence in tree or low confidence genotype ancestral state reconstruction in gray. C) Histogram of the change in phenotype per edge for high confidence tree edges. Genotype transition edges in red and genotype non-transition edges in gray. D) Null distribution of $\beta_{genotype} \cap \beta_{phenotype}$; observed value in red. E) Ancestral reconstruction of phenotype. F) Manhattan plot for all tested genotypes. The significance threshold is indicated in red. G) Heatmap with tree edges in the rows and genotypes in the columns. The genotypes are hierarchically clustered. The genotypes are classified as being a transition edge in black or non-transition edge in white. The column annotations pertain to the *P*-value; purple indicates the *P*-value and blue indicates that the *P*-value is more significant that the user-defined threshold. The row annotation shows the absolute value in the phenotype change per edge. Gray indicates a low confidence tree edge; low confidence can be due to low genotype ancestral state reconstruction likelihood, low tree bootstrap value, or long edge length.

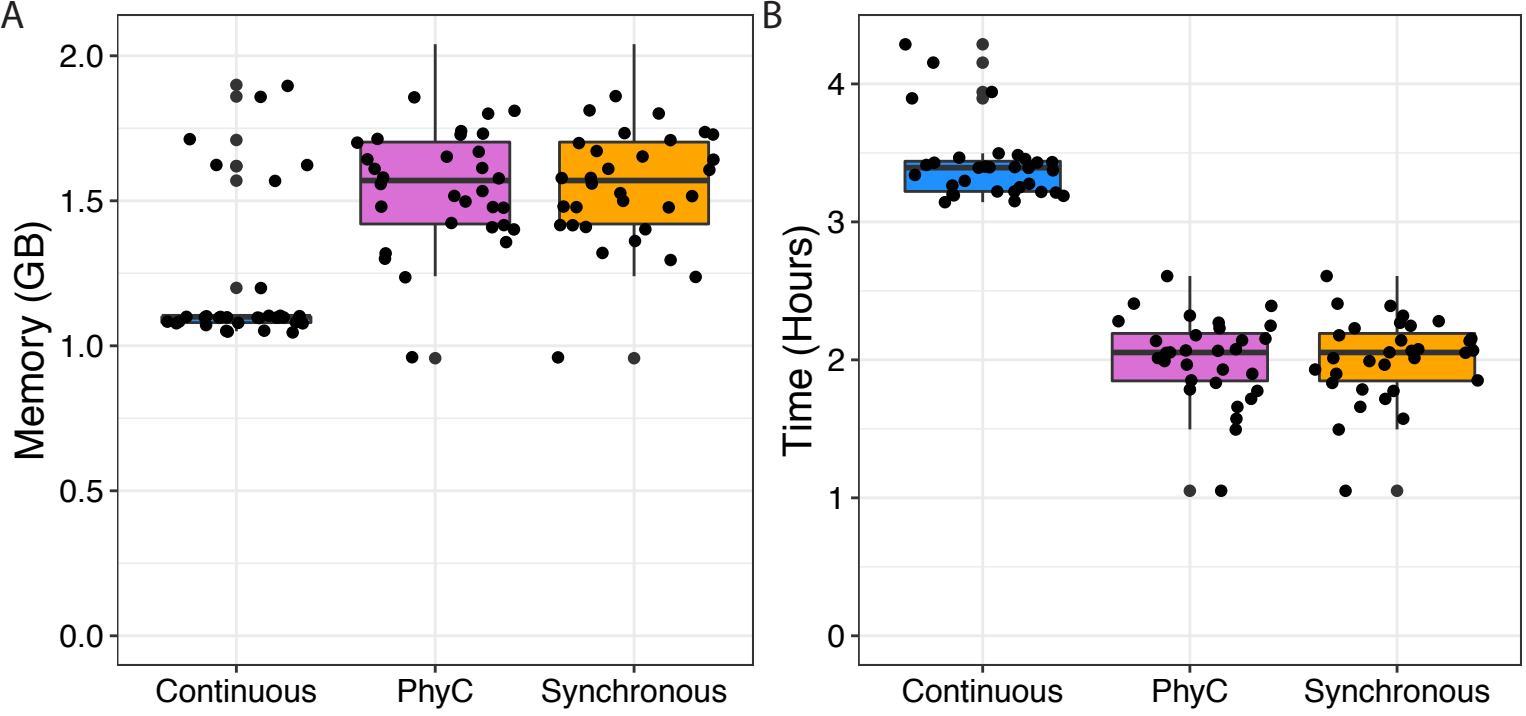


Figure S3. Memory usage and run time for hogwash on simulated data.