Genome analysis

Predictive rule inference for epistatic interaction detection in genome-wide association studies

Xiang Wan\textsuperscript{1,*}, Can Yang\textsuperscript{1}, Qiang Yang\textsuperscript{2}, Hong Xue\textsuperscript{3}, Nelson L.S. Tang\textsuperscript{4} and Weichuan Yu\textsuperscript{1,*}

\textsuperscript{1}Department of Electronic and Computer Engineering, \textsuperscript{2}Department of Computer Science, \textsuperscript{3}Department of Biochemistry, The Hong Kong University of Science and Technology and \textsuperscript{4}Laboratory for Genetics of Disease Susceptibility, Li Ka Shing Institute of Health Sciences, The Chinese University of Hong Kong, Hong Kong, China

Received on May 18, 2009; revised on October 15, 2009; accepted on October 28, 2009

ABSTRACT

Motivation: Under the current era of genome-wide association study (GWAS), finding epistatic interactions in the large volume of SNP data is a challenging and unsolved issue. Few of previous studies could handle genome-wide data due to the difficulties in searching the combinatorially explosive search space and statistically evaluating high-order epistatic interactions given the limited number of samples. In this work, we propose a novel learning approach (SNPRuler) based on the predictive rule inference to find disease-associated epistatic interactions.

Results: Our extensive experiments on both simulated data and real genome-wide data from Wellcome Trust Case Control Consortium (WTCCC) show that SNPRuler significantly outperforms its recent competitor. To our knowledge, SNPRuler is the first method that guarantees to find the epistatic interactions without exhaustive search. Our results indicate that finding epistatic interactions in GWAS is computationally attainable in practice.

Availability: http://bioinformatics.ust.hk/SNPRuler.zip

Contact: eixiangw@ust.hk, eeyu@ust.hk

Supplementary information: Supplementary data are available at Bioinformatics online.

1 INTRODUCTION

In genetic epidemiology, a genome-wide association study (GWAS) is an examination of genetic variations across a given genome, designed to identify genetic associations with observable traits (Thomas and Teri, 2008). GWASs use high-throughput genotyping technologies to assay hundreds of thousands of single nucleotide polymorphisms (SNPs) and relate them to clinical conditions and measurable traits. A SNP generally refers to a stable variant of a single base in human genome. SNPs are often used as genetic markers to identify the causes and risks of diseases. It has been well established in the field that SNP profiles characterize a variety of diseases. By investigating the SNP profiles associated with a disease trait, researchers would be able to reveal genes relevant with the disease. However, most complex diseases have not been amenable to conventional methods based on univariate analysis of association between individual SNP and the disease trait. Researchers start to suspect that many common diseases in humans are not caused by one genetic variation within a single gene, but determined by complex epistatic interactions among multiple genes. Epistasis is generally defined as interactions among different genes. In the literature, discussion of epistasis has been considerably confused by different definitions and assumptions and by the use of the same terminology to describe different statistical as well as biological concepts (Cordell, 2002). In essence, epistasis refers to departure from marginal effects of different genetic loci in a way that they combine to cause disease. Marginal effects are commonly used in practice to quantify the effect of individual variables on an outcome of interest. Finding epistatic interactions is a topic of current interest in molecular and quantitative genetics. Many significant epistatic interactions have been found contributing to diseases such as breast cancer (Ritchie \textit{et al}, 2001), coronary heart disease (Nelson \textit{et al}, 2001) and Alzheimer’s disease (Zabcken \textit{et al}, 2001). It is worth noting that the studies in both Ritchie \textit{et al} (2001) and Nelson \textit{et al} (2001) found important interactions between loci that did not display noticeable marginal effects. Since the sheer volume of data in genome-wide studies makes it difficult to analyze data by hand, a feasible computational model is in expectation to detect and evaluate interaction patterns which are most likely associated with the disease.

The properties of epistatic interactions have been investigated for decades and many methods have been proposed: multifactor dimensionality reduction (MDR; Ritchie \textit{et al}, 2001) was developed as a non-parametric data mining strategy, which generated a new attribute by pooling genotypes from multiple SNPs. Bayesian epistasis association mapping (BEAM, Zhang and Liu, 2007) designed a Bayesian marker partition model and used Markov Chain Monte Carlo (MCMC) sampling strategy to maximize the posterior probability of the model. Monte Carlo logic regression (Kooperberg and Ruczinski, 2005) combined the logic regression (Kooperberg \textit{et al}, 2001) and MCMC in searching SNP interactions. The penalized regression (Park and Hastie, 2007) used a variant of logistic regression model with a quadratic penalization and then applied a forward stepwise procedure to find the best fitting model. Stepwise method (Marchini \textit{et al}, 2005) combined the...
logistic regression and forward stepwise procedure. HapForest
(Chen et al., 2007) proposed a forest-based approach to use the
haplotype information in heuristic filtering. Backward genotype-
trait association (BGTA; Zheng et al., 2006) proposed a screening
algorithm to repeatedly evaluate a large number of random marker
subsets on the basis of a new defined measurement. MegaSNPHunter
(Wan et al., 2009) searched for interactions between SNPs by using
a tree-based learning model.

These methods perform well on small datasets. Unfortunately,
most of them have difficulties to handle genome-wide data due
to the combinatorially explosive search space. A comprehensive
comparison study was conducted in Liu and Zhang (2007) among
BEAM, the stepwise logistic regression, logic regression and MDR.
The comparison results demonstrated that BEAM is more powerful
in GWASs. In BEAM, the problem of detecting epistatic interactions
is formulated as a Bayesian marker partition problem and MCMC
is used to estimate posterior probabilities of associated markers
and/or epistatic interactions. BEAM has successfully demonstrated
its capability of handling large datasets in simulation experiments.
But it did not provide convincing evidence using real data. The
authors applied BEAM to an Aged-related Macular Degeneration
(AMD) dataset with 96 cases and 50 controls (Klein et al., 2005)
and did not report any interactions. Their explanation was that
the number of samples was not sufficient to detect statistically
significant interactions. In this work, we tested BEAM on large-scale
real data containing 2000 cases and 1500 controls from Wellcome
Trust Case Control Consortium (WTCCC, 2007). While the sample
size is significantly larger, BEAM did not report any statistically
significant interactions. This made us suspect that the number of
samples may not be the reason. Our conjecture is that the data
from real studies are too complex to be formulated by one Bayesian
marker partition model in BEAM.

Given a dataset with a huge amount of SNPs and a limited number
of samples, it is difficult to detect and evaluate epistatic interactions
in a traditional manner. Typical feature selection methods often use
univariate ranking to reduce the number of relevant features. These
methods may filter out those SNPs with weak marginal effect, while
their joint behavior may significantly contribute to disease traits.
Figure 1 presents a toy example to illustrate such a case. While
neither SNP1 nor SNP2 shows marginal effect between cases and
controls, the joint distribution of SNP1 and SNP2 shows strong
contrast between cases and controls.

In this article, we introduce a novel learning approach based on
the predictive rule learning to detect epistatic interactions. The
predictive rule describes the relationships between feature and class
variables, which can be applied to predict the class label for new data.
There are two reasons using rules learning to infer interactions: first,
each epistatic interaction implicitly contains some predictive rules
(explained in the following section). Second and more importantly,
finding and evaluating rules are much easier and faster than finding
and evaluating interactions. Our proposed learning approach seeks
to identify those rules and uses them to infer possible epistatic
interactions. Although a predictive rule may not guarantee the
existence of epistatic interaction among SNPs in the rule, we can first
narrow down the range of possible interactions using our approach
and then seize statistically significant interactions in the next stage
by using a quick procedure of validation based on statistic tests.

2 METHODOLOGY

2.1 Background

One popular measurement of two SNP interaction is the mutual
information, which is defined as

$$I(S_i; S_j) = \sum_{\zeta} p(S_i, S_j | \zeta) \log \frac{p(S_i, S_j | \zeta)}{p(S_i | \zeta) p(S_j | \zeta)}$$

where $\zeta$ is the sample label, $S_i$ is the genotype of $i$-th SNP and $p(\cdot)$ is
the joint probability. Accordingly, the measurement of the high-order
interaction is defined as

$$I(S_1, ..., S_j) = \sum_{\zeta} p(S_1, ..., S_j | \zeta) \log \frac{p(S_1, ..., S_j | \zeta)}{p(S_1 | \zeta) ... p(S_j | \zeta)}$$

In theory, finding epistatic interactions requires an exponential
search space $|O|^{2^n}$, where $P$ is the number of SNPs.
(Almuallim and Dietterich., 1994). In this article, we propose
a predictive rule learning approach to solve this problem. Our
motivation is that each epistatic interaction implicitly contains
some predictive rules. Given an epistatic interaction among SNPs
$S_1, ..., S_j$ with $I(S_i; S_j) > 0$, there exists at least one item
$p(S_1, ..., S_j | \zeta) > p(S_i | \zeta) p(S_j | \zeta)$ (otherwise $I(S_1, ..., S_j | \zeta) \leq 0$, which
provides a predictive rule $\zeta \rightarrow C$: $S_i \rightarrow S_j \rightarrow \zeta$ (formal definition is given
in the following section). The connection between the interaction
and the predictive rule is also visualized in the toy example shown
in Figure 1. In this figure, six cells in the 3 x 3 table represent six
predictive rules. For example, the top left cell gives a predictive
rule as ‘If SNP1 of a given sample is AA and SNP2 of this sample is
BB, then there is a probability of 0.75 that this sample is normal’. To
detect this epistatic interaction, it is not necessary to find all included
rules. As long as one of them is identified, the corresponding
interaction will be determined with a fast validation. This is the
big advantage of using predictive rule learning in finding epistatic
interactions. Therefore, we propose to first use the predictive rule
learning to select predictive rules with high confidence and then use
some statistic tests to identify interactions from the selected rules.

2.2 Rule learning

A SNP $S$ is considered to be a set of independent values
{1, 2, 3} corresponding to homozygous reference genotype (AA),
heterozygous genotype (Aa), and homozygous variant genotype
2.2.1 Predictive rule measurement

Given a predictive rule \((r, \zeta): s_1 \land \cdots \land s_n \Rightarrow \zeta\), suppose \(\zeta = 0\) and its contingency table is obtained from observations in Table 1.

<table>
<thead>
<tr>
<th>(\zeta = 0)</th>
<th>(\zeta \neq 0)</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>(X(r))</td>
<td>(a)</td>
<td>(b)</td>
</tr>
<tr>
<td>(X(\neg r))</td>
<td>(c)</td>
<td>(d)</td>
</tr>
<tr>
<td>Total</td>
<td>(a + c)</td>
<td>(b + d)</td>
</tr>
</tbody>
</table>

One typical measurement of rule relevance is to use \(\chi^2\) statistic, which is defined as

\[
\chi^2(r, \zeta) = C \cdot \frac{(ad - bc)^2}{(a+b)(c+d)(a+d)(b+c)}
\]

where \(C = \frac{(a+b+c+d)}{4}\) is a constant.

Suppose \(a \geq b\) (otherwise we just need to switch the two columns of the contingency table). Let \(\delta = b/(a + \gamma)\). We obtain a new measurement \(U(r)\) (Please check the Supplementary Material for the deduction) from the \(\chi^2\) statistic,

\[
U(r, \zeta) = \frac{(R - \delta)^2}{(1 - \delta)(1 + \delta)}
\]

where \(\delta \in [0, 1]\) and \(R = \frac{ad}{bc}\) (a constant). We name \(U(r, \zeta)\) rule utility. It is proportional to the \(\chi^2\) statistic with a constant \((a+b+c+d)/(a+b+c+d)\).

A predictive rule \((r, \zeta)\) is closed if and only if \(U(r, \zeta) > U(r - s_i, \zeta)\). Here, \(r - s_i\) means removing \(s_i\) from \(r\).

Given a dataset \(M = (S_1, \ldots, S_n, \zeta)\), our target is to find all closed predictive rules \((r, \zeta)\), where \(U(r, \zeta) > R\) with \(R\) being a user-specified threshold to remove weak rules. Predictive rule learning is a popular topic in machine learning. Many algorithms (e.g., Liu et al., 1998; Li et al., 2001; Yin and Han, 2003) have been developed for learning predictive rules. However, they are not suitable for our task due to two critical reasons:

1. Most of the algorithms focus on achieving high classification accuracy. For SNP datasets including a large number of features (SNPs) and a limited number of samples, these algorithms are prone to overfitting.
2. Many algorithms first generate all possible rules and then prune weak rules in the next stage. These algorithms cannot be applied to large-scale data because the number of possible rules is exponential to the data size.

In this article, our goal is to precisely rank predictive rules in order to identify true interactions. Concretely, we first define a new relevance measurement \(U(\cdot)\) to quantify the relevance or importance of predictive rules. Then, we derive an upper bound of \(U(\cdot)\) for expanding a given rule. Based on the derived upper bound, we finally design a branch and bound algorithm for the predictive rule learning.

2.2.2 SNPRuler

Based on the upper bound in Equation (6), we design a branch and bound algorithm (named ‘SNPRuler’) presented in Algorithm 1 to infer predictive rules. Instead of directly finding interactions, our rule learning algorithm only needs to find one of the following items.
Algorithm 1: SNPRuler: branch-and-bound rule-based learning

Given:
- \( L \): the maximum of rule length
- \( U_{\text{cut}} \): the minimum utility value of rule
- \( N \): the maximum number of leaf nodes

Procedure:
compute contingency table \( T \) for each SNP
root = [ ]
depth = 1
counter = 0
for SNP \( s \), do
  if \( U_{\text{max}}(s) > U_{\text{cut}} \) then
    root.addChild(\( s \))
  end if
end for
while depth < \( L \) do
  iterator = first node in current depth
  while iterator != NIL do
    for \( t \) \( \in T \) do
      if \( U_{\text{max}}(\text{iterator rule}\( t \)) > U_{\text{cut}} \) then
        child = createNode(\text{iterator rule}\( t \))
        iterator.addChild(child)
        counter = counter + 1
      end if
    end for
    if counter > \( N \) then
      Delete the bottom counter \( \times N \) ranked leaf nodes
    end if
    iterator = iterator.nextSibling
  end while
  depth = depth + 1
end while
Select rules with utility values above \( U_{\text{cut}} \)
Generate interactions from the search tree and evaluate interactions using \( \chi^2 \) statistic

many possible rules contained in each interaction and a validation process can easily identify it in the next stage. The key point of using rule learning is that we can use an upper bound to evaluate predictive rules. Such an approximation has not been seen in the field so far. Basically, our algorithm starts from a single literal (one SNP with a specified value in our case) and continues to expand it using rule learning is that we can use an upper bound to evaluate the number of generated rules in the search process is still exponential. Therefore, the parameter \( N \) is used in Algorithm 1 to control the number of leaf nodes in the search tree, which is equivalent to the number of generated predictive rules. In general, only the top \( N \) rules (ranked by utility values) will be kept in the search process. This parameter could be selected on the basis of available memory space. The utility threshold \( U_{\text{cut}} \) in Algorithm 1 is provided by users to remove weak rules, which can be computed with a specified significance level of \( \chi^2 \) test. The output of SNPRuler is a list of ranked interactions based on their \( \gamma^2 \) values. A post-processing step computes the raw \( P \)-value of each interaction by using the \( \chi^2 \) test and adjusts it by using the Bonferroni correction.

3 RESULTS
The performance of our approach is evaluated through comparative studies with existing works. Our goal is to discover epistatic interactions from genome-wide data. Among many methods recently proposed, BEAM is a very powerful one which can handle large-scale data and finish in a reasonable time. Besides, comprehensive comparison studies have been conducted by Zhang and Liu (2007) among BEAM, the stepwise logistic regression, logic regression and MDR. Therefore, we mainly compare our method with BEAM on using simulated data and three real genome-wide datasets from WTCCC using Affymetrix GeneChip 500K Mapping Array Set.

3.1 Results on simulated data
Simulation experiments are developed to provide the validation of our approach and BEAM in correctly determining the disease-associated interactions defined by epistatic models. To accomplish this, we design four cases in our simulation experiments.

3.1.1 Case I: disease loci with marginal effects
We use three two-locus epistatic models in Neuman and Rice (1992) and one three-locus epistatic model in Zhang and Liu (2007) (please check the Supplementary Materials for details).

We simulate test data based on four epistatic models under different parameter settings (the details are provided in the Supplementary Material) to compare the discrimination power between SNPRuler and BEAM. We use the same simulation program in BEAM with the same parameter settings to generate the data. The discrimination power is computed as the proportion of generated datasets in which the true epistatic interactions are identified. The output of SNPRuler is a list of ranked interactions based on their \( \gamma^2 \) statistics. For each test, the top one with adjusted \( P \)-value (Bonferroni correction) smaller than 0.3 is selected as the true interaction. BEAM also uses Bonferroni correction and the same \( P \)-value to select true interactions. The number of tests for Bonferroni correction is \( \frac{P(P-1)}{2} \) for two-way interaction, \( \frac{P(P-1)(P-2)}{3} \) for three-way interaction and so forth, where \( P \) is the number of SNPs in the data. The results obtained via BEAM roughly match those in Zhang and Liu (2007). Please note that the results of BEAM in the published paper were reported by using 0.1 as the significance threshold. In the released software, the significance threshold is set as 0.3. We use the setting in the released BEAM software when running the experiments.
Fig. 2. The performance comparison between SNPRuler (R) and BEAM (B) on four epistatic models with marginal effects. For each model, 100 datasets are generated. Under each parameter setting, 2000 samples (1000 cases and 1000 controls) and 4000 samples (2000 cases and 2000 controls) are simulated. The comparison shows that SNPRuler outperforms BEAM on four epistatic models with marginal effects except for one setting ($\lambda = 0.3, r^2 = 0.7, \text{MAF} = 0.2$) of Model 1. Here, $\lambda$ controls the marginal effects, $r^2$ controls the LD, $N_c$ is the number of cases and $N_u$ the number of controls.

The experimental results in Figure 2 show that:

1. SNPRuler performs the same or slightly better than BEAM, when disease loci present marginal effects (except for one setting [$\lambda = 0.3, r^2 = 0.7$, minor allele frequency (MAF) = 0.2] of Model 1, SNPRuler performs a little worse).
2. The power of both methods can be increased by increasing the sample size.
3. If the disease locus is unobserved, then it becomes more difficult to identify the locus by the linkage disequilibrium (LD) markers (the performance of both methods is worse in cases with $r^2 = 0.7$ than those in cases with $r^2 = 1.0$).

3.1.2 Case 2: disease loci without marginal effects A wide spectrum of interaction models without marginal effects have been discussed in Velez et al. (2007). In this experiment, we choose 60 pure epistatic models (no marginal effect) to compare the performance. The details of these models are available in the supplementary document. The heritability $h^2$ [see detailed definition in Velez et al. (2007)] controls the phenotypic variation of these 60 models, which ranges from 0.025 to 0.4 and MAF from 0.2 to 0.4. For each model, 100 datasets are generated. Each datasets consists of 1000 SNPs and includes 200 cases and 200 controls.

Figure 3 shows that SNPRuler is significantly superior to BEAM for detecting epistatic interactions without marginal effects. For the models with MAF = 0.2, 0.4 and $h^2 \geq 0.1$, the power of SNPRuler is $\sim 75\%$, while that of BEAM is $\sim 20\%$. The performance of two methods degrades as the heritability $h^2$ decreases. BEAM almost totally loses its power when MAF = 0.2 and $h^2 \leq 0.05$, while SNPRuler still maintains its power at $\sim 35\%$ for some models and performs better for models with MAF = 0.4. This comparison illustrates that BEAM has little chance to sample the simulated true pair if marginal effects are not present.

3.1.3 Case 3: large-scale simulation experiment To demonstrate the power of our approach on genome-wide data, we design this large-scale simulation experiment on the basis of our available resources. We use the same four epistatic models in Case 1. For each epistatic model, two settings (MAF = 0.2 and 0.5) are used for the MAFs of disease loci. The MAFs of non-disease loci are randomly selected between 0.1 and 0.5. For each setting, 100 datasets are generated. Each dataset contains 2000 samples (1000 controls and 1000 cases) and 10000 SNPs. Note that under this scale of simulation, most existing methods could not finish in a reasonable amount of time.

Figure 4 shows that SNPRuler performs much better than BEAM for the first setting, while it does slightly better for the second setting. Figure 4 also shows that the power of both methods increases with the increment of MAF of disease loci and the performance of BEAM is improved significantly. This is because increasing the MAF of disease loci makes those loci exhibit stronger marginal effects and hence easier to be detected. But in reality, the disease loci may not always possess distinguishable MAFs.

3.1.4 Case 4: null simulation The last simulation experiment is designed to approximate the type I error rates of both BEAM and SNPRuler. We permute the case-control status to generate 1000 datasets representing samples with no genetic effects. Each dataset...
Fig. 3. The performance comparison between SNPRuler (R) and BEAM (B) on 60 pure epistatic models (without marginal effects). For each model, 100 datasets are generated. Each dataset contains 400 samples (200 cases and 200 controls) and 1000 SNPs. The comparison shows that SNPRuler significantly outperforms BEAM on the 60 epistatic models with no marginal effects.

Fig. 4. Comparison between SNPRuler and BEAM on the large-scale simulated data. For each simulation setting, the power is calculated as the proportion of 100 datasets in which the true epistatic interactions are identified. Each dataset contains 2000 samples (1000 cases and 1000 controls) and 10 000 SNPs. λ controls the marginal effect.

3.2 Results on WTCCC data

The WTCCC is a collaboration of many British research groups. To date, the WTCCC has examined the genetic signals of seven common human diseases: rheumatoid arthritis, hypertension, Crohn’s disease, coronary artery disease, bipolar disorder and types 1 and 2 diabetes. We have obtained data from three studies (all of them contain ~500K SNPs): bipolar disorder study (1998 cases and 1504 controls), Crohn’s disease study (2005 cases and 1504 controls) and rheumatoid arthritis study (1999 cases and 1504 controls). Bipolar disorder is well known as manic depression and characterized by extreme mood states alternating between euphoric peaks and terrible depression. Crohn’s disease is a chronic and recurrent inflammatory disease of the intestinal tract. Rheumatoid arthritis is a chronic autoimmune disorder that usually affects the joints.

Both SNPRuler and BEAM are tested on these three WTCCC datasets. BEAM could not find any significant interaction, while our SNPRuler identifies many significant epistatic interactions (please check the Supplementary Materials for the complete results). The reason may be that the data from real studies are too complex to be formulated by one Bayesian marker partition model in BEAM and the distribution assumptions in BEAM may not be true. Please note that the reported \( P \)-values in the Supplementary Material are not adjusted by multiple test correction. It is easily observed that after multiple test correction such as Bonferroni correction, all reported SNPs will not be picked by univariate analysis while many of them still show significant interaction patterns. The top ranked interaction in the Crohn’s disease data is between rs7154773 and rs10130695. The individual \( P \)-values are 0.004 and 0.476, respectively, while the significance level of their interaction is \( 4.435 \times 10^{-43} \). Many reported genes, such as \( PLXNA2 \), \( PTPRT \) and \( PPM1A \), have strong connections with the target traits. The study in Mah et al. (2006) identifies semaphorin receptor \( PLXNA2 \) as a candidate susceptible to schizophrenia and bipolar disorder. The \( PTPRT \), known as the receptor type T of protein tyrosine phosphatase, is confirmed in Julia...
et al. (2008) to have strong association with rheumatoid arthritis. In Wrighton and Feng (2008), PPMIA is discussed in details about its critical role on the dephosphorylation of Smad proteins. It is well known that aberrant post-translational modifications of Smad proteins are associated with Crehn’s disease. Those genes are not reported in WTCCC (2007) because individual SNPs are not significant in statistics. The connection between our finding and previous works implies that some interactions reported by our method may have associations with diseases. More evidences from biological aspect are to be investigated.

In this experiment, we also use the pruning ratio to report the pruning efficiency of our approach. The pruning ratio for each internal node in the search tree is computed as the number of pruned branches over the number of all possible branches. The average pruning ratio is 99.1%. Regarding the computation time, it takes $\sim 2$ days for SNPRuler to complete the analysis of one WTCCC dataset on a PC (CPU: Intel 3.0 GHz and RAM 8 GB), while BEAM needs $\sim 10$ days to finish the same analysis.

### 3.3 The parameter setting of SNPRuler

There are three parameters in SNPRuler, including the maximum rule length $L$, the minimum utility value of rule $U_{cut}$ and the maximum number of leaf nodes $N$.

1. $L$ indicates the maximum depth of SNP interaction. In simulation studies, it is set to 2 for two-way interaction model and to 3 for three-way interaction model. In real studies, the setting of this parameter depends on the sample size. We use 5 for tests on WTCCC data.

2. $U_{cut}$ is particularly useful for real studies to speed up the search process by removing those weak rules. In simulation studies, $U_{cut}$ is set to 0. In real studies, we use the $P$-value 0.05 of $\chi^2$ test on the predictive rule as a significance threshold. Using our formulation, $U_{cut}$ is set as 0.001.

3. The setting of $N$ decides the size of search tree, which depends on the available memory and the data size. We use 50000 in simulation studies and 500000 in real studies.

### 4 DISCUSSION

#### 4.1 Relationship between SNPRuler and existing methods

- SNPRuler versus MegaSNPHunter (Wan et al., 2009). MegaSNPHunter is a hierarchical approach which searches for epistatic interactions by using a tree-based learning model. To handle the genome-wide data, MegaSNPHunter first partitions the whole genome into multiple short subgenomes. For each subgenome, MegaSNPHunter builds boosting tree classifiers based on multi-SNP interactions using 10-fold cross-validation (CV). MegaSNPHunter extracts relevant SNPs from these classifiers, merges them together and then repeats the same partition and interaction finding procedure. If the interacted SNPs are not located in the same subgenome, MegaSNPHunter requires that their marginal effects must be above the medium of marginal effects of their resided subgenomes. If either of interacted SNPs only has trivial marginal effect, MegaSNPHunter would have little chance to find them. Therefore, MegaSNPHunter only identified interactions which occur between closely placed SNPs (haplotype effects).

SNPRuler is designed to detect significant interactions in the absence of marginal effects. Instead of directly searching for interactions, SNPRuler only needs to find one of the many possible rules contained in one interaction and a validation process can easily identify it in the next stage. The key point of using rule learning is that we can find an upper bound to evaluate rules. Before adding a new SNP into a rule, we can use the computed upper bound to decide if adding it could possibly improve the current rule. Those SNPs without this possibility will be ignored in the search process. As the results of Case 1 in simulation experiments show, SNPRuler performs very well for models without marginal effects. For those models, MegaSNPHunter will perform much worse. For the WTCCC data, SNPRuler identified many interactions between distantly placed SNPs (gene effects).

- SNPRuler versus Monte Carlo (MC) logic regression (Kooperberg and Ruczinski, 2005). MC logic regression applied MCMC in fitting a logic regression model, which involves various combinations of SNPs (multiple logic trees) that are associated with the outcome of interest. Instead of identifying a single model, this approach tried to find a large number of alternative models that fit the data almost equally well as the single model. The drawback of MC logic regression is that if a significant interaction is explored in one of the models visited in the search, it is very likely that this interaction will be identified to be important. But if the SNPs composing this interaction do not jointly occur in any of these models (this happens more frequently for a genome-wide study), this interaction will not be found.

SNPRuler is a two-stage search method. It aims at finding interactions in GWASs. In the first stage, the rule searching algorithm is applied to find potential interactions. In the second phase, it uses the $\chi^2$ statistic to evaluate all selected pairs of SNPs and filter out those with low confidence. SNPRuler currently can only handle the conjunctive predictive rules.\(^1\) Finding the predictive rules including both conjunctive literals and disjunctive rules is much more difficult. We are still doing the investigation on this issue. A possible solution is to combine these two approaches together. SNPRuler can be first applied to reduce the number of SNPs and then the MC logic regression is invoked to find the complex rules. In this sense, these two approaches complement each other.

#### 4.2 The advantages of SNPRuler

The development of SNPRuler was triggered by the limitations of existing works on finding high-order SNP interactions from genome-wide data. Many existing methods either fail to report the statistically significant interactions due to the limited samples, or cannot complete in a reasonable amount of time due to the explosive search space. Although some recent approaches, such as BEAM, are able to handle the large-scale data, the performance of those

---

\(^1\)The conjunctive predictive rule is the association between the conjunction of $n$ literals, $s_1 \land s_2 \land \cdots \land s_n$, and a class label $\xi$, whereas the disjunctive predictive rule is the association between the disjunction of $n$ literals, $s_1 \lor s_2 \lor \cdots \lor s_n$, and a class label $\xi$. 
approaches is still sensitive to marginal effects of individual SNPs. SNP Ruler displays many advantages over existing methods:
- SNP Ruler detects epistatic interactions from genome-wide data without exhaustive enumeration;
- SNP Ruler uses a predictive rule learning algorithm to detect possible interactions. It is superior to univariate feature selection techniques in finding SNPs with weak marginal effect but significant joint effect;
- SNP Ruler is a non-parametric method and does not assume any prior distribution;
- SNP Ruler does not assume any particular epistasis model. This is very important for real studies because the patterns of SNP interactions are generally unknown and could be very complex;
- SNP Ruler could be applied to build a classifier for discriminating two classes of samples;
- SNP Ruler provides a list of ranked interaction based on their significance. For those experiments with large number of features and limited samples, strictly using multiple test correction may not report anything significant in statistics. In contrast, SNP Ruler still reports a ranked list.

4.3 The limitations of SNP Ruler

The current version of SNP Ruler is limited to detect those epistatic interactions containing conjunctive rules. The genetic heterogeneity model that consists of disjunctive rules, will be addressed in our future work. There are several other issues we need to consider in the future, including statistically evaluating interactions with sparse data, detecting spurious SNP effects and so on.

How to reduce false positive errors is a challenging problem in GWASs. Although our method does not directly address this issue, our method is able to reduce the number of possibly disease-associated epistatic interactions into a very small set and rank those epistatic interactions based on their relevances to the disease trait. Extra filters can be applied to remove false positives. We plan to incorporate the haplotype information and pathway information to help reduce false positive errors.

5 CONCLUSIONS

In this article, we propose a novel rule-based learning algorithm (SNP Ruler) to find epistatic interactions in GWASs. Our method first uses the predictive rule learning to narrow down possible interactions among SNPs and then captures true interactions using χ^2 statistic test. The rule-based strategy in our non-parametric learning approach enables our new method to search for interaction algorithms that consist of disjunctive rules, will be addressed in our future work. There are several other issues we need to consider in the future, including statistically evaluating interactions with sparse data, detecting spurious SNP effects and so on.

How to reduce false positive errors is a challenging problem in GWASs. Although our method does not directly address this issue, our method is able to reduce the number of possibly disease-associated epistatic interactions into a very small set and rank those epistatic interactions based on their relevances to the disease trait. Extra filters can be applied to remove false positives. We plan to incorporate the haplotype information and pathway information to help reduce false positive errors.

Acknowledgments

This work was supported by the Bioinformatics PhD Training Grant from the Hong Kong Jockey Club Charities Trust and the Hong Kong Special Administrative Region Government under the “Grants for University Research Expenditure” Award (GUR 03/04). The authors would like to thank the anonymous reviewers for their constructive comments and suggestions that have improved the quality of the manuscript.

REFERENCES


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.


Conflict of Interest: none declared.