SoCal: Genotype Calling via Ellipsoidal Separation

Huwenbo Shi (603-778-363) shihuwenbo@ucla.edu

ABSTRACT

In this article, I present SoCal, a supervised genotype calling algorithm for Affymetrix SNP arrays. For each SNP, SoCal first efficiently identify ellipsoidal decision regions for each genotype from reference genotype calls, and then uses these regions to classify future SNPs into different genotypes. Using only a small portion of training genotype calls from the HapMap Project, SoCal achieves an accuracy of 97.5% during validation.

1 Introduction

Single nucleotide polymorphisms (SNPs), genomic positions at which a single nucleotide differs between individuals, are responsible for many phenotypic variations among individuals. To study the associations between SNPs and phenotypes, one must be able to accurately call genotypes for case and control samples at these SNPs. Although next generation sequencing (NGS) technology provides cheap and whole-genome sequences for genotyping SNPs, SNP arrays are still more cost-effective for specific experiments [1].

Affymetrix SNP arrays use oligonucleotide probes to measure intensities I_A and I_B for alleles A and B of SNPs in samples. If a sample has genotype AA or BB for a SNP, one observes higher value of I_A or I_B respectively. For genotype AB, one observes similar values of I_A and I_B. If one plots the data point (I_A, I_B) of a SNP for a number of samples, normally 3 clusters are observed, one for each genotype (Figure 1). The computational challenge is to correctly assign each (I_A, I_B) data point to a cluster and call its genotype accordingly.

Many genotype calling algorithms use model-based unsupervised clustering methods to identify clusters and then assign genotypes to each cluster. For instance, Norlén et al. proposed a Gaussian mixture model for clustering [2]. Lin et al. proposed a statistical model and a modified K-means algorithm to incorporate pedigree information in clustering [3]. These methods use EM algorithm to estimate model parameters, which is sensitive to starting parameters and slow to converge [4].

A few other genotype calling algorithms exploit reference genotype calls. For instance, Rabbee et al. proposed the RLMM algorithm, which estimates allele intensities of SNPs on a chip by fitting a linear mixed model and compares them with the means obtained from training data to call genotypes [5]. This method models probe effect in the

linear mixed model and is able to make more accurate genotype calls. However, fitting a linear mixed model can be computationally intensive, and may involve optimizing a non-convex function [6].

As the number of probes on SNP arrays and the number of individuals involved in association studies continue to increase, both fast and accurate genotype calling algorithms are needed. In this article, I developed a supervised genotype calling algorithm (SoCal) that uses reference genotype calls from HapMap to efficiently find elliptic boundaries separating the three genotypes from one another for each SNP. SoCal then uses these boundaries to call the same SNP in different samples with unknown genotypes in linear time. SoCal can also control outlier effect by using different criteria for finding the separating boundaries, a feature not provided by simply fitting a Gaussian distribution.

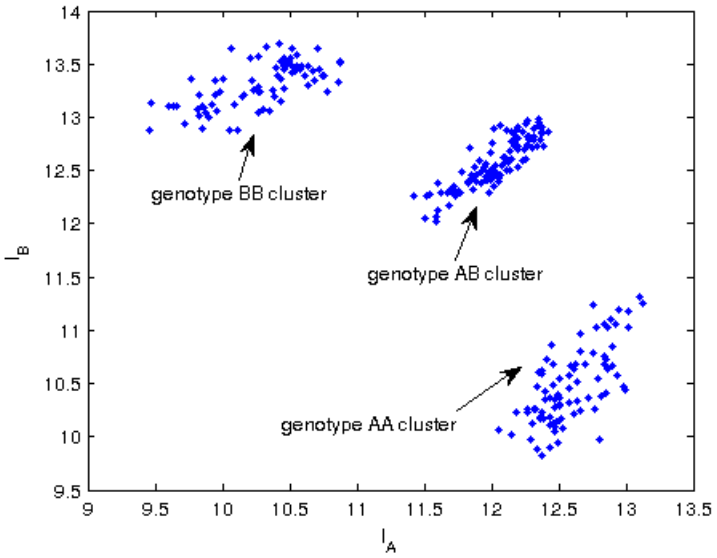


Figure 1: Genotype clusters obtained from Affymetrix SNP array allele intensity values

2 Method

SoCal calls genotypes in two steps. The first step of SoCal is finding decision regions for each genotye using reference genotype calls for a specific SNP. The second step of SoCal is using these decision regions to classify the same SNP into different genotypes for different samples.

2.1 Finding decision regions

Affymetrix SNP arrays generate allele intensity values that roughly follow bivariate Gaussian distributions. Because of outliers in reference genotype calls, simply fitting bivariate Gaussian distributions and then applying likelihood ratio test might not the

best way of classifying genotypes. Instead, SoCal uses ellipses as decision regions, and controls outlier effect by specifying different criteria for finding the ellipses.

An ellipsoid $\mathcal{E} \subseteq \mathbb{R}^n$ can be expressed as $\mathcal{E} = \{x \in \mathbb{R}^n | (x-c)^T E(x-c) \leqslant 1\}$, where c is the center of the ellipsoid, and E a positive semidefinite matrix. Let a_i be the points to be included in an ellipsoidal region, and b_j be the points to be excluded, the problem is to find a c and E that separate a_i from b_j.

For SoCal, the dimension of data is 2 ($n = 2$). a_i correponds to the allele intensity data points of one genotype, and b_j correponds to the allele intensity data points of all other genotypes. For each SNP, SoCal obtains three c's and three E's, one for each genotype.

In the following subsections, three different ways of finding elliptical decision regions are discussed.

2.1.1 Maximal separation ratio method (MAXSEP)

MAXSEP, discussed in detail in [7], tries to find ellipses that separate a_i from b_j with the maximum separation ratio. This problem can be expressed as a conic programming problem:

$$\begin{aligned} &\text{minimize } -k \\ \text{subject to } &(1, a_i)^T \tilde{E}(1, a_i) \leqslant 1 \ \forall i \\ &(1, b_j)^T \tilde{E}(1, b_j) \geqslant k \ \forall j \\ &\tilde{E} \succeq 0. \end{aligned}$$

Let

$$\tilde{E}^* = \begin{bmatrix} s & v^T \\ v & F \end{bmatrix}$$

be the optimal solution to the problem above. The maximum margin separating ellipsoid $\mathcal{E}^*$ is defined as

$$\mathcal{E}^* = \{x \in \mathbb{R}^n | (x - c^*)^T E^* (x - c^*) \leqslant 2(1 + k)\},$$

where

$$c^* = -F^{-1}v, \ E^* = \frac{F}{(1 - s + c^{*T} F c^*)}.$$

Intuitively, the first constraint in the problem formulation above forces the ellipsoid to contain all the a_i points. The second constraint tries to find ellipses that separate b_j from a_i as much as possible. The final result c^* and E^* separates a_i and b_j using an ellipse with the largest margin.

2.1.2 Minimum volume method (MINVOL)

MINVOL,discussed in detail in [7], tries to enclose a set of points in an ellipsoid with minimum volume. Finding a minimum volume enclosing ellipsoid can also be expressed as a conic programming problem:

$$\begin{aligned} &\text{minimize } trace(T) \\ &\text{subject to } (1,a_i)^T\tilde{E}(1,a_i) \leqslant 1 \ \forall i \\ &\tilde{E} = \begin{bmatrix} s & v^T \\ v & F \end{bmatrix} \geq 0 \\ &\begin{bmatrix} F & I \\ I & T \end{bmatrix} \geq 0 \end{aligned}$$

The minimum volume encolsing ellipsoid $\mathcal{E}^*$ is defined as $\mathcal{E}^* = \{x \in \mathbb{R}^n | (x - c^*)^T E^* (x - c^*) \leqslant 1\}$, where $c^* = -F^{-1}v$, $E^* = F/(1 - s + c^{*T} F c^*)$.

In the problem formulation above, $trace(T)$ models the volume of ellipsoid, as it can be shown that $boxsize(E) = \sqrt{\sum_{i=1}^{n} \lambda_i(E)^{-1}}$ approximates the volume of a ellipsoid E (Figure 2). The constraint of the problem formulation in essence guarantees that $T \geq E^{-1}$, which can be derived by applying Schur complement.

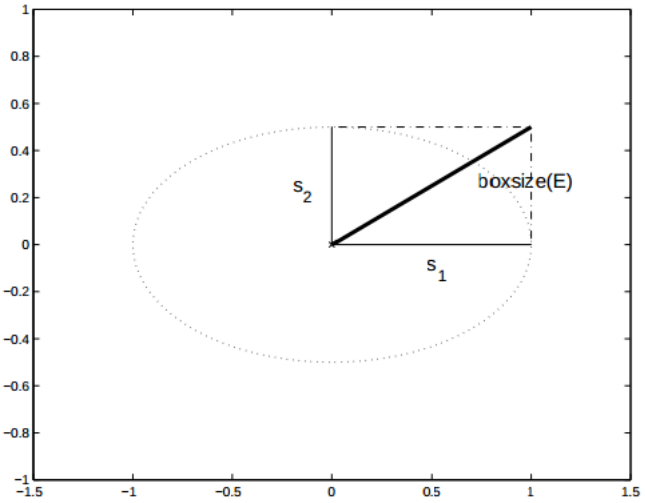


Figure 2: *boxsize* approximation of ellipsoid volume, adapted from [7]

2.1.3 Combined approach (COMB)

The previous two methods for finding separating ellipsoids always find the ones that enclose all the points of one class. However, because of outliers in dataset, not all data points should be used - elliminating some data points during ellipsoid construction may lead to better classification result. For an extreme example, in Figure 3, if one tries to separate the red points from the blue points using MINVOL, one will get the outer

ellipse as the decision region for the red points, which is obviously not a good decision region.

Instead, COMB provides a flexible way to control outlier effect using an approach that combines the previous two methods with an additional l_1-norm regularization. The method of COMB can be expressed as a convex programming problem:

$$\begin{aligned} \text{minimize } & -\beta_1 k + \beta_2 trace(T) + \beta_3 \|u - 1\|_1 \\ \text{subject to } & (1, a_i)^T \tilde{E}(1, a_i) \leqslant u_i \ \forall i \\ & (1, b_j)^T \tilde{E}(1, b_j) \geqslant k \ \forall j \\ & \tilde{E} = \begin{bmatrix} s & v^T \\ v & F \end{bmatrix} \succeq 0 \\ & \begin{bmatrix} F & I \\ I & T \end{bmatrix} \succeq 0, \end{aligned}$$

where $\beta_i > 0$ are the weights assigned to each sub-objectives: maximizing separation ratio, minimizing ellipsoid volume, and controlling outliers. The final ellipsoidal decision region $\mathcal{E}^*$ is defined as $\mathcal{E}^* = \{x \in \mathbb{R}^n | (x - c^*)^T E^* (x - c^*) \leqslant 2(1 + k)\}$, where $c^* = -F^{-1}v$, $E^* = F/(1 - s + c^{*T} F c^*)$.

In Figure 3, if one sets $\beta_1 = \beta_2 = \beta_3 = 1$, one will get the inner ellipse as the decision region for red points, which is clearly a much better decision region than the outer ellipse obtained using MINVOL.

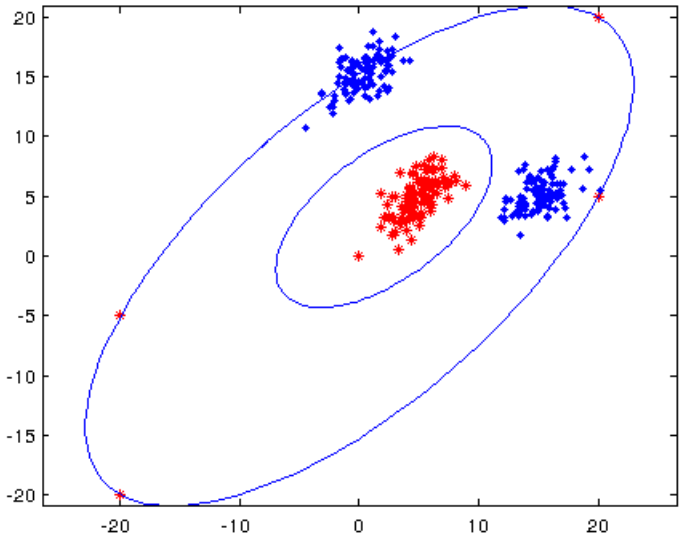


Figure 3: COMB finds better ellipsoidal decision regions when there are outliers

2.2 Classification

Once the elliptical decision regions, $\mathcal{E}_t^* = \{x \in \mathbb{R}^n | (x - c_t^*)^T E_t^* (x - c_t^*) \leqslant 1\}$, for $t = AA, AB, BB$, for a specific SNP, are obtained, SoCal uses them to call genotypes for samples with unknown genotype for the same SNP.

For each sample with allele intensity I_A and I_B, SoCal computes

$$D_t = (y - c_t^*)^T E_t^* (y - c_t^*),$$

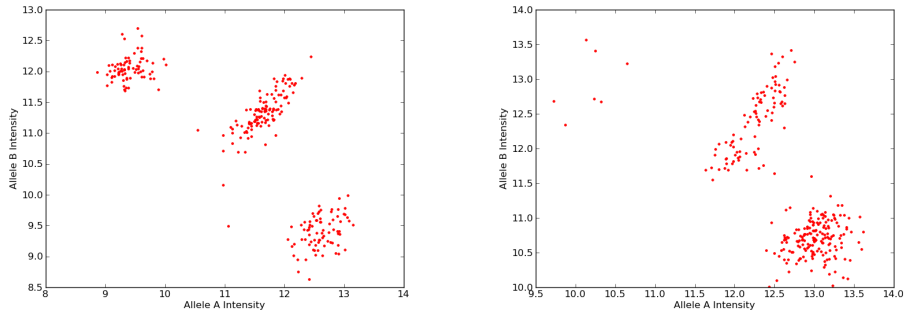
where $y = (I_A, I_B)$, for $t = AA, AB, BB$. SoCal then assigns genotype with the minimum D_t to that sample, that is

$$G = \underset{t \in \{AA, AB, BB\}}{\arg\min} (y - c_t^*)^T E_t^* (y - c_t^*).$$

3 Data

This project uses raw Affy100K SNP array allele intensity data and reference genotype calls all from the HapMap Project for both training and validating the genotype caller. In total, there are 115,428 SNPs, and 270 individual samples. Due to time constraint, I randomly selected 5,000 SNPs for training and validation instead of using all the SNPs. For each selected SNP, not all the 270 individual samples have reference genotype calls. On average, each SNP has 83 individual samples with reference genotype calls.

Each SNP has its own minor allele frequency (MAF), which indicates how often the minor (less frequent) allele of the SNP appears in the population. For SNPs with MAF close to 0.5, one observes three well defined genotype clusters (Figure 3), whereas for SNPs with very low MAF, some genotype cluster may not be well defined (Figure 3). These SNPs pose a challenge in genotype calling because there's much less data to train the genotype caller.



(a) SNPs with well defined genotype cluster

(b) SNPs with less well defined genotype cluster

Figure 4: Genotype clusters for SNPs with different minor allele frequency

Figure 5 shows the distribution of minor allele frequencies of the SNPs selected for this project. Most of the SNPs used in this project have moderate minor allele frequencies.

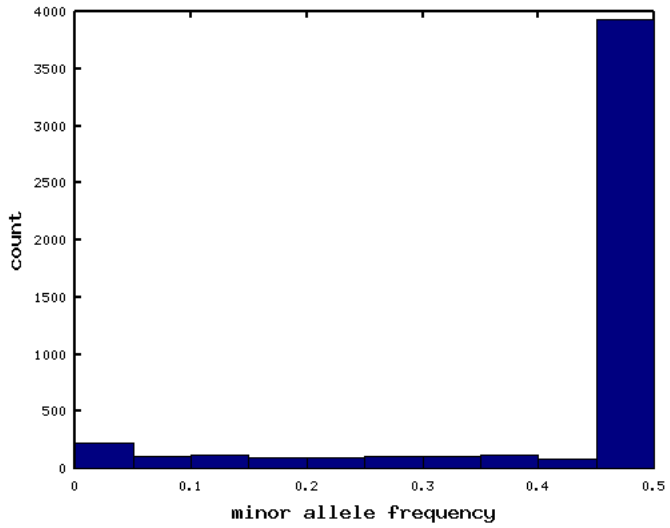


Figure 5: MAF distribution for the selected SNPs used in this project

4 Result

4.1 Implementation

The SoCal algorithm is implemented in Matlab with heavy use of CVX, which uses interior method to solve conic programming problems.

In general, the conic programming problems in this project can be solved in $O(\sqrt{v}log\frac{1}{\epsilon})$ iterations, with each iteration consisting of a damped Newton step [7]. The v in the notation refers to the v-self-concordant barrier of the domain of the conic programming problem, and ϵ the accuracy of the result [7].

4.2 Training and validation

For each SNP, 33% of individual samples data is randomly selected to train the genotype caller to find the decision regions for each SNP. And the rest 67% of the data is used to validate the caller. Three different methods for constructing ellipsoids, MAXSEP, MINVOL, and COMB are tested on the dataset. For COMB, β_1 is set to 1, β_2 is set 10, and β_3 is set to 100. The parameters are meant to punish outlier effect. Their performance is compared in the next section.

4.3 Performance comparison

Table 1 summarizes the performance of each method. It's clear that although the COMB method is slower, it's much more accurate than MAXSEP and MINVOL. And after finding the elliptical decision regions, future genotype calls can be done in linear time.

Method	Accuracy (%)	Training time (sec/SNP)	Calling time (sec/SNP)	Total training time (h)	Total calling time (sec)
MINVOL	41.1	1.68	0.0	2.34	0.27
MAXSEP	72.7	1.43	0.0	1.98	0.33
COMB	97.5	2.58	0.0	3.59	0.41

Table 1: Performance comparison for three different methods. K-means algorithm was tested in a previous project. The accuracy was 88.7%, and it took about 5 seconds per SNP. SoCal demonstrates that it's possible to use little reference genotype call to achieve high genotype calling accuracy.

Figure 6 shows an example of the elliptical decision regions found by SoCal for the three genotypes using the COMB method.

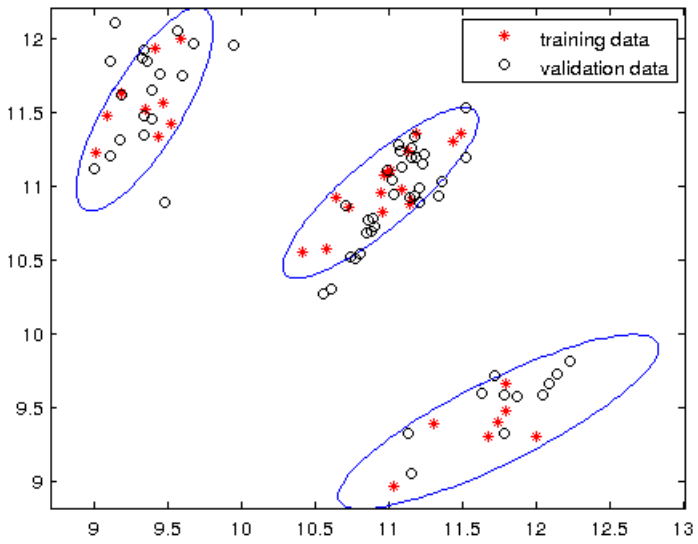


Figure 6: Decision regions found by SoCal for all the three genotypes for SNP_A-1641748

Figure 7 shows that the COMB method performs well even for SNPs with low frequency, and is much better than the MINVOL and MAXSEP methods. This is

because COMB takes into consideration three different objectives: separation ratio, ellipsoid volume, and controlling outliers, where as the other two methods only consider a single objective. Therefore, COMB generalizes well even if the amount of data is limited. But the other two methods tend to overfit the data.

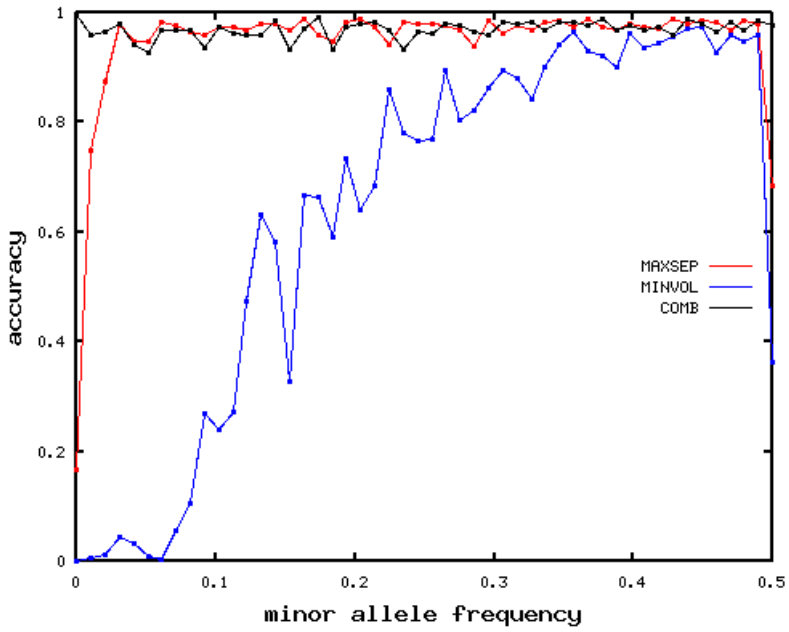


Figure 7: Genotype calling accuracy for SNPs with different minor allele frequencies for MAXSEP, MINVOL, and COMB

5 Discussion

SoCal demonstrates the possibility of using ellipsoids as decision regions for classifying SNPs into genotypes. In experiments involving HapMap data, SoCal is able to achieve 97.5% genotype calling accuracy using only a small portion of reference genotype calls.

SoCal is different from many other genotype calling algorithms in different aspects. First, SoCal is a supervised genotype caller that uses reference genotype calls to form decisions regions and then call genotypes for future SNPs. Second, SoCal is efficient - the problem of finding separating ellipsoids can be formulated as a conic programming problem and solved in polynomial time, and is guaranteed to find a global optimum. Third, the decision regions of SoCal can be used repeatedly. Once the decision regions of a SNP for a particular SNP array are found, future genotype calls can repeatedly use these regions without having to first train a genotype caller.

SoCal can be improved and extended in many ways. First, for a SNP with low minor allele frequencies, the ellipsoid learned directly from one genotype may not be the best ellipsoid for classification because of limited amount of data. Instead, one can use

the ellipsoids learned for other genotypes, which have more data, to infer the ellipsoid for the genotype with relatively less data. Second, SoCal can be extended to detect copy number variations in samples. Individual samples with copy number variations may have extra A's or B's at a SNP location. And if one plots the allele intensities (I_A, I_B) for these individuals, these data point will appear as outliers. SoCal can use the ellipsoids learned from reference genotype calls to detect outliers and then call copy number variations accordingly.

In summary, SoCal is an efficient and accurate genotype caller for Affymetrix SNP arrays. It can also be extended and improved to have more and better functionalities.

References

[1] Rho, S. W., Abell, G. C., Kim, K., Nam, Y., & Bae, J. (2010). Comparing microarrays and next-generation sequencing technologies for microbial ecology research. Trends in Biotechnology, 28, 291-299.

[2] Norlén, H., Pettersson, E., Ahmadian, A., Lundeberg, J., & Sundberg, R. (2008). Classification of SNP genotypes by a Gaussian mixture model in competitive enzymatic assays. Mathematical Statistics Stockholm University Research Report, 3, 1-26.

[3] Lin, Y., Tseng, G. C., Cheong, S. Y., Bean, L. J., Sherman, S. L., & Feingold, E. (2008). Smarter clustering methods for SNP genotype calling. Bioinformatics, 24, 2665-2671.

[4] Wu, C. F. (1983). On the Convergence Properties of the EM Algorithm. The Annals of Statistics, 11, 95-103.

[5] Rabbee, N., & Speed, T. P. (2005). A genotype calling algorithm for affymetrix SNP arrays. Bioinformatics, 22, 7-12.

[6] Zhou, X., & Stephens, M. (2014). Efficient multivariate linear mixed model algorithms for genome-wide association studies. Nature Methods, 11, 407-409.

[7] Glineur F. (1998). Pattern separation via ellipsoids and conic programming. (MS Thesis). Facult Polytechnique de Mons, Mons, Belgium.