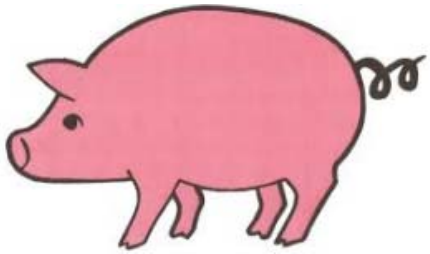
THE UNIVERSITY of WISCONSIN MADISON

Ensemble-based imputation for genomic selection: an application to Angus cattle

Chuanyu Sun, Xiao-Lin Wu, Kent A. Weigel, Guilherme J.M. Rosa , Stewart Bauck , Brent W. Woodward, Robert D. Schnabel, Jeremy F. Taylor , Daniel Gianola

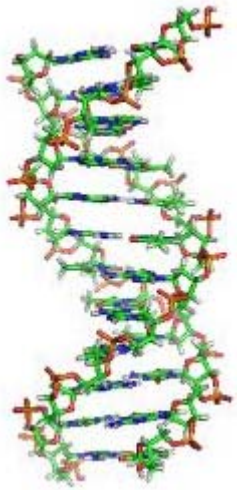




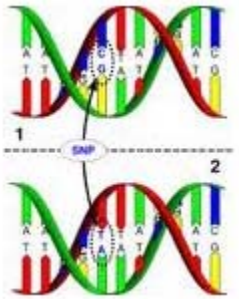

Introduction

Why imputation?

7k SNP



50k SNP



Reduce genotyping costs

Improving accuracy of GS relative to low density SNP?

Introduction

Imputation Principle

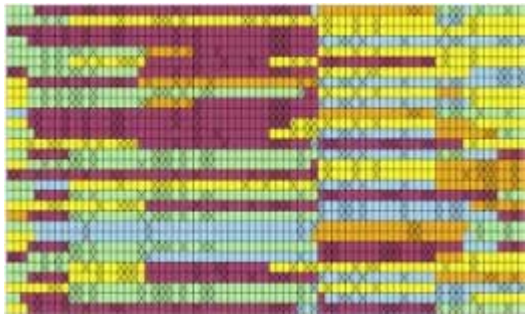
Imputation methods: two groups:

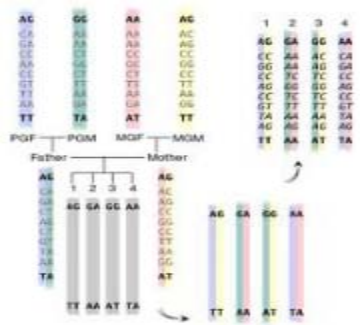
1)Family based algorithms

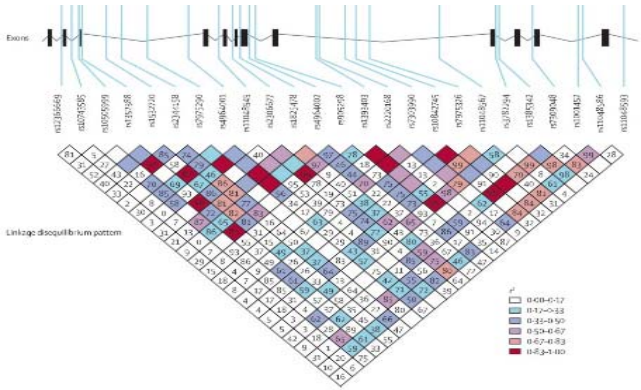
- ✓ *Use linkage and Mendelian segregation rules*

2)Population based algorithms

- ✓ *Use linkage disequilibrium information between missing SNP and the observed flanking SNPs*







Introduction

Imputation software

Beagle

Impute

fastPHASE

AlphaImpute

Findhap

Fimpute

PHASEBOOK

CHROMIBD

PLINK

MACH

●●●●●

Introduction

Imputed genotypes may be inconsistent among different programs.

How such inconsistencies can be solved is a another challenge in imputation.

Introduction

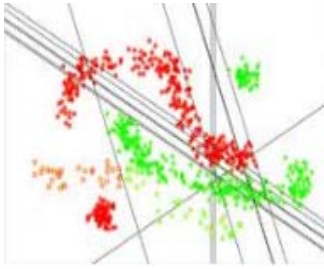
Machine learning:

Imputation of genotypes is a classification problem
Each imputation method is an "independent" classifier

Ensemble learning algorithms:

Combine predictions from multiple models, and yield classification results that are more robust relative to individual procedures

AdaBoost: one of the most widely-used ensemble methods



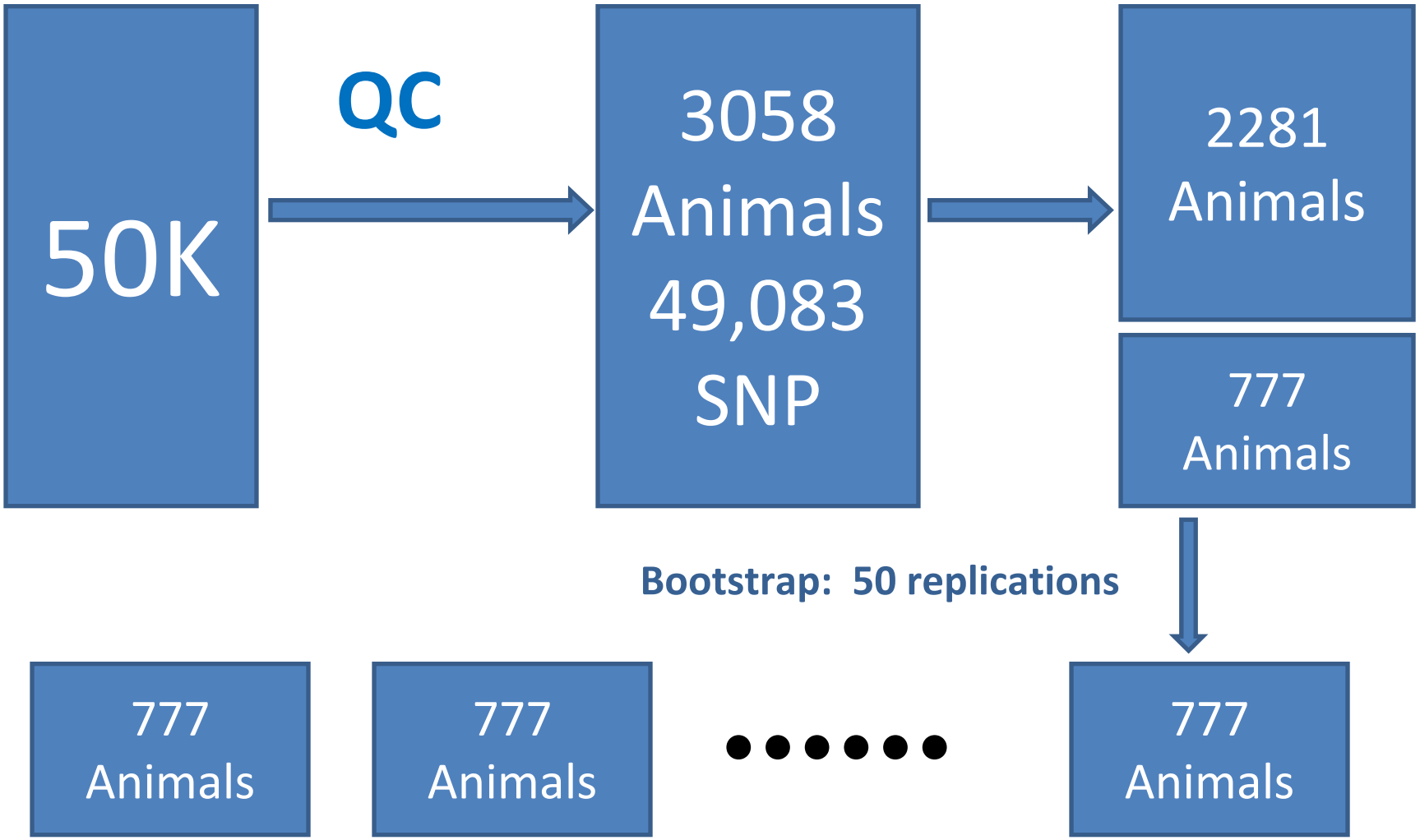
Introduction

Objectives:

Investigate performance of an ensemble approach to imputation of high-density genotypes.

Application to imputation of 50K genotypes from 7K genotypes in Angus cattle.

Data and Methods



Imputation done chromosome by chromosome

Data and Methods

1. **Focused on three representative chromosomes only:**
   - ***Chromosome 1 (longest)***
   - ***Chromosome 16 (moderate size)***
   - ***Chromosome 28 (one of the shortest).***

2. **Six imputation softwares:**
   - ***Beagle,***
   - ***IMPUTE***
   - ***fastPHASE1.4***
   - ***findhap***
   - ***AlphaImpute***
   - ***Fimpute***

3. **AdaBoost-like ensemble algorithm**

Data and Methods

AdaBoost-like ensemble algorithm

Let *x* = imputed genotypes, y =observed ("true") SNP genotypes, and T = the number of independent classifiers

Initialize: each sample (animal) was initialized with an equal weight.

Training:

For t =1, 2,… , T classifiers

1.Call classifier t, which in turn generates hypothesis h_t

2.At a given SNP locus, calculate the error of :

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

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

4.Update weight distribution :

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

Data and Methods

AdaBoost-like ensemble algorithm

Test: In the test set, each "unknown" genotype is classified via so called "weighted majority voting".

1. computes total vote received by each genotype (class)

$$v_j = \sum_{t=1}^{T} \left\{ \beta_t I' \left(h_t(x_i) = g_j \right) \right\} \qquad j = 1, 2, 3$$

2. assigns the genotype (class) that receives the highest total vote as the final ("putatively true") genotype.

Classifiers working together!!

Simulation:

1. Genotypes for 800 animals and one marker locus simulated: genotypes are “AA”, “AB” and “BB”.
2. Classifiers with different accuracy of imputation simulated
3. Final results gotten as:

$$v_j = \sum_{t=1}^{T} \left\{ \beta_t I' \left(h_t(x_i) = g_j \right) \right\} \qquad j = 1, 2, 3$$

Classifiers	Number of classifiers				
	5	10	20	50	100
C_1 (0.5625)	0.7250	0.8250	0.9363	0.9950	1.0000
C_2 (0.6250)	0.8038	0.9113	0.9725	1.0000	1.0000
C_3 (0.6875)	0.8775	0.9400	0.9925	1.0000	
C_4 (0.7500)	0.9313	0.9788	0.9975	1.0000	
C_5 (0.8125)	0.9475	0.9863	0.9913	1.0000	
C_6 (0.8750)	0.9800	0.9950	1.0000		
C_7 (0.9375)	0.9950	0.9963	1.0000		
C_8 (0.9750)	0.9988	1.0000			

Results: individual performance

Imputation accuracy by software package

	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

Results: implementation

AdaBoost-like ensemble algorithm

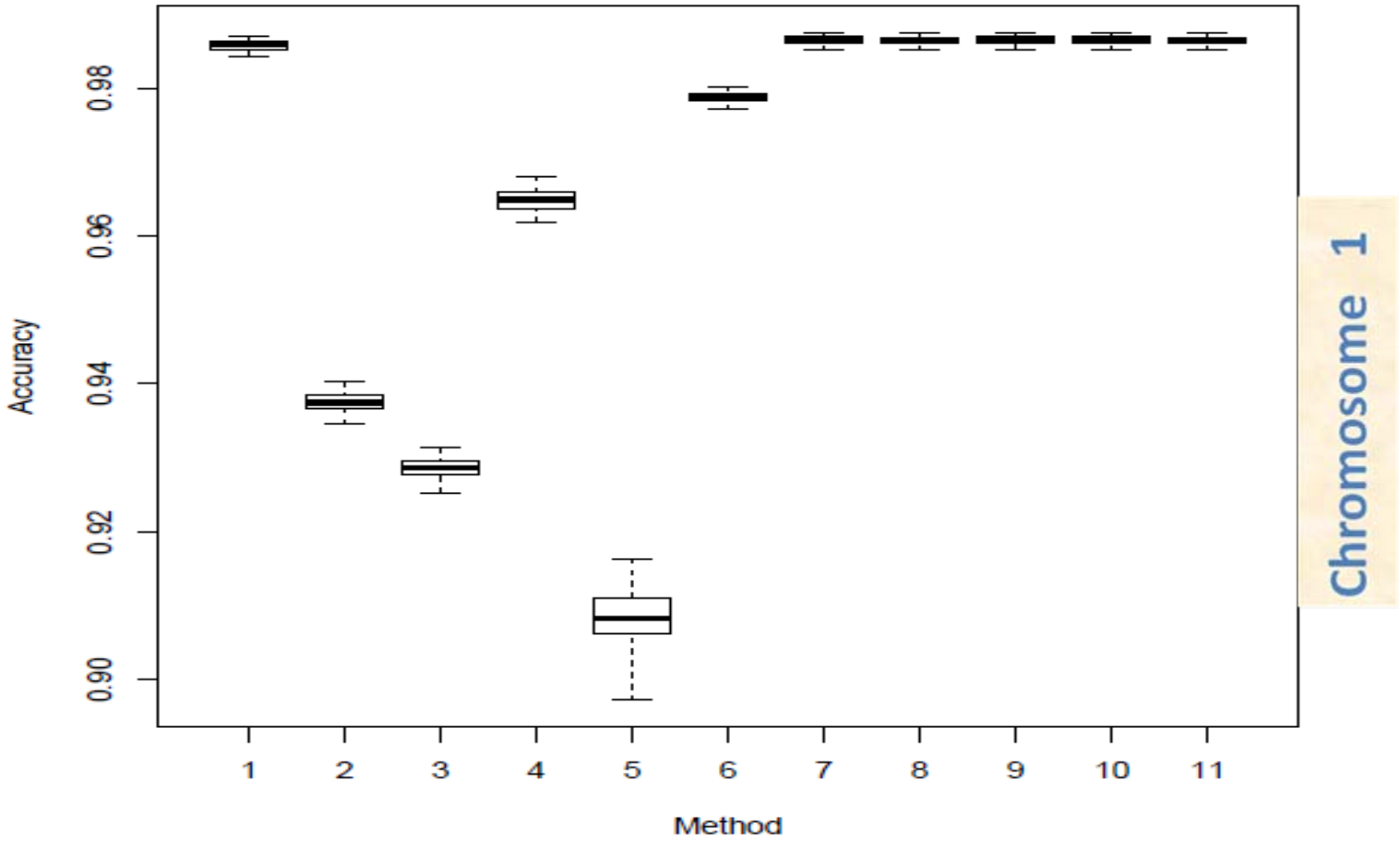
1. Given six packages, there were 6!=720 combinations, each defining a unique ensemble system

2. For each chromosome, there are 720*50 =36000 jobs

3. Distributed high-throughput computing used on University of Wisconsin Condor systems and Open Science Grid

Results: accuracy and uncertainty

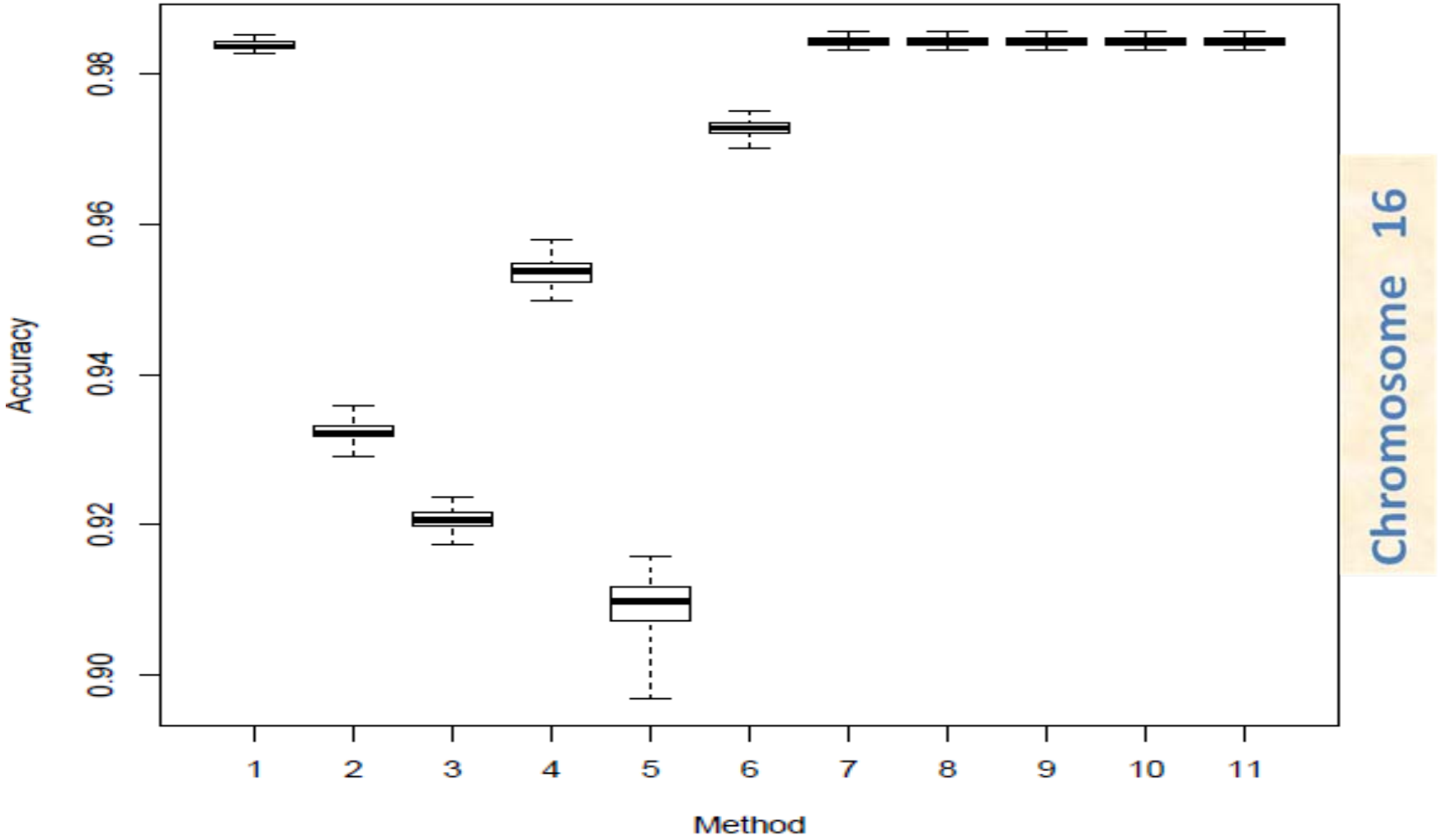
AdaBoost-like ensemble algorithm



1 = "Beagle3.3", 2 = "IMPUTE2.0"; 3 = "fastPHASE1.4"; 4 = "findhap version 2";
5 = "AlphaImpute"; 6 = "Fimpute version 2"; 7 ~ 11 = five ensemble systems.

Results

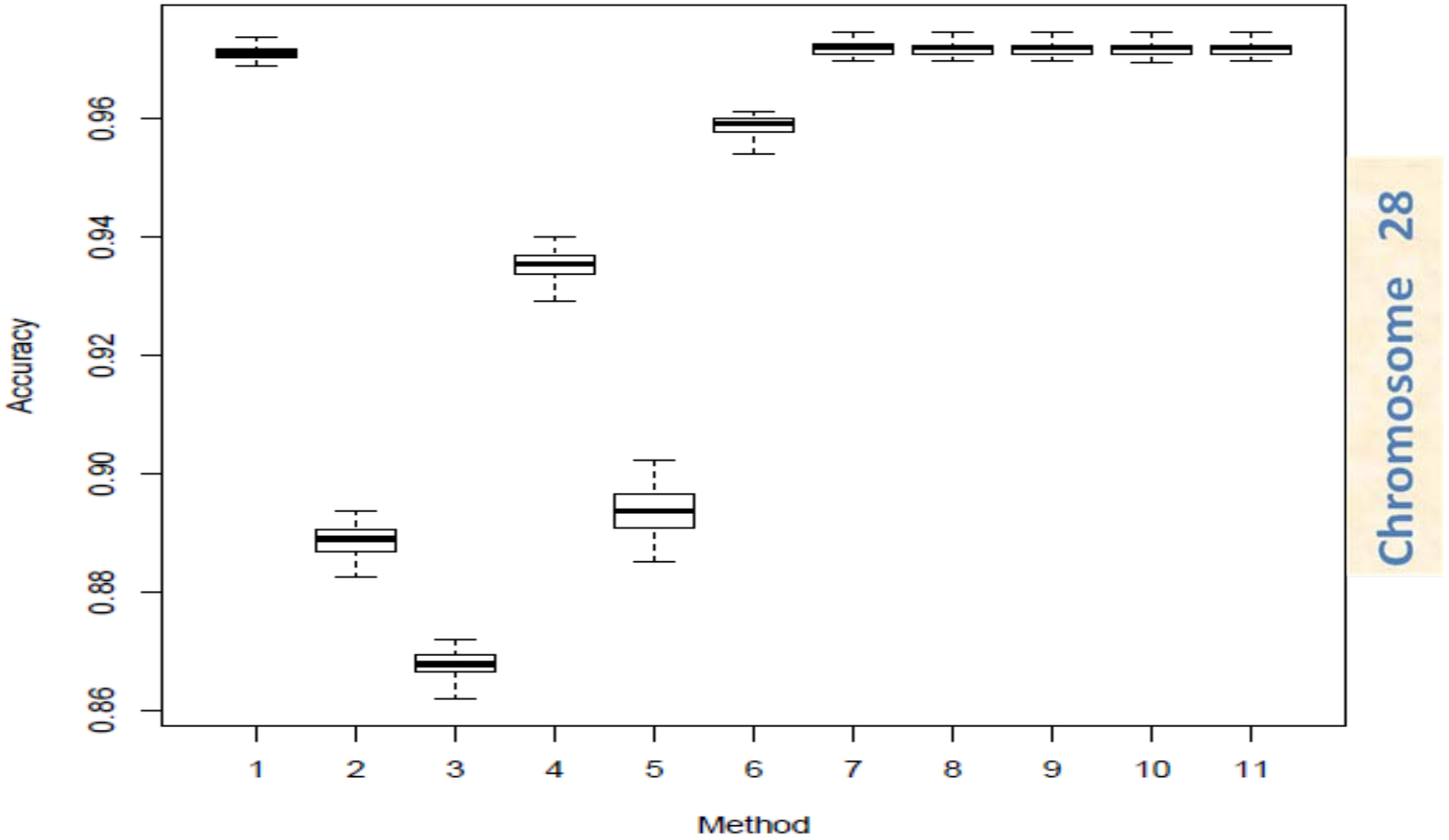
AdaBoost-like ensemble algorithm



1 = “Beagle3.3”, 2 = “IMPUTE2.0”; 3 = “fastPHASE1.4”; 4 = “findhap version 2”; 5 = “AlphaImpute”; 6 = “Fimpute version 2”; 7 ~ 11 = five ensemble systems.

Results

AdaBoost-like ensemble algorithm



1 = "Beagle3.3", 2 = "IMPUTE2.0"; 3 = "fastPHASE1.4"; 4 = "findhap version 2";
5 = "AlphaImpute"; 6 = "Fimpute version 2"; 7 ~ 11 = five ensemble systems.

Results

AdaBoost-like ensemble algorithm

1. Ensemble method better than any of these 6 packages.

2. Sequences of packages in voting affect imputation accuracy

3. Ensemble systems with best accuracies had **Beagle** as first classifier, followed by one or two methods that utilized pedigree information.

Results

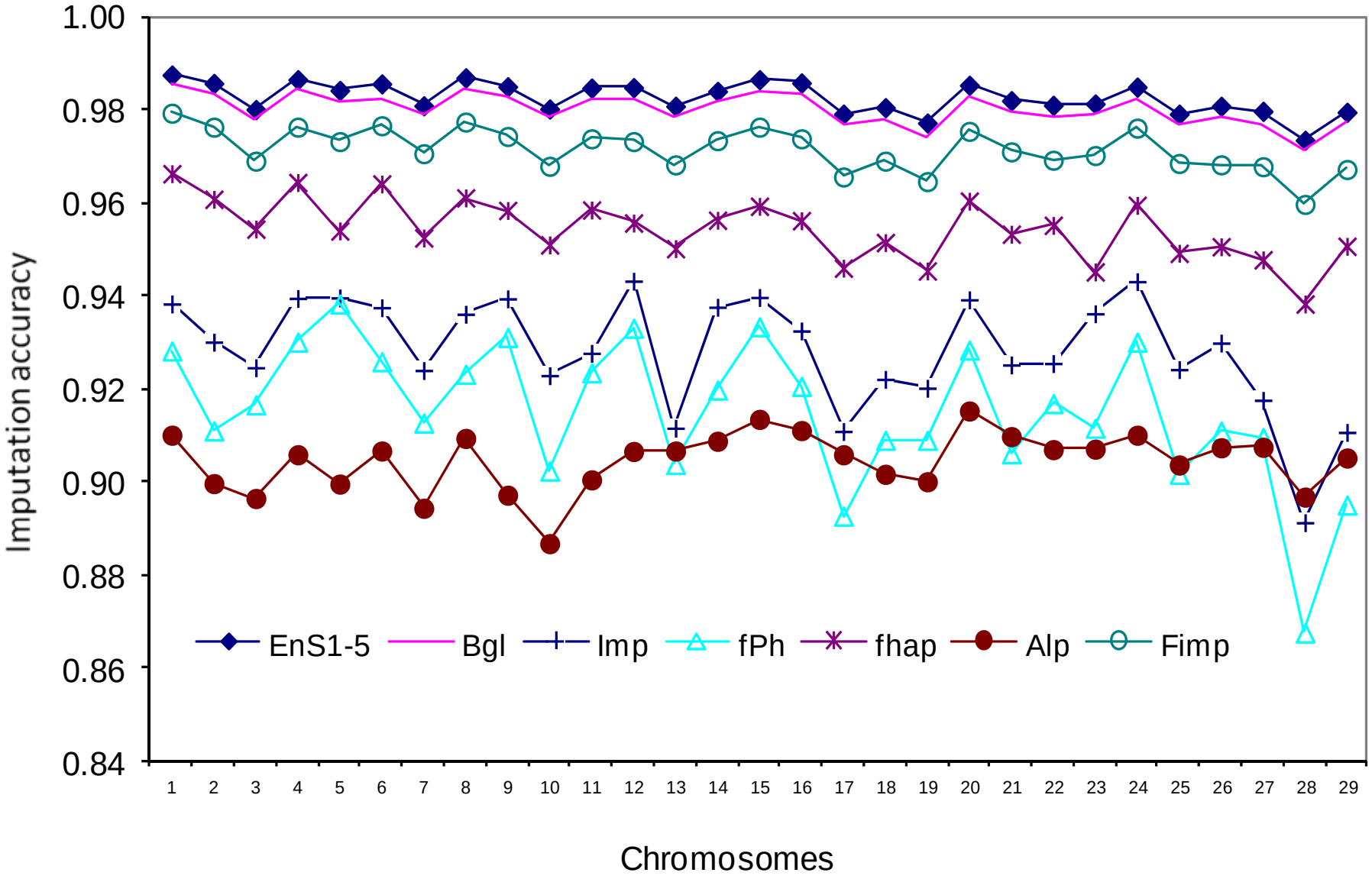
Application to Angus population: LOW DENSITY PANEL

1. Based on a 7K-genotype panel, medium-density (50K) genotypes involving 29 chromosomes were imputed for 3,078 animals using the six imputation packages and five ensemble systems.

2. All five ensembles had Beagle and AlphaImpute as the first two classifiers.

    - Bgl-Alp-fhap-Imp-fPH-Fimp
    - Bgl-Alp-Fimp-Imp-fPH-fhap
    - Bgl-Alp-Fimp-fPH-Imp-fhap
    - Bgl-Alp-Imp-fhap-fPH-Fimp
    - Bgl-Alp-Imp-Fimp-fPH-fhap

Results


1.00
0.98
0.96
0.94
0.92
0.90
0.88
0.86
0.84
Imputation accuracy
EnS1-5
Bgl
Imp
fPh
fhap
Alp
Fimp
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
Chromosomes

Conclusions

1. Bgl and Fimp were the most accurate softwares

2. Ensemble methods solve inconsistencies, and can increase imputation accuracy even further

3. Improvement depended on order of classifiers in the ensemble systems.

4. Best ensemble: those with Bgl as first classifier, followed by one or two softwares that use pedigree information during imputation.

Thank you