MUSE: A MULTI-LOCUS SAMPLING-BASED EPISTASIS ALGORITHM FOR QUANTITATIVE GENETIC TRAIT PREDICTION

DAN HE and LAXMI PARIDA
IBM T.J Watson Research
Yorktown Heights, NY
E-mail: {dhe, parida}@us.ibm.com

Quantitative genetic trait prediction based on high-density genotyping arrays plays an important role for plant and animal breeding, as well as genetic epidemiology such as complex diseases. The prediction can be very helpful to develop breeding strategies and is crucial to translate the findings in genetics to precision medicine. Epistasis, the phenomena where the SNPs interact with each other, has been studied extensively in Genome Wide Association Studies (GWAS) but received relatively less attention for quantitative genetic trait prediction. As the number of possible interactions is generally extremely large, even pairwise interactions is very challenging. To our knowledge, there is no solid solution yet to utilize epistasis to improve genetic trait prediction. In this work, we studied the multi-locus epistasis problem where the interactions with more than two SNPs are considered. We developed an efficient algorithm MUSE to improve the genetic trait prediction with the help of multi-locus epistasis. MUSE is sampling-based and we proposed a few different sampling strategies. Our experiments on real data showed that MUSE is not only efficient but also effective to improve the genetic trait prediction. MUSE also achieved very significant improvements on a real plant data set as well as a real human data set.

Keywords: Genetic Trait Prediction, Mutual Information, Epistasis, Weighted Maximum Independent Set

1. Introduction

Given its relevance in the fields of plant and animal breeding as well as genetic epidemiology,[1–3] whole genome prediction of complex phenotypic traits using high-density genotyping arrays recently received great attentions. Complex traits prediction and association are crucial to translate the findings in genetics to precision medicine. Given the genotype values encoded as $\{0, 1, 2\}$ of a set of biallelic molecular markers (we use “feature”, “marker”, “genotype” interchangeably), such as Single Nucleotide Polymorphisms (SNPs), on a collection of plant, animal or human samples, quantitative genetic traits, such as weight, height, fruit size etc. of these samples can be predicted effectively. More accurate genetic trait prediction can help breeding companies to develop more effective breeding strategies.

One of the most popular algorithms for the genetic trait prediction problem is *rrBLUP* (Ridge-Regression BLUP),[1,4] which assumes all the markers contribute to the trait value more or less. The algorithm fits an additive linear regression model where all the markers are invovled. It fits the coefficient computed for each marker, which quantifies the importance of the marker. The rrBLUP method has the benefits of the underlying hypothesis of normal distribution of the trait value and the marker effects (well suited for highly polygenic traits). Its performance is as good as or better than other popular predictive models such as Elastic-Net, Lasso, Ridge Regression,[5,6] Bayes A, Bayes B,[1] Bayes Cπ,[7] and Bayesian Lasso,[8,9] as well as other machine learning methods.

Epistasis is the phenomenon where different markers, or genes, can interact with each other. The problem of epistasis detection has been widely studied in GWAS (Genome Wide Association Studies). Lots of work, mainly greedy strategies,[10–16] have been proposed to detect epistasis effects. These greedy strategies all assume that significant epistasis effects come from only strong marginal effects, or the markers that are highly relevant to the trait. While most existing methods target epistasis detection on GWAS, some recent developments have been achieved on quantitative genetic trait prediction. He et al.[17] proposed a sampling-based method MINED to detect significant pairwise epistasis effects and to improve the genetic trait prediction. He and Parida[18] further proposed a two-stage sampling algorithm SAME to handle multi-locus epistasis effects where the number of markers involved can be greater than two. They showed that the prediction can be significantly improved with the help of epistasis. In the meanwhile SAME has a few advantages over the existing methods: It is highly scalable; It captures epistasis effects from both strong and weak marginal effects. However, SAME still has a few drawbacks: Its sampling strategy is based on random sampling where for all interactions the same number of samplings is conducted; It does not check the redundancy of the sampled interactions thus many sampled interactions might be redundant given the huge sample space; Its interaction values are based on multiplications of the genotype values, which does not distinguish all the possible genotype combinations.

In this work, we studied the multi-locus epistasis problem where the interactions with more than two SNPs are considered. We developed an efficient algorithm MUSE (Multi-locus Sampling-based Epistasis algorithm) to improve the genetic trait prediction with the help of multi-locus epistasis. MUSE conducts bidirectional sampling: It samples k-locus interactions from (k-1)-locus interactions and it decomposes the k-locus interactions into multiple (k-1)-locus interactions for further sampling. The motivation comes from the observation made in[17] that when a (k-1)-locus interaction is involved in a significant k-locus interaction, no matter whether it is a strong marginal effect or not, it is likely to be involved in multiple significant k-locus interaction. The main contribution of this work is a set of sampling strategies, including constraint-based sampling, encoding-based sampling and iterative sampling. More details will be given in the method section. Our experiments showed that MUSE is not only efficient but also effective to improve the genetic trait prediction. We also observed significant improvements on a real plant data set as well as a real human data set over the state-of-the-art methods.

2. Preliminaries

Genetic trait prediction problem is usually represented as the following linear regression model:

$$Y = \beta_0 + \sum_{i=1}^{d} \beta_i X_i + e$$

where Y is the phenotype and X_i is the i-th genotype value, d is the total number of genotypes and β_i is the regression coefficient for the i-th marker, e is the error term which usually follows a normal distribution. We call the above model *single marker model.*

Epistasis is the phenomenon where different markers can interact with each other. With the pairwise epistasis effects, the traditional linear regression model becomes the following non-linear additive model:

$$Y = \beta_0 + \sum_{i=1}^{d} \beta_i X_i + \sum_{i,j}^{d} \alpha_{i,j} X_i X_j + e \tag{1}$$

where $X_i X_j$ is the product of the genotype values of the i-th and j-th marker and it denotes the interaction of the two genotypes.

Multi-locus epistasis model is more complicated as more than two markers are involved in the interactions. When n-markers are involved in the interaction, we call it *n-locus* interaction or *n-way* interaction, which are interchangeable and we call n as the *order* of the interaction. The model is shown as below:

$$Y = \beta_0 + \sum_{i=1}^{d} \beta_i X_i + \sum_{i,j}^{d} \alpha_{i,j} X_i X_j + \cdots + \sum_{i_1,i_2,\ldots,i_n}^{d} \alpha_{i_1,i_2,\ldots,i_n} X_{i_1} X_{i_2} \ldots X_{i_n} + e \tag{2}$$

For example, the regression model involving both 2-locus and 3-locus interactions is:

$$Y = \beta_0 + \sum_{i=1}^{d} \beta_i X_i + \sum_{i,j}^{d} \alpha_{i,j} X_i X_j + \sum_{i,j,k}^{d} \alpha_{i,j,k} X_i X_j X_k + e$$

3. Multi-locus Sampling-based Epistasis Algorithm

In this work, we follow the pipeline of SAME[18] to conduct the bi-directional search. We start sampling in a forward manner from the significant (k-1)-locus interactions to obtain the significant k-locus interactions. Then we search in backwards where we take the significant k-locus interactions to guide what extra (k-1)-locus interactions we should consider to sample. This is based on the observations made in the work of He et al.[17] that if a (k-1)-locus interaction is involved in a significant k-locus interaction, no matter whether this (k-1)-locus interaction is significant or not, it is likely to be involved in multiple significant k-locus interactions.

We first use a queue Q to store the features (can be 1-locus to (k-1)-locus interactions) from which the sampling is conducted. We define *sampling* a t-locus effect as that for the t-locus effect, we randomly sample a set of single markers to be combined with the t-locus effect to obtain (t+1)-locus effects. We define a feature is *significant* if its r^2 (The square of the Pearson's correlation coefficient between the feature vector and the trait vector) to the trait is higher than a threshold s (We will show how to determine the threshold later). We use r^2 here as it is the most popular metric for genetic trait prediction (or genomic selection). We start from significant single markers and store all of them in Q. Then we sample each single marker X to obtain a set of significant 2-locus interactions where the marker X is involved in. If the 2-locus interaction is significant, we store it in Q. Then for the significant 2-locus interaction, we decompose it into two 1-locus effects, or two single markers. One of the markers will be X, the other one is either a strong or weak marginal effect. If the other marker is not in Q yet, we store it in Q so that it will be sampled later on.

We then repeat the sampling process for 2-locus interactions upto (k-1)-locus interactions. When we sample a (k-1)-locus interaction, if we obtain a significant k-locus interaction, we then decompose the k-locus interaction into k (k-1)-locus interactions and store them in Q. For example, given a significant 2-locus interaction AB, we randomly sample one single marker and by chance we obtain a significant 3-locus interaction ABF. Then we decompose it into three 2-locus interactions AB, BF, AF and store them in Q if they have not been stored yet. They will be sampled in a later stage.

3.1. *Significance Threshold*

The significance threshold s is determined dynamically. This is because we only keep the top K most significant features and thus the threshold is set naturally as the r^2 of the top K-th feature. We maintain a sorted list of the features according to their r^2 score (notice we consider both epistasis effects and single marker effects). When we check an interaction, we insert the interaction into the top-K feature set if its r^2 score is better than s and we remove the last feature from the list. If the interaction does not have a higher r^2 score than s, we do not change the list. We then set the threshold s as the r^2 score of the current K-th feature. We keep on updating the threshold as we insert more interactions, while keeping the order of the list according to the r^2 scores. As the threshold becomes higher, it becomes harder for an interaction to be selected.

3.2. *P-value*

As the feature space is extremely large, in order to avoid over-fitting problem, we also computed the p-value of the features. We ignore features with high r^2 score if the p-value of the features are not small enough. Similar to GWAS, where a typical p-value threshold is 5×10^{-8} after Bonferroni corrections for multiple testing, we used very small p-values. We observed that we can not use a fixed p-value. Instead, for larger feature space, we need to use smaller p-values. For example, for a feature space of size $O(10^7)$, we use p-value 5×10^{-6}. For a feature space of size $O(10^{10})$, we use p-value as 5×10^{-8} to 5×10^{-11}. The p-value to be used is determined by a grid search using cross validation.

3.3. *Estimate Interaction Probability*

Another thing to notice is that when we conduct the sampling, we do not sample all the single markers as it would be very time consuming for a large number of markers. We conduct an initial sampling with size f. It is shown in[17] that the scores follow a truncated normal distribution. Then using the f sampled r^2 scores, we can fit the truncated normal distribution to estimate the mean and the standard deviation. Using this distribution, and given the total number of single markers as d, we compute the probability of seeing at least one significant r^2 score out of the $O(d)$ possible interactions, where a score is significant if it is higher than the current significance threshold s. If the probability is higher than a threshold P, we will test the interactions between the marker and all the remaining markers. In order to capture as many epistasis interactions as possible, we generally use a small value for P, say 0.005.

As we can see, the performance of MUSE is heavily dependent on the sampling strategy. In SAME,[18] a simple random sampling is conducted which has been shown to have certain disadvantages. Next we introduce three sampling strategies that could significantly improve the random sampling:

3.4. *Sampling Strategies*

3.4.1. *Constraint-based Sampling*

Significant interaction selection can be considered as a feature selection process if we consider each significant interaction as a feature. A popular feature selection criteria is called MRMR (Maximum Relevance and Minimum Redundancy),[19] where the objective is to select a set of features which are maximumly relevant to the trait but minimally redundant with each other. It is shown[19] that minimizing the redundancy of the selected features leads to better prediction. In our approach, the selection of the top-k most significant interactions is equivalent to maximizing the relevance of the selected interactions to the trait. However, the redundancy of the selected interactions is not taken into consideration yet.

It is observed in[17] that a t-locus interaction might be involved in multiple significant (t+1)-locus interactions. However, these multiple significant (t+1)-locus interactions might be highly redundant with each other, as all of them share the same t-locus interaction. As the size k is fixed for the top-k most significant interactions, including many redundant interactions might not improve the prediction according to the MRMR criterion. An extreme case is that all the top-k most significant interactions are redundant, which is equivalent to using only one interaction for prediction. This will obviously lead to poor performance.

Thus here we add a constraint on the sampling process: we require every t-locus interaction involved in at most N (t+1)-locus interactions. We call N the *overlap threshold*. Therefore, any of the top-k interactions should at most overlap with N other top-k interactions, where overlap means two (t+1)-locus interactions share the same t-locus interaction. We call this sampling *Constraint-based Sampling*.

To solve the constraint-based sampling problem, we construct an *Interaction Graph*, where the nodes are (t+1)-locus interactions, the edges indicate that the two (t+1)-locus interactions share the same t-locus sub-interaction. Each node is associated with a weight, indicating the r^2 of the node to the trait. Notice we build a graph for each t. Once we moved from t to t+1, we build a new graph and delete the old graph. As an example, we can see in Figure 1, the interaction ABC share the sub-interaction AB with the interaction ABD. Thus the number of edges associated with a node indicates the degree of overlaps of the node and we call it *connectivity*. In this example, the node ABC has connectivity as 3, the node ABD has connectivity as 4. If we set the overlap threshold N as 1, we can only select the nodes that is connected to one other node.

The constraint-based sampling problem is then converted to the problem where we would like to select a set K of k nodes such that the total weights of the nodes is maximized and in the meanwhile the constraint is satisfied, namely in the node set K, there is no node with more than N edges connecting to the other nodes in the set. The problem is similar to a Weighted Maximum Independent Set (WMIS) problem. The WMIS problem seeks to select a set of

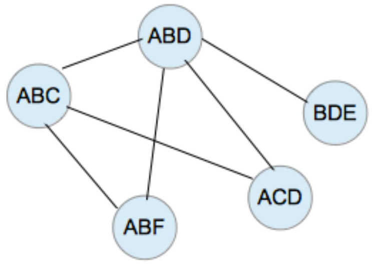

Fig. 1. An example of interaction graph.

nodes from a graph to form an independent set, where all the nodes are not adjacent, such that the sum of the weights on the nodes is maximized. As all the nodes are not adjacent in the independent set, all selected interactions are guaranteed non-overlapping. This is equivalent to allowing the degree of connectivity as 0. In our case, we set the degree of the connectivity of the selected nodes to be no greater than N.

The WMIS problem is NP-complete and what's more, it requires generating the complete interaction graph. However, in our problem, we sample the t-locus interactions one by one. Thus we conducted a greedy algorithm, where we maintain a count for every t-locus interaction. During the samplings, when we sample a t-locus interaction I and find one significant $(t{+}1)$-locus interaction, we increase the count of I by one. If the count is less than N, we keep on sampling. Otherwise we have two options:

(1) We stop the sampling immediately
(2) We do not stop the sampling, instead we continue the sampling process. However, we maintain only N significant $(t{+}1)$-locus interactions sampled from I and we call the set S. Once we identify a significant $(t{+}1)$-locus interaction I', we compare its r^2 score with the r^2 scores of the interactions in S. If its r^2 score is greater than the minimum r^2 score in S, we remove the interaction in S with the minimum r^2 score and replace it with I'.

Obviously, by taking option one, the sampling process can be terminated quickly but it may miss the significant $(t{+}1)$-locus interactions that might arrive later. By taking option two, we can guarantee that all significant $(t{+}1)$-locus interactions could be captured. However, we only store N significant $(t{+}1)$-locus interactions and thus the constraint can be satisfied. By setting N small, we could include more $(t{+}1)$-locus interactions that have different sub-interactions so that the redundancy of the top-k interactions can be reduced. In MUSE, we choose option two.

3.4.2. *Encoding-based Sampling*

By using the multiplication model and assuming the genotypes are encoded as $\{0, 1, 2\}$, a pairwise epistasis effect contains only 4 different possible values $\{0, 1, 2, 4\}$ (by pairwise multiplication of the values from $\{0, 1, 2\}$) while in reality there are nine different possible combinations of the alleles. It is not clear why a pair of markers with genotypes (0, 1) should have the same interaction value 0 as the pairs with genotypes (0, 2). Thus instead of using the values

$\{0, 1, 2, 4\}$, we could consider using nine different values {0, 1, 2, 3, 4, 5, 6, 7, 8} to differentiate the nine different combinations. However, there is no order for the combinations. For example, we can not determine the order of “AA/Bb” and “Aa/BB”. Similarly, we can not determine the order of “Aa/bb” and “aa/Bb”. Thus we do not have a systematic way to assign the nine different values {0, 1, 2, 3, 4, 5, 6, 7, 8} to the nine different combinations.

Therefore, we developed the following encoding formula:

$$encoding = \sum_{i=1}^{n} X_i \times 10^{(n-i)}$$

where n is the number of markers involved, X_i is the encoding of the genotype of the i-th marker, which is one of $\{0, 1, 2\}$. Thus instead of multiplication, we use the above encodings for the n-way epistasis interactions. For example, for pairwise interactions, assuming the encoding $\{0, 1, 2\}$ are for “AA, Aa, aa” respectively and the same for “BB, Bb, bb” respectively, we have the following encodings for the nine combinations:

$$\begin{array}{rrr} AA/BB = 0 \times 10 + 0 = 0, & AA/Bb = 0 \times 10 + 1 = 1, & AA/bb = 0 \times 10 + 2 = 2 \\ Aa/BB = 1 \times 10 + 0 = 10, & Aa/Bb = 1 \times 10 + 1 = 11, & Aa/bb = 1 \times 10 + 2 = 12 \\ aa/BB = 2 \times 10 + 0 = 20, & aa/Bb = 2 \times 10 + 1 = 21, & aa/bb = 2 \times 10 + 2 = 22 \end{array}$$

Thus using this encoding, we guarantee that different combinations of epistasis effects have different encodings and we do not need to worry about the assignment of different values to these combinations. Another benefit is that the encoding can be applied to any t-locus interactions in a systematic way. We call this sampling *Encoding-based Sampling.*

3.4.3. *Iterative Sampling*

As we are using sampling to estimate the mean and standard deviation of the normal distribution, it is critical to determine the sample size first. Given an expected error rate, we could estimate the sample size via Equation 3.

$$ME = z\frac{s}{\sqrt{n}} \tag{3}$$

Where ME is the desired margin of error, z is the z-score that depends on the desired confidence level, s is the standard deviation and n is the sample size we want to find. Given the desired margin of error and the confidence level, if we know the standard deviation or we could make a guess on it, we could compute the required sample size n.

However, our problem is much more complicated in that every t-locus interaction has different mean and standard deviation. Therefore it is not appropriate to use an universal sample size and there is no systematic way to estimate the standard deviation for each t-locus interaction.

To address the problem, we propose an iterative sampling method. In iteration one, for every t-locus interaction, we start from a small initial sample size, say, 500, and estimate mean μ_1 and standard deviation δ_1. Then we increase the sample size by 500 for every iteration. In iteration i, we estimate mean μ_i and standard deviation δ_i. If $\frac{abs(\mu_i - \mu_{i-1})}{\mu_i} \leq \epsilon$ and $\frac{abs(\delta_i - \delta_{i-1})}{\delta_i} \leq \epsilon$, where ϵ is a small number such as 0.01, or the number of iterations is greater than a prespecified number, such as 10, we say that the sampling converges.

Notice that MUSE selects the top-k most significant interactions. After the selection, we combine these interactions with the original set of single markers as a new data set. Regression methods such as rrBLUP are then applied on the new data set to make predictions. Notice k is a user defined parameter. The smaller k is, the more efficient MUSE is. Ideally k could be selected using cross-validation. However, given the extremely large feature space, it is not feasible to try all possible k's. Therefore in our work, we just simply set k as 500, a small number. Our experiments showed that by setting k as 500, we could already achieve significant improvements and yet the program is highly efficient.

4. Experimental Results

We first evaluated MUSE on a plant data set: Maize data set,[2] the Dent and Flint panels, developed for the European CornFed program. We do not consider using simulated data here as the rational for how high order multi-locus interactions contribute to the trait is indeed not clear. As the number of multi-locus interactions is extremely high when the order is high, it is not clear what is a reasonable number of the interactions that contribute to the trait.

The Maize data set indeed consists of 6 sub data sets. The Dent panel were genotyped using a 50k SNP array, which after removing SNPs with high rate of missing markers and high average heterozygosity, yielded 29,094 and 30,027 SNPs respectively. Both of them contain 261 samples and three traits. In all experiments, we perform 10-fold cross-validations and measure the average r^2 between the true and the predicted outputs, where higher r^2 indicates better performance. The parameters are learned from the training data. The baseline method is rrBLUP with single marker model using all markers. For a fair comparison, we use the top-500 most significant interactions (for k-locus interactions where $k \geq 2$) captured by MUSE and we combine them with the original set of single markers as a new data set where rrBLUP is then applied. This will indicate whether the extra information from the interactions benefit the prediction. Notice we mark the performance as "NA" for cases where no significant interaction is captured.

We evaluate the performance of MUSE with the constraint-based sampling (MUSE-C), with the encoding-based sampling (MUSE-E) and with iterative sampling (MUSE-I). We consider only 2-locus scenarios where the p-value p=5×10^{-8}. For the constraint-based sampling, overlap threshold N=5. The baseline method is rrBLUP with single marker model using all markers. As we can see in Table 1, MUSE improves the performance over rrBLUP significantly. As MINED does not use p-values as a criteria to select interactions, its performance is worse than SAME and MUSE. MUSE with the constraint-based sampling (MUSE-C) generally is able to improve the prediction accuracy over SAME, as the constraint-based sampling is able to naturally reduce the redundancy of the sampled interactions, which further leads to improvement on the prediction. MUSE with both the constraint-based sampling and the encoding-based sampling (MUSE-CE) achieve better results except for Flint Trait 3, indicating that both constraint-based sampling and encoding-based sampling are effective in improving the prediction accuracy. For Flint Trait 3, when constraint-based sampling is used, MUSE can not capture any interaction with p-value lower than 5×10^{-8}. However, after we conducted the iterative sampling, MUSE is able to capture interactions with p-value lower than 5×10^{-8} and thus MUSE-CEI achieved the best performance among all the methods. This clearly indicates the power of iterative sampling. In general combining all three sampling strategies gives us the best performance.

Table 1. The r^2 of rrBLUP, MINED, SAME, MUSE on Maize Dent and Flint data sets. We show only 2-locus scenarios where p-value p=5 × 10^{-8}, overlap threshold N=5. For MUSE-C and MUSE-CE, the number of initial sampling is 500. Here for MUSE, "-C" stands for constraint-based sampling, "-E" stands for encoding-based sampling, "-I" stands for iterative sampling.

Trait	rrBLUP	MINED	SAME	MUSE-C	MUSE-CE	MUSE-CEI
Dent Trait 1	0.59	0.59	0.615	0.65	0.65	0.67
Dent Trait 2	0.552	0.552	0.583	0.572	0.59	0.61
Dent Trait 3	0.321	0.356	0.432	0.39	0.486	0.49
Flint Trait 1	0.47	0.476	0.514	0.558	0.576	0.595
Flint Trait 2	0.301	0.316	0.356	0.364	0.419	0.429
Flint Trait 3	0.057	0.096	0.113	NA	NA	0.135

In Table 2, we evaluated 2-locus, 3-locus and 4-locus interactions for MUSE. As we have already shown that MUSE-CEI in general achieves the best performance, we only evaluate the performance of MUSE-CEI. We also varied the overlap thresholds as 5, 20, 50. The running times for MUSE-CEI are 226 sec., 979 sec. and 2056 sec. respectively. As we can see, although the size of the feature space increased exponentially, the running time of MUSE-CEI did not change much, indicating that MUSE-CEI is highly scalable due to its effective sampling process. The baseline method is again rrBLUP with single marker model using all markers.

Overall, we can see that MUSE-CEI achieved very significant improvements over rrBLUP on the single marker model (For Dent data, 21% for trait 1, 22% for trait 2, 59% for trait 3. For Flint Data, 33% for trait 1, 46% for trait 2, 138% for trait 3). We can see that both the p-value and the overlap threshold N are critical to the prediction. The best p-value and N are usually different without clear pattern for different traits and we need to use grid search to find their best values.

By varying the p-values, the prediction performance varies significantly. In general, the p-value should be small enough to achieve the best prediction. However, we do not see a clear pattern on setting the p-values. For different traits, the best p-value could be different. And it is not necessarily the case that using smaller p-value leads to better prediction accuracy. This is because smaller p-values may only produce a small set of statistically significant epistasis effects where larger p-values may produce a larger set of statistically significant epistasis effects. If the size of the set of statistically significant epistasis effects is too small and in the meanwhile they do not have very high r^2 score, they might not be able to improve the prediction performance. In the worst case, we might not be able to identify any significant k-locus interaction given a too small p-value might lead, such as Dent Trait 3 with 3-locus

$p = 5 \times 10^{-12}$ and Flint Trait 3 with 3-locus $p = 5 \times 10^{-12}$ and 4-locus $p = 5 \times 10^{-11}$. As we did not observe a clear pattern between p-values and the prediction performance, grid search with cross-validation should be applied in order to detect the best p-value.

Table 2. The r^2 of rrBLUP and MUSE on Maize Dent and Flint data set. For MUSE, we tested 2-locus, 3-locus and 4-locus interactions with different p-value thresholds. We applied all the sampling strategies. We vary the p-value and the constraint threshold N.

Methods	N=5	N=20	N=50	N=5	N=20	N=50
	Dent Trait 1			**Flint Trait 1**		
rrBLUP	0.59			0.47		
MUSE-CEI 2-locus (p=5×10^{-8})	0.67	0.581	0.58	0.595	0.591	0.568
MUSE-CEI 2-locus (p=5×10^{-10})	0.645	0.655	0.616	**0.626**	0.615	0.586
MUSE-CEI 2-locus (p=5×10^{-11})	0.63	0.693	0.656	0.56	0.583	0.556
MUSE-CEI 3-locus (p=5×10^{-10})	0.538	0.644	0.491	0.578	0.618	0.59
MUSE-CEI 3-locus (p=5×10^{-11})	0.675	**0.714**	0.59	0.617	0.62	0.57
MUSE-CEI 3-locus (p=5×10^{-12})	0.606	0.65	0.673	0.601	0.61	0.581
MUSE-CEI 4-locus (p=5×10^{-11})	0.27	0.384	0.601	0.47	0.488	0.301
	Dent Trait 2			**Flint Trait 2**		
rrBLUP	0.552			0.301		
MUSE-CEI 2-locus (p=5×10^{-8})	0.61	0.552	0.563	0.429	0.412	0.403
MUSE-CEI 2-locus (p=5×10^{-10})	0.663	0.557	0.564	0.413	0.427	0.394
MUSE-CEI 2-locus (p=5×10^{-11})	**0.671**	0.595	0.574	0.417	0.415	0.373
MUSE-CEI 3-locus (p=5×10^{-10})	0.608	0.459	0.459	0.428	**0.439**	0.418
MUSE-CEI 3-locus (p=5×10^{-11})	0.623	0.491	0.491	0.423	0.421	0.402
MUSE-CEI 3-locus (p=5×10^{-12})	0.582	0.625	0.549	0.382	0.395	0.399
MUSE-CEI 4-locus (p=5×10^{-11})	0.3	0.335	0.258	0.37	0.365	0.298
	Dent Trait 3			**Flint Trait 3**		
rrBLUP	0.321			0.057		
MUSE-CEI 2-locus (p=5×10^{-8})	0.49	0.424	0.361	0.135	0.12	0.087
MUSE-CEI 2-locus (p=5×10^{-10})	0.355	0.476	0.466	0.115	0.126	0.103
MUSE-CEI 2-locus (p=5×10^{-11})	0.332	0.397	0.465	0.097	0.067	0.048
MUSE-CEI 3-locus (p=5×10^{-10})	0.482	0.391	0.443	0.089	0.111	0.103
MUSE-CEI 3-locus (p=5×10^{-11})	0.453	0.347	0.398	0.120	**0.136**	0.119
MUSE-CEI 3-locus (p=5×10^{-12})	NA	NA	0.358	NA	NA	0.026
MUSE-CEI 4-locus (p=5×10^{-11})	0.341	**0.511**	0.444	NA	NA	0.046

Another observation is that smaller N in general leads to better performance. This clearly indicates the effects of redundancy: when N is large, we allow more redundant interactions to be selected and thus the performance drops. However, a small N may prevent selecting significant interactions as the pool of interactions to be sampled is dramatically reduced for small N. For example, for Dent Trait 3, p=5×10^{-12}, when N=5 and 20, MUSE can not capture any 3-locus significant interactions. However, when $N = 50$, MUSE could capture some 3-locus significant interactions. Similarly, for Flint Trait 3, 3-locus and 4-locus significant interactions are only captured when $N = 50$.

In summary we observed that although there is no clear pattern for the optimal p-value and overlap threshold N, we see that in general a too large N or a too small p-value lead to poorer performance. Also for higher order interactions, the number of detected significant interactions might be too small to lead improvements.

One more thing to notice is that we do not conduct biological validation on the interactions MUSE selected. This is because we assume all the interactions contribute to the trait more or less. The selected interactions also have lots of peers which have similar r^2 scores. However, we are only able to select a small set of interactions due to efficiency concerns. These interactions are selected by random chance from the pool of interactions with similar r^2 scores. But our experiments illustrated that a small set of interactions is sufficient to improve the genetic trait prediction accuracy dramatically.

Besides plant traits, we also conducted experiments on complex trait for humans. Complex traits prediction and association are crucial to translate the findings in genetics to precision medicine. We studied the data set from the Finland-United States Investigation of NIDDM Genetics (FUSION) study,[20] which is a long-term effort to identify genetic variants that predispose to type 2 diabetes (T2D) or that impact the variability of T2D-related quantitative traits. The dataset has 5000 individuals, 317503 SNPs and 10 traits. For illustration purpose, we show the results on two randomly selected traits (trait 2: HDL-cholesterol, trait 10: Height).

In Table 3, we showed the performance of MUSE on two human complex traits. We can see that in general the predictions are poor, indicating the difficulties of complex trait prediction. However, even on complex traits, we see that by integrating interactions into the predictive model, we can still achieve significant improvements. And by tuning the parameters carefully, MUSE can achieve better performance compared with existing methods. Again, we see that with relatively small N and p-value, MUSE achieved better performance.

5. Conclusion and Future Work

In this work, we studied the multi-locus epistasis problem where the interactions with more than two SNPs are considered. We developed an algorithm MUSE which is very efficient for multi-locus epistasis model. We also showed that the algorithm is very effective in improving the performance of the genetic trait prediction. Three sampling strategies are developed which could improve the overall prediction accuracy. More accurate trait predictions can be very helpful to develop breeding strategies and is crucial to translate the findings in genetics to precision medicine.

References

1. T. Meuwissen, B. Hayes and M. Goddard, *Genetics* **157**, 1819 (2001).
2. R. Rincent, D. Laloë, S. Nicolas, T. Altmann, D. Brunel, P. Revilla, V. M. Rodriguez, J. Moreno-Gonzalez, A. Melchinger, E. Bauer *et al.*, *Genetics* **192**, 715 (2012).
3. M. A. Cleveland, J. M. Hickey and S. Forni, *G3: Genes— Genomes— Genetics* **2**, 429 (2012).
4. J. Whittaker, R. Thompson and M. Denham, *Genet Res* **75**, 249 (2000).

Table 3. The r^2 of rrBLUP, MINED, SAME and MUSE on Finland data set. For MUSE, we tested 2-locus, 3-locus and 4-locus interactions with different p-value thresholds. We applied all the sampling strategies. We vary the p-value and the constraint threshold N.

Methods	N=5	N=20	N=50	N=5	N=20	N=50
	Trait HDL-cholesterol			**Trait Height**		
rrBLUP	0.11			0.03		
MINED	0.15			0.07		
SAME	0.18			0.10		
MUSE-CEI 2-locus (p=5×10^{-8})	0.14	0.15	0.16	0.04	0.03	0.04
MUSE-CEI 2-locus (p=5×10^{-10})	0.15	0.17	0.17	0.05	0.07	0.05
MUSE-CEI 2-locus (p=5×10^{-11})	0.16	0.18	0.19	0.05	0.06	0.06
MUSE-CEI 3-locus (p=5×10^{-10})	0.16	0.18	0.18	0.07	0.08	0.06
MUSE-CEI 3-locus (p=5×10^{-11})	0.17	0.2	0.19	0.08	0.11	0.1
MUSE-CEI 3-locus (p=5×10^{-12})	0.2	**0.22**	0.21	0.1	0.09	0.08
MUSE-CEI 4-locus (p=5×10^{-11})	0.12	0.15	0.18	0.1	**0.12**	0.11

5. R. Tibshirani, *Journal of the Royal Statistical Society, Series B* **58**, 267 (1994).
6. S. S. Chen, D. L. Donoho, Michael and A. Saunders, *SIAM Journal on Scientific Computing* **20**, 33 (1998).
7. K. Kizilkaya, R. Fernando and D. Garrick, *Journal of animal science* **88**, 544 (2010).
8. A. Legarra, C. Robert-Granié, P. Croiseau, F. Guillaume, S. Fritz *et al.*, *Genetics research* **93**, p. 77 (2011).
9. T. Park and G. Casella, *Journal of the American Statistical Association* **103**, 681 (June 2008).
10. K. A. Pattin, B. C. White, N. Barney, J. Gui, H. H. Nelson, K. T. Kelsey, A. S. Andrew, M. R. Karagas and J. H. Moore, *Genetic epidemiology* **33**, 87 (2009).
11. J. Marchini, P. Donnelly and L. R. Cardon, *Nature genetics* **37**, 413 (2005).
12. N. R. Cook, R. Y. Zee and P. M. Ridker, *Statistics in medicine* **23**, 1439 (2004).
13. C. Yang, Z. He, X. Wan, Q. Yang, H. Xue and W. Yu, *Bioinformatics* **25**, 504 (2009).
14. Y. Zhang and J. S. Liu, *Nature genetics* **39**, 1167 (2007).
15. G. Fang, M. Haznadar, W. Wang, H. Yu, M. Steinbach, T. R. Church, W. S. Oetting, B. Van Ness and V. Kumar, *PloS one* **7**, p. e33531 (2012).
16. X. Zhang, S. Huang, F. Zou and W. Wang, *Bioinformatics* **26**, i217 (2010).
17. D. He, Z. Wang and L. Parada, Mined: An efficient mutual information based epistasis detection method to improve quantitative genetic trait prediction, in *Bioinformatics Research and Applications*, (Springer, 2015) pp. 108–124.
18. D. He and L. Parida, Same: a sampling-based multi-locus epistasis algorithm for quantitative genetic trait prediction, in *Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics*, 2015.
19. H. Peng, F. Long and C. Ding, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **27**, 1226 (2005).
20. E. Zeggini, L. J. Scott, R. Saxena, B. F. Voight, J. L. Marchini, T. Hu, P. I. de Bakker, G. R. Abecasis, P. Almgren, G. Andersen *et al.*, *Nature genetics* **40**, 638 (2008).