IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON SPECIAL SECTION ON BIG DATA LEARNING AND DISCOVERY

Received October 28, 2018, accepted December 10, 2018, date of publication December 14, 2018,
date of current version January 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886926

Big Data Driven Oriented Graph Theory Aided tagSNPs Selection for Genetic Precision Therapy

TIANSHUO CONG[1], JINGJING WANG[2], (Student Member, IEEE), SANGHAI GUAN[2], YIFEI MU[3], TONG BAI[4], (Student Member, IEEE), AND YONG REN[2], (Senior Member, IEEE)

[1]Institute for Advanced Study, Tsinghua University, Beijing 100084, China
[2]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
[3]School of International Economics and Trade, Dongbei University of Finance and Economics, Dalian 116025, China
[4]School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K.

Corresponding author: Yifei Mu (yifeim@sina.com)

This work was supported in part by the Ministry of Education of China Youth Fund Program under Grant 17YJC790110, in part by the Department of Education of Liaoning Province's Program under Grant LN2017QN017, and in part by the Shenzhen Science and Technology Program under Grant JCYJ20150831192224146.

ABSTRACT Recently, the world-wide human genome-related projects have been vigorously launched and implemented. Gene-sequencing techniques play a critical role in disease diagnosis, prediction, and population stratification relying on efficiently mining genetic features in the gene pool. Exploring the association between the sites of the genetic mutation and the disease-based population classification becomes a hot topic, which beneficially supports disease diagnosis and treatment on the molecular level. However, there are numerous variable sites even on a single chromosome in the human gene pool, and hence, the traditional classifiers are not able to dig out all single nucleotide polymorphism (SNP) sites without clearly excavating the characteristic SNP sites, termed tagSNPs, in SNP clusters. By applying big data mining techniques, in this paper, we, first of all, propose a principal component analysis-based algorithm for reducing the gene data dimension in order to cluster SNP sites in the low-dimensional space. Moreover, an oriented graph theory-based tagSNPs selection algorithm is designed. Finally, relying on the real-world 1000 Genomes Project dataset, we can achieve fewer tagSNPs than the traditional methods by invoking the complete process of our designed SNP classifier.

INDEX TERMS Genetic feature mining, big data, data dimension reduction, SNP site clustering.

I. INTRODUCTION

The international HapMap project research plan was launched for recording the similarities and differences of human genes in 2002. Since then, HapMap has become a popular big dataset for both the genetics and the data science. The 1000 Genomes Project commenced in 2008, which aimed for building the largest public genotype database catalogue for human gene variation and for finding the most genetic variants with the occurrence rate of at least 1% among the population [1], which can provide molecular-level help for fundamentally treating genetic diseases. The single nucleotide polymorphism (SNP) primarily refers to the phenomenon of deoxyribonucleic acid (DNA) sequence polymorphism resulted from a single nucleotide variation at the genomic level, which is the most common kind of human heritable variation. This polymorphism can cause a range of diseases for different people [2]. Moreover, in the study of pathobiology, the SNP site is regard as a high-resolution marker for comparing morbid traits and normal traits. However, there are millions of SNP sites and some of them have similar characteristics. Hence, excavating tagSNPs, i.e. representative SNP sites, can reduce the computational complexity and improve the genetic feature extraction efficiency [3]. Machine learning algorithms have a huge potential in numerous aspects for automated understanding and analysis of big data [4], especially in the field of healthcare [5], [6].

In the literature, a range of researches have been investigated both for analyzing SNP sites and for excavating tagSNPs. To elaborate a little further, Bertin *et al.* [7] carried out experiments to extend genomic resources of pearl millet into a SNP-based marker system. Afterwards, 1000 Genomes Project disclosed the human SNP data collected for a few years to the public [8]. There have been some clustering algorithms invoked for SNP site clustering. More explicitly,

Frommlet [9] proposed a SNP data clustering algorithm relying on the graph theoretical concept of the dominant set having the internal homogeneity feature and the external heterogeneity feature. This study showed that clustering SNP data can both provide good statistical power and improve tagSNP selection algorithm performance. Furthermore, Liao *et al.* [10] proposed a new multi-locus linkage disequilibrium (LD) measurement based on information theory to calculate the relationship between multiple-markers, which was utilized to cluster SNPs for overcoming the traditional LD shortcomings. In [11], Müller *et al.* considered the clustering problem relying on adequate similarity measurement. The similarity between SNP sites can be calculated by traditional method such as Euclidean distance, cosine similarity and so on. In recent years, some new measurement are proposed [12]. In [13], Kumar *et al.* discuss the availability of Gaussian similarity measure for intrusion detection.

As for the tagSNP selection, in [14], İlhan and Tezel proposed a genetic algorithm to select tagSNPs and a support vector machine (SVM) algorithm for further SNP prediction. Yeh and Jheng [15] proposed an information entropy aided iterative algorithm for the tagSNP selection. With the spirit of dynamic programming, it iteratively divided the SNP data into blocks relying on the information entropy until the number of tagSNPs met the requirement. Furthermore, Wang *et al.* [16] proposed a site-clustering graph based tagSNP selection algorithm, which combined the graph theory for finding the tagSNPs. For the first time, it displayed SNP sites in the form of directed graphs by calculating the subgraph's density, and selected the sites with the highest density as the candidate tagSNPs. The algorithm converged until the number of tagSNPs was unchanged. Lee and Shatkay [17] constructed a Bayesian network for the tagSNP selection, while Chang *et al.* [18] proposed a hybrid algorithm, which combined the branch-and-bound algorithm and the greedy algorithm, for obtaining a better selection performance.

In this paper, we try to propose a whole process for selecting tagSNPs by using 1000 Genomes Project. This dataset contains more populations, using the above-mentioned algorithm may cause high complexity. Meanwhile, because the dataset we used is new and some work are incomplete, our work is meaningful for the future research. Our tagSNP selection algorithm can select less tagSNPs than traditional method. This can help bioinformatician save research costs. The selection algorithm also give visual results that can present the correlation between SNP sites. Then we use support vector machine algorithm to predict unknown SNP sites by using training SNP sites. The high predicting accuracy rate verifies the effectiveness of our algorithm. And the prediction algorithm can help bioinformatician to diagnose some pathogenic gene sites. Inspired by the above open issues, in the paper, we conceive a novel oriented graph theory assisted SNP classifier for selecting tagSNPs, with the following contributions.

- Considering a large amount of high dimensional gene samples, we conceive a PCA aided data dimension reduction algorithm for accelerating the data pre-processing. Moreover, several SNP sites clustering algorithms are discussed to categorize the similarity of SNP sites for simplifying further tagSNPs selection. We choose the most efficient similarity measure for gene data.
- Relying on aforementioned cluster results and inspired by the site-clustering graph based tagSNP selection algorithm in [16], we propose an oriented graph theory aided algorithm for efficiently extracting tagSNPs with the aid of the node priority. And we make the tagSNP selection rule to fit for 1000 Genomes Project dataset. Our algorithm is also appropriate for concurrent processing which can reduce big data running time.
- Relying on real-world 1000 Genomes Project dataset, which is the largest public catalogue of human variation and genotype data for studying SNP related issues, we verify the effectiveness and efficiency of our proposed oriented graph theory aided tagSNPs selection algorithm by predicting the unknown SNP sites. The SVM algorithm can get high predicting accuracy rate.

The rest of this paper is organized as follows. Section II provides a brief introduction of the dataset considered. In Section III, we introduce the data pre-process, i.e. data dimension reduction and SNP clustering. In Section IV, a graph theory aided tagSNP selection algorithm is proposed. We show simulation results and corresponding discussions in Section V, followed by our conclusions in Section VI.

II. BRIEF INTRODUCTION OF DATASET

In this paper, we invoke the dataset from 1000 Genomes Project, which can be accessed through a VCF file from the official website.[1] The whole dataset contains 84.4 million variants on 23 chromosomes, that are collected from 2504 samples living in 26 populations. Fig. 1 illustrates a representative capture of the dataset. To elaborate, there are 8 fixed fields per record and we utilize 7 of them, explained as follows.

- CHROM: an identifier from the reference genome or an angle-bracketed ID String pointing to a contig in the assembly file;
- POS: the reference position, with the 1st base having position 1;
- ID: semi-colon separated list of unique identifiers where available;
- REF: each base must be one of A,C,G,T,N;
- ALT: comma separated list of alternate non-reference alleles;
- QUAL: phred-scaled quality score for the assertion made in ALT;

[1] http://www.internationalgenome.org/data

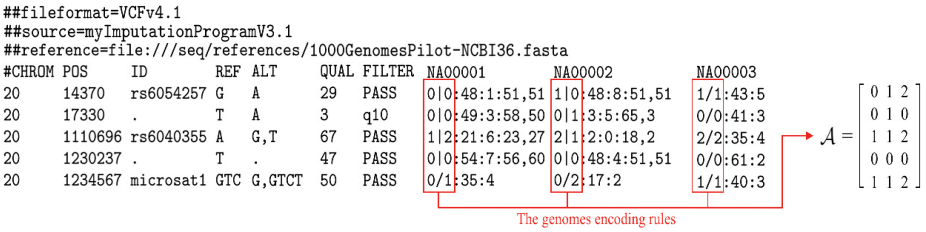


FIGURE 1. Illustration of a representative capture of the 1000 Genomes Project dateset, where the abbreviations in the head line are defined as follows: chromosome (CHROM), position (POS), identifier (ID), reference base (REF), alternate base (ALT), quality (QUAL), and filter status (FILTER). Moreover, the symbol '/' and '|' refer to unphased genotype and phased genotype, respectively.

TABLE 1. The genomes encoding rules.

Genotype (GT)	Encoded result
0/0	0
0/1,1/0,1/2,2/1,0/2,2/0	1
1/1,2/2	2

- FILTER: PASS if this position has passed all filters while "q10" might indicate that at this site the quality is below 10;
- NA0000x: individual sample.

As for the genotype (GT), it is composed of two numbers, representing human's diploid organism. Specifically, the allele value of 0 refers to the reference allele (in the REF field), while 1 denotes the first allele listed in ALT and 2 represents the second allele list in ALT and so on. Particularly, the GT with two same numbers is termed as homozygous mutation, whilst the one with two different numbers is named by heterozygous mutation. Let '|' represent the phased genotype, while '/' be the unphased genotype. The genomes encoding rules are summarized in Table. 1, indicating that the encoding results of homozygous mutation without ALT, heterozygous mutation and homozygous mutation having two ALT types are 0, 1 and 2, respectively. For the sake of simplification, We use '/' to represent both '/' and '|' in Table. 1.

III. DATA DIMENSION REDUCTION AND SNP CLUSTERING

A. DATA DIMENSION REDUCTION

First of all, in order to reduce the computing memory space and running time, we adopt principal component analysis (PCA) to reduce vectors' dimension before clustering data. We put the data into an $n \times m$ matrix $\mathcal{A}$, where n stands for SNP quantity, and m represents human sample quantity. Hence the rows of $\mathcal{A}$ contain all the information of the SNP site. The covariance matrix of $\mathcal{A}$ is denoted as:

$$\mathcal{C} = cov(\mathcal{A}) = \mathcal{A}^T \mathcal{A}. \tag{1}$$

Then, we can reduce the dimension of $\mathcal{A}$ through the eigenvalue and eigenvector of matrix $\mathcal{C}$. To elaborate a little further, if we want to transform $\mathcal{A}$ into a k ($k < m$) dimension matrix, we choose the top k eigenvectors with the biggest eigenvalues. These eigenvectors form a new $n \times k$ matrix $\mathcal{K}$ for SNP site clustering, which is also called the preprocess of the tagSNP selection.

B. SNP SITE CLUSTERING

TagSNPs possess the main characteristics of diverse SNP sites. If the position distance between two SNP sites is more than 200KB (1KB = 1000sites), we can deem that their correlation is weak. Moreover, position partition can be achieved by a clustering algorithm. However, a graph with more than 200000 nodes has a high complexity even though using similar points as test dateset usually yields better performance.

In this paper, we adopt K-means as the clustering algorithm, which aims for partitioning n observations $(x_1, x_2, \ldots, x_n)$ into k groups [19]. Specifically, it regards vector's mean value as the group center, i.e. $(\mu_1, \mu_2, ..., \mu_k)$. Moreover, the distortion function of K-means algorithm can be expressed as:

$$J(k, \mu) = \sum_{i=1}^{N} \|x^{(i)} - \mu_k^{(i)}\|^2. \tag{2}$$

At the beginning, the algorithm randomly selects k centers. Then, it commences on iterating until the distortion function $J(\cdot)$ converges to the minimum, which means that the clustering results tend to be stable. $J(\cdot)$ in (2) relies on the Euclidean distance, which calculates the true distance in a two-dimensional space.

A large distance value means a low similarity between two vectors. Meanwhile, there are also some other distance metrics to denote the degree of the similarity, which may affect the clustering results. In the following, we will introduce four distance metrics which can be adopted in SNP site clustering. Their clustering performances are compared in Section V.

The first one is *Pearson correlation coefficient* function, i.e.,

$$T_{\text{Pearson}}(x, y) = \frac{n \sum_{i=1}^{N} x_i y_i - \sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i}{\sqrt{n \sum_{i=1}^{N} x_i^2 - (\sum_{i=1}^{N} x_i)^2} \sqrt{n \sum_{i=1}^{N} y_i^2 - (\sum_{i=1}^{N} y_i)^2}}. \tag{3}$$

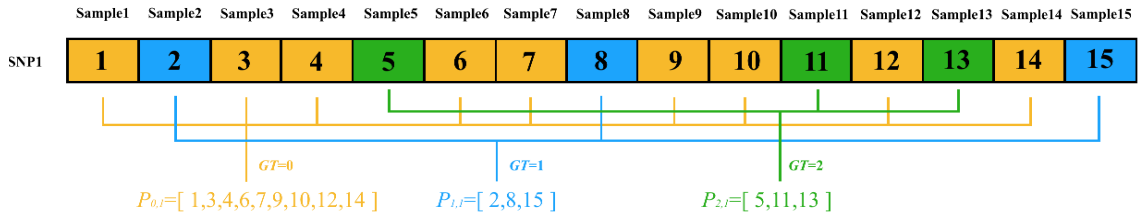


FIGURE 2. A toy example of constructing $P_j = \{P_{j,i}, j = 0, 1, 2\}$, where the yellow square represents $GT = 0$, the blue square denotes $GT = 1$, while the green square refers to $GT = 2$.

It spans from $[-1, 1]$, where 1 denotes completely positive correlation, while -1 represents completely negative correlation.

The second one is *cosine similarity*, which utilizes the cosine value of the two vectorized data in the high dimensional space to represent the difference between them. It can be calculated as:

$$T_{\text{Cosine}}(x, y) = \frac{\sum_{i=1}^{N} x_i y_i}{\sqrt{\sum_{i=1}^{N} x_i^2}\sqrt{\sum_{i=1}^{N} y_i^2}}. \tag{4}$$

Compared with other distance metrics, cosine similarity focuses its attention on the difference of two vectors in terms of their direction, other than distance or length.

The third one is *Tanimoto coefficient*, which uses the proportion of the common features of two vectors to determine the similarity, i.e.,

$$T_{\text{Tanimoto}}(x, y) = \frac{\sum_{i=1}^{N} x_i y_i}{\sqrt{\sum_{i=1}^{N} x_i^2} + \sqrt{\sum_{i=1}^{N} y_i^2} - \sum_{i=1}^{N} x_i y_i}, \tag{5}$$

The last one is *linkage disequilibrium* (LD), which demonstrates the population genetics phenomenon. It means the alleles at different loci have a non-random association. Here we suppose that allele A occurs at one locus with frequency p_A, allele B occurs at another locus with frequency p_B, and allele A and allele B both occurs with frequency p_{AB}. Then we can define coefficient of linkage disequilibrium as:

$$D_{AB} = p_{AB} - p_A p_B, \tag{6}$$

which is widely used to compare two sites. For the convenience of calculation, in this paper, we adopt another form of LD, i.e.,

$$r_{AB}^2 = \frac{D_{AB}^2}{p_A(1 - p_A)p_B(1 - p_B)}. \tag{7}$$

The value of r_{AB}^2 is between 0 and 1. If $r_{AB}^2 = 1$, these two sites have completely synergistic genetic. While two sites have irrelevant genetic if $r_{AB}^2 = 0$. Hence $1 - r_{AB}^2$ can be regarded as a distance metric to compare two SNP sites.

IV. GRAPH THEORY AIDED TAGSNP SELECTION

TagSNPs have the linkage disequilibrium phenomenon against the SNP sites which they stand for. The calculation results obtained by *Haploview* software [20] are shown

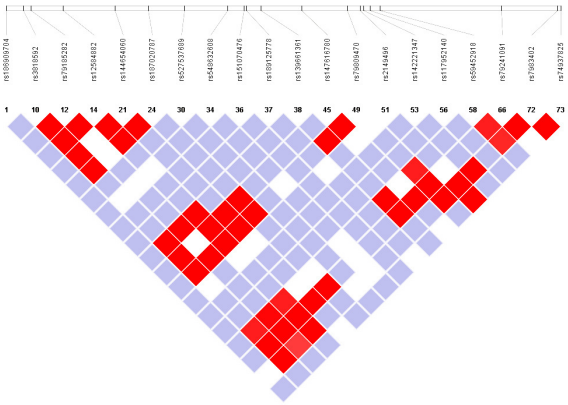
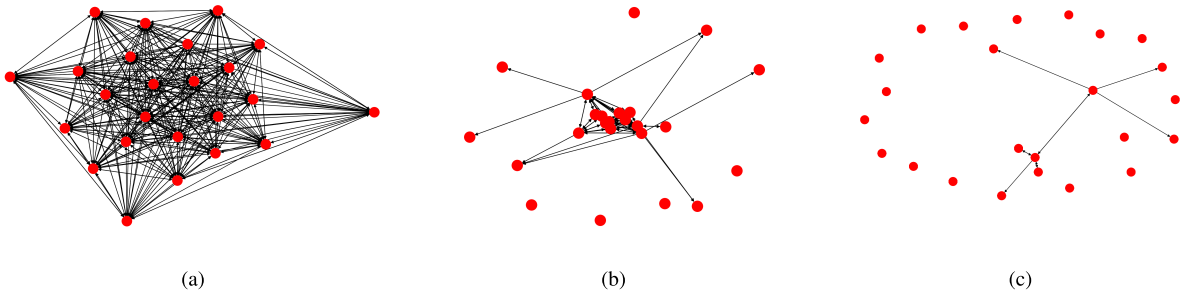


FIGURE 3. Linkage disequilibrium phenomenon, where the red square in the graph denotes the D value, while the violet square represents r^2 value. The strength between two SNP sites is proportional to the depth of the color. Hence, we can use the LD value to divide SNP data into haplotype blocks.

in Fig. 3, where the red block represents the degree of association, and the deeper the color, the greater the degree of association. SNP sites associated with each other appear in blocks, and as the distance between the sites increases, the degree of association decreases. Hence, we can divide them according to the position of the locus, which is also a clustering method suitable for genetic data.

In Section III, we focus our attention on the data preprocessing. In the following, we conceive an oriented graph theory based algorithm for selecting tagSNPs. Since we already encode the genotype into three codes, i.e. 0, 1 and 2, relying on genomes encoding rules in Table. 1, as shown in Fig. 2, the yellow square represents GT = 0, the blue square denotes GT = 1, while the green square refers to GT = 2. Let L_i be the i-th row of the SNP matrix $\mathcal{A}$. We use the set $P_j = \{P_{j,i}, j = 0, 1, 2\}$, for storing the information of the sample's position in each SNP site, where vector $P_{j,i}$ represents the sample's position corresponding to the i-th row of the SNP matrix in terms of three genotype encodings.

Let us construct three oriented graphs as shown in Fig. 4, which represents three kinds of relationship between SNP sites. More explicitly, the nodes refer to the SNP sites. The direction of each edge in the oriented graph represents the relationship between SNP_i and SNP_j. If there exists an inclusion relationship between the pair of SNP sites, we connect them by portraying a directed edge from the large SNP site



(a) (b) (c)

FIGURE 4. The structure of three kinds of oriented graphs, i.e. G_0, G_1 and G_2. Specifically, the nodes refer to the SNP sites and the direction of each edge represents the mutual inclusion relation. Moreover, the weight of the edge denotes the distance between the related two nodes. The structure of the graph is constructed by invoking Algorithm 1 relying on the dataset, i.e. CHROM13: 23947562-23948200 in 1000 Genomes Project Phase 3. (a) The structure of G_0. (b) The structure of G_1. (c) The structure of G_2.

Algorithm 1 Constructing Three Oriented Graphs

Input: Vector L_i, SNP dataset S
Output: Oriented graph $\{G_0, G_1, G_2\}$

```
m ← the number of human samples
for k = 1 to m do
    for i = 1 to n do
        for j = 0 to 2 do
            add k to P_{j,i} when L_i(k) = j
        end for
    end for
end for
P_j ← {P_{j,i}}, j = 0, 1, 2
P ← {P_0, P_1, P_2}
P ← FILTER (P)
for j = 0 to 2 do
    for i, k = 0 to n do
        if X_{j,i} ⊃ X_{j,k} then
            G_j add an oriented edge from i to k
        else if X_{j,i} ⊂ X_{j,k} then
            G_j add an oriented edge from k to i
        else if X_{j,i} ⊄ X_{j,k}, X_{j,i} ⊅ X_{j,k}, X_{j,i} ∩ X_{j,k} ≠ ∅ then
            add a bi-directional between k and i in G_j
        else
            no edge between k and i in G_j
        end if
    end for
end for
```

to the small SNP site. If the two SNP sites do not have any inclusion relationship with each other but have an intersection relationship, then a bi-directional edge is established. If the two sites do not have any relationship, there is no edge between them. The algorithm of constructing aforementioned oriented graphs is summarized in Algorithm 1.

In Algorithm 1, the function of FILTER () is used to filter the same SNP sites. If two sites contain the same information, we can use any one of them to represent. Meanwhile, Fig. 4 also can be viewed as big data driven visualized results of SNP dataset, where we can figure out which SNP site contains the most information with the aid of the connection density.

After completing the construction of the oriented graphs, we commence on selecting tagSNPs. First of all, let us introduce the priority of the nodes. Herein, we utilize the distance to calculate the node's weight in the graph. To elaborate a little further, the initial weight of all the nodes is zero. When updating their weights, we should add the in-edge weight as well as subtract the out-edge weight. The size of the node and the length of the oriented edge in the graphs is proportional to the weight of the node and the weight of the edge. Hence, the node's priority can be explicitly seen from the big data driven visualized results.

Thus, relying on the achieved oriented graphs and the defined node's priority, we can use the inclusion relationship between the SNP sites to judge if the SNP site can be a tagSNP. Let us take three kinds of nodes A, B and C for example. Considering the SNP site, which is represented by node A, if it is a component of the set of the sites which have oriented edges from the node A, which means that it can be replaced, this SNP site is not a tagSNP and we can remove the node A as well as all the edges from it or to it from the graph. By contrast, if the SNP site, represented by node B, is not a component of the set of the sites which have directed edges from it, which means that this SNP site can not be replaced, we regard this SNP site as a candidate tagSNP. Hence, we have SNPscore $\leftarrow$ SNPscore $+ 1$. If the SNP site, represented by node C, does not have any relationship with the set the sites which have directed edges from it, we can conclude that the node C is unique and no other SNP sites can substitute it. Hence, it can be regarded as a candidate tagSNP. Specially, the node in G_1 which only has one edge with other nodes should not be selected. Finally, once an SNP site can be regarded as a tagSNP in all three graphs in terms of structure G_0, G_1 and G_2, we have SNPscore ≥ 1, which is the final tagSNP. The whole tagSNP selection algorithm is shown in Algorithm 2.

V. SIMULATION RESULTS AND DISCUSSIONS

In this section, we perform the performance evaluation of our proposed algorithm relying on real-world dataset.

Algorithm 2 tagSNP Selection

Input: Directed graph $\{G_0, G_1, G_2\}$
Output: tagSNP

SNPscore $\leftarrow 0$
for $j = 0$ to 2 **do**
  $n \leftarrow$ the number of G_j node
  for $i = 0$ to n **do**
    for $k = 0$ to n **do**
      edgeWeight$_i \leftarrow$ DISTANCE (node$_i$,node$_k$)
      nodeWeight$_i \leftarrow$ getNodeWeight (node$_i$,node$_k$)
    end for
  end for
  SORT (nodeWeight$_i$)
  for $i = 0$ to n **do**
    for $k = 0$ to n **do**
      $X_{\text{link},f} \leftarrow$ set which has an oriented edge from $X_{j,i}$
      if $X_{j,i} \supset \bigcup X_{\text{link},f}$ **then**
        SNPscore$_i \leftarrow$ SNPscore$_i + 1$
      else if $X_{j,i} \subseteq \bigcup X_{\text{link},f}$ **then**
        remove the node i and its related edges from G_j
      else if $X_{j,i} \cap \bigcup X_{\text{link},f} \neq \phi$ **then**
        SNPscore$_i \leftarrow$ SNPscore$_i + 1$
      else
        SNPscore$_i \leftarrow$ SNPscore$_i + 1$
      end if
    end for
  end for
end for
tagSNP $\leftarrow$ SNPscore$_i > 2$

Simulations are implemented by the MATLAB R2018a in a personal computer with Intel Core i7-4790 CPU including 3.6GHz and 8GB RAM. The dataset in our simulations are truncated from the 1000 Genomes Project dataset. Specifically, dataset 1 is from the genomes on the chromosome 13: 23551994-23552136, which includes 13 SNP sites. Dataset 2 is from the genomes on the chromosome 13: 23947562-23948200, containing 24 SNP sites. Moreover, dataset 3 is from the genomes on the chromosome 13: 99484338-99486883, which consist of 76 SNP sites. Dataset 4 is from the genomes on the chromosome 6: 133098746-133108745, including 299 SNP sites, and dataset 5 is from the genomes on the chromosome 6: 74123238-74161999, having 1104 SNP sites. All the datasets contain 2504 individuals.

First of all, we evaluate the performance of the data dimension reduction relying on the PCA algorithm parameterized by different number of SNP sites. As shown in Fig. 5, each point represents an eigenvalue of the covariance matrix $\mathcal{C}$. We show portray these eigenvalues in terms of a descending order. The value of the eigenvalue represents the data information in a new space. We select some of the largest eigenvectors as the principal components in the

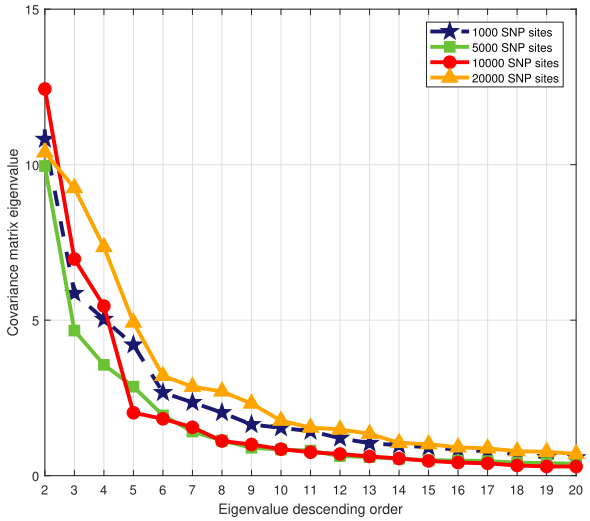


FIGURE 5. The eigenvalue of covariance matrix $\mathcal{C}$ in terms of the descending order parameterized by the number of SNP sites. Here, simulation is conducted based on part of 99367 SNP sites on chromosome 22.

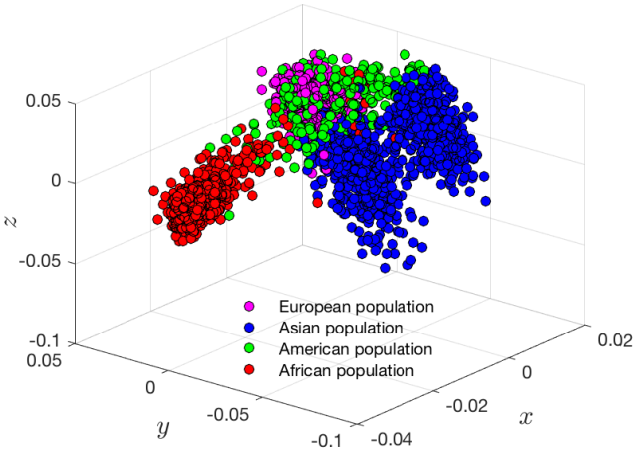


FIGURE 6. Population distribution by selecting three principle components (The x axis represents the first principal component of PAC, termed as PCA-1; the y axis denotes the second principal component of PAC, termed as PCA-2; the z axis represents the third principal component of PAC, termed as PCA-3).

PCA algorithm. In Fig. 5, we can conclude that the value of eigenvalues declines rapidly, which decides, to some degree, we are capable of reducing the dimension of the dataset. Meanwhile, the eigenvalue is close to zero after the 20th eigenvalue, and hence we reduce the dimension of the matrix $\mathcal{A}_{n\times 2504}$ into $\mathcal{A}_{n\times 20}$.

As a toy example, we select three largest eigenvectors in Fig. 5 associated with our three principal components, and we show the corresponding population distributions in Fig. 6, where the axis of x, y and z represent the first, the second and the third principal component of PCA, respectively. Individuals living in four different continents are demonstrated. We can find that individuals living in the same continent are clustered together. Our simulation results are consistent with the biological phenomena, i.e. the allopatric speciation.

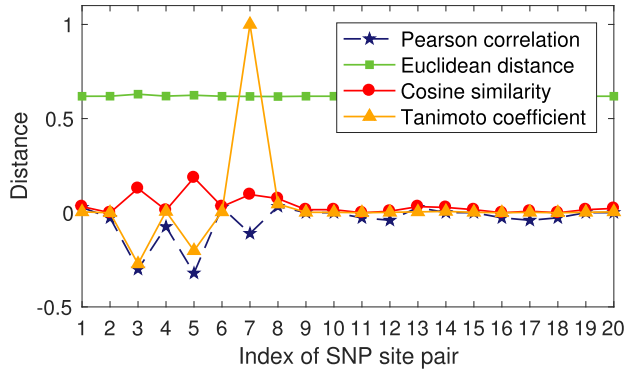


FIGURE 7. The distance of SNP site pairs calculated by four different clustering metrics.

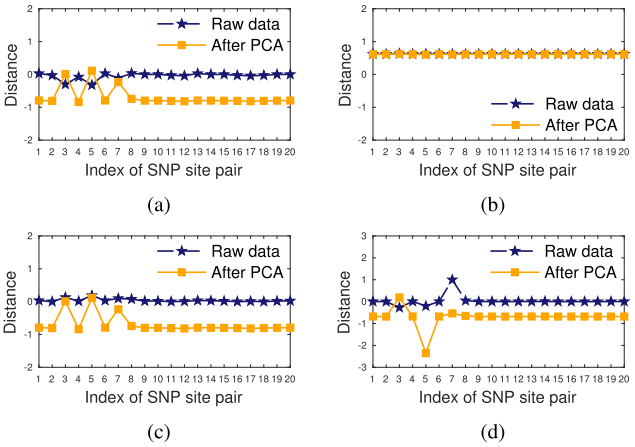


FIGURE 8. Distance relying on four different clustering metrics. (a) Pearson correlation. (b) Euclidean distance. (c) Cosine similarity. (d) Tanimoto coefficient.

Hence, we can draw the conclusion that the PCA algorithm is efficient in analyzing the genetic data.

In the following, we calculate the similarity of the SNP sites based on the genetic datasets considered, which is calculated by the distance metrics proposed in Section III. As shown in Fig. 7, we show the distance of each SNP site pair relying on the raw data. More explicity, we can conclude that the Euclidean distance function cannot beneficially distinguish the difference between different SNP site pairs, which means that if the Euclidean distance is used for clustering SNP sites, we may not obtain a superior result because most of SNP sites are categorized into the same cluster group. By contrast, the other three distance functions have more explicitly distinctive results.

Furthermore, we compare the performance of the distance between SNP site pairs relying on both raw data and preprocessed data calculated by four different clustering metrics, which is shown in Fig. 8. As shown in Fig. 8 (a) and (d), the distance performance calculated by both the Pearson correlation and the Tanimoto coefficient after the PCA algorithm with 20 principle components has a wrong tendency with that relying on the raw SNP sites. It means that the PCA algorithm may influence the result of distance calculated by the Pearson

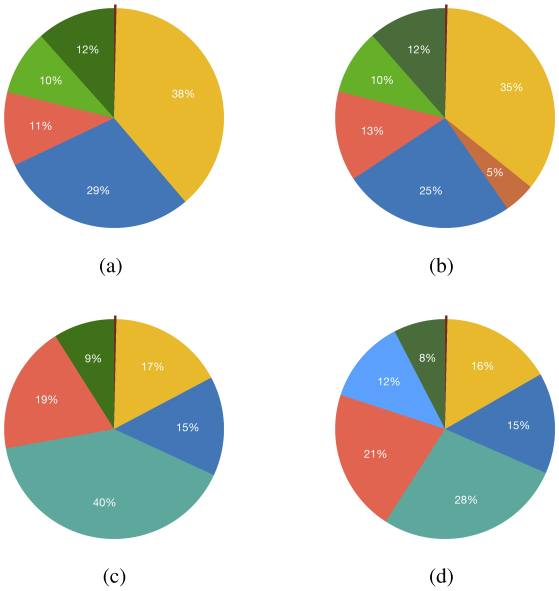


FIGURE 9. Cluster SNP sites by using K-means. (a)(b) shows the result of clustering 1104 SNP sites into 6 and 7 groups. (c)(d) shows the result of clustering 20000 SNP sites into 6 and 7 groups.

correlation and the Tanimoto coefficient. Fig. 8 (b) shows that the Euclidean distance cannot distinguish the difference between SNP site pairs relying on both raw data and the data preprocessed by PCA algorithms. However, as shown in Fig. 8 (c), the distance calculated by the cosine similarity after PCA preprocessing has the same tendency with that relying on the raw SNP sites. Combining the results from both Fig. 7 and Fig. 8, we select the cosine similarity as the metric for calculating the distance between SNP site pairs in our following simulations including clustering sites and tagSNP selection algorithm. As Fig. 9 (a)(b) shown, clustering SNP sites into small groups won't improve the efficiency of algorithms if the quantity of SNP sites is small due to similarity of SNP sites. The experiment results show that we get 112, 189, 162 and 212 tagSNPs when clustering these sites into 2, 3, 4 and 5 groups. So we get more tagSNPs than we use whole SNP sites directly. As Fig. 9 (c)(d) shown, we can cluster 20000 SNP sites into 5 and 6 groups. And cluster algorithm can reduce the whole process running time because of simplifying graph method.

Fig.10 shows the number of candidate tagSNPs in different graph structures relying on our proposed tagSNP selection algorithm. The y axis of the histogram represents the number of candidate tagSNPs in each graph structure. we can see the number of candidate tagSNPs in the G_2 is the largest, because there are many special points as shown in Fig. 4. These points do not have any edges connecting to the others, which means that no other points can stand for them, and hence they are selected as the candidate tagSNPs. By contrast, there are few candidate tagSNPs in graph G_0, because in our SNP matrix $\mathcal{A}$, most of the genotype elements are 0, indicating that there exists edges connecting to other points, they have common characteristics with other tagSNPs.

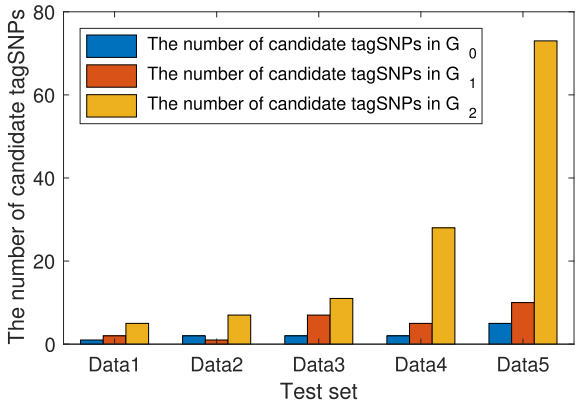


FIGURE 10. The number of candidate tagSNPs in different graph structures.

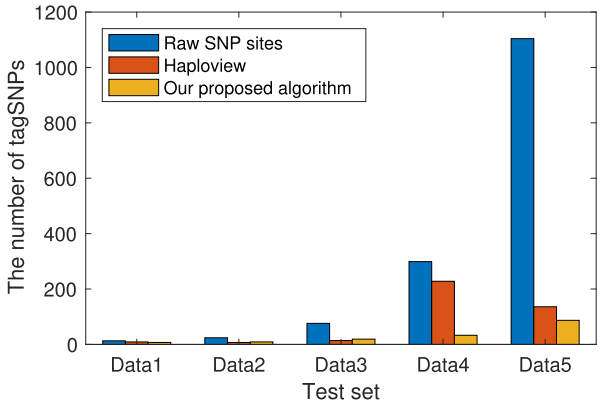


FIGURE 11. The performance comparison in terms of the number of candidate tagSNPs between Haploview and our proposed algorithm.

For the performance comparison in terms of the number of candidate tagSNPs. This paper shows that our algorithm can select fewer tagSNPs than traditional method. Here we choose the authoritative software Haploview as the comparison results. As shown in Fig. 11, based on dataset 1, dataset 2 and dataset 3, the number of tagSNPs selected by our proposed algorithm is close to that selected by Haploview. However, relying on dataset 4 and dataset 5, the number of candidate tagSNPs selected by our proposed algorithm is much fewer than that by Haploview. Specifically, based on dataset 4, Haploview selects almost 76% SNP sites as tagSNPs, while we only use 11% SNP sites for representation. Similarly, relying on dataset 5, Haploview selects 12.3% SNP sites acting as the tagSNPs, while we choose 7.8% SNP sites.

There are several prediction algorithms for predicting SNPs relying on tagSNPs, such as STAMPA [3] and SVM [21]. In [21], He and Zelikovsky utilized the SVM algorithm for predicting SNPs, which yielded a high accuracy rate in comparison to the multiple linear regression (MLR) method and the STAMPA algorithm. In this paper, given the advantage of SVM method in SNP prediction, we also use SVM algorithm to show the effectiveness of our SNP prediction algorithm. More specifically, the tagSNPs selected

TABLE 2. Accuracy rate of our proposed tagSNP selection algorithm.

Dataset	Successfully predicted SNPs	Accuracy Rate
Dataset 3	2850	99.30%
Dataset 4	13300	97.68%
Dataset 5	101700	96.52%

by our proposed tagSNP selection algorithm are acted as training data, and the un-tagSNPs are used for testing. We try to predict a single individual genotype of those un-tagSNPs and compare the predicted value with the true value. The prediction accuracy rate of our proposed tagSNP selection algorithm is shown in TABLE. 2. The result shows that the tagSNPs we select can be used to predict unknown SNP genotype.

VI. CONCLUSIONS

In this paper, we propose an efficient tagSNPs selection method based on big data mining techniques. The PCA based algorithm reduces the dimension of the raw gene data, which improves both the efficiency and the accuracy of SNP site clustering. Furthermore, our tagSNPs selection algorithm based on oriented graphs extracts key tagSNPs relying on defined node priority. Experiments on 1000 Genomes Project dataset verify the effectiveness and efficiency of our designed SNP classifier. In our future research, we would like to develop a software for biologists to accelerate the process of selecting tagSNPs, which is beneficial in terms of both diagnosing genetic diseases as well as of reducing the cost of medical research.

REFERENCES

[1] The 1000 Genomes Project Consortium, "A map of human genome variation from population-scale sequencing," *Nature*, vol. 467, no. 7319, pp. 1061–1073, Oct. 2010.
[2] R. Alzubi, N. Ramzan, H. Alzoubi, and A. Amira, "A hybrid feature selection method for complex diseases SNPs," *IEEE Access*, vol. 6, pp. 1292–1301, 2018.
[3] E. Halperin, G. Kimmel, and R. Shamir, "Tag SNP selection in genotype data for maximizing SNP prediction accuracy," *Bioinformatics*, vol. 21, no. 1, pp. i195–i203, 2005.
[4] M. G. Kibria, K. Nguyen, G. P. Villardi, O. Zhao, K. Ishizu, and F. Kojima, "Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks," *IEEE Access*, vol. 6, pp. 32328–32338, 2018.
[5] Z. K. Gao, Q. Cai, Y. X. Yang, N. Dong, and S. S. Zhang, "Visibility graph from adaptive optimal kernel time-frequency representation for classification of epileptiform EEG," *Int. J. Neural Syst.*, vol. 27, no. 4, p. 1750005, Jun. 2017.
[6] W. Zhang, J. Wang, X. Zhang, K. Zhang, and Y. Ren, "A novel cardiac arrhythmia detection method relying on improved DTW method," in *Proc. IEEE 2nd Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Chongqing, China, Mar. 2017, pp. 862–867.
[7] I. Bertin, J. H. Zhu, and M. D. Gale, "SSCP-SNP in pearl millet—A new marker system for comparative genetics," *Theor. Appl. Genet.*, vol. 110, no. 8, pp. 1467–1472, May 2005.
[8] The 1000 Genomes Project Consortium, "A global reference for human genetic variation," *Nature*, vol. 526, no. 7571, pp. 68–74, Oct. 2015.
[9] F. Frommlet, "Tag SNP selection based on clustering according to dominant sets found using replicator dynamics," *Adv. Data Anal. Classification*, vol. 4, no. 1, pp. 65–83, 2010.
[10] B. Liao, X. Wang, W. Zhu, X. Li, L. Cai, and H. Chen, "New multilocus linkage disequilibrium measure for tag SNP selection," *J. Bioinf. Comput. Biol.*, vol. 15, no. 1, p. 1750001, 2017.

[11] T. Müller, K. Ickstadt, and S. Selinski, "Cluster analysis: A comparison of different similarity measures for SNP data," Tech. Univ. Dortmund, Dortmund, Germany, Tech. Rep. SFB 475, 2005.
[12] V. Radhakrishna, P. V. Kumar, and V. Janaki, "Krishna sudarsana: A z-space similarity measure," in *Proc. 4th Int. Conf. Eng.*, 2018, pp. 1–4.
[13] G. R. Kumar, N. Mangathayaru, and G. Narsimha. (2016). "A novel similarity measure for intrusion detection using Gaussian function." [Online]. Available: https://arxiv.org/abs/1604.07510
[14] I. Ihan and G. Tezel, "Tag SNP selection using clonal selection and majority voting algorithms," *Int. J. Data Mining Bioinf.*, vol. 16, no. 4, pp. 290–311, 2016.
[15] C.-H. Yeh and J.-W. Jheng, "An iterative algorithm for tag SNP selection based on information entropy analysis," *J. Signal Process. Syst.*, vol. 64, no. 2, pp. 233–239, Aug. 2011.
[16] J. Wang, M.-Z. Guo, and C.-Y. Wang, "CGTS: A site-clustering graph based tag SNP selection algorithm in genotype data," *Bioinformatics*, vol. 10, no. 1, p. s71, 2009.
[17] P. H. Lee and H. Shatkay, "BNTagger: Improved tagging SNP selection using Bayesian networks," *Bioinformatics*, vol. 22, no. 14, pp. e211–e219, Jul. 2006.
[18] C.-J. Chang, Y.-T. Huang, and K.-M. Chao, "A greedier approach for finding tag SNPs," *Bioinformatics*, vol. 22, no. 6, pp. 685–691, Mar. 2006.
[19] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.
[20] J. C. Barrett, B. Fry, J. Maller, and M. J. Daly, "Haploview: Analysis and visualization of LD and haplotype maps," *Bioinformatics*, vol. 21, no. 2, pp. 263–265, Jan. 2005.
[21] J. He and A. Zelikovsky, "Informative SNP selection methods based on SNP prediction," *IEEE Trans. Nanobiosci.*, vol. 6, no. 1, pp. 60–67, Mar. 2007.

TIANSHUO CONG received the B.S. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2017, where he is currently pursuing the Ph.D. degree with the Center for Cryptology Study, Institute for Advanced Study. His research interests include the field of big data modeling and data mining in bioinformatics, efficient implementation of cryptographic algorithms, and integration of biometric and cryptographic technology.

JINGJING WANG (S'14) received the B.S. degree (Hons.) in electronic information engineering from the Dalian University of Technology, in 2014. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Tsinghua University, Beijing. From 2017 to 2018, he was a joint Ph.D. Student with the Next Generation Wireless Group chaired by Prof. L. Hanzo, University of Southampton, U.K. His research interests include the resource allocation and network association, learning theory-aided modeling, analysis and signal processing, and information diffusion theory for mobile wireless networks. He received the Tsinghua GuangHua Scholarship Award, in 2016, and the Graduate China National Scholarship Award, in 2017.

SANGHAI GUAN received the B.Eng. degree in electronic engineering from the Dalian University of Technology, Dalian, Liaoning, China, in 2017, with the honor of excellent graduate of Liaoning Province. He is currently pursuing the M.S. degree in information and electronic engineering with Tsinghua University, Beijing, China. His research interests include complex networks and systems, multi-agent networked systems, and network association.

YIFEI MU received the B.S. and M.S. degrees from Xi'an Jiaotong University, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree from Clemson University, USA, in 2013. He is currently an Associate Professor with the Dongbei University of Finance and Economics. His research interests include big data modeling and analysis, game theory, and machine learning algorithms both in data science and financial fields.

TONG BAI (S'15) received the B.Sc. degree in telecommunications from Northwestern Polytechnical University, Xi'an, China, in 2013, and the M.Sc. degree (Hons.) in wireless communications from the University of Southampton, U.K., in 2014, where he is currently pursuing the Ph.D. degree with the Next Generation Wireless Group. His research interests include the optimization and signal processing in both wireless and wireline communications.

YONG REN (SM'16) received the B.S., M.S., and Ph.D. degrees in electronic engineering from the Harbin Institute of Technology, China, in 1984, 1987, and 1994, respectively. He held a Postdoctoral position with the Department of Electronics Engineering, Tsinghua University, China, from 1995 to 1997. He is currently a Professor with the Department of Electronics Engineering and the Director of the Complexity Engineered Systems Laboratory, Tsinghua University. He holds 12 patents and has authored or co-authored more than 100 technical papers in the behavior of computer network, P2P network, and cognitive networks. His current research interests include complex systems theory and its applications to the optimization and information sharing of the Internet, the Internet of Things, and ubiquitous network, cognitive networks, and cyber-physical systems. He serves as a Reviewer of the *IEICE Transactions on Communications*, *Digital Signal Processing*, *Chinese Physics Letters*, the *Chinese Journal of Electronics*, the *Chinese Journal of Computer Science and Technology*, and the *Chinese Journal of Aeronautics*.