BIOINFORMATICS

Vol. 00 no. 00 2010
Pages 1–11

TEAM: Efficient Two-Locus Epistasis Tests in Human Genome-Wide Association Study

Xiang Zhang [1], Shunping Huang [1], Fei Zou [2] and Wei Wang [1]

[1]Department of Computer Science, University of North Carolina at Chapel Hill
[2]Department of Biostatistics, University of North Carolina at Chapel Hill

ABSTRACT

As a promising tool for identifying genetic markers underlying phenotypic differences, genome-wide association study (GWAS) has been extensively investigated in recent years. In GWAS, detecting epistasis (or gene-gene interaction) is preferable over single locus study since many diseases are known to be complex traits. A brute force search is infeasible for epistasis detection in the genome-wide scale because of the intensive computational burden. Existing epistasis detection algorithms are designed for dataset consisting of homozygous markers and small sample size. In human study, however, the genotype may be heterozygous, and number of individuals can be up to thousands. Thus existing methods are not readily applicable to human datasets. In this paper, we propose an efficient algorithm, TEAM, that significantly speeds up epistasis detection for human GWAS. Our algorithm is exhaustive, i.e., it does not ignore any epistatic interaction. Utilizing the minimum spanning tree structure, the algorithm incrementally updates the contingency tables for epistatic tests without scanning all individuals. Our algorithm has broader applicability and is more efficient than existing methods for large sample study. It supports any statistical test that is based on contingency tables, and enables both family-wise error rate (FWER) and false discovery rate (FDR) controlling. Extensive experiments show that our algorithm only needs to examine a small portion of the individuals to update the contingency tables, and it achieves at least an order of magnitude speedup over the brute force approach.

1 INTRODUCTION

Genetic association analysis examines the statistical correlation between an organism's genotype with its phenotype. With the development of high-throughput genotyping technologies, genetic variation of human and other model organisms has been measured at genome-wide scale. As the most abundant source of genetic variation, the number of single nucleotide polymorphism (SNPs) in public databases (dbGaP, JAX) is up to millions. Genome-wide association study (GWAS) has been shown to be a promising tool to locate the genetic factors that cause phenotypic differences (Saxena *et al.*, 2007; Scuteri *et al.*, 2007; WTCCC, 2007; Weedon *et al.*, 2007). Epistasis, or gene-gene interaction detection, has received increasing attention in complex trait analysis. Different from single-locus approach, the goal of two-locus epistasis detection is to identify interacting SNP-pairs that have strong association with the phenotype. Please refer to Balding (2006); Hirschhorn and Daly (2005); Hoh and Ott (2003); Musani *et al.* (2007) for reviews of current progress and challenges in epistasis detection in GWAS.

There are two grand challenges in epistasis detection. The first is to develop statistical tests that can effectively capture the interaction between SNPs. Various tests have been proposed for two-locus association study, such as the chi-square test, likelihood-ratio test, and entropy-based test (Balding, 2006). Another crucial challenge in two-locus association study is the intensive computational burden imposed by the enormous search space. Suppose that there are N SNPs for M individuals. The overall search space of pairwise interactions is $MN(N-1)/2$. The large number of tests also causes the multiple testing problem (Miller, 1981). Controlling the family-wise error rate (FWER) and false discovery rate (FDR) are standard ways to control the error rate (Dudoit and Laan, 2008; Westfall and Young, 1993). In the FWER and FDR controlling, permutation test is preferred over simple Bonferroni correction since many SNPs are correlated (Churchill and Doerge, 1994). The correlation structure among genotype profiles is preserved across permutations and is incorporated into permutation p-value estimation. The idea of permutation test is to randomly shuffle the phenotype values among the individuals and recalculate the test statistics. The distribution of these test values are used to estimate the null distribution. Permutation test dramatically increases the search space. With K permutations, the entire search space of two-locus association mapping is $KMN(N-1)/2$. Consider a moderate GWAS setting, in which $M = 1,000$, $N = 100,000$, and $K = 1,000$. The size of the search space is about 5×10^{15}. Apparently, a brute force enumeration of the search space is infeasible and thus efficient algorithms are in demand.

Although the computational challenge of epistasis detection has been well recognized, the algorithmic development is still very limited. For a small number of SNPs, e.g., from tens to a few hundreds, exhaustive algorithms that explicitly enumerate all possible SNP combinations have been developed (Nelson *et al.*, 2001; Ritchie *et al.*, 2001). These methods are not scalable for genome-wide computing. Genetic algorithm (Carlborg *et al.*, 2000) has been proposed. This approach is heuristic, which does not guarantee to find the optimal solution. To avoid explicitly exploring the entire search space, a common heuristic used in epistasis detection is a two-step approach (Evans *et al.*, 2006; Hoh *et al.*, 2000; Yang *et al.*, 2009). First, a subset of SNPs are selected according to certain criteria. Then the selected SNPs are used for subsequent epistatic analysis. However, the SNP screening process suffers from the same problem as the single-locus approach. SNPs

© Oxford University Press 2010.

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}	S_{13}	S_{14}	S_{15}	S_{16}	S_{17}	S_{18}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}
X_1	0	0	0	1	2	0	2	0	2	0	0	2	0	0	0	2	0	2	1	0	0	2	2	0
X_2	2	2	0	2	0	2	0	2	2	2	2	0	1	0	0	2	0	2	1	0	2	2	2	2
X_3	2	0	0	2	0	2	0	1	2	1	2	2	1	0	2	2	0	2	1	2	2	2	2	2
X_4	0	2	2	0	0	0	2	1	0	2	2	0	0	0	0	0	0	0	1	0	1	2	0	0
X_5	0	2	2	0	0	0	1	1	2	1	2	0	0	0	0	0	0	2	1	0	2	2	0	2
X_6	0	2	2	0	0	0	2	1	0	1	2	0	0	0	0	2	0	2	1	0	2	2	0	0
Y_0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Y_1	0	1	0	0	1	1	1	0	0	0	1	1	1	1	1	0	1	0	0	1	0	1	0	0
Y_2	0	0	1	0	1	1	1	1	1	1	0	0	1	0	0	1	1	1	0	0	0	1	0	0
Y_3	1	0	1	1	1	1	0	1	1	1	0	0	0	1	0	1	0	0	1	0	0	0	1	0
Y_4	0	1	1	1	1	1	0	0	0	1	0	0	1	0	1	0	0	0	1	1	0	1	1	0
Y_5	1	0	1	1	0	0	0	0	0	1	1	1	1	0	1	1	0	0	0	0	0	1	1	1

Table 1. An example dataset consisting of 6 SNPs $\{X_1, \cdots, X_6\}$, the original phenotype Y_0 and 5 phenotype permutations $\{Y_1, \cdots, Y_5\}$ for 24 individuals $\{S_1, \cdots, S_{24}\}$

with strong epistasis but low marginal effects are likely to be filtered out (Zhang *et al*., RECOMB2009).

Recently, the approach based on search space pruning has been shown to be able to dramatically speed up the process of epistasis detection without compromising the optimality of the results. FastANOVA (Zhang *et al*., 2008) and FastChi (Zhang *et al*., PSB2009) are specifically designed for ANOVA test and chi-square test respectively. The COE algorithm (Zhang *et al*., RECOMB2009) is a more general approach that is applicable to all convex tests. Utilizing an upper bound derived for the test being used, these algorithms only need to examine a small number of promising SNP-pairs and prune the SNP-pairs that are proven to have no strong association with the phenotype. Unlike heuristic approaches, these algorithms are guaranteed to find the optimal solution. Although these methods provide promising alternatives for GWAS, there are two major drawbacks that limit their applicability. First, they are designed for relatively small sample size and only consider homozygous markers (i.e., each SNP can be represented as a $\{0, 1\}$ binary variable). In human study, however, the sample size is usually large and most SNPs contain heterozygous genotypes and are coded using $\{0, 1, 2\}$. These make existing methods intractable. Second, although the FWER and the FDR are both widely used for error controlling, existing methods are designed only to control the FWER. From a computational point of view, the difference in the FWER and the FDR controlling is that, to estimate FWER, for each permutation, only the maximum two-locus test value is needed. To estimate the FDR, on the other hand, for each permutation, all two-locus test values must be computed. Please refer to Section 2 for further details of the FWER and the FDR controlling.

In this paper, we propose an exhaustive algorithm, TEAM[1], for efficient epistasis detection in human GWAS. TEAM has several advantages over previous methods.

- It supports to both homozygous and heterozygous data.
- By exhaustively computing all two-locus test values in permutation test, it enables both FWER and FDR controlling.
- It is applicable to all statistics based on the contingency table. Previous methods either are designed for specific tests or require the test statistics satisfy certain property.
- Experimental results demonstrate that TEAM is more efficient than existing methods for large sample study.

TEAM incorporates permutation test for proper error controlling. The key idea is to incrementally update the contingency tables of two-locus tests. We show that only four of the eighteen observed frequencies in the contingency table need to be updated to compute the test value. In the algorithm, we build a minimum spanning tree (Cormen *et al*., 2001) on the SNPs. The nodes of the tree are SNPs. Each edge represents the genotype difference between the two connected SNPs. This tree structure can be utilized to speed up updating process for the contingency tables. A majority of the individuals are pruned and only a small portion are scanned to update the contingency tables. This is advantageous in human study, which usually involves thousands of individuals. Extensive experimental results demonstrate the efficiency of the TEAM algorithm.

2 THE PROBLEM OF TWO-LOCUS EPISTASIS DETECTION IN HUMAN GWAS

Suppose that the genotype dataset consists of N SNPs $\{X_1, \cdots, X_N\}$ for M individuals $\{S_1, \cdots, S_M\}$. We adopt the convention of using 0 and 2 to represent the homozygous majority and homozygous minority genotype respectively, and 1 to represent the heterozygous case. Let $Y_0 \in \{0, 1\}$ be the phenotype of interest (0 for controls and 1 for cases). Let $Y' = \{Y_1, \cdots, Y_K\}$ be the set of K permutations of Y_0. In each permutation Y_k, the phenotype labels are randomly reassigned to individuals with no replacement.

Table 1 shows an example dataset of SNPs and phenotype permutations. The genotype dataset consists of 6 SNPs $\{X_1, \cdots, X_6\}$ for 24 individuals $\{S_1, \cdots, S_{24}\}$. Individuals $\{S_1, \cdots, S_{12}\}$ are cases and $\{S_{13}, \cdots, S_{24}\}$ are controls. The phenotype is permuted 5 times, i.e., $Y' = \{Y_1, \cdots, Y_5\}$.

Let $\mathscr{T}$ denote the statistical test to be used. Specifically, we represent the test value of SNP X_i and phenotype Y_k $(0 \leq k \leq K)$ as $\mathscr{T}(X_i, Y_k)$, and represent the test value of SNP-pair $(X_i X_j)$ and Y_k as $\mathscr{T}(X_i X_j, Y_k)$. A contingency table, which records the observed values of certain events, is the basis of many statistical tests. Table 2 shows contingency tables for the single-locus test $\mathscr{T}(X_i, Y_k)$ and $\mathscr{T}(X_j, Y_k)$, genotype relationship between SNPs X_i and X_j, and two-locus test $\mathscr{T}(X_i X_j, Y_k)$.

[1] TEAM stands for Tree-based Epistasis Association Mapping.

	$X_i = 0$	$X_i = 1$	$X_i = 2$	Total
$Y_k = 0$	event A	event B	event E	
$Y_k = 1$	event C	event D	event F	
Total				M

(a) Contingency table for $\mathscr{T}(X_i, Y_k)$

	$X_j = 0$	$X_j = 1$	$X_j = 2$	Total
$Y_k = 0$	event G	event H	event I	
$Y_k = 1$	event J	event L	event O	
Total				M

(b) Contingency table for $\mathscr{T}(X_j, Y_k)$

	$X_i = 0$	$X_i = 1$	$X_i = 2$	Total
$X_j = 0$	event S	event T	event R	
$X_j = 1$	event P	event Q	event U	
$X_j = 2$	event V	event W	event Z	
Total				M

(c) Contingency table for two SNPs X_i and X_j

	$X_i = 0$			$X_i = 1$			$X_i = 2$			Total
	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$	
$Y_k = 0$	event a_1	event a_2	event a_3	event b_1	event b_2	event b_3	event e_1	event e_2	event e_3	
$Y_k = 1$	event c_1	event c_2	event c_3	event d_1	event d_2	event d_3	event f_1	event f_2	event f_3	
Total										M

(d) Contingency table for (X_iX_j) and Y_k

Table 2. Contingency tables for single-locus tests $\mathscr{T}(X_i, Y_k)$, $\mathscr{T}(X_j, Y_k)$, genotype relation between (X_i, X_j), and two-locus test $\mathscr{T}(X_iX_j, Y_k)$

Because of the large number of hypotheses being tested, multiple testing problem has received considerable attention in GWAS. Controlling the FWER and FDR are two widely used approaches to control the error rate. The FWER is the probability of having at least one false positive. The FDR is the expected proportion of false positives among rejected hypotheses. Permutation test is the standard way to estimate the null distribution in both approaches. Next, we briefly describe the typical procedures of the FWER and FDR control. For statistical background of these approaches, please refer to Dudoit and Laan (2008); Westfall and Young (1993) for details.

The FWER controlling procedure: For each permutation $Y_k \in Y'$, let $\mathscr{T}_{Y_k}$ represent the maximum test value among all SNP-pairs, i.e., $\mathscr{T}_{Y_k} = \max\{\mathscr{T}(X_iX_j, Y_k) | 1 \leq i < j \leq N\}$. The distribution of $\{\mathscr{T}_{Y_k} | Y_k \in Y'\}$ is used as the null distribution. Given an error rate threshold α, the *critical value* $\mathscr{T}_\alpha$ is the αK-th largest value in $\{\mathscr{T}_{Y_k} | Y_k \in Y'\}$. A SNP-pair (X_iX_j) is considered significant if its test value with the original phenotype Y_0 exceeds the critical value, i.e., $\mathscr{T}_(X_iX_j, Y_0) \geq \mathscr{T}_\alpha$.

The FDR controlling procedure: Let PV represent the set of the pooled test values of all permutation tests, i.e., $PV = \{\mathscr{T}(X_iX_j, Y_k) | 1 \leq i < j \leq N, 1 \leq k \leq K\}$. The *p*-value of test $\mathscr{T}(X_iX_j, Y_0)$ can be calculated as $p(\mathscr{T}(X_iX_j, Y_0)) = |\{t \geq \mathscr{T}(X_iX_j, Y_0) | t \in PV\}| / |PV|$, i.e., the proportion of the values in PV that are no less than $\mathscr{T}(X_iX_j, Y_0)$. Let $p_{(1)} \leq p_{(2)} \cdots \leq p_{(N(N-1)/2)}$ be the ordered *p*-values of the original tests. Let $v = \max\{u : p_{(u)} \leq \frac{u\alpha}{N(N-1)/2}\}$. The classic Benjamini-Hochberg method rejects all hypotheses for which the corresponding p-values are in the set $\{p_{(1)}, p_{(2)}, \cdots, p_{(v)}\}$.

In the FWER controlling, we only need the maximum test value of each permutation. To control the FDR, all test values need to be computed to estimate the p-value of the original tests. The existing algorithms, such as FastChi (Zhang *et al.*, PSB2009) and COE (Zhang *et al.*, RECOMB2009), prune the SNP-pairs having weak associations. Thus they cannot be used to control the FDR. Our algorithm, TEAM, exhaustively computes the test values of all SNP-pairs for every permutation. It can be used for both the FWER and the FDR controlling. In this paper, we mainly focus on the problem of permutation test, since it is the most computationally intensive procedure. Testing SNP-pairs using original phenotype can be treated as a special case of permutation test.

3 FREE VARIABLES IN THE CONTINGENCY TABLE OF TWO-LOCUS TEST

Let E_{event} and O_{event} denote the expected frequency and observed frequency of an event in Table 2. Note that each event represents a subset of individuals. For example, event D is a subset of individuals satisfying $(X_i = 1 \wedge Y_k = 1)$, and O_D represents its observed frequency, i.e., $O_D = |D|$. Using the dataset in Table 1, consider X_3 and Y_4 (i.e., $i = 3$ and $k = 4$), we have $D = \{S_{10}, S_{13}, S_{19}\}$, and $O_D = 3$.

Many statistics, such as chi-square test and likelihood ratio test are defined as functions of the observed frequencies in contingency tables. For any test $\mathscr{T}$ based on the contingency table, to calculate the two-locus test value $\mathscr{T}(X_iX_j, Y_k)$, one needs all eighteen observed frequencies for the events in the two-locus contingency table shown in Table 2(d). The following theorem shows that we only need four of the eighteen values to calculate the two-locus test value given the three contingency tables in Tables 2(a), (b), and (c).

THEOREM 3.1. *For SNPs X_i, X_j, and permutation Y_k, given the observed frequencies in Tables 2(a), (b), and (c), specifically, the values of $\{O_D, O_F, O_J, O_L, O_O, O_S, O_P, O_V, O_T, O_Q, O_W, O_R, O_U, O_Z\}$, all of the observed frequencies in Table 2(d) can be determined if the values of $\{O_{d_2}, O_{d_3}, O_{f_2}, O_{f_3}\}$ are known.*

PROOF. See Appendix.

Suppose that we have all the single-locus contingency tables, i.e., Tables 2(a) and (b). Given a SNP-pair (X_i, X_j), Table 2(c)

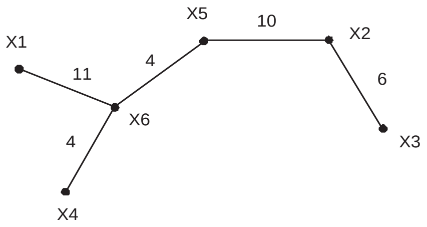


Fig. 1. The minimum spanning tree built on the SNPs in the example dataset shown in Table 1

is fixed. Thus, from Theorem 3.1, for permutation Y_k, once we have the values of $\{O_{d_2}, O_{d_3}, O_{f_2}, O_{f_3}\}$, $\mathscr{T}(X_iX_j, Y_k)$ can be calculated accordingly. In the following, we show that these values can be computed incrementally utilizing a minimum spanning tree built on SNPs. We focus on the incremental process for O_{d_2}. The same process can be applied to update O_{d_3}, O_{f_2}, and O_{f_3}. We first discuss how to update O_{d_2} for a specific permutation. Then we show that the procedure can also handle all the permutations in a batch mode.

4 BUILDING THE MINIMUM SPANNING TREE ON THE SNPS

To build a minimum spanning tree (Cormen *et al.*, 2001) on the SNPs, let the SNPs $\{X_1, X_2, \cdots, X_N\}$ be the nodes and SNP-pairs (X_iX_j) $(i \neq j)$ be the (undirected) edges. For each edge (X_iX_j), we denote its weight (the number of individuals having different genotypes in the two SNPs) as $w(X_iX_j)$. A *spanning tree* $\mathcal{T}$ is a tree that spans (connects) all SNPs. Let $V(\mathcal{T})$ be its node set and $E(\mathcal{T})$ be its edge set. A *minimum spanning tree* is a spanning tree whose weight $W_{\mathcal{T}} = \sum w(X_iX_j)$, where $(X_iX_j) \in E(\mathcal{T})$, is no greater than any other spanning tree. Figure 1 shows the minimum spanning tree built using the example dataset in Table 1. The number on each edge represents its weight. For example, in X_3 and X_2, there are 6 individuals, $\{S_2, S_8, S_{10}, S_{12}, S_{15}, S_{20}\}$, having different genotypes.

For any individual, the genotype difference from X_i to X_j can be any one of the six combinations, i.e., $0 \rightarrow 1$ (indicating that the genotype in X_i is 0, and the genotype in X_j is 1), $1 \rightarrow 0$, $0 \rightarrow 2$, $2 \rightarrow 0$, $1 \rightarrow 2$, and $2 \rightarrow 1$. Using the example dataset in Table 1, Table 3 shows the genotype differences between the connected two SNPs in the minimum spanning tree in Figure 1. We use $(X_iX_j)_{\{u\rightarrow v\}}$ $(u, v \in \{0, 1, 2\})$ to represent the set of individuals whose genotype in X_i is u and genotype in X_j is v. For example, $(X_3X_2)_{\{1\rightarrow 2\}} = \{S_8, S_{10}\}$, and $(X_3X_2)_{\{1\rightarrow 2\}\cup\{0\rightarrow 2\}} = \{S_2, S_8, S_{10}\}$.

5 INCREMENTALLY UPDATING OBSERVED FREQUENCY O_{d_2}

In this section, we discuss how to update O_{d_2} by utilizing the minimum spanning tree. For clarity, from now on, we use $d_2(X_iX_j, Y_k)$ to denote the specific event d_2 for the SNP-pair (X_iX_j) and permutation Y_k, i.e., the subsets of individuals satisfying $(X_i = 1 \wedge X_j = 1 \wedge Y_k = 1)$. We use $O_{d_2}(X_iX_j, Y_k)$ to represent its observed frequency, i.e., $O_{d_2}(X_iX_j, Y_k) = |d_2(X_iX_j, Y_k)|$. This notation also applies to other events in the contingency tables shown in Table 2. For example, $D(X_i, Y_k)$ represents the subset of individuals satisfying $(X_i = 1 \wedge Y_k = 1)$, and $O_D(X_i, Y_k) = |D(X_i, Y_k)|$.

Next we show that for any SNP-pair (X_iX_j) and an edge $(X_jX'_j) \in E(\mathcal{T})$, given $O_{d_2}(X_iX_j, Y_k)$, how to update the value for $O_{d_2}(X_iX'_j, Y_k)$. From the contingency tables in Table 2, it is easy to see that

$$O_{d_2}(X_iX_j, Y_k) = |D(X_i, Y_k) \cap Q(X_i, X_j)|,$$

and

$$O_{d_2}(X_iX'_j, Y_k) = |D(X_i, Y_k) \cap Q(X_i, X'_j)|.$$

The following theorem shows that, given $O_{d_2}(X_iX_j, Y_k)$ and $D(X_i, Y_k)$, using the genotype difference associated with edge $(X_jX'_j)$, we can get the value of $O_{d_2}(X_iX'_j, Y_k)$.

THEOREM 5.1. *For any SNP-pair (X_iX_j) and an edge $(X_jX'_j) \in E(\mathcal{T})$, we have $O_{d_2}(X_iX'_j, Y_k) = O_{d_2}(X_iX_j, Y_k) +$ $|D(X_i, Y_k) \cap (X_jX'_j)_{\{0\rightarrow 1\}\cup\{2\rightarrow 1\}}|$- $|D(X_i, Y_k) \cap (X_jX'_j)_{\{1\rightarrow 0\}\cup\{1\rightarrow 2\}}|$.*

PROOF. See Appendix.

EXAMPLE 5.2. *Using the example dataset in Table 1, let $i = 3$, $j = 2$, $j' = 5$, and $k = 4$, i.e., we consider SNP-pair (X_3X_2), permutation Y_4, and the edge (X_2X_5) in Figure 1. Suppose that we already know that $O_{d_2}(X_3X_2, Y_4) = 2$, and event $D(X_3, Y_4) = \{S_{10}, S_{13}, S_{19}\}$. From Table 3, we have $(X_2X_5)_{\{0\rightarrow 1\}\cup\{2\rightarrow 1\}} = \{S_7, S_8, S_{10}\}$, and $(X_2X_5)_{\{1\rightarrow 0\}\cup\{1\rightarrow 2\}} = \{S_{13}\}$. Thus according to Theorem 5.1, we have $O_{d_2}(X_3X_5, Y_4) = O_{d_2}(X_3X_2, Y_4) + |\{S_{10}\}| - |\{S_{13}\}| =$ 2. Note that by this way, we get the value of $O_{d_2}(X_3X_5, Y_4)$ from $O_{d_2}(X_3X_2, Y_4)$without scanning all individuals.*

So far, we have discussed the procedure to update the value of $O_{d_2}(X_iX'_j, Y_k)$ from $O_{d_2}(X_iX_j, Y_k)$ for a specific phenotype permutation Y_k. This procedure can be easily extended to handle all the permutations. From Theorem 5.1, for any permutation Y_k, to update the value of $O_{d_2}(X_iX'_j, Y_k)$ from $O_{d_2}(X_iX_j, Y_k)$, we need the value of $D(X_i, Y_k)$ and the genotype difference associated with edge $(X_jX'_j)$. Note that the genotype difference is fixed once the minimum spanning tree is built. Next, we discuss how to compute $D(X_i, Y_k)$ for all permutations $\{Y_1, Y_2, \cdots, Y_K\}$ in a batch mode in detail.

Let $D_K(X_i)$ be a list of M entries, with each entry corresponding to an individual. For each individual S_m, we record in $D_K(X_i)[m]$ the set of phenotypes satisfying $(X_i = 1 \wedge Y_k = 1)$. For example, consider the dataset in Table 1, we have that $D_K(X_3)[8] = \{Y_2, Y_3\}$. Table 4(a) shows the entries of $D_K(X_3)$. Only non-empty entries, i.e., $D_K(X_i)[m] \neq \emptyset$, are shown in the table. It is easy to see that, for any X_i and Y_k, we can get $D(X_i, Y_k)$ from $D_K(X_i)$ as follows: $D(X_i, Y_k)$ is the set of individuals whose corresponding entries in $D_K(X_i)$ contain Y_k as an element, i.e.,

$$D(X_i, Y_k) = \{S_m | Y_k \in D_K(X_i)[m]\}. \quad (1)$$

For example, using the example dataset in Table 1, from Table 4(a), we know that $D(X_3, Y_4) = \{S_{10}, S_{13}, S_{19}\}$.

For SNP-pair (X_iX_j), let $O_{d_2}(X_iX_j) = [O_{d_2}(X_iX_j, Y_1),$ $O_{d_2}(X_iX_j, Y_2), \cdots, O_{d_2}(X_iX_j, Y_K)]$. From Theorem 5.1 and

	$0 \to 1$	$1 \to 0$	$0 \to 2$	$2 \to 0$	$1 \to 2$	$2 \to 1$
(X_3X_2)	$\emptyset$	$\emptyset$	$\{S_2\}$	$\{S_{12}, S_{15}, S_{20}\}$	$\{S_8, S_{10}\}$	$\emptyset$
(X_2X_5)	$\{S_7\}$	$\{S_{13}\}$	$\{S_3\}$	$\{S_1, S_4, S_6, S_{16}, S_{23}\}$	$\emptyset$	$\{S_8, S_{10}\}$
(X_5X_6)	$\emptyset$	$\emptyset$	$\{S_{16}\}$	$\{S_9, S_{24}\}$	$\{S_7\}$	$\emptyset$
(X_6X_1)	$\{S_4\}$	$\{S_8, S_{10}\}$	$\{S_5, S_9, S_{12}, S_{23}\}$	$\{S_2, S_3, S_{11}, S_{21}\}$	$\emptyset$	$\emptyset$
(X_6X_4)	$\emptyset$	$\emptyset$	$\emptyset$	$\{S_{16}, S_{18}\}$	$\{S_{10}\}$	$\{S_{21}\}$

Table 3. Genotype difference between the connected SNPs in the minimum spanning tree shown in Figure 1

individual id.	phenotype permutations
S_8	$\{Y_2, Y_3\}$
S_{10}	$\{Y_2, Y_3, Y_4, Y_5\}$
S_{13}	$\{Y_1, Y_2, Y_4, Y_5\}$
S_{19}	$\{Y_3, Y_4\}$

(a) $D_K(X_3)$ with empty entries omitted

	Y_1	Y_2	Y_3	Y_4	Y_5
$O_{d_2}(X_3X_5)$ after initializing	1	1	1	2	1
$O_{d_2}(X_3X_5)$ after updating for S_7	1	1	1	2	1
$O_{d_2}(X_3X_5)$ after updating for S_8	1	2	2	2	1
$O_{d_2}(X_3X_5)$ after updating for S_{10}	1	3	3	3	2
$O_{d_2}(X_3X_5)$ after updating for S_{13}	0	2	3	2	1

(b) Updating $O_{d_2}(X_3X_5)$ from $O_{d_2}(X_3X_2)$

Table 4. Updating $O_{d_2}(X_3X_5)$ from $O_{d_2}(X_3X_2)$ for all permutations in a batch mode

Equation (1), for any SNP-pair (X_iX_j) and an edge $(X_jX'_j) \in E(\mathcal{T})$, we can get $O_{d_2}(X_iX'_j)$ from $O_{d_2}(X_iX_j)$ using $D_K(X_i)$ and the genotype difference information associated with edge $(X_jX'_j)$. First, initialize $O_{d_2}(X_iX'_j) = O_{d_2}(X_iX_j)$. Next, for every m $(1 \leq m \leq M)$, if $Y_k \in D_K(X_i)[m]$, we update $O_{d_2}(X_iX'_j)$ as follows:

$$\begin{cases} \text{increase } O_{d_2}(X_jX'_j, Y_k) & \text{if } S_m \in (X_jX'_j)_{\{0\to1\}\cup\{2\to1\}}; \\ \text{decrease } O_{d_2}(X_jX'_j, Y_k) & \text{if } S_m \in (X_jX'_j)_{\{1\to0\}\cup\{1\to2\}}. \end{cases}$$

EXAMPLE 5.3. *Following Example 5.2, we consider the two SNP-pairs (X_3X_2) and (X_3X_5), with (X_2X_5) being an edge of the tree in Figure 1. Assume that $D_K(X_3)$ is as shown in Table 4(a), and $O_{d_2}(X_3X_2) = [1, 1, 1, 2, 1]$. From Table 3, the genotype difference on edge (X_2X_5) is $(X_2X_5)_{\{0\to1\}\cup\{2\to1\}} = \{S_7, S_8, S_{10}\}$, and $(X_2X_5)_{\{1\to0\}\cup\{1\to2\}} = \{S_{13}\}$. For individual $S_m \in \{S_7, S_8, S_{10}\}$ ($S_m \in \{S_{13}\}$), we need to increase (decrease) the corresponding values in $O_{d_2}(X_3X_2)$ according to $D_K(X_3)$. Table 4(b) shows the updating process for $O_{d_2}(X_3X_5)$. Initially, $O_{d_2}(X_3X_5) = O_{d_2}(X_3X_2)$. For individual S_7, since its corresponding entry in $D_K(X3)$, $D_K(X3)[7] = \emptyset$, $O_{d_2}(X_3X_5)$ remains unchanged. For individual S_8, $D_K(X3)[8] = \{Y_2, Y_3\}$, we increase the values of $O_{d_2}(X_3X_5, Y_2)$ and $O_{d_2}(X_3X_5, Y_3)$ by 1. Similarly, we increase and decrease the values in $O_{d_2}(X_3X_5)$ according to $D_K(X3)$ for S_{10} and S_{13}. For individual S_{19}, we do not have any update because $S_{19} \notin \{S_7, S_8, S_{10}\}$ and $S_{19} \notin \{S_{13}\}$. The final result is $O_{d_2}(X_3X_5) = [0, 2, 3, 2, 1]$.*

Note that to get the value of $O_{d_2}(X_iX_j)$, using a brute force approach, we need to scan a $(2 + K) \times M$ matrix consisting of the genotype of (X_iX_j) and permutations $\{Y_1, Y_2, \cdots, Y_K\}$ for the M individuals. In the previous example, to compute the value of $O_{d_2}(X_3X_5)$, the cost of the brute force approach is $(3 + 5) \times 24 = 192$. Using our approach, the total number of updates is $|D_K(X3)[8]| + |D_K(X3)[10]| + |D_K(X3)[13]| = 10$, which is significantly less than the cost of the brute force approach. More formally, given $D_K(X_i)$, the time complexity of updating $O_{d_2}(X_iX'_j)$ from $O_{d_2}(X_iX_j)$ is $O(w(X_jX'_j)K)$.

The procedure of updating $O_{d_2}(X_iX'_j)$ from $O_{d_2}(X_iX_j)$ can also be applied to update the remaining free variables $O_{d_3}(X_iX_j)$, $O_{f_2}(X_iX_j)$, $O_{f_3}(X_iX_j)$. Note that, to update $O_{f_2}(X_iX_j)$, $O_{f_3}(X_iX_j)$, we will need $F_K(X_i)$, which can be defined in a similar way to that of $D_K(X_i)$: for each individual S_m, we record in $F_K(X_i)[m]$ the set of phenotypes satisfying $(X_i = 2 \wedge Y_k = 1)$.

Algorithm 1: The TEAM Algorithm

```
Input: SNPs $X' = \{X_1, X_2, \cdots, X_N\}$, phenotype
       permutations $Y' = \{Y_1, Y_2, \cdots, Y_K\}$
Output: $\mathscr{T}(X_iX_j, Y_k)$ for all possible two-locus tests
compute and store all single-locus contingency tables;
build minimum spanning tree $\mathcal{T}$;
for every $X_i \in L(\mathcal{T})$, do
    compute $D_K(X_i)$ and $F_K(X_i)$;
    compute $O_{d_2d_3f_2f_3}(X_iX_a)$;
    compute $\mathscr{T}(X_iX_a, Y_k)$ $(1 \leq k \leq K)$ and output;
    $EnumStack.push(O_{d_2d_3f_2f_3}(X_iX_a))$;
    while $EnumStack \neq \emptyset$ do
        $O_{d_2d_3f_2f_3}(X_iX_j)$ = $EnumStack.pop()$;
        for every $X'_j = adj(X_j)$ do
            update $O_{d_2d_3f_2f_3}(X_iX'_j)$ from $O_{d_2d_3f_2f_3}(X_iX_j)$;
            compute $\mathscr{T}(X_iX'_j, Y_k)$ $(1 \leq k \leq K)$ and output;
            $EnumStack.push(O_{d_2d_3f_2f_3}(X_iX'_j))$;
        end
    end
    delete $X_i$ from $\mathcal{T}$;
end
```

6 THE TEAM ALGORITHM

TEAM examines SNP pairs through a double loop, where the outer loop visits a leaf node at a time, and the inner loop traverse the rest of the tree, starting from the parent node of the leaf.

Let $O_{d_2 d_3 f_2 f_3}(X_i X_j) = [O_{d_2}(X_i X_j), O_{d_3}(X_i X_j), O_{f_2}(X_i X_j), O_{f_3}(X_i X_j)]$. Let $L(\mathcal{T}) \in V(\mathcal{T})$ be the set of leaf nodes of the minimum spanning tree $\mathcal{T}$. For any *leaf* node $X_i \in L(\mathcal{T})$, let $AP(X_i) = \{(X_i X_j) | i \neq j, X_j \in V(\mathcal{T})\}$. Let X_a be the parent node of X_i. Since all SNPs are connected in $\mathcal{T}$, once we have $O_{d_2 d_3 f_2 f_3}(X_i X_a)$, we can update all $O_{d_2}(X_i X_j) \in AP(X_i)$ by enumerating the edges in $E(\mathcal{T})$ in a breath-first traversal starting from X_a.

EXAMPLE 6.1. *Consider the tree in Figure 1. Let* $X_i = X_3$ *and* $X_a = X_2$. *We have* $AP(X_3) = \{(X_3X_2), (X_3X_5), (X_3X_6), (X_3X_1), (X_3X_4)\}$. *Starting from* X_3, *a breadth first search will enumerate edges* $\{(X_2X_5), (X_5X_6), (X_6X_1), (X_6X_4)\}$, *which can be utilized to update* $O_{d_2 d_3 f_2 f_3}(X_i X_j)$ *for the SNP-pairs in* $AP(X_3)$.

Once the SNP-pairs in $AP(X_i)$ have been processed, we delete X_i from $L(\mathcal{T})$, and repeat the same process for another leaf node. The overall algorithm is summarized in Algorithm 1. Given the SNPs $X' = \{X_1, X_2, \cdots, X_N\}$, phenotype permutations $Y' = \{Y_1, Y_2, \cdots, Y_K\}$, we first enumerate and store all single-locus contingency tables. We then build the minimum spanning tree $\mathcal{T}$, with genotype difference associated with each edge. For leaf node X_i, we compute $D_K(X_i)$, $F_K(X_i)$, and $O_{d_2 d_3 f_2 f_3}(X_i X_a)$. This information is then used to incrementally update $O_{d_2 d_3 f_2 f_3}(X_i X_j')$ for all SNP-pairs in $AP(X_i)$. After processing $AP(X_i)$, we delete X_i from $\mathcal{T}$ and repeat the procedure for the remaining leaf nodes.

Time Complexity: The time complexity on generating all single-locus contingency tables and building the minimum spanning tree is $O(MNK)$ and $O(MN^2)$ respectively. The time complexity to compute $D_K(X_i)$ and $F_K(X_i)$ for all SNPs is $O(MNK)$. The total updating cost for all $AP(X_i)$ is $O(W_{\mathcal{T}}NK)$. Thus the overall time complexity of TEAM is $O(MNK + MN^2 + W_{\mathcal{T}}NK)$. Note that the complexity of the brute force approach is $O(MN^2K)$. The number of SNPs N is the dominant factor.

Space Complexity: The dataset size is $O(M(N + K))$. The space needed to store all single-locus contingency tables is $O(NK)$. The size of tree $\mathcal{T}$ is $O(W_{\mathcal{T}})$. The size of $D_K(X_i)$ and $F_K(X_i)$ is $O(MK)$. Thus the total space complexity of TEAM is $O(M(N + K) + K(N + M) + W_{\mathcal{T}})$.

Note that we can do incremental computation using any exploration order. TEAM utilizes minimum spanning tree to update the contingency tables. The reason is that the cost of such update depends on the difference between the SNPs. The more similar they are, the lower the cost. Since minimum spanning tree has the minimum weight $W_{\mathcal{T}}$ over all spanning trees, using it to guide the computation leads to optimal efficiency. It is not absolutely necessary to use a minimum spanning tree. As long as the tree is close to a minimum spanning tree, we should expect good performance. An implementation issue in building the minimum spanning tree is that we need $O(N^2)$ space to store all pair-wise differences between the SNPs. In practise, we divide the SNPs into sub-groups of equal size. A minimum spanning tree is built for each group. Then the sub-trees are merged to a larger tree by randomly connecting leave nodes. The tree built in this way is an approximate minimum spanning tree. Our focus in this paper is not to build an optimal minimum spanning tree, but to use the tree structure for efficient updating. Please refer to Eisner (1997); Graham and Hell (1985) for surveys on minimum spanning tree construction. In the experiments, we show the performance evaluation using different spanning trees.

7 EXPERIMENTAL RESULTS

In this section, we present extensive experimental results on the performance of the TEAM algorithm. TEAM is implemented in C++. We first evaluate the efficiency of TEAM. Then we present the findings of epistasis detection in simulated human genome-wide study.

7.1 Efficiency Evaluation

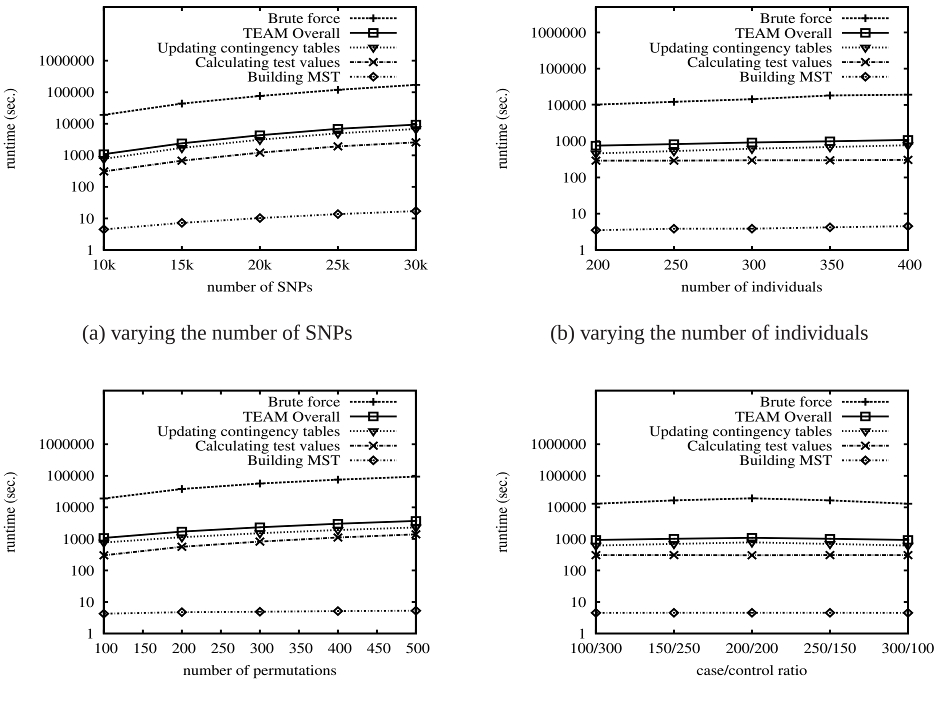
We use both simulated human datasets and real mouse datasets for the efficiency evaluation experiments. The experiments are performed on a 2.6 GHz PC with 8G memory running Linux system.

Human data: The human datasets are generated by the simulator Hapsample (Wright *et al.*, 2007), which is publicly accessible from the website `http://www.hapsample.org`. We evaluate the performance of TEAM by comparing it with the brute force approach since there is no previous algorithm readily applicable to human datasets. Note that the brute-force approach is very time consuming, we use a moderate number of SNPs and permutations in the experiments so that the brute-force approach can finish in a reasonable amount of time. Unless otherwise specified, the default experimental setting is the following: #individuals = 400, #SNPs=10,000, #permutations=100, and the case/control ratio is 1. These experimental settings are chosen to demonstrate the efficiency gain offered by TEAM over the brute-force implementation. TEAM can handle much larger datasets. The performance of TEAM is expected to follow the same trends presented in this section.

TEAM contains three major components: building the minimum spanning tree, updating the contingency tables, and calculating the actual test values. Note that TEAM can be applied to any statistics defined on the contingency table. With different statistics, the only difference in runtime would be caused by the last component calculating the statistics. In the experiments, we choose chi-square test as our statistic. Figure 2 shows the running time comparison of TEAM and the brute-force approach using different experimental settings. The y-axis is in logarithm scale. In these figures, we also show the detailed runtime of these three components.

Table 5 shows the percentage of individuals pruned by TEAM under different experimental settings. Since in theory we can update the contingency tables in any exploration order, in the table, we also show the pruning effect of using a random spanning tree and a linear spanning tree to guide the updating process. The random spanning tree is generated by starting from a randomly picked SNP and growing edges that connect the remaining SNPs in a random order. The linear tree is a single path connecting all SNPs sequentially. From the table, we can see that TEAM prunes more effectively than the other two updating methods. In the table, we also show the ratio of the tree weights and the size of the SNP dataset, i.e., $W_{\mathcal{T}}/(M \times N)$, which is a determining factor of the pruning ratio. Note that varying the number of permutations and the case/control ratio does not effect the tree being built.

Figures 2(a) depicts the runtime comparison when varying the number of SNPs. TEAM is more than an order of magnitude faster than the brute-force approach. Among the three components of TEAM, the procedures on building the minimum spanning tree



(a) varying the number of SNPs

(b) varying the number of individuals

(c) varying the number of permutations

(d) varying the case/contral ratio

Fig. 2. Comparison between TEAM and the brute-force approach on human datasets under various experimental settings

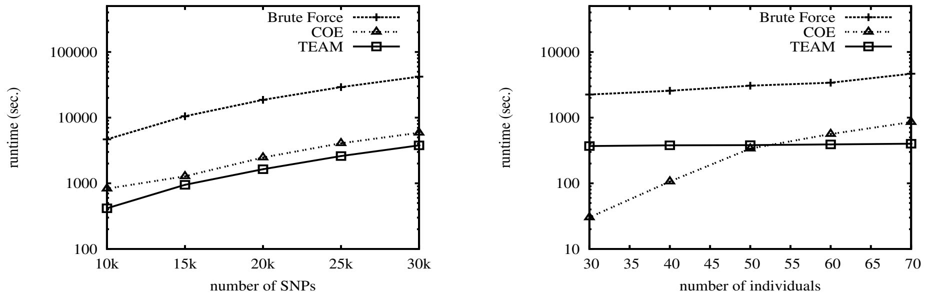
		TEAM		Updating by Random Tree		Updating by Linear Tree	
	Settings	**Tree weight**	**Pruning ratio**	**Tree weight**	**Pruning ratio**	**Tree weight**	**Pruning ratio**
# SNPs	10k	17.721%	94.104%	53.326%	88.722%	53.158%	89.210%
	20k	18.692%	93.981%	52.881%	88.895%	52.851%	89.390%
	30k	19.314%	93.802%	53.011%	88.823%	52.946%	89.380%
# Individuals	200	16.641%	94.376%	53.358%	88.749%	53.179%	89.205%
	300	17.342%	94.209%	53.343%	88.730%	53.142%	89.213%
	400	17.721%	94.104%	53.326%	88.722%	53.158%	89.210%
# Permutations	100	17.721%	94.104%	53.326%	88.722%	53.158%	89.210%
	300	17.721%	94.105%	53.326%	88.724%	53.158%	89.212%
	500	17.721%	94.104%	53.326%	88.724%	53.158%	89.212%
Case/control ratio	100/300	17.721%	97.049%	53.326%	94.355%	53.158%	94.599%
	200/200	17.721%	94.104%	53.326%	88.722%	53.158%	89.210%
	300/100	17.721%	97.049%	53.326%	94.355%	53.158%	94.599%

Table 5. The tree weight and the proportion of the individuals pruned by TEAM on the human datasets

and calculating test values only take a small portion of the total runtime of TEAM. The runtime of TEAM is dominated by the cost of updating the contingency tables. As will be shown later, TEAM prunes most of the individuals when updating the contingency tables. In Figures 2(b), 2(c), and 2(d), we can also observe a similar one to two orders of magnitude speedup of TEAM over the brute force approach when varying the number of individuals, the number of permutations, and the case/control ratio.

Mouse data: The mouse datasets is extracted from a set of combined SNPs from the 10k GNF (`http://www.gnf.org/`) mouse dataset and the 140k Broad/MIT mouse dataset (Wade and Daly, 2005). This merged dataset has 156,525 SNPs for 71 mouse strains. The missing values in the dataset are imputed using NPUTE (Roberts *et al*., 2007). We compare TEAM and the recently proposed COE (Zhang *et al*., RECOMB2009) algorithm, which is specifically designed for association study in mouse datasets. The default experimental setting is as follows: #individuals = 70, #SNPs=10,000, #permutations=100, and the case/control ratio is 1.

Figure 3 shows the comparison results. In the figure, we also plot the runtime of the brute force approach. Figure 3(a) shows



(a) varying the number of SNPs

(b) varying the number of individuals

Fig. 3. Comparison between TEAM, COE, and the brute force approach on mouse datasets under various experimental settings

Dataset	Significant SNP-Pair	Chromosome and Location	FDR	FWER
1	(rs768529, rs3804940)*	(chr1: 51946762, chr3: 7520545)	0.00067	0
	(rs768529, rs756084)	(chr1: 51946762, chr3: 7536149)	0.00067	0
	(rs768529, rs779742)	(chr1: 51946762, chr3: 7558058)	0.00067	0
	(rs768529, rs1872393)	(chr1: 51946762, chr3: 7546236)	0.00067	0.004
	(rs768529, rs779744)	(chr1: 51946762, chr3: 7555121)	0.00067	0.004
	(rs768529, rs6764561)	(chr1: 51946762, chr3: 7514592)	0.00067	0.004
2	(rs10495728, rs521882)*	(chr2: 22811773, chr8: 16688797)	0.004	0.004
3	(rs1016836, rs2783130)*	(chr10: 31935845, chr13: 79068161)	0	0
4	(rs648519, rs1012273)*	(chr11: 98972936, chr16: 58525067)	0.002	0.002

Table 6. Identified significant SNP-pairs in the simulated human GWAS datasets

the runtime of the three approaches when varying the number of SNPs. It is clear that both TEAM and COE are orders of magnitude faster than the brute force approach. TEAM is about twice faster than COE. Figure 3(b) shows the runtime comparison when varying the number of individuals. From the figure, COE is more suitable for datasets having small number of individual. As the number of individuals increases, the TEAM algorithm becomes more efficient than COE. Note that in human study, the number of individuals usually ranges up to thousands, much larger than that in typical mouse datasets.

7.2 Epistasis Detection in Simulated Human GWAS

In this section, we report the results of epistasis detection using simulated human GWAS data generated by Hapsample. In total, we generate 4 datasets, each of which has 112,036 SNPs for 250 cases and 250 controls. In each dataset, a disease causal interacting SNP-pair is embedded. The embedded SNP-pairs are: (rs768529, rs3804940) in dataset 1, (rs10495728, rs521882) in dataset 2, (rs1016836, rs2783130) in dataset 3, and (rs648519, rs1012273) in dataset 4. We use standard chi-square test with 500 permutations. Similar results can be found by using likelihood-ratio test.

With an overall FDR threshold of 0.005, Table 6 shows the identified significant SNP-pairs using TEAM. TEAM successfully identified the embedded SNP-pairs in all simulated datasets. The embedded SNP-pairs are labelled with stars ”*”. The table shows the SNP loci on the genome. For example, in dataset 1, we embed SNP-pair rs768529 and rs3804940, which are located on chromosome 1 at position 51946762 base-pair and chromosome 3 at 7520545 base-pair respectively. The FWER for each reported SNP-pair is also shown. Note that, for a SNP-pair, a FDR (or FWER) value of 0 indicates that permutation tests do not generate any test value larger than value of the reported SNP-pair. In dataset 1, except for the embedded SNP-pair (rs768529, rs3804940), 5 other SNP-pairs are also reported. One of the embedded SNP, rs768529, is involved in all the 5 pairs. A closer look at the other SNPs in the reported SNP-pairs shows that they are all adjacent to the embedded SNP rs3804940. The normalized linkage disequilibrium (Lewontin and Kojima, 1960) between rs3804940 and the other 5 SNPs are D'(rs3804940, rs756084)= 1, D'(rs3804940, rs779742)= 0.477, D'(rs3804940, rs1872393)= 0.442, D'(rs3804940, rs779744)= 0.442, and D'(rs3804940, rs6764561)= 0.454, indicating there is strong linkage disequilibrium between them.

8 CONCLUSION AND FUTURE WORK

The large number of SNPs genotyped in the genome-wide scale poses great computational challenges in two-locus epistasis detection. The permutation test used for proper error rate controlling makes the problem computationally even more intensive. In this paper, we propose an efficient algorithm, TEAM, for epistasis detection human GWAS. TEAM has the same strength as the recently developed epistasis detection methods, i.e., it guarantees to find the optimal solution. Compared to existing methods, TEAM is more efficient in large sample study, and offers broader applicability. Existing methods designed for homozygous SNPs cannot be used for human data where most SNPs are heterozygous. TEAM, on the other hand, can handle both homozygous and heterozygous SNPs. Since it exhaustively enumerate all SNP-pairs, TEAM can be used to control the FWER and the FDR, both of which are widely used in controlling error in GWAS; while previous methods only control the FWER. Existing methods need to exam the formulation of the statistic. TEAM is focused on efficiently updating contingency tables rather than any specific statistic. It can therefore be use for any statistical test based on contingency table regardless of its formulation.

In this paper, we focus on the disease phenotypes which can be represented as binary variables. Many association studies involve phenotypes measured as continuous variables. We will investigate how to apply the idea of the current algorithm to quantitative phenotypes in the future study.

ACKNOWLEDGEMENT

The work is partially supported by NSF awards IIS0448392, IIS0812464.

REFERENCES

Balding, D. J. (2006). A tutorial on statistical methods for population association studies. *Nature Reviews Genetics*, 7(10):781–791.

Carlborg, O., Andersson, L., and Kinghom, B. (2000). The use of a genetic algorithm for simultaneous mapping of multiple interacting quantitative trait loci. *Genetics*, 155:2003–2010.

Churchill, G.A. and Doerge, R.W. (1994). Empirical threshold values for quantitative trait mapping. *Genetics*, 138(3):963–971.

Cormen, T.H., Leiserson, C.E., Rivest, R.L., and Stein, C. (2001). *Introduction to Algorithms*. MIT Press and McGraw-Hill.

Dudoit, S., and Laan, M.J. (2008). *Multiple testing procedures with applications to genomics*. Springer.

Eisner, J. (1997). State-of-the-art algorithms for minimum spanning trees: A tutorial discussion. *Manuscript,University of Pennsylvania*.

Evans, D.M., Marchini, J., Morris, A.P., and Cardon, L.R. (2006). Two-stage two-locus models in genome-wide association. *PLoS Genetics*, 2: e157.

Graham, R.L. and Hell, P. (1985). On the history of the minimum spanning tree problem. *Ann. History Comput.*, 7:43–57.

Hirschhorn, J.N. and Daly, M.J. (2005). Genome-wide association studies for common diseases and complex traits. *Nature Reviews Genetics*, 6:95–108.

Hoh, J., and Ott, J. (2003). Mathematical multi-locus approaches to localizing complex human trait genes. *Nature Reviews Genetics*, 4:701–709.

Hoh, J. *et al*. (2000). Selecting snps in two-stage analysis of disease association data: a model-free approach. *Annals of Human Genetics*, 64:413–417.

Lewontin, R.C., and Kojima, K. (1960). The evolutionary dynamics of complex polymorphisms. *Evolution*, 14(4):458–472.

Miller., R.G. (1981). *Simultaneous Statistical Inference*. Springer Verlag New York.

Musani, S.K., Shriner, D., Liu, N., and et al. (2007). Detection of gene x gene interactions in genome-wide association studies of human population data. *Human Heredity*, 63(2):67–84.

Nelson, M.R., Kardia, S.L., Ferrell, R.E., and Sing, C.F. (2001). A combinatorial partitioning method to identify multilocus genotypic partitions that predict quantitative trait variation. *Genome Research*, 11:458–470.

Ritchie, M.D., Hahn, L.W., Roodi, N., and et al. (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *American Journal of Human Genetics*, 69:138–147.

Roberts, A., McMillan, L., Wang, W., and et al. (2007). Inferring missing genotypes in large snp panels using fast nearest-neighbor searches over sliding windows. In *Proc. ISMB*.

Saxena, R., Voight, B.F., Lyssenko, V., and et al. (2007). Genome-wide association analysis identifies loci for type 2 diabetes and triglyceride levels. *Science*, 316:1331–1336.

Scuteri, A., Sanna, S., Chen, W.-M., and et al. (2007). Genome-wide association scan shows genetic variants in the fto gene are associated with obesity-related traits. *PLoS Genetics*, 3(7):1200–1210.

The Wellcome Trust Case Control Consortium. (2007). Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447:661–678.

Wade, C.M., and Daly, M.J. (2005). Genetic variation in laboratory mice. *Nature Genetics*, 37:1175–1180.

Weedon, M., Lettre, G., Freathy, R., and et al. (2007). A common variant of hmga2 is associated with adult and childhood height in the general population. *Nature Genetics*, 39:1245–1250.

Westfall, P.H., and Young, S.S. (1993). *Resampling-based Multiple Testing*. Wiley, New York.

Wright, F.A., Huang, H., Guan, X., Gamiel, K., and et al. (2007). Simulating association studies: a data-based resampling method for candidate regions or whole genome scans. *Bioinformatics*, 23(19):2581–2588.

Yang, C., He, Z., Wan, X., Yang, Q., Xue, H., and Yu, W. (2009). SNPHarvester: a filtering-based approach for detecting epistatic interactions in genomewide association studies. *Bioinformatics*, 25(4):504–511.

Zhang, X., Pan, F., Xie, Y., Zou, F., and Wang, W. (2009). COE: a general approach for efficient genome-wide two-locus epistatic test in disease association study. In *Proc. RECOMB*.

Zhang, X., Zou, F., and Wang, W. (2008). FastANOVA: an efficient algorithm for genome-wide association study. In *Proc. KDD*.

Zhang, X., Zou, F., and Wang, W. (2009) FastChi: an efficient algorithm for analyzing gene-gene interactions. In *Proc. PSB*.

APPENDIX

Proof of Theorem 3.1

PROOF. From the four contingency tables shown in Table 2, it is easy to get the following linear equation system:

$$\begin{pmatrix}
1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1\\
1&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&1&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&1&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&1\\
1&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1
\end{pmatrix}
\begin{pmatrix}
O_{a_1}\\ O_{a_2}\\ O_{a_3}\\ O_{b_1}\\ O_{b_2}\\ O_{b_3}\\ O_{c_1}\\ O_{c_2}\\ O_{c_3}\\ O_{d_1}\\ O_{d_2}\\ O_{d_3}\\ O_{e_1}\\ O_{e_2}\\ O_{e_3}\\ O_{f_1}\\ O_{f_2}\\ O_{f_3}
\end{pmatrix}
=
\begin{pmatrix}
O_A\\ O_B\\ O_C\\ O_D\\ O_E\\ O_F\\ O_G\\ O_H\\ O_I\\ O_J\\ O_L\\ O_O\\ O_S\\ O_P\\ O_V\\ O_T\\ O_Q\\ O_W\\ O_R\\ O_U\\ O_Z
\end{pmatrix}$$

The rank of the above linear system is 14. We thus take 14 rows $\{4, 6, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21\}$, which form a full rank matrix. The row reduced echelon form of this non-redundant linear system is

$$\left(\begin{array}{cccccccccccccccccc|l}
1&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&1&1&O_S-O_W+O_D+O_F\\
0&1&0&0&0&0&0&0&0&0&-1&0&0&0&0&0&-1&0&O_P-O_V\\
0&0&1&0&0&0&0&0&0&0&0&-1&0&0&0&0&0&-1&O_G-O_U\\
0&0&0&1&0&0&0&0&0&0&-1&-1&0&0&0&0&0&0&O_T-O_D\\
0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&0&O_Q\\
0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&O_H\\
0&0&0&0&0&0&1&0&0&0&-1&-1&0&0&0&0&-1&-1&O_W-O_D-O_F\\
0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&1&0&O_V\\
0&0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&1&O_U\\
0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&O_D\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&-1&-1&O_R-O_F\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&O_O\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&O_L\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&O_F
\end{array}\right)$$

Thus we have the following solution:

$$\begin{pmatrix}
O_{a_1}\\ O_{a_2}\\ O_{a_3}\\ O_{b_1}\\ O_{b_2}\\ O_{b_3}\\ O_{c_1}\\ O_{c_2}\\ O_{c_3}\\ O_{d_1}\\ O_{e_1}\\ O_{e_2}\\ O_{e_3}\\ O_{f_1}
\end{pmatrix}
=
\begin{pmatrix}
O_S-O_W+O_D+O_F\\ O_P-O_V\\ O_G-O_U\\ O_T-O_D\\ O_Q\\ O_H\\ O_W-O_D-O_F\\ O_V\\ O_U\\ O_D\\ O_R-O_F\\ O_O\\ O_L\\ O_F
\end{pmatrix}
-
\begin{pmatrix}
1\\ -1\\ 0\\ -1\\ 1\\ 0\\ -1\\ 1\\ 0\\ 1\\ 0\\ 0\\ 0\\ 0
\end{pmatrix} O_{d_2}
-
\begin{pmatrix}
1\\ 0\\ -1\\ -1\\ 0\\ 1\\ -1\\ 0\\ 1\\ 1\\ 0\\ 0\\ 0\\ 0
\end{pmatrix} O_{d_3}
-
\begin{pmatrix}
1\\ -1\\ 0\\ 0\\ 0\\ 0\\ -1\\ 1\\ 0\\ 0\\ -1\\ 1\\ 0\\ 1
\end{pmatrix} O_{f_2}
-
\begin{pmatrix}
1\\ 0\\ -1\\ 0\\ 0\\ 0\\ -1\\ 0\\ 1\\ 0\\ -1\\ 0\\ 1\\ 1
\end{pmatrix} O_{f_3}$$

Clearly, only four variables $\{O_{d_2}, O_{d_3}, O_{f_2}, O_{f_3}\}$ are free. Once the values of these free variables are known, the observed frequencies of remaining events in the two-locus contingency table are also known.

Proof of Theorem 5.1

PROOF. It suffices to show that

$$D(X_i, Y_k) \cap Q(X_i, X_j') = [D(X_i, Y_k) \cap Q(X_i, X_j)] \cup [D(X_i, Y_k) \cap ((X_j X_j')_{\{0\to1\}\cup\{2\to1\}})] - [D(X_i, Y_k) \cap ((X_j X_j')_{\{1\to0\}\{1\to2\}})].$$

This is the same as to show that

$$Q(X_i, X_j') = Q(X_i, X_j) \cup ((X_j X_j')_{\{0\to1\}\cup\{2\to1\}}) - ((X_j X_j')_{\{1\to0\}\{1\to2\}}).$$

This is clearly true, hence completes the proof.