Final Report

Project Title: **Iterative Neighbor-Joining Tree Clustering Framework for Population Structure Studies**

By

Assistant Professor Dr. Tulaya Limpiti

December 2016

Contract No. TRG5780045

Final Report

Project Title: Iterative Neighbor-Joining Tree Clustering Framework for Population Structure Studies

Researcher	Institute
1. Assistant Professor Dr. Tulaya Limpiti	King Mongkut's Institute of Technology Ladkrabang
2. Dr. Sissades Tongsima	National Center for Genetic Engineering and Biotechnology (BIOTEC)

This project is granted by the Thailand Research Fund

Table of contents

Abstract

Project Code: TRG5780045

Project Title: Iterative Neighbor-Joining Tree Clustering Framework for Population Structure Studies

Investigator: Assistant Professor Dr. Tulaya Limpiti

E-mail Address: tulaya.li@kmitl.ac.th

Project Period: 2 June 2014 to 1 December 2016

Abstract:

In population structure analysis, genetic variations, e.g., single nucleotide polymorphisms, are used to characterize commonality and difference of individuals from various populations. At present, high-complexity, high-dimensional genotypic data sets are common. Thus, an efficient way to handle such data sets is desirable.

In this research project, we design two algorithms for population structure studies. First, we develop a new, efficient graph-based clustering framework for resolving population structure called the *iNJclust* algorithm. The algorithm operates iteratively on the Neighbor-Joining (NJ) tree. The framework uses well-known genetic measurements, namely the allele-sharing distance and the fixation index. The behavior of the fixation index is proven mathematically and is utilized as a stopping criterion. The algorithm provides an estimated number of populations, individual assignments, and relationships between populations in terms of a binary population tree as outputs. The accurate clustering performance and robustness of the iNJclust algorithm are demonstrated using simulated and real data sets from bovine, sheep, and human populations.

To cope with high computational cost and faulty substructure detected from noisy data due to redundant or non-informative SNPs, efforts have been done to extract a smaller informative SNP subset that still represents the same intrinsic structure of populations as the full panel of SNPs. The second part of this research describes an informative marker selection technique based on principal component analysis (PCA). It improves upon another technique called PCA-correlated SNPs. A new informativeness score based on a basis function expansion of the SNP variation patterns across individuals is introduced. Such score is computed for each SNP to select a subset of SNPs with the best scores. Using a bovine data set, we demonstrate that our technique is superior to the PCA-correlated SNPs method. Our method is simple, efficient, and is robust to the assumed rank of the data. High data representation accuracy is also achieved after a significant reduction of the number of SNPs while retaining information about the underlying population structure from the original data.

Keywords: clustering, neighbor-joining tree, population structure, principal component analysis, informative SNPs

บทคัดย่อ

รหัสโครงการ: TRG5780045

ชื่อโครงการ: การแบ่งกลุ่มแบบวนซ้ำบนเนเบอร์จอยนิงทรีสำหรับการศึกษาโครงสร้างประชากร

ชื่อนักวิจัย: ผู้ช่วยศาสตราจารย์ ดร. ตุลยา ลิมปิติ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

E-mail Address: tulaya.li@kmitl.ac.th

ชื่อนักวิจัยที่ปรึกษา: ดร.ศิษเฎศ ทองสิมา

ระยะเวลาโครงการ: 2 มิถุนายน 2557 ถึง 1 ธันวาคม 2559

บทคัดย่อ:

การศึกษาโครงสร้างประชากรจากความแตกต่างของความถี่อัลลีลในข้อมูลทางพันธุกรรม เช่นข้อมูลสนิป (Single nucleotide polymorphisms หรือ SNPs) เป็นหนึ่งในหัวข้อที่สำคัญในงานวิจัยด้านพันธุกรรมประชากร ความเข้าใจในโครงสร้างดังกล่าวทำให้นักวิจัยสามารถระบุถึงตัวแปรในการวิวัฒนาการของเผ่าพันธุ์สิ่งมีชีวิตที่ทำให้เกิดความหลากหลายทางพันธุกรรม ในปัจจุบันชุดข้อมูลสนิปมักมีขนาดใหญ่และมีความซับซ้อนสูง อัลกอริธึมที่สามารถวิเคราะห์และจัดการข้อมูลได้อย่างมีประสิทธิภาพจึงเป็นที่ต้องการยิ่ง โครงการวิจัยนี้ได้ออกแบบอัลกอริธึมสำหรับการศึกษาโครงสร้างประชากรไว้สองวิธี งานวิจัยส่วนแรกได้พัฒนากรอบการวิเคราะห์เชิงกราฟสำหรับการแบ่งกลุ่มประชากรชื่อว่าอัลกอริธึม iNJclust ซึ่งทำงานแบบวนซ้ำบนแผนภูมิเนเบอร์จอยนิงทรี โดยใช้ปริมาณทางพันธุกรรมที่เป็นที่รู้จักดี ได้แก่ ระยะห่างอัลลีล (allele-sharing distance) และดัชนีฟิกเซชัน (fixation index) และค้นพบทฤษฎีที่เกี่ยวข้องกับคุณสมบัติของดัชนีฟิกเซชันโดยใช้การพิสูจน์ทางคณิตศาสตร์ และนำมาใช้เป็นเงื่อนไขการหยุดทำงานของอัลกอริธึม ผลลัพธ์ของอัลกอริธึมได้แก่ การประมาณจำนวนกลุ่มประชากรย่อย การแบ่งกลุ่ม และจำแนกข้อมูลตัวอย่าง และแผนภูมิต้นไม้ที่แสดงโครงสร้างความสัมพันธ์ระดับกลุ่มประชากร การทดสอบการทำงานกับชุดข้อมูลสนิปขนาดใหญ่จากวัว แกะ และมนุษย์ พบว่าอัลกอริธึมไอเอ็นเจคลัสต์ให้ผลลัพธ์ที่ถูกต้อง

เพื่อแก้ปัญหาการใช้ทรัพยากรการคำนวณที่สิ้นเปลือง และการระบุโครงสร้างย่อยที่ไม่ถูกต้องจากสัญญาณรบกวนที่เกิดจากข้อมูลซ้ำซ้อนหรือสนิปบางส่วนไม่ได้มีข้อมูลเชิงโครงสร้างที่สำคัญ งานวิจัยในส่วนที่สองจึงมุ่งเน้นหาหลักเกณฑ์ในการเลือกกลุ่มของสนิปขนาดเล็กที่มีข้อมูลสำคัญเกี่ยวกับโครงสร้างประชากรเทียบเท่ากับข้อมูลดั้งเดิมโดยใช้การวิเคราะห์บนมิติของพรินซิเพิลคอมโพเนนท์ โดยในงานวิจัยได้พัฒนาการจัดอันดับคะแนนความสำคัญของสนิปแต่ละตำแหน่งโดยปรับปรุงมาจากไอเดียของอัลกอริธึม PCA-correlated SNPs การวัดคะแนนแบบใหม่ที่ได้พัฒนาขึ้นมีพื้นฐานมาจากการอธิบายการกระจายตัวของสนิปแต่ละตำแหน่งบนกลุ่มตัวอย่างโดยใช้การขยายด้วยเบสิสฟังก์ชัน และทำการลดขนาดข้อมูลให้เหลือเพียงกลุ่มของสนิปขนาดเล็กที่มีคะแนนสูงสุด การทดสอบอัลกอริธึมที่พัฒนาขึ้นกับข้อมูลจากวัวแสดงให้เห็นว่าอัลกอริธึมมีความเรียบง่ายและมีประสิทธิภาพการทำงานเหนือกว่าอัลกอริธึม PCA-correlated SNPs และยังไม่ขึ้นกับลำดับของข้อมูล นอกจากนี้ยังสามารถลดขนาดข้อมูลลงอย่างมีนัยสำคัญ โดยยังคงรักษาข้อมูลสำคัญด้านโครงสร้างประชากรจากข้อมูลดั้งเดิมไว้อย่างครบถ้วน

คำหลัก: การแบ่งกลุ่ม เนเบอร์จอยนิงทรี โครงสร้างประชากร การวิเคราะห์พรินซิเพิลคอมโพเนนท์ สนิปสำคัญ

I. Executive Summary

Introduction:

Understanding genetic differences among populations is one of the most important issues in population genetics. Genetic variations, e.g., single nucleotide polymorphisms, are used to characterize commonality and difference of individuals from various populations and analyze population structures. Applications of such discovery include population ancestry and migration studies, association studies, or even breed composition and traceability of livestock. Using today's high throughput genotyping platforms, a data set may contain several thousands of individuals, each of which has a million of SNPs to be analyzed. There are two approaches in designing algorithms for population studies. The first approach relies on genetic model in which each individual is assigned with inferred ancestral contributions. Bayesian inference is implemented directly into these algorithms in order to cluster the individuals and analyze the underlying population structure. The second approach is non-parametric, and population structure is analyzed on different spaces, e.g. using principal component analysis (PCA) or various distance matrices. Limitations of existing algorithms in both categories include high computational cost, inaccurate clustering, or obscurity of inferred population structure. Faulty substructure may also be detected if the data is noisy from redundant or non-informative SNPs. Considerable efforts have been done to extract a smaller informative SNP subset that still represents the same intrinsic structure of populations within a data set as the full panel of SNPs. Thus, more efficient ways to handle such high-complexity, high-dimensional data sets are desirable.

Objectives:

This research project addresses the problem of analyzing population structure within genotypic data sets with high complexity. First, we aim to develop a graph-based clustering framework that uses relatedness information between populations provided by phylogenetic trees to resolve complex population structure. Secondly, we would like to investigate the problem of finding the subset of SNPs markers that contain important information on the intrinsic population structure using principal component analysis.

Methodology:

We design two different algorithms for population structure studies. First, we develop a new computational framework for automatically classifying individuals to clusters and call it an "iterative neighbor-joining clustering" or *iNJclust* algorithm. Genetic similarity information inherits in neighbor-joining (NJ) tree, which is a commonly used phylogenetic tree, is used for resolving population structure. Instead of clustering from PCA-derived data points, the iNJclust algorithm performs a graph-based clustering on the NJ tree constructed using an allele-sharing distance (ASD) matrix. Data points are viewed as nodes of a graph, whereas the graph topology captures the pattern of

clusters. We perform clustering on a graph by selectively cutting the longest edge of a minimum spanning tree. An iterative process is also adopted so that in each iteration, a new NJ tree is constructed for clustering. The framework uses well-known genetic measurements, namely the allele-sharing distance, and the fixation index. The behavior of the fixation index is investigated and utilized in the algorithm's stopping criterion. The iNJclust algorithm provides an estimated number of populations, individual assignments, and relationships between populations as outputs. The clustering result is reported in the form of a binary population tree, whose terminal nodes represent the final inferred populations and the tree structure preserves the genetic relationships among them.

For the second part of our research, we establish a simple and efficient PCA-based informative marker selection technique. We improve upon another spectral analysis technique called PCA-correlated SNPs method by Paschou et al. proposed in 2007. A basis function expansion viewpoint of the SNP variation patterns across individuals is adopted to suggest a new informativeness score. The construction of our method is such that the bases are orthonormal. This informative score is computed for each SNP loci. The score for all SNP markers are ranked and a subset of SNPs with the best scores is deemed the most informative.

Results and Discussion:

The clustering performance and the robustness of the iNJclust algorithm are tested using simulated and real data sets from bovine, sheep, and human populations. We compare the iNJclust algorithm against existing clustering algorithms of similar natures, namely the AWclust algorithm and the NJclust algorithm and also corroborate its results with the Admixture patterns. The results illustrate that the iNJclust algorithm outperforms the other algorithms. It is observed that our proposed algorithm operates in a computationally efficient manner. In addition, the iterated tree reconstruction process is crucial for accurate clustering results. The results also indicate that it can effectively handle irregular cluster patterns. The result indicates that the number of populations within each data set is reasonably estimated, the individual assignment is robust, and, although primitively, the structure of the inferred population tree corresponds to the intrinsic relationships among populations within the data. However, there is a limitation of inferring population tree topology with admixed individuals, as people with admixture may be assigned to different branches on the tree to which they have similarities.

We give mathematical prove for the behavior of the fixation index after each iteration and utilize it as the iNJclust algorithm's stopping criterion called ΔF. Although the stopping criterion based on this fixation index property is mathematically sound and flexible, no process is discovered to produce an optimal value of the ΔF criterion. In stead, ranges of decent ΔF values for different data complexity have been suggested heuristically after an extensive investigation on many experimental data sets.

Using a bovine data set, we demonstrate that our technique for identifying structure informative SNPs is superior to the PCA-correlated SNPs method, which requires accurate rank estimation to perform well. In contrast, it is demonstrated that our result is robust to the assumed rank of the data, i.e., the choice of a rank estimation technique has little effect on the final selection of informative SNPs. In fact, rank estimation may be bypassed with negligible degradation in data representation accuracy. Additionally, sizable dimensional reduction can be achieved using a very small subset of structure informative SNPs, while retaining information on the underlying population structure from the original data.

Future direction:

For an extension of this work, we plan to look at the performance of our methods on additional human or animal data sets with varying complexities. We would like to validate the hypothesis that our techniques are advantageous in the cases where we want to study the population structure at a finer scale, e.g. populations within continents or with common ancestry.

II. Objectives

This research project addresses the problem of analyzing population structure within genetic data by applying graph-based data clustering with genetic similarity information inherits in neighbor-joining (NJ) tree, which is a commonly used phylogenetic tree. We also investigate the problem of finding the subset of SNPs markers that contain important information on the intrinsic population structure using principal component analysis. The specific aims are as follows:

(1) To develop a new, efficient framework for resolving population structure which can handle genotype data sets with high complexity without a statistical data model.

(2) To develop an accurate graph-based iterative clustering algorithm with an appropriate stopping criterion.

(3) To derive a measure used for determining homogeneity of a subpopulation.

(4) To construct a condition for selecting a small set of informative SNPs markers based on principal component analysis.

(5) To demonstrate the efficacy of the proposed algorithms using simulated and real genetypic data sets.

III. Research Methodology

III.A The iNJclust framework: A new, efficient graph-based iterative clustering algorithm

We develop a new computational framework for automatically classifying individuals to clusters and call it "iterative neighbor-joining clustering" or *iNJclust*. The framework is non-parametric. However, instead of clustering from PCA-derived data points as typically done for existing non-parametric algorithms, the framework uses relatedness information between populations provided by the phylogenetics-based methods to resolve complex population structure. The iNJclust algorithm performs a graph-based clustering on the NJ tree constructed using an allele-sharing distance (ASD) matrix. Data points are viewed as nodes of a graph, whereas the graph topology captures the pattern of clusters. We perform clustering on a graph by selectively cutting the longest edge of a minimum spanning tree. We also adopt the iterative process from so that the iNJclust algorithm is computationally efficient.

(1) The iNJclust process

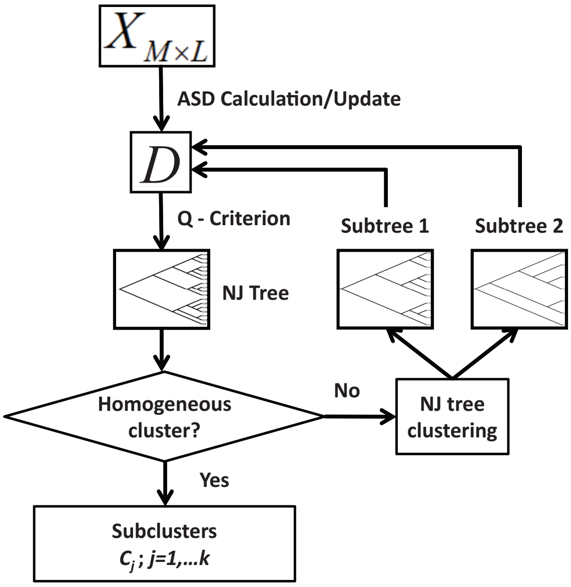

Figure 1: The system flowchart of our algorithm

The system flowchart of our algorithm is depicted in Fig. 1. SNP sequences of M individuals forms the input matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M]^T$. Each row vector $\mathbf{x}_i$, $i = 1,\ldots, M$ is genotyped from L loci of individual i, at which there are two alleles of either A (major allele) or a (minor allele) for three possible genotypes (AA, Aa, aa). The SNP sequence is encoded by counting the number of minor allele a. Therefore, $\mathbf{x}_i$ is an L-dimensional vector of numerical values 0,1, or 2.

After receiving the input matrix **X**, the algorithm computes the ASD matrix **D**, whose elements are

$$\mathbf{D}(i,j) = \frac{1}{L} \| \mathbf{x}_i - \mathbf{x}_j \|_1, i, j = 1, \ldots, M. \tag{1}$$

Therefore, **D**(*i,j*) is proportional to the L1-norm of the pairwise difference between the SNP sequences of individual *i* and individual *j*. The smaller the value of **D**(*i,j*), the closer the pair are genetically. The ASD matrix **D** is subsequently used to construct the NJ tree. Treating each individual as a leaf node, we compute the minimum evolution criteria or the Q criterion,

$$Q(i,j) = (r-1)d(i,j) - \sum_{k=1}^{r} d(i,k) - \sum_{k=1}^{r} d(j,k) \tag{2}$$

where *d*(*i,j*) is the Fitch-Margoliash distance between node *i* and *j*, and *r* is the number of the remaining nodes. A pair of nodes who are nearest to each other are merged into a parent node. The tree construction process continues until all nodes are merged onto the NJ tree. The algorithm then determines if the cluster is homogeneous, i.e., all individuals on the tree come from the same population. A theoretical measure for determining subpopulation homogeneity has been derived (details in the next section). If the cluster is said to be heterogeneous, the algorithm performs clustering on the NJ tree by bisecting the tree into two subtrees. The NJ tree is split at the longest branch (edge) between two nodes within the tree. In the next stage, both subtrees cycle back to the ASD update step where they are processed independently. The method iterates until all populations (in the form of subtrees) are considered homogeneous and output from the process as the final subpopulations.

Besides resolving population structure in three major aspects commonly performed in a clustering method: detecting population structure, predicting the number of populations within the dataset, and assigning individuals to predicted populations, the iNJclust algorithm also derives a bifurcated population tree based on the order at which each population are separated from the original dataset. The terminal nodes of the tree represent the final inferred populations and the tree structure preserves the genetic relationships among them.

(2) A measure for determining subpopulation homogeneity

The fixation index F_{st} , which is wildly used to measure homogeneity of populations in genetic, is defined as follows. Suppose a dataset of *M* individuals is composed of *N* populations {C_1, C_2, …, C_N} containing m_1, m_2, …, m_N individuals, respectively. If the corresponding dominant allele frequencies of each population are p_1, p_2, …, p_N, the average dominant allele frequencies of the entire dataset is $\overline{p} = \frac{1}{M}\sum_{i=1}^{N} p_i m_i$ and the quantity $H_T = 2\overline{p}(1-\overline{p})$ represents the expected heterozygosity of the entire dataset. The expected heterozygosity of all populations is computed from

$$H_s^N = \frac{1}{M}\sum_{j=1}^{N} H_j(p_j)m_j \tag{3}$$

where $H_j(p_j) = 2p_j(1-p_j)$ is the local expected heterozygosity of population *j*. By definition, the F_{st} value is

$$F_{st} = \frac{H_T - H_s}{H_T} \tag{4}$$

That is, the normalized difference between the expected heterozygosities of all populations and the total dataset. Large value of F_{st} indicates that the intrinsic populations are highly dissimilar when viewed as a whole, i.e., the population is heterogeneous. Small value of F_{st} means that the population is more homogenous.

We first investigate the behavior of the fixation index value after data clustering and prove theoretically that the fixation index monotonically increases at each iteration until a homogeneous cluster is formed. We have derived the following proposition.

Proposition: Let F_{k} be the F_{st} value computed at the k^{th} iteration of the algorithm. F_{k} is non-decreasing, i.e.,

$$F_{k+1} \geq F_k$$

Proof:

At iteration *k*, the original cluster of *M* individuals is divided into $k \geq 2$ subclusters containing m_1, m_2, …, m_k individuals, respectively. Let the corresponding dominant allele frequencies within each subcluster be p_1, p_2, …, p_k. The average dominant allele frequencies of the entire dataset is $\overline{p} = \frac{1}{M}\sum_{i=1}^{k} p_i m_i$ and the quantity $H_T = 2\overline{p}(1-\overline{p})$ represents the expected normal-type allele frequency of the data.

Since $F_k = \frac{H_T - H_s^k}{H_T} = 1 - \frac{H_s^k}{H_T}$, to prove that $F_{k+1} \geq F_k$ it is equivalent to showing $H_s^{k+1} \leq H_s^k$.

$$H_s^k = \frac{1}{M}\sum_{j=1}^{k} H_j(p_j)m_j = C + \frac{1}{M}H_k(p_k)m_k$$

where

$$C \equiv \frac{1}{M}\sum_{j=1}^{k-1} H_j(p_j)m_j.$$

If the cluster at iteration *k* is considered heterogeneous, the NJ tree clustering bisects the cluster. Consequently,

$$H_s^{k+1} = C + \frac{1}{M}H_{k+1}^1(p_{k+1}^1)m_{k+1}^1 + \frac{1}{M}H_{k+1}^2(p_{k+1}^2)m_{k+1}^2$$

where

$$m_k = m_{k+1}^1 + m_{k+1}^2 \quad .$$

Trivially,

$$p_k = \frac{p_{k+1}^1 m_{k+1}^1 + p_{k+1}^2 m_{k+1}^2}{m_k}.$$

For notational convenience, hereafter we drop the subscript denoting the iteration number and shorthand p_{k+1}^1 as p^1, p_{k+1}^2 as p^2, m_{k+1}^1 and m_{k+1}^2 as m^1 and m^2, respectively.

Thus,

$$\begin{aligned} H_s^{k+1} &= C + \frac{2}{M}\left[p^1(1-p^1)m^1 + p^2(1-p^2)m^2\right] \\ &= C - \frac{2(m^1+m^2)}{M}\left[-p^1(1-p^1)\frac{m^1}{m^1+m^2}\right] - \frac{2(m^1+m^2)}{M}\left[-p^2(1-p^2)\frac{m^2}{m^1+m^2}\right] \\ &= C - \frac{2(m^1+m^2)}{M}\left[f(p^1)\lambda_1 + f(p^2)\lambda_2\right] \end{aligned}$$

using

$$f(y) = -y(1-y), \lambda_1 = \frac{m^1}{m^1+m^2}, \quad \lambda_2 = \frac{m^2}{m^1+m^2}.$$

Since $\lambda_1, \lambda_2 > 0, \lambda_1 + \lambda_2 = 1$, and $f\left(y\right)$ is a continuous concave up function, it follows from Jensen's inequality that

$$\begin{aligned} H_s^{k+1} &\le C - \frac{2(m^1+m^2)}{M}\left[f(p^1\lambda_1 + p^2\lambda_2)\right] \\ &= C - \frac{2(m^1+m^2)}{M}\left[f(p_k)\right] \\ &= C + \frac{1}{M}H_k(p_k)m_k \\ &= H_s^k \end{aligned}$$

If the cluster is already homogenous before splitting, the average dominant allele frequencies $p^1 \approx p^2 \approx p_k$, thus $H_s^{k+1} \approx H_s^k$.

From the proposition the F_{st} value of the data after each iteration increases monotonically and converges after all populations have been identified. We propose using the difference,

$$\Delta F = F_{k+1} - F_k \tag{5}$$

as a measure to detect homogeneous clusters. Using ΔF, the difference between clustered populations is now quantifiable and has a basis in genetics. If ΔF is sufficiently small we announce that the cluster is homogeneous and terminates the process. The smaller the ΔF threshold, the higher the sensitivity of the algorithm to differentiate between clusters. Thus, the threshold can be adjusted to obtain the desirable clustering resolution/sensitivity.

(3) Explore the performance of the iNJclust algorithm as a function of the ΔF stopping criterion

We explore the effects of evolution time and number of populations on the optimal value of ΔF, as well as consistency of the clustering results. We compare the iNJclust clustering results at each value of ΔF to the ground truth and compute the F-measure [1]

$$\Psi(\mathcal{S},\mathcal{C}) = \sum_{i=1}^{N} \frac{|S_i|}{M} \max_j \left(\frac{2 \cdot \dfrac{|S_i \cap C_j|^2}{|S_i||C_j|}}{\dfrac{|S_i \cap C_j|}{|S_i|} + \dfrac{|S_i \cap C_j|}{|C_j|}} \right) \tag{6}$$

where $C = \{C_1, C_2, \ldots, C_N\}$ is the clustering result from the iNJclust algorithm and S is the ground truth. The higher the F-measure value, the closer the result is to the simulated model (an F-measure value of 1 occurs when the iNJclust clustering result is exactly the same as the true clusters). From the F-measure plot we select the appropriate ΔF threshold value that gives the best clustering performance. We also perform bootstrapping to investigate the robustness of the iNJclust algorithm and examine how two factors, namely evolution time and number of populations, may affect the optimal value of the ΔF threshold.

(4) Investigation on the properties of the ΔF threshold and a criterion for selecting its value

For the iNJclust algorithm, the number of final populations and their individual assignments are dependent on the stopping point of the iterative process. If the ΔF criterion is too high, the predicted result may include a cluster with mixed populations. On the other hand, a too-low value of the criterion may split a single population into multiple clusters. Both situations give rise to clustering errors. Therefore, It is desirable to select the best possible value of the stopping criterion to minimize the clustering error. Here we investigate whether there is an optimal or near-optimal value for the ΔF terminating criterion. If so, the value should be selected systematically. Otherwise, the value can be selected heuristically from extensive sets of data.

For the clustering problem, the objectives are to select compact clusters by maximizing the intra-cluster connectivity, and to minimize the inter-cluster connectivity such that different clusters are well-separated [2]. Therefore, the most appropriate value of the ΔF threshold should produce the best clustering result possible. Different cluster validation techniques have been developed to analyze cluster structure, e.g., [3-5]. We consider two cluster validity indices based on inter- vs. intra-cluster connectivity, namely the Dunn's index and Davies Bouldin index [3]. We also look at the Silhouette index, which is an index based on node's neighborhood [2]. The indices parameters are summarized in Table 1 followed by their equations.

Table 1: Cluster validity index parameters

Index parameters	Meaning
k	Number of clusters
$d(x,y)$	Distance between individuals x and y *Note:* We use distance on the dissimilarity matrix in our computation.
C_i	i^{th} cluster
$d(C_i,C_j) = \min\limits_{x\in C_i, y\in C_j} \{d(x,y)\}$	Distance between cluster C_i and C_j
$diam(C_i) = \max\limits_{x,y\in C_i} \{d(x,y)\}$	Diameter of cluster C_i
$d(v_i,C_j)$	Distance between node v_i and cluster C_j
N_i	Number of node (individual) within cluster C_i

1) Dunn's index:

$$D = \min_{i=1:k}\left\{ \min_{j=i+1:k}\left(\frac{d(C_i,C_j)}{\max\limits_{l=1:k}(diam(C_l))} \right) \right\} \tag{7}$$

For well-separated clusters, the distances among the clusters should be large and the diameter of each cluster should be small. Hence, the larger value of the index implies better cluster structure.

2) Davies-Bouldin index:

$$DB = \frac{1}{k}\sum_{i=1}^{k} \max_{j\neq i}\left[\frac{diam(C_i)+diam(C_j)}{d(C_i,C_j)} \right] \tag{8}$$

The smaller Davies-Bouldin index, the more compact the clusters are.

3) Silhouette index:

$$s(v_i) = \frac{d(v_i, C_h) - d(v_i, C_j)}{\max(d(v_i, C_j), d(v_i, C_h))} \tag{9}$$

where C_h is the closest cluster to node v_i, which belongs to cluster C_j.

Note that $-1 \leq s(v_i) \leq 1$ and $s(v_i)$ is well-clustered when the value is near 1. The silhouette S_j for cluster C_j is given by

$$S_j = \frac{1}{N_j}\sum_{i=1}^{N_j} s(v_i) \tag{10}$$

and finally, the global silhouette index is

$$GS = \frac{1}{k}\sum_{j=1}^{k} S_j \tag{11}$$

Other possible cluster validity indices include compactness index, modularization quality, RMSSDT and RS validity indices, and SD validity index. We hope to find an index that can automatically provide a clear choice for the iNJclust stopping criterion.

III.B PCA-based informative SNP selection for analyzing population structure

When dealing with large number of SNPs, there are an intensive computational requirement of existing algorithms for population studies, and high genotyping cost. Moreover, genotyping platform errors may introduce small perturbation that could cause spurious patterns. Thus, methods that can identify a smaller set of SNPs containing information about intrinsic population structures are appealing. In particular, we are interested in an approach termed *PCA-correlated SNPs* technique [6], which infers these structure informative markers using PCA. The technique is simple and very effective. However, PCA-correlated SNPs requires estimating the rank of data matrix, and the selected set of informative SNPs varies greatly with different assumed ranks. Consequently, the inferred underlying structures are not consistent. In this part of the research, we modify the PCA-correlated method. The process diagram of the PCA-correlated SNPs technique is depicted on Fig. 2, with the modified parts highlighted in gray.

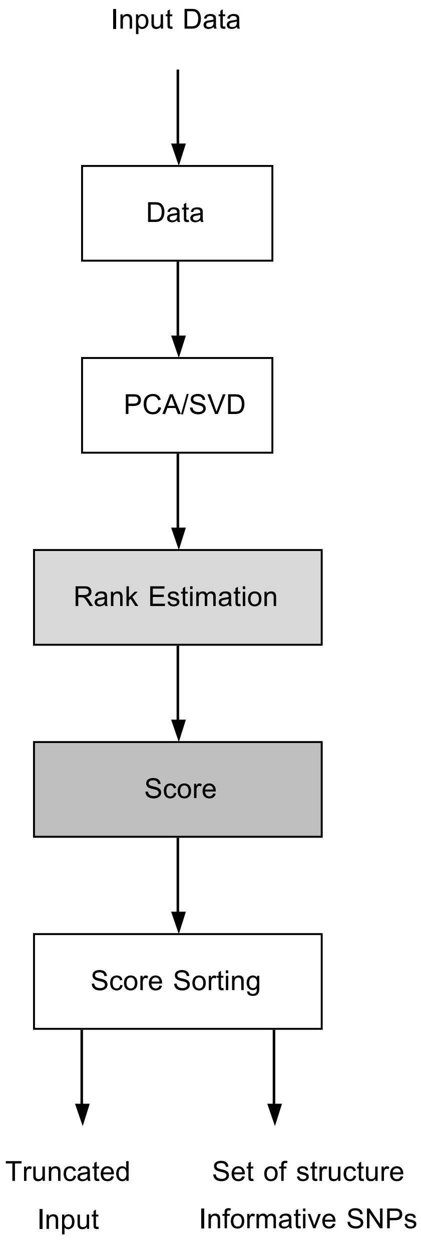

Figure 2: Process diagram for modified PCA-correlated SNPs method.

(1) Selecting structure informative SNPs

Consider the data of M individuals genotyped with L SNP markers in the form of an M x L matrix $\mathbf{X}$. The i^{th} row of $\mathbf{X}$ represents the SNP sequence of individual i. The j^{th} column of $\mathbf{X}$ gives the variation of SNP at location j across all individuals. Typically, we have $M \leq L$. The biallelic SNP representation at each locus is encoded as 0 (homozygous wild type), 1 (heterozygous), or 2 (homozygous mutant). To reveal the structure within the data using PCA, the singular value decomposition (SVD) is performed so that $\mathbf{X}$ can be written as

$$\mathbf{X} = \mathbf{U\Sigma V}^T = \sum_{i=1}^{M} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \tag{12}$$

where $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_M]$ is the matrix containing left singular vectors. The diagonal matrix Σ contains the singular values $\{\sigma_1, \sigma_2, \ldots, \sigma_M\}$ in descending order; $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_L]$ is the matrix of right singular vectors. The construction is such that $\mathbf{U}$ and $\mathbf{V}$ are unitary. Equivalently, $\mathbf{U}$ contains the principal components computed from the sample covariance matrix of $\mathbf{X}$. To observe population structure within the data, it is common that the data is projected onto the first few dominant principal components and visualized or used in subsequent clustering technique of choice.

To gauge whether a particular SNP greatly contributes in shaping the underlying population substructure using the PCA framework, Paschou et al. [6] has suggested that we look at the j^{th} column of $\mathbf{X}$ corresponding to values of the j^{th} SNP across individuals, defined as

$$\mathbf{a}_j = \sum_{i=1}^{M} \sigma_i \mathbf{u}_i v_i^j, \tag{13}$$

where v_i^j is the j^{th} element of $\mathbf{v}_i$.The so-called PCA-correlated SNPs method for identifying a smaller set of SNPs computes the score for SNP j

$$p_j = \sum_{i=1}^{R} (v_i^j)^2, \quad j = 1, \ldots, L \tag{14}$$

and selects the desired number of SNPs with the largest p_j values. The resulting SNP locations are presumably the most informative. In terms of a basis function expansion, PCA-correlated SNPs approximates the column vector $\mathbf{a}_j$ using R bases $\{\sigma_i \mathbf{u}_i,\ i = 1, \ldots, R\}$ and the v_i^j 's are the basis expansion coefficients. The parameter R is the rank of matrix $\mathbf{X}$, i.e., the number of significant principal components. It is observed that the norms of the basis vectors $\{\sigma_i \mathbf{u}_i\}$ usually vary greatly, depending upon the singular value distribution of the data. Consequently, the coefficients v_i^j whose corresponding singular values σ_i are very small do not give significant contributions to $\mathbf{a}_j$. Nevertheless, they have been given equal importance for the score computation. There is also a rank parameter R to be estimated. The error of the selected rank R has an effect on the final selection of SNPs that are deemed informative.

This work presents an improvement on computing an informativeness score of each SNP. Starting with the representation in Eq. (13), we select the left singular vectors $\{\mathbf{u}_i\}$ as our bases. Hence, the basis expansion coefficients are $\{\sigma_i v_i^j$, i = 1, …, R}, which are a function of both the singular values and the elements of the right singular vectors. The updated score is now computed as

$$\tilde{p}_j = \sum_{i=1}^{R} (\sigma_i v_i^j)^2, \quad j = 1, \ldots, L. \tag{15}$$

Notice that the bases {$\mathbf{u}_i$} are orthonormal. This is a nice property in the case where the basis expansion coefficients are unknown and need to be estimated. The singular values appropriately weight the contribution of v_i^j in column j in the same manner as the right singular vectors $\mathbf{v}_j$'s have been weighted for constructing the original data matrix $\mathbf{X}$.

(2) Data representation accuracy

It is desirable that, even with much fewer SNPs, the new data matrix retains the underlying population structure. To investigate the results using k principal components, we denote the matrix of k left singular vectors from the original data matrix corresponding to the k largest singular values as $\mathbf{U}_k = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k]$. A new M x P data matrix $\tilde{\mathbf{X}}$ with reduced dimension is formed by keeping only P columns of $\mathbf{X}$ corresponding to P largest $\tilde{p}_j$ values. The principal components of the new data matrix are computed from

$$\tilde{\mathbf{X}} = \tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^T \tag{16}$$

Similarly, we define $\tilde{\mathbf{U}}_k$ as the left singular matrix $\tilde{\mathbf{U}}$ with only the first k columns. Using the same number of significant principal components, the structure representation accuracy is defined as

$$\gamma(k) = \frac{\mathrm{trace}\{\mathbf{U}_k^T \tilde{\mathbf{U}}_k \tilde{\mathbf{U}}_k^T \mathbf{U}_k\}}{\mathrm{trace}\{\mathbf{U}_k^T \mathbf{U}_k\}} \tag{17}$$

This measures the fraction of signal energy captured by the first k principal components of the original data matrix that can be represented using the k dominant principal components of the reduced data matrix. Ideally, we would like $\gamma(k)$ to be as close to 1 as possible.

The comparisons between our method and the PCA-correlated method is summarized in Table 2.

Table 2: comparisons between our method and the PCA-correlated method

	Our Method	PCA-correlated SNPs
Informativeness Score	based on singular vectors weighted by singular values	based on singular vectors only
Accurate rank estimation	NOT essential	Crucial
Basis functions	Orthonormal	Orthogonal
Performance assessment	Data representation accuracy	Clustering accuracy

IV. Results and Discussions

IV.A Test Data

We design and generate two simulated data sets using GENOME simulator. The data sets are used to explore the effects of different genetic parameters on the optimal value of the stopping criterion, as well as the consistency of the clustering results for the iNJclust algorithm. The first simulated data set contains 20 clusters of 60 individuals each (for a total of 1,200 individuals), and 10,000 SNPs per individual. The second data set contains 1,200 individuals separated into 10 clusters of varying sizes ranging from 60 to 330 individuals per cluster, also genotyped at 10,000 SNPs. We use the following parameters:

Data set 1:

```
-pop 20 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60
60 60 60 60    -c 20 -s 500 -N UD1model.txt
```

Data set 2:

```
-pop 10 330 150 60 60 60 300 60 60 60 60 -c 20 -s 500 -N
UD2model.txt
```

The tree files *UD1model.txt* and *UD2model.txt* for generating data sets 1 and 2 are represented graphically in Fig. 3 and 4, respectively. The branch lengths represent the evolution time of populations in terms of the number of generations they evolve.

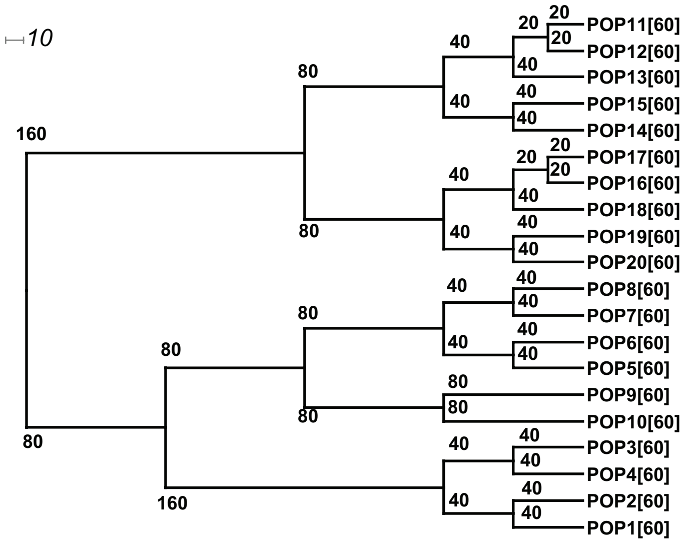

Figure 3: Simulated population history tree for simulated data set 1

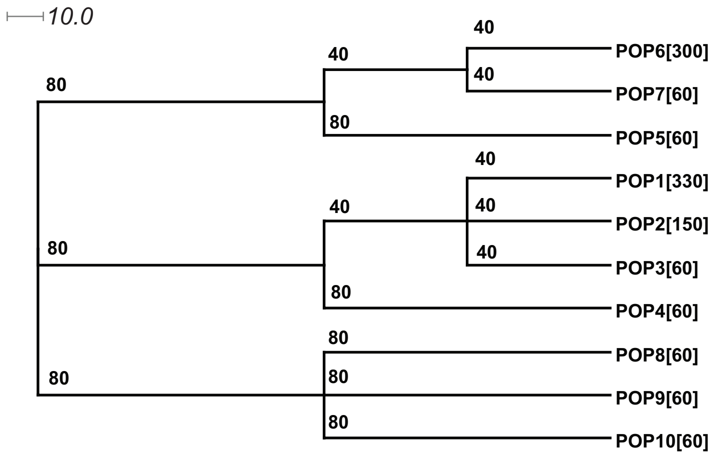

Figure 4: Simulated population history tree for Data set 2

For real data used to assess the performance of our algorithms, we obtain two large animal data sets with different complexities for our preliminary investigation. The first data set is a 28-breed sheep data set [7], which contains 392 individuals and 1,046 SNPs. The second data set is from 47 breeds of 1,089 bovines [8], genotyped at 44,706 SNPs. We also acquire the third data set comprises of 27 human populations spanning Europe, East Asia, India, and Africa [9] for a total of 554 individuals and 243,855 SNPs. The last data set is the 13 hilltribes data set from the PanAsian SNP Initiative genotyped using the 50K Affymetrix SNP array.

IV.B Performances of the iNJclust algorithm

(1) Performance of the iNJclust algorithm on simulated datasets using ΔF stopping criterion

We use the two simulated datasets to explore the effects of evolution time and number of populations on the optimal value of ΔF, as well as consistency of the clustering results. We compare the iNJclust clustering results at each value of ΔF to the ground truth and compute the F-measure. From the F-measure plot we select the appropriate ΔF threshold value that gives the best clustering performance.

The optimal values of ΔF stopping threshold for clustering simulated dataset 1 and 2 are determined by scanning the iNJclust algorithm over possible ΔF threshold values ranging from 10^{-5} to 10^{-1}, i.e., the difference of 10% to 0.001% in the fixation indices of successive iterations.

Figure 5 depicts the F-measure value as a function of ΔF threshold for the two simulated datasets. If the ΔF threshold is too high, the iNJclust process undersplits the clusters. Contrastly, a too-low value of the threshold oversplits the cluster. Both situations results in the decreases of the F-measure values. We also observe a step-like behavior of the F-measure values, which indicates that the optimal threshold for the ΔF stopping criterion is not a single point but rather a range, making selecting an appropriate value for the threshold slightly flexible. From the graph, we select the threshold of 0.001 for simulated dataset 1 and 0.003 for simulated dataset 2 to stop the iNJclust iterations.

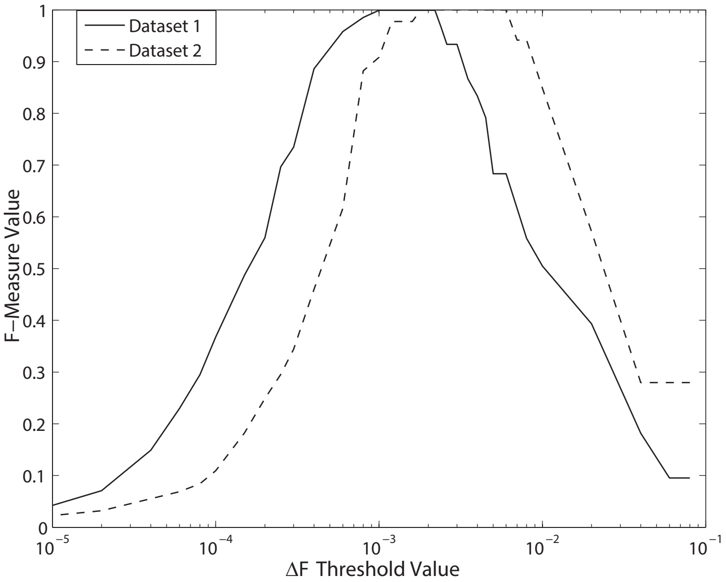

Figure 5: F-Measure values of simulated data sets as a function of the *ΔF* thresholds.

We also employ the bootstrapping method to investigate clustering consistency of the iNJclust algorithm. To perform bootstrapping, simulated datasets 1 and 2 are each resampled with replacement to obtain 100 bootstrap datasets with 400 individuals. Each bootstrap dataset is then clustered by the iNJclust algorithm. The corresponding results are given below.

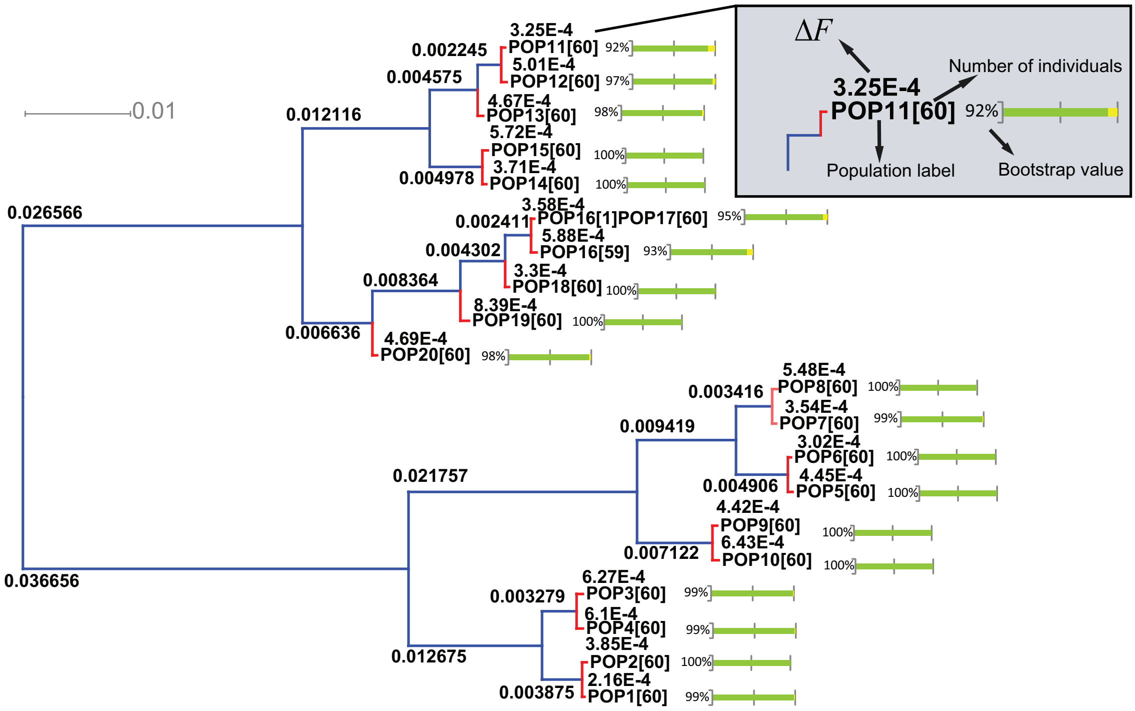

Figure 6: Hierarchical population tree of simulated dataset 1 from iNJclust (*ΔF* is set at 0.001).

The iNJclust results are depicted in the form of hierarchical population trees inferred from the full datasets in Fig. 6 and 7. The branch lengths on the trees correspond to the computed ΔF values at each iteration. Observe that ΔF monotonically decreases as the iteration progresses. The terminal nodes of the tree contain iNJclust individual assignments, where the individuals are labeled by their true cluster number. The numbers in the square brackets represent the number of individuals within each cluster. A careful examination of the individual assignment results of the full datasets confirms that the iNJclust algorithm is able to correctly assign most individuals to their respective clusters. For simulated dataset 1, one individual from population POP16 is grouped with POP17. This is possible since the pair of populations POP16 and POP17 only differs by 20 generations; they are closely related populations in the dataset. The total individual assignment is 99.92% correct. In simulated dataset 2, we also vary the number of individuals in each population to investigate the ability of the iNJclust algorithm to handle varying cluster sizes. The result in Fig. 7 shows that the clustering performance remains excellent in this situation with no individual assignment error.

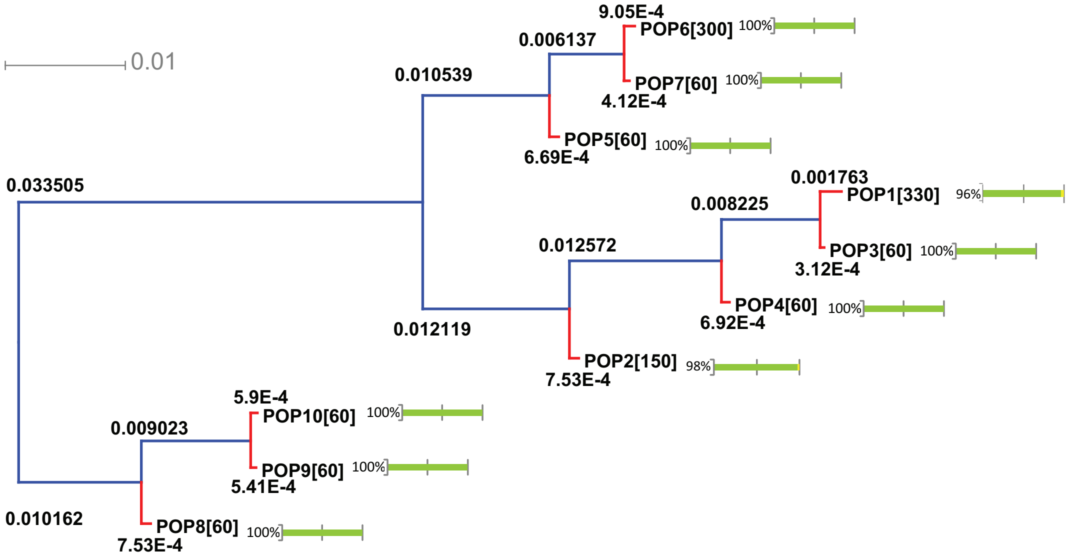

Figure 7: Hierarchical population tree of simulated dataset 2 from iNJclust (ΔF is set at 0.003).

The bootstrap percentage is represented as a green slider bar at the end of each terminal node in Fig. 6 and 7. These bootstrap values correspond to the optimal performance of the algorithm, because the ΔF threshold has been selected at the point where the F-measure equals to 1, i.e., the optimal stopping point. It is discovered that most of the terminal nodes have bootstrap percentages of nearly 100%, validating the consistency of the iNJclust's clustering ability. The drop in bootstrap percentage at some terminal nodes happens when individuals from one or more clusters are not resampled.

We observe that the topology of the inferred tree may be changed slightly by swapping small branches during the bifurcation. For example, in simulated dataset 1 the POP19 branching is in a different order from the simulated model. However, this does not greatly affect the overall tree topology, as the trend of the inferred tree structure is consistent with the underlying model. For simulated dataset 2, the relationships among populations in the iNJclust tree output also largely follow the structure of the simulated model, even though the simulated tree is not binary.

(5) Effects of evolution time and number of populations on the optimal value of ΔF

We investigate how two factors, namely evolution time and number of populations, may affect the optimal value of the ΔF threshold. To eliminate other possible parameters that may influence the threshold value, e.g. tree structure, we simulate a one-layer tree with varying number of populations and generations. The evolution time is manifested in the number of generations in the simulation, i.e., the longer the evolutions, the further apart the populations are. Hence, we vary the genetic distance between populations by simulating data ranges between 40 to 500 generations (corresponding to approximately 80 to 10,000 years of genetic evolution). We also vary the number of populations in the data from 2 to 64 populations.

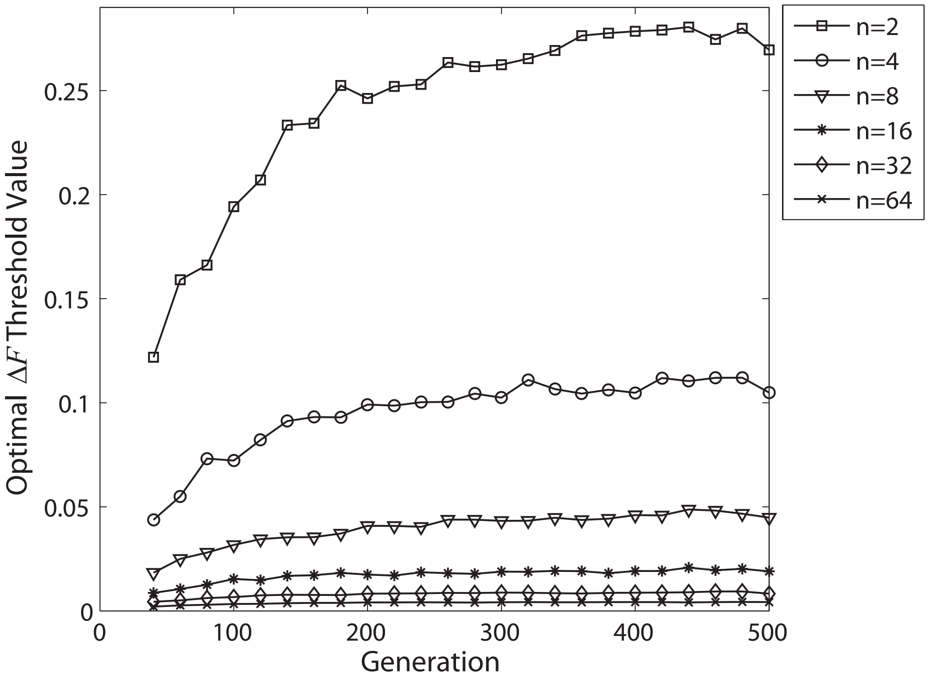

Figure 8: Relationship between the optimal ΔF threshold value and generations/number of populations.

From the result depicted in Fig. 8, data with larger number of generations are further apart genetically, so the threshold increases with generations as expected. On the other hand, when the number of populations (n) increases the threshold value decreases, since more sensitivity is needed to differentiate between populations. We observe that the ΔF threshold is also fairly robust with the number of generations (representing the evolution time of the population) when the structure is sufficiently complex.

(6) Clustering performance of the iNJclust algorithm on real datasets

Three real data sets are used to investigate the performance of our algorithms—the 27 human populations, the 28-breed sheep, and the 47-breed bovine data sets. Since there is no ground truth on the underlying populations within the data, we compare the individual assignments to data labels. We also compare clusters given by our algorithm to the ancestry ratios produced by the Admixture algorithm.

To test the iNJclust algorithm on these real datasets, The ΔF threshold of 0.001 has been chosen as a stopping criterion. We compare the clustering performance of iNJclust with two similar algorithms in terms of the F-measure value. First is the AWclust algorithm [10], which also uses ASD matrix for inputs but employs hierarchical clustering for individual assignments. We also compare iNJclust with the so-called *NJclust* algorithm, which is basically the iNJclust algorithm without the successive NJ tree rebuilding step. Since there is no ground truth on the underlying populations within the data, we compare the individual assignments to data labels. The result is summarized in Table 3.

Table 3: Comparison between iNJclust, NJclust, and AWclust F-measure values and their estimated number of populations on real datasets

	AWclust		iNJclust		NJclust	
	F-measure	No. of pop.	F-measure	No. of pop.	F-measure	No. of pop.
Sheep 28 breeds	0.76	16	0.92	30	0.87	30
Bovine 47 breeds	0.10	2	0.92	39	0.80	33
Human 27 populations	N/A	N/A	0.80	22	0.68	16

The F-measures for the three algorithms—iNJclust, NJclust, and AWclust, and the estimated numbers of populations, are reported in Table 3. The result confirms that the iNJclust algorithm produces the best clustering results. The estimated number of populations are 30 for the 28-breed sheep dataset and 39 for 47-breed bovine dataset. These estimated numbers of populations are reasonable. The estimated number of populations for the bovine data is low because the dataset is more complex and contains many breeds. Some breeds, e.g., three *B.*

indicus breeds (GIR, NEL, BRM) are very similar and are lumped into one cluster. The estimated numbers of cluster from the AWclust algorithm are far from the truth due to its computational limitation. The individual assignments of the AWclust algorithm are also worse than the assignments of iNJclust. We believe that the NJ tree based clustering proposed in our algorithm is more appropriate in distinguishing between populations than hierarchical clustering for genetic data. The NJclust clustering result is more erroneous than iNJclust, illustrating the necessity of reconstructing the NJ tree at each iteration. For Human 27 populations data, it is worthy to note that the AWclust algorithm cannot be completed in a reasonable amount of time due to large data dimension and complexity, hence its F-measure value is not reported here. The F-measure value of iNJclust is 0.8. We think this low value may in part be the consequence of the wrong self-reported labels.

(7) Comparison between iNJclust and Admixture results on real datasets

Since self-reported labels may not always correspond to the individual's intrinsic genetic pattern, we also compare the iNJclust results with the results from the Admixture method [11] as a way to alternatively verify the clustering results on the real datasets. The Admixture algorithm estimates the ancestry contributions within each dataset.

In Fig. 9-11 we corroborate the iNJclust clustering results with the admixture patterns by looking at the admixture pattern for each of the cluster assigned by iNJclust. Each panel of admixture patterns separated by the black lines is one cluster assigned by the iNJclust algorithm. The corresponding self-reported labels of individuals are displayed below the panels, with the number of individuals from that label shown in the bracket. Overall, the admixture results are in very good agreements with the iNJclust individual assignments. That is, each assigned iNJclust cluster has a distinct admixture pattern. Because 28-breed sheep dataset contains much fewer SNPs per individuals than the SNP profiles for 47-breed bovine or Human 27 populations, some admixture ratios in Fig. 9 on the right-hand side are visually less distinguishable from one another, for example {CHA(14)} and {COM(16)DOS(4)}. Nevertheless, the iNJclust algorithm is able to correctly separate these clusters. For 47-breed bovine dataset, the iNJclust cluster with mixed populations {GIR, NEL, BRM, and OBB} corresponds to non-uniform admixture patterns, whereas other homogeneous populations correspond to uniform and distinct admixture patterns.

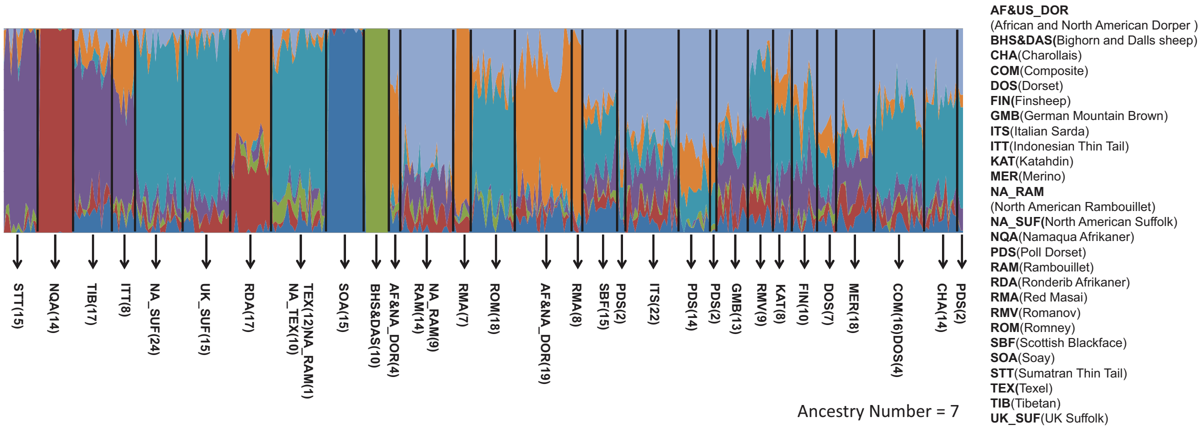

Figure 9: Admixture results of the 28-breed sheep dataset (the ancestry number is 7).

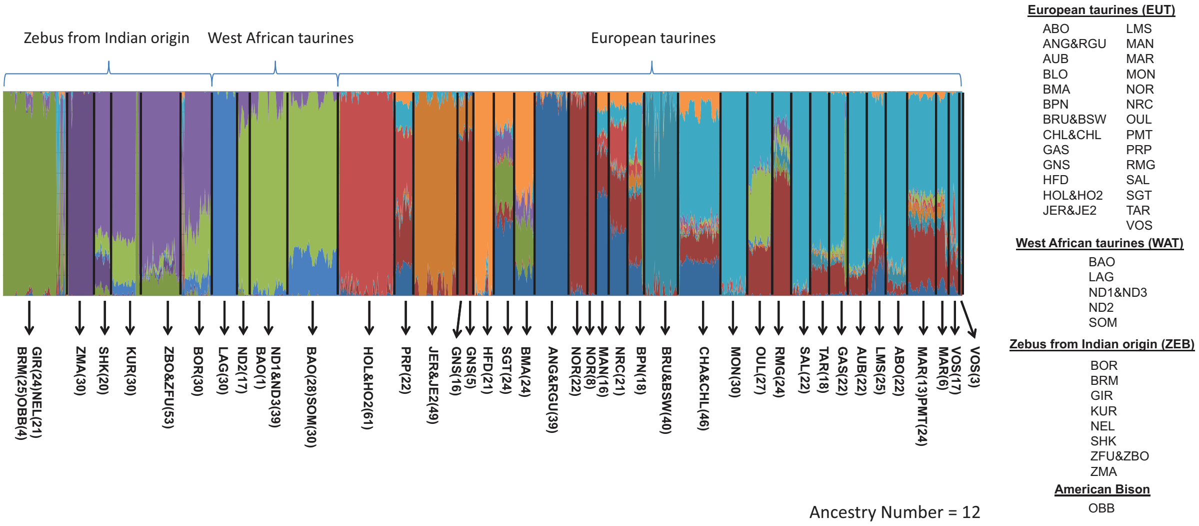

Figure 10: Admixture results of the 47-breed bovine data set (the ancestry number is 12).

As expected for Human 27 populations dataset, some given labels differ from their genetic patterns. For example, {UEP, CEU}, {STK, URK}, and {KHM, CHN, CHB, JPT, VNM} estimated clusters contain individuals with mixed labels. However, they all have similar admixture patterns. Blind to the labels, the iNJclust algorithm is able to correctly cluster them into the same cluster. In contrast, individuals from the KNG and HMA populations, though carrying the same labels, are assigned to different clusters by the iNJclust algorithm. The admixture patterns confirm their genetic differences. Using these labels to calculate the F-measure results in the lower F-measure values, which does not necessarily reflect the real clustering efficacy of the iNJclust algorithm in human dataset.

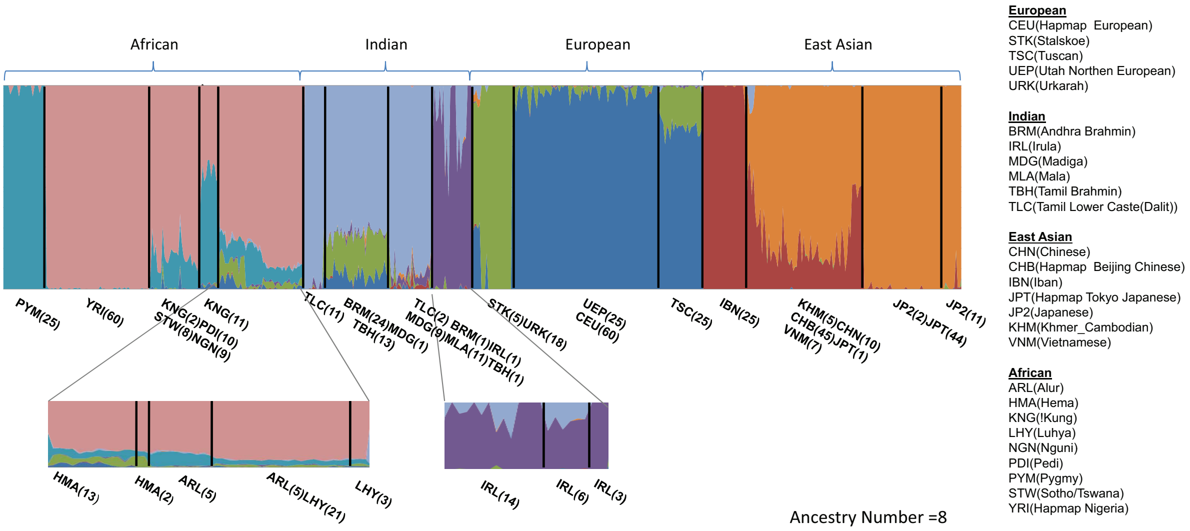

Figure 11: Admixture results of the 27-population human dataset (the ancestry number is 8).

(8) Inferred population trees of the real datasets

Another output that can be used to infer genetic similarities between populations is the population tree output. We try to construct the population tree from the order in which each cluster is bifurcated in the iterative clustering process.

The population trees generated by the iNJclust algorithm for the sheep, bovine, and human datasets are depicted in Fig. 12-14, respectively. The individual labels for each estimated population are reported with the number of individuals from each labels in the square brackets. We observe an interesting phenomenon in the structure of the inferred trees. Population that is most distinct genetically, or has the largest number of individuals, tends to be first identified. For example, for the Bovine dataset in Fig. 13 the European taurines are first separated from the West African taurines and the Zebus from Indian origin. At the second step the cluster containing *B. indicus* breeds is removed in the lower branch of the tree. At later iteration the West African taurines are broken away from the Zebus. Similarly, in Fig. 14 African individuals are separated at the first iteration, possibly because their genetic profiles are the most distinctive. Then, the East Asians populations are recognized. The Europeans and Indians, who have common ancentry as illustrated in their admixture patterns in Fig. 11, are divided at later iterations. The order at which each population are bisected in the inferred tree is very much agreeable with their corresponding admixture patterns. We believe that the resolved tree may partially reflect the actual history of population diversity.

We notice that even though the orders at which each population is bisected in the inferred tree look as if they follow the actual history of population diversity, the history is not observable using only a single snapshot of population variations. Hence, the resulting iNJclust population tree can only reflect the underlying relationships among populations. Similar populations tend to be clustered together or branched off from the same parental node. Also, the branch lengths of our tree are equal. They do not reflect the actual genetic distances. Therefore, this inferred population tree is still limited in illustrating the evolutionary relationships among the populations and warrant further investigation.

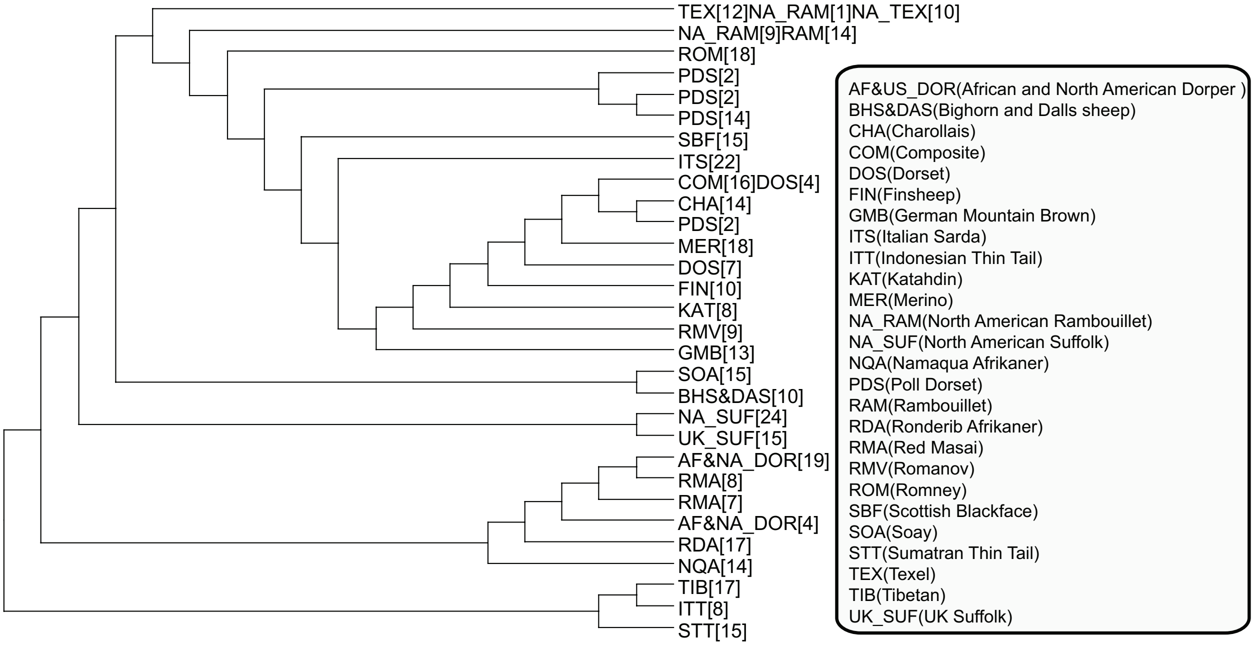

Figure 12: Inferred population tree of the 28-breed sheep dataset.

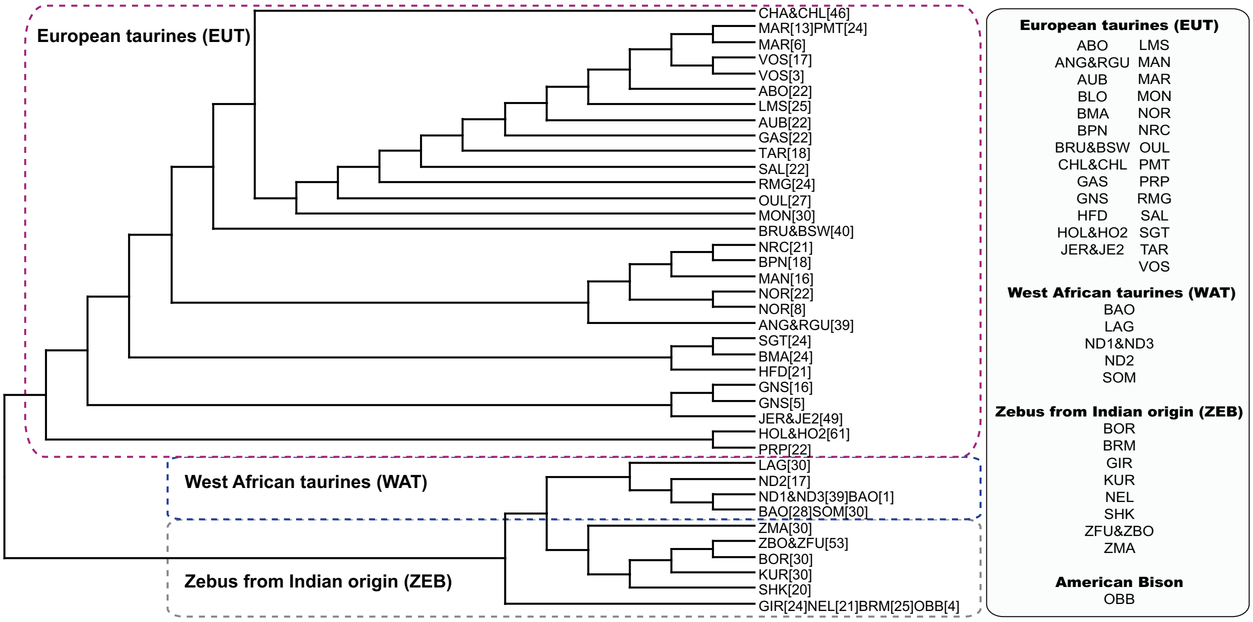

Figure 13: Inferred population tree of the 47-breed bovine dataset.

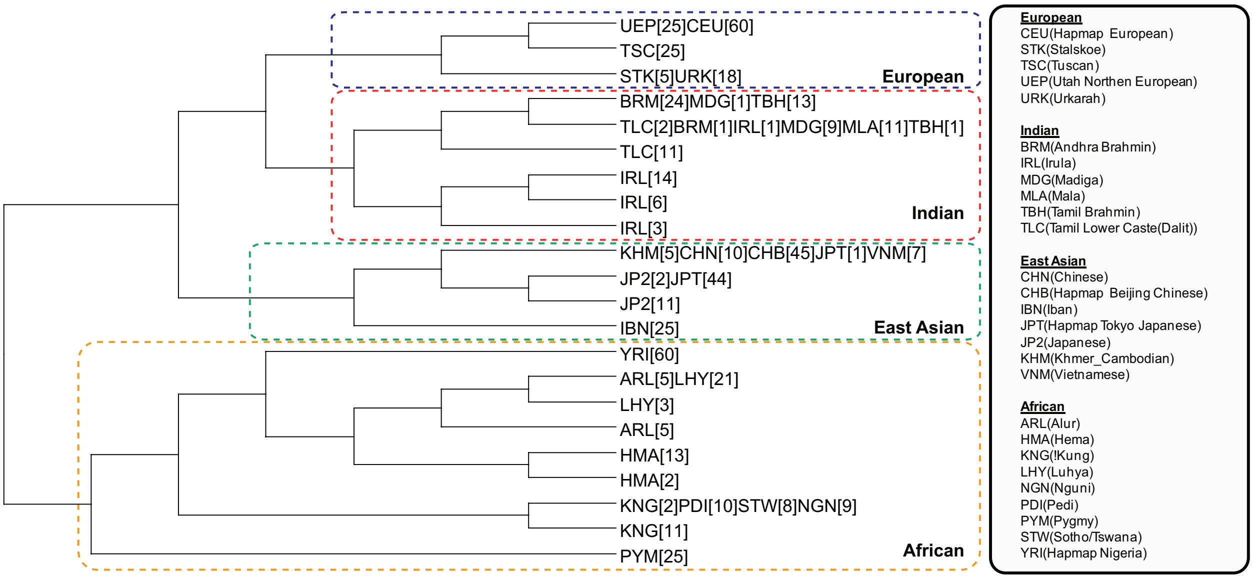

Figure 14: Inferred population tree of the Human 27 populations dataset.

IV.C Investigation on cluster validity indices as the ΔF selection criterion

We compute the three indices, namely Dunn's index, Davies Bouldin index and the Silhouette index for different values of the ΔF stopping criterion using the two simulated datasets and four real datasets—the 27 human populations dataset, sheep 28 breeds dataset, bovine 47 breeds dataset, and the 13 hilltribes dataset from the PanAsian SNP Initiative. We also compare the indices with the F-Measure values. The F-Measure is particularly useful for simulated datasets, because it provides the reference point at which the clustering error is zero. However, for real datasets the F-measures are calculated using the self-reported labels, so the values are typically lower than the truth.

The shaded areas in Fig. 15 and 16 correspond to the ranges of threshold values that give zero clustering error (F-measure = 1). Unfortunately, it is observed that there is no evident trend of any cluster validity indices following the clustering accuracy measured from the F-measure values. Similarly, results for all real datasets in Fig. 17-20 suggest that there is no obvious "optimal" point of the cluster validity indices that can be used as a guide for selecting the stopping criterion.

From our extensive testing on data sets with varying sizes and complexities, the threshold of 0.001 is suitable and is chosen as the default threshold value for data having more than 20 populations. We suggest using a threshold of 0.002-0.003 for data containing 10-20 populations, and a threshold of 0.01 for data with fewer than five populations. Note that we provide these values only as a rough guideline. There is some flexibility in selecting the value of the ΔF threshold, since there is not an optimal point of the threshold, but rather an optimal range for a particular population structure. Note that the threshold does not influence the structure of the inferred population tree at early iterations. It only determines the amount of branching at later iterations of the process. Thus, the threshold can be adjusted to obtain the desirable clustering resolution/sensitivity.

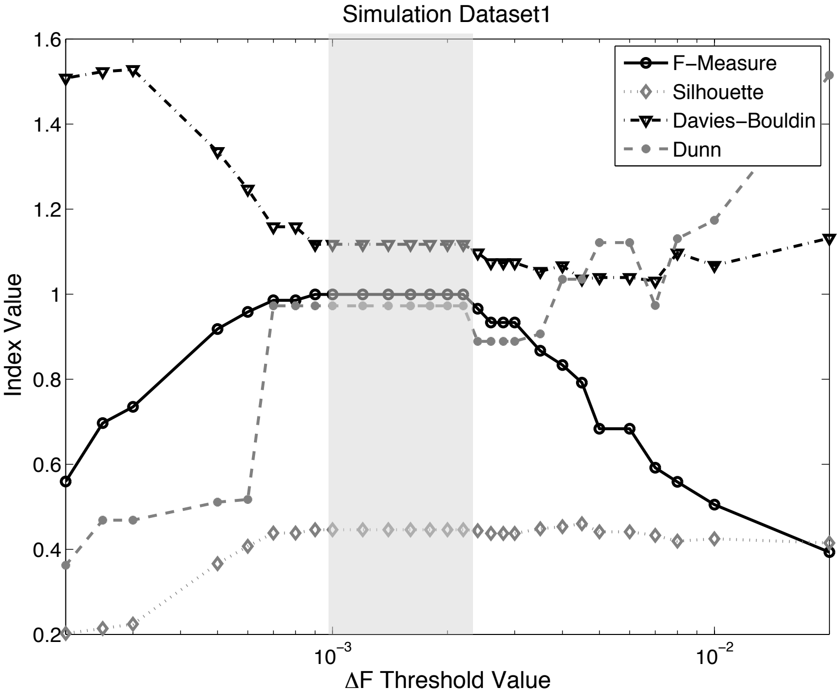

Figure 15: Cluster validity indices as a function of the *ΔF* stopping criterion for simulated dataset 1.

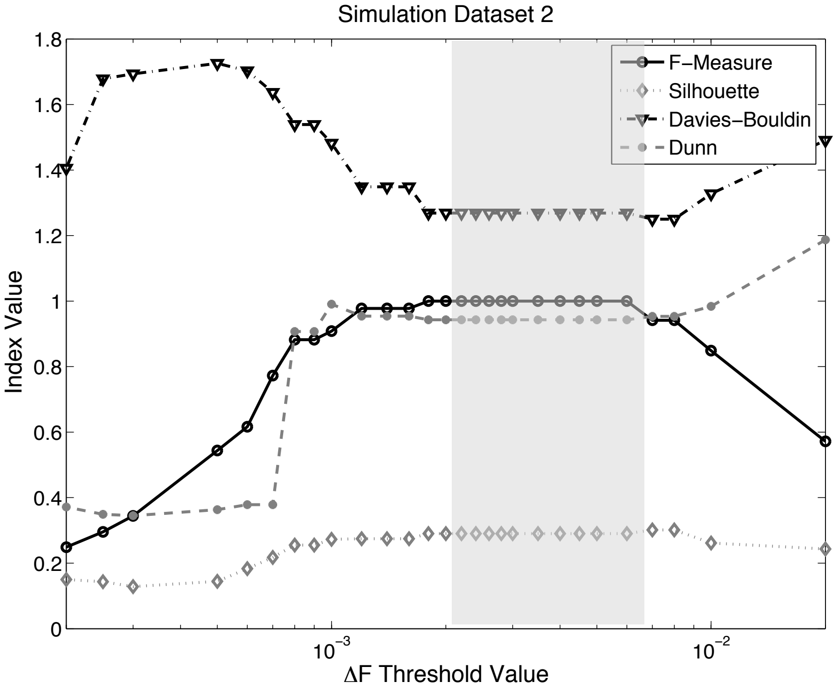

Figure 16: Cluster validity indices as a function of the *ΔF* stopping criterion for simulated dataset 2.

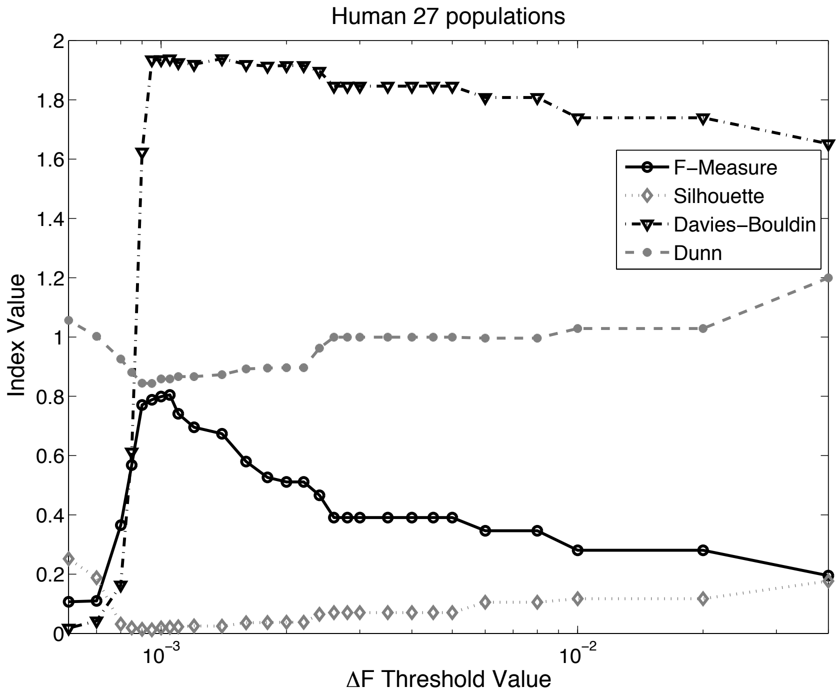

Figure 17: Cluster validity indices as a function of the *ΔF* stopping criterion for 27 Human populations.

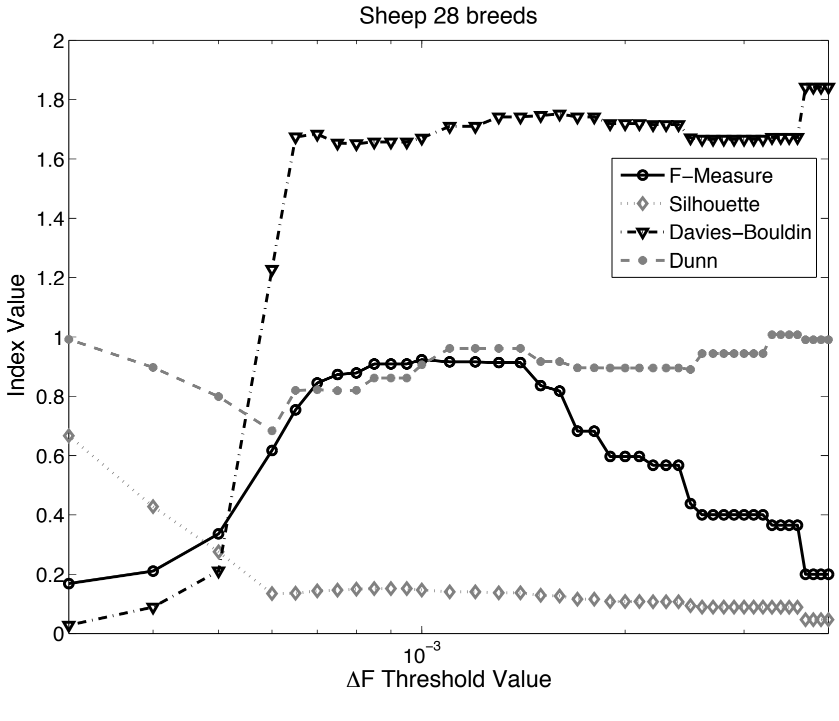

Figure 18: Cluster validity indices as a function of the *ΔF* stopping criterion for sheep 28 breeds.

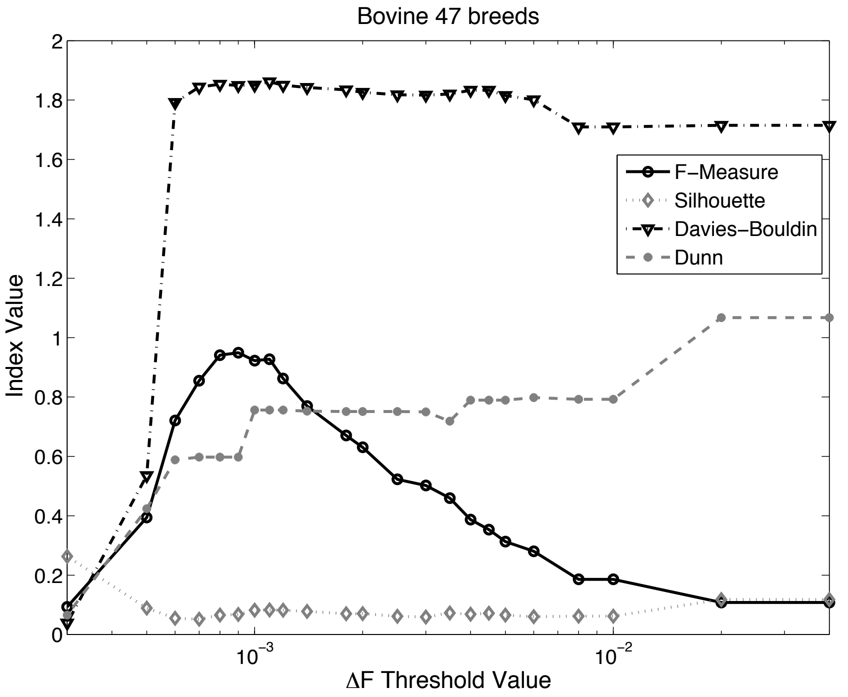

Figure 19: Cluster validity indices as a function of the *ΔF* stopping criterion for bovine 47 breeds.

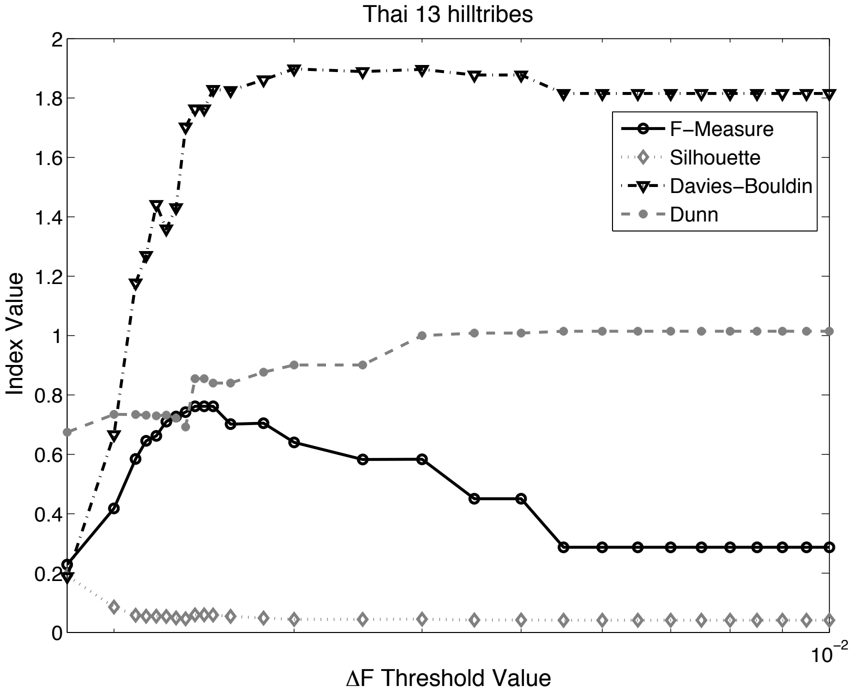

Figure 20: Cluster validity indices as a function of the *ΔF* stopping criterion for 13 Thai hilltribes.

IV.D Testing the PCA-based SNP selection on real data

The experiment is conducted using a subset of the 47-breed bovine data set. It is comprised of 230 individuals from 9 breeds of cattle genotyped at 8781 SNPs. This smaller data set is chosen because the structure in bovine data set is much more evident than the human data set, which usually have complex structures. We also would like to minimize the effects of noise or obscurity in data structure when assessing the performance of our technique. However, note that data sets with smaller numbers of SNPs are more challenging to analyze since there are less information.

The population data is represented by an *M* x *L* matrix, where *M* = 230 and *L* = 8781. In order to eliminate the effect of genetic drift and amplify structures within the data, we normalize it so that each column is zero-mean with unit variance. Effectively, the full rank of the data matrix equals *M-1* = 229.

In order to visualize the population structure within the data, *k*=3 is used. The PCA analysis for population structure of the original data matrix (full 8781 SNPs) using three principal components is shown in Fig. 21 where each breed is color-coded. The population structure within the data is obvious, with individuals from five out of nine breeds formed nicely separated clusters. Individuals from the other four breeds are conglomerated in the middle.

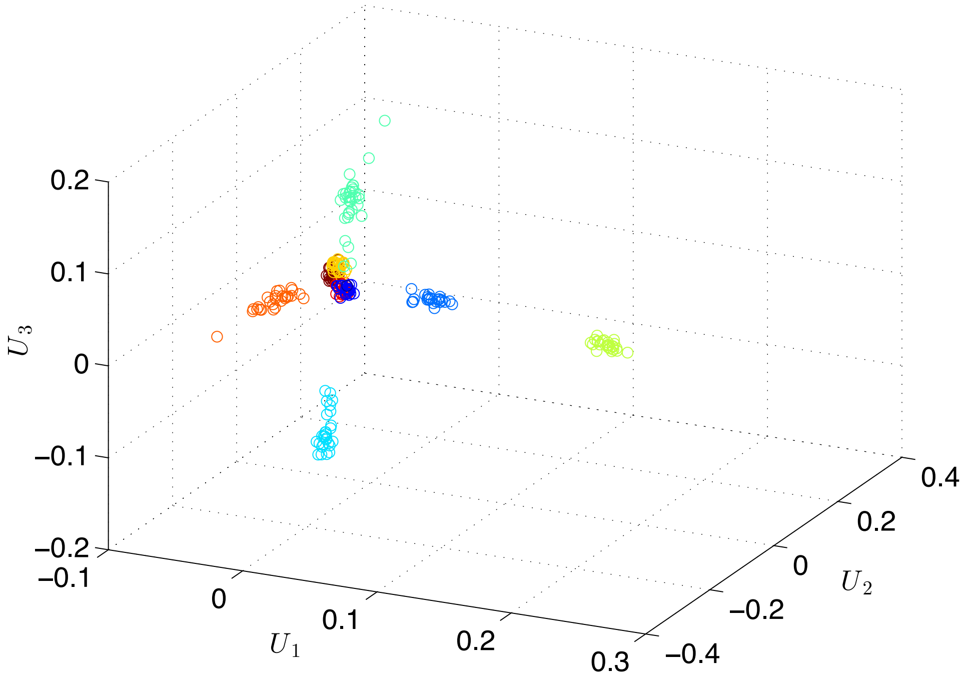

Figure 21. Structure within the bovine data set using full set of SNPs (*L* = 8781) and three dominant principal components.

(1) Robustness of informative SNPs selection

Since both the PCA-correlated SNPs scores in [6] and the informativeness scores of our proposed method are computed from R < M basis vectors, we investigate the effect of rank selection in the score computations on the selection of structure informative SNPs. To do this, the scores in Eq. (14) and (15) are computed with rank R varies from 1 (using only the first dominant principal component) to the full rank of *M-1* (using all principal components). For each R, two sets of informative SNPs are selected from the largest p_j and $\tilde{p}_j$ scores, respectively. The percent overlaps between the selected SNP loci for successive values of R are computed, as shown in Fig. 22. The numbers of selected SNPs are chosen to be 200, 500, 1000, 5000 SNPs. These are equivalent to 2.28%, 5.69%, 11.39%, and 56.94% of all available SNPs, respectively. Usually, the rank of the data is close to the number of subpopulation within the data. Therefore, the percent overlaps for R = 1-15 when 200 SNPs are kept (equivalent to about 2.28% of all available SNPs) is also reported in Table 4.

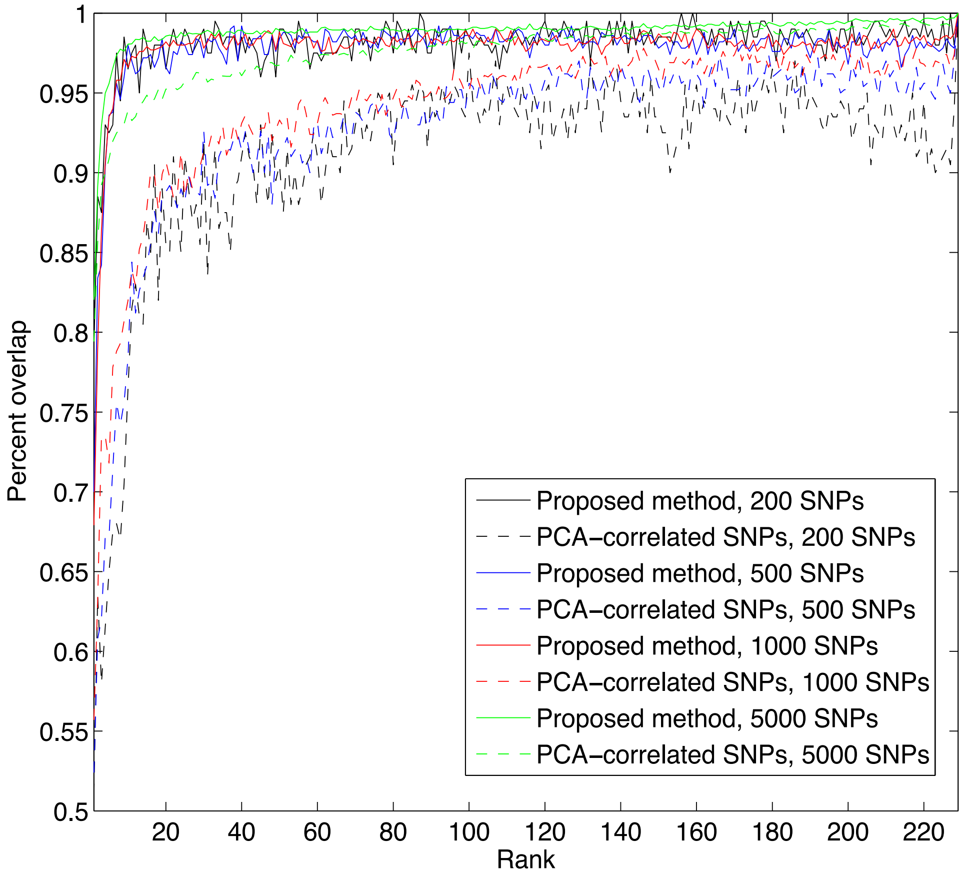

Figure 22. Percent overlap of selected SNP markers between successive ranks for R = 1-229. The numbers of selected SNPS are 200, 500, 1000, and 5000.

Table 4: Percent overlap of 200 selected SNPs between successive ranks for *R* = 1-15.

Estimated Rank (*R*)	Percent overlap of selected SNP markers (%)	
	Our method	PCA-correlated SNPs method
1	79.0	51.0
2	88.5	63.5
3	87.5	58.0
4	93.0	61.5
5	92.5	65.0
6	93.0	67.5
7	97.5	68.0
8	94.5	67.0
9	98.5	70.5
10	96.5	76.5
11	97.5	81.5
12	98.0	83.0
13	95.0	81.0
14	99.0	80.5
15	98.0	87.0

For 200 selected SNPs, the percent overlaps for the PCA-correlated SNPs method are lower than our method. Particularly for small values of *R (R* < 9*)*, the overlaps are between 50-70%. This implies that if there were an error in estimating the rank of data matrix, even if we are off by one rank, it would give a resulting set of SNPs that are vastly different. In contrast, our method is fairly lenient to the chosen value of *R*. It is seen that more than 93% of markers are similar after *R* = 3, and the similarities are on average at around 98% with *R* > 5. Hence, the proposed method for selecting informative SNPs is very robust to the assumed rank of the data matrix. We may use the full rank *R*=*M-1* (or *R*=*M* without the normalization) to compute $\tilde{p}_j$ and eliminate the need for rank estimation completely. Otherwise, a low rank-*R* approximation with *R* << *M* can be used with negligible difference. This trend replicates with larger sets of SNPs. An exception occurs when we keep 5000 SNPs, or around 57% of the total number of SNPs. The PCA-correlated SNPs method is almost as robust as our proposed method. However, the data dimensional reduction is not very high in this case.

(2) Structure representation accuracy

We compare the structure representation accuracy of our technique with the PCA-correlated SNPs method for low-rank and full-rank basis expansions. In order to estimate *R*, the singular values of the bovine data is plotted in Fig. 23 in descending order

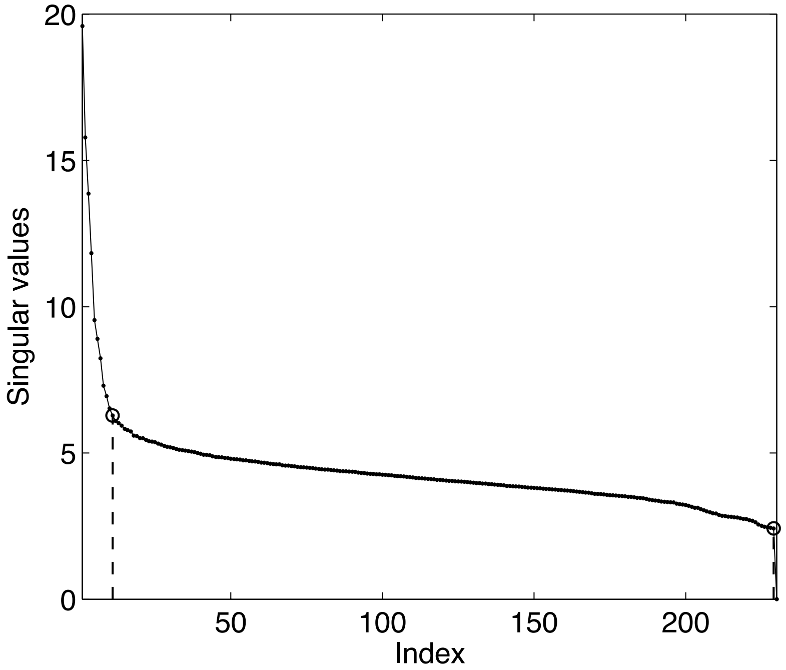

Figure 23. Singular values of the bovine data set.

The first principal component explains 7.44% of the variance within the data. We notice that there is a sharp drop of the singular values before tapering off. We identify the bend of the graph, which signifies the changing point in the singular value contributions (depicted in Fig. 23 by the dotted line), using gradients of the singular values. This corresponds to the point where the gradient is less than 5%, which occurs at the 11th singular value. So we choose *R* = 11 for the low-rank basis expansion of the data matrix in our subsequent analysis. These eleven dominant principal components account for 26.9% of the variance within the data. Each of the remaining principal components contributes only 0.34% on average. For the full-rank counterpart, we use *R* = *M-1*.

Although we have not tried to estimate the rank of the data matrix with the technique used in [6], we observe that the selected ranks therein always equal or are close to the number of the underlying populations within the data. We anticipate that for our bovine data set, the rank estimated by the original PCA-correlated SNPs method would be close to 9, so using *R*=11 is not unreasonable.

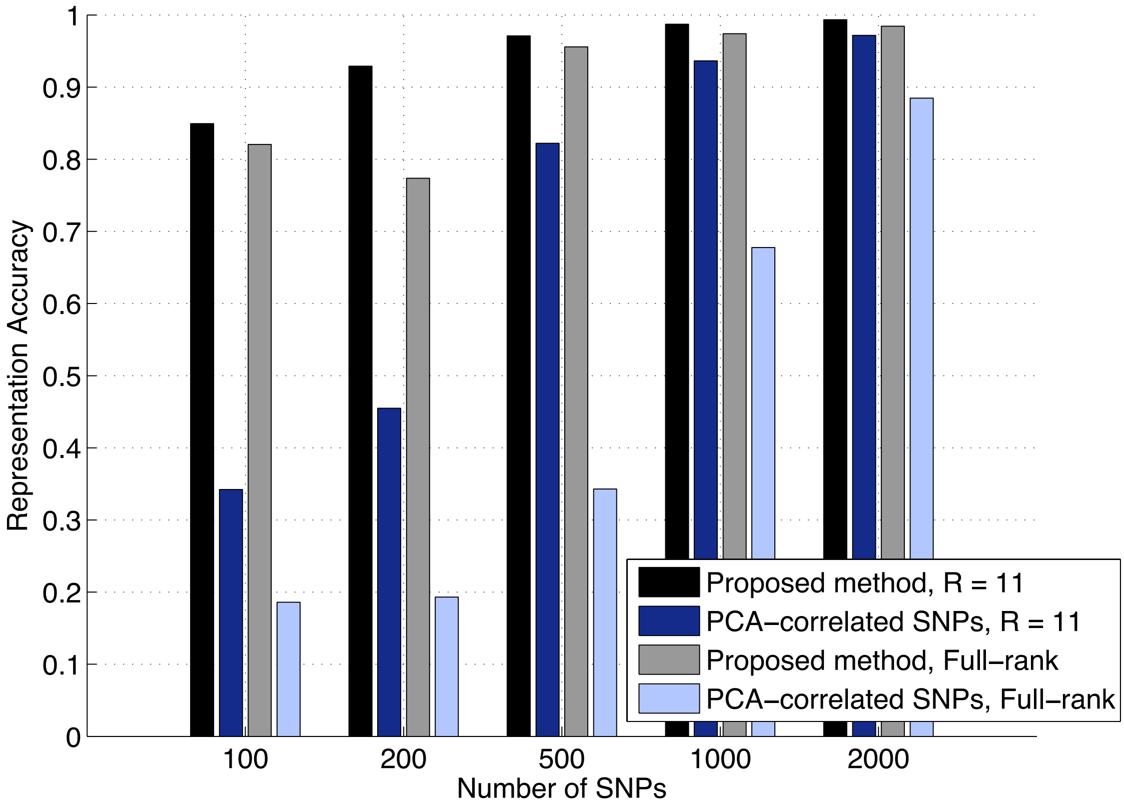

Figure 24. Data representation accuracy.

Fig. 24 depicts the values of $\gamma(3)$ for the numbers of SNPs ranging between 100 to 2000 markers. The data representation accuracy of our method is superior to the PCA-correlated SNPs method for both low-rank and full-rank results. When R=11 is used to compute the scores, the representation accuracy of our method is greater than 0.85 when we use only 100 SNPs, or just slightly over 1% of the total number of available SNPs. However, PCA-correlated SNPs selects 100 SNPs that can capture only 34% of signal energy ($\gamma(3)$= 0.34). With our proposed modification, we achieve over 0.93 accuracy with merely 200 SNPs. The PCA-correlated SNPs technique reaches the same representation accuracy using 1000 SNPs. At 2000 SNPs, or 23% of the original dimension, both methods perform well with the accuracies of 0.99 and 0.97 for our method and PCA-correlated SNPs, respectively.

For full-rank results, the representation accuracies decrease slightly for our method when small sets of 100 and 200 SNPs are used. This is because more bases representing “noise” are included in the score computation. The accuracies are comparable to the low-rank results when the number of SNPs is greater than 500, as more SNPs provide more structure information. In contrast, the degradation in representation accuracy is more substantial for PCA-correlated SNPs. This is a direct consequence of its rank-dependency and improper weighting of basis coefficient v_i^j as discussed earlier.

(3) Visualizing population structures

The population structures within the bovine dataset using two sets of 200 informative SNPs selected with our technique and the PCA-correlated SNPs technique are visualized on three dominant principal component axes in Fig. 25 and 26. We compare the results for *R* = 11, 30, 70, and 229 (full-rank). Regardless of the rank, the population structure within the original data in Fig. 21 is correctly retained using our proposed method. Separations of individuals from the same five breeds are still noticeable, although the individuals are slightly more dispersed when larger values of R are used.

For PCA-correlated SNPs method, lower values of *R* produce structures that differ from the original, as seen in Fig. 25. For *R*=11, individuals from three breeds are separated from the remaining breeds. However, only two breeds are similar to those seen in Fig. 21. For *R*=30, only two breeds are separated out. In Fig. 26, the structure also changes drastically for the PCA-correlated SNPs method when *R* becomes large (*R*=70 and *R*=229). No visible structure can be detected.

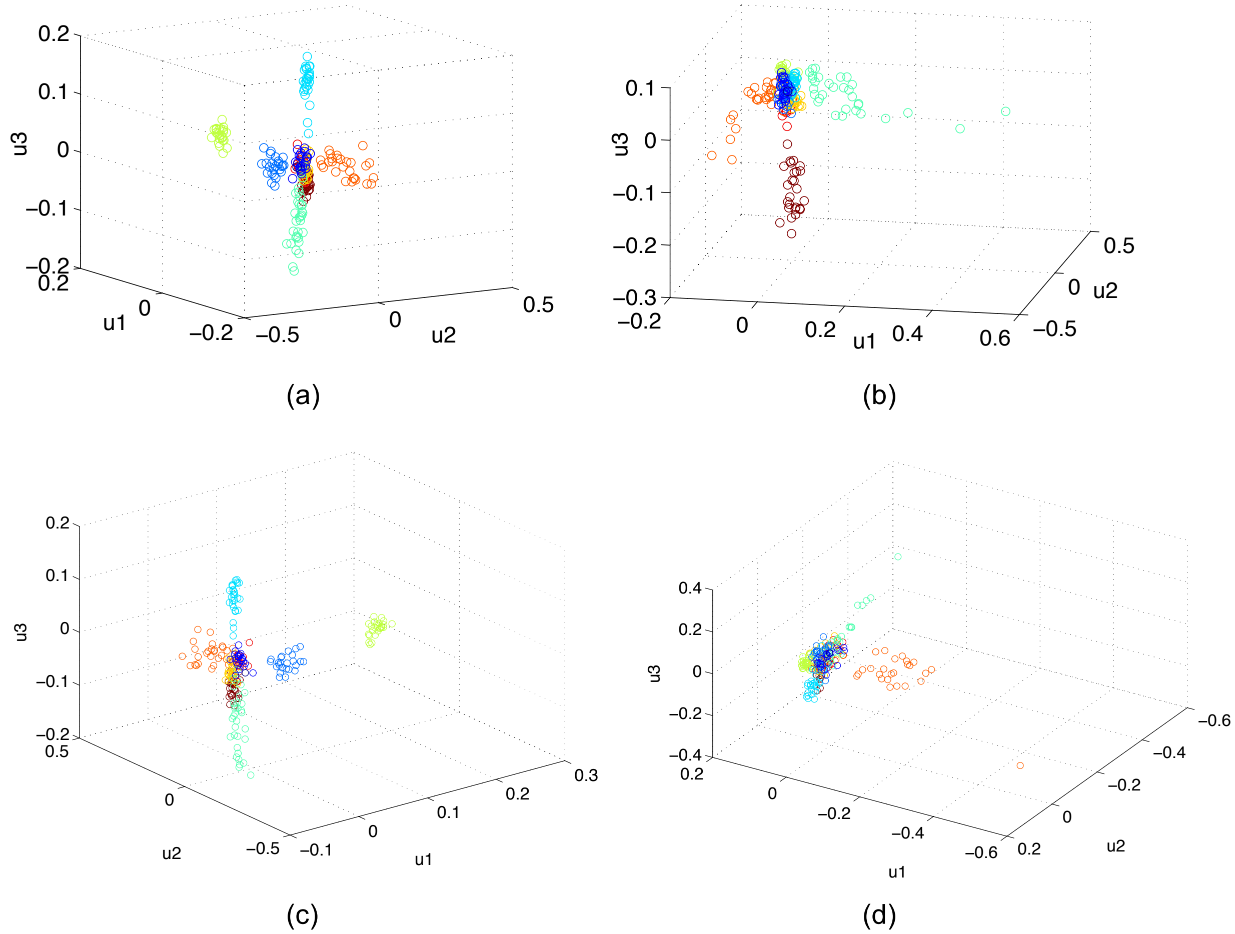

Figure 25. Structures within the bovine data set using selected sets of 200 SNPs. (a) Proposed method, *R*=11. (b) Proposed method, PCA-correlated SNPs, *R*=11. (c) Proposed method, *R*=30. (d) PCA-correlated SNPs,*R*=30.

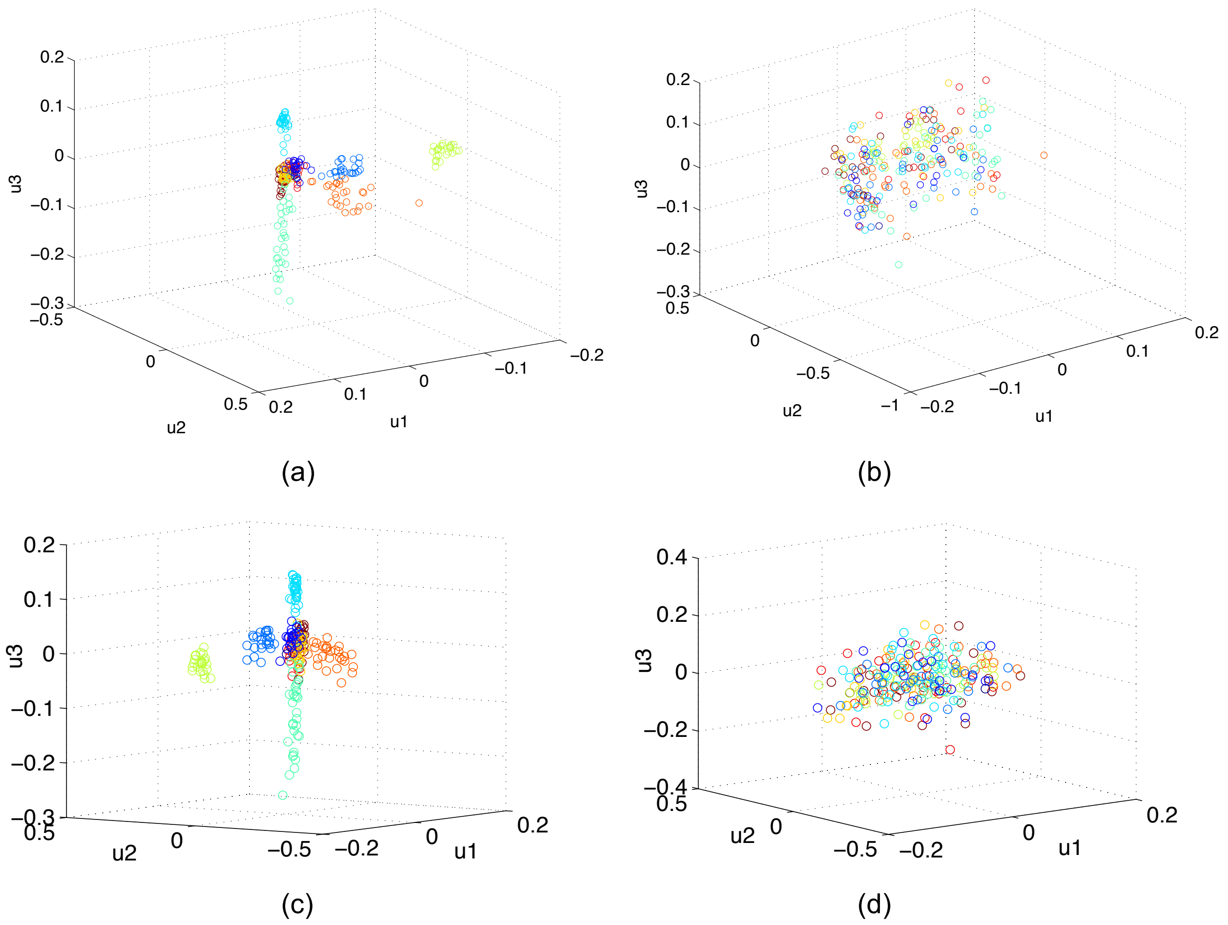

Figure 26. Structures within the bovine data set using selected sets of 200 SNPs. (a) Proposed method, *R*=70. (b) Proposed method, PCA-correlated SNPs, *R*=70. (c) Proposed method, *R*=229 (full-rank). (d) PCA-correlated SNPs, *R*=229 (full-rank).

V. Conclusions

In the first part of our research project, a graph-theoretic approach has been applied to develop an unsupervised, iterative algorithm for clustering genotypic data. Popular genetic measures are exploited to produce clustering results that are genetically meaningful. The NJ tree clustering is used to distinguish between populations based on their intrinsic genetic relationships. It is discovered that the iterated tree reconstruction process is crucial. The stopping criterion based on the fixation index is mathematically sound and is flexible. The sensitivity of the clustering result can be controlled by adjusting the threshold of the stopping criterion. However, no process is discovered to produce an optimal value of the ΔF criterion. In stead, ranges of decent ΔF values for different complexity of data sets have been suggested heuristically. The threshold of 0.001 is chosen as the default threshold value for data having more than 20 populations. We suggest using a threshold of 0.002-0.003 for data containing 10-20 populations, and a threshold of 0.01 for data with fewer than five populations. Since the threshold only determines the amount of branching at later iterations of the process, it can be adjusted to obtain the desired clustering resolution or sensitivity.

The iNJclust algorithm has been tested extensively against existing clustering algorithms of similar natures, namely the AWclust algorithm and the NJclust algorithm. Our proposed algorithm operates in a computationally efficient manner. The results illustrate that the iNJclust algorithm outperforms the other algorithms. It can effectively handle irregular cluster patterns and provide reasonable estimate of the number of populations as well as accurate individual assignments. However, because of the model choice, there is a limitation of inferring population tree topology with admixed individuals, as people with admixture may be assigned to different branches on the tree to which they have similarities.

For the second part of the research, we have modified the PCA-correlated SNPs technique for identifying structure informative SNPs by improving the calculation of the informativeness score for each SNP and select a small subset of SNPs with the best scores. The proposed technique is simple and efficient. It is demonstrated that the result is robust to the assumed rank of the data, i.e., the choice of a rank estimation technique has little effect on the final selection of informative SNPs. In fact, rank estimation may be bypassed with negligible degradation in data representation accuracy. Additionally, sizable dimensional reduction can be achieved while retaining information on the underlying population structure from the original data.

For an extension of this work, we plan to look at the performance of our methods on more human data sets with varying complexities, including the ones used in [8, 9]. We believe that our techniques are advantageous in the cases where we want to study the population structure at a finer scale, e.g. populations within continents or with common ancestry.

VI. Output (Acknowledge the Thailand Research Fund)

VI.A Publications

(1) International Journal Publication

Limpiti, T.; Amornbunchornvej, C.; Intarapanich, A.; Assawamakin, A.; Tongsima, S., "iNJclust: Iterative Neighbor-Joining Tree Clustering Framework for Inferring Population Structure," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 5, 2014, pp. 903-914.

(2) International Conference Proceedings

T. Limpiti, A. Intarapanich and S. Tongsima, "PCA-based informative SNP selection for analyzing population structure," *Proceedings of the 7th International Conference on Computational Systems-Biology and Bioinformatics (CSBio2016)*, 19-22 Dec 2016, Macao, Macau.
* The conference paper will be published in the Scopus-indexed International Conference Proceedings Series by ACM.

(See the appendix for the reprints of both publications.)

VI.B Software package

We have developed executable version of the iNJclust algorithm and make the software available for the public. A copyright request has also been filed with the Software Industry Promotion Agency (SIPA). The executable and source codes of the iNJClust algorithm for Windows and Linux platforms can be downloaded from the website at http://www.biotec.or.th/GI/tools/injclust.

VII. References

[1] N. Chinchor, “Evaluation metrics,” in Proc. 4th Message Understanding Conf., pp. 22–29, 1992.

[2] F. Boutin and M. Hascoet, “Cluster validity indices for graph partitioning,” *Proceedings of the International Conference on Information Visualization (IV’2004),* London, UK, July 2004.

[3] F. Kovacs, C. Legany and A. Babos, “Cluster validity measurement techniques,” *6th International Symposium of Hungarian Researchers on Computational Intelligence*, Budapest, Hungary, November 2005.

[4] S.J. Peter and S.P. Victor, “Clustering validity with minimum spanning tree based clustering,” *Journal of Theoretical and Applied Information Technology*, vol. 17, no. 1/2, pp. 89-96, 2010.

[5] J. Handl, J. Knowles, and D.B. Kell, “Computational cluster validation in post-genomic data analysis,” *Bioinformatics*, vol. 21, no. 15, pp. 3201-3212, 2005.

[6] P. Paschou, E. Ziv, E. Burchard, S. Choudhry, W. Rodriguez-Cintron, M. Mahoney and P. Drineas, “PCA-correlated SNPs for structure identification in worldwide human populations. *PLoS Genet.*, vol. 3, no. 9, pp. 1672–1686, 2007.

[7] J. Kijas, D. Townley, B. Dalrymple, M. Heaton, J. Maddox, A. McGrath, P. Wilson, R. G. Ingersoll, R. McCulloch, S. McWilliam, D. Tang, J. McEvan, N. Cockett, V. H. Oddy, F. W. Nicholas, H. Raadsma, Int. Sheep, and Genomics Consortium, “A genome wide survey of SNP variation reveals the genetic structure of sheep breeds,” *PLoS ONE*, vol. 4, no. 3, p. e4668, 2009.

[8] M. Gautier, D. Laloë, and K. Moazami-Goudarzi, “Insights into the genetic history of french cattle from dense SNP data on 47 worldwide breeds,” *PLoS ONE*, vol. 5, no. 9, p. e13038, 2010.

[9] J. Xing, W. S. Watkins, D. J. Witherspoon, Y. Zhang, S. L. Guthery, R. Thara, B. J. Mowry, K. Bulayeva, R. B. Weiss, and L. B. Jorde, “Fine-scaled human genetic structure revealed by SNP microarrays,” *Genome Res.*, vol. 19, no. 5, pp. 815–825, 2009.

[10] X. Gao and J. Starmer, “AWclust: Point-and-click software for non-parametric population structure analysis,” *BMC Bioinformatics*, vol. 9, pp. 77, 2008.

[11] D. H. Alexander, J. Novembre, and K. Lange, “Fast model-based estimation of ancestry in unrelated individuals,” *Genome Res.*, vol. 9, no. 19, pp. 1655–1664, Jul. 31 2009.

Appendix

Manuscript reprints

iNJclust: Iterative Neighbor-Joining Tree Clustering Framework for Inferring Population Structure

Tulaya Limpiti, Chainarong Amornbunchornvej, Apichart Intarapanich,
Anunchai Assawamakin, and Sissades Tongsima

Abstract—Understanding genetic differences among populations is one of the most important issues in population genetics. Genetic variations, e.g., single nucleotide polymorphisms, are used to characterize commonality and difference of individuals from various populations. This paper presents an efficient graph-based clustering framework which operates iteratively on the Neighbor-Joining (NJ) tree called the *iNJclust* algorithm. The framework uses well-known genetic measurements, namely the allele-sharing distance, the neighbor-joining tree, and the fixation index. The behavior of the fixation index is utilized in the algorithm's stopping criterion. The algorithm provides an estimated number of populations, individual assignments, and relationships between populations as outputs. The clustering result is reported in the form of a binary tree, whose terminal nodes represent the final inferred populations and the tree structure preserves the genetic relationships among them. The clustering performance and the robustness of the proposed algorithm are tested extensively using simulated and real data sets from bovine, sheep, and human populations. The result indicates that the number of populations within each data set is reasonably estimated, the individual assignment is robust, and the structure of the inferred population tree corresponds to the intrinsic relationships among populations within the data.

Index Terms—Allele-sharing distance, clustering, fixation index, neighbor-joining tree, population structure analysis

1 Introduction

The study of differences in allele frequencies, called population stratification or structure, is one of the key topics in population genetics. Understanding complex structure facilitates researchers to comprehend the effects of evolutionary forces that shape the current populations. Not only used in population ancestry and migration studies, e.g., [1], the impact of population structure has been echoed in the field of association studies where the population structure should be detected and corrected [2], [3], [4]. Furthermore, the concept of population structure can also be applied to study breed composition and traceability of livestock [5]. Distinct sequences of genetic data, e.g., single nucleotide polymorphisms (SNPs), represent differences among individuals [6]. With the advent of parallel genotyping technology, over a million of SNPs can now be genotyped for each person to create an individual SNP profile. A genotypic data set may contain several thousands of individuals, each of which has a million of SNPs to be analyzed. Thus, an efficient way to handle such high-complexity, high-dimensional data sets is desirable.

There are different approaches in designing algorithms for population studies. The first approach relies on genetic model in which each individual is assigned with inferred ancestral contributions. Bayesian inference is implemented directly into these algorithms in order to cluster the individuals. Widely-used methods include STRUCTURE [7], and its variations, e.g., [8], [9]. ADMIXTURE [10] also estimates the ancestry ratio, but does not implement the Bayesian clustering from the admixture results. The second approach is non-parametric. For example, EIGENSTRAT/SmartPCA [2], utilizes principal component analysis (PCA) by means of spectral decomposition [2], [11]. The ipPCA algorithms [12], [13] perform clustering after PCA by clustering arrays of genetic profiles that are transformed to the principal component subspace, whereas the AWclust algorithm [14] utilizes an allele-sharing distance (ASD) matrix. Thus distance among the individuals can be used to distinguish clusters. It can be illustrated that information obtained from the admixture ratio using the first class of approaches can be used complementarily to corroborate the resulting clusters from the algorithms in this second class of methods [13].

A different viewpoint for understanding populations from genetic data focuses on the demographic or evolutionary history of populations. This genetic relationship may be inferred by a phylogenetic tree [15]. A widely-used phylogenetic tree construction algorithm is the Neighbor-joining (NJ) tree [16], which estimates an additive tree from a genetic distance matrix. The cost of an edge on an NJ tree, represented by its length, corresponds to the genetic

- *T. Limpiti and C. Amornbunchornvej are with the Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok 10520, Thailand. E-mail: kltulaya@kmitl.ac.th, grandca@gmail.com.*
- *A. Intarapanich is with the National Electronics and Computer Technology Center (NECTEC), Klongluang, Pathumthani 12120, Thailand. E-mail: apichart.intarapanich@nectec.or.th.*
- *A. Assawamakin is with the Department of Pharmacology, Faculty of Pharmacy, Mahidol University, Rajathevi, Bangkok 10400, Thailand. E-mail: anunchaiice@gmail.com.*
- *S. Tongsima is with the National Center for Genetic Engineering and Biotechnology (BIOTEC), Klongluang, Pathumthani 12120, Thailand. E-mail: sissades@biotec.or.th.*

Manuscript received 27 May 2013; revised 12 Feb. 2014; accepted 23 Apr. 2014. Date of publication 6 May 2014; date of current version 2 Oct. 2014.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TCBB.2014.2322372

distance between individuals (the tree nodes). The quality of an NJ tree depends upon the quality of the distance matrix used for tree construction [17]. Due to the tree's ability to capture spatial structure relationship, Li et al. [18] employ NJ tree to correct population structure. Several new phylogenetics-based methods for analyzing human populations genetics data have been proposed recently. Treemix [19] accounts for both population splits and gene flows, and creates an admixture tree containing all populations in the data set. On the other hand, Mixmapper [20] proposes a two-stage tree building mechanism where admixed populations are added to an initial unadmixed scaffold tree. Both methods are similar in nature and construct admixture trees from genome-wide allele frequencies data. The phylogenetic trees have been used to interpret genetic relationships among populations by inspection of their hierarchical structures, but has not yet been viewed in a clustering sense. Nevertheless, they provide complementary information to those obtained from the clustering methods for analyzing population structure.

We present a new computational framework for automatically classifying individuals to clusters called iterative neighbor-joining tree clustering or *iNJclust*. Instead of clustering from PCA-derived data points, the framework uses relatedness information between populations provided by the phylogenetics-based methods to resolve complex population structure. The iNJclust algorithm performs a graph-based clustering on the NJ tree constructed using an ASD matrix. Graph-partitioning techniques have been utilized to cluster complex patterns due to its robustness [21], [22], [23]. Data points are viewed as nodes of a graph, whereas the graph topology captures the pattern of clusters. One of the most common techniques to perform clustering on a graph is by selectively cutting the graph edges, e.g., the longest edge of a minimum spanning tree [21], until the number of clusters, k, is achieved. To our knowledge, the AWclust algorithm [14] is the first algorithm to adopt the graph-based clustering scheme to resolve population structure. This tool constructs a hierarchical tree from an ASD matrix. Each entry of the ASD matrix represents an average allele difference between a pair of individuals. The clustering step is done by partitioning the tree at a certain depth that gives the number of populations derived from Gap statistics [24]. However, due to high complexity of Gap statistics, the method works well only when the number of clusters is small. Furthermore, edges in a hierarchical tree do not reflect the underlying genetic relationships among individuals that are of importance in population genetics. Thus, we instead use the NJ tree, which is constructed from the intrinsic genetic relationships. We also adopt the iterative process from, [12], [13] so that the iNJclust algorithm is computationally efficient.

We consider our algorithm to be a non-parametric clustering algorithm, most similar to the AWclust algorithm, which also perform graph-based clustering on a tree. So we choose to compare our algorithm with AWclust. Instead of using the ASD, other model-based clustering algorithms, e.g., [7], [8], [9], [10], assign individuals to populations using the posterior probability that an individual belongs to each of the populations. These STRUCTURE-based algorithms shed different light on the data, hence their results are complimentary to our method.

In the preliminary version of iNJclust proposed in [25], a criterion for detecting cluster homogeneity based on a particular topological pattern of the NJ tree branch (called UT1 topology) has been proposed. However, this UT1 criterion is purely ad hoc. It is speculated from observing the NJ tree of the test data sets. Although the iNJclust framework with the UT1 criterion can cluster well to a certain degree, the relationship between the terminating topological pattern and the genetic distances between populations is not clear. Moreover, the criterion places a limit on the number of individuals within a single population that can be clustered correctly. In this paper we present an improved version of the iNJclust framework. Significant updates have been added to the algorithm. A novel criterion ΔF, which is derived from a measure of population structure difference called the fixation index (F_{st}) [26], is used to terminate iNJclust's process. Using ΔF, the difference between clustered populations is now quantifiable and has a basis in genetics. The behavior of F_{st} after clustering can also be proved mathematically. Besides resolving population structure in three major aspects commonly performed in a clustering method: detecting population structure, predicting the number of populations within the data set, and assigning individuals to predicted populations, the iNJclust algorithm also derives a bifurcated population tree based on the order at which each population is separated from the original data set. The terminal nodes of the tree represent the final inferred populations and the tree structure preserves the genetic relationships among them.

The remaining of this paper is organized as follows. The iNJclust algorithm and its properties are presented in details in the next section. We thoroughly investigate the performance of the new iNJclust algorithm with the ΔF stopping criterion in Section 3, using both simulated and real data sets. The paper ends with some insightful discussions in Section 4.

2 METHODS

2.1 The iNJclust Algorithm

The system flowchart of the proposed algorithm is depicted in Fig. 1. SNP sequences of M individuals form the input matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M]^T$. Each column vector $\mathbf{x}_i, i = 1, \ldots, M$ is genotyped from L loci of individual i, at which there are two alleles of either A (major allele) or a (minor allele) for three possible genotypes (AA, Aa, aa). The SNP sequence is encoded by counting the number of minor allele a. Therefore, $\mathbf{x}_i$ is an L-dimensional vector of numerical values 0, 1, or 2.

After receiving the input matrix $\mathbf{X}$, the iNJclust algorithm computes the ASD matrix $\mathbf{D}$, whose elements are

$$\mathbf{D}(i,j) = \frac{1}{L}\|\mathbf{x}_i - \mathbf{x}_j\|_1, i, j = 1, \ldots, M. \tag{1}$$

Therefore, $\mathbf{D}(i,j)$ is proportional to the L1-norm of the pairwise difference between the SNP sequences of individual i and individual j. The smaller the value of $\mathbf{D}(i,j)$, the closer the pair are genetically.

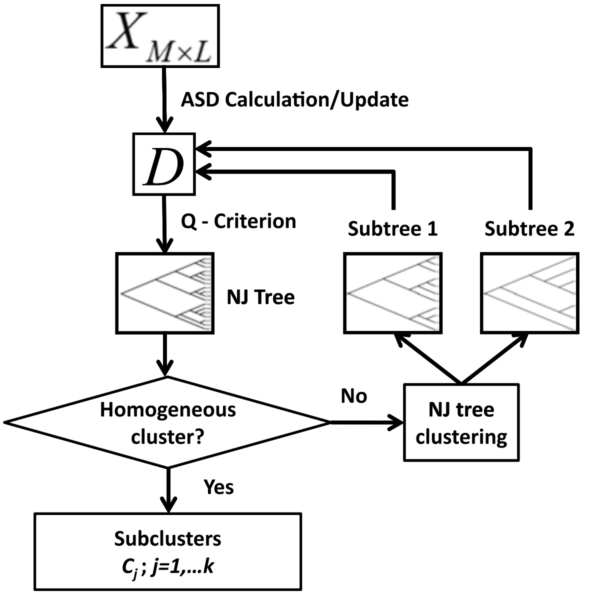

Fig. 1. The iNJClust system flowchart.

We choose ASD as our distance matrix because it has been suggested that it contains accurate information for clustering if the number of SNPs are sufficiently large [27]. It is also adopted by the AWClust algorithm [14].

The ASD matrix $\mathbf{D}$ is subsequently used to construct the NJ tree. Treating each individual as a leaf node, a pair of nodes who are nearest to each other as measured by the minimum evolution criteria or the Q criterion [28]

$$Q(i,j) = (r-1)d(i,j) - \sum_{k=1}^{r} d(i,k) - \sum_{k=1}^{r} d(j,k) \tag{2}$$

are merged into a parent node. $d(i,j)$ is the Fitch-Margoliash distance [29] between node i and j, and r is the number of the remaining nodes. The tree construction process continues until all nodes are merged onto the NJ tree. The algorithm then determines if the cluster is homogeneous, i.e., all individuals on the tree come from the same population. If the cluster is said to be heterogeneous, the algorithm performs clustering on the NJ tree by bisecting the tree into two subtrees. The NJ tree is split at the longest branch (edge) between two nodes within the tree. In the next stage, both subtrees cycle back to the ASD update step where they are processed independently. The method iterates until all populations are considered homogeneous. The iterative scheme is similar to the previous PCA-based algorithms [12], [13], which are shown to be computationally efficient and increase clustering resolution.

To avoid the effect of noise or outliers, we give a threshold for cluster size when constructing a phylogenetic tree, similar to [18]. The cutoff is set at 10 percent. Note that the 10 percent cutoff is calculated from the number of individuals remained in the new tree at each iteration, not the global 10 percent of individuals in the entire data set. Put in another way, at later stages of the iteration process, the tree contains far fewer number of individuals, thus 10 percent is a relatively small number. This technique should not affect clusters that are less homogeneous due to inbreeding, as they tend to be separated later in the process. However, the tradeoff is that some intrinsic populations that are originally small in size cannot be differentiated.

For iNJclust, the ASD matrix is only calculated once at the initial iteration for the original input matrix $\mathbf{X}$. The ASD matrix of individual within each subtree is readily available by selecting elements of the original ASD matrix $\mathbf{D}$ corresponding to the appropriate subset of individuals. After the new ASD matrix is obtained, the NJ tree of each subset of individuals is constructed. We note that the new NJ tree reconstructed at each iteration may have different topology from the old tree, since the genetic relationships illustrated in the new tree only account for those individuals within the subcluster, thus enhancing the sensitivity of the algorithm to differentiate closely-related populations. The cluster homogeneity criterion is then checked against the new tree. We explain this newly proposed criterion in details in the following section. Once the subtree is deemed homogenous, it is output from the process as one of the final population $\mathcal{C}_i$. The choice to compare individuals within each tree means that it can only observe differences within a data set, not differences with respect to some third population. The selection of the next cluster to evaluate is done using a breadth-first-search approach. Because the NJ tree clustering only bisects the tree in each iteration, the number of final iterations $\hat{k}$ provides an estimated number of populations. We choose to present the clustering result of the iNJclust algorithm in the form of a binary tree, whose terminal nodes represent the final populations and the tree structure preserves the genetic relationship among inferred populations.

The executable and source codes of the iNJClust algorithm for Windows and Linux platforms can be downloaded from our website [http://www.biotec.or.th/GI/tools/injclust].

2.2 Determining Cluster Homogeneity

One factor which is widely used to measure homogeneity of populations is the Fixation index or F_{st} [26]. It is defined as follows. Suppose a data set of M individuals is composed of N populations $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_N$ containing $m_1, m_2, \ldots, m_N$ individuals, respectively. Let the corresponding dominant allele frequencies of each population be $p_1, p_2, \ldots, p_N$. Hence, the average dominant allele frequencies of the entire data set is $\overline{p} = \frac{1}{M}\sum_{i=1}^{N} p_i m_i$, and the quantity

$$H_T = 2\overline{p}(1-\overline{p}) \tag{3}$$

represents expected heterozygosity of the entire data set. The expected heterozygosity of all populations is computed from

$$H_s = \frac{1}{M}\sum_{j=1}^{N} H_j(p_j)m_j, \tag{4}$$

where $H_j(p_j) = 2p_j(1-p_j)$ is the local expected heterozygosity of population j. By definition, the F_{st} value is

$$F_{st} = \frac{H_T - H_s}{H_T}. \tag{5}$$

That is, the normalized difference between the expected heterozygosities of all populations and the total data set. Large value of F_{st} indicates that the intrinsic populations are highly dissimilar when viewed as a whole, i.e., the population is heterogeneous. Small value of F_{st} means that the population is more homogenous.

Let us first investigate the behavior of the fixation index value after data clustering. It can be proven that the fixation index monotonically increases at each iNJclust iteration until a homogeneous cluster is formed.

2.2.1 F_{st} Behavior After Data Clustering

Proposition 1. *Let F_k be the F_{st} value computed at the k^{th} iteration of the iNJclust algorithm. F_k is non-decreasing, i.e.,*

$$F_{k+1} \geq F_k. \tag{6}$$

Proof. At the k^{th} iteration, the original cluster of M individuals is divided into $k \geq 2$ subclusters containing $m_1, m_2, \ldots, m_k$ individuals, respectively. Let the corresponding dominant allele frequencies within each subcluster be $p_1, p_2, \ldots, p_k$.

Recall that the F_{st} value at the k^{th} iteration is

$$F_k = \frac{H_T - H_s^k}{H_T} = 1 - \frac{H_s^k}{H_T}.$$

Hence, to prove that $F_{k+1} \geq F_k$ it is equivalent to showing $H_s^{k+1} \leq H_s^k$.

From (4)

$$H_s^k = \frac{1}{M}\sum_{j=1}^{k} H_j(p_j)m_j = C + \frac{1}{M}H_k(p_k)m_k, \tag{7}$$

where

$$C \equiv \frac{1}{M}\sum_{j=1}^{k-1} H_j(p_j)m_j. \tag{8}$$

If the cluster at iteration k is considered heterogeneous, the NJ tree clustering bisects the cluster. Consequently,

$$\begin{aligned} H_s^{k+1} &= C + \frac{1}{M}H_{k+1}^1\left(p_{k+1}^1\right)m_{k+1}^1 \\ &\quad + \frac{1}{M}H_{k+1}^2\left(p_{k+1}^2\right)m_{k+1}^2, \end{aligned} \tag{9}$$

where

$$m_k = m_{k+1}^1 + m_{k+1}^2. \tag{10}$$

Trivially,

$$p_k = \frac{p_{k+1}^1 m_{k+1}^1 + p_{k+1}^2 m_{k+1}^2}{m_k}. \tag{11}$$

Hereafter we drop the subscript denoting the iteration number and shorthand p_{k+1}^1 as p^1, p_{k+1}^2 as p^2, m_{k+1}^1 and m_{k+1}^2 as m^1 and m^2, respectively. Thus,

$$H_s^{k+1} = C + \frac{2}{M}\left[p^1(1-p^1)m^1 + p^2(1-p^2)m^2\right] \tag{12}$$

$$\begin{aligned} &= C - \frac{2(m^1+m^2)}{M}\left[-p^1(1-p^1)\frac{m^1}{m^1+m^2}\right] \\ &\quad - \frac{2(m^1+m^2)}{M}\left[-p^2(1-p^2)\frac{m^2}{m^1+m^2}\right] \end{aligned} \tag{13}$$

$$= C - \frac{2(m^1+m^2)}{M}\left[f(p^1)\lambda_1 + f(p^2)\lambda_2\right] \tag{14}$$

using $f(y) = -y(1-y)$, $\lambda_1 = \frac{m^1}{m^1+m^2}$, and $\lambda_2 = \frac{m^2}{m^1+m^2}$.

Since $\lambda_1, \lambda_2 > 0$, $\lambda_1 + \lambda_2 = 1$, and $f(y)$ is a continuous concave up function, it follows from Jensen's inequality that

$$H_s^{k+1} \leq C - \frac{2(m^1+m^2)}{M}\left[f\left(p^1\lambda_1 + p^2\lambda_2\right)\right] \tag{15}$$

$$= C - \frac{2(m^1+m^2)}{M}\left[f(p_k)\right] \tag{16}$$

$$= C + \frac{1}{M}H_k(p_k)m_k \tag{17}$$

$$= H_s^k. \tag{18}$$

If the cluster is already homogenous before splitting, the average dominant allele frequencies $p^1 \approx p^2 \approx p_k$, thus

$$H_s^{k+1} \approx H_s^k.$$

To generalize, the proposition holds when cluster $\mathcal{C}_k$ is divided into $L \geq 2$ subclusters, since we can write

$$p_k = \frac{\sum_{l=1}^{L} p_l m_l}{\sum_{l=1}^{L} m_l}, \tag{19}$$

$$\lambda_l = \frac{m_l}{\sum_{l=1}^{L} m_l}, \tag{20}$$

and Jensen's inequality still applies. □

2.2.2 A Novel Stopping Criterion for Terminating the iNJclust Algorithm

From the proposition in Section 2.2.1, the F_{st} value of the data after each iNJclust iteration increases monotonically and converges after all populations have been identified. Recall that F_k is the F_{st} value computed at the k^{th} iteration of the iNJclust algorithm. We propose using the difference,

$$\Delta F = F_{k+1} - F_k \tag{21}$$

to detect homogeneous clusters. We announce that the cluster is homogeneous and iNJclust iteration terminates if ΔF is sufficiently small. The smaller the ΔF threshold, the higher the sensitivity of the iNJclust algorithm to differentiate between clusters.

From our extensive testing on data sets with varying sizes and complexities, the ΔF threshold of 0.001 is suitable, and is chosen as the default threshold value, for data having more than 20 populations. We suggest using a threshold of 0.002-0.003 for data containing 10-20 populations, and a threshold of 0.01 for data with fewer than five populations. Note that we provide these values only as a rough guideline. As will be shown in the following section, there is some flexibility in selecting the value of the ΔF threshold. There is not an optimal point of the threshold, but rather an optimal range for a particular population structure. Note that the threshold does not influence the structure of the

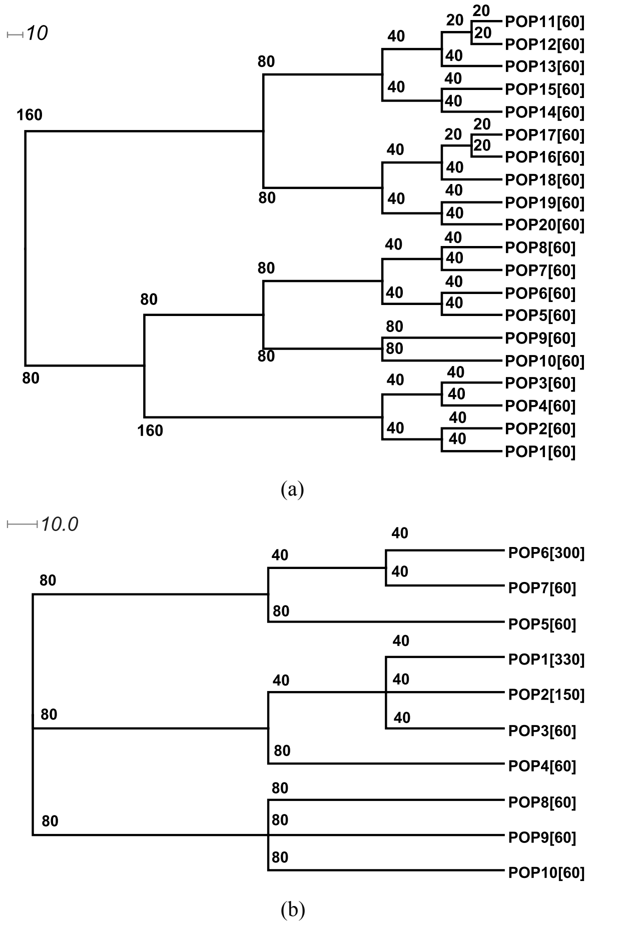

Fig. 2. Simulated population history trees. The branch lengths represent the generations of populations. (a) data set 1. (b) data set 2.

inferred population tree at early iterations. It only determines the amount of branching at later iterations of the process. Thus, the threshold can be adjusted to obtain the desirable clustering resolution/sensitivity.

For the final iNJclust algorithm, a flag has also been added to the algorithm to warn if the user-defined ΔF threshold may be too small, i.e., resulting with clusters containing only a single individual as member.

3 Results

3.1 Simulation Data Sets

In this section we first explore the performance of the iNJclust algorithm as a function of the ΔF stopping criterion. Simulated data is useful in investigating the efficacy of the algorithm since the inherent number of populations in a real genotype data set is usually unknown or self-reported. We adopt Dendroscope [30] for tree visualization of our results.

We use two simulated data sets to explore the effects of evolution time and number of populations on the optimal value of ΔF, as well as consistency of the clustering results. The data sets are generated using the GENOME tool [31]. GENOME is a whole genome simulator utilizing coalescent approach, assuming neutral genetic drift. The program is fast and is able to vary the mutation rates. We use the following parameters:

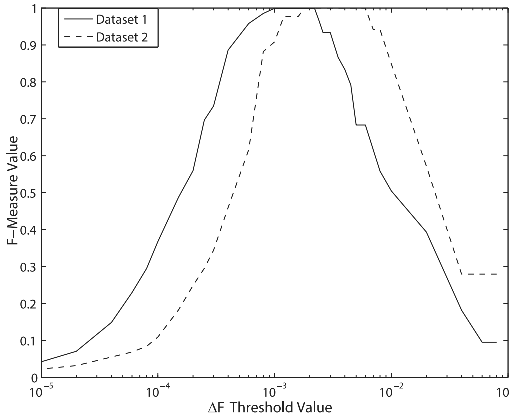

Fig. 3. F-Measure values of simulated data sets as a function of the ΔF thresholds.

Data set 1:

```
-pop 20 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 -c 20 -s 500 -N UD1model.txt
```

Data set 2:

```
-pop 10 330 150 60 60 60 300 60 60 60 60 -c 20 -s 500 -N UD2model.txt
```

The tree files *UD1model.txt* and *UD2model.txt* for generating data sets 1 and 2 are represented graphically in Figs. 2a and 2b, respectively. The branch lengths represent the evolution time of populations in terms of the number of generations they evolve. The first simulated data set contains 20 clusters $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{20}\}$ of 60 individuals each (for a total of 1,200 individuals), and 10,000 SNPs per individual. The second data set contains 1,200 individuals separated into 10 clusters $\{\mathcal{S}_1, \ldots, \mathcal{S}_{10}\}$ of varying sizes ranging from 60 to 330 individuals per cluster, also genotyped at 10,000 SNPs.

The optimal values of ΔF stopping threshold for clustering data set 1 and 2 are determined by scanning the iNJclust algorithm over possible ΔF threshold values ranging from 10^{-5} to 10^{-1}, i.e., the difference of 10 to 0.001 percent in the fixation indices of successive iterations. We compare the iNJclust clustering results $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_N\}$ at each value of ΔF to the ground truth $\mathcal{S}$ and compute the F-measure [32],

$$\Psi(\mathcal{S},\mathcal{C}) = \sum_{i=1}^{N} \frac{|S_i|}{M} \max_j \left(\frac{2 \cdot \frac{|S_i \cap C_j|^2}{|S_i||C_j|}}{\frac{|S_i \cap C_j|}{|S_i|} + \frac{|S_i \cap C_j|}{|C_j|}} \right), \qquad (22)$$

where $N = 20$ for data set 1 and $N = 10$ for data set 2. The $|\cdot|$ symbol denotes cluster size. For both data sets $M = 1{,}200$. The higher the F-measure value, the closer the result is to the simulated model. An F-measure value of 1 occurs when the iNJclust clustering result is exactly the same as the true clusters. Fig. 3 depicts the F-measure value as a function of ΔF threshold for the two simulated data sets. If the ΔF threshold is too high, the iNJclust process undersplits the clusters. Contrastly, a too-low value of the threshold oversplits the clusters. Both situations result in the decreases of the F-measure values. We also observe a step-like behavior of the F-measure values, which indicates that the optimal threshold for the ΔF stopping criterion is not a single point but rather a range, making selecting an appropriate value for the threshold slightly flexible. From

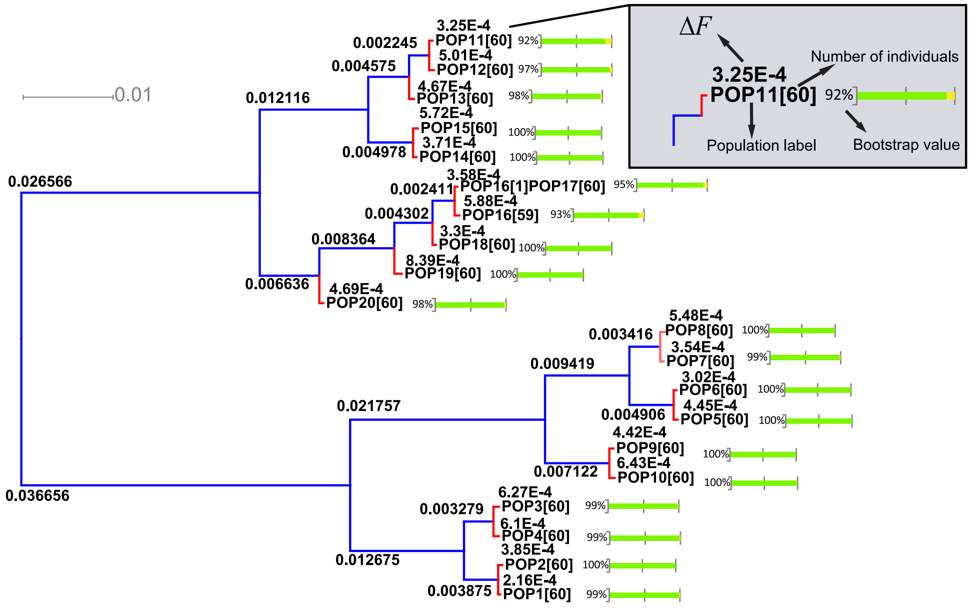

Fig. 4. Hierarchical population tree of data set 1 generated by iNJclust. The ΔF threshold is set at 0.001.

the graph, we selected the threshold of 0.001 for data set 1 and 0.003 for data set 2.

The iNJclust results are depicted in the form of hierarchical population trees inferred from the full data sets in Figs. 4 and 5. The branch lengths on the trees correspond to the computed ΔF values at each iteration. Observe that ΔF monotonically decreases as the iteration progresses. The terminal nodes of the tree contain iNJclust individual assignments, where the individuals are labeled by their true cluster number. The numbers in the square brackets represent the number of individuals within each cluster. A careful examination of the individual assignment results of the full data sets confirms that the iNJclust algorithm is able to correctly assign most individuals to their respective clusters. For data set 1, one individual from population POP16 is grouped with POP17. This is possible since the pair of populations POP16 and POP17 only differs by 20 generations; they are closely related populations in the data set. The total individual assignment is 99.92 percent correct. In data set 2, we vary the number of individuals in each population to illustrate the ability of our algorithm to handle varying cluster sizes. The result in Fig. 5 shows that the clustering performance remains excellent in this situation with no individual assignment error.

We employ the bootstrapping method [33] to investigate clustering consistency of the iNJclust algorithm. To perform bootstrapping, simulated data sets 1 and 2 are each resampled with replacement to obtain 100 bootstrap data sets with 400 individuals. Each bootstrap data set is then clustered by the iNJclust algorithm. The bootstrap percentage is represented as a slider bar at the end of each terminal node in Figs. 4 and 5. These bootstrap values correspond to the optimal performance of the algorithm, because the ΔF threshold has been selected at the point where the F-measure equals to 1, i.e., the optimal stopping point. It is discovered that most of the terminal nodes have bootstrap percentages of nearly 100 percent, validating the consistency of the iNJclust's clustering ability. The drop in bootstrap percentage at some terminal nodes happens when individuals from one or more clusters are not resampled.

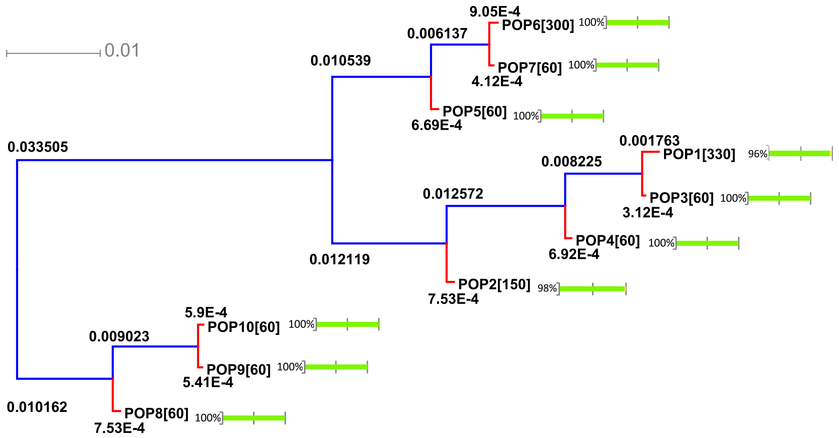

Fig. 5. Hierarchical population tree of data set 2 generated by iNJclust. The ΔF threshold is set at 0.003.

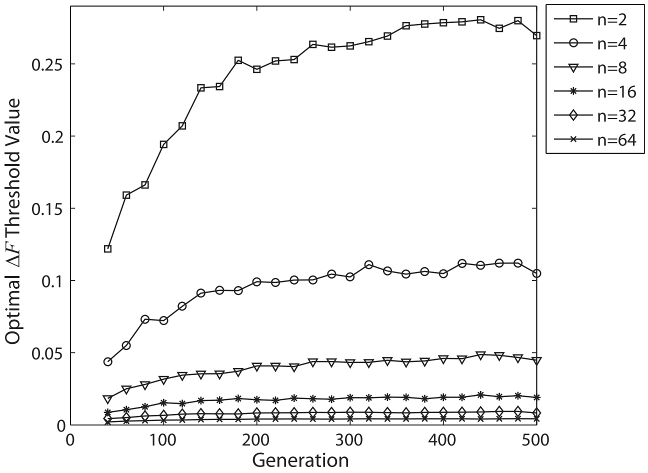

Fig. 6. Relationship between the optimal ΔF threshold value and generations. The number of populations are varied from 2 to 64.

Note that the topology of the inferred tree may be changed slightly by swapping small branches during the bifurcation. For example, in data set 1 the POP19 branching is in a different order from the simulated model. However, this does not greatly affect the overall tree topology, as the trend of the inferred tree structure is consistent with the underlying model. For data set 2, the relationships among populations in the iNJclust tree output also largely follow the structure of the simulated model, eventhough the simulated tree is not binary.

We investigate how two factors, namely evolution time and number of populations, may affect the optimal value of the ΔF threshold. To eliminate other possible parameters that may influence the threshold value, e.g., tree structure, we simulate a one-layer tree with varying number of populations and generations. The evolution time is manifested in the number of generations in the simulation, i.e., the longer the evolutions, the further apart the populations are. Hence, we vary the genetic distance between populations by simulating data ranges between 40 to 500 generations (corresponding to approximately 80 to 10,000 years of genetic evolution). We also vary the number of populations in the data from 2 to 64 populations. The result is depicted in Fig. 6. Data with larger number of generations are further apart genetically, so the threshold increases with generations as expected. On the other hand, when the number of populations increases the threshold value decreases, since more sensitivity is needed to differentiate between populations.

From Fig. 6, we observe that the threshold is also fairly robust with the number of generations (representing the evolution time of the population) when the structure is sufficiently complex. So the suggested range of the ΔF threshold in Section 2.2.2 should work quite well in practice.

3.2 Real Data Sets

We test the iNJclust algorithm on three large data sets with different complexities. The first data set is a 28-breed sheep data set [34], which contains 392 individuals and 1,046 SNPs. The second data set is from 47 breeds of 1,089 bovines [35], genotyped at 44,706 SNPs. The third data set comprises of 27 human populations spanning Europe, East Asia, India, and Africa [36] for a total of 554 individuals and 243,855 SNPs. The ΔF threshold of 0.001 has been chosen for all data sets.

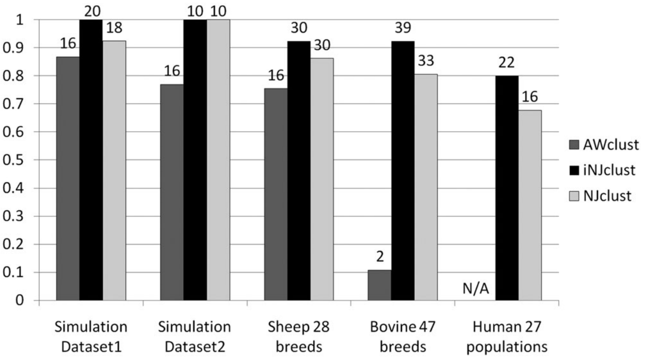

Fig. 7. F-measures of the AWclust, iNJclust, and NJclust algorithms on all data sets. The estimated numbers of clusters are displayed on top of the bar graphs.

We compare the clustering performance of iNJclust with two similar algorithms in terms of the F-measure value. *AWclust* [14] is an algorithm that also uses ASD matrix for inputs, but employs hierarchical clustering for individual assignments. A main drawback of AWclust is the fact that it can automatically detect the number of populations only up to $k = 16$. For data with more populations, k have to be supplied as one of the inputs. For fair comparison we let both iNJclust and AWclust estimate k. We also compare iNJclust with the so-called *NJclust* algorithm, which is basically the iNJclust algorithm without the successive NJ tree rebuilding step (using the same ΔF threshold). Since there is no ground truth on the underlying populations within the data, we compare the individual assignments to data labels.

The F-measures depicted in Fig. 7 confirm that the iNJclust algorithm produces the best clustering results. It estimates the correct numbers of populations for the simulated data sets. The estimated number of populations are $\hat{k} = 30$ for the 28-breed sheep data set and $\hat{k} = 39$ for 47-breed bovine data set. These estimated numbers of populations are reasonable. The estimated number of populations for the bovine data is low because the data set is more complex and contains many breeds. Some breeds, e.g., three *B. indicus* breeds {GIR, NEL, BRM} are very similar and are lumped into one cluster. For the simulated and animal data sets, the estimated numbers of cluster from the AWclust algorithm are far from the truth due to its computational limitation. The individual assignments of the AWclust algorithm are also worse than the assignments of iNJclust. We believe that the NJ tree based clustering proposed in our algorithm is more appropriate in distinguishing between populations than hierarchical clustering for genetic data. The NJclust clustering result is more erroneous than iNJclust, illustrating the necessity of reconstructing the NJ tree at each iteration.

For Human 27 populations data, $\hat{k} = 22$. It is worthy to note that the AWclust algorithm cannot be completed in a reasonable amount of time due to large data dimension and complexity, hence its F-measure value is not reported here. The F-measure value of iNJclust is 0.8. Since self-reported labels may not always correspond to the individual's intrinsic genetic pattern, we adopt the Admixture method [10], to

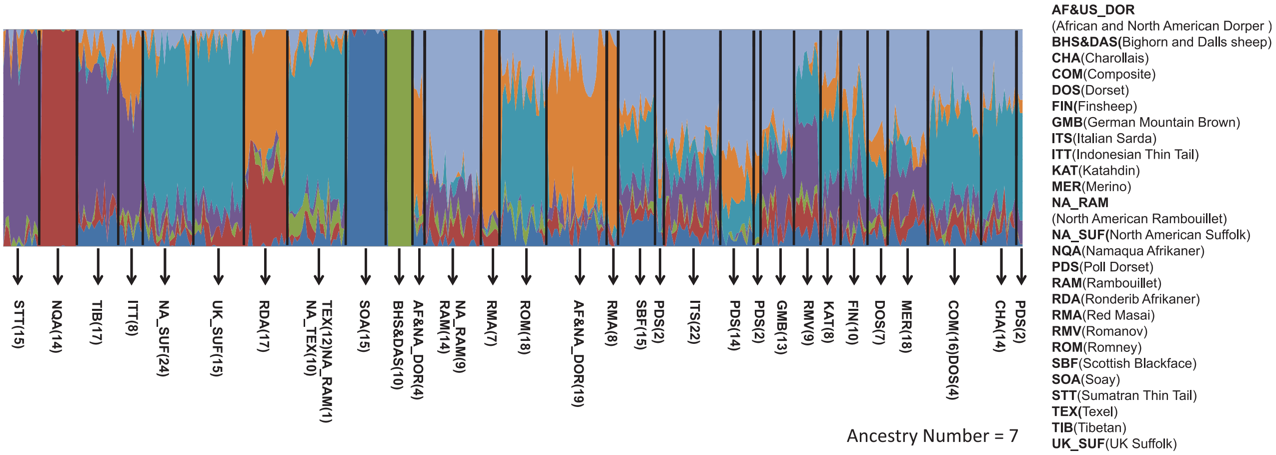

Fig. 8. Admixture results of the 28-breed sheep data set, with the ancestry number of 7.

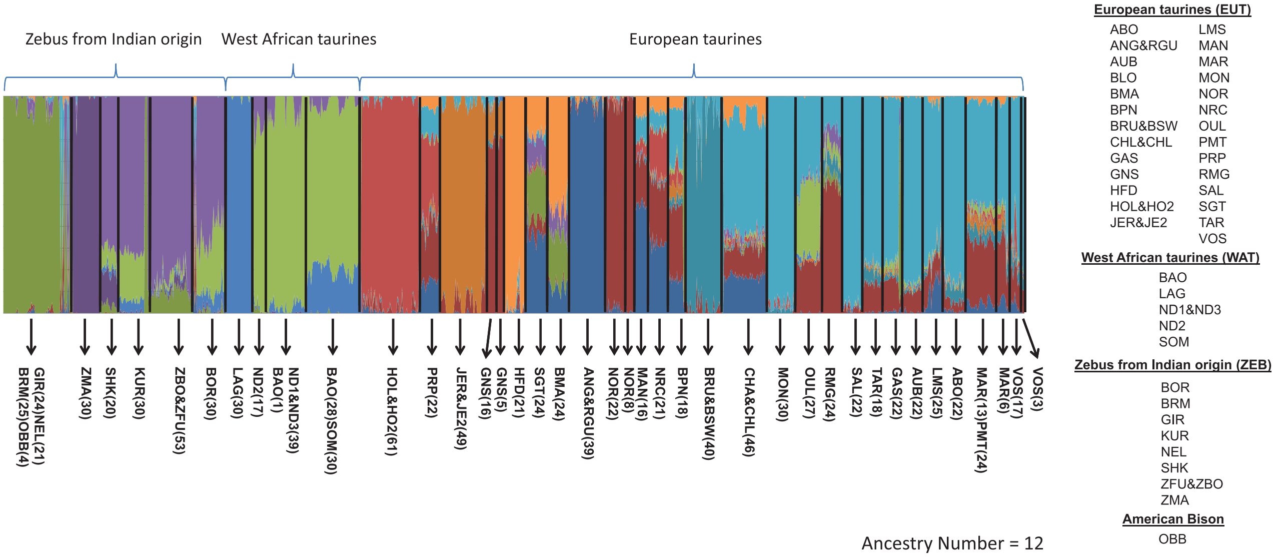

Fig. 9. Admixture results of the 47-breed bovine data set, with the ancestry number of 12.

alternatively verify the clustering results of the iNJclust algorithm on the human data set.

To gain more information on the underlying population structure by observing their ancestry contributions, in Figs. 8, 9, and 10 we corroborate the iNJclust clustering results with the admixture patterns. Each panel of admixture patterns separated by the black lines is one cluster assigned by the iNJclust algorithm. The corresponding data labels of individuals are displayed below the panels, with the number of individuals from that label shown in the bracket. Overall, the admixture results are in very good agreements with the iNJclust individual assignments. That is, each assigned iNJclust cluster has a distinct admixture pattern. Because the 28-breed sheep data set contains much fewer SNPs per individuals than the SNP profiles for the 47-breed bovine or Human 27 populations, some admixture ratios on the right-hand side of Fig. 8 are visually less distinguishable from one another, for example, {CHA(14)} and {COM(16)DOS(4)}. Nevertheless, the iNJclust algorithm is able to correctly separate these clusters. For the 47-breed bovine data set in Fig. 9, the iNJclust cluster with mixed populations {GIR, NEL, BRM, and OBB} (first panel) corresponds to non-uniform admixture patterns, whereas other homogeneous populations correspond to uniform and distinct admixture patterns.

As expected for the Human 27 populations data set in Fig. 10, some self-reported labels differ from their genetic patterns. For example, {UEP, CEU}, {STK, URK}, and {KHM, CHN, CHB, JPT, VNM} estimated clusters contain individuals with mixed labels. However, they all have similar admixture patterns. Blind to the labels, the iNJclust algorithm is able to correctly cluster them into the same cluster. In contrast, individuals from the KNG and HMA populations, though carrying the same labels, are assigned to different clusters by the iNJclust algorithm. The admixture patterns confirm their genetic differences. Using these labels to calculate the F-measure results in the lower F-measure values, which does not necessarily reflect the real clustering efficacy of the iNJclust algorithm in the human data set.

As shown in Table 1, the computational time of the iNJclust algorithm is also superior to that of the AWclust algorithm. The

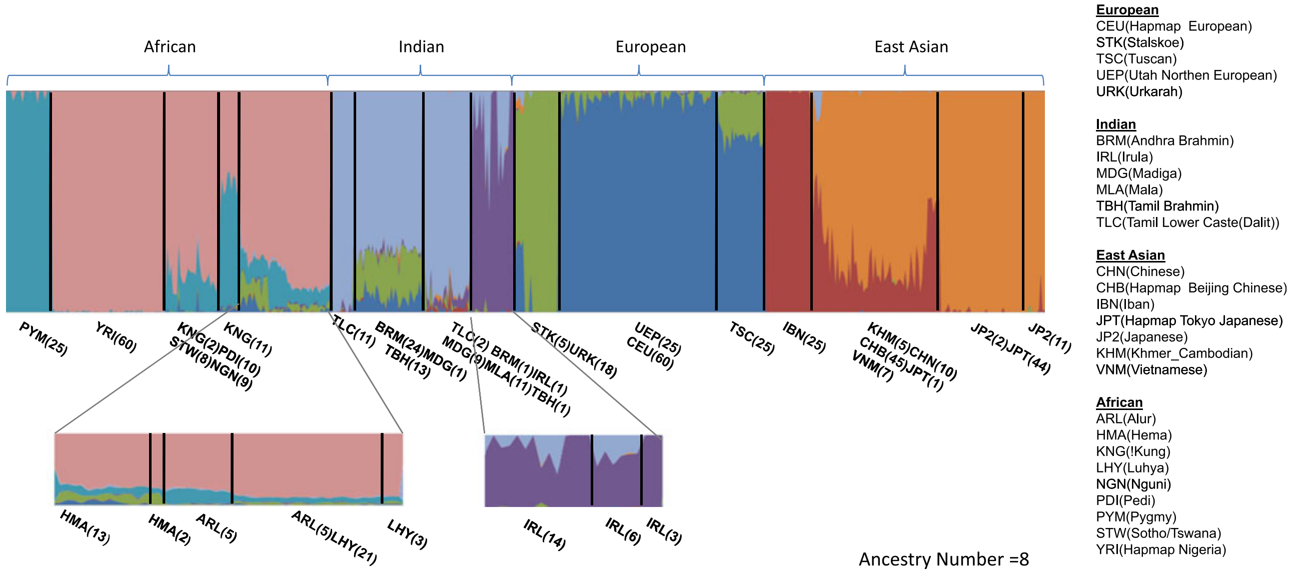

Fig. 10. Admixture results of the Human 27 populations data set, with the ancestry number of 8.

AWclust algorithm has the computational complexity of $O(n^6)$ due to the use of Gap statistic [24], while iNJclust has the complexity of only $O(n^3)$. The Gap statistic may also be the cause of estimation error in the AWclust algorithm, since it is not a genetic model. The proposed iNJclust algorithm, in contrast, uses the fixation index F_{st}, which is a measurement of true genetic distances among populations.

Another output that can be used to infer genetic similarities between populations is the population tree output. The trees generated by the iNJclust algorithm for the sheep, bovine, and human data sets are depicted in Figs. 11, 12, and 13, respectively. The structure of the population tree is constructed from the order at which each cluster is bifurcated in the iterative clustering process. The individual labels for each estimated population are reported with the number of individuals from each label in the square brackets. We observe an interesting phenomenon in the structure of the inferred trees. Population that is most distinct genetically, or has the largest number of individuals, tends to be first identified. For example, for the Bovine data set in Fig. 12 the European taurines are first separated from the West African taurines and the Zebus from Indian origin. At the second step the cluster containing *B. indicus* breeds is removed in the lower branch of the tree. At later iteration the West African taurines are broken away from the Zebus. Similarly, in Fig. 13 African individuals are separated at the first iteration, possibly because their genetic profiles are the most distinctive. Then, the East Asians populations are recognized. The Europeans and Indians, who have common ancentry as illustrated in their admixture patterns in Fig. 10, are divided at later iterations. The order at which each population are bisected in the inferred tree is very much agreeable with their corresponding admixture patterns. We believe that the resolved tree may partially reflect the actual history of population diversity. Nevertheless, the history is not observable using only a single snapshot of population variations. Hence, the resulting iNJclust population tree can only reflect the underlying relationships among populations. Similar populations tend to be clustered together or branched off from the same parental node.

TABLE 1
Computational Times of the AWclust and iNJclust Algorithms

	AWclust	iNJclust
Simulation data set 1	2 Days	48 seconds
Simulation data set 2	2 Days	46 seconds
Sheep data set	23 minutes	4 seconds
Bovine data set	10 Days	397 seconds
Human data set	N/A	1365 seconds

4 Discussion

A graph-theoretic approach has been applied to develop an unsupervised, iterative algorithm for clustering genotypic data. Popular genetic measures are exploited to produce clustering results that are genetically meaningful. The NJ tree clustering is used to distinguish between populations based on their intrinsic genetic relationships. It is discovered that the iterated tree reconstruction process is crucial. The stopping criterion based on the fixation index is mathematically sound and is flexible. The sensitivity of the clustering result can be controlled by adjusting the threshold of the stopping criterion.

The iNJclust algorithm has been tested extensively against existing clustering algorithms of similar natures, namely the AWclust algorithm and the NJclust algorithm. Our proposed algorithm operates in a computationally efficient manner. The results illustrate that the iNJclust algorithm outperforms the other algorithms. It can effectively handle irregular cluster patterns and provide reasonable estimate of the number of populations as well as accurate individual assignments.

However, because of the model choice, there is a limitation of inferring tree topology with admixed individuals, as people with admixture may be assigned to different branches on the tree to which they have similarities.

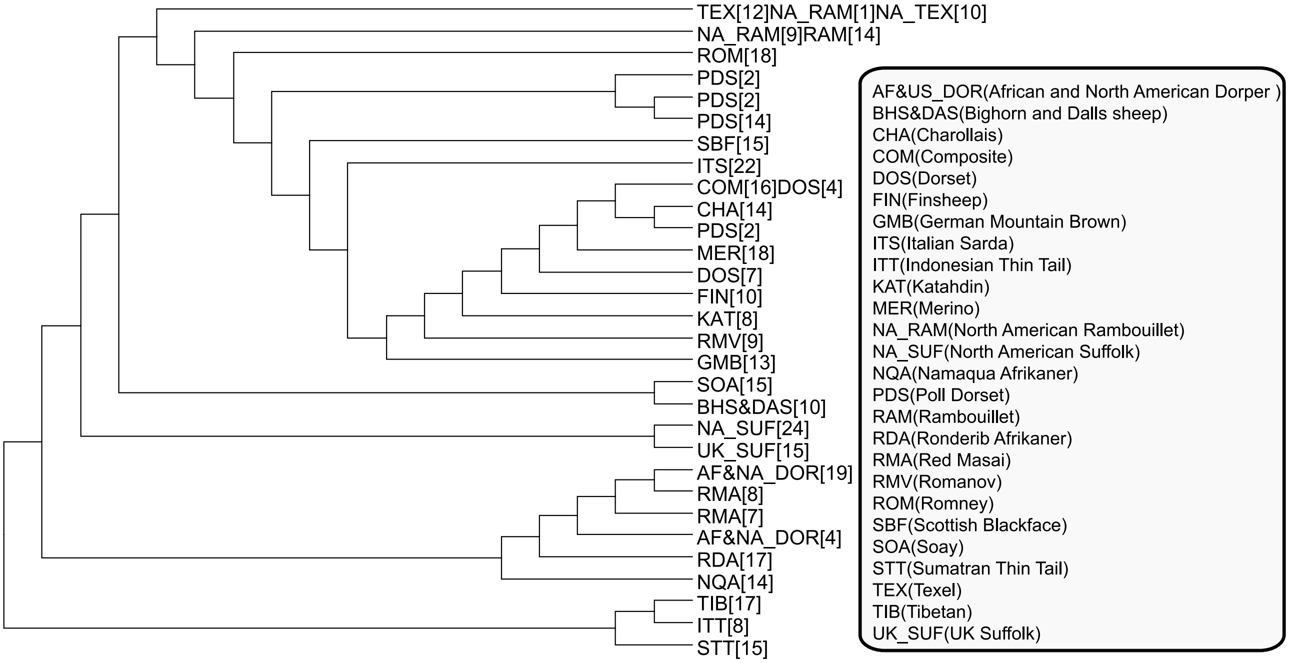

Fig. 11. Inferred population tree of the 28-breed sheep data set.

However, distinguishing individuals with admixture from other populations is possible. In practice, we do not intend to infer genetic differences solely on the inferred tree of our algorithm. In fact, we illustrate that the estimated genetic relationships between populations in the inferred population tree complement the ancestry patterns of the individuals within the data set. The iNJclust algorithm addresses more information not seen in admixture, namely the structure of the genetic similarities and accurate automatic clustering.

The iNJclust algorithm can also be used for a classic classification task where an individual with unknown origin (population) is to be classified. To perform the placement, we can append this individual's genotyping profile to the input genotyping matrix and use iNJclust to cluster the input individuals. This new entry will be clustered together with his/her affiliated population or placed as an outlier if the profile is not compatible with the rest.

Although using the algorithm outputs to adjust for stratification in association studies is beyond the scope of this paper, it is worth noting that it is plausible to do so with some concerns. Traditionally, researchers deploy genomic control and/or EIGENSTRAT [11] to detect and correct the SNPs that are likely to carry population signals.

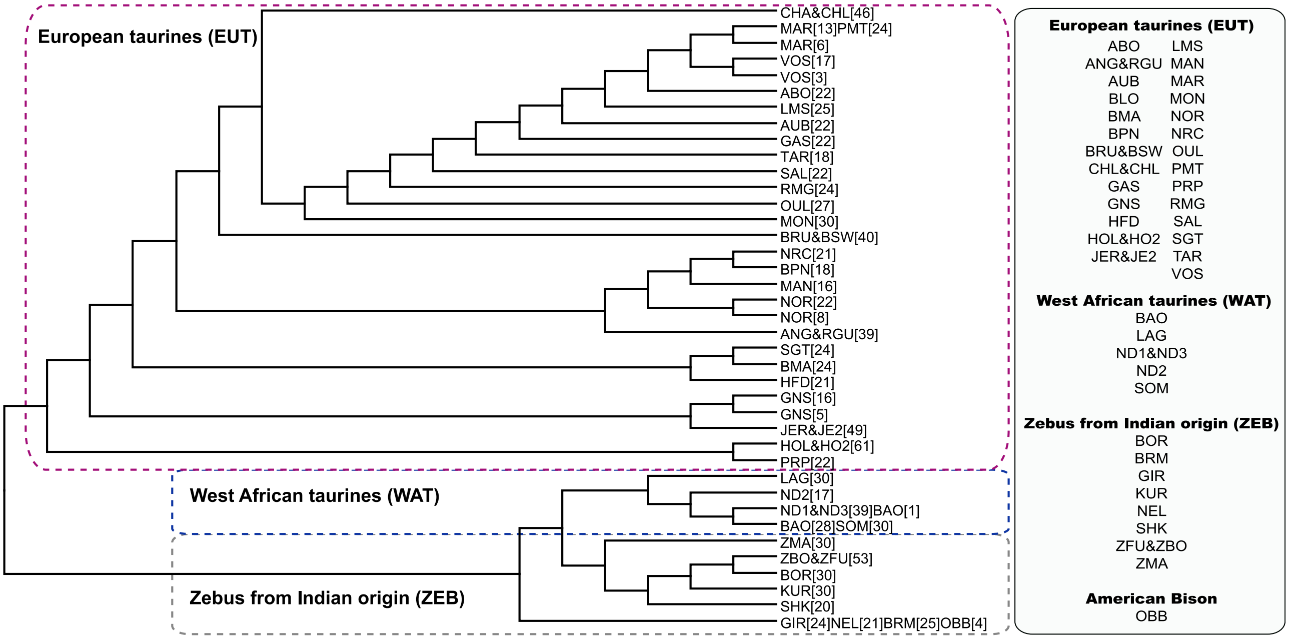

Fig. 12. Inferred population tree of the 47-breed bovine data set.

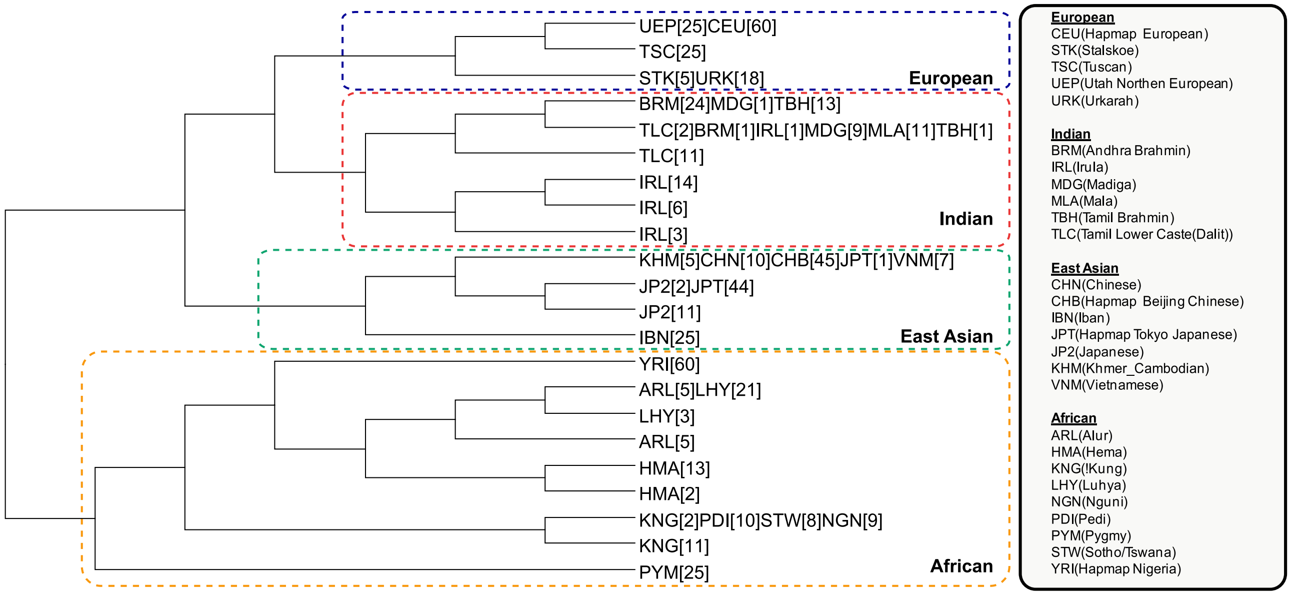

Fig. 13. Inferred population tree of the Human 27 populations data set.

Alternatively, after clustering individuals according to their genetic relatedness, e.g., using the iNJClust algorithm, genome-wide association studies (GWAS) can be performed on the cases and controls that come from the same cluster. However, sub-clustering the cohort will result in smaller number of samples that will affect the Chi-square statistical significance in GWAS. We recommend that the iNJClust algorithm may be used to recruit more individual samples of the similar genetic profile prior to performing more association studies.

Finally, our proposed algorithm can be generalized for clustering many other types of high-dimensional data such as corpus texts, images, or biosignals simply by modifying the similarity matrix, the distance measure, and constructing an appropriate type of graph before clustering.

Acknowledgments

The work of T. Limpiti is jointly supported by the Thailand Research Fund (TRF) and KMITL under Grant TRG5780045. C. Amornbunchornvej thanks the National Science and Technology Development Agency (NSTDA) for funding supports through research grant JSTP-06-54-07E. S. Tongsima would like to acknowledge the TRF under Project no. RSA5480026 and the NSTDA Research Chair Grant. The authors also thank Dr. Mathieu Gautier for the 47-breed bovine genotype data.

References

[1] The HUGO Pan-Asian SNP Consortium, “Mapping human genetic diversity in asia,” *Science*, vol. 326, no. 5959, pp. 1541–1545, 2009.

[2] N. Patterson, A. L. Price, and D. Reich, “Population structure and Eigenanalysis,” *PLoS Genetics*, vol. 2, no. 12, p. e190, 2006.

[3] Y. Zhao, F. Chen, R. Zhai, X. Lin, Z. Wang, L. Su, and D. C. Christiani, “Correction for population stratification in random forest analysis,” *Int. J. Epidemiol.*, vol. 41, no. 6, pp. 1798–1806, 2012.

[4] L. R. Cardon and L. J. Palmer, “Population stratification and spurious allelic association,” *Lancet*, vol. 361, no. 9357, pp. 598–604, 2003.

[5] A. Frkonja, B. Gredler, U. Schnyder, I. Curik, and J. Sölkner, “Prediction of breed composition in an admixed cattle population,” *Animal Genetics*, vol. 43, no. 6, pp. 696–703, 2012.

[6] L. B. Barreiro, G. Laval, H. Quach, E. Patin, and L. Quintana-Murci, “Natural selection has driven population differentiation in modern humans,” *Nature Genetics*, vol. 40, no. 3, pp. 340–345, 2008.

[7] J. K. Pritchard, M. Stephens, and P. Donnelly, “Inference of population structure using multilocus genotype data,” *Genetics*, vol. 155, no. 2, pp. 945–959, 2000.

[8] J. Huelsenbeck, P. Andolfatto, and E. Huelsenbeck, “Structurama: Bayesian inference of population structure,” *Evol. Bioinformatics*, vol. 7, pp. 55–59, 2011.

[9] D. J. Lawson, G. Hellenthal, S. Myers, and D. Falush, “Inference of population structure using dense haplotype data,” *PLoS Genetics*, vol. 8, p. e1002453, 2012.

[10] D. H. Alexander, J. Novembre, and K. Lange, “Fast model-based estimation of ancestry in unrelated individuals,” *Genome Res.*, vol. 9, no. 19, pp. 1655–1664, 2009.

[11] A. L. Price, N. J. Patterson, R. M. Plenge, M. E. Weinblatt, N. A. Shadick, and D. Reich, “Principal components analysis corrects for stratification in genome-wide association studies,” *Nature Genetics*, vol. 38, pp. 904–909, 2006.

[12] A. Intarapanich, P. J. Shaw, A. Assawamakin, P. Wangkumhang, C. Ngamphiw, K. Chaichoompu, J. Piriyapongsa, and S. Tongsima, “Iterative pruning PCA improves resolution of highly structured populations,” *BMC Bioinformatics*, vol. 10, p. 382, 2009.

[13] T. Limpiti, A. Intarapanich, A. Assawamakin, P. J. Shaw, P. Wangkumhang, J. Piriyapongsa, C. Ngamphiw, and S. Tongsima, “Study of large and highly stratified population datasets by combining iterative pruning principal component analysis and structure,” *BMC Bioinformatics*, vol. 12, p. 255, 2011.

[14] X. Gao and J. D. Starmer, “AWclust: Point-and-click software for non-parametric population structure analysis,” *BMC Bioinformatics*, vol. 9, p. 77, 2008.

[15] J. Felsenstein, “PHYLIP-phylogeny inference package (version 3.2),” *Cladistics*, vol. 5, pp. 164–166, 1989.

[16] N. Saitou and M. Nei, “The neighbor-joining method: a new method for reconstructing phylogenetic trees,” *Mol. Biol. Evol.*, vol. 4, no. 4, pp. 406–425, 1987.

[17] Z. Yang and B. Rannala, “Molecular phylogenetics: Principles and practice,” *Nat. Rev. Genetics*, vol. 13, no. 5, pp. 303–314, 2012.

[18] M. Li, M. P. Reilly, D. J. Rader, and L.-S. Wang, “Correcting population stratification in genetic association studies using a phylogenetic approach,” *Bioinformatics*, vol. 26, no. 6, pp. 798–806, 2010.

[19] J. K. Pickrell and J. K. Pritchard, “Inference of population splits and mixtures from genome-wide allele frequency data,” *PLoS Genetics*, vol. 8, no. 11, p. e1002967, 2012.

[20] M. Lipson, P. Loh, A. Levin, D. Reich, N. Patterson, and B. Berger, "Efficient moment-based inference of admixture parameters and sources of gene flow," *Mol. Biol. Evol.*, vol. 30, no. 8, pp. 1788–1802, 2013.

[21] Y. Xu, V. Olman, and D. Xu, "Clustering gene expression data using a graph-theoretic approach: An application of minimum spanning trees," *Bioinformatics*, vol. 18, no. 4, pp. 536–545, 2002.

[22] S. E. Schaeffer, "Graph clustering," *Comput. Sci. Rev.*, vol. 1, no. 1, pp. 27–64, 2007.

[23] O. Grygorash, Y. Zhou, and Z. Jorgensen, "Minimum spanning tree based clustering algorithms," in *Proc. 18th IEEE Int. Conf. Tools Artif. Intell.*, pp. 73–81, Arlington, VA, Nov. 2006.

[24] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *J. Roy. Statist. Soc.: Series B (Statist. Methodol.)*, vol. 63, no. 2, pp. 411–423, 2001.

[25] C. Amornbunchornvej, T. Limpiti, A. Assawamakin, A. Intarapanich, and S. Tongsima, "Iterative neighbor-joining tree clustering algorithm for genotypic data," in *Proc. 21st Int. Conf. Pattern Recognit.*, pp.1827–1830, Tsukuba Sci. City, Japan, Nov. 2012.

[26] K. E. Holsinger and B. S. Weir, "Genetics in geographically structured populations: Defining, estimating and interpreting f_{st}," *Nature Rev. Genetics*, vol. 10, no. 9, pp. 639–650, 2009.

[27] X. Gao and E. Martin, "Using allele sharing distance for detecting human population stratification," *Human Heredity*, vol. 68, no. 3, pp. 182–191, 2009.

[28] O. Gascuel and M. Steel, "Neighbor-joining revealed," *Molecular Biol. Evol.*, vol. 23, no. 11, pp. 1997–2000, 2006.

[29] W. Fitch and E. Margoliash, "Construction of phylogenetic trees," *Science*, vol. 155, no. 3760, pp. 279–284, 1967.

[30] D. Huson, D. Richter, C. Rausch, T. Dezulian, M. Franz, and R. Rupp, "Dendroscope: An interactive viewer for large phylogenetic trees," *BMC Bioinformatics*, vol. 8, p. 460, 2007.

[31] L. Liang, S. Zöllner, and G. R. Abecasis, "Genome: A rapid coalescent-based whole genome simulator," *Bioinformatics*, vol. 23, no. 12, pp. 1565–1567, 2007.

[32] N. Chinchor, "Evaluation metrics," in *Proc. 4th Message Understanding Conf.*, pp. 22–29, 1992.

[33] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*. London, U.K.: Chapman & Hall/CRC, 1993.

[34] J. W. Kijas, D. Townley, B. P. Dalrymple, M. P. Heaton, J. F. Maddox, A. McGrath, P. Wilson, R. G. Ingersoll, R. McCulloch, S. McWilliam, D. Tang, J. McEwan, N. Cockett, V. H. Oddy, F. W. Nicholas, and H. Raadsma, "A genome wide survey of SNP variation reveals the genetic structure of sheep breeds," *PLoS ONE*, vol. 4, no. 3, p. e4668, 2009.

[35] M. Gautier, D. Laloë, and K. Moazami-Goudarzi, "Insights into the genetic history of french cattle from dense SNP data on 47 worldwide breeds," *PLoS ONE*, vol. 5, no. 9, p. e13038, 2010.

[36] J. Xing, W. S. Watkins, D. J. Witherspoon, Y. Zhang, S. L. Guthery, R. Thara, B. J. Mowry, K. Bulayeva, R. B. Weiss, and L. B. Jorde, "Fine-scaled human genetic structure revealed by SNP microarrays," *Genome Res.*, vol. 19, no. 5, pp. 815–825, 2009.

Tulaya Limpiti (S'02-M'08) received the BS degree (with highest honors) from Northwestern University, Evanston, IL, in 2002 and the MS and PhD degrees from the University of Wisconsin–Madison, Madison, WI, in 2004 and 2008, respectively, all in electrical engineering. In 2008, she was a postdoctoral researcher at the University of Wisconsin–Madison. She is currently an assistant professor at the Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Thailand. Her research interests include statistical signal processing, multidimensional signal representations and their applications in biomedicine, bioinformatics, communication systems, and agricultural technology. She is a member of the IEEE.

Chainarong Amornbunchornvej received the bachelor of engineering degree (with honor) in computer engineering in 2011 and the master's degree in telecommunications engineering in 2013, both from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. He is a scholarship recipient of the National Science and Technology Development Agency Junior Science Talent Project from 2007 to 2012. He has plan to pursue a PhD study in the near future. His research interests include graph theory, data clustering, machine learning, and pattern recognition.

Apichart Intarapanich received the bachelor of engineering degree from Khon Kaen University, Khon Kaen, Thailand, in 1993, the MS degree from Wright State University, Dayton, OH, in 1996, and the PhD degree from the University of Calgary, Calgary, AB, Canada, in 2005, all in electrical engineering. Between 1993 and 1994, he was an engineer with TelecomAsia Public Company. Since 1997, he has been a researcher at the National Science and Technology Development Agency, Pathumtani, Thailand. His current research interests are wireless channel measurement and modeling, and bioinformatics.

Anunchai Assawamakin received the bachelor's degree in pharmacy from Mahidol University, Thailand in 2001. He received the Royal Golden Jubilee scholarship to pursue the MS and PhD degrees in immunology. He had a postdoctoral training at the Biostatistics and Bioinformatics Laboratory, Genome Institute, BIOTEC, Thailand. He is currently a lecturer at the Faculty of Pharmacy, Mahidol University, Thailand. His research interests include bioinformatics, gene mapping of human diseases, systems pharmacology, human population genomics, and medical informatics.

Sissades Tongsima received the bachelor's degree in electrical engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand in 1991. He received the Royal Thai Government scholarship to pursue MS and PhD degrees in computer science and engineering (parallel and distributed computing) and received the degrees in 1995 and 1999 from the University of Notre Dame, Indiana. He had a postdoctoral training at the Centre National de Genotypage, Evry, France, in 2003. He is currently a principal researcher and the head of biostatistics and bioinformatics laboratory, Genome Institute, BIOTEC, Thailand. Internationally, he serves as an executive committee of Asia Pacific Bioinformatics Network (APBioNet) and a steering committee of the Pan Asian Population Genomics Initiative (PAPGI). He is also an associate editor of the *Journal of Human Genetics (JHG)*, Nature Publishing Group. His research interests are on bioinformatics, gene mapping of human diseases, transposable elements, human population genomics, plant genomics, and medical informatics.

PCA-based Informative SNP Selection for Analyzing Population Structure

Tulaya Limpiti
Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang
Bangkok 10520, Thailand
(66)2-329-8324
tulaya.li@kmitl.ac.th

Apichart Intarapanich
National Electronics and Computer Technology Center
Thailand Science Park
Pathumthani 12120, Thailand
(66)2-564-6900 ext. 2533
apichart.int@nectec.or.th

Sissades Tongsima
National Center for Genetic Engineering and Biotechnology
Thailand Science Park
Pathumthani 12120, Thailand
(66)2-564-6700 ext. 5551
sissades@biotec.or.th

ABSTRACT

Phenotypic differences among individuals of the same species are the result of a set of genetic variations which can be observed in the DNA sequence. To conduct a population genetic study, a high throughput genotyping platform such as Single Nucleotide Polymorphism (SNP) array is popularly used to obtain a large set of SNPs for each individual. However, analyzing today's genotypic data can be computationally expensive due to its large size and complexity. Faulty substructure may also be detected if the data is noisy from redundant or non-informative SNPs. Considerable efforts have been done to extract a smaller informative SNP subset that still represents the same intrinsic structure of populations within a data set as the full panel of SNPs. This work describes a foundation of a PCA-based informative marker selection technique. The proposed technique is simple and efficient. It improves upon another spectral analysis technique called PCA-correlated SNPs. A new informativeness score based on a basis function expansion of the SNP variation patterns across individuals is introduced. Such score is computed for each SNP to select a subset of SNPs with the best scores. Using a bovine data set, we demonstrate that our technique is superior to the PCA-correlated SNPs method, which requires accurate rank estimation to perform well. In contrast, our method is robust to the assumed rank of the data. High data representation accuracy is also achieved after a significant reduction of the number of SNPs while retaining information about the underlying population structure from the original data.

CCS Concepts

• **Applied computing~Population genetics** • *Applied computing ~Computational biology* • Mathematics of computing ~Nonparametric statistics

CSBio'16, December 19–22, 2016, Macau, Macao.
© 2016 ACM ISBN 978-1-4503-4794-5/16/12$15.00
DOI: http://dx.doi.org/10.1145/3029375.3029379

Keywords

Structure informative markers; population structure; principal component analysis; single nucleotide polymorphisms.

1. INTRODUCTION

In population genetics, evolutionary forces that differentiate different populations are investigated. The underlying phenotypic distinctions come from allele differences, called genotypes, in the DNA sequence. With an advent of high throughput genotyping, scientists can quickly observe millions of Single Nucleotide Polymorphisms (SNPs) of an individual. When observed at a population level, the samples' variations belonging to one population reveal a unique genotypic population pattern, called population stratification or population structure. By studying population structure, we can understand key mechanisms of evolution that help shape a population.

Population structure impacts various applications ranging from disease association studies in human [4, 5] to livestock breeding program [3]. Such allele frequency differences defined for each population cast back the underlying evolutionary traces [1, 7]. Hence, researchers strive to identify informative SNPs responsible for population substructure that can equivocally describe and/or correct the underlying faulty substructure signals.

STRUCTURE [11] has laid a foundation to study population structure by constructing a Bayesian model that can effectively predict a contribution of founders. Alternatively, to avoid potential unrealistic genetic model assumptions, nonparametric approach based on principal component analysis (PCA) has been introduced by Cavalli-Sforza [6] and made popular by Patterson et al. [10] to detect and identify population structures.

When dealing with large number of SNPs, there are an intensive computational requirement of existing algorithms for population studies, and high genotyping cost. Moreover, genotyping platform errors may introduce small perturbation that could cause spurious patterns. Thus, methods that can identify a smaller set of SNPs containing information about intrinsic population structures, e.g., [9, 12], are appealing. In particular, we are interested in an approach termed *PCA-correlated SNPs* technique [9], which infers these structure informative markers using PCA. The technique is simple and very effective, and has been applied to several population genetic studies including [7, 8]. However, PCA-correlated SNPs requires estimating the rank of data matrix, and the selected set of informative SNPs varies greatly with different assumed ranks. Consequently, the inferred underlying structures are not consistent.

This paper demonstrates how to efficiently select structure informative SNPs as suggested by the basis function expansion in PCA. The proposed algorithm modifies the PCA-correlated SNPs method, and greatly improves the robustness to rank selection. The construction of our method is such that the bases are also orthonormal. The performance improvements over the PCA-correlated SNPs method are illustrated using a previously published bovine data set [2].

2. METHODS

2.1 Selecting structure informative SNPs

Consider the data of M individuals genotyped with L SNP markers in the form of an M x L matrix $\mathbf{X}$. The i^{th} row of $\mathbf{X}$ represents the SNP sequence of individual i. The j^{th} column of $\mathbf{X}$ gives the variation of SNP at location j across all individuals. Typically, we have $M \leq L$. The biallelic SNP representation at each locus is encoded as 0 (homozygous wild type), 1 (heterozygous), or 2 (homozygous mutant). To reveal the structure within the data using PCA, the singular value decomposition is performed so that $\mathbf{X}$ can be written as

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{i=1}^{M} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \tag{1}$$

where $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_M]$ is the matrix containing left singular vectors. The diagonal matrix $\mathbf{\Sigma}$ contains the singular values $\{\sigma_1, \sigma_2, \ldots, \sigma_M\}$ in descending order; $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_L]$ is the matrix of right singular vectors. The construction is such that $\mathbf{U}$ and $\mathbf{V}$ are unitary. Equivalently, $\mathbf{U}$ contains the principal components computed from the sample covariance matrix of $\mathbf{X}$. To observe population structure within the data, it is common that the data is projected onto the first few dominant principal components and visualized or used in subsequent clustering technique of choice.

To gauge whether a particular SNP greatly contributes in shaping the underlying population substructure using the PCA framework, Paschou et al. [9] has suggested that we look at the j^{th} column of $\mathbf{X}$ corresponding to values of the j^{th} SNP across individuals, defined as

$$\mathbf{a}_j = \sum_{i=1}^{M} \sigma_i \mathbf{u}_i v_i^j, \tag{2}$$

where v_i^j is the j^{th} element of $\mathbf{v}_i$.The so-called PCA-correlated SNPs method for identifying a smaller set of SNPs computes the score for SNP j

$$p_j = \sum_{i=1}^{R} (v_i^j)^2, \quad j = 1,\ldots,L \tag{3}$$

and selects the desired number of SNPs with the largest p_j values. The resulting SNP locations are presumably the most informative. In terms of a basis function expansion, PCA-correlated SNPs approximates the column vector $\mathbf{a}_j$ using R bases $\{\sigma_i\mathbf{u}_i$, i = 1, … , $R\}$ and the v_i^j 's are the basis expansion coefficients. The parameter R is the rank of matrix $\mathbf{X}$, i.e., the number of significant principal components. It is observed that the norms of the basis vectors $\{\sigma_i\mathbf{u}_i\}$ usually vary greatly, depending upon the singular value distribution of the data. Consequently, the coefficients v_i^j whose corresponding singular values σ_i are very small do not give significant contributions to $\mathbf{a}_j$. Nevertheless, they have been given equal importance for the score computation. There is also a rank parameter R to be estimated. The error of the selected rank R has an effect on the final selection of SNPs that are deemed informative.

This work presents an improvement on computing an *informativeness score* of each SNP. Starting with the representation in Eq. (2), we select the left singular vectors $\{\mathbf{u}_i\}$ as our bases. Hence, the basis expansion coefficients are $\{\sigma_i v_i^j$, i = 1, …, $R\}$, which are a function of both the singular values and the elements of the right singular vectors. The updated score is now computed as

$$\tilde{p}_j = \sum_{i=1}^{R} (\sigma_i v_i^j)^2, \quad j = 1,\ldots,L\,. \tag{4}$$

Notice that the bases $\{\mathbf{u}_i\}$ are orthonormal. This is a nice property in the case where the basis expansion coefficients are unknown and need to be estimated. The singular values appropriately weight the contribution of v_i^j in column j in the same manner as the right singular vectors $\mathbf{v}_j$'s have been weighted for constructing the original data matrix $\mathbf{X}$.

2.2 Representation accuracy

It is desirable that, even with much fewer SNPs, the new data matrix retains the underlying population structure. To investigate the results using k principal components, we denote the matrix of k left singular vectors from the original data matrix corresponding to the k largest singular values as $\mathbf{U}_k = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k]$. A new M x P data matrix $\tilde{\mathbf{X}}$ with reduced dimension is formed by keeping only P columns of $\mathbf{X}$ corresponding to P largest $\tilde{p}_j$ values. The principal components of the new data matrix are computed from

$$\tilde{\mathbf{X}} = \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^T \tag{5}$$

Similarly, we define $\tilde{\mathbf{U}}_k$ as the left singular matrix $\tilde{\mathbf{U}}$ with only the first k columns. Using the same number of significant principal components, the structure representation accuracy is defined as

$$\gamma(k) = \frac{\text{trace}\{\mathbf{U}_k^T \tilde{\mathbf{U}}_k \tilde{\mathbf{U}}_k^T \mathbf{U}_k\}}{\text{trace}\{\mathbf{U}_k^T \mathbf{U}_k\}} \tag{6}$$

This measures the fraction of signal energy captured by the first k principal components of the original data matrix that can be represented using the k dominant principal components of the reduced data matrix. Ideally, we would like $\gamma(k)$ to be as close to 1 as possible.

3. RESULTS

Due to space limitation, we only illustrate the performance of our method using a single data set. The experiment is conducted using a bovine data set of 230 individuals [2]. The data is from 9 breeds of cattle genotyped at 8781 SNPs. This data set is chosen because the structure in bovine data set is much more evident than the human data set, which usually have complex structures. We would like to minimize the effects of noise or obscurity in data structure when assessing the performance of our technique. However, note that data sets with smaller numbers of SNPs are more challenging to analyze since there are less information.

The population data is represented by an M x L matrix, where M=230 and L=8781. In order to eliminate the effect of genetic drift and amplify structures within the data, we normalize it

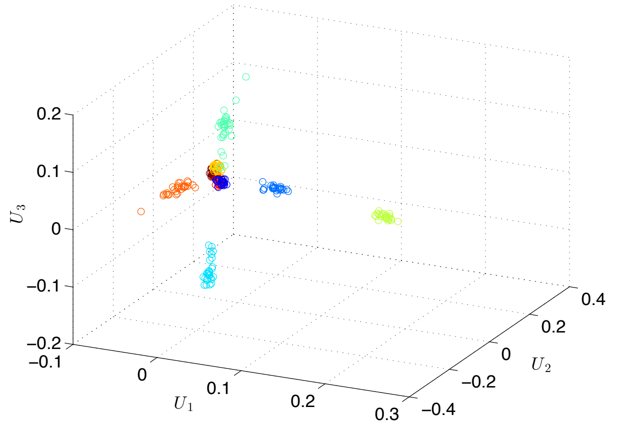

Figure 1. Structure within the bovine data set using full set of SNPs (L = 8781) and three dominant principal components.

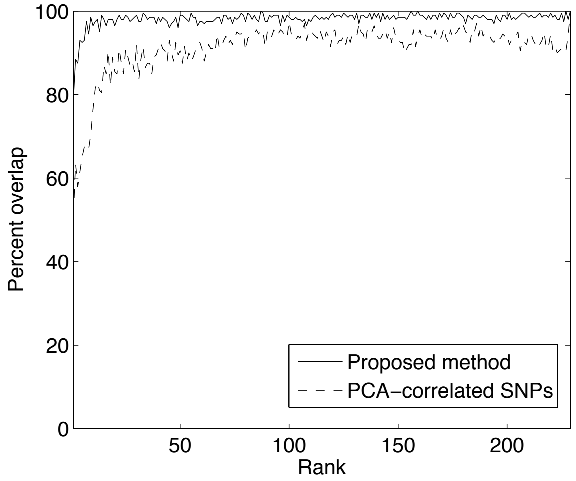

Figure 2. Percent overlap of selected SNP markers between successive ranks.

according to [10] so that each column is zero-mean with unit variance. Effectively, the full rank of the data matrix equals *M-1* = 229.

In order to visualize the population structure within the data, k=3 is used in this paper. The PCA analysis for population structure of the original data matrix (full 8781 SNPs) using three principal components is shown in Fig. 1 where each breed is color-coded. The population structure within the data is obvious, with individuals from five out of nine breeds formed nicely separated clusters. Individuals from the other four breeds are conglomerated in the middle.

3.1 Robustness of informative SNPs selection

Since both the PCA-correlated SNPs scores in [9] and the informativeness scores of our proposed method are computed from $R < M$ basis vectors, we investigate the effect of rank selection in the score computations on the selection of structure informative SNPs. To do this, the scores in Eq. (3) and (4) are computed with rank R varies from 1 (using only the first dominant principal component) to the full rank of *M-1* (using all principal components). For each R, two sets of 200 informative SNPs are selected from the largest p_j and $\tilde{p}_j$ scores, respectively. The percent overlaps between the selected SNP loci for successive values of R are computed, as shown in Fig. 2.

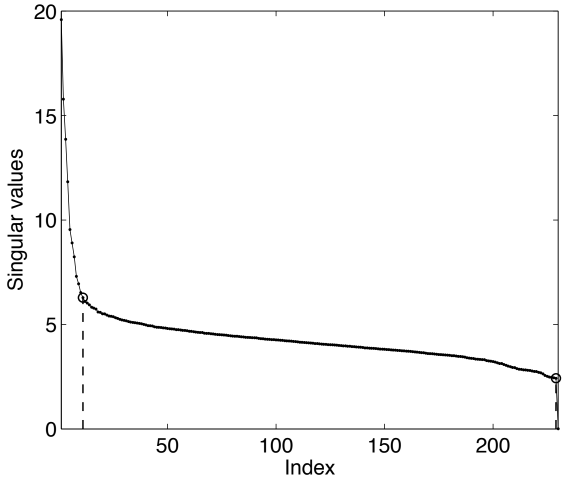

Figure 3. Singular values of the bovine data set.

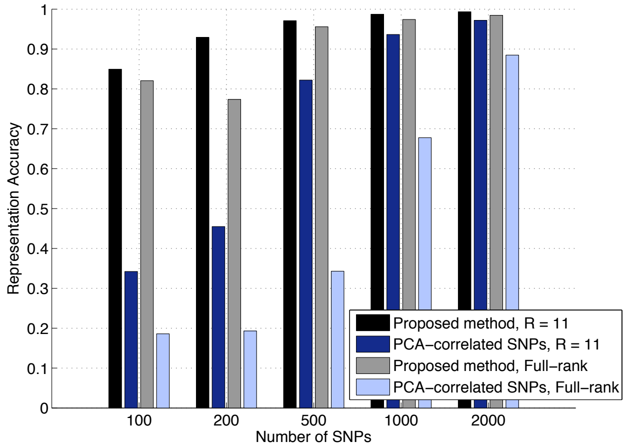

Figure 4. Data representation accuracy.

The percent overlaps for the PCA-correlated SNPs method in [9] are lower than our method. Particularly for small values of R *(*R < 9*)*, the overlaps are between 60-70%. This implies that if there were an error in estimating the rank of data matrix, even if we are off by one rank, it would give a resulting set of SNPs that are vastly different. In contrast, our method is fairly lenient to the chosen value of R. It is seen that more than 93% of markers are similar after R = 3, and the similarities are on average at around 98% with R > 5. Hence, the proposed method for selecting informative SNPs is very robust to the assumed rank of the data matrix. We may use the full rank *R=M-1* (or *R=M* without the normalization) to compute $\tilde{p}_j$ and eliminate the need for rank estimation completely. Otherwise, a low rank-R approximation with $R << M$ can be used with negligible difference. Although not shown here, this trend replicates with larger sets of SNPs.

3.2 Structure representation accuracy

We compare the structure representation accuracy of our technique with the PCA-correlated SNPs method for low-rank and full-rank basis expansions. In order to estimate R, the singular values of the bovine data is plotted in Fig. 3 in descending order.

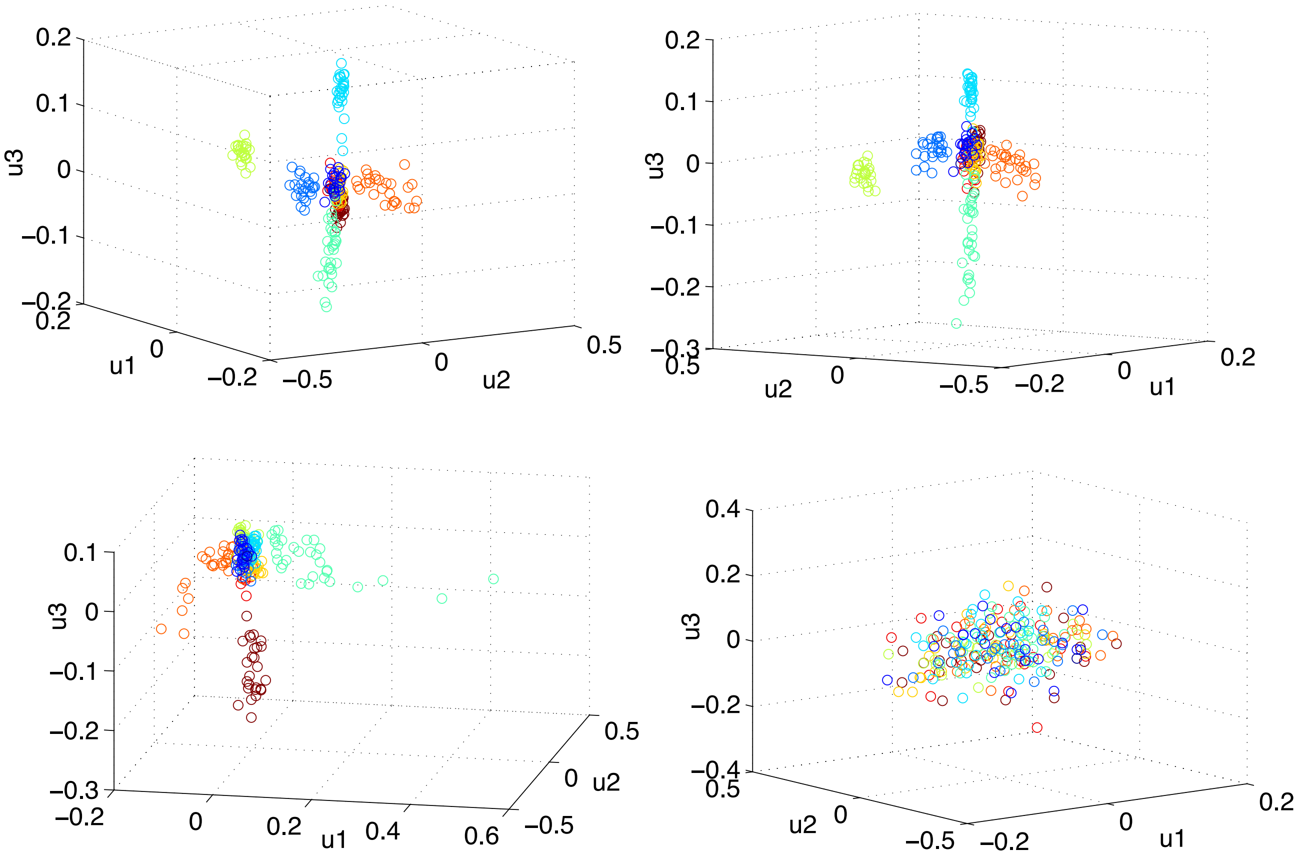

Figure 5. Structures within the bovine data set using selected sets of 200 SNPs. (a) Proposed method, R = 11. (b) Proposed method, full-rank. (c) PCA-correlated SNPs, R = 11. (d) PCA-correlated SNPs, full-rank.

The first principal component explains 7.44% of the variance within the data. We notice that there is a sharp drop of the singular values before tapering off. We identify the bend of the graph, which signifies the changing point in the singular value contributions (depicted in Fig. 3 by the dotted line), using gradients of the singular values. This corresponds to the point where the gradient is less than 5%, which occurs at the 11th singular value. So we choose R = 11 for the low-rank basis expansion of the data matrix in our subsequent analysis. These eleven dominant principal components account for 26.9% of the variance within the data. Each of the remaining principal components contributes only 0.34% on average. For the full-rank counterpart, we use $R = M$-1.

Although we have not tried to estimate the rank of the data matrix with the technique used in [9], we observe that the selected ranks therein always equal or are close to the number of the underlying populations within the data. We anticipate that for our bovine data set, the rank estimated by the original PCA-correlated SNPs method would be close to 9, so using R=11 is not unreasonable.

Fig. 4 depicts the values of $\gamma(3)$ for the numbers of SNPs ranging between 100 to 2000 markers. The data representation accuracy of our method is superior to the PCA-correlated SNPs method for both low-rank and full-rank results. When R=11 is used to compute the scores, the representation accuracy of our method is greater than 0.85 when we use only 100 SNPs, or just slightly over 1% of the total number of available SNPs. However, PCA-correlated SNPs selects 100 SNPs that can capture only 34% of signal energy ($\gamma(3)$= 0.34). With our proposed modification, we achieve over 0.93 accuracy with merely 200 SNPs. The PCA-correlated SNPs technique reaches the same representation accuracy using 1000 SNPs. At 2000 SNPs, or 23% of the original dimension, both methods perform well with the accuracies of 0.99 and 0.97 for our method and PCA-correlated SNPs, respectively.

For full-rank results, the representation accuracies decrease slightly for our method when small sets of 100 and 200 SNPs are used. This is because more bases representing “noise” are included in the score computation. The accuracies are comparable to the low-rank results when the number of SNPs is greater than 500, as more SNPs provide more structure information. In contrast, the degradation in representation accuracy is more substantial for PCA-correlated SNPs. This is a direct consequence of its rank-dependency and improper weighting of basis coefficient v_i^j as discussed earlier.

3.3 Visualizing population structures

The population structures within the bovine dataset using two sets of 200 informative SNPs selected with our technique and the PCA-correlated SNPs technique are visualized on three dominant principal component axes in Fig. 5. Again we compare the full-rank and the low-rank results. Regardless of the rank, the population structure within the original data in Fig. 1 is correctly retained using our proposed method. Separations of individuals from the same five breeds are still noticeable, although the individuals are slightly more dispersed.

For low rank, PCA-correlated SNPs produces a structure that differs from the original. Individuals from three breeds are separated from the remaining breeds. However, only two breeds are similar to those seen in Fig. 1. The structure also changes drastically for the PCA-correlated SNPs full-rank result. No visible structure can be detected.

4. CONCLUSIONS

We modify the PCA-correlated SNPs technique for identifying structure informative SNPs by improving the calculation of the informativeness score for each SNP and select a small subset of SNPs with the best scores. The proposed technique is simple and efficient. It is demonstrated that the result is robust to the assumed rank of the data, i.e., the choice of a rank estimation technique has little effect on the final selection of informative SNPs. In fact, rank estimation may be bypassed with negligible degradation in data representation accuracy. Additionally, sizable dimensional reduction can be achieved while retaining information on the underlying population structure from the original data.

For an extension of this work, we plan to look at the performance of our method on human data sets with varying complexities, including the ones used in [8, 9]. We believe that our technique is advantageous in the cases where we want to study the population structure at a finer scale, e.g. populations within continents or with common ancestry.

5. ACKNOWLEDGMENTS

The work of T. Limpiti is supported by the Thailand Research Fund (TRF) new researcher grant TRG5780045. S. Tongsima is supported in part by the TRF mid-career research grant RSA5860081.

6. REFERENCES

[1] Cavalli-Sforza, L. and Feldman, M. 2003. The application of molecular genetic approaches to the study of human evolution. *Nat. Genet.* (Mar. 2003) 33, 266–275.

[2] Gautier, M., Laloë, D., and Moazami-Goudarzi, K. 2010. Insights into the genetic history of french cattle from dense SNP data on 47 worldwide breeds. *PLoS ONE 5*, 9 (Sep. 2010), e13038. DOI= http://dx.doi.org/10.1371/journal.pone.0013038

[3] Karimi, K., Strucken, E., Moghaddar, N., Ferdosi, M., Esmailizadeh, A., and Gondro, C. 2016. Local and global patterns of admixture and population structure in Iranian native cattle. *BMC Genet.*, 17 (Jul. 2016), 108.

[4] Lander, E. and Schork, N. 1994. Genetic dissection of complex traits. *Science*, 265 (Sep. 1994), 2037–2048.

[5] Marchini, J., Cardon, L. R., Phillips, M. S., and Donnelly, P. 2004. The effects of human population structure on large genetic association studies. *Nat. Genet.*, 36, 5 (May 2004), 512–517.

[6] Menozzi, P., Piazza, A., and Cavalli-Sforza, L. 1978. Synthetic maps of human gene frequencies in Europeans. *Science*, 201, 4358 (Sep. 1978), 786–792.

[7] Paschou, P., Drineas, P., Lewis, J., Nievergelt, C., Nickerson, D., Smith, J. et al. 2008. Tracing sub-structure in the European American population with PCA-Informative markers. *PLoS Genet.*, 4, 7 (Jul. 2008), e1000114. DOI= http://dx.doi.org/10.1371/journal.pgen.1000114

[8] Paschou, P., Lewis, J., Javed, A., and Drineas, P. 2010. Ancestry informative markers for fine-scale individual assignment to worldwide populations. *J. Med. Genet.*, 47, 12 (Dec. 2010), 835–847.

[9] Paschou, P., Ziv, E., Burchard, E., Choudhry, S., Rodriguez-Cintron, W., Mahoney, M. and Drineas, P. 2007. PCA-correlated SNPs for structure identification in worldwide human populations. *PLoS Genet.*, 3, 9 (Sep. 2007), 1672–1686. DOI= http://dx.doi.org/10.1371/journal.pgen.0030160

[10] Patterson, N., Price, A., and Reich, D. 2006. Population structure and Eigenanalysis. *PLoS Genet.*, 2, 12 (Dec. 2006), e190. DOI= http://dx.doi.org/10.1371/journal.pgen.0020190

[11] Pritchard, J. K., Stephens, M., and Donnelly, P. 2000. Inference of population structure using multilocus genotype data. *Genetics*, 155 (Jun. 2000), 945–959.

[12] Rosenberg, N., Li, L., Ward, R., and Pritchard, J. 2003. Informativeness of genetic markers for inference of ancestry. *Am. J. Hum. Genet.*, 73, 6 (Dec. 2003), 1402–1422. DOI= http://dx.doi.org/10.1086/380416