Ensemble-Based Imputation for Genomic Selection: an Application to Angus Cattle

Chuanyu Sun[1*], Xiao-Lin Wu [1,2], Kent A. Weigel [1], Guilherme J.M. Rosa [2,3], Stewart Bauck [4], Brent W. Woodward [4], Robert D. Schnabel [5], Jeremy F. Taylor [5], Daniel Gianola [1,2,3]

1 Department of Dairy Science, University of Wisconsin, Madison, WI 53706, USA;
2 Department of Animal Sciences, University of Wisconsin, Madison, WI 53706, USA;
3 Department of Biostatistics and Medical Informatics, University of Wisconsin, Madison, WI 53706, USA;
4 Merial Limited, Duluth, GA 30096, USA;
5 Division of Animal Sciences, University of Missouri, Columbia, MO 65211, USA.

Abstract

Imputation of moderate-density genotypes from low-density panels is of increasing interest in genomic selection, because it can markedly reduce genotyping costs. Several imputation software packages have been developed; however, these vary in imputation accuracy and imputed genotypes may be inconsistent over methods. An AdaBoost-like approach was developed to combine imputation results from several independent software packages, i.e., Beagle (v3.3), IMPUTE (v2.0), fastPHASE (v1.4), AlphaImpute, findhap (v2), and Fimpute (v2), with each package serving as a basic classifier in an ensemble-based system. The ensemble method computes weights sequentially for all classifiers, and combines results from component methods via weighted majority "voting" to determine unknown genotypes. The data included 3,078 registered Angus cattle, each genotyped with the Illumina BovineSNP50 BeadChip. SNP genotypes on three chromosomes (BTA1, BTA16, and BTA28) were used to compare imputation accuracy among methods, and our application involved imputation of 50K genotypes covering 29 chromosomes based on a set of 5K genotypes. Beagle and Fimpute had the greatest accuracy, which ranged from 0.8677 to 0.9858. The proposed ensemble method was better than any of these packages, but the sequence of independent classifiers in the voting scheme affected imputation accuracy. The ensemble systems yielding the best imputation accuracies were those that had Beagle as first classifier, followed by one or two methods that utilized pedigree information. A salient feature of our ensemble method is that it can solve imputation inconsistencies among different imputation methods, hence leading to a more reliable system for imputing genotypes relative to use of independent methods.

Key words: AdaBoost, cattle, ensemble-based system, genomic selection, imputation, single nucleotide polymorphisms (SNP)

Introduction

Single nucleotide polymorphism (SNP) genotyping chips have enabled an era of genomic selection, in which dense SNP genotypes covering the genome are used to predict the genetic merit of candidate individuals or lines for breeding purposes (Meuwissen *et al.,* 2001). However, commercial moderate density SNP arrays, such as the Ilumina BovineSNP50 Beadchip, are costly, which has limited their applications in beef cattle to males and elite females. As a cost-effective alternative solution to generating moderate density genotypes, various imputation strategies have been sought. The idea is to genotype candidate animals with a low-density platform comprising equally spaced SNPs, and then to impute moderate-density genotypes via some appropriate statistical model (e.g., Habier *et al.,* 2009; Weigel *et al.,* 2009).

Several software packages have been developed for genotype imputation in humans or livestock. Based on the sources of information used to infer missing genotypes, imputation methods can be divided into family-based or population-based, or those that make use of both sources. The family-based approach makes use of linkage and Mendelian segregation rules, and is most accurate for animals having genotyped relatives. The population-based approach utilizes linkage disequilibrium (LD) information between missing SNPs and the observed flanking SNPs,

and is well suited for a set of unrelated individuals or for animals whose close ancestors have not been genotyped. In practice, however, choosing an appropriate method is not always easy. One may wish to choose a method that yields the greatest imputation accuracy, but such information is not available before the data at hand are actually analyzed. In addition, none of the current methods provides perfect imputation, and imputed genotypes may be inconsistent among programs. Solving such inconsistencies poses another challenge in imputation. From the viewpoint of machine learning, genotype imputation can be considered as a classification problem, and each imputation method can be viewed as an independent classifier. Ensemble learning algorithms (e.g. Polikar, 2006) can be helpful for combining predictions from alternative models, and can yield final classification results that are more robust than those from individual classifiers.

Ensemble learning is a machine learning paradigm in which several learners are trained to solve the same problem, and AdaBoost (Freund and Schapire, 1996) is one of the most widely used ensemble methods. The basic principle of AdaBoost is to combine multiple base classifiers to produce a committee, whose performance is better than that of any of the base classifiers. The latter are trained in sequence using a weighted form of the data set in which the weights associated with each data point depend on the performance of the previous classifiers. Points that are misclassified by one of the base classifiers are given greater weight when used to train the next classifier in the sequence. Once all classifiers have been trained, their predictions are combined through a weighted majority voting scheme (Bishop, 2006). At present, no reports are available on the application of Adaboost to genotype imputation in animal genetics and breeding.

The objective of this study was to investigate the performance of an ensemble approach to imputing moderate-density SNP genotypes. This approach was used to impute 50K genotypes from 5K genotypes in a registered Angus cattle population.

Materials and Methods

Data

Data was from Merial Limited and consisted of 3,078 Angus animals, each genotyped using the Illumina BovineSNP50 BeadChip. After data edition and Quality control, the final data set has 3,059 animals and 51,911 SNPs across the whole genome. To assess imputation accuracy, cross-validation was used, with the dataset randomly divided into three approximately equal portions. Two of the portions were used for training the imputation models, and the remaining portion was used for testing imputation accuracy. We focused on three representative chromosomes: 1 (longest), 16 (moderate size), and 28 (one of the shortest). There were 3348 SNPs on chromosome 1, 1628 SNPs on chromosome 16, and 944 SNPs on chromosome 28 in the training sets. In the testing sets, there were 357, 192, and 103 SNPs with known genotypes on these three chromosomes, respectively, which corresponded to subsets of 5K (now known as the Illumina BovineLD 7K assay) genotypes. All of the remaining genotypes for animals in the testing set were treated as "missing" and were subsequently imputed.

Imputation programs

The six software packages used to impute "missing" genotypes in the testing set were Beagle3.3 (Browning and Browning, 2009), IMPUTE2.0 (Howie *et al.*, 2009), fastPHASE1.4 (Scheet and Stephens, 2006), findhap version 2 (VanRaden *et al.*, 2011), AlphaImpute (Hickey *et al.*, 2011), and Fimpute version 2 (Sargolzaei *et al.*, 2011), and the abbreviations used for the six packages were Bgl, Imp, fPh, fhap, Alp and Fimp, respectively.

AdaBoost-like ensemble algorithm

An AdaBoost-like algorithm was designed to combine imputation results from the aforementioned software packages. A wrapper

program was used to coordinate individual packages and to implement computations for the proposed ensemble method. Let **X** be a set of imputed genotypes, and **y** be a vector of observed ("true") genotypes at a given SNP locus. Let T=6 be the number of independent classifiers (i.e. the imputation softwares). Given a training set of *N* individuals, we have

$$\mathbf{S} = \left[\left(x_1, y_1 \right), ..., \left(x_i, y_i \right), ..., \left(x_N, y_N \right) \right],$$

where

$$x_i \in \mathbf{X} = \{ x_{i1}, x_{i2}, x_{i3} | i = 1, 2, ..., N \} \quad ,$$
$$y_i \in \mathbf{y} = \left(g_1 \quad g_2 \quad g_3 \right), \text{ and } g_1, \ g_2 \text{ and } g_3$$

are the three genotypes at the SNP, in question, for individual *i* in the training sample.

Initialize: each individual was assigned an equal weight, $W_1(i) = 1/N$, for $i = 1, ..., N$.

Training: For t =1, 2,…, T classifiers

1. Call classifier t, which in turn generates hypothesis h_t (i.e., inferred haplotypes and genotypes in the training set)

2. Calculate the error of h_t:

   $$\varepsilon_t = \frac{\sum_{i=1}^{N} W_t(i) I\left(h_t\left(x_i \right) \neq y_i \right)}{\sum_{i=1}^{N} W_t(i)} \ .$$

   Where $I\left(h_t\left(x_i \right) \neq y_i \right)$ is an indicator function that equals 1 when $h_t(x_i) \neq y_i$ and 0 otherwise. Looping is aborted if $\epsilon_t > 1/2$.

3. Set $\beta_t = \log\left(\frac{1-\varepsilon_t}{\varepsilon_t} \right)$

4. Update the weight distribution $W_t(i)$ for the next classifier as:

   $$W_{t+1}(i) = W_t(i) \exp\left(\beta_t I\left(h_t(x_i) \neq y_i \right) \right)$$

Testing: In the testing set, each "unknown" genotype is classified via so-called "weighted majority voting". Briefly, the wrapper program:

1. Computes the total vote received by each genotype (class)

   $$v_i = \sum_{t=1}^{T} \left\{ \beta_t I'\left(h_t(x_i) = g_j \right) \right\} \quad , \quad \text{for}$$

   $j = 1, 2, 3$. Where $I'\left(h_t(x_i) = g_j \right)$ is an indicator function that equals 1 when $h_t(x_i) = g_j$ and 0 otherwise.

2. Assigns the genotype (class) that received the largest total vote as the final ("putative") genotype.

Above, the algorithm maintains a weighted distribution $W_t(i)$ of training samples x_i, for $i = 1.....N$, from which a sequence of training data subsets S_t is chosen for each consecutive classifier (package) *t*.

Bootstrap sampling and parallel computing

Bootstrapping was used to generate empirical confidence intervals of imputation accuracy for the six packages and for the ensemble systems as well. For each method, 50 replicates were created by drawing random samples with replacement from the original testing set, each conducted on the genotype data for one of the three chromosomes, and the size of each bootstrap sample equaled the size of the original testing set. Finally, summary statistics were computed from the 50 bootstrap samples. Note that the distribution is conditional on the training set. Given six independent packages, there were 6! = 720 combinations, each defining a unique ensemble system. The computing task was formidable. For example, given this design, there were a total of $\left(720 + 6\right) \times 50 \times 3 = 108,900$ independent jobs. Hence, distributed high-throughput computing solutions were utilized and these jobs were submitted to run on the University of Wisconsin Condor Systems and Open Science Grid (Wu *et al.*, 2012).

Results

Comparing imputation accuracy among software packages

The six packages varied in imputation accuracy when evaluated on Angus chromosomes 1, 16 and 28 (Table 1). Bgl had the greatest imputation accuracy on all three chromosomes, followed by Fimp and fhap. On chromosome 1, for example, mean imputation accuracy obtained with Bgl was 0.9858 and that obtained with each of the remaining five packages ranged from 0.9084 (Alp) to 0.9788 (Fimp). Similar patterns were observed on the other two chromosomes: imputation accuracy varied from 0.9092 (Alp) to 0.9837 (Bgl) on chromosome 16, and from 0.8677 (fPh) to 0.9712 (Bgl) on chromosome 28.

Table 1. Summary statistics of the bootstrap distribution of imputation accuracy obtained using each of the six software packages on chromosomes 1, 16 and 28 .

	Method	Median	Mean	SD
Chrom_1	Bgl	0.9858	0.9858	0.0006
	Imp	0.9375	0.9375	0.0013
	fPh	0.9286	0.9286	0.0014
	fhap	0.9649	0.9648	0.0014
	Alp	0.9083	0.9084	0.0039
	Fimp	0.9789	0.9788	0.0007
Chrom_16	Bgl	0.9836	0.9837	0.0006
	Imp	0.9323	0.9325	0.0012
	fPh	0.9207	0.9208	0.0015
	fhap	0.9537	0.9536	0.0018
	Alp	0.9098	0.9092	0.0041
	Fimp	0.9728	0.9728	0.0010
Chrom_28	Bgl	0.9712	0.9712	0.0011
	Imp	0.8890	0.8887	0.0024
	fPh	0.8679	0.8677	0.0021
	fhap	0.9355	0.9354	0.0025
	Alp	0.8937	0.8937	0.0039
	Fimp	0.9592	0.9589	0.0015

Comparing imputation accuracy between ensemble methods and individual packages

Within the 720 unique ensemble systems, imputation accuracies of the top five ensemble systems, evaluated on each of the three chromosomes, were compared with those of each of the individual packages (Fig 1).

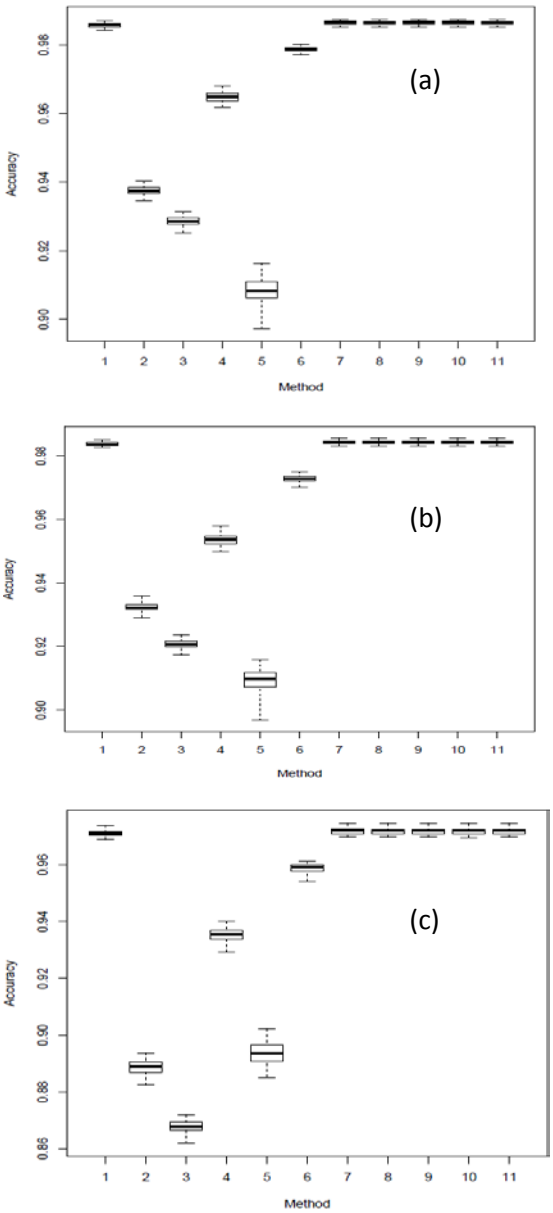

Figure 1. Box plots of imputation accuracy on (a) chromosome 1, (b) chromosome 16, and (c) chromosome 28, obtained using six imputation software packages and five ensemble methods
Results are obtained from 50 boostrap replicates. For x-axis labels, 1 = "Beagle3.3", 2 = "IMPUTE2.0"; 3 = "fastPHASE1.4"; 4 = "findhap version 2"; 5 = "AlphaImpute"; 6 = "Fimpute version 2"; 7 ~ 11 = five ensemble systems.

These ensemble systems performed similarly to each other, and all were at least as good as each of the six individual imputation packages. We observed that imputation accuracy varied with the order of the software packages in the ensemble system. For each of the three chromosomes, all top 120 ensemble systems with the highest accuracy of imputation had Bgl as the first classifier. Ensemble systems with Fimp and Bgl as the first two classifiers also had high imputation accuracy. The lowest imputation accuracy was observed with fPH and Imp appearing as the first two classifiers. Our results indicate that an ensemble method starting with the best individual classifier (i.e., Bgl) could have the best overall performance. Also, alternating population-based and family-based approaches could enhance imputation as well. Therefore, optimal ensemble systems, as supported by the present data, turned out to be those starting with Bgl, followed by one or two of the packages that can use pedigree information for imputation (e.g. fhap, Fimp and Alp).

An application: imputation of moderate-density genotypes in Angus cattle

Based on a 5K-genotype panel, moderate-density (50K) genotypes on 29 chromosomes were imputed for 3,078 animals using the aforementioned six imputation packages and five ensemble systems. All five selected ensemble systems had Bgl and Alp as the first two classifiers, and were as follows: 1) Bgl-Alp-fhap-Imp-fPH-Fimp, 2) Bgl-Alp-Fimp-Imp-fPH-fhap, 3) Bgl-Alp-Fimp-fPH-Imp-fhap, 4) Bgl-Alp-Imp-fhap-fPH-Fimp, and 5) Bgl-Alp-Imp-Fimp-fPH-fhap. Imputing accuracies using the six packages and the five ensemble systems are illustrated in Fig 2. The five ensemble systems gave similar imputation results, and were consistently better than each of the six imputation packages. Among the 29 autosomes, imputation accuracy ranged from 0.9715 (chromosome 28) to 0.9855 (chromosome 1) with the five ensemble systems, and it varied from 0.8869 (Alp, chromosome 10) to 0.9853 (Bgl, chromosome 1) with the six independent packages.

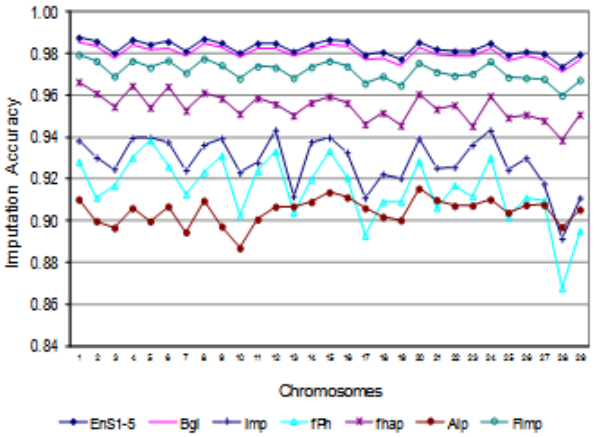

Figure 2. Comparison of imputation accuracy evaluated on 29 autosomes in registered Angus cattle using 6 independent imputation packages and 5 ensemble systems. For EnS1-5, the figure gives the average accuracy of the 5 ensembles.
EnS1-5 = five ensemble methods

Discussion

Genotype imputation can be viewed as a classification problem. Several imputation methods (i.e., software packages) are available, but results may be inconsistent among them. Ensemble methods can be used to solve such inconsistencies, and thus further improve imputation accuracy. This was corroborated in our study. The proposed ensemble method resembles AdaBoost, in that weights for each of classifiers are computed sequentially and imputed genotypes are decided by weighted majority voting. The idea is intuitive: classifiers that have a good performance during training are rewarded with higher voting weights than the others. Our ensemble systems combined results from six imputation packages: Bgl, Imp, fPH, Alp, fhap and Fimp. In this set, Bgl and Fimp had the highest imputation accuracy. The proposed ensemble systems improved imputation accuracy in our data, but the degree of improvement depended on the order of these classifiers in the ensemble systems. The best ensemble systems were those with Bgl as the first classifier, followed by one or two software packages that used pedigree information during imputation. Rotating different types of imputation packages in the ensemble systems is desirable,

because training by consecutive classifiers may be better geared toward increasingly hard-to-classify instances (Polikar, 2006).

Boosting algorithms have been developed for enhancing weak learners so, when the extant methods are strong classifiers, there is little room for improvement. This was confirmed in our study, the first of its kind in the context of genotype imputation. Further improvements through adjustment of the proposed ensemble method may be possible. First, one may form a committee of classifiers with higher diversity, each focusing on a different scenario guiding imputation. This is an essential idea of AdaBoost, which works well provided that each classifier can produce an imputation that is slightly better than a random guess. We have included two types of imputation packages, i.e., family-based and population-based. Additional options may include imputation based on population frequencies only (a weak classifier) and imputation based on posterior probabilities of unknown genotypes given observed phenotypes and prior information about the genotypes (also a weak classifier). The latter two options, however, were not investigated, because the six packages we used provided relatively accurate imputation, and including these two weak classifiers would have made little difference in imputation accuracy. Also, individual packages can be modified so that a set of classifiers can be trained more efficiently and adaptively, but this may not always be possible due to the lack of availability of source code. Nevertheless, there are some ensemble methods that do not require modification of each independent imputation package, such as stacked generalization (Wolpert, 1992; Polikar, 2006) or mixture of experts (Jacobs *et al.*, 1991; Jordan and Jacobs, 1994). These two ensemble methods can use the outputs of a set of individual classifiers as inputs to a second level meta-classifier, which then learns the mapping between the ensemble outputs and the correct classes. These methods may be worth investigating in future studies.

Acknowledgments

This research was supported by the Wisconsin Agriculture Experiment Station, and a Genomic Selection Grant by Merial Ltd. JFT was supported by National Research Initiative grants number 2008-35205-04687, 2008-35205-18864 and 2009-35205-05099 from the USDA Cooperative State Research, Education and Extension Service, and grant number 2009-65205-05635 from the USDA Agriculture and Food Research Initiative.

References

Bishop, C.M. 2006. *Pattern Recognition and Machine Learning.* New York, USA: Springer.

Browning, B.L. & Browning, S.R. 2009. A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. *Am J Hum Genet 84*, 210–223.

Freund, Y. & Schapire, R.E. 1996. Experiments with a new boosting algorithm. In: *Machine Learning: Proceedings of the Thirteenth International Conference*, pp. 148–156.

Habier, D., Fernando, R.L. & Dekkers, J.C.M. 2009. Genomic selection using low-density marker panels. *Genetics 182*, 343–353.

Hickey, J.M., Kinghorn, B.P., Tier, B., Wilson, J.F., Dunstan, N. & van der Werf, J.H.J. 2011. A combined long-range phasing and long haplotype imputation method to impute phase for SNP genotypes. *Genet Sel Evol 43*, 12.

Howie, B.N., Donnelly, P. & Marchini, J. 2009. A flexible and accurate genotype imputation method for the next generation of genome-wide association studies. *PLoS Genet* ***5***, *e1000529.*

Meuwissen, T.H.E., Hayes, B.J. & Goddard, M.E. 2001. Prediction of total genetic value using genome-wide dense marker maps. *Genetics 157*, 1819–1829.

Polikar, R. 2006. Ensemble Based Systems in Decision Making. *IEEE Circuits and Systems Magazine 6,* 21-45.

Sargolzaei, M., Chesnais, J.P. & Schenkel, F.S. 2011. FImpute-An efficient imputation algorithm for dairy cattle populations. *J. Anim. Sci. 89(E-Suppl. 1)/J. Dairy Sci. 94(E-Suppl. 1): 421 (abstr. 333).*

Scheet, P. & Stephens, M. 2006. A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase. *Am J Hum Genet 78,* 629–644.

VanRaden, P.M., O'Connell, J.R., Wiggans, G.R. & Weigel K.A. 2011. Genomic evaluations with many more genotypes. *Genet Sel Evol 43,* 10.

Weigel, K.A., de los Campos, G., Gonzalez-Recio, O., Naya, H., Wu, X.L., Long, N., Rosa, G.J.M., & Gianola, D. 2009. Predictive ability of direct genomic values for lifetime net merit of Holstein sires using selected subsets of single nucleotide polymorphism markers. *J. Dairy Sci. 92,* 5248–5257.

Wolpert, D.H. 1992. Stacked generalization. *Neural Networks 5,* 241–259.

Wu, X-L., Hayrettin, O., Duan, H., Bessinger, T., Bauck, S., Woodward, B., Rosa, G.J.M., Weigel, K.A., de Leon, N., Taylor, J.F. & Gianola, D. 2012. Parallel-BayesCpC on OSG: grid-enabled high-throughput computing for genomic selection in practice. *International Plant & Animal Genome 2012* (http://www.intlpag.org/web/index.php/abstracts/poster-abstracts)