Journal of Bioinformatics and Computational Biology
Vol. 4, No. 2 (2006) 241–257

Imperial College Press
www.icpress.co.uk

CHOOSING SNPs USING FEATURE SELECTION

TU MINH PHUONG*
Department of Information Technology
Posts & Telecommunications Institute of Technology
Hanoi, Vietnam
phuongtm@fpt.com.vn

ZHEN LIN† and RUSS B. ALTMAN‡
Department of Genetics, Stanford University
Stanford, CA, 94305 USA
†*zlin@helix.stanford.edu*
‡*russ.altman@stanford.edu*

Received 15 September 2005
Accepted 31 January 2006

A major challenge for genomewide disease association studies is the high cost of genotyping large number of single nucleotide polymorphisms (SNPs). The correlations between SNPs, however, make it possible to select a parsimonious set of informative SNPs, known as "tagging" SNPs, able to capture most variation in a population. Considerable research interest has recently focused on the development of methods for finding such SNPs. In this paper, we present an efficient method for finding tagging SNPs. The method does not involve computation-intensive search for SNP subsets but discards redundant SNPs using a feature selection algorithm. In contrast to most existing methods, the method presented here does not limit itself to using only correlations between SNPs in local groups. By using correlations that occur across different chromosomal regions, the method can reduce the number of globally redundant SNPs. Experimental results show that the number of tagging SNPs selected by our method is smaller than by using block-based methods.

Supplementary website: http://htsnp.stanford.edu/FSFS/.

1. Introduction

The abundance of single nucleotide polymorphisms (SNPs) in the human genome provides powerful tools for studying the association between sequence variation and the genetic component of common diseases. Although genome-wide SNP scans can give the most complete information for association studies, it is currently expensive

*This work was done when the first author was visiting Stanford university.
‡Corresponding author.

to genotype all available SNPs across the human genome. An alternative strategy in this situation is to genotype enough SNPs to provide the majority of information required for association studies, and ignore those that are redundant given typed SNPs. This strategy is enabled by the correlations between SNPs as manifested as *linkage disequilibrium* (LD). A subset of SNPs that are selected to represent the original information embedded in the full set of SNPs is referred to as the set of tagging SNPs (tSNPs). The problem of finding this set of tagging SNPs is called tagging SNP selection problem.

Several algorithms have been proposed for selecting tagging SNPs. A common approach is to define a measure of goodness for SNP sets and search for SNP subsets that: (i) are small in size, and (ii) attain high value of the defined measure.[2,25,26] Unfortunately, examining every SNP subset to find good ones is computationally infeasible for all but smallest data sets. To overcome this difficulty, investigators have exploited apparent features of haplotypes, which sometimes form haplotype blocks of limited diversity. Automatic algorithms first partition chromosomal regions into haplotype blocks,[20,27,28,15] then subsets of tagging SNPs are searched within each haplotype block. This approach is widely known as the block-based approach.

A main drawback of block-based methods is that the definition of blocks is not always straightforward and there is no consensus on how blocks must be formed. In addition, selecting tagging SNPs based only on the local correlations between markers of each block ignores inter-block correlations. Recent empirical studies reported LD distances with upper range extending to hundreds of Kb,[8] which are much longer than maximum block sizes reported in Refs. 11 and 29. Tagging SNP selection therefore can benefit from using information about these global correlations. Indeed, a recent study[1] shows that using long range LD reduces the number of tagging SNPs.

Another approach to selecting tagging SNPs uses data reduction techniques such as principal component analysis (PCA) to find subsets of SNPs capturing majority of the data variance.[18,17] Although not requiring exponential search time, PCA is still computationally complex, especially for large chromosomal data sets. The "sliding windows" method proposed by Meng *et al.*,[18] which applies PCA repeatedly to short chromosomal regions, can make PCA more efficient.

Approaches that look for tagging SNPs globally are known as block-free approaches.[23,2,12] Sebastiani *et al.*[23] represent non-tagging SNPs as boolean functions of tagging SNPs and use set-theoretic techniques to reduce search space. Bafna, Halldorsson and their colleagues[2,12] allow their algorithm to search for subsets of markers that can come from non-consecutive blocks. They reduce the search space by introducing the notion of neighborhood of markers, which in some sense is an extension of the block notation. Carlson *et al.*[5] group SNPs into bins such that in each bin there is at least one SNP (seed) in high LD with all the other SNPs of the bin, then iteratively remove all SNPs but the seed from current largest bin in greedy manner. The seed-SNPs then serve as tagging SNPs.

In this work, we take a block-free approach to make use of all the LD information. To avoid computational complexity, we do not look for subsets of SNPs but discard redundant markers using a feature selection method. While this strategy does not guarantee optimal solutions, it can give better performance on large data sets when exhaustive search can only be applied locally to short chromosomal regions.

2. Methods

Assume we are given N haploid sequences consisting of m bi-allellic SNPs. The N sequences can be represented as a matrix of size $m \times N$ where rows are sequences and columns are SNPs. Each element (i, j) of the matrix is the allele of the ith sequence at the jth SNP locus and can be 0, 1 or 2 where 1 and 2 are the two alleles and 0 indicates missing data.

We treat the problem of selecting tagging SNPs as a feature selection problem. Each haploid chromosomal sequence (row) is a learning instance belonging to a class. Each class consists of identical rows. SNPs (columns) are attributes or features, based on which sequences can be classified into classes. The problem is to select a subset of SNPs that can be used to classify the haploid sequences with the accuracy close to that of classification using all the SNPs.

There are a number of feature selection methods in the literature, which obviously are not equally good for our purposes. A feature selection method which is suitable for selecting tagging SNPs must have the following characteristics: (1) it should scale well for large number of SNPs; (2) it should not require explicit class labeling and should not assume the use of a specific classifier because classification is not the goal of tagging SNP selection; (3) it should allow the user to select different numbers of tagging SNPs for different amounts of tolerated information loss; and (4) it should have good performance among the methods satisfying the three first conditions.

Methods for selecting features fall into two categories: *filter methods* and *wrapper methods*. Filter algorithms are general preprocessing algorithms that do not assume the use of a specific classification method. Wrapper algorithms, in contrast, "wrap" the feature selection around a specific classifier and select a subset of features based on the classifier's accuracy using cross-validation. While there are strong arguments in favor of both approaches, wrapper algorithms are generally slower and do not satisfy condition (2). Therefore, we will consider only filter methods that do not require explicit class labeling.

Here we adopt the filtering feature selection method described in Ref. 19, which has all the characteristics mentioned above including good reported performance. The method uses feature correlation/similarity to remove redundant features and does not require knowledge about class labels. It has a parameter that can be used to control the degree of information loss (condition 3). It is fast because it does not explicitly search for subsets of features. We next describe the method, which is called *Feature Selection using Feature Similarity* (FSFS).[19]

A feature is a good feature not only is it good differentiating classes by itself or in combination with the other features in a feature subset, but also not redundant given the other features. FSFS involves grouping features in clusters so that features within each cluster are similar. A single feature from each cluster is then selected to present the other cluster members. The next two subsections describe FSFS in more details.

2.1. *Measures of feature similarity*

In order to use FSFS, we need to define a measure of similarity between a pair of features (SNPs in our case). There are a number of pairwise correlation/similarity measures between two random variables. These measures can be categorized as *linear* or *nonlinear* as they give the amount of linear or higher dependency between the two variables. Examples of linear measures are well-known *correlation coefficient* ρ, the LD measure r^2,[8] and the *least squared regression error* e. The authors of FSFS also introduced a linear measure of similarity between two numerical random variables called *maximal information compression index* λ^2. An example of nonlinear similarity measures is *symmetrical uncertainty* SU.[21]

It has been proved that if there is a linear dependency between some features, and if the data are linearly separable in the original representation, then the data remain linearly separable if all but one feature of the linearly dependent features are removed.[7] It is also easy to demonstrate that haplotype classes are linearly separable when there are only two alleles at a locus. Linear similarity measures are therefore more suitable when using FSFS to select tagging SNPs.

In our experiments, we used r^2 to measure the similarity/correlation between two SNPs:

$$r^2 = \frac{(p_{AB} \cdot p_{ab} - p_{Ab} \cdot p_{aB})^2}{p_A \cdot p_B \cdot p_a \cdot p_b} \tag{1}$$

where A and a are the two possible alleles at one locus, B and b are the two possible alleles at the other locus; p_{xy} denotes the frequency of observing x and y together in the same haplotype; p_x denotes the frequency of x. A r^2 value of 1 indicates the highest LD or highest similarity while the value of 0 indicates no LD.

The LD measure r^2 is directly related to recombination rate. r^2 is equal to 1 if and only if the two SNPs have not been separated by recombination and their allele frequencies are the same. In this case, only two out of four possible haplotypes are present in the sample. The value of r^2 decreases as the genetic distance between the pair of markers increases. More details on the biological meaning and appropriateness of r^2 for genetic mapping can be found in Refs. 9 and 22.

2.2. *Tagging SNP selection using FSFS*

FSFS selects features by first grouping them into homogeneous subsets and then choosing a representative feature from each subset. In what follows the terms "feature" and "SNP" are exchangeable.

Let the original SNP set of N SNPs be $S = \{F_i : i = 1, \ldots, N\}$. Let $D(F_i, F_j)$ denote the distance or dissimilarity between SNPs F_i and F_j (the notion of distance used here should not be confused with chromosomal distance between SNPs). The higher $D(F_i, F_j)$ the less similarity between the two features. $D(F_i, F_j)$ may be computed using one of similarity measures mentioned above, e.g. $D(F_i, F_j) = 1 - r^2(F_i, F_j)$. Let R denote the reduced tagging SNP subset to be selected. The FSFS algorithm is given in Fig. 1.

FSFS takes as input a set S of SNPs, a parameter k, where k is an integer less than the number of SNPs in S and returns a reduced set R of tagging SNPs. In the first step, the algorithm initializes R to S. It then discards SNPs from R through a number of iterations (steps 2–7). During an iteration, for each feature F_i of R, FSFS calculates the distance d_i^k between F_i and its kth nearest neighbor SNP (step 2). The neighborhood is defined in terms of dissimilarity between SNPs and should not be confused with the subset of SNPs located nearby in the chromosome. The algorithm then finds SNP F_0 for which d_0^k is minimum, retains this SNP (*seed SNP*) in R and discards its k nearest SNPs from R (step 3). By doing that, the algorithm always discards SNPs from the most compact cluster and F_0 is the SNP for which removing k nearest neighbors causes minimum information lost (Fig. 2).

Input: $S(F_1, F_2, ..., F_N)$ // original SNP set S
k $(k \le N-1)$ // a parameter k

Output: R // a tSNP subset R.

1. $R \leftarrow S$ // initialize R to S.
2. **for** each $F_i \in R$ **do**
$d^k{}_i = D(F_i, F^k{}_i)$ where $F^k{}_i$ is the k-th nearest neighbor of F_i in R.
end for
3. find F_0 such that $d^k{}_0 = \arg\min_{F_i \in R}(d^k{}_i)$
let $F^1{}_0, ..., F^k{}_0$ be the k nearest SNPs of F_0
$R \leftarrow R / \{F^1{}_0, ..., F^k{}_0\}$
if first iteration **then** set $\theta = d^k{}_0$
4. **if** $k > |R| - 1$ **then** $k = |R| - 1$
5. **if** $k = 1$ **goto** 8.
6. **while** $d^k{}_0 > \theta$ **do**
$k = k - 1$
if $k = 1$ **goto** 8
recompute $d^k{}_0$.
end while
7. **goto** 2
8. **return** R

Fig. 1. The FSFS algorithm.

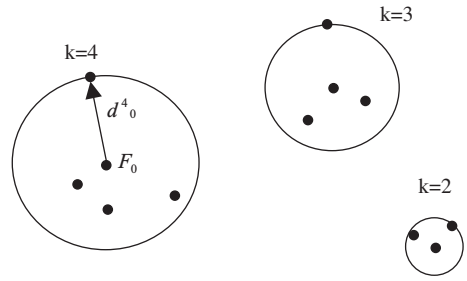

Fig. 2. Feature clusters for different k.

For the first iteration, a constant error threshold θ is set $\theta = d_0^k$. Step 4 compares the cardinality of R after step 3 with k and adjusts k if necessary. In step 6, FSFS gradually decreases k and recomputes d_0^k until d_0^k is not greater than threshold θ. This ensures that no SNP which is more θ-dissimilar to a seed will be discarded. The algorithm ends when no SNP in R can be discarded with error less than or equal to θ.

FSFS has one parameter k — the number of the nearest neighbors of each feature. As noted by Mitra *et al.*,[19] the choice of k controls the representation of data at different degrees of details and provides a direct way to control the maximum information loss when choosing features. In general, different values of k result in different reduction degrees of the feature set. The bigger k, the more features are discarded and vice versa.

In the context of choosing tSNPs, there are two possible ways to select k. (1) Select k so that the distance between a seed SNP to its k-nearest neighbor is less than some threshold, which implies that for any non-tagging SNP there exist a tSNP such that the r^2 between them is greater than some threshold. For example, in Ref. 4, a r^2 threshold of 0.8 was used for choosing tSNPs. (2) Select k to achieve desired prediction accuracy via cross-validation. The accuracy evaluation will be given in more detail in Sec. 3.

The computational complexity of FSFS with respect to the number of features N is $O(N^2)$. If the data set contains m rows (m sequences in the current problem), the complexity of computing the similarity of a pair of features depends on the chosen similarity measure. In particular, the complexity of computing r^2 is $O(m)$. Thus, the overall complexity of the method is $O(N^2mk)$ taking into account the iteration number.

For large data sets with N achieving tens of thousands, the complexity $O(N^2)$ is still high. In our implementation for such large data sets we added a preprocessing step. In this step all SNPs that are in perfect LD with each other ($r^2 = 1$) are considered identical and only one of them is retained. The algorithm is then run on the reduced SNP set where there are no SNP pairs with $r^2 = 1$. For shorter-sequence data sets ($N < 5000$) the preprocessing step is not necessary.

2.3. *Evaluation methods*

There are several ways to assess the accuracy of a tagging SNP selection method. Stram *et al.*[23] introduced a quality measure R^2, which is the measure of association between the true numbers of haplotype copies defined over the full set of SNPs and the predicted number of haplotype copies where the prediction is based on the subset of tagging SNPs. This measure assumes diploid data and explicit inference of haplotypes from genotypes and thus is not suitable for our purpose.

Another assessment method due to Clayton (http://www-gene.cimr.cam.ac.uk/clayton/software/stata/htSNP/htsnp.pdf) is based on a measure of the diversity of haplotypes. The diversity is defined as the total number of differences in all pairwise comparison between haplotypes. The difference between a pair of haplotypes is the sum of differences over all the SNPs. The Clayton's diversity measure can be used to define how well a set of tagging SNPs differentiate different haplotypes. This measure is suitable only for haplotype blocks with limited haplotype diversity and it is not clear how to use it for large data set consisting of multiple haplotype blocks.

Some recent works[17,12] evaluate tagging SNPs selection algorithms based on how well the tagging SNPs can be used to predict non-tagging SNPs. The prediction accuracy is determined using cross-validation such as leave-one-out or hold out. In leave-one-out cross-validation, for each sequence in a data set, the algorithm is run on the rest of the data set to select a minimum set of tagging SNPs. The alleles of the left out haplotype are then predicted from "typed" SNPs (tagging SNPs). The prediction precision is calculated as

$$\frac{\text{number of correctly predicted alleles}}{\text{all predicted alleles}}.$$

The precision is then averaged over all sequences to give the measure of accuracy for a tagging SNP algorithm on the data set.

Depending on how tagging SNPs are selected, different prediction methods have been used during cross-validation process. Halldorsson *et al.*,[12] who select tagging SNPs based on their ability to differentiate haplotypes, use a modification of the kNN machine learning method to predict the left-out haplotype. First, the training haplotypes that are most similar to the left-out haplotype are determined. The similarity is defined as the Hamming distance over tagging SNP. Then, the alleles are predicted by a majority vote of the respective alleles from the most similar training haplotypes.

In contrast, Lin and Altman[17] predict the alleles of a non-tagging SNP n from the tagging SNPs that have the highest correlation coefficient with n. If a single highly correlated tagging SNP t is found, the alleles are assigned so that their frequencies agree with the allele frequencies of t. When multiple tagging SNPs have the same (high) correlation coefficient with n, the common allele of n has advantage. It is easy to see that in this case the prediction method agrees well with the selection method, which uses principal component analysis on the matrix of correlation coefficients between SNPs.

Since the method of selecting tagging SNPs described here is based on the pairwise similarity of SNPs, we take the prediction method similar to that of Ref. 17. In particular, for each non-tagging SNP n we look for the most similar tagging SNP t, which is the seed of the corresponding cluster (see algorithm above). The allele A_n of n is then chosen so that it agrees well with the corresponding allele A_t of t. In other words, A_n is chosen so that $P(A_n|A_t)$ is maximum, where $P(A_n|A_t)$ is the conditional probability that A_n appears in a haplotype at locus n when A_t appears in the haplotype at locus t.

2.4. *Dealing with diploid data*

Up to this point we assumed the input sequences are haploid. In practice, experimental determination of haploid data is much more difficult than that of diploid data. The use of LD measure r^2 can overcome this problem by computationally inferring haplotype frequencies, e.g. using the EM algorithm of Ref. 10, over each pair of SNPs, for which r^2 needs to be computed. Specifically, to compute r^2 by formula (1) one needs to know only the probabilities of the four possible haplotypes but not the haplotypes themselves. This approach was used to compute r^2 from diploid data in Ref. 8.

3. Experiments and Results

3.1. *Data sets*

To assess the method on haploid data, we used two data sets of different sizes. First, to see the performance of the method in large scale data sets, we use the data set of human chromosome 21 described in Ref. 20. The data set consists of 24 047 SNPs typed on 20 haploid copies of chromosome 21. Despite the small number of sampled chromosomes and the high rate of missing data, the data set was used as a test set in a number of studies.[27,28,12] In our experiments we ignored alleles with missing data. The cross-validation procedure was done on full data set as well as on the first 1000 SNPs of the set.

The second data set is the IBD 5q31 data set from an inflammatory bowel disease study of father-mother-child trios.[6] Here we used the haploid version of the data set described in Ref. 17 in which the haplotype phase was solved by applying PHASE 2.0.2.[24] The haploid data set after phasing contains 103 biallelic non-singletons from 774 phased chromosomes. This data set contains no missing data.

These two data sets present different experimental conditions to evaluate tagging SNP selection methods. While the former contains genome-wide sequences of a small number of samples, the latter contains relatively shorter sequences of a large number of samples.

To assess how the method can deal with diploid data, we used the data set of human chromosome 22 described in Ref. 14. The data set consists of 20 360 SNPs genotyped in 71 individuals from three populations: 23 African Americans,

24 European Americans, and 24 Han Chinese. The data set has low rate of missing data (about 2%). We used the unphased version of the data set and ran experiments on the three subsets of three populations separately. Here we report the results for the first two populations only due to the similarity of the results.

For all the data sets, we did not consider SNPs with minor allele frequencies (MAF) of less than 10%.

3.2. *Comparison*

We compared the method using FSFS with the block-based method of Ref. 28. This method was chosen because it can deal with large data sets. Another method that can be used for large data set is the block-free method by Halldorsson *et al.*[12] Unfortunately, we could not obtain the code that implements this method for our experiments.

The method presented in Ref. 28 uses dynamic programming algorithms to partition chromosomes into blocks of limited haplotype diversity and searches for tagging SNP within each block. In our experiments we used the program Haploblock version 3.0 which is the implementation of the algorithms. To select tagging SNP subsets of different sizes we ran Haploblock in "*Block partition with a fixed number of tag SNPs"* mode with the chromosome coverage for each tagged SNP set to 1. We also ran FSFS with different values of k to select tagging SNP subsets of different sizes.

To evaluate the performance of FSFS on diploid data, we compared tagging SNP sets chosen by the method with those chosen randomly.

3.3. *Results*

To limit the amount of computation, we followed Ref. 12 and performed leave-one-out cross-validation of FSFS and the block-based dynamic programming method on the first 1000 SNPs of the chromosome 21 data set. As noted by those authors, this subset is highly representative for the overall data set. Figure 3 shows the cross-validation accuracy plotted against the number of tagging SNPs selected by each method. As mentioned above, different numbers of tagging SNPs selected by FSFS resulted from different values of k.

The fraction of correctly predicted non-tagging SNPs is higher for FSFS than for the block-based method for most selected SNP numbers. The accuracy of the two methods increases rapidly until reaching about 85%, after that a more gradual improvement is observed, which may be explained by the presence of rare haplotypes.

Due to relatively large number of sequences of the IBD1 data set, we performed 10-fold cross-validation on it. The results are plotted in Fig. 4. The FSFS-based method results in smaller tagging SNP sets to achieve a slightly better accuracy than that of the block-based method. A possible explanation for the better performance

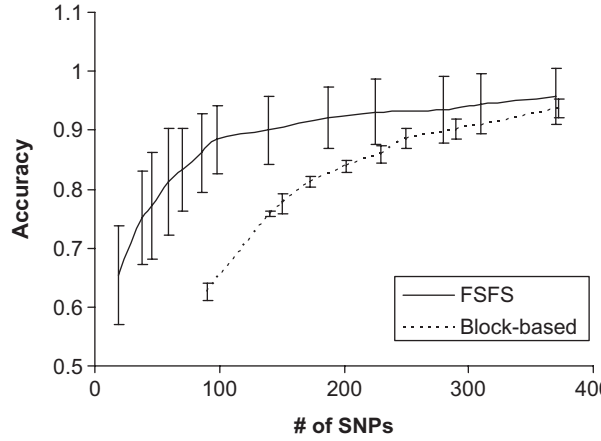

Fig. 3. Results of leave-one-out experiments on the first 1000 SNPs of the chromosome 21 data set. The solid and doted curves present the results when using the FSFS, and the block-based method of Zhang *et al.*[28] respectively. The x-axis shows the number of selected tagging SNPs; the y-axis shows the fraction of correctly predicted non-tagging SNPs. The results are plotted with 1-std error bars.

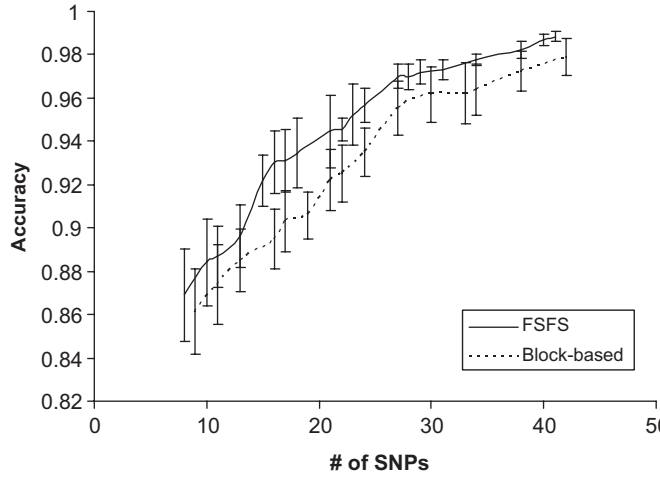

Fig. 4. Results of tenfold cross-validation experiments on IBD1 data set. The solid and doted curves present the results when using the FSFS, and the block-based method of Zhang *et al.*[28] respectively. The x-axis shows the number of selected tagging SNPs; the y-axis shows the fraction of correctly predicted non-tagging SNPs. The results are plotted with 1-std error bars.

of FSFS on the IBD1 data set is that despite the relatively small number of SNPs considered, the data set consists of several small haplotype blocks. The block-based method does not remove SNPs that are correlated with SNPs from other blocks and therefore are redundant. A closer look at the output of the block-based algorithm

verifies this hypothesis. The algorithm partitions the chromosome region into from 5 to 11 blocks depending on the input parameters.

We performed 10-fold cross-validation on the diploid data sets of chromosome 22. To predict non-tagging SNPs in test sets we used a procedure similar to the one described in Sec. 2.3. The only difference is that instead of predicting individual alleles, the procedure predicts the two possible alleles for each non-tagging SNPs. A prediction is counted as correct if both predicted alleles agree with the reference.

To compare tagging SNP sets chosen by FSFS with tagging SNP sets chosen randomly, for each training subset we used FSFS to select a tagging set T_{FSFS}, then randomly selected ten tagging sets T_R of the same size as T_{FSFS}. The tagging sets were then used to predict non-tagging SNPs using the same procedure described above. The accuracy of random selection is averaged over ten generated sets T_R for each cross-validation fold.

The results for the African and European populations are plotted in Figs. 5 and 6 respectively. Tagging sets chosen by FSFS lead to higher prediction accuracy than that of random tagging sets of same sizes. The results also show that to achieve the same prediction accuracy fewer tagging SNPs are required on the data set of European population than those of African population.

It is of question if different populations share the same tSNPs when using the FSFS based method. To partially answer this, we ran the preprocessing step of the algorithm (Sec. 2.2) on the pooled data set from all the three populations to see how many SNPs are in perfect LD with at least one other SNP. In total 5276 such SNPs

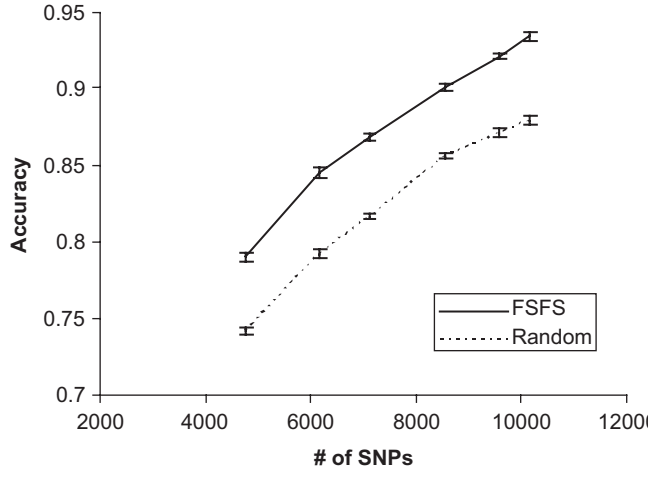

Fig. 5. Results of tenfold cross-validation experiments on chromosome 22 diploid data set, African population. The solid and doted curves present the results when using the FSFS, and randomly selected tagging SNPs respectively. The x-axis shows the number of selected tagging SNPs; the y-axis shows the fraction of correctly predicted non-tagging SNPs. The results are plotted with 1-std error bars.

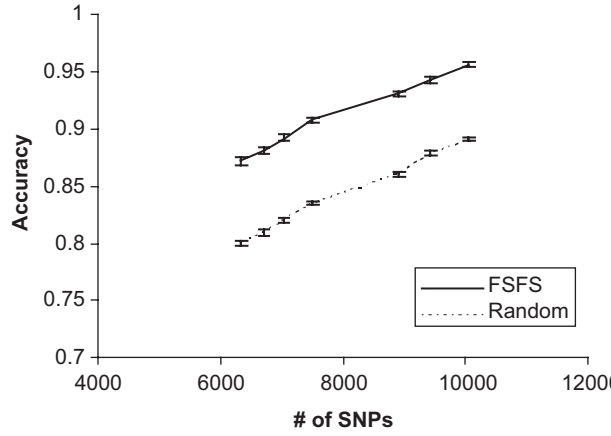

Fig. 6. Results of tenfold cross-validation experiments on chromosome 22 diploid data set, European population. The solid and doted curves present the results when using the FSFS, and randomly selected tagging SNPs respectively. The x-axis shows the number of selected tagging SNPs; the y-axis shows the fraction of correctly predicted non-tagging SNPs. The results are plotted with 1-std error bars.

are found for the pooled data set. For African, European, and Chinese populations, the numbers of SNPs defined in this way are 7034, 8364 and 8249 respectively. These results show that, for the most restrictive threshold of LD measure r^2, the three populations share a considerable fraction of tagging SNPs.

3.4. *Cluster organization*

The method presented in this works uses the correlations between SNPs that are located across the chromosome region considered and thus the performance of the method depends largely on how correlated SNPs are distributed. To understand the behavior of the method, we analyze the clusters created when running the algorithm. The algorithm was run on the full chromosome 21 data set and k was chosen to achieve 80% cross-validation accuracy. These settings resulted in 3009 tagging SNPs in average. The size and content of clusters created during selection process were saved and visualized graphically. In all, 1993 clusters were chosen by the algorithm when discarding SNPs. The maximum size of the neighborhood/cluster created (in the first iteration) is 481. In Fig. 7, the locations of SNPs from the six largest clusters are presented.

In Fig. 7, each triangle corresponds to one cluster. The interpretation of triangles is as follows. The whole chromosome consisting of 24 047 SNPs is divided into 81 regions; each contains 300 consecutive SNPs (the last region has only 47 SNPs). Each row/column corresponds to one such region. Each cell contains the number of the cluster's members from the respective row multiplied by the number of the cluster's members from the respective column. For example, if a cluster has 5 SNPs

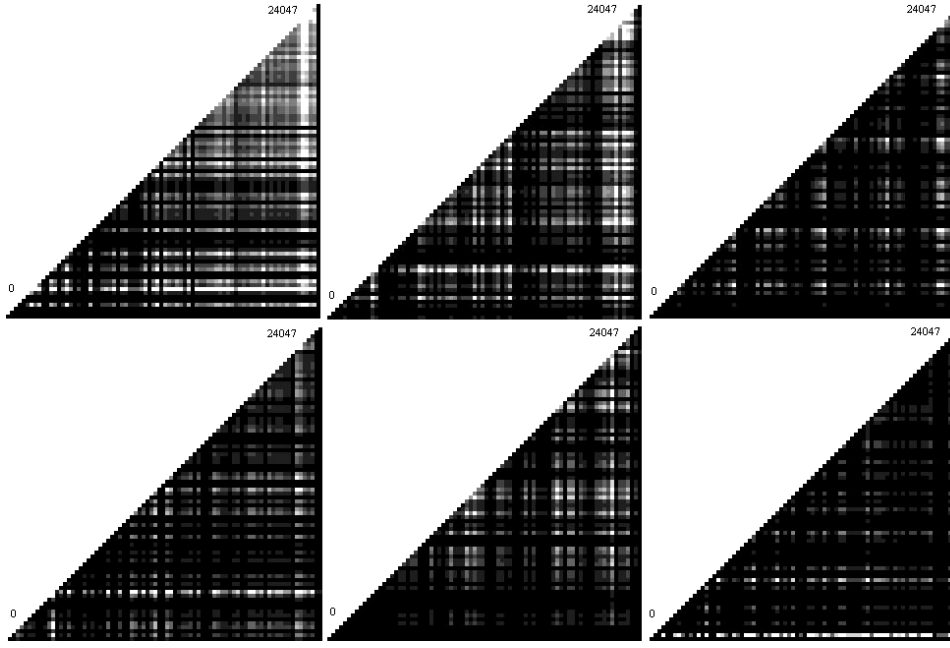

Fig. 7. The largest six clusters created by FSFS from the chromosome 21 data set. Each triangle corresponds to one cluster. Rows and columns are regions in the chromosome. Each cell presents the product of the numbers of clusters members from the respective row and column. Black denotes low numbers and white denotes high numbers. Thus, the first cluster (upper left) has members from all along the chromosomal segment.

coming from region x and 10 SNPs coming from region y, then cell (x, y) of the respective triangle contains value $5 * 10 = 50$. Gray-scale levels are used to present digital values. Black denotes 0, and white denotes maximum number. Other gray levels denote values between 0 and maximum.

The figure shows that large clusters consist of SNPs from different regions of the chromosome. Since the algorithm groups SNPs into clusters based on within cluster LD distances, these figures shows that SNPs, which are in high LD can be located distantly but not only within haplotype blocks. This observation is consistent with the findings reported in Ref. 7, which show high LD between distantly located SNPs.

In comparing FSFS with the block-based method, there are two questions of interest: (1) what is the fraction of redundant SNPs that can be removed by the block-based method due to within-block correlations but cannot be removed by FSFS; (2) what is the fraction of redundant SNPs that can be removed by FSFS due to remote correlations but cannot be removed by the block-based method. To answer the first question we ran the block-based method on the IBD1 data set with parameters chosen to achieve about 90% cross-validation accuracy. We then ran FSFS on each block generated by the block-based method to achieve the same

90% accuracy. Because FSFS was run on individual blocks it was limited to use only within-block correlations. The numbers of tSNPs from the blocks were then sum up and compared with those selected by the block-based method. The IBD1 data set was chosen for this experiment to reduce computational complexity. These settings resulted in 86 SNPs discarded by the block-based method and 78 SNPs discarded by FSFS.

The answer to the second question is not so straightforward because a redundant SNP can be removed by FSFS due to either local or remote correlations. Thus we tried to answer this question indirectly. Both methods were run on the IBD1 data set to achieve about 90% cross-validation accuracy. We then mapped the SNPs removed by FSFS to the blocks formed by the block-based method. If a FSFS-removed SNP and its tagging SNP are within the same block, the correlation between the two SNPs is considered local, otherwise it is considered global. The experiment shows that among redundant SNPs removed by FSFS, 58% have local correlations with their tSNPs and 42% have global correlations with their tSNPs.

4. Conclusion

We investigated an efficient block-free SNP-tagging method and compared it to an existing block-based method. The new block-free method showed good performances in finding smaller tagging SNP set to achieve the same cross-validation prediction accuracy in two experimental datasets.

The method has two major characteristics. First, it does not involve subset search. Instead, SNPs are removed individually to form tagging sets based on pairwise similarity. Second, global similarity/correlations between SNPs across chromosomes are used to find redundant markers. While the first characteristic does not allow finding tagging SNPs that in combination with other tagging SNPs can predict non-tagging ones,[26] it makes computation less complex. This enables the realization of the second characteristic. The overall effect is that while being not optimal, the method presented here can have performance comparable or better than methods based on block-partitioning when applied to chromosome regions with high haplotype diversity.

The method presented in this paper is similar to the one suggested by Carlson *et al.*[5] in that: (1) both methods use pairwise LD as measure of similarity; (2) both methods group SNPs in clusters and find a representative SNP for the best cluster. The difference is in the way FSFS forms clusters and selects the best cluster. While the method by Carlson *et al.*[5] forms clusters based on a LD threshold, FSFS groups SNPs in clusters of the same size and selects the most compact one. We also explicitly evaluated the goodness of tagging SNPs in predicting non-tagging SNPs using cross-validation. Cross-validation is also a natural way to choose the main parameter of the algorithm — the start size of clusters.

The main reason our method is able to find smaller sets of tagging SNPs than block-based methods is that it takes advantage of using both local and long range

LD across chromosomes. This is demonstrated from the analysis of SNP clusters formed during FSFS's iterations. The presence of such long-range LD and the benefits of using them to select tagging SNPs have also been reported in a recent paper.[1] These results together give more support to block-free approaches to finding tagging SNPs. With more block-free tagging methods become accessible, we can further analyze them and compare them to FSFS.

Acknowledgments

Tu Minh Phuong was supported by a Fulbright fellowship. RBA and ZL were supported by NIH GM61374. ZL was supported by NIH/NLM Biomedical Informatics Training Grant LM007033. We thank the two anonymous reviewers for their comments and suggestions.

References

1. Ahmadi K, Weale M, Xue Z, *et al.*, A single-nucleotide polymorphism tagging set for human drug metabolism and transport, *Nature Genet* **37**:84–89, 2004.
2. Avi-Itzhak H, Su X, de la Vega F, Selection of minimum subsets of single nucleotide polymorphisms to capture haplotype block diversity, *Proc Pacific Symp Biocomput*, Vol. 8, pp. 466–477, 2003.
3. Bafna V, Halldorsson B, Schwartz R, *et al.*, Haplotypes and informative SNP selection: Don't block out information, In *Proceedings of Recomb 2003*, 19–27, 2003.
4. Carlson C, Eberle M, Rieder M, *et al.*, Additional SNPs and linkage-disequilibrium analyses are necessary for whole-genome association studies in humans, *Nature Genet* **33**:518–521, 2003.
5. Carlson C, Eberle M, Rieder M, *et al.*, Selecting a maximally informative set of single-nucleotide polymorphisms for association analyses using linkage disequilibrium, *Am J Hum Genet* **74**:106–120, 2004.
6. Daly M, Rioux J, Schaffner S, *et al.*, High-resolution haplotype structure in the human genome, *Nature Genet* **29**:229–232, 2001.
7. Das S, Feature selection with a linear dependence measure, *IEEE Trans Comput* **20**:1106–1109, 1971.
8. Dawson E, Abecasis G, Bumpstead S, *et al.*, A first-generation linkage disequilibrium map of human chromosome 22, *Nature* **418**:544–548, 2002.
9. Devlin B, Risch N, A comparison of linkage disequilibrium measures for fine-scale mapping, *Genomics* **29**:311–322, 1995.
10. Excoffier L, Slatkin M, *et al.*, Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population, *Mol Biol Evol* **12**(5):921–927, 1995.
11. Gabriel S, Schaffner S, Nguyen H, *et al.*, The structure of haplotype blocks in the human genome, *Science* **296**:2225–2229, 2002.
12. Halldórsson B, Bafna V, Lippert R, *et al.*, Optimal haplotype block-free selection of tagging snps for genome-wide association studies, *Genome Res* **14**:1633–1640, 2004
13. Johnson G, Esposito L, Barratt B, *et al.*, Haplotype tagging for the identification of common disease genes, *Nature Genet* **29**:233–237, 2001.
14. Hinds D, Stuve L, Nilsen G, *et al.*, Whole-genome pattern of DNA variation in three human populations, *Science* **307**:1072–1079, 2005.
15. Ke X, Cardon L, Efficient selective screening of haplotype tag SNPs, *Bioinformatics* **19**:287–288, 2003.

16. Lewontin R, The interaction of selection and linkage. I. General considerations; heterotic models, *Genetics* **49**:49–67, 1964.
17. Lin Z, Altman R, Finding haplotype tagging SNP by use of principle component analysis, *Am J Hum Genet* **75**:850–861, 2004.
18. Meng Z, Zaykin D, Xu C, *et al.*, Selection of genetic markers for association analyses, using linkage disequilibrium and haplotypes, *Am J Hum Genet* **73**:115–130, 2003.
19. Mitra P, Murthy C, Pal S, Unsupervised feature selection using feature similarity, *IEEE Trans Pattern Analysis Machine Intelligence*, **24**:301–312, 2002.
20. Patil N, Berno A, Hinds D, *et al.*, Blocks of limited haplotype diversity revealed by high resolution scanning of human Chromosome 21, *Science* **294**:1719–1723, 2001.
21. Press W, Teukolsky S, Vetterling W, Flanery B, *Numerical Recipes in C*, Cambridge University Press, Cambridge, 1988.
22. Pritchard J, Przeworski M, Linkage disequilibrium in humans: models and data, *Am J Hum Genet* **69**:1–14, 2001.
23. Sebastiani P, Lazarus R, Weiss S, *et al.*, Minimal haplotype tagging, *Proc Natl Acad Sci* **100**:9900–9905, 2003.
24. Stephens M, Smith N, Donnelly P, A new statistical method for haplotype reconstruction from population genotype data, *Am J Hum Genet* **69**:906–1014, 2001.
25. Stram D, Leigh Pearce C, Bretsky P, *et al.*, Modeling and E-M estimation of haplotype-specific relative risks from genotype data for a case-control study of unrelated individuals, *Hum Heredity* **55**:179–190, 2003.
26. Weale M, *et al.*, Selection and evaluation of tagging SNPs in the neuronal-sodium-channel gene SCN1A: implications for linkage-disequilibrium gene mapping, *Am J Hum Genet* **73**:551–565, 2003.
27. Zhang K, Deng M, Chen T, *et al.*, A dynamic programming algorithm for haplotype block partitioning, *Proc Natl Acad Sci* **99**:7335–7339, 2002.
28. Zhang K, Sun F, Waterman S, Chen T, Haplotype block partition with limited resources and applications to human chromosome 21 haplotype data, *Am J Hum Genet* **73**:63–73, 2003.
29. Zhang K, Sun F, Waterman M, Chen T, Dynamic programming algorithms for haplotype block partitioning: Applications to human chromosome 21 haplotype data, *Proc Seventh Annu Int Conf Comput Mol Biol*, ACM Press, 2003.
30. Zhang K, Qin Z, Liu J, *et al.*, Haplotype block partitioning and tag SNP selection using genotype data and their applications to association studies, *Genome Res* **14**:908–916, 2004.

Tu Minh Phuong received a B.Eng from Tashkent University of Technology and a Ph.D in Control in technical systems from Uzbekistant National Academy of Science. Currently, he is a lecturer in the Department of Computer Science, Posts & Telecommunications Institute of Technology, Vietnam. His research interests include fuzzy logic, multiagent systems, machine learning, and bioinformatics.

Zhen Lin received a MS in Nursing from the University of California at San Francisco, and a PhD in Biomedical Informatics from the Stanford University. Her research interests include genetic privacy protection and functional genomics.

Russ B. Altman is professor of genetics, bioengineering, & medicine (and of computer science by courtesy) at Stanford University. His primary research interests are in the application of computing technology to basic molecular biological problems of relevance to medicine. He is currently developing novel resources for biological data, particularly for pharmacogenomics (e.g. http://www.pharmgkb.org/). Other work focuses on the analysis of functional microenvironments within macromolecules and the application of algorithms for determining the structure, dynamics and function of biological macromolecules (e.g. http://simbios.stanford.edu/).