Local Ancestry Inference in Admixed Populations

Naveen Arivazhagan
Department of Computer Science
Stanford University
Stanford, CA 94305, USA
Email: naveen67@stanford.edu

Hye Ji Kim
Department of Electrical Engineering
Stanford University
Stanford, CA 94305, USA
Email: hyejikim@stanford.edu

Edwin Yuan
Department of Applied Physics
Stanford University
Stanford, CA 94305, USA
Email: edyuan@stanford.edu

I. Introduction

Contemporary human sub-populations exhibit great differences in the frequency of various alleles, or the set of variations of a particular gene. Advances in genome sequencing have rapidly improved speed, cost, and accuracy, allowing unprecedented opportunity to map the functionality and location of such genetic variation. Of particular interest is the mapping of disease associated loci in the genome. In admixed populations, or populations that are the result of a mixing of two or more ancestral populations over a small number of generations, one technique that has been used extensively is mapping by admixture linkage disequilbrium (MALD). The rationale behind MALD is that disease-affected individuals in the admixed populations should share higher levels of ancestry near disease-carrying loci with the ancestral population from which the disease loci arose. The accuracy of MALD thus depends crucially on the accuracy with which one can infer the ancestry around any loci in an admixed genome. This particular task has been termed Local Ancestry Inference (LAI).

Much of the early work in local ancestry inference took form around the assumptions of hidden markov models. While this method is computationally efficient, the markov assumptions fail to model the correlation in inheritance between base pairs, or linkage disequilibrium (LD). Later models developed at Stanford, such as SABER [1] and HAPAA [2], explicitly model linkage disequilibrium within the HMM framework, by extending the dependence of the posterior probabilities to previous states even further behind in the chain. In doing so they are also computationally expensive. A later approach LAMP [3], utilized probability maximization within a sliding window, assuming that only one recombination event took place within each window. This is based on the fact that, biologically, ancestral haplotypes are inherited in blocks of alleles, and thus between any two blocks there is a single recombination event. LAMP is considered amongst the gold standard for LAI in recently admixed populations, such as the Chinese and Japanese.

In 2013, the 1000 Genomes Projects Phase I released a data set of 1092 individuals from 14 populations that was unprecedented in its detail, genomic completeness, and scope. Soon after, Maples, Gravel, Kenn, and Bustamante released a discriminative model (uses p(Y|X) as opposed to p(x,y)) called RFMix [4] which uses conditional random fields (CRF) based on random forests within windowed sections of the chromosome. RFMix was shown to be both faster and more accurate than the existing LAMP method. The modern challenges for local ancestry inference are: efficiency in light of increasingly large and dense genomic data sets, discrimination between recently divergent ancestries, and overall algorithm accuracy.

Understanding Ancestry Inference As the previous section illustrates, there is great variety in nature of the algorithms implemented for global ancestry inference, with different levels of performance depending on the admixing scenario in question. Fundamentally, there are two approaches to this problem of ancestry inference. The first takes an entirely non-biological approach, treating this task as one analogous to identifying which ancestry a particular sequence of nucleotide letters is most statistically related to. The second approach is highly motivated by the biology of the genome, attempting to incorporate mechanisms for recombination, mutation, etc. Most models in the field have been of the second type.

To explain in detail how these algorithms work in general, we take RFMix as an example. In the most basic framework, RFMix segments an input strand of DNA (a sequence of SNPs) from an admixed individual into contiguous windows of single nucleotide polymorphisms (SNPs) and then assigns each of these windows of SNPs to one of several reference ancestries. This is shown in Figure 1. The statistics for determining the SNPs ancestry come from a training set of reference panels, which are entire sequences of SNPs that have been globally assigned to one of the ancestries in consideration.

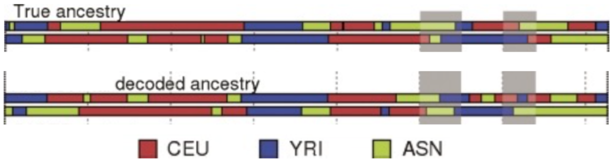

Fig. 1. **Illustration of Ancestry Inference Problem** [2] Two admixed chromosomes are shown with the true ancestries above and with the decoded ancestries below. The admixed individuals are mixtures of 3 ancestral populations.

RFMix has a second functionality. The previous approach is fast and works well when one has an abundance of reference panels. This is, however, not typically the case because despite large organized efforts of HapMap and the 1000 genomes project, publicly available population-size data sets remain sparse. The admixed samples, on which RFMix is tested, itself contains ancestry information of SNPs, albeit in a jumbled form. RFMix thus is able to iteratively infer and then incorporate ancestry information from the admixed (test) samples using EM.

Finally, RFMix models phase errors that are produced as part of the local ancestry inference and then attempts to autocorrect these errors. In an example simulation the paper provides, it was shown that, by this procedure, RFMix significantly improves long-range phasing error. By comparing the fraction of SNP pairs correctly phased relative to each other, the new phasing generated by RFMix achieved 75$ on this metric, compared to 50% achieved by the original Beagle phased data. Beagle is a standard phasing algorithm that uses haplotype clustering methods.

II. Data and Features

We were able to utilize some pre-processed data that the authors of the RFMix paper provided. The data set consists of 51213 SNPs from both chromosome ones of 362 individuals. The SNPs were assumed to be bi-allelic. The test set consists of 10 admixed, Latino individuals, whose genomes were created using a Wright-Fischer simulation to sample 12 generations after admixture. The simulated Latino genomes were generated from existing data sets and have 45% Native American (NAT), 50% European (HapMap CEU), and 5% African (Yoruba in Ibadan) ancestry. Other simulated samples were used to construct genomes of the reference panels, of which there were 170 Native American (NAT), 194 African (YRI), and 340 European (CEU). The SNP's used were created to be perfectly phased, and so untangling phasing error was not a part of the following analysis.

A. *Principal Component Analysis*

Despite the high dimensionality of the data set, with each training example containing 51213 SNP's, the 3 separate ancestries, Native American, African, and European could very easily be distinguished by a 2-3 component principal component analysis, shown in Figure 2. The yellow admixed ancestries indeed lie between the 3 ancestral populations in the principal component space. The yellow admixed individuals show much larger variation within the group compared to any of the ancestral populations. PCA also shows graphically, as expected, that the admixed group as a whole is closer to Native American and European ancestries than to African. This is expected given that the admixed individuals are on average only 5% African.

III. Methods

We use a pipeline approach consisting of two steps. In this first step we identify 'windows' : sections of the genome that are believed to be highly correlated to each other and therefore tend to be inherited together. Since they are inherited together, they will have the same population ancestry. Therefore, in our second step we classify the windows that we have identified into one of the 3 source populations.

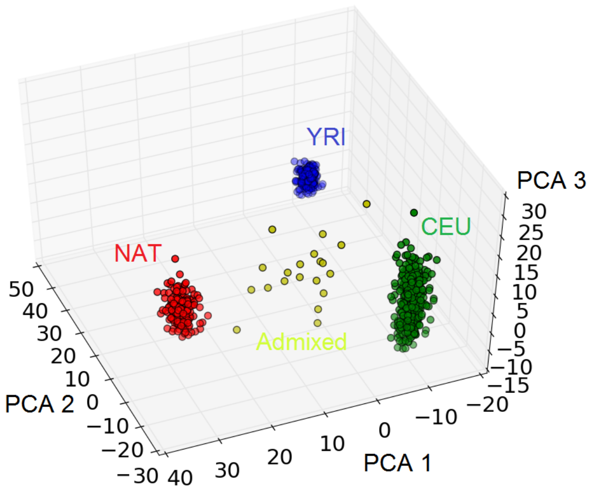

Fig. 2. **PCA** The full training set of 85 Native American (NAT, red), 97 African (YRI, blue), and 170 European (CEU, green) individuals projected onto the first 3 principal axes. The 10 admixed individuals are shown in yellow.

In our paper we use a simple heuristic for identifying the windows. We divide the chromosome into windows of fixed centi-morgans. By the definition of a centi-mogran, there is thus a variable number of SNPs per window, but they are grouped according to the average number of chromosomal crossovers expected within the group.

Having identifies the windows, for the second step, we use a variety of classifiers to correctly classify the window of an admixed genome into the correct population ancestry based on its similarity with the corresponding windows from the reference panel.

The full training set consists of 85 Native American (NAT), 97 African (YRI), and 170 European (CEU) individuals. A more moderate and realistic training set consists of 30 Native American (NAT), 30 African (YRI), and 30 European (CEU) individuals. Finally, the extreme case in which one has a scarcity of well-sequenced reference panels is represented by 3 training examples of each ancestry, 3 Native American (NAT), 3 African (YRI), and 3 European (CEU) individuals. In reality, the possibility of having such large cohorts of, accurately sequenced data is unlikely given modern sequencing technologies. There is also increasingly a push to move beyond the heavy reliance on reference panels in order to perform ancestry inference.

A. *Manhattan Method*

The first classification method we implemented was a simple criteria of determining how closely related two sequences of nucleotides are. We devised a notion of similarity between windows in the reference samples and those in the admixed samples by counting the number of replacements needed to get from one window to another. For example if in some reference window one has 0 0 1 0, and in an admixed window, 0 1 1 0, the number of replacements needed is just one. The fewer replacements needed to convert between the sequences the more similar they are. This is the gist of the so-called Manhattan metric. We identify the windows amongst the reference panels to which the admixed window has the highest similarity. We then use a voting scheme where the ancestry of the admixed window is assigned to the reference population in which it has largest number of the highest similarity values.

A result of the algorithm labeled ancestry when the window size=2 cM is shown in Figure 3 below for the entire length of one chromosome of one admixed individual:

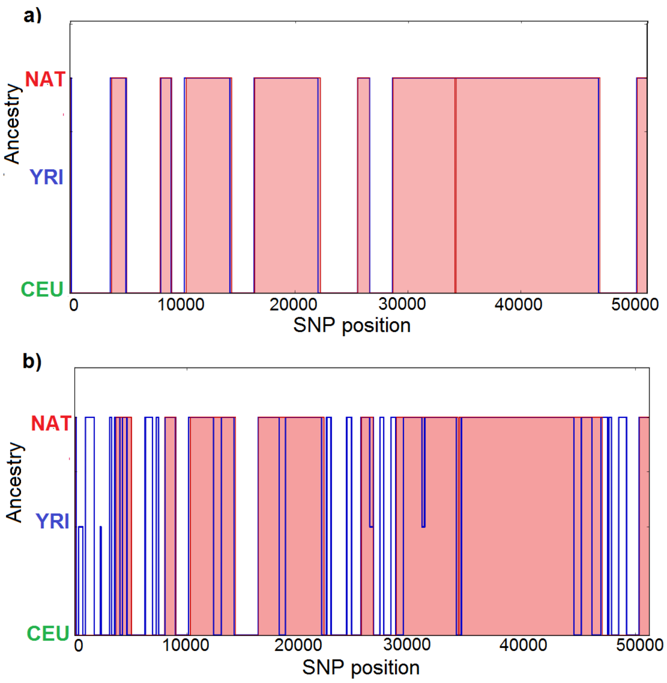

Fig. 3. **Manhattan Method Labeling** Plots showing the ancestry labels of each of the 51231 SNP's under consideration for windows of 2 cM of a single admixed chromosome. The red bars show the true ancestry while the blue over-layed lines show the ancestry predicted by our Manhattan algorithm a) Compares the ancestry labels when the algorithm is trained on a set of 30 individuals of each ancestry b) shows the same when trained on a set of 3 individuals of each ancestry. Note that there are no SNPs inherited from YRI simply because the admixed genome under review doesn't have any.

In Figure 3a, we see that, when trained on moderately large data sets of 30 individuals of each ancestry, the Manhattan method is extremely accurate at predicting ancestry. The haploblocks are large and the Manhattan method finds the correct label but only up to small shifts. It similarly misses changes in the ancestry that occur over just a few SNP's. The overall accuracy here of the Manhattan method is around 96.5% compared to 97.5% achieved by RFMix.

On the other hand, the algorithm performs much more poorly when training on a smaller data set of only 3 individuals. Although the overall accuracy of labeling is still relatively high at 81.18%, it's clear from Figure 3b that the Manhattan method does very little to infer the overall shape of the haploblocks. RFMix achieves 87.8% accuracy but can iteratively incorporate the admixed predictions into EM to boost performance.

Varying window size Because inheritance of genes takes place through haploblocks, each of a single ancestry, choosing the correct window size is essential for achieving optimal performance. The result of varying window size on overall accuracy is shown in Figure 4 when using the full training set.

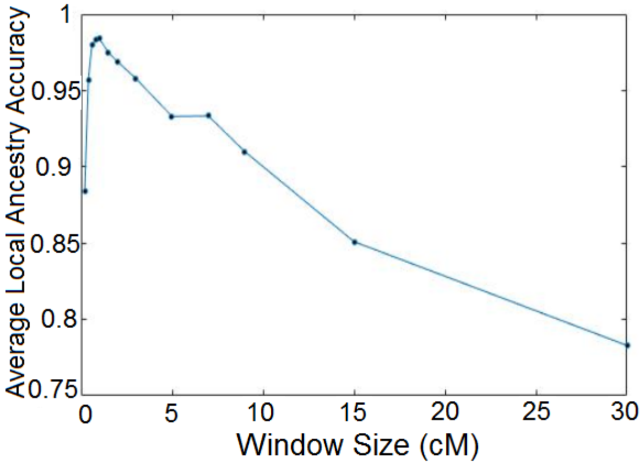

Fig. 4. **Manhattan Method Window Size** After training on the full reference panels, the overall accuracy of the Manhattan algorithm is shown as a function of window size

The results suggest very high performance, compared to RFMix, peaking at around 98.4% accuracy for a window size of 1.0 cM. For the same window size RFMix uses, 0.2 cM, the accuracy is only 88.4%. As window size is increased, the accuracy peaks and then falls rapidly. It is important here to keep in mind that the benchmark accuracy, that achieved by random guessing, is already 33.3%, given that we have 3 ancestral populations.

B. *Support Vector Machine*

As a point of comparison we also applied a support vector machine (SVM) classifier to our data. Again we take the approach of fixed window size and use the SVM on training data to classify vectors with length equal to the number of SNP's that exist within each window. The results below are trained on the full set of reference panels. We find that the performance of the SVM depends significantly on parameters like the type of kernel employed, the window size, and the value of an internal parameter C which is explained below. Figure 5 compares the overall accuracy of the SVM using a linear kernel versus that of one using the radial basis kernel for different window sizes.

It is evident from the figure that the linear kernel outperforms the radial basis kernel (with the default parameters) at all plausible window sizes. To investigate this further we note that the SVM with the radial basis kernel depends on two parameters, namely C and gamma, we vary the parameters,

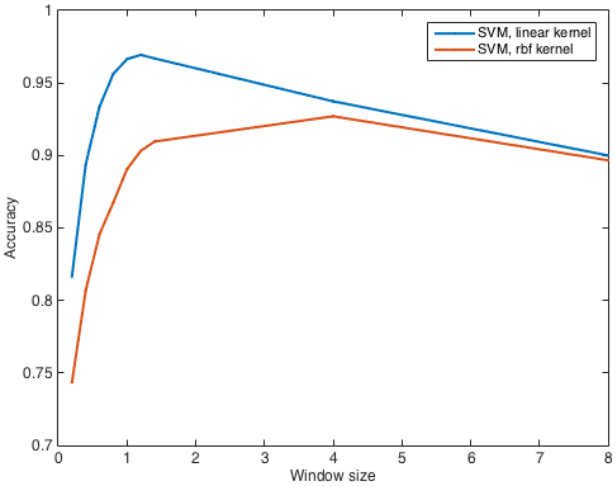

Fig. 5. **SVM Accuracy vs. window size** A plot of the overall accuracy achieved as a function of the fixed window size (centi-morgans) used, for the radial basis kernel and the linear kernel. In both SVM's are trained on the full set of reference panels.

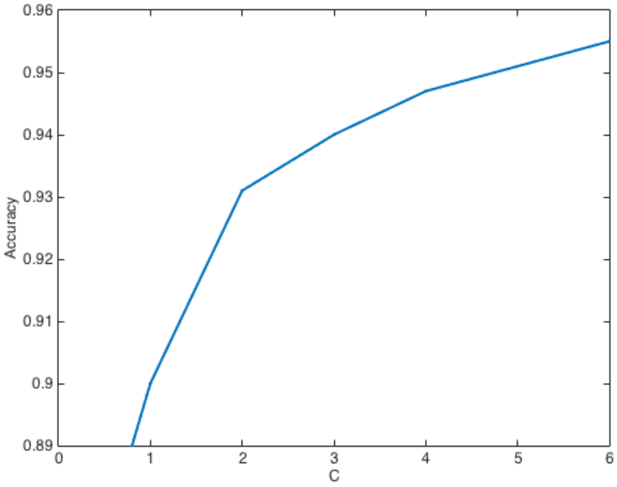

Fig. 6. **SVM Accuracy vs. parameter 'C'** For an SVM using the radial basis kernel, we plot the overall accuracy as a function of the internal SVM parameter 'C', whose function is also explained below. The training set is the full set of reference panels.

fixing the window size at 1.4 cM. For the radial basis kernel, $K(x, y) = \exp^{-\gamma|x-y|^2}$. Gamma thus determines how much weight to put on a single training data for a given euclidean distance between that single training data and the test data. The larger γ is, the more weight placed on training data that are closer in distance to the test data.

We found that as we decreased gamma, the accuracy increases. For $\gamma = 0.3$, the accuracy is 96%, and for $\gamma = 0.15$, the accuracy is 97.2%). We conjecture that this takes place because when gamma is smaller, more of the training data is taken into account. Another factor of consideration is that the Euclidean metric may not be the best indicator of how far a given test point is from a training data point. The discrete Manhattan distance may characterize the notion of distance between two sequences more functionally and increase classifier performance.

Another large determinant of the SVM's performance is the value of the internal parameter C. The parameter C controls the tradeoff between classification correctness on the training data and the largeness of the largest minimal margin. A large value of C indicates a willingness to increase the classifier's accuracy rating by giving weight to outlier training examples that are quite far from the mean of the data. In a general sense, a large C value tolerates overfitting behavior. As expected, as we increase C, the accuracy of the SVM increases as shown in Figure 6. Here we are using the radial basis kernel while training on the full reference panels.

Finally, we evaluated the performance of our SVM's using only small numbers of training data. We first tested with 30 training examples from each ancestry, and in that case, we get about 96% overall accuracy for the SVM with the linear kernel and the SVM with the radial basis kernel. This is quite a small reduction from the SVM performance on the full set of training data, and indicates that in the regime of large reference panels we are gaining very little performance by adding more panels. On the other hand, when we evaluate the performance using the extreme scenario of 3 training reference panels from each group, we achieve 76% accuracy using the SVM with radial basis and 82% accuracy using the SVM with linear kernel. Again, the linear kernel yields superior performance to the radial basis kernel.

C. Random Forest

We use an ensemble of trees to make prediction on the windows. The random forest generates multiple decision trees and take the average vote to predict the label of test data. As the number of decision trees increase, the accuracy increases, but as a drawback, the run time also increases. We also observe that increasing the number of trees does not tend to cause overfitting easily.

We then tested the random forest method using only three training examples. We set the number of estimators as $\sqrt{window\ length}/N_c$ and vary N_c from 3 to 0.05. For $N_c = 3$, we get 69% accuracy, and for $N_c = 0.05$, we get 80% accuracy.

In Figure 7, we plot the accuracies of Manhattan method, SVM method, and the random forests method as a function of the window size (we use all the training data.) We see that for all methods, the accuracies peak at around window size 1cM. For a smaller number of window size (including 1cM), Manhattan method and random forest method perform better than SVM.

D. Hidden Markov Model

We use hidden Markov model. State i is the ancestry (African, European, etc) at the i-th position of a haplotype,

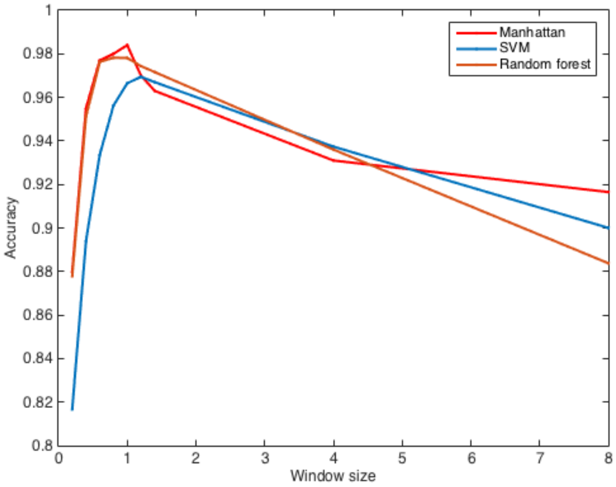

Fig. 7. Accuracy of Manhattan method, SVM method, and the random forests method vs. window size

and the observed variable is SNP at the i-th position of a haplotype.

The HMM requires three probability matrices. One is the probability of each hidden state, and the second is the emission probability, and the third is the transition probability from one state to another state.

For the first probability, we assume every ancestries are equally likely. Secondly, to estimate the probability of emission probability of i-th state, we use the empirical probability in the reference haplotypes. Note that this emission probability is not stationary,i.e., it depends on i. Lastly, we assume with probability 0.9, there is a transition from one ancestry to another ancestry at time i, and for the remaining probability, there is a transition to a new ancestry with equal probabilities.

Using this approach we get only get a 0.506% accuracy. This is because the HMM does not pay any attention to the index of the SNP and is therefore not able to capture the distribution of the specific columns in the data. We also notice that the predictions are skewed n favor of population 3 because of its high start probability and the low transition probabilities.

IV. Error Analysis

We describe two independent sources of error in not just our classifiers, but also other more complex local ancestry inference algorithms:

1) Windowing of the SNP's. In perfect windowing, all SNP's within a window originate from not just the same ancestry but also the same ancestral individual within a population. If SNP's from two different ancestries fall within the same window, then we will inevitably misclassify one of the two segments within that window. Alternatively, even if a window contained two segments from the same ancestry, but from different people, any similarity measure may fail. Because said similarity measures only compare a given test window against the corresponding training window of a single individual, the classification will not be ideal, and may lead to errors.
2) Assuming now, that windowing is correct, an independent source of error is classification error within a given window. This error exists because in a real world scenario, the reference panels used to train the classifier are not directly ancestors of the admixed individuals.

Classifier	train-size=3	train-size=30	full-train
SVM	0.99	0.99	0.99
Random Forest	0.85	0.99	0.99

TABLE I
The performance of SVM, and random forests when evaluated on the true windows

Classifier	train-size=3	train-size=30	full-train
SVM	0.78	0.96	0.97
Random Forest	0.84	0.97	0.97

TABLE II
The performance of SVM, and random forests when evaluated on the the heuristic based windows

We sought to investigate whether the majority of error in our simulated data came from the first or the second source. To this end, we used the true windows of an admixed individual while training the classifier, instead of the fixed centimorgan window sizes we had been using. We then again tested with different classifiers and training sizes. Comparing tables I and II, we find that we can achieve near perfect performance if we are given the correct admixed windows. This is the case even when the number of reference panels is very few, 3 per ancestry. Thus we find that it is in fact the windowing algorithm that is the main bottleneck in our approach and further work should be devoted to this step of the process. Various of the more recently published algorithms such as WinPop take steps to deliberately optimize the search for the best window length at each locus along the chromosome.

V. Concluding Remarks

Our conclusion from running various different learning algorithms, is that a large majority of them work very well (above 95% accuracy) given an abundance of reference panel training data. This is true for even relatively simple algorithms such as the one based on the Manhattan metric. When the number of reference panels is few however, EM is valuable method for iteratively improving performance. Furthermore our error analysis suggests that a large proportion of the error comes from poor choices of windowing. By windowing more ideally, many algorithms can achieve near perfect performance even when reference panels are scarce. Thus developing methods for judiciously choosing window size are an important effort in local ancestry inference.

References

[1] H. Tang, M. Coram, P. Wang, X. Zhu, and N. Risch, "Reconstructing genetic ancestry blocks in admixed individuals," *American Journal of Human Genetics*, 2006.

[2] A. Sundquist, E. Fratkin, C. B. Do, and S. Batzoglou, “Effect of genetic divergence in identifying ancestral origin using hapaa,” *Genome Research*, vol. 18, no. 4, pp. 676–682, 04 2008. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2279255/

[3] S. Sankararaman, S. Sridhar, G. Kimmel, and E. Halperin, “Estimating local ancestry in admixed populations,” *American Journal of Human Genetics*, 2008.

[4] B. K. Maples, S. Gravel, E. E. Kenny, and C. D. Bustamante, “Rfmix: A discriminative modeling approach for rapid and robust local-ancestry inference,” *The American Journal of Human Genetics*, vol. 93, no. 2, pp. 278–288, 2015/12/10. [Online]. Available: http://dx.doi.org/10.1016/j.ajhg.2013.06.020