Mining massive SNP data for identifying associated SNPs and uncovering gene relationships

Amy Webb\textsuperscript{1,*}, Aaron Albin\textsuperscript{1,2*}, Zhan Ye\textsuperscript{3}, Majid Rastegar-Mojarad\textsuperscript{3}, Kun Huang\textsuperscript{1}, Jeffrey Parvin\textsuperscript{1}, Wolfgang Sadee\textsuperscript{4}, Lang Li\textsuperscript{5}, Simon Lin\textsuperscript{3}, and Yang Xiang\textsuperscript{1†}

\textsuperscript{1}Department of Biomedical Informatics, The Ohio State University, Columbus, OH 43210. 
\textsuperscript{2}Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210. 
\textsuperscript{3}Biomedical Informatics Research Center, Marshfield Clinic Research Foundation, WI 54449. 
\textsuperscript{4}Department of Pharmacology, The Ohio State University, Columbus, OH 43210. 
\textsuperscript{5}Center for Computational Biology and Bioinformatics, Indiana University, Indianapolis, IN 46202.

Received on XXXXX; revised on XXXXX; accepted on XXXXX.

Associate Editor: XXXXXXX

ABSTRACT

Motivation: Studies on SNP correlations have been focused on SNPs located on the same chromosome since SNPs on different chromosomes are expected to segregate randomly. However, we conjecture that there exist associations among SNPs, even for SNPs on different chromosomes. Therefore, our goal is to find SNPs that coexist in a significant number of samples, especially the long-range SNP associations, and the relationships among these associated SNPs and corresponding genes. This problem is computationally challenging and motivates us to design an efficient data mining algorithm to efficiently mine SNP associations from massive SNP data.

Results: A data mining workflow and an efficient algorithm FCI-RC are developed for mining SNP associations. By applying our method on the original SNP data and random chromosome permutation data, we demonstrate that our method is able to find non-random SNP associations across multiple chromosomes. Among the large amount of associated SNPs identified by our method, many of them involve multiple chromosomes. Some SNP associations also suggest novel relationships among the corresponding genes, and some may imply biological and disease mechanisms related to corresponding genes.

Source code and results will be made publicly available upon the publication of the manuscript.

Contact: yxiang@bmi.osu.edu

1 INTRODUCTION

The exploration of SNPs, or Single Nucleotide Polymorphisms has provided the ability to systematically search across an entire genome for variation associated with traits and disease. In typical Genome-wide Association Studies, millions of SNPs are measured to capture the common variation in a population and the distribution of genotype calls for each SNP is compared to a phenotype (Sachidanandam et al., 2001) These studies have identified an overwhelming number of associations between SNPs and phenotypic traits (Hindorff et al., 2009; Johnson and O’Donnell, 2009). Alternative to GWAS, phenome-wide association study (PheWAS) makes a reverse approach and has successfully identified genetic variants for association with phenotypes (Pendergrass et al., 2013).

Researchers can use a subset of SNPs to represent the larger genetic variation due to linkage disequilibrium (LD (Reich et al., 2001)) leading to limited haplotype diversity. LD refers to the coinheritance of different SNPs leading to a reduction in the total number of SNP combinations or haplotypes observed in a population. SNPs are inherited together along a chromosome and after recombination, nearby SNPs are more likely to be inherited together and thus exist on a common haplotype. Catalogued listings of LD provide a pairwise measurement of SNP correlation. Studies of linkage disequilibrium among SNPs have focused on SNPs located on the same chromosome since SNPs on different chromosomes are expected to segregate randomly. Thus correlations that occur between SNPs on separate chromosomes will occur due to reasons other than recombination.

It is worthwhile to point out that several machine learning methods have been successfully used in identifying combination of SNPs that are associated with diseases or phenotypes. For example, SVM, random forest, and neural network methods have been used to identify SNP combinations that are associated with type 2 diabetes (Ban et al., 2010), childhood allergic asthma (Tomita et al., 2004), rheumatoid arthritis (Liu et al., 2011), and other diseases (Ritchie et al., 2003). However, these methods need to focus on a small set of candidate SNPs, often on candidate disease genes, in order to identify SNP combinations for targeted diseases or phenotypes. In addition, they also needs training datasets which are not always available. Thus, these methods are not suitable for

\textsuperscript{*}The two first authors contribute equally.

\textsuperscript{†}to whom correspondence should be addressed
identifying SNP associations from massive SNP data that are not disease specific.

In contrast, our goal is to mine genome-wide SNP data for SNP lists that have a common genotype with identical allele information over a significant population. The majority of investigations of LD between SNPs do not consider long distance relationships, specifically whether correlations exist between distant SNPs and how to efficiently identify those correlations. Such a correlation may suggest some potential relationships among these SNPs and possibly among the related genes.

As we will see in the next section, our problem can be converted to multiple associate rule mining problems. For example, for one allele type BB, if we consider each sample (individual) a transaction, and each SNP with this allele type an item, then we obtain a transactional data. By applying association rule learning method on a transactional data, we may be able to find itemsets with a minimum support level $p\%$, i.e. itemsets that exist in at least $p\%$ of samples. Since SNPs from the same haplotype are likely to form associated SNP lists, it is not surprising if our method detects them. Thus, we focus on SNP lists that span multiple chromosomes. In order to show that our mining effort on SNP lists across multiple chromosomes is meaningful, we use the random chromosome permutation test (Sections 2.4 and 3.1) to demonstrate that our method is able to identify non-random associations of SNPs on multiple chromosomes.

The associate rule mining algorithms target on identifying frequent itemsets (fi), frequent closed itemsets (fcii), or maximal frequent itemsets (mfi) (Han et al., 2006). As we will discuss in the following, these mining problems are NP-hard, and none of the associate rule mining algorithms (such as Apriori (Borgelt, 2003) or MAFIA (Burdick et al., 2005)) or further summarization algorithms (such as Hyper (Xiang et al., 2011) or Krimp (Vreeken et al., 2011)) are able to effectively mine data with similar orders of magnitude as the SNP data. The computational cost of identifying all fi, fcii, or mfi for a large dataset may far exceed the available computational resources (i.e., CPU power, memory limit, and hard disk limit). Given this challenge, we design a resource constraint mining algorithm for mining associated SNP lists. Our method is not only flexible in mining large SNP data, but also has a basic quality guarantee on the mining results. By applying our method to the massive SNP data, we not only identified associated SNP lists corresponding to a more complex haplotype structure, but also reveals interesting gene-gene interactions corresponding to these SNP associations.

2 METHODS

We use the publicly available SNP dataset GSE26105 (224 normal human livers) (Innocenti et al., 2011) to demonstrate our methods. To enable efficient association rule mining on the dataset, we first decompose it into three sub datasets SNP_AA, SNP_AB, and SNP_BB as described in Section 2.1. Then we design and apply the novel FCI-RC algorithm on these sub datasets to obtain associated SNP lists (Section 2.2). In Section 2.3, we merge the results obtained for each sub datasets and get extended results of SNP associations between type AA and AB, and between type AB and BB.

Figure 1 provides an illustration of the general workflow of our SNP mining methods.

2.1 SNP data decomposition

The GSE26105 dataset contains 620,902 SNPs and 224 samples. Each SNP has three types, AA, AB, BB, plus NC for each sample. Thus, the dataset forms a matrix of $620,902 \times 224$, with each entry being one of the four categories. To enable efficient mining on this dataset, we divide it into three transactional sets: SNP_AA, SNP_AB, and SNP_BB, which records the samples with type being AA, AB, and BB, respectively. For example, in SNP_AA, it records the samples with type AA for each SNP. After the division, we use the associate rule mining approach as described below to identify associated SNPs.

2.2 Resource constraint frequent closed itemset Mining

Mining frequent closed itemset has been a focus for many data mining applications (Xiang et al., 2012). However, it is a NP-hard problem to list all frequent closed itemset. This can be easily inferred from the NP-hardness of the maximum edge biclique problem (Peeters, 2003). The current methods of mining frequent closed itemset generally use branch-and-bound approaches to reduce the search space. MAFIA (Burdick et al., 2005) is one of such algorithms that have been widely used. However, to our knowledge, none of the available frequent itemset mining algorithms are able to effectively handle transactional data with sizes similar to the SNP data in this work. Given the NP-hardness nature of the mining task, we are interested in designing an algorithm that can leverage the available computing resource in identifying major frequent closed itemsets from transactional databases that model the SNP data (i.e., SNP_AA, SNP_AB, and SNP_BB). Thus, we develop a Frequent Closed Itemset mining algorithm under Resource Constraint (FCI-RC), which is able to use given resource quotas to identify frequent closed itemsets from a very large transaction database (TDB).

In the following, we use $\mathcal{I}$ to denote the complete set of items (i.e., SNPs), and $\mathcal{T}$ to denote the complete set of transactions (i.e., samples). We use $\mathcal{I}$ to denote an itemset (i.e., a SNP list for a given allele type), and $\mathcal{T}(\mathcal{I})$ denote its supporting transactions (i.e., the list of samples containing this allele type at all SNPs of $\mathcal{I}$). Correspondingly, $\mathcal{I}$ and $\mathcal{T}(\mathcal{I})$ are their vector formats. Let $\mathcal{U} = \{(\mathcal{I}, \mathcal{T}(\mathcal{I}))| \mathcal{I} \in \mathcal{S}\}$ where $\mathcal{S} \subset 2^\mathcal{I}$. We define bit merge operation on $\mathcal{U}$ to be:

$$\text{bitmerge}(\mathcal{U}) = (\alpha, \beta)$$

where $\alpha = OR_{\mathcal{I} \in \mathcal{U}} \mathcal{I}$ and $\beta = AND_{\mathcal{I} \in \mathcal{U}} \mathcal{T}(\mathcal{I})$. 

![Figure 1: The workflow diagram of mining an input SNP data for identifying associated SNPs.](image-url)
The pseudo code of FCI-RC is given in Algorithm 1, which takes $TDB$ (transactional database), $\minsup$ (minimum support), and $\iter_{\text{out}}$ (the iteration threshold for output) as the input parameters. The algorithm starts with $U$, which initially contains every single item in $I$ as its itemsets. Then, the algorithm iteratively merge the itemsets in $U$ until $U$ is empty. In each iteration, an itemset is considered seeds, if it is merged into an itemset in $X$, which will be the $U$ for the next iteration, or if it is merged into an itemset in $Y$, which will be outputted during this iteration. All non-seed itemsets will be outputted at the end of this iteration. Since in each iteration, $U$ will be updated to $X$, which contains only clique of size $k^{\iter}$, Thus, we have the following important observation.

**Observation 1.** The while loop of Algorithm 1 runs no more than $\log_2 n$ iterations.

Observation 1 tells us that Algorithm 1 will converge in a very limited number of iterations. In the following, we will show that FCI-RC has two important features, as implied by Lemma 1 and Lemma 2.

**Algorithm 1 FCI-RC($TDB$, $\minsup$, $\iter_{\text{out}}$)**

1. Initialize $U; \iter = 1$;
2. while $U \neq \emptyset$
3. Initialize is_seed vector to be all true;
4. for all $U' \subseteq U$ where $|U'| = k$
5. $\gamma = \text{bitmerge}(U')$;
6. continue if $\text{SUPP}(\gamma) < \minsup$;
7. if $\text{size}(\gamma) = k^{\iter}$ then
8. continue if $\text{QuotaUpdate}(\gamma) == \text{false}$;
9. $X = X \cup \{\gamma\}$;
10. for all $y \in Y$
11. $Y = Y \setminus \{y\}$ if $\gamma$ covers $y$;
12. end for
13. else
14. $\text{cover} = \text{false};$
15. for all $x \in X \cup Y$
16. if $x$ covers $\gamma$ then
17. $\text{cover} = \text{true};$
18. else if $\gamma$ covers $x$ then
19. Delete $x$ from $Y$;
20. end if
21. end for
22. $Y = Y \cup \{\gamma\}$ if $\text{cover} == \text{false}$;
23. end if
24. end for
25. if $\iter > \iter_{\text{out}}$ then
26. Output (1) all non-seeds in $U$, i.e., itemsets in $U$ that were not involved in Step 9 and 22; (2) $Y$.
27. end if
28. $U = X; \iter = \iter + 1$;
29. end while

**Lemma 1.** Algorithm 1 FCI-RC successfully identifies all frequent closed itemset with size no less than $k$ when quota is unlimited.

**Proof.** To prove this theorem, we first claim that all frequent itemset with size $k^i$ where $i$ is a positive integer, is identified and saved in $X$ in the $i$th iteration in Algorithm 1. This can be proved using mathematical induction. It is clear that when $i = 1$ the claim is correct. Since each frequent itemset of $k^{i+1}$ items can be obtained by bit merge $k$ disjoint frequent itemsets of size $k^i$, we conclude that if the claim applies to $i = j$, it also applies to $i = j + 1$. Thus, a frequent itemset with size $k^i$ can be identified at $i$th iteration.

Now let us consider frequent closed itemsets of any sizes. Given a frequent closed itemset $C$, i.e., $i = \lceil \log_2 |C| \rceil$. It is easy to see that $k^i \leq |C|$ and $k^{i+1} > |C|$. Thus, it is easy to see that $C$ can be obtained by bit merge at most $k$ frequent itemsets of size $k^i$. Since $C$ is a frequent closed itemset, it cannot be covered by any other frequent itemset. Thus, $C$ will be outputted as an element in $Y$ at $i+1$th iteration if $k^i < |C|$, or outputted as a non-seed element at $i+1$th iteration if $k^i = |C|$.

Lemma 1 suggests that FCI-RC has the potential to find all the frequent closed itemsets if there is sufficient resource for the input data. Thus, we can adjust the quota for different datasets. For large datasets, we can set up a tight quota function to enable FCI-RC to finish in a reasonable amount of time. For small datasets, we can set up a loose quota function to enable FCI-RC to find more frequent closed itemsets or even the complete set of frequent closed itemsets. The quota function is flexible and independent of the main algorithm. In our implementation, we assign each item the same positive integer quota. If any item in the itemset $\gamma$ does not have a positive quota, the QuotaUpdate (step 8) will return false; otherwise, it will return true and the quota of each involved item will be reduced by one.

Although the quota setup can reduce the computational load of the FCI-RC algorithm thus enable it to mine frequent closed itemsets from massive transactional datasets, the mining results has a quality guarantee as suggested by Lemma 2. Although FCI-RC may skip many frequent closed itemsets when quota is limited, any frequent closed itemsets in the transactional database $TDB$ is completely covered by the output itemsets of FCI-RC.

**Lemma 2.** Let $R$ be the set of all itemsets outputted by Algorithm 1 FCI-RC. For any frequent closed itemset $I$ of $TDB$, we conclude that $I \subseteq \bigcup_{R \subseteq \mathbb{R}} R$.

**Proof.** By mathematical induction, we can show that for any itemset $I \in U$ at any iteration $p$, $I \subseteq \bigcup_{R \subseteq \mathbb{R}} R$. This is because if $I$ is not outputted at iteration $p$, according to Algorithm 1, $I$ must be merged with another itemset into an itemset in $U$ of iteration $p + 1$. By Observation 1, the Algorithm 1 will terminate in no more than $\log_2 n$ iterations. Thus, we conclude that $I$ or a superset of $I$ will be outputted in no more than $\log_2 n$ iterations.

Now we can prove Lemma 2 by contradiction. Assume there is a frequent closed itemset $I$ of $TDB$ such that $I \not\subseteq \bigcup_{R \subseteq \mathbb{R}} R$. It implies there exists an item $i \in I$ such that $i \not\in \bigcup_{R \subseteq \mathbb{R}} R$. Since $i$ is an itemset in $U$ at the first iteration, we conclude that this is a contradiction.

### 2.3 Merging Frequent Closed Itemset Mining results

We would like to extend our study from the association of same type of SNPs, to multiple type of SNPs. In particular, we would like to find the associations among SNP type AA and AB, and among SNP type AB and BB. By doing this we create clusters of SNPs where the allele of interest, A for AA/AB and B for AB/BB, is always present. This method identifies lists of SNPs based on allele rather than by genotype.

A set $I$ of SNPs is considered an associated SNPs of type AA and AB, if and only if there exists a list of samples $T$ such that for each SNP $s \in I$, its corresponding alleles in a given sample list $T$ are either type AA only, or type BB only. Similarly, we can define the associated SNPs for type AB and BB.

To find the associated SNPs, for example, between type AA and AB, we merge the results from FCI-RC algorithm. Assume mining SNPs of type AA and AB we obtain a collection of associated SNPs $\text{result}_{AA}$ and $\text{result}_{AB}$. For any two SNP associations $(I, T(I)) \in \text{result}_{AA}$ and $(J, T(J)) \in \text{result}_{AB}$, we merge them together into $(I \cup J, T(I \cap J))$ if and only if $|T(I) \cap T(J)| > f(\minsup)$. To ensure enough merged SNP lists are generated, in our experiments we set $f(\minsup)$ slightly smaller than $\minsup$, i.e., $f(\minsup) \approx \minsup - 10\% \times \text{Sample Size}$.
In order to perform random chromosome permutation test, we generated 10 random chromosome permutation datasets following the method described in Section 2.4. Then we apply our data mining method on these 10 random permutation datasets with minimum support levels ranging from 50% (112 samples) to 90% (202 samples). In all experiments for a minimum support level, the parameters of our data mining method are made strictly identical for all datasets (random or original) to ensure fairness of the comparison.

### 3.1 Random chromosome permutation test

Five sets of tests are carried out at minimum support levels from 50% to 90% on 10 random permutation datasets as well as on the original dataset. The results are classified according to the allele types AA, AB, and BB. Results of AA and BB are presented in Figures 3 to 8, in which X-axis is the minimum support for FCI-RC, and Y-axis is the percentage. From these figures we conclude that there is a higher percentage of SNP Lists that span multiple chromosomes from the original data than from the random permuted ones in every case. The difference is especially large when the support level or the chromosome number is high.

For allele type AB, no SNP lists are identified at support levels 80% and 90%, and no SNP lists that span 4 or more chromosomes are identified at support level 70%. The complete results are listed in Table 1, from which we can observe a huge advantage of original data over the random permutation data in producing multi-chromosome SNP lists. Similar huge advantages exist for merged results of allele type AA-AB and allele type AB-BB that were generated by following the method in Section 2.3. For succinctness, we omit these results in the manuscript.

The above results show that our method always identifies a larger percentage of SNP lists spanning multiple chromosomes on the original dataset than on the random permutation datasets. They provide a clear support that our method is able to identify non-random associations between SNPs on separate chromosomes.

### 3 RESULTS

We applied SNP decomposition and FCI-RC algorithm as described in Section 2 to the SNP data GSE26105 in order to identify associated SNPs. To remove noisy information, we also preprocess the SNP datasets SNP_{AA}, SNP_{AB}, and SNP_{BB} by core mining (Xiang, 2014) such that after the preprocess, an item (SNP) or a transaction (sample) will be further considered for mining if and only if it is associated with a sufficient number of transactions or items. When using the FCI-RC algorithm, we apply a flexible quota assignment approach. During the first few iterations, each item is assigned a unit quota so that FCI-RC will have a limited search space on large number of small frequent itemsets. After that, we make the quota unlimited and FCI-RC algorithm will be able to generate all possible frequent closed itemsets based on the results from the first few iteration. Using this strategy, we identified a large number of associated SNPs, and many of them are across multiple genes or even multiple chromosomes, as reported in Section 3.2 and Section 3.3.

### 3.2 Identified associated SNPs

In order to generate a large number of candidate SNP lists for our analysis, we set the minimum support \( \minsup = 67 \approx 30\% \) of 224 for mining SNPs of allele types AA, AB, and BB, and we set \( f(\minsup) = 44 \) for merging allele types AA AB, and AB BB. As a result, we identified a large number of associated SNP lists as reported in Table 2. A good number of these SNP lists span multiple
Figures 3 to 8: Blue columns represent the results generated from the original data, and red columns represent the averages of the results generated from 10 sets of random chromosome permutation data.

Table 1. Comparison of multi-chromosome percentages on AB SNP lists. The row “original” contains the results obtained from the original data, and the row “random” contains the average results obtained from the 10 sets of random permuted data. The title of each column is a support level for the Algorithm FCI-RC.

<table>
<thead>
<tr>
<th></th>
<th>2+ chr</th>
<th>3+ chr</th>
<th>4+ chr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50%</td>
<td>60%</td>
<td>70%</td>
</tr>
<tr>
<td>original</td>
<td>3.31%</td>
<td>33.33%</td>
<td>100%</td>
</tr>
<tr>
<td>random</td>
<td>0.076%</td>
<td>0%</td>
<td>0%</td>
</tr>
</tbody>
</table>

We would expect SNPs placed in clusters together with a short genomic distance between them to also have documented LD between them. To investigate the detection of local LD, we subdivided the clusters into bin SNPs where SNPs with less than 20 genotyped SNPs between them were placed together. This allowed the bin assignment to be independent of length. When investigating the length of bins within a cluster, it was found the largest bin from any cluster was 1010 Kb in length and located in a known inversion at 8p23.1 (Ma and Amos, 2012). While several clusters were identified containing SNPs from this region, one cluster contained 8 SNPs from a 137 kb region. Figure 9 shows the plot of LD across this 137 kb region to illustrate the level of correlation documented in Caucasians by The International HapMap project (The International HapMap Consortium, 2010) (Altshuler et al., 2010). The 8 SNPs from the cluster are indicated by the blue star. A lack of cluster containing all 8p23.1 SNPs suggests that this method is identifying a more complex haplotype structure than simply LD or perhaps requires a more stringent level of correlation.
<table>
<thead>
<tr>
<th>Cluster ID</th>
<th>Identified gene</th>
<th>Other genes</th>
<th>Biological links</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AGTR1</td>
<td>CD82, MRPL43</td>
<td>coexpression, miRNA</td>
</tr>
<tr>
<td>2</td>
<td>ALK</td>
<td>COMMD1, RGNEF</td>
<td>ligand binding, SOCS1 interactions, miRNA</td>
</tr>
<tr>
<td>3</td>
<td>ALMS1</td>
<td>CDC42, SAPS2 (PPP6R2)</td>
<td>coexpression, miRNA, functions: actin bundle assembly, organism growth, cell projection assembly, actin organization, spindle, microtubule</td>
</tr>
<tr>
<td>4</td>
<td>APOB</td>
<td>UIMC1, HK3</td>
<td>publications, coexpression, miRNA, drugs (1,5-anhydroglucitol, tungstopterin)</td>
</tr>
<tr>
<td>5</td>
<td>BTK/GLA</td>
<td>LAP3, KIAA1276</td>
<td>coexpression human/mouse, drug interaction, shared miRNA</td>
</tr>
<tr>
<td>6</td>
<td>BTK/GLA</td>
<td>Ipo9, LOD1, SHISA4</td>
<td>coexpression, miRNA, drug interaction, transcription factor</td>
</tr>
<tr>
<td>7</td>
<td>BTK/GLA</td>
<td>IL7R</td>
<td>overlapping biological processes, human phenotypes, coexpression, miRNA, drug interaction</td>
</tr>
<tr>
<td>8</td>
<td>BTK/GLA</td>
<td>FAT3</td>
<td>coexpression, miRNA</td>
</tr>
<tr>
<td>9</td>
<td>BTK/GLA</td>
<td>AIP13A5, HRA5S</td>
<td>coexpression, miRNA, drug interaction, transcription factor</td>
</tr>
<tr>
<td>10</td>
<td>C3</td>
<td>FKBP12-Exp2, FKBP1A, SARDH, SULT1B1</td>
<td>mouse phenotype hepatic necrosis, coexpression, miRNA, drugs</td>
</tr>
<tr>
<td>11</td>
<td>C7orf16 (PPP1R17)</td>
<td>ARPC2, GPR146</td>
<td>miRNA</td>
</tr>
<tr>
<td>12</td>
<td>CEP290</td>
<td>SH3R2F, IGS9F, C1orf1186</td>
<td>localized on cell surface, coexpression, miRNA</td>
</tr>
<tr>
<td>13</td>
<td>DNAH11</td>
<td>CDH18, CTINN3</td>
<td>cell adhesion, cadherin and Wnt signalling pathways, coexpression, miRNA</td>
</tr>
<tr>
<td>14</td>
<td>DNAH11</td>
<td>CDH18, IL13RA</td>
<td>miRNA</td>
</tr>
<tr>
<td>15</td>
<td>DNAH11</td>
<td>C9orf93 (CCDC171)</td>
<td>coexpression, miRNA</td>
</tr>
<tr>
<td>16</td>
<td>F5</td>
<td>C2CD2L, DEC1, NME7</td>
<td>coexpression, miRNA, publications: venous thrombosis and coexpression in breast</td>
</tr>
<tr>
<td>17</td>
<td>FADS2</td>
<td>EIF2R3, PTCH2, LONRF3</td>
<td>miRNA</td>
</tr>
<tr>
<td>18</td>
<td>HPSE2</td>
<td>CCDC147, CNNM1, NFKBI12 (TOSNL), SLC39A4</td>
<td>transcription factor, coexpression, miRNA, drugs</td>
</tr>
<tr>
<td>19</td>
<td>IKBKAP/EPB41L4</td>
<td>SH3R2F</td>
<td>coexpression, miRNA</td>
</tr>
<tr>
<td>20</td>
<td>JAK2/INS1L6</td>
<td>FLJ40082 (C9orf171)</td>
<td>JAK2/INS1L6 susceptibility locus for ulcerative colitis (PMID19915573)</td>
</tr>
<tr>
<td>21</td>
<td>JAK2/INS1L6</td>
<td>FLJ40082 (C9orf171), NFKBI12, SLC39A4</td>
<td>coexpression, miRNA, drug interaction (JAK2/NFKBI12)</td>
</tr>
<tr>
<td>22</td>
<td>KCN1</td>
<td>EIF2R3, PTCH2</td>
<td>human phenotype fever, transcription factor</td>
</tr>
<tr>
<td>23</td>
<td>PPARG/KALRN</td>
<td>DLL3, SUIPTSH</td>
<td>regulation of neurogenesis, glial cell differentiation, gliogenesis</td>
</tr>
<tr>
<td>24</td>
<td>PPARG/KALRN</td>
<td>ZNF30, ZNF792</td>
<td>coexpression, miRNA, transcription factor</td>
</tr>
<tr>
<td>25</td>
<td>PPARG/KALRN</td>
<td>ALDH9A1</td>
<td>coexpression, miRNA, drug interaction, transcription factor</td>
</tr>
<tr>
<td>26</td>
<td>SLCT2A3</td>
<td>GZFT1, KIAA0802 (SOCA2), CNT1</td>
<td>miRNA, transcription factor</td>
</tr>
<tr>
<td>27</td>
<td>XYLT1</td>
<td>IL7R</td>
<td>human phenotype vision loss, transcription factor, coexpression, drug (heparin)</td>
</tr>
<tr>
<td>28</td>
<td>XYLTI</td>
<td>ANKR6D, MDN1, PLEKHG1, KIAA0957, CANKS3</td>
<td>negative regulation of response to stimulus, coexpression</td>
</tr>
<tr>
<td>29</td>
<td>ABCA1/AGTPBP1</td>
<td>15 total genes including: MGST1, IFNAR2, ABCA1, IL10RB</td>
<td>related to drug 1-Methyl-3-isobutylxanthine, hepatitis B susceptibility, retinal degeneration, cholesterol transport</td>
</tr>
</tbody>
</table>

Table 3. Table lists gene clusters selected for genes enriched for involvement in cholesterol, hypertension, or small molecule transport. Listed is the gene identified based on function and the other genes in clusters containing that primary gene. Gene links were made through function enrichment analysis in toppgene.

### 3.3 Identified SNP-associative gene relations

To observe the gene relations corresponding to the SNP associations, we limited the consideration of clusters identified with SNPs within genes (exon, intron, UTR), represented by multiple genomic regions (assuming SNPs from the same region are in LD), and those genes related to hypertension and cholesterol as annotated by toppgene (Chen et al., 2007). We surveyed these 30 clusters to find biological links between genes. These clusters and the links between genes are listed in Table 3. Some clusters (Clusters 26 and 29) are results of combining several clusters sharing overlapping gene lists. For Cluster 29, it is a combination of more than 20 clusters all of which contain ABCA1 and AGTPBP1. Functional analysis of 15 genes within this cluster found that 4 genes were related to a drug, 1-Methyl-3-isobutylxanthine. Other genes in the cluster shared GO terms and disease involvement including “hepatitis B susceptibility”, “retinal degeneration”, “cholesterol transport”, as listed in Table 3. Thus, we further explored the disease implications of each cluster of SNPs using GeneAnswers (Feng et al., 2012) and Disease Ontology annotation of the human genome (Peng et al., 2013). The results are illustrated in Table 4.

These results suggest that SNP associations may imply biological and disease mechanisms related to corresponding genes, and our method can be used to identify these relations.
testing, and higher order correlation testing exponentially increases the number of tests (Cordell, 2009). A variety of statistical machine learning methods have been proposed to detect gene-gene interactions (Koo et al., 2013). The method presented here is independent of prior knowledge and allows for the detection of multi-SNP clusters that bypasses the multiple statistical tests needed for pairwise comparisons.

It is possible that the detection of correlations between distant SNPs could be by chance for alleles with high frequency. The best opportunity for nonrandom correlation should come from the BB clusters as the minor allele of one SNP is less likely to be detected concurrently with the minor allele of other SNPs. This correlation is less likely to happen by chance when the minor allele frequency is low.

By limiting clusters to a single genotype or allele, we identify haplotypes of SNPs inherited together. Detected clusters, whether biologically relevant or not, have implications for association studies. If correlations exist between distant SNPs, then a detected association signal from a SNP could represent the variation found in any of the regions included in the cluster. Commonly in association studies a single SNP is expected to tag or represent surrounding, unmeasured SNPs due to LD. If there is an unknown, long-distance correlation for an associated SNP, the actual causal variant could exist in the correlated genomic region. By understanding the more complex correlations in the genome, we have a greater possibility of identifying a causal variant.

Typical searches for SNP correlations are limited to nearby SNPs or those on the same chromosome as positional LD is dependent on genomic distance. We have shown examples where this method has detected previously described localized LD. The proposed method provides an efficient manner to detect correlations between groups of SNPs regardless of distance or chromosome. When applied to a dataset with phenotype information, this approach could identify sets of genes which contribute to a trait or disease. It is important to note that even if the detected correlation between SNPs is true, statistical correlation with a trait may not equate with biological correlation or causation. Even so, this SNP mining method combined with biological knowledge may provide insight into the mechanics of human disease.

The proposed method is able to detect localized LD between SNPs as supported by patterns of LD documented in HapMap. But the ultimate goal is to identify links between genes in order to infer gene-gene interactions. The majority of links shown in the example clusters are related to coexpression, miRNA binding, and transcription factor binding. Realistically, the gene lists for these types of connections are very large, so links such as these could easily happen by chance. As said before, the real value of these clusters is to find an association with a phenotype. If we found a cluster associated with a phenotype, a real functional inference could be made. As it is, we can detect clusters of SNPs within genes and find functional links between the genes.

SNPs in enhancer regions can have a potential regulatory effect on certain regions but the target regions for the majority of putative enhancer sites are unknown (Visel et al., 2007). This method presents a unique opportunity to highlight potential enhancer interactions. Two regulatory region databases were used to survey the detected SNP clusters—ORegAnno is an open source database of known regulatory elements documented in scientific publications and Vista (Griffith et al., 2008) is an enhancer specific database.

### Table 4

<table>
<thead>
<tr>
<th>Disease Clusters</th>
<th>Total</th>
<th>Cancer</th>
<th>Atherosclerosis</th>
<th>Colon Cancer</th>
<th>Leukemia</th>
<th>Diabetes</th>
<th>AA</th>
<th>AB</th>
<th>BB</th>
<th>AA/AB</th>
<th>AB/BB</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total</strong></td>
<td>137</td>
<td>16</td>
<td>49</td>
<td>3</td>
<td>3</td>
<td>11</td>
<td>7</td>
<td>2</td>
<td>10</td>
<td>23</td>
<td>65</td>
</tr>
<tr>
<td><strong>Cancer</strong></td>
<td>16</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>10</td>
<td>0</td>
<td>5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td><strong>Atherosclerosis</strong></td>
<td>49</td>
<td>4</td>
<td>45</td>
<td>5</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td><strong>Colon Cancer</strong></td>
<td>3</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td><strong>Leukemia</strong></td>
<td>3</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>Diabetes</strong></td>
<td>11</td>
<td>1</td>
<td>11</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 9: Cluster AB:6658: The figure shows the plot of LD across a 137 kb region on chromosome 8 to illustrate the level of correlation of SNPs identified within a cluster as documented by The International HapMap project for the CEPH population (HapMap Data Rel 24/phaseII Nov08, on NCBI B36 assembly, dbSNP b126). The 8 SNPs from the cluster are indicated by the blue star. The deepness of red indicates the strength of correlation. Despite the gap in LD seen by the gray bar, SNPs in the upper part of the region are linked with SNPs in the lower part of the region.

Table 4. Row 1: Total number of disease clusters identified by GeneAnswers (Feng et al., 2012) and Disease Ontology annotation of the human genome (Feng et al., 2013). Rows 2-6: Number of clusters of major identified diseases.

### 4 DISCUSSION

The method proposed here detects clusters of correlated SNPs representing the patterns of local LD and correlations between SNPs that should be segregating independently. This method was developed to detect gene-gene interactions from SNP genotype data defined as a correlation between genes that are found to affect a trait with the goal of understanding the function of the genes, pathways that affect disease, and potential treatments (Koo et al., 2013). When individuals share the genotype for several SNPs across genes also share a trait, we can infer there is a correlation between inheriting these regions with the trait. Typical searches for correlation test pairwise combinations of SNPs. These methods suffer from the need for many tests requiring statistical multiple testing correction or computationally intensive permutation
curated by computational and experimental approaches. From the ORegAnno database, we identify 2103 clusters from 1583 regions. With the Vista database, we identify 89 clusters from 51 regions. Of possible biological interest, SNP rs1538115 in the intronic region of ALDH9A1 shows a connection to an enhancer region (rs1538101 in Vista hs307) on chromosome 9 located within the intronic region of BNC2. ALDH9A1 is involved in alcohol metabolism and may be of interest in liver biology. Similarly, a connection was found between SNP rs12922744 in ABCCI, a transporter gene involved in multidrug resistance, and SNP rs12922698 located 33 Mb downstream within the intergenic space between CYLD and SALL1.

5 CONCLUSION

We have successfully identify SNP associations from massive SNP data by an approach consisting of SNP data decomposition, resource-constraint mining, and merging of the mining results. We have shown that such an approach is able to identify non-random SNP associations across multiple chromosomes. The identified SNP associations are valuable resources for us to understand the corresponding gene relationships and disease mechanisms. These results cannot be obtained by traditional LD analysis. In the future, we would like to advance the research by developing algorithms that are more productive in mining massive SNP data for long-range SNP associations, and by applying the method to mine disease specific SNP data.

ACKNOWLEDGEMENT

The project described was partially supported by the Clinical and Translational Science Award (CTSA) program, through the NIH National Center for Advancing Translational Sciences (NCATS), grant UL1TR000427, and by the U01GM092655 grant from National Institute of General Medical Sciences. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

Funding: NIH UL1TR000427, NIGMS U01GM092655

REFERENCES


Nature, 467(7311), 52–58.


IEEE Trans Knowl Data Eng., 17(11), 1490–1504.


BMC bioinformatics, 8(1), 392.


Nature Reviews Genetics, 10(6), 392–404.


Nucleic acids research, 36(suppl 1), D107–D113.


Proceedings of the National Academy of Sciences, 106(23), 9362–9367.


PloS genetics, 7(5), e1002078.


BMC medical genetics, 10(1), 6.


Human genetics, 129(5), 473–485.


PloC one, 7(7), e40224.


PloS genetics, 9(1), e1003087.


Nucleic acids research, 41(D1), D553–D560.


Nature, 411(6834), 199–204.


BMC bioinformatics, 5(1), 120.


Nucleic acids research, 35(suppl 1), D88–D92.


Data Mining and Knowledge Discovery, 23(1), 169–214.


