Genotype Prediction with SVMs

Nicholas Johnson

December 12, 2008

1 Summary

A tuned SVM appears competitive with the FastPhase HMM (Stephens and Scheet, 2006), which is the current state of the art in genotype prediction. To improve the performance of the SVM, a large number of simulated training examples must be created by randomly pairing haplotypes in the training set. A window size parameter which determines how many adjacent SNP's act as features was also investigated and it is coarsely tuned by cross validation.

2 Introduction

A SNP marker (Single Nucleotide Polymorphism) is a position in the genome in which a single nucleotide varies from individual to individual. The flanking sequence is generally shared, so a microarray probe can be designed to detect the presence of either state of the SNP. For example one individual may have a sequence AAGTTA and another will have the sequence AAGGTA. The microarray provides estimates of an individual's genotypes, and the haplotypes must be inferred by statistical methods.

Scheet and Stephens (2006) introduce a Hidden Markov Model, named FastPHASE, for haplotype inference. They find that it gives worse performance than the PHASE II method (also proposed by Stephens and Scheet), but that surprisingly it is better at imputing genotypes at missing SNP sites. They then argue that imputing missing genotypes may ultimately be more important than accurate haplotype inference. Servin and Stephens (2007) develop a methodology for genotype association studies. The implementation available to practictioners uses FastPHASE and the markers measured by a microarray to infer the genotypes at the much denser set of markers in the datasets provided by the HapMap project.

3 SVM's for genotype imputation

This is a pure prediction problem, and the SVM may provide better performance despite making no use of the special structure of the data. The SVM is a black box model, but so is the HMM since its performance comes from averaging over multiple restarts of the EM algorithm.

3.1 Simulation Setup

The data in experiments to follow will be simulated from the CUE (European), YRI (Yoruban), CHB (Chinese) and JPT (Japanese) panels of the HapMap dataset. The HapMap consortium has posted inferred haplotypes at the set of bi-allelic markers for these populations and they have used the time consuming PHASE II model. The first two panels made use of parent-child trios to phase all but the triple-heterozygous positions, so these haplotypes are very accurate. I have taken the posted haplotypes from chromosome 1, and thinned to the set of SNP's which are shared in all four panels. The reason for this thinning is that the sample sizes are small and including the Asian samples may improve the predictive performance when inferring genotypes in European samples.

I then take a set of 1000 adjacent SNP's and divide them into two groups. Three of four SNPs are marked as unobserved and will be missing in test samples. The remaining 25% of SNP's will be used to predict at the unobserved locations. 100% of SNP's are present in the training samples.

3.2 Increasing training set sizes

If we are inferring genotypes for an individual of European descent, the HapMap CEU panel only provides us with 60 samples (after removing children from the trios). The Asian panel can be used to augment the training set and double the sample size. If we had many European training samples, adding samples from a different ethnic population could hurt performance. In this situation we are starved for training samples, so adding the Asian panel improves performance when predicting genotypes in samples of European descent.

The next way to increase the training set size is to simulate individuals based on inferred haplotypes. The HapMap data is provided phased and so we can random select two individuals from our training set, and randomly pair two of the four haplotypes to produce a new set of genotypes not observed in the training set. We repeat this to produce 1,000 simulated individuals to increase the training set size. A graphic representing the process is shown in figure (**??**).

As an example, suppose we select the one haplotype from the 10'th training sample and it is 001011 (it will actually be of length 1,000 in the experiments to follow). We select another random individual, not excluding the 10'th, and take one of their haplotypes : 101001. We then pair these to get a simulated sequence of genotypes, 102012, and we add this into the training set. We allow the possibility of selecting the same sample/haplotype twice, so we could have ended up with the homozygous sequence 002022.

3.3 SVM Kernel and Features

I have tried both the radial basis kernel $K(x, y) = \exp(-\gamma\|x - y\|^2)$ and the polynomial kernel $(\gamma x'y + c_0)^3$, the parameters γ and c_0 are left untunedand set to their default values of $1/\dim(x)$ and 0. The SVM package I am using, 'e1071' which is an R interface to LibSVM, has a default value of the complexity parameter $C = 1$. The software also implements regression SVM's and they gave similar performance.

A more important tuning parameter is the number of flanking SNP's to use as features. If we are predicting the genotype at a position T_0 on the chromosome, then a SNP at position T_1 is less likely to be informative if $|T_1 - T_0|$ is large. If we predict with too few flanking SNP's we will not have enough information, whereas if we predict using too many, the distance $\|x - y\|^2$ will be random.

We define a window of size K to be the K nearest observed SNP's. Distance could be defined in terms of the index of the SNP (i.e. 1,2,3,...), the physical position on the chromosome, or the genetic distance. The genetic distance is in fact estimated using an HMM, so rather than take this approach I have fit a distance measure to the observed correlation matrix between markers instead (next section).

3.4 SNP distances

If we have N SNP's we can construct the N by N correlation matrix C whose ij'th element C_{ij} equals the correlation between the phased, binary SNPs with indices i and j. We then model $|C_{ij}|$ by $\exp(-\sum_{k=i}^{j-1} d_k)$ for positive distances $d_k > 0$. The idea is to create a distance function $d(i,j)$ which is increasing in $|i - j|$ and takes into account the correlation structure of the data yet is still fast to compute.

By differentiating the squared error criteria $f(d)$ we get a gradient descent formula

$$f(d) = \sum_{i,j} \left(|C_{ij}| - \exp\left(- \sum_{k=i}^{j-1} d_k \right) \right)^2 \tag{1}$$

$$\Rightarrow \partial f(d)/\partial d_s = \Delta 2 \sum_{ij:i \le s < j} \exp\left(- \sum_{k=i}^{j-1} d_k \right) \left(|C_{ij}| - \exp\left(- \sum_{k=i}^{j-1} d_k \right) \right) \tag{2}$$

If we define a matrix A with elements $A_{ij} = 1\{i > j\} \exp(-\sum_{k=i}^{j-1} d_k)(|C_{ij}| - \exp(-\sum_{k=i}^{j-1} d_k))$, and a second matrix $B_{ij} = 1\{i > j\} \sum_{k=1}^{i-1} A_{ik}$, then we have a formula allowing efficient computation

$$\sum_{ij:i \le s < j} \exp\left(- \sum_{k=i}^{j-1} d_k \right) \left(|C_{ij}| - \exp\left(- \sum_{k=i}^{j-1} d_k \right) \right) = \sum_{j \ge k} B_{jk} \tag{3}$$

Figure (1) shows an observed correlation matrix and the fit. These distances allow the window of predictive SNP's to use more to the left than say the right if those on the left are more correlated overall.

3.5 Coding Genotypes

If we arbitrarily label the three genotypes as AA, AB, BB, then the genotype G of j'th predicting SNP can be used in a few different ways, but I will focus on two. The first is two code genotypes AA, AB and BB as 0,1 and 2 respectively. The second is two use to binary features. The first being 1 for and AB or a BB, and the second being 1 for a BB. This coding yields a larger distance between AA and BB than AB and BB in the Gaussian kernel. The 0-1 coding seems a reasonable choice, but in results shown later it can hurt the performance quite a bit.

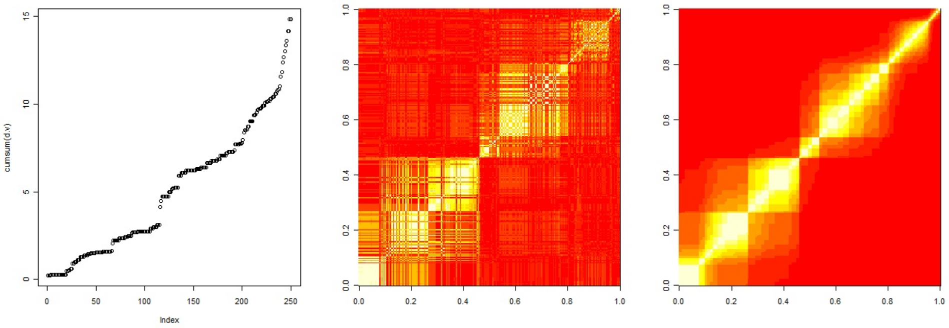

Figure 1: The left panel shows fitted positions $p_i := \sum_{j \leq i} d_j$. The middle panel shows the observed absolute correlation matrix. The right panel shows the fitted matrix $|\hat{C}_{ij}| = \exp(-\sum_{k=i}^{j-1} d_k)$.

3.6 Comparison

Test error rates will be measured on six CEU (European ancestry) samples at 750 of 1,000 SNP's on Chromosome 1. These six samples have 3 of 4 genotypes masked as missing. Figure (2) compares FastPhase (trained with phased data, predicting on unphased), to the SVM at various settings. The Asian panel samples are added to the training set, but all test individuals are of European ancestry.

The legend indicates whether voting amongst 3 classification SVM's was performed ('class') or a regression SVM was fitted ('eps-reg'). All SVM's were fitted using simulated training samples except for the points labeled 'n'. The addition of these extra training samples seems to have a bigger impact on performance than the choice of kernel.

The x axis of the figure is the window size (# flanking SNPs). In this case we do not use the fitted distances to choose the nearest SNP's, but just look at the marker index instead. Small window sizes do not provide enough information to predict the unobserved genotype, but performance does not appear to be too sensitive to window size as long as it is large enough.

Next I looked at a disjoint set of 1,000 SNP's and repeated the experiment on 5 different test sets consisting of 10% of the CEU samples. Now I use the fitted distances described above and choose the window size at each marker based on an inner loop of 3-fold cross validation. To reduce computation time the choice is restricted to a window size of either 10 or 24. The fitted distances were derived from all of the samples, so there may be some bias in their use.

Figure (3) has five columns each showing 5 ratios of the SVM error rate to the FastPHASE error rate on each of the 5 test sets. Unless otherwise noted, the fitted distances were used and genotypes were coded as 0,1, or 2 in a single feature. The SVM cost and kernel parameters were left at default values (I saw no benefit in attempts to tune these). The difference between the five columns is that the 1'st uses a Gaussian kernel rather than polynomial; the second codes genotypes with two 0-1 valued features each as described earlier; the third uses marker indices rather than the fitted distances; the fifth uses no simulated genotypes in the training set.

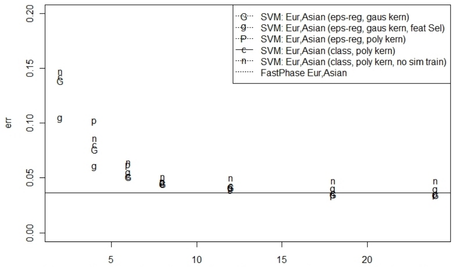

Figure 2. We show test set performance at several window sizes and settings of the SVM. A large enough window size and the use of simulated genotypes are the biggest factors in predictive performance.

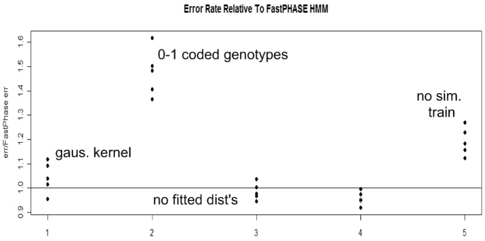

Figure 3. We compare error rates on a second set of SNP's and five different test sets. Unless otherwise noted a polynomial kernel using the fitted distance and simulated training set was used. A height of 1.5 indicates that the error rate on that test set was 50% worse than the FastPHASE HMM.

The polynomial kernel with fitted distances and simulated samples dominates the HMM, but only slightly. The benefit to using distances fitted to the correlation matrix is very minor and simply using the SNP index would suffice. On the othe hand, simulating extra samples by randomly pairing haplotypes improves performance significantly. Overall the performance is competitive with the HMM except in the case of column (2). I see no reason that this alternative coding of the genotypes should worsen performance by this much.

4 Conclusions

The SVM appears to be competitive in the dataset considered, but I would not recommend it over the HMM with such a small training set size. The reason is that the HMM requires almost no tuning and can easily handle missing predictor genotypes. So without a clear performance advantage, there is no reason to use the SVM. It is also worrisome that a reasonable change to the genotype coding can result in such a large increase in error rate.

With a larger training set size, I believe the SVM would dominate the HMM. It is not clear when such a training set would become available, however.

5 Bibliography

- Scheet, P. and Stephens, M. (2006). ”A fast and flexible statistical model for large-scale population genotype data: Applications to inferring missing genotypes and haplotypic phase” American Journal of Human Genetics. 78:629–44.

- Servin, B. and Stephens, M. (2007). ”Imputation-Based Analysis of Association Studies: Candidate Regions and Quantitative Traits.” PLOS Genetics.

- Stephens, M. and Scheet, P. (2005). ”Accounting for decay of linkage disequilibrium in haplotype inference and missing-data imputation” American Journal of Human Genetics. 76(3):449-62.