Application of optimization and simulation models in Genomic prediction and Genomic selection

by

Fatemeh Amini

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Industrial Engineering

Program of Study Committee:
Guiping Hu, Major Professor
Kris M. De Brabanter
Sigurdur Olafsson
Derrick K. Rollins
Lizhi Wang

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2022

DEDICATION

I would like to dedicate this dissertation to my supportive parents and to my beloved husband, Keyvan Mollaeian who was always beside me throughout this journey.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

ACKNOWLEDGMENTS

I would like to take this opportunity to express my thanks to those who helped me with various aspects of conducting research and the writing of this dissertation. First and foremost, Dr. Guiping Hu for her guidance, patience and support throughout this research and the writing of this dissertation. Her insights and words of encouragement have often inspired me and renewed my hopes for completing my Ph.D. education. I would like to thank Dr. Wang for his guidance throughout the research journey and his critical thinking.

I would additionally like to thank my other committee members, Dr. Olafsson, Dr. Rollins, and Dr. Brabanter for their inspirational teaching style which taught me how to deal with research problems.

ABSTRACT

Population growth, climate change, and biofuel consumption for agricultural products have been estimated to be doubled by 2050. Due to population growth, the agricultural production system has to become ever efficient and robust to ensure the food security. To address this challenge, crop improvement and plant breeding process have to be employed to enhance the quality and quantity of crop productions. In this dissertation, we adopt simulation and data analytics methods to tackle a few challenges in the crop improvement and breeding process. First of all, we address the high-dimensionality issue in the genetic data and introduce a novel two-layer feature selection method to reduce the feature space dimension while improving the genetic prediction accuracy in genomic selection algorithms. Furthermore, we design a realistic simulator that can be adopted to simulate the breeding process where the goal is to imitate the uncertainty of nature to provide reliable genetic outcome. Moreover, from the decision making perspective, we propose a new selection strategy called, look ahead trace back selection, that aims at improving the performance of a single trait at the end of breeding cycle. Additionally, a new optimization model is introduced to maximize the performances of multiple traits simultaneously. Multiple challenges in the breeding process make its improvement more difficult. Two of these challenges, namely, improving prediction accuracy in genomic prediction, and uncertainties due to the recombination events in the mating process are addressed. To address the first challenge, we tune the hyper-parameters of the adopted prediction methods inside cross-validation loops and the proposed two-layer feature selection parameters constrained by the available computational capacity. To address the second challenge, we develop a comprehensive simulation platform in which multiple simulation runs are conducted independently to ensure the robustness of the results of any proposed approaches in comparison with the conventional methods.

This dissertation includes 5 chapters in which chapter 1 presents a general overview along with the problem statements and a summary of contributions. In chapter 2, we develop a two-layer feature selection, a hybrid of wrapper-embedded method to reduce the feature dimension in genomic prediction while maintaining/improving the prediction accuracy. In chapter 3, we design look ahead trace back selection algorithm that improve the genetic gain in the breeding process. Moreover, a realistic opaque simulator is introduced in this chapter which accounts for nature uncertainties. In chapter 4, a L-shaped selection algorithm is proposed to improve the genetic gain in multi-trait genomic selection. The aim of this algorithm is to maximize multiple traits at the same time by capturing all Pareto optimal individuals and maintain the population diversity. In chapter 5, we analyze the performance of integrating the proposed two-layer feature selection in improving the genetic gain in different genomic selection algorithms. Moreover, a comprehensive comparison framework has been formulated that can integrate different prediction methods, multiple genomic selection algorithms and different simulation methods, such as transparent and opaque simulator.

CHAPTER 1. GENERAL INTRODUCTION

1.1 Overview

Increasing human population, meat and dairy consumption from growing affluence, and biofuel consumption have resulted in a rising demand for food and put more pressure on global agriculture to increase yield [Ray et al. (2013); Delgado et al. (2019)]. It is anticipated that the food demand will be doubled by 2050, therefore, significant crop production will be necessary for food security Foley et al. (2011). To address this challenge, the agriculture community has focused on improving the genetic gain given the limited available resources. Plant breeding introduced by Fehr (1991) has made significant contributions in increasing the quantities and qualities of the crops. Plant breeding is the art of selecting the most desirable plants or seeds to improve the genetic gain of plants, instead of randomly taken what nature provided Fehr (1991). The effectiveness of plant breeding has contributed to the increases the crop genetic gain over generations that assists the global food production. At the beginning of plant breeding era, breeders select the superior individual visually based on their phenotype performance. Although rapid development of phenotyping technologies helped breeders to collect and analyze phenotypic data, it may not be time and cost efficient [Fehr (1991)]. Therefore, Genomic Selection (GS) was introduced by Meuwissen et al. (2001) to overcome this issue. The significant merit of GS over PS lies in the fact that genetic (DNA) information are expected to be achieved faster than phenotypic data. It estimates the breeding value of individuals using a set of markers distributed across the genome to predict the performance of quantitative traits [Meuwissen et al. (2001); Goddard (2009)]. The effectiveness of GS, significantly depends on its genomic prediction ability and the selection and mating strategies. Genomic prediction methods have been used in GS to predict the plants' phenotype based on their genotype information. Afterward, a selection and mating strategy is adopted to select and mate the elite parents based on their predicted

phenotypic performances. This dissertation will focus on improving both mentioned aspects of GS to improve the crop production by enhancing the genetic gain.

1.2 Problem Statement

Although the rapid development of genotyping and phenotyping technologies have led to increasingly comprehensive databases, relatively large number of markers (p) in comparison with limited number of phenotyped individuals remains a challenging problem in genomic prediction Meuwissen et al. (2001). This curse of dimensionality not only increase the cost of data storage, but also it might decrease the prediction accuracy. Therefore, a feature selection method is needed to decrease the dimensionality of genotype data to avoid overfitting while maintaining the prediction accuracy in phenotype prediction [Crossa et al. (2014); Bhat et al. (2016)]. Having a well-performing genomic prediction model does not guarantee of achieving the desired genetic gain over generations. Identification of elite individuals within the population that should be selected and crossed as breeding parents to produce the next generation of individuals plays a significant role on improving the genetic gain. Many researchers have address this problem [Daetwyler et al. (2015); Goiffon et al. (2017); Moeinizade et al. (2019)], however, a simulation platform is necessary to imitate the nature as close as possible to simulate the breeding process. This simulation platform is required to capture the uncertainty of nature. Adopting a good feature selection method in the genomic prediction along with a well-performing selection and mating strategy will not always result in best outcome in terms of the genetic gain. The interplay between genotypes and phenotypes is complex and varies widely from trait to trait, therefore the best general GS does not exist to performs well in all the cases [Whalen et al. (2020)]. So, a comprehensive GS platform is needed to adopt different prediction model and different selection strategies and selects the outperforming one regarding the crop/traits. Improving the genetic gain of one trait would not be economically worthy regarding the breeders' perspectives. In real world, breeders are more likely to invest on a GS methods that can improve multi traits of a crop or maintain an acceptable level of multi traits over multiple generations. Although, there is

significant improvements in genetic gain using different selection strategies, a few of them have focused on obtaining sustainable genetic gain in multi traits. Multi-trait GS may include multiple and even competing objectives regarding to each trait and these complex decisions can be supported by considering the multi-objective optimization principles [Akdemir et al. (2019); Cowling et al. (2019)]. Therefore, the problem of achieving a robust, well-performing and time-efficient optimization model is being still under consideration. To address the aforementioned challenges in genomic selection, we develop a two-layer feature selection method to decrease the genotype dataset's feature space while maintaining the prediction accuracy in genomic prediction. In addition, we propose an opaque simulator which imitates the uncertainty of nature along with a new selection strategy that outperformed the exiting methods. Moreover, we develop a time and cost-efficient optimization model to handle the multi-trait genomic selection. And finally we design a holistic comparison platform in which different feature selection and prediction methods along with the existing selection methods are provided to facilitate the selection of the best GS for the desired trait.

1.3 Summary of Contributions and Dissertation Structure

The objective of this dissertation focuses on improving genetic gain in crops to fill the gaps between analytics and plant breeding using operations research, simulation-based optimization and data analytic methods. This dissertation includes four papers as follows: The first paper address the curse of dimensionality challenge in genetic data. A feature selection method has been introduced to eliminate irrelevant SNP markers in predicting the RNA-sequencing of multiple genes while maintain the prediction accuracy. It is a two-layer feature selection method that has Genetic Algorithms (a wrapper method) as its first layer and Elastic Net (an embedded method) as its second layer to refine the features as to result in the most informative features while maintaining the prediction accuracy. This method has been adopted on Maize data sets to refine the SNP markers to the most relevant ones in predicting the RNA-sequencing of different genes of individuals. The results demonstrated that, it outperforms the existing feature selection methods

in terms of feature reduction rate and prediction accuracy. The second paper develops a new selection strategy along with an opaque simulator in GS to not only improve the genetic gain but also simulate the breeding process in a more realistic simulated platform. The selection strategy named Look Ahead Trace Back (LATB) approach is an extension of look ahead selection that selects breeding parents based on their chance of producing elite progeny over generations in future and not necessarily from those who performed well in the current generation. In addition, an opaque simulator that is partially observable, explicitly capture both additive and non-additive genetic effects, and simulate uncertain recombination events more realistically, despite the existing GS simulation setting that are transparent. In addition, the paper includes the performance of existing GS methods under the opaque simulator as well.

The third paper focuses on improving the genetic gain of multiple traits at the same time considering different significance of each trait. The main contribution of this paper is to adopt a non-convex objective function to select the breeding parents, while it has the same complexity as the index selection (linear objective function), it is able to produce better-performing and more diverse progeny in the final generation. The proposed method has been tested on multiple traits with different characteristics, simultaneously. Regardless of the fact that the considering traits are continuous or binary, the method outperforms the index selection in improving the genetic gain and genetic diversity in the final generation.

In the last paper, the performance of the proposed two-layer feature selection method on improving the crops genetic gain has been analyzed. Although in the first paper, the effectiveness of adopting the two-layer feature selection on feature space dimension reduction and prediction accuracy improvement has been shown, this paper focuses on genetic gain improvement of using the two-layer feature selection method in different scenarios. A comprehensive comparison platform is designed to incorporate different scenarios such as, different genomic selection algorithms with different prediction methods under different simulators using the two-layer feature selection. This allows breeders and analyst to assess the performance of any genomic

selection algorithms with any prediction methods under some pre-defined simulation platforms and select the best-performing one regarding the target trait..

To summarize the contributions of this dissertation, we have addressed the curse of dimensionality challenge in the genetic data with proposing a novel two-layer feature selection method. Also a new genomic selection strategy with an opaque simulator that imitates the nature has been proposed to not only improve the genetic gain but also simulate the nature as closely as possible. Moreover, the challenge of multi-trait genomic selection has been addressed with introducing a non-convex optimization model. Furthermore, the effectiveness of the proposed feature selection model on improving the genetic gain of different genomic selection algorithms under an opaque and a transparent simulator has been discussed.

The first paper, is presented in Chapter 2 [Amini and Hu (2021)]. The second paper, in Chapter 3 [Amini et al. (2021)]. The third paper is accepted in *Genetics* and the last paper is under preparation presented in Chapter 4 and 5, respectively. Finally, we conclude this dissertation and suggest future directions in Chapter 6.

1.4 References

Akdemir, D., Beavis, W., Fritsche-Neto, R., Singh, A. K., and Isidro-Sánchez, J. (2019). Multi-objective optimized genomic breeding strategies for sustainable food improvement. *Heredity*, 122(5):672–683.

Amini, F., Franco, F. R., Hu, G., and Wang, L. (2021). The look ahead trace back optimizer for genomic selection under transparent and opaque simulators. *Scientific Reports*, 11(1):1–13.

Amini, F. and Hu, G. (2021). A two-layer feature selection method using genetic algorithm and elastic net. *Expert Systems with Applications*, 166:114072.

Bhat, J. A., Ali, S., Salgotra, R. K., Mir, Z. A., Dutta, S., Jadon, V., Tyagi, A., Mushtaq, M., Jain, N., Singh, P. K., et al. (2016). Genomic selection in the era of next generation sequencing for complex traits in plant breeding. *Frontiers in Genetics*, 7:221.

Cowling, W. A., Li, L., Siddique, K. H., Banks, R. G., and Kinghorn, B. P. (2019). Modeling crop breeding for global food security during climate change. *Food and Energy Security*, 8(2):e00157.

Crossa, J., Perez, P., Hickey, J., Burgueno, J., Ornella, L., Cerón-Rojas, J., Zhang, X., Dreisigacker, S., Babu, R., Li, Y., et al. (2014). Genomic prediction in CIMMYT maize and wheat breeding programs. *Heredity*, 112(1):48–60.

Daetwyler, H. D., Hayden, M. J., Spangenberg, G. C., and Hayes, B. J. (2015). Selection on optimal haploid value increases genetic gain and preserves more genetic diversity relative to genomic selection. *Genetics*, 200(4):1341–1348.

Delgado, J. A., Short Jr, N. M., Roberts, D. P., and Vandenberg, B. (2019). Big data analysis for sustainable agriculture on a geospatial cloud framework. *Frontiers in Sustainable Food Systems*, 3:54.

Fehr, W. (1991). *Principles of cultivar development: theory and technique*. Macmillian Publishing Company.

Foley, J. A., Ramankutty, N., Brauman, K. A., Cassidy, E. S., Gerber, J. S., Johnston, M., Mueller, N. D., O'Connell, C., Ray, D. K., West, P. C., et al. (2011). Solutions for a cultivated planet. *Nature*, 478(7369):337–342.

Goddard, M. (2009). Genomic selection: prediction of accuracy and maximisation of long term response. *Genetica*, 136(2):245–257.

Goiffon, M., Kusmec, A., Wang, L., Hu, G., and Schnable, P. S. (2017). Improving response in genomic selection with a population-based selection strategy: optimal population value selection. *Genetics*, 206(3):1675–1682.

Meuwissen, T., Goddard, M., Hayes, et al. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, 9(7):2123–2133.

Ray, D. K., Mueller, N. D., West, P. C., and Foley, J. A. (2013). Yield trends are insufficient to double global crop production by 2050. *PloS one*, 8(6):e66428.

Whalen, I., Banzhaf, W., Al Mamun, H. A., and Gondro, C. (2020). *Evolving SNP Panels for Genomic Prediction*, pages 467–487. Springer International Publishing, Cham.

CHAPTER 2. A TWO-LAYER FEATURE SELECTION USING GENETIC ALGORITHM AND ELASTIC NET

Authors: Fatemeh Amini * and Guiping Hu *

* Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript published in *Expert Systems with Applications* journal

2.1 Abstract

Feature selection, as a critical pre-processing step for machine learning, aims at determining representative predictors from a high-dimensional feature space data set to improve the prediction accuracy. However, the increase in feature space dimensionality, comparing to the number of observations, poses a severe challenge to many existing feature selection methods considering computational efficiency and prediction performance. This paper presents a new two-layer feature selection approach that combines a wrapper and an embedded method in constructing an appropriate subset of predictors. In the first layer of the proposed method, Genetic Algorithm(GA) has been adopted as a wrapper to search for the optimal subset of predictors, which aims to reduce the number of predictors and the prediction error. As one of the meta-heuristic approaches, GA is selected due to its computational efficiency; however, GAs do not guarantee the optimality. To address this issue, a second layer is added to the proposed method to eliminate any remaining redundant/irrelevant predictors to improve the prediction accuracy. Elastic Net(EN) has been selected as the embedded method in the second layer because of its flexibility in adjusting the penalty terms in the regularization process and time efficiency. This two-layer approach has been applied on a Maize genetic data set from NAM population, which consists of multiple subsets of data sets with different ratios of the number of predictors to the number of observations. The numerical results confirm the superiority of the proposed model.

2.2 Introduction

Advances in information technology have led to increasingly large data sets in both number of instances and number of predictors, such as applications in text mining and bioinformatics [Guyon and Elisseeff (2003)]. One significant problem for prediction with high dimensional data is that the number of predictors exceeds the number of observations [Yu and Liu (2003)]. In these situations, some of the predictors may be redundant, irrelevant, and harmful for the model training [Cilia et al. (2019); Xue et al. (2015)]. Redundant predictors provide information that is already represented with other predictors, while irrelevant predictors do not contribute to model training [Welikala et al. (2015)]. In fact, these predictors unnecessarily increase the computation time and deteriorate the performance of the classification/regression models [Lin et al. (2015); Oztekin et al. (2018)]. Thus, extracting a smaller subset of predictors with most relevant predictors would be essential since it saves time in data collection and computation, and avoids overfitting problem in the prediction models [Aytug (2015)]. Feature selection methods have been introduced to filter out the irrelevant and redundant predictors to achieve the smallest, most powerful subset of predictors in order to not only reduce the computation time but also improve the prediction accuracy [Huang and Wang (2006); Lin et al. (2015)].

Feature selection approaches can be categorized into three broad classes: the filter methods, the wrapper methods, and embedded methods. For filter methods, each predictor is evaluated with a statistical performance metric and then ranked according to its performance indicator. Truncation selection is then applied to select the top-performing features before applying machine learning algorithms. Filter methods serve as a pre-processing step since they do not consider the complex interactions between predictors and are independent of learning algorithms [Guyon and Elisseeff (2003); Hu et al. (2015)]. These methods are computationally efficient; however, they suffer from getting stuck in local optimum because the complex interactions among predictors may have been ignored [Cheng et al. (2016); Hong and Cho (2006); Welikala et al. (2015)]. The second class, wrapper methods, incorporate prediction models into a predetermined objective function that evaluates the appropriateness of the predictor subsets through an exhaustive search

[Kabir et al. (2010)]. Although wrapper methods consider the interaction among predictors, they are not as computationally efficient as filter methods because of the larger space to search [Cilia et al. (2019); Hall (1999); Hu et al. (2015); Kabir et al. (2010)]. The issue arises that evaluating all possible 2^P predictor combinations is neither effective nor practical in terms of computation time, especially when the number of predictors, P, gets larger [Cilia et al. (2019); De Stefano et al. (2008); Peng et al. (2005)]. Feature selection is among NP-hard problems in which the search space grows exponentially as the number of predictors increases [Hu et al. (2015); Jeong et al. (2015)]. The third class, embedded methods, are more efficient than wrappers since they incorporate feature selection as part of the training process and select those features which contribute the most to the model training [Guyon and Elisseeff (2003)]. Regularization methods, also called penalization methods, are the most common embedded methods. These methods would push the model toward lower complexity by eliminating those predictors with coefficients less than a threshold. The underlying assumption of regularization methods is the linear relation between predictors and response variable, which may not hold, especially in high dimensional data sets.

To avoid the aforementioned shortcomings of the existing feature selection methods, a two-layer feature selection method has been proposed in this study. The proposed method is a hybrid wrapper-embedded approach, which complements wrapper and embedded methods with their inherent advantages. For the wrapper part, a population-based evolutionary algorithm (the GA), has been adopted in the first layer of the proposed method due to the efficiency in the searching process. It can achieve excellent performance as well as avoid the exhaustive search for the best subset of predictors. This reduces the computation time of the wrapper component while finds near-optimal solution through an efficient process. However, as a meta-heuristic algorithm, there is no guarantee finding the optimal solution. Therefore, in the second layer, an embedded method is applied on the reduced subset of predictors to eliminate those remaining irrelevant predictors[Jeong et al. (2015)]. The assumption of linearity between a reduced subset of predictors and response variable is much relaxed than linearity assumption among the full original predictors and response variable. In the implementation of this proposed two-layer feature

selection scheme, Elastic Net (EN) is selected as the training model because of its flexibility in adjusting the penalty terms in the regularization process and time efficiency.

The rest of the paper is organized as follows. Section 2 describes the related literature, the motivations of this study, along with the contributions of this paper. Section 3 provides background on the mathematical model of the GA, EN method, and the proposed two-layer approach. A description of the case study, which the proposed method has been applied to is also covered in Section 3. Section 4 explains the detailed experimental setting, discusses the results of the two-layer Genetic Algorithm-Elastic Net (GA-EN) method, and compares the results with selected counterparts in terms of the prediction accuracy. Finally, section 5 concludes this study and suggests future research directions.

2.3 Related Work

Genetic Algorithms, as a meta-heuristic search strategy, have mainly been adopted to find the optimal hyper-parameters for machine learning algorithms. A modified genetic algorithm, known as a real-value GA, was constructed to find the optimal parameters for a Support Vector Machine (SVM) algorithm. The algorithm was then applied to predict aquaculture quality [Liu et al. (2013)]. Similarly, the set of optimal parameters for both SVM and Random Forest (RF) have been found using GA. The SVM and RF models were then applied to construct a fire susceptibility map for Jiangxi Province in China [Hong et al. (2018)]. SVM has been adopted widely for feature selection. A wrapper method using SVM with a specific kernel has been designed to iteratively eliminate those features with the least impact on classification importance until a stop criteria has been met [Maldonado and Weber (2009)]. Furthermore, an embedded feature selection method using SVM with Gaussian kernel was proposed to mainly handle the case of class-imbalance and high-dimensionality in the feature space in classification problems [Maldonado and López (2018)].

Recently, the applications of GA are going beyond the hyper-parameter tuning of prediction models. They have been adopted as a search strategy inside the feature selection methods

because of their ability to avoid exhaustive search that reduces high dimensional feature spaces. So far, many studies have combined GA with machine learning algorithms to improve prediction accuracy, especially in classification problems. Cerrada et al. (2016) implemented GA to reduce the feature space to construct a more efficient RF model that predicts multi-class fault diagnosis in spur gears. As Oztekin et al. (2018) illustrated, GA was combined with three different machine learning methods, K-Nearest Neighbor (KNN), SVM, and Artificial Neural Network (ANN) to improve the prediction accuracy of the patient quality of life after lung transplantation. Although the GA-SVM model outperforms both the GA-KNN and GA-ANN approach, these last two models still yield high prediction accuracy. Among the hybrid methods of different machine learning with GA, deep synergy adaptive-moving window partial least square-genetic algorithm (DSA-MWPLS-GA), was designed to obtain accurate predictions of common properties of coal [Wang et al. (2019)]. Additionally, Cheng et al. (2016) combined a GA with a Successive Projections Algorithm to select the most relevant wavelengths. The most five important wavelengths were then used to establish Least-Squares Support Vector Machine (LS-SVM) and Multiple Linear Regression (MLR) models in order to predict drop loss in grass carp fish. This is further evidenced by Cornejo-Bueno et al. (2016) in which a new hybrid feature selection method was proposed. The method combines Grouping Genetic Algorithm with an Extreme Learning Machine approach (GGA-ELM). The GGA was used as a search strategy to find the ideal subsets, while the ELM was implemented as the GGA's fitness function to evaluate the candidate subsets. The GGA-ELM model yielded a significantly smaller RMSE value than the ELM model using all features, validating that combining feature selection approaches can improve overall model performance. The model was then applied to marine energy data sets to predict the significant wave height and energy flux. Moreover, most of the hybrid approaches have been applied to classification problem and not much attention has been devoted to regression problems. This serves as one of the primary motivations in this study.

It should be noted that GAs can only be combined with supervised learning algorithms with a response variable. For data sets without response variable, clustering, and classifying based on

the feature space should be applied before implementing GAs. Sotomayor et al. (2018) firstly, applied K-means clustering approach to classify the water station into two types based on their associated water quality. A hybrid model was then developed with K-nearest neighbor and GA to reduce the dimension of feature space and achieve higher prediction accuracy.

One of the most common concerns on high-dimensional data sets is that models are prone to overfitting, which is aggravated as the ratio of predictors to observations increases [Guyon and Elisseeff (2003)]. It can be observed that the performance of a feature selection mechanism can be improved if it is carefully designed in conjunction with another filter or wrapper approach, as it will further reduce the feature space and facilitate the design of a more efficient and accurate prediction model. Therefore, two-layer feature selection approaches have been proposed to extract the best subset from the selected predictors obtained from the first layer feature selection. Hu et al. (2015) proposed a hybrid filter-wrapper method that uses a Partial Mutual Information (PMI) based filter method as the first layer to remove the unimportant predictors. Once the dimensions of feature space are reduced, a wrapper process consisting of a combination of a SVM and the Firefly Algorithm (FA), which is a population-based meta-heuristic technique, was then applied on the reduced feature space. However, since filter methods, such as the PMI approach, do not take into account the possible dependencies/interactions among predictors, the performance, when applied for high dimensional feature spaces, is not satisfactory. This is due to the fact that two factors may be independently counted as irrelevant and/or redundant predictors when keeping both in the model could result in a performance gain. In this paper, the proposed algorithm adopts a wrapper, as its first layer of feature selection and an embedded method, EN regularization algorithm, as the second layer in order to reduce feature space dimension while improving the prediction accuracy.

The contributions of this study can be summarized as follows. Firstly, unlike most existing studies, which focused on classification problems, our proposed model has combined a wrapper (GA) and an embedded feature selection method (EN) that focuses on regression problems. Elastic Net is selected as the embedded method and it is the generalized form of LASSO and

Ridge regression that has been adopted widely for high-dimensional feature space regression problems. Secondly, the fitness function of Genetic Algorithm has been designed to incorporate root mean square error ($RMSE$) minimization as well as reduction of feature space dimension.

2.4 Methods and Materials

This section describes the proposed two-layer feature selection method. In the first layer, a wrapper has been designed to select the best subset of predictors with the lowest prediction error while includes as few predictors as possible. This is done with a GA-based search strategy. In the second layer, EN has been applied to further eliminate the remaining redundant/irrelevant predictors to improve the prediction accuracy, using the best subset of predictors outputted from the first layer. Additionally, the case study adopted to validate the proposed method has been described in this section.

2.4.1 Elastic Net Regularization Method

EN regularization is a modification of the multiple linear regression approaches designed to solve high-dimensional feature selection problems [Fukushima et al. (2019)]. Using two penalty terms (L1-norm and L2-norm), the EN selects variables automatically and performs continuous shrinkage to improve the prediction accuracy. This method works like a stretchable fishing net that keeps "all big fish", i.e., important predictors and eliminates those irrelevant ones [Park and Mazer (2018); Zou and Hastie (2005)].

Suppose that we have $p = 1, \ldots, P$ predictors denoted by $x_1, \ldots, x_P$, an estimate of the response variable Y can be written as $\hat{Y} = \beta_0 + \beta_1 x_1 + \ldots + \beta_P x_P$, based on linear regression. The coefficients ($\hat{\beta} = [\beta_0, \ldots, \beta_P]^\top$) are calculated by minimizing the sum of the squares of the error residuals in Eq.(2.1).

$$SSE = ||Y - X\hat{\beta}||^2 \tag{2.1}$$

Where:

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ \vdots \\ y_N \end{bmatrix} \quad \text{and} \quad X = \begin{bmatrix} 1 & x_{11} & \dots & x_{1P} \\ 1 & x_{21} & \dots & x_{2P} \\ \vdots & & \ddots & \\ 1 & x_{N1} & \dots & x_{NP} \end{bmatrix}$$

In the case where the dimensions of the feature space are higher than the number of observations, the coefficients are calculated by minimizing the L function (Eq.(2.2)) instead of minimizing SSE [Wei et al. (2019)]:

$$L = SSE + \alpha\rho||\hat{\beta}||_1 + \alpha(1-\rho)||\hat{\beta}||^2 \tag{2.2}$$

Where $||\hat{\beta}||_1$, $||\hat{\beta}||^2$, α , and ρ are defined in Eqs.(3-5), respectively.

$$||\hat{\beta}||_1 = \sum_{p=0}^{P} |\beta_p| \tag{2.3}$$

$$||\hat{\beta}||^2 = \sum_{p=0}^{P} \beta_p^2 \tag{2.4}$$

$$\alpha > 0 \ , \quad 0 \leq \rho \leq 1 \tag{2.5}$$

The degree to which model complexity is penalized is controlled by weighting terms α and ρ. As the outcome of the Elastic Net is affected by α and ρ, tuning them should be done within the learning process. Two special cases for EN are when $\rho = 1$ and $\rho = 0$. When $\rho = 1$, EN regression is reduced to *LASSO*, which aims to reduce the number of non-zero linear coefficients to zero in order to create a sparse model. When $\rho = 0$, EN regression is reduced to ridge regression, which allows the model to include a group of correlated predictors to remove the limitation of the number of selected predictors [Chen et al. (2019); Park and Mazer (2018); Wei et al. (2019)]. It is

shown that as EN is able to select a subset of highly correlated features, it avoids the shortcoming of high-dimensional feature selection when solely using $LASSO$ or ridge regression methods [Zou and Hastie (2005)].

2.4.2 Genetic Algorithms

GAs are one of the meta-heuristic search methods that implement a probabilistic, global search process that emulates the biological evolution of a population, inspired by Darwin's theory of evolution [Cheng et al. (2016); Welikala et al. (2015)]. GAs are powerful tools for achieving the global optimal solution of large-scale problems [Cerrada et al. (2016); Liu et al. (2013)]. The GA process can be described in these steps:

1. Individual encoding: Each individual is encoded as binary vector of size P, where the entry $b_i = 1$ states for the predictor p_i that is defined for that individual, $b_i = 0$ if the predictor p_i is not included in that particular individual ($i = 1, \ldots, P$) [Cerrada et al. (2016)].

2. Initial population: Given the binary representation of the individuals, the population is a binary matrix where its rows are the randomly selected individuals, and the columns are the available predictors. An initial population with a predefined number of individuals is generated with a random selection of 0 and 1 for each entry [Cerrada et al. (2016)].

3. Fitness function: the fitness value of each individual in the population is calculated according to a predefined fitness function [Welikala et al. (2015)]. Individual with the lowest prediction error and fewer predictors have been selected for next generation.

4. Applying genetic operators to create the next generation.

    - Selection: The elite individuals that have been selected based on their fitness value, are selected as parents to produce children through crossover and mutation processes. In this study, instead of selecting all parents from the highest qualified individuals, a random individual will also be added to the parent pool in order to maintain

generational diversity. Each pair of parents produces some children to create the next generation, which has the same size as the initial population. To stabilize the size of each generation, Eq. (2.6) should be satisfied.

$$\frac{\#ofBS + \#ofRS}{2} * \#ofchildren = \text{initial population size.} \tag{2.6}$$

BS is the best selected individuals, while RS is the randomly selected individuals.

- Crossover: It is a mechanism in which the new generation is created by exchanging entries between two selected parents from the previous step. A single point crossover technique has been used in this paper [Liu et al. (2013); Welikala et al. (2015)].
- Mutation: This operation is applied after crossover and determines if an individual should be mutated in next generation or not and makes sure no predictors have been removed from GA's population permanently [Brown and Sumichrast (2005)].

5. Stop criteria: Two stop criteria are widely used in GAs. The first one, used in this study, is reaching the maximum number of generations. The other one is the lack of fitness function improvement in two successive generations [Cheng et al. (2016)]. Steps 2 and 3 are performed iteratively until the stop criterion is met.

2.4.3 Proposed GA-EN feature selection approach

The proposed feature selection method has two layers. In the first layer, GA has been implemented to reduce the search space to find the best subset of predictors. Thus a small subset of predictors can be identified to reduce the computational cost and improve prediction accuracy. In the second layer, EN regularization method is adopted to eliminate those remaining redundant predictors in the feature space given in the first layer. The reason for choosing EN as the regressor is that not only the EN makes use of shrinkage to reduce the high-dimensional feature space, but also it tends to outperform other models in regression problems. Thus, the probability

of having redundant/irrelevant predictors in the final model would decrease, resulting in a prediction model without any significant sign of overfitting.

The architecture of the proposed two-layer feature selection method is described in Figure 2.1. Following pre-processing the data, using k-fold cross-validation technique, data is split into k folds in which $k-1$ folds are considered as the training set and 1 fold as the validation set. y_i and $\hat{y}_i$ are the actual value and the predicted value of response variable in the validation set included N_t observations. This procedure is repeated k times such that each fold will be used once for validation. Averaging the $RMSE$ over the k trials would provide an estimation of the expected prediction error (Eq.(5.6)), which is the performance evaluation metric. The main idea behind the k-fold cross-validation is to minimize any potential bias of random sampling of training and validation data subset [Oztekin et al. (2018)].

$$RMSE_{CV} = \frac{1}{\#folds} \sum_{1}^{\#folds} \frac{1}{N_t} \sum_{i=1}^{N_t} \sqrt{(y_i - \hat{y}_i)^2} \tag{2.7}$$

In the first layer of the proposed method, the training set is fed into GA to search for the best subset of predictors. Throughout the GA search procedure, after building the initial population, individuals are ranked according to their fitness values and the highest ranked ones are more likely to be selected in the selection process to create the next generation. The GA runs multiple times, and each iteration outputted a best subset of predictors. Then, the predictors that have been repeated frequently in the best subset of predictors in each iteration of GA would be included in the final subset of predictors. Therefore, the most important predictors can be identified as those repeated more often in the best subset given by GA. A threshold is considered to specify how often a predictor should be repeated in the best subset of GA to be included in the final subset of predictors given in the first layer of the proposed method. The higher this threshold is defined, the stricter the model in selecting important predictors.

In the second layer, the EN was applied to the new dataset composed of the predictors selected by the GA to eliminate those redundant predictors which are not eliminated by GA. Elastic Net is a powerful tool that helps further reduce the number of predictors selected in the

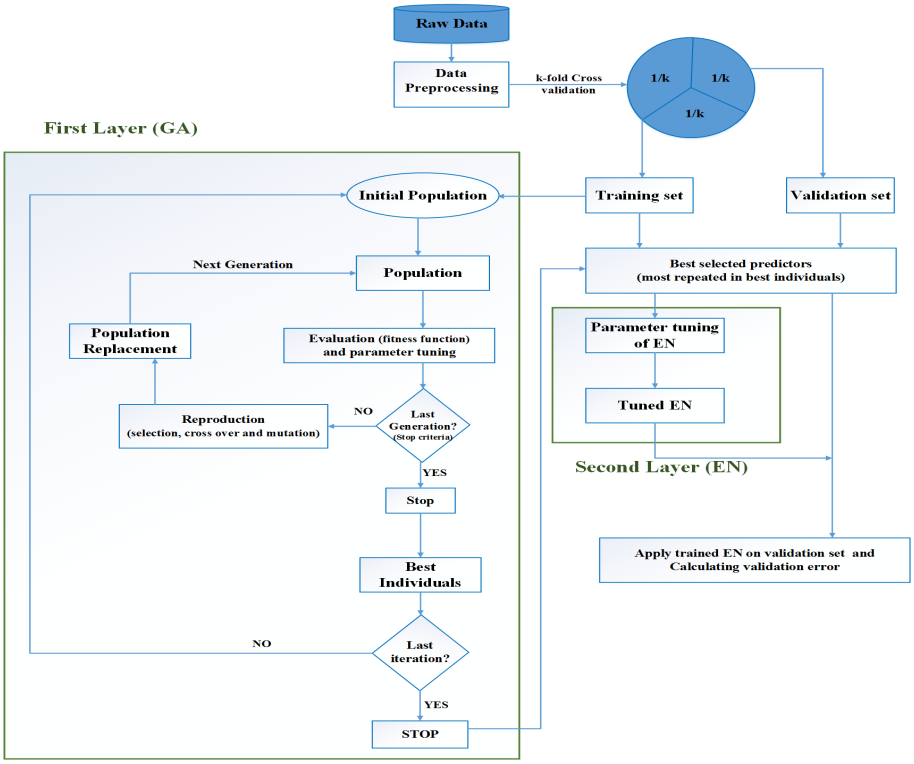


Figure 2.1: Flowchart for the proposed GA-EN approach

first layer, and, thus, improving the performance of the model. However, as its performance significantly depends on the hyper-parameters, α and ρ , they are required to be tuned through the training process. Finally, the tuned model is ready to be evaluated on the validation set. It should be noted that the hyper-parameters tuning is calculated through a k-fold cross-validation process, as well.

The applied fitness function of GA (FF_{GA}), which incorporates EN as the regression model evaluates the fitness value associated with each individual based on Eq.(8).

$$FF_{GA} = \quad w_r * r_{RMSE} + w_p * R_p \tag{2.8}$$

Individuals will be sorted based on their FF value and selected number of individuals with lowest FF value are considered as the parents for the next generation (the objective function is a minimization).

w_r and w_p in Eq.(8) are the weights of the prediction error and the number of selected predictors, respectively, which satisfy the following conditions (Eqs.(9-10)).

$$w_r + w_p = 1 \tag{2.9}$$

$$w_r, w_p \geq 0 \tag{2.10}$$

These weights are determined empirically by considering the importance between r_{RMSE} and the feature selection intensity. The optimal values of w_r and w_p aim at minimizing cross-validation error.

Eq.(11) defines r_{RMSE} based on the $RMSE_{CV}$ and the average of response variable, $\bar{y}$. This has been done to scale the prediction error for all datasets with different range of response variable. The fraction of selected predictors is defined as R_p in Eq.(12), where f_p is a binary variable that denotes if predictor p is included in a particular individual or not (Eq.(13)).

$$r_{RMSE} = \frac{RMSE_{CV}}{\overline{y}} \tag{2.11}$$

$$0 \leq R_p = \frac{\sum_{p=1}^{P} f_p}{P} \leq 1 \tag{2.12}$$

$$f_p \in \{0, 1\} \tag{2.13}$$

2.4.4 Data Description and Pre-processing

Motivated by the importance of the agricultural system in food production, particularly Maize plants in the US (Figure 2.2), a case study on Maize traits prediction has been carried out to demonstrate the outperforming of the proposed two-layer feature selection method. In this case study, the SNPs (Single Nucleotide Polymorphisms) data of Maize parents are collected to predict their expression level (RNA-seq) information. The US-NAM parents' data is used in this paper, which is publically available at NCBI SRA under SRA050451 (shoot apex), and SRA050790 (ear, tassel, shoot, and root) and at NCBI dbSNP handle PSLAB, batch number 1062224.

This dataset contains the expression level information for about 6000 genes of 27 Maize parents in addition to about 4 million SNP data associated with those parents. As a biological pre-processing step to reduce the number of SNPs, the co-Expression quantitative trait loci

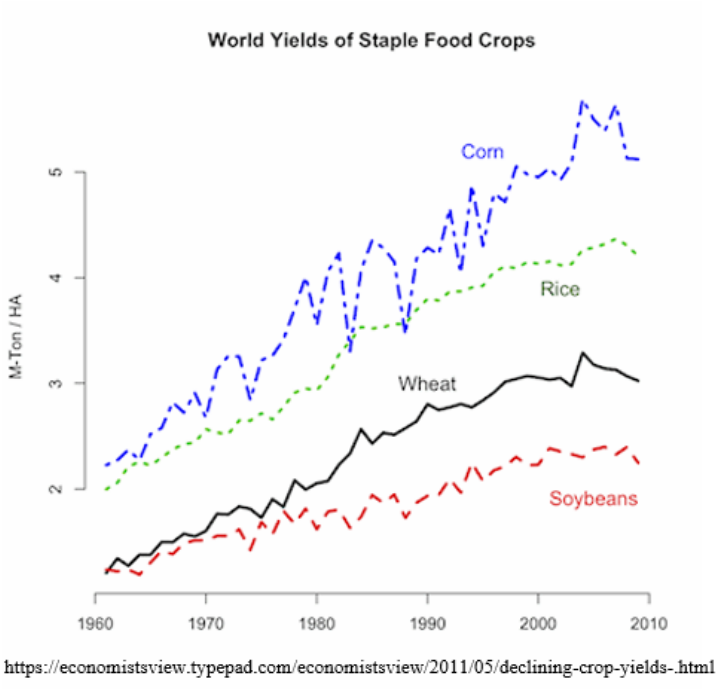


Figure 2.2: Crops production in US

(eQTL) analysis has been conducted to identify the most important SNPs related to each gene. SNPs importance level is determined by a predefined distance around each gene and those included in this distance are counted as important SNPs [Kusmec et al. (2017)]. The shorter this distance is defined, the fewer the number of SNPs would be included. The distance considered for the eQTL analysis in this study resulted in, on average, 123 SNPs for each gene. This process reduces the number of SNPs from ∼4 million to ∼728000. Moreover, SNPs data are converted to binary representation. For missing data, a linear regression-based imputation method has been implemented based on the two nearest SNPs. Thus, a prediction model is defined for each gene that aims to predict the expression level of Maize parents based on their SNPs information.

All gene datasets contain the same number of observations (27 parents) while the number of predictors (SNPs) is different. In order to validate the prediction accuracy improvement of the proposed model on datasets with different ratio of the number of predictors to the number of

observations, ten gene datasets have been selected in a way that a diverse range of this ratio has been covered.

2.5 Numerical Results and Analysis

The objective of this section is to evaluate the proposed two-layer feature selection method in terms of reduction in feature space dimension and prediction accuracy. Moreover, tuning the hyper-parameters in both GA and EN should be carried out before model evaluation since it is expected to improve the performance of the model.

2.5.1 Performance Metrics

In this paper, due to the continuity of the response variable, *relative* $RMSE_{CV}$ is considered as the performance evaluation. It is calculated through a 3-fold cross-validation process. 3-fold is chosen since 27 is dividable by 3, thus each observation will be included in just one fold at a time. *relative* $RMSE_{CV}$ is calculated by Eq.(2.14).

$$\text{relative } RMSE_{CV} = \frac{RMSE_{CV}}{\bar{y}} \tag{2.14}$$

2.5.2 Hyper-parameter Tuning

There are no universal fixed parameters for GA and as they significantly affect the GA efficiency, they need to be generally tuned to specific problems. Therefore, GA parameters should be set in such a way that the highest exploitation is achieved. To do that, GA is required to find a perfect solution in the early stages of its process. In order to increase the chance of fast improvement in the GA's response, the highest possible elitism, limited initial population size, and quite a high probability of mutation have been applied [Leardi (2000)]. Besides, some random individuals have been selected in each generation to keep the next generation diverse at the same time. Additionally, in order to follow the time constraint, the number of generations has to remain low [Welikala et al. (2015)]. Table 2.1 summarizes the tuned GA parameters applied in this study.

Table 2.1: Tuned GA Parameters

GA Parameters	**Values/Method**
Initial population size	50
#of generations	10
Population type	Bit string
#of BS	19
#of RS	1
#of offspring	5
Crossover function	single-point
Mutation rate	0.05

Moreover, the weights w_r and w_p inside the GA$'$s fitness function should be tuned for each gene dataset, separately. A grid search approach has been designed to select the best weights with the lowest prediction error. Four different values are considered for these weights in the grid search subset to cover all possible scenarios.

Table 2.2: Weights in fitness function

Scenario	1	2	3	4
w_r	0.15	0.5	0.85	1
w_p	0.85	0.5	0.15	0

Table 5.1 shows the different scenarios in which the higher the weight, the more emphasis is imposed on the minimization of the associated term. From scenario 1 to scenario 4, more emphasis has been imposed on reducing the prediction error than decreasing the number of selected predictors. The particular scenario with $w_r = 0, w_p = 1$ is not considered since the primary purpose in this study is to improve the prediction accuracy and solely focusing on minimizing the number of selected predictors would not achieve this goal. The best pair of weights with lowest *relative* $RMSE_{CV}$ is then selected for each gene dataset and further analyses are implemented with the selected weights. Figure 2.3 demonstrates the comparison of the *relative* $RMSE_{cv}$ among all different scenarios for each gene dataset.

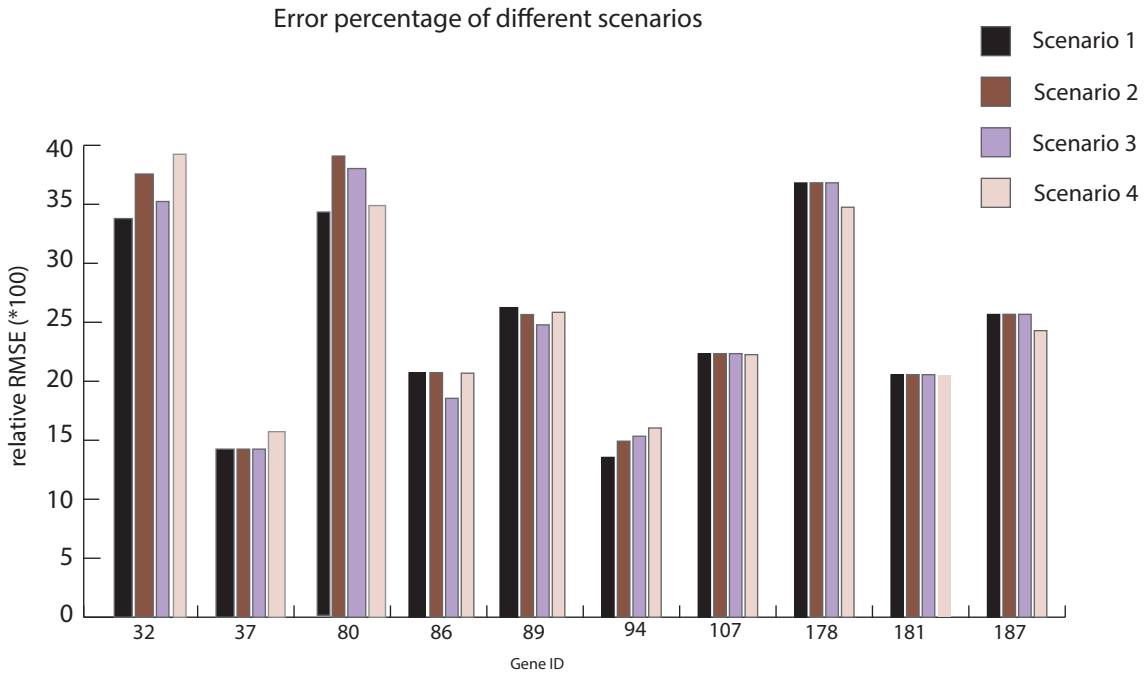


Figure 2.3: Relative RMSE for different scenarios

Following w_r and w_p, the next significant parameter to be tuned is the Fraction of Selected Predictors (FSP). It is a threshold that defines how often a particular predictor should be included in the best individuals of GA in each iteration, in order to be included in the final subset of predictors in the first layer of the proposed method. The larger this threshold is, the stricter the model is in selecting predictors. In this study, GA is repeated five times and each iteration provided us with the best individual (best subset of predictors) throughout ten generations. To tune this parameter, three values $(0.3, 0.5,$ and $0.7)$ have been considered in a similar grid search approach to select the best FSP in terms of lowest *relative* $RMSE_{CV}$ for each gene dataset. This grid search subset is designed in a way to incorporate low, medium, and high strictness of the method. *Relative* $RMSE_{CV}$ results associated with different FSP in the grid search subset for each gene dataset is illustrated in Figure 2.4. The best FSP which gives the minimum *relative* $RMSE_{CV}$ is selected for further analyses.

With FSP, w_r, and w_p fixed, the EN hyper-parameters (ρ and α) should be tuned within the second layer of the proposed two-layer feature selection method. The numerical results have been

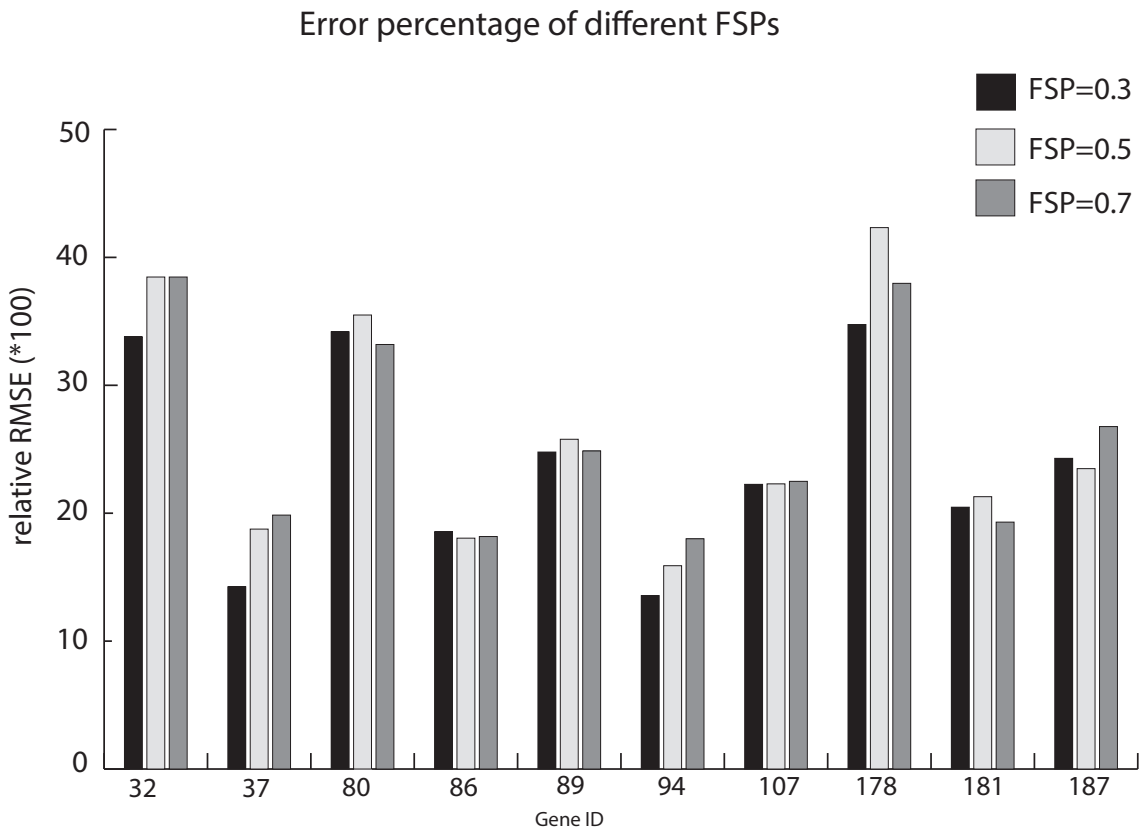


Figure 2.4: Relative RMSE for different FSPs

compiled in *Python 3*. Thus, EN selects best α from 10 non-zero values considered in *sklearn* library provided in *Python*. These values are set automatically to ensure the range of the values are from less than one to above one. For this case study, α values are considered within the range $(0.004, 50)$. Moreover, ρ would be selected from the grid search subset of $\{0, 0.1, 0.3, 0.5, 0.7, 0.9, 1\}$. These hyper-parameters are tuned via a 3-fold cross-validation process and the average of the best values regarding each k partitioning, along with the tuned hyper-parameters of the first layer are listed in Table 2.3.

Table 2.3 also includes *Gene Shape Ratio*, which defines the ratio of the number of predictors to the number of observations for each gene dataset. As most of the gene dataset contains, on average, about 123 predictors (SNPs), their shape ratio belongs to $(2, 3)$ interval. However, it can

Table 2.3: Tuned hyper-parameters of the proposed method

Gene ID	Gene Shape Ratio	α_{best}	ρ_{best}	w_r	w_p	FSP	CPU time (sec.)
32	1.33	0.017	0.36	0.15	0.85	0.3	377.7
37	9.074	0.457	0.36	0.85	0.15	0.3	610.83
80	2.89	30.05	0.36	0.15	0.85	0.7	420.3
86	3.18	0.435	0.3	0.85	0.15	0.5	450.6
89	2.44	3.38	0.23	0.85	0.15	0.3	421.65
94	7.33	0.21	0.36	0.15	0.85	0.3	651.01
107	3.66	9.74	0.36	1	0	0.3	457.05
178	2.63	3.31	0.43	1	0	0.3	432.7
181	10.62	3.08	0.1	1	0	0.7	611.1
187	2.33	0.56	0.63	1	0	0.5	348.87

be seen in Table 2.3 that datasets whose ratio is out of this range also have been considered in this paper to validate the performance of the proposed method for datasets with different shape ratio.

Also, the CPU time of running the GA-EN algorithm for each gene dataset on *Windows 10, Core(TM) i7-4770 CPU with a 3.4 GHz PowerPC processor and 16 GB RAM* is provided in Table 2.3.

2.5.3 Model Validation

The results of our numerical experiments from comparing the proposed two-layer feature selection method with the following benchmarks are included in this section.

1. EN (embedded method)

2. GA combined with linear regression (wrapper method)

Both benchmarks are considered as single-layer feature selection methods. The first one is an embedded method, while the second one (GA-Lr) is a wrapper. The reason to choose GA as the wrapper method benchmark is due to its flexibility among other metaheuristic methods. Linear regression has been adopted as the GA's learning algorithm, which does not incorporate any feature selection, thus GA-Lr will be a wrapper method. Furthermore, Elastic Net has been

selected as the embedded method benchmark, since it is the generalized form for LASSO and Ridge regression in the embedded class. With carefully selected hyper-parameters, the performance of Elastic Net method would represent the state-of-art outcome. Outperforming these benchmarks, it confirms that the superiority of the model is not only because of GA or EN separately, but it successfully demonstrates higher prediction accuracy because of the combination of GA and EN which designs the two-layer feature selection approach. The proposed model with tuned hyper-parameters has been evaluated through 3-fold cross-validation and the performance is compared in terms of *relative* $RMSE_{CV}$.

Figure 2.5 compares the *relative* $RMSE_{CV}$ of the benchmarks with the proposed method. The results confirm that combining GA with EN that has regularization characteristics inside not only outperforms the combination of the GA with non-regularized prediction method (wrapper method) but also it does achieve better performance than applying that regularized prediction method without GA assistance(embedded method) in predicting the expression level of Maize parents. The reason behind of the outperforming of GA-EN hybrid method is that not only, the most parsimonious set of predictors along with the highest level of prediction accuracy are selected in GA process in the first layer, but also the EN eliminates those insignificant and redundant predictors that still exist in the selected predictors subset in the second layer, to improve the prediction accuracy. Moreover, it can be seen in Figure 2.5, for some gene datasets such as gene 37, 89 and gene 94, the *relative* $RMSE_{CV}$ of GA-Lr method is greater than one which means that the prediction error associated with the wrapper method is greater than the average of the response variable. In these cases, the embedded method in the second layer of the proposed method would be able to ignore redundant/or irrelevant predictors to improve prediction accuracy.

Table 2.4 demonstrates the number of predictors in the original gene datasets and the number of predictors that each model selects through cross-validation. Also, the *relative* $RMSE_{CV}$ associated with each model with their selected predictors are presented in Table 2.4. The highlighted values show the minimum number of selected predictors and the minimum *relative*

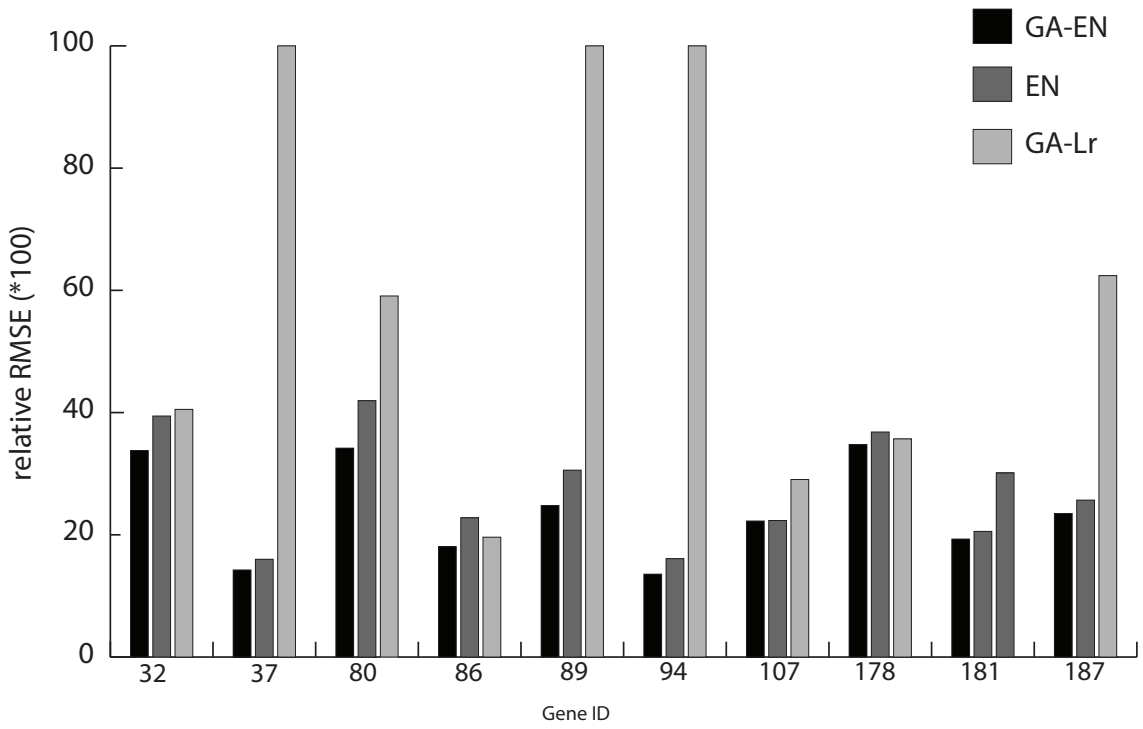


Figure 2.5: Relative RMSE of different methods

$RMSE_{CV}$ for all ten gene datasets. For most gene datasets, GA-EN method demonstrates the most reduction in number of predictors along with minimum *relative* $RMSE_{CV}$. In other words, this method not only reduces the dimension of the data, its complexity, and required storage but also, it results in smaller prediction error. However, for some datasets such as genes 32 and 86, GA-Lr selects the smallest subset, but it achieves higher prediction error.

It can importantly be said that the proposed feature selection method improves the performance of the prediction model by ignoring the irrelevant and useless predictors. An important task in such a process is to capture necessary information in selecting critical predictors; otherwise, the performance of the prediction model might be degraded as can be seen for gene 32 and 86. Although GA-EN selected a bulkier predictor subset compared to others, it provides lower prediction error for these datasets. In fact, the results presented for other methods

presented in Table 2.4 indicate that the smallest or largest predictor subset does not guarantee the best or worst prediction accuracy.

Table 2.4: Result of experiment

Gene ID	**Method**	**# of original predictors**	**# of final predictors**	**relative** $RMSE_{CV}(\%)$
32	GA-EN	36	5	33.79
	EN		25.66	39.44
	GA-Lr		2	40.52
37	GA-EN	245	73.33	14.25
	EN		119	15.99
	GA-Lr		134	> 100
80	GA-EN	78	1	34.19
	EN		40.67	41.95
	GA-Lr		24	59.1
86	GA-EN	86	6	18.05
	EN		22.67	22.8
	GA-Lr		3	19.61
89	GA-EN	66	8.33	24.78
	EN		19.33	30.57
	GA-Lr		19	> 100
94	GA-EN	198	67.33	13.56
	EN		76	16.1
	GA-Lr		95	> 100
107	GA-EN	99	6	22.25
	EN		70	22.34
	GA-Lr		74	29.05
178	GA-EN	71	28.6	34.75
	EN		43.67	36.82
	GA-Lr		63	35.7
181	GA-EN	287	36	19.3
	EN		161.67	20.56
	GA-Lr		56	30.12
187	GA-EN	63	17.3	23.48
	EN		38.67	25.67
	GA-Lr		30	62.4

The comparison of results shows the effectiveness of the two-layer wrapper-embedded method in improving the prediction accuracy for regression problems. The statistical differences of the results have been tested via a two-sample t-test. μ_{GA-EN}, μ_{EN}, and μ_{GA-Lr} stand for the average of *relative* $RMSE_{CV}$ of GA-EN, EN, and GA-Lr, respectively. The first two-sample t-test, $H_o : \mu_{GA-EN} = \mu_{EN}$ vs $H_a : \mu_{GA-EN} < \mu_{EN}$, demonstrates that in significance level of $\alpha = 0.1$, the average *relative* $RMSE_{CV}$ of GA-EN is less than the average *relative* $RMSE_{CV}$ of EN which confirms the superiority of the two-layer feature selection method (GA-EN) over the

embedded method (EN). The second two-sample t-test, $H_o : \mu_{GA-EN} = \mu_{GA-Lr}$ vs $H_a : \mu_{GA-EN} < \mu_{GA-Lr}$, demonstrates that in significance level of $\alpha = 0.05$, the average *relative* $RMSE_{CV}$ of GA-EN is less than the average *relative* $RMSE_{CV}$ of GA-Lr, i.e., the proposed GA-EN method outperforms the GA-Lr method in significance level of $\alpha = 0.05$.

Through the above study, we can conclude that the combination of EN with GA, including a modified fitness function in which the smallest subset of predictors with the lowest *relative* $RMSE$ has been found, demonstrates higher prediction accuracy in comparison with EN and GA-EN methods in predicting the expression level of Maize plants. This hypothesis has been implemented on datasets with different ratios of the number of predictors to the number of observations and the results validate the superiority of the proposed model for all datasets.

2.6 Conclusion

This paper proposed a novel two-layer feature selection approach to select the best subset of salient predictors in order to improve the prediction accuracy of regression problems. It is a two-layer method, which is a hybrid wrapper-embedded method composed of GA, as the wrapper, and EN as the embedded method. In the first layer of GA-EN method, GA searches for the smallest subset of predictors with minimum prediction error. It can reduce the computation time of finding the best subset of predictors by avoiding the exhaustive search through all possible subsets. In the second layer, adopting the best subset of predictors outputted from GA, EN has been applied to eliminate the remaining redundant and irrelevant predictors. The regularization approach within the EN removes predictors with no significant relationship with the response variable. Therefore, the main contribution of this paper lies in combining a regularized learning method with GA to achieve higher prediction accuracy dealing with regression problems.

The proposed two-layer feature selection model has been applied on a real dataset of Maize genetic data, which has multiple subsets of high-dimesional feature space datasets with different numbers of predictors. Based on the numerical results, the two-layer wrapper-embedded (GA-EN) method that consists of two layers of feature elimination process results in smaller root mean

square error for all datasets with different feature space dimension, compared to the embedded (EN) method and the wrapper (GA-Lr). The outcome of the present study revealed that combining a wrapper and an embedded feature selection method particularly, GA and EN, would reduce the dimension of feature space by more than eighty percent on average without negatively affecting accuracy.

This study is subject to few limitations which suggest future research directions. Firstly, this model selects the best w_r, w_p, and FSP from discrete subsets due to insufficient computational capacity and time limitation. It can be addressed in future research by letting the model select the best value of them from the continuous interval of $(0, 1)$, which may improve the prediction accuracy. Secondly, although GA is more effective than exhaustive search, large number of evaluations are still required, which leads to high computational cost, especially for large size problems. To address this issue in the future studies, EN can be adopted as a complementary method to seed the initial population of GA, i.e., incorporate some individuals in the initial population of GA that includes the predictors yielded by EN. Thirdly, the proposed method can be applied on datasets with different nature from what has been analyzed in this study, in terms of feature space dimension, type of the response variable, etc. These should be reserved as future research topics.

2.7 References

Aytug, H. (2015). Feature selection for support vector machines using generalized benders decomposition. *European Journal of Operational Research*, 244(1):210–218.

Brown, E. C. and Sumichrast, R. T. (2005). Evaluating performance advantages of grouping genetic algorithms. *Engineering Applications of Artificial Intelligence*, 18(1):1–12.

Cerrada, M., Zurita, G., Cabrera, D., Sánchez, R.-V., Artés, M., and Li, C. (2016). Fault diagnosis in spur gears based on genetic algorithm and random forest. *Mechanical Systems and Signal Processing*, 70:87–103.

Chen, W., Xu, C., Zou, B., Jin, H., and Xu, J. (2019). Kernelized elastic net regularization based on markov selective sampling. *Knowledge-Based Systems*, 163:57–68.

Cheng, J.-H., Sun, D.-W., and Pu, H. (2016). Combining the genetic algorithm and successive projection algorithm for the selection of feature wavelengths to evaluate exudative characteristics in frozen–thawed fish muscle. *Food chemistry*, 197:855–863.

Cilia, N. D., De Stefano, C., Fontanella, F., and di Freca, A. S. (2019). Variable-length representation for ec-based feature selection in high-dimensional data. In *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)*, pages 325–340. Springer.

Cornejo-Bueno, L., Nieto-Borge, J., García-Díaz, P., Rodríguez, G., and Salcedo-Sanz, S. (2016). Significant wave height and energy flux prediction for marine energy applications: A grouping genetic algorithm–extreme learning machine approach. *Renewable Energy*, 97:380–389.

De Stefano, C., Fontanella, F., and Marrocco, C. (2008). A ga-based feature selection algorithm for remote sensing images. In *Workshops on Applications of Evolutionary Computation*, pages 285–294. Springer.

Fukushima, A., Sugimoto, M., Hiwa, S., and Hiroyasu, T. (2019). Elastic net-based prediction of ifn-β treatment response of patients with multiple sclerosis using time series microarray gene expression profiles. *Scientific reports*, 9(1):1822.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

Hall, M. A. (1999). Feature selection for discrete and numeric class machine learning.

Hong, H., Tsangaratos, P., Ilia, I., Liu, J., Zhu, A.-X., and Xu, C. (2018). Applying genetic algorithms to set the optimal combination of forest fire related variables and model forest fire susceptibility based on data mining models. the case of dayu county, china. *Science of the total environment*, 630:1044–1056.

Hong, J.-H. and Cho, S.-B. (2006). Efficient huge-scale feature selection with speciated genetic algorithm. *Pattern Recognition Letters*, 27(2):143–150.

Hu, Z., Bao, Y., Xiong, T., and Chiong, R. (2015). Hybrid filter–wrapper feature selection for short-term load forecasting. *Engineering Applications of Artificial Intelligence*, 40:17–27.

Huang, C.-L. and Wang, C.-J. (2006). A ga-based feature selection and parameters optimizationfor support vector machines. *Expert Systems with applications*, 31(2):231–240.

Jeong, Y.-S., Shin, K. S., and Jeong, M. K. (2015). An evolutionary algorithm with the partial sequential forward floating search mutation for large-scale feature selection problems. *Journal of The Operational research society*, 66(4):529–538.

Kabir, M. M., Islam, M. M., and Murase, K. (2010). A new wrapper feature selection approach using neural network. *Neurocomputing*, 73(16-18):3273–3283.

Kusmec, A., Srinivasan, S., Nettleton, D., and Schnable, P. S. (2017). Distinct genetic architectures for phenotype means and plasticities in zea mays. *Nature plants*, 3(9):715.

Leardi, R. (2000). Application of genetic algorithm–pls for feature selection in spectral data sets. *Journal of Chemometrics*, 14(5-6):643–655.

Lin, K.-C., Huang, Y.-H., Hung, J. C., and Lin, Y.-T. (2015). Feature selection and parameter optimization of support vector machines based on modified cat swarm optimization. *International Journal of Distributed Sensor Networks*, 11(7):365869.

Liu, S., Tai, H., Ding, Q., Li, D., Xu, L., and Wei, Y. (2013). A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. *Mathematical and Computer Modelling*, 58(3-4):458–465.

Maldonado, S. and López, J. (2018). Dealing with high-dimensional class-imbalanced datasets: embedded feature selection for svm classification. *Applied Soft Computing*, 67:94–105.

Maldonado, S. and Weber, R. (2009). A wrapper method for feature selection using support vector machines. *Information Sciences*, 179(13):2208–2217.

Mitteroecker, P., Cheverud, J. M., and Pavlicev, M. (2016). Multivariate analysis of genotype–phenotype association. *Genetics*, 202(4):1345–1363.

Oztekin, A., Al-Ebbini, L., Sevkli, Z., and Delen, D. (2018). A decision analytic approach to predicting quality of life for lung transplant recipients: A hybrid genetic algorithms-based methodology. *European Journal of Operational Research*, 266(2):639–651.

Park, I. W. and Mazer, S. J. (2018). Overlooked climate parameters best predict flowering onset: Assessing phenological models using the elastic net. *Global change biology*, 24(12):5972–5984.

Peng, H., Long, F., and Ding, C. (2005). Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (8):1226–1238.

Sotomayor, G., Hampel, H., and Vázquez, R. F. (2018). Water quality assessment with emphasis in parameter optimisation using pattern recognition methods and genetic algorithm. *Water research*, 130:353–362.

Wang, J., Zareef, M., He, P., Sun, H., Chen, Q., Li, H., Ouyang, Q., Guo, Z., Zhang, Z., and Xu, D. (2019). Evaluation of matcha tea quality index using portable nir spectroscopy coupled with chemometric algorithms. *Journal of the Science of Food and Agriculture*, 99(11):5019–5027.

Wei, C., Chen, J., Song, Z., and Chen, C.-I. (2019). Adaptive virtual sensors using snper for the localized construction and elastic net regularization in nonlinear processes. *Control Engineering Practice*, 83:129–140.

Welikala, R. A., Fraz, M. M., Dehmeshki, J., Hoppe, A., Tah, V., Mann, S., Williamson, T. H., and Barman, S. A. (2015). Genetic algorithm based feature selection combined with dual classification for the automated detection of proliferative diabetic retinopathy. *Computerized Medical Imaging and Graphics*, 43:64–77.

Xue, B., Zhang, M., Browne, W. N., and Yao, X. (2015). A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Evolutionary Computation*, 20(4):606–626.

Yu, L. and Liu, H. (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 856–863.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.

2.8 Appendix: Supplementary Data

In this section, the proposed two-layer feature selection method has been applied on additional datasets to demonstrate its superiority over other existing feature selection methods.

The additional datasets are based on a regression problem in which the feature space dimension is higher than the number of observations. This contains genotype and phenotype information for 2060 specimens of the *F2* and *F3* generations of an intercross of inbred *LG/J* and *SM/J* mice that were genotyped at 384 polymorphic SNPs, at *The Jackson Laboratory*. In this study, individuals without missing data and the 353 SNPs on the 19 autosomal chromosomes have been considered [Mitteroecker et al. (2016)]. Without loss of generality, two sample of 2060 individuals, which contain 27 individuals in each samples, have been analyzed to predict a specific phenotype based on their genotype information. Dataset is available on *www.coepra.org*.

Table 2.5 summarizes the tuned GA parameters applied for these datasets. The procedure is presented in section 4.2.

Table 2.5: Tuned GA Parameters for new datasets

GA Parameters	Values/Method
Initial population size	50
#of generations	10
Population type	Bit string
#of BS	19
#of RS	1
#of offspring	5
Crossover function	single-point
Mutation rate	0.05

Optimal values of w_r and w_p inside the GA$'$s fitness function have been selected through a grid search approach which considers 4 different scenarios (Table 5.1). The optimal weights along with optimal α, ρ, and FSP have been summarized in Table 2.6.

Table 2.6: Tuned hyper-parameters of the proposed method for new datasets

Sample number	α_{best}	ρ_{best}	w_r	w_p	FSP	CPU time (sec.)
1	0.015	0.23	0.5	0.5	0.3	576.03
2	0.043	0.63	0.85	0.15	0.3	603.35

To demonstrate the superiority of the proposed two-layer model on a new dataset, two state-of-art feature selection methods that have been adopted in section 4.3 also are considered here. The performance of the proposed method in terms of *relative* $RMSE_{CV}$, is compared with Elastic Net and Genetic Algorithm that adopts Linear Regression as its learning algorithm.

Table 2.7 demonstrates the number of predictors in the original genotype-phenotype datasets and the number of predictors that each model selects through 3-fold cross-validation. Also the *relative* $RMSE_{CV}$ associated with each model with their own selected predictors are presented in Table 2.7. The highlighted values show the minimum number of selected predictors and the minimum *relative* $RMSE_{CV}$ for two sample datasets. As can be seen in Table 2.7, GA-EN method demonstrates the most reduction in number of predictors along with minimum *relative* $RMSE_{CV}$. In other words, this method not only reduces the dimension of the data but also

smaller prediction error. The GA-EN improves the prediction accuracy by more than 2% over EN and 4% over GA-Lr.

Table 2.7: Result of experiment for new datasets

Sample number	**Method**	**# of original predictors**	**# of final predictors**	**relative** $RMSE_{CV}(\%)$
1	GA-EN	353	33.33	19.10
	EN		50	21.27
	GA-Lr		205	24.32
2	GA-EN	353	17.66	15.17
	EN		23.66	17.37
	GA-Lr		199	19.4

CHAPTER 3. THE LOOK-AHEAD-TRACE-BACK OPTIMIZER FOR GENOMIC SELECTION UNDER TRANSPARENT AND OPAQUE SIMULATORS

Authors: Fatemeh Amini *, Felipe Restrepo Franco *, Guiping Hu *, and Lizhi Wang *

* Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript to be published in *Scientific Reports* journal

3.1 Abstract

Recent advances in genomic selection (GS) have demonstrated the importance of not only the accuracy of genomic prediction but also the intelligence of selection strategies. The look ahead selection algorithm, for example, has been found to significantly outperform the widely used truncation selection approach in terms of genetic gain, thanks to its strategy of selecting breeding parents that may not necessarily be elite themselves but have the best chance of producing elite progeny in the future. This paper presents the look ahead trace back algorithm as a new variant of the look ahead approach, which introduces several improvements to further accelerate genetic gain especially under imperfect genomic prediction. Perhaps an even more significant contribution of this paper is the design of opaque simulators for evaluating the performance of GS algorithms. These simulators are partially observable, explicitly capture both additive and non-additive genetic effects, and simulate uncertain recombination events more realistically. In contrast, most existing GS simulation settings are transparent, either explicitly or implicitly allowing the GS algorithm to exploit certain critical information that may not be possible in actual breeding programs. Comprehensive computational experiments were carried out using a maize data set to compare a variety of GS algorithms under four simulators with different levels of opacity. These results reveal how differently a same GS algorithm would interact with different simulators,

suggesting the need for continued research in the design of more realistic simulators. As long as GS algorithms continue to be trained *in silico* rather than *in planta*, the best way to avoid disappointing discrepancy between its simulated and actual performances may be to make the simulator as close to the complex and opaque nature as possible.

3.2 Introduction

Plant breeders have been relying primarily on phenotypic selection (PS) to select the breeding parents to maximize the genetic gain and increase grain yield [Bhat et al. (2016)]. However, multiple studies have demonstrated that the current annual global crop yield growth rates are below the 2.4% growth rate required to meet projected crop demand in 2050 [Ray et al. (2013); Iizumi et al. (2018)]. Genomic Selection (GS), pioneered by Meuwissen et al. (2001), has been widely accepted as a game changer in animal and plant breeding [Hickey et al. (2017)]. Contrary to PS, GS allows breeders to identify superior individuals in the breeding population using genotypic in addition to phenotypic data.

Rapid development of genotyping and phenotyping technologies alongside deployment of modern computational capabilities has led to increasingly comprehensive databases and intelligent algorithms, further enabling the application of GS. Next-generation sequencing has enabled fast genome-wide marker mapping at low costs, increasing the availability of high-density marker information that improves model accuracy [Crossa et al. (2014); Bhat et al. (2016)]. Furthermore, high throughput phenotyping has allowed for rapid and accurate collection of phenotypical data via non-invasive imaging [Singh et al. (2019)]. The use of these novel phenotyping and genotyping methods has increased the availability of high-quality data sets required to create accurate GS models [Meuwissen and Goddard (2010)]. Lorenzana and Bernardo (2009) demonstrated that cumulative response from three cycles of genome wide biparental GS via best linear unbiased prediction in maize would yield 1.5 times more genetic gain than that of a PS cycle. Heffner et al. (2011) found that the overall GS prediction accuracy for thirteen agronomic traits was 14% higher than that of PS.

The effectiveness of GS has been found to rely on the accuracy of genomic prediction [Heffner et al. (2010)], especially in complex traits due to the predominance of epistatic effects [Crossa et al. (2014)]. González-Camacho et al. (2018) found ridge regression and Bayesian models to perform exceptionally well when additive traits are modeled. A study conducted by Crossa et al. (2014) further corroborated these findings by comparing various linear and nonlinear models across multiple traits and environmental conditions using maize and wheat data sets. Under low-density marker conditions, Bayesian Lasso yielded the highest prediction accuracy for additive traits male and female flowering. However, when high-density markers were used, reproducing kernel Hilbert space (RKHS) slightly outperformed Bayesian Lasso. This could be attributed to RKHS's ability to better capture epistatic interactions under high-density marker conditions [Crossa et al. (2014)]. More recently, Shikha et al. (2017) conducted a similar study, where the prediction accuracy of seven different prediction models was evaluated for multiple traits in different environments. Their study found that Bayes B, a linear approach, yielded the best overall prediction accuracy, closely followed by RKHS. These results suggest that accuracy of genomic prediction models may be sensitive to trait type (additive or complex), environmental effects, and marker density.

Although genomic prediction accuracy plays an essential role in achieving genetic gain, few studies have addressed how improved designs of selection and mating strategies can provide room for higher and faster genetic gain. Prior to Goddard (2009), truncation selection was used as the default strategy for selecting breeding parents, as genetic gain was treated as a consequence of implementing genomic prediction, whose accuracy was found to be positively correlated with genetic gain [Desta and Ortiz (2014)]. Goddard (2009) used the weighted genomic estimated breeding values (WGEBV) as a variation of the conventional genomic selection (CGS) approach by Meuwissen et al. (2001), where rarer alleles were given higher weights to increase their frequency and the long term response. Daetwyler et al. (2015) suggested to use the optimal haploid value (OHV) to for selecting breeding parents, focusing on haploid selection to generate an elite fixed line. Goiffon et al. (2017) presented optimal population value (OPV), a

population-based selection strategy, where the merit of a breeding population is evaluated based on the complementarity of the group rather than the summation of individual parent's contributions. More recently, Moeinizade et al. (2019) proposed the look ahead selection (LAS) approach, which attempts to improve genetic gain by maximizing the probability of producing elite progeny by a target deadline. LAS has been shown to outperform previous selection and mating strategies due to its unique capability to anticipate, or look ahead, how decisions made in the current generation would affect the progeny in the target generation. In Section 3.3.3, we propose a new approach, the look ahead trace back (LATB) algorithm, to further improve the performance of LAS in terms of genetic gain, especially with imperfect prediction of allele effects.

Besides selection and mating strategies, another important component that has not received enough attention in the GS literature is the simulator that we use to evaluate the performance of selection algorithms. Previously, GS approaches have been tested in transparent simulation settings, in which full genotype data and additive allele effects are assumed to be known, and no dominance effect, epistases, or genotype by environmental interactions are explicitly captured. However, such transparent simulators may not realistically reflect the opaque and complex nature that we live in, where selection and mating decisions are made based on partially observable information under uncertainty. To alleviate the discrepancy between simulation and nature, we propose our design of an opaque simulator in Section 3.3.2, which simulates nature with high dimensional data of assumed ground truth with multiple sources of uncertainty that are genetically meaningful, and only a subset of which is observable by the GS algorithms. We conducted a comprehensive computational experiment in Section 3.4 to test the performances of four GS algorithms under four different variants of simulators, which produced insightful results.

3.3 Materials and Methods

Numerous decisions must be made, by either experienced breeders or automated tools, in multiple stages of a breeding program under a great deal of uncertainty. The effectiveness of these decisions has overarching and long-lasting implications to the success of the breeding program.

Historically, breeders have accumulated wisdom from decades or centuries of trial-and-error in breeding practices. In the era of data-driven molecular breeding, more emerging tools such as high performance computing resources and sophisticated algorithms are becoming accessible to help breeders accelerate genetic gains. However, the time-consuming, resource-intensive, and high-risk nature of the breeding process makes it prohibitive to design, validate, and train the algorithms directly during the actual breeding process. Therefore, an *in silico* "simulator" that mimics nature reasonably well becomes critical for training and evaluating the "optimizer" in GS research, as illustrated in Figure 3.1. The optimizer determines the crosses to make based on historical genotype and phenotype data, nature determines the next generation genotype as a result of the crosses and produces the next generation phenotype as a result of the genotype and environment interactions, and the simulator attempts to mimic how nature works. In the following subsections, we propose a new design of simulator and a new algorithm as the optimizer.

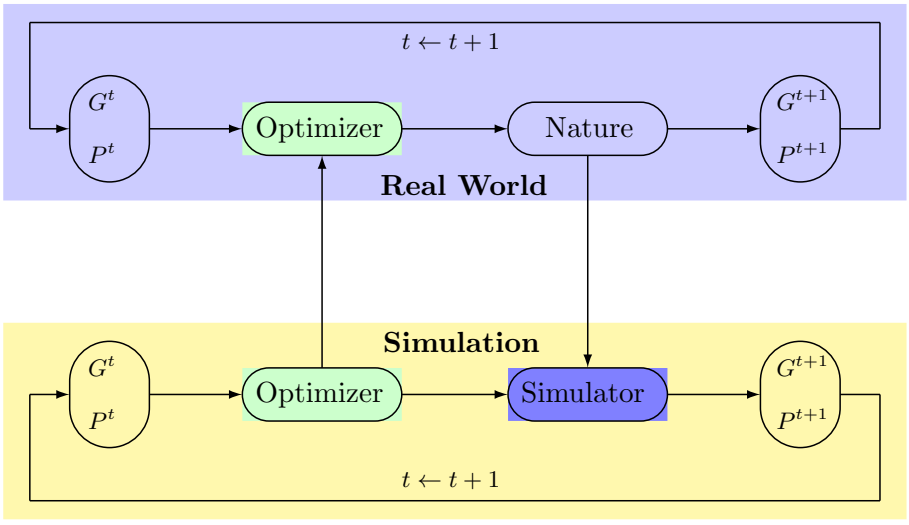


Figure 3.1: Roles of simulator and optimizer in genomic selection.

3.3.1 Transparent Simulator

In conventionally used simulators [Daetwyler et al. (2015); Goiffon et al. (2017); Moeinizade et al. (2019)], almost all information is known to the optimizer, so we refer to these as transparent simulators. To make such a simulator, historical genotype ($G^0 \in \mathbb{B}^{n^0,p,2}$) and phenotype ($P^0 \in \mathbb{Q}^{n^0}$) data are used to estimate the allele effect vector $\beta \in \mathbb{Q}^P$ for a linear model $P^0 = G^0\beta$ [González-Camacho et al. (2018); Crossa et al. (2014); Zhang et al. (2010); Karaman et al. (2016)], and the trained parameter β is then used in the simulator. Here, n^0 is the number of individuals in the initial population, p is the number of markers, and the third dimension in G^0 represents the two chromosomes in a diploid species. Oftentimes parameter β is also passed along to the optimizer as known information [Daetwyler et al. (2015); Goiffon et al. (2017); Moeinizade et al. (2019)]. Function $h(G^t|r, S)$ simulates the creation of the $(t+1)$st generation genotype from the tth generation according to the `Reproduce` function from Han et al. (2017), with $r \in [0, 0.5]^{p-1}$ being the recombination frequencies vector and S denoting the selection decision from an optimizer, which specifies the breeding parents selected from G^t.

3.3.2 Opaque Simulator

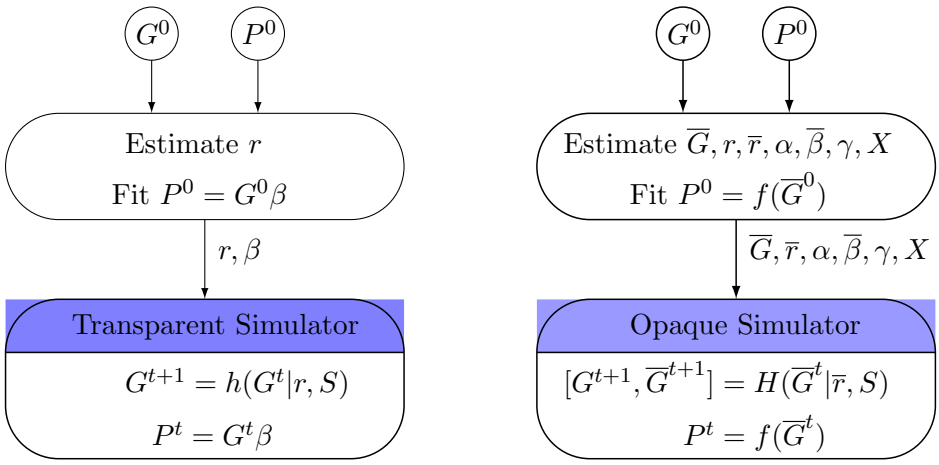


Figure 3.2: Designs of transparent (left) and opaque (right) simulators.

The proposed simulator attempts to serves as a more realistic representation of nature. We call it an opaque simulator because only partial information is observable to the optimizer. Figure 3.2 illustrates the differences between a transparent and an opaque simulator. Details of an opaque simulator are described as follows.

- The opaque simulator treats the observed genotype data as samples at a subset of marker loci, denoted as $\mathcal{P}$, of an assumed whole genome loci, denoted as $\overline{\mathcal{P}}$. The assumed whole genome is constructed by augmenting the observed genotype data of the initial generation, $G^0 \in \mathbb{B}^{n^0,p,2}$, to a much higher-dimensional space. The resulting genotype, $\overline{G}^0 \in \mathbb{B}^{n^0,\overline{p},2}$, contains G^0 at the marker loci, i.e., $\overline{G}^0_{:,\mathcal{P},:} = G^0$. The whole genome will be used throughout the breeding process inside the opaque simulator, which will evolve over time as a result of recombination, whereas only the genotype at the marker loci $\mathcal{P}$ is observable to the optimizer in each generation.

- We construct the recombination frequencies vector $\overline{r} \in [0, 0.5]^{\overline{p}-1}$ for the assumed whole genome based on the estimated recombination frequencies vector $r \in [0, 0.5]^{p-1}$ at the marker loci $\mathcal{P}$. Suppose two adjacent loci $i, i+1 \in \mathcal{P}$ correspond to two non-adjacent loci $j, j+k \in \overline{\mathcal{P}}$ separated by $k-1$ other loci in between. Given r_i, the recombination frequency between loci i and $i+1$, the recombination frequencies $\overline{r}_j, \overline{r}_{j+1}, ..., \overline{r}_{j+k-1}$ must satisfy the following equations:

$$w_{j,1} = 1 \tag{3.1}$$

$$w_{j,2} = 0 \tag{3.2}$$

$$w_{l,2} = w_{l-1,1}(1 - \overline{r}_{l-1}) + w_{l-1,2}\overline{r}_{l-1}, \forall l \in \{j+1, ..., k\} \tag{3.3}$$

$$r_i = w_{k,2}. \tag{3.4}$$

  Mathematically, these equations ensure that the probability of a recombination between two adjacent marker loci i and $i+1$ is the same as the probability of a recombination between two non-adjacent marker loci j and $j+k$. Intuitively, using the same water pipe model proposed in Han et al. (2017), the two aforementioned probabilities are analogous to the

amounts of water coming out of the left and right plumbing systems in Figure 3.3 when a unit amount of water is poured into the valve $w_{j,1}$. In Equation (3.4), $w_{j,c}$ is the amount of water that comes out of valve c of level j. The recombination frequency r_i is equal to $w_{k,2}$ because it is the amount of water that comes out from the last layer of valve at column 2 when one unit of water was poured into the first layer of valve at column 1 after k layers of redistribution (recombination).

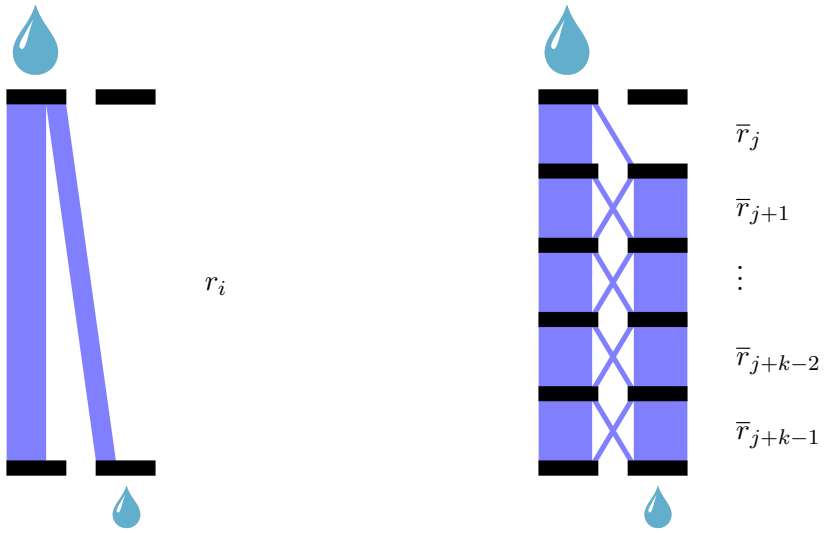


Figure 3.3: Illustration of relationship between recombination frequencies r_i and $\overline{r}_j, ..., \overline{r}_{j+k-1}$ using a water pipe model from Han et al. (2017).

- Similar with function $h(G^t|r, S)$, function $H(\overline{G}^t|\overline{r}, S)$ simulates the creation of the $(t+1)$st generation genotype $\overline{G}^{t+1}$ from the tth generation according to the `Reproduce` function from Han et al. (2017), with $\overline{r}$ being the recombination frequencies vector and S denoting the selection decision from an optimizer, which defines the breeding parents selected from G^t.

- Phenotype P^t that corresponds to genotype G^t is determined as follows:

$$
\begin{aligned}
P_i^t &= f(\overline{G}_i^t) = \sum_j \overline{\beta}_j (\overline{G}_{i,1,j}^t + \overline{G}_{i,2,j}^t) && (3.5)\\
&+ \sum_j \alpha_j I(\overline{G}_{i,1,j}^t \neq \overline{G}_{i,2,j}^t) && (3.6)\\
&+ \sum_k \gamma_k \prod_{j,m} I(\overline{G}_{i,m,j}^t + X_{m,j,k} \neq 1) && (3.7)\\
&+ \epsilon_i, \qquad \forall i. && (3.8)
\end{aligned}
$$

Here, $\overline{\beta}_j$ in equation (3.5) is the additive effect of gene j in the assumed whole genome; α_j in Equation (3.6) is the dominance effect at locus j; γ_k in Equation (3.7) is the epistatic effect of interaction k; matrix $X \in \{0, 0.5, 1\}^{\overline{p} \times 2 \times K}$ defines the membership of genes that are involved in the interactions, with $X_{m,j,k} = 1$ indicating that gene $(m, j) = 1$ is necessary to trigger interaction k, $X_{m,j,k} = 0$ indicating that gene $(m, j) = 0$ is necessary to trigger interaction k, and $X_{m,j,k} = 0.5$ indicating that gene (m, j) is not involved in the interaction k; and ϵ_i is a random noise, representing environmental effects and other effects not accounted for in the model. Equation (3.6) means that dominance effect at locus j is triggered if and only if α_j is non-zero and the two alleles are heterozygous. The indicator function $I(\overline{G}_{i,m,j}^t + X_{m,j,k} \neq 1)$ in Equation (3.7) means that epistatic effect k is triggered if and only if the genotype $G_{i,m,j} = X_{m,j,k}$ for all genes i that are involved in the effect k. Equations (3.5)-(3.8) are essentially overfitting the observed relationship between genotype G^0 and phenotype P^0 to integrate the dominance and epistatic effects. As a result, there may exist infinitely many solutions to satisfy Equations (3.5)-(3.8) with $P^0 = f(\overline{G}^0)$, and any one could be used as an opaque simulator as long as the parameters are within a reasonable range. This is because the purpose of an opaque simulator is to reveal how an optimizer might interact with an opaque nature and not to predict how nature will act.

3.3.3 The LATB Optimizer

The "optimizer" in Figure 3.1 has two main tasks: prediction and selection. Previous research effort in GS has disproportionately focused on genomic prediction with truncation selection being the default selection strategy.

In recent years, a series of algorithms have been proposed for making more strategic selection decisions. These previous algorithms such as CGS, WGEBV, OHV, OPV, and LAS were designed and tested for transparent simulators, and their performance under an opaque simulator has not been tested. A major challenges is the fact that the estimated additive allele effects may no longer be consistent with the true relationship between genotype and phenotype, which is assumed to be non-additive, unknown, partially observable, and noisy under an opaque simulator. These recent algorithms achieved improved genetic gains by strategically combining favorable alleles and removing unfavorable ones; when the accuracy of the estimated allele effects becomes questionable, so does the superiority of these algorithms.

In this section, we present the LATB algorithm as a new optimizer for GS under opaque simulators. This algorithm consists of four major steps, which are illustrated in Figure 3.4 and described as follows.

Step 1: Genomic prediction

Conventionally, genomic prediction is to estimate an allele effect vector β based on genotype data G and phenotype data P to fit a linear relationship $P = G\beta$. In the LATB algorithm, instead of estimating only one β, we use different algorithms or different hyper-parameters of a same algorithm to produce a number of allele effect vectors $\beta^s, \forall s \in \mathcal{S}$. The purpose is to reduce the chance for the selections to be biased by arbitrary choices in genomic prediction methods rather than statistically significant allele effects. We refer to these vectors as different scenarios. Intuitively, a larger number of scenarios is more likely to enclose the truth.

Step 2: Candidate crosses

A large pool of candidate crosses is created, which can be random crosses of high potential individuals (based on phenotype, genotype, or pedigree). To ensure the quality of the selection,

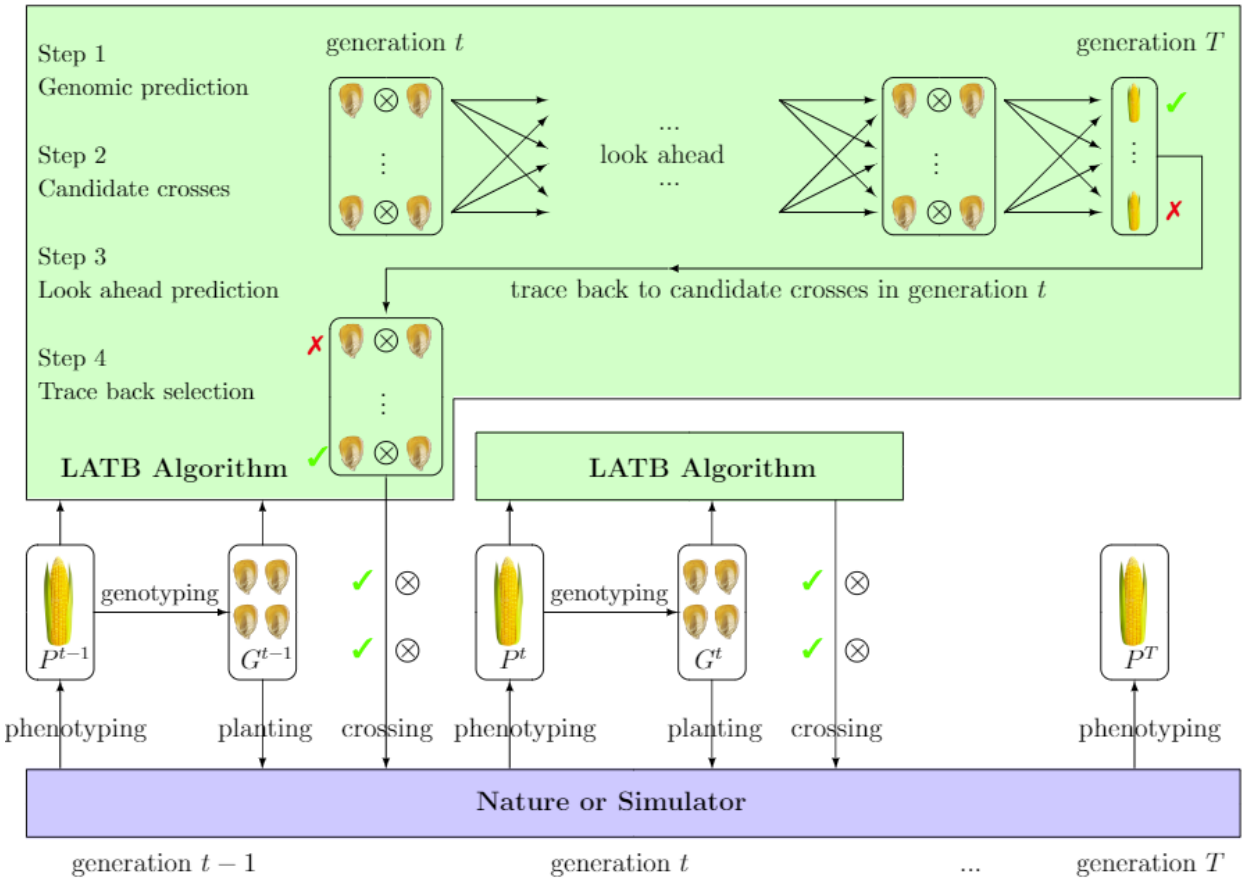


Figure 3.4: Illustration of how the LATB optimizer interacts with nature or a simulator.

the candidate pool should be large enough to include a variety of crosses subject to computational constraints on time and storage capacity.

Step 3: Look ahead prediction

Progeny from the candidate crosses are simulated and estimated using all scenarios of the allele effect vectors, and then the top performers are randomly mated with each other to produce the next generation; this process iterates until the final generation T. The purpose of this step is to look ahead the consequences of the candidate crosses, and the multiple allele effect vectors are used to provide a more robust performance assessment, i.e., an individual whose performance is

sensitive to β is less robust and may be less preferable than another that performs reasonably well under all scenarios.

In essence, this step uses observed genotype G and estimated linear function $P = G\hat{\beta}$ (as opposed to the assumed true genotype $\overline{G}$ and phenotype function $P = f(\overline{G})$, which is unbeknownst to the optimizer) to look ahead, or anticipate, the consequences of the candidate crosses in order to identify the optimal set of crosses.

Step 4: Trace back selection

We trace all individuals in the final generation of step 3 back to their ancestors in the candidate pool. Let M denote the binary relationship matrix between candidate crosses and these individuals, with $M_{c,i} = 1$ indicating that individual i is an offspring of cross c and $M_{c,i} = 0$ otherwise. Let $\hat{P}_i^T$ denote the estimated phenotype of individual i in the final generation T of step 3. Then the selection problem can be formulated as the following integer linear program.

$$\max_{x,y} \quad \sum_i \hat{P}_i^T y_i \tag{3.9}$$

$$\text{s. t.} \quad \sum_c x_c = s \tag{3.10}$$

$$y_i - x_c + M_{c,i} \leq 1 \quad \forall i, c \tag{3.11}$$

$$x_c, y_i \text{ binary} \quad \forall i, c. \tag{3.12}$$

The objective function (4.4) is to maximize the phenotypic performance of the individuals that could be produced. Decision variable $y_i = 1$ indicates that individual i can be produced (because all its ancestors in the candidate pool have been selected) and $y_i = 0$ otherwise. Constraint (4.5) means that no more than s crosses can be made. Decision variable $x_c = 1$ indicates that cross c is made and $x_c = 0$ otherwise. Since only a subset of the candidate crosses will be made, not all individuals from step 3 could be produced. Constraint (4.6) specifies the relationship among x_c, y_i, and $M_{c,i}$: individual i could not be produced unless all of its founding crosses were made.

3.4 Computational experiments

3.4.1 Simulator settings

We used a dataset that consists of phase single nucleotide polymorphisms (SNPs) and simulated phenotype data for 369 maize inbred lines of shoot apical meristem population from ISU (2020). In each simulation, 200 individuals were randomly selected from the 369 inbred lines to form an initial population. The purpose is to test the performance of GS algorithms using different initial breeding materials. The duration of the breeding process is set to be $T = 10$ generations. In each generation, updated genotype and phenotype data will be provided to the optimizer, which will then select 10 crosses from the current population. The simulator will simulate the creation of 20 progeny from each cross so that a constant population size of 200 is maintained throughout the breeding process. We conducted 500 independent simulations in order to account for the uncertainty in initial breeding materials and in the breeding process. For fair comparison, the same set of 500 random initial populations were used for all simulator-optimizer combinations in our experiments. We designed four versions of simulators to compare the performances of different optimizers. Each simulator represents a possibility of nature, with S1 being the most transparent and S4 the most opaque. In all simulators, the environmental effects are assumed to follow a normal distribution with zero mean and a standard deviation approximately 2% of the mean phenotype of the initial 369-line dataset.

- **Simulator S1: Transparent simulator with known allele effects.** The whole genome consists of 1,000 genes, all of which are assumed to have their additive effects to the phenotype, but no dominance or epistatic effects are assumed to exist. The optimizer is assumed to have perfect knowledge of the additive allele effects. This simulator represents a nature in which a sufficiently large number of genetic markers are used, little to no dominance effects or epistatic effects exist, and the accuracy of genomic prediction algorithm is perfect.

- **Simulator S2: Transparent simulator with unknown allele effects.** The whole genome consists of 1,000 genes, all of which are assumed to have their additive effects to the phenotype, but no dominance or epistatic effects are assumed to exist. These allele effects are unknown to the optimizer, so the genomic prediction algorithm is used to estimate them, whose accuracy depends on both the effectiveness of the algorithm and the magnitude of environmental effects. This simulator represents a nature in which a sufficiently large number of genetic markers are used, little to no dominance effects or epistatic effects exist, and the accuracy of genomic prediction algorithm is imperfect and sensitive to noisy environmental effects.

- **Simulator S3: Opaque simulator with additive effects.** The whole genome consists of 100,000 genes, all of which are assumed to have their additive effects to the phenotype, which are unknown to the optimizer. No dominance or epistatic effects are assumed to exist. Only 1,000 genetic markers are used to acquire the genotype information, and a genomic prediction algorithm is used to estimate the additive effects at these markers. This simulator represents a nature in which an insufficient number of genetic markers are used, little to no dominance effects or epistatic effects exist, and the accuracy of genomic prediction algorithm is sensitive to noisy environmental effects.

- **Simulator S4: Opaque simulator with additive and non-additive effects.** The whole genome consists of 100,000 genes, all of which are assumed to have their additive effects to the phenotype; moreover, heterozygosity at 20 loci will trigger dominance effects, and there are 10 epistatic effects, each involving alleles at a few loci. None of these effects are unknown to the optimizer. Only 1,000 genetic markers are used to acquire the genotype information, and a genomic prediction algorithm is used to estimate the additive effects at these markers. This simulator represents a nature in which an insufficient number of genetic markers are used, considerable dominance effects and epistatic effects exist, and the accuracy of genomic prediction algorithm is sensitive to non-additive genetic effects and noisy environmental effects.

The coefficients for additive (β or $\overline{\beta}$), dominance (α), and epistatic (γ) effects were determined to satisfy two constraints: (1) β and $\overline{\beta}$ are non-negative vectors with $\sum_{i\in\mathcal{P}}\beta_i = \sum_{i\in\overline{\mathcal{P}}}\overline{\beta}_i = 50$ and (2) the resulting phenotype values (including additive, non-additive, and environmental effects) for the initial population of 369 lines are approximately the same under all simulators. Dominance effects α and epistatic effects γ may take positive or negative values. The total additive effects in all four simulators add up to 100, which is the theoretical upper bound for simulators S1, S2, and S3; S4 may have a higher theoretical upper bound due to dominance and epistatic effects. The breakdown of phenotype of 369 lines in the initial population under four simulators are summarized in Table 3.1. Figure 3.5 shows the three β vectors in transparent simulators S1 and S2 (top), opaque simulator S3 (middle), and opaque simulator S4 (bottom). Figure 3.6 shows the recombination frequency vector r for the transparent simulators S1 and S2 and $\overline{r}$ for the opaque simulators S3 and S4, which satisfy Equations (3.1)-(3.4).

Table 3.1: Breakdown of phenotype of 369 lines in the initial population under four simulators.

	S1	S2	S3	S4
Additive	45.1 ± 4.0	45.1 ± 4.0	44.8 ± 3.4	45.5 ± 2.0
Dominance	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.2 ± 2.7
Epistatic	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	-0.8 ± 0.9
Environmental	0.0 ± 1.0	0.0 ± 1.0	0.0 ± 1.0	0.0 ± 1.0
Phenotype	45.0 ± 4.2	45.0 ± 4.1	44.9 ± 3.5	44.9 ± 3.4

3.4.2 Optimizer settings

3.4.2.1 Genomic prediction

Genomic prediction is unnecessary under simulator S1, since true allele effects are assumed to be known. Under simulators S2, S3, and S4, we use ridge regression [Hoerl and Hoerl (1962); Hoerl and Kennard (1970)] for estimating the allele effect vector β for all optimizers so that the different outcomes can be attributed to the selection algorithms rather than the accuracy of

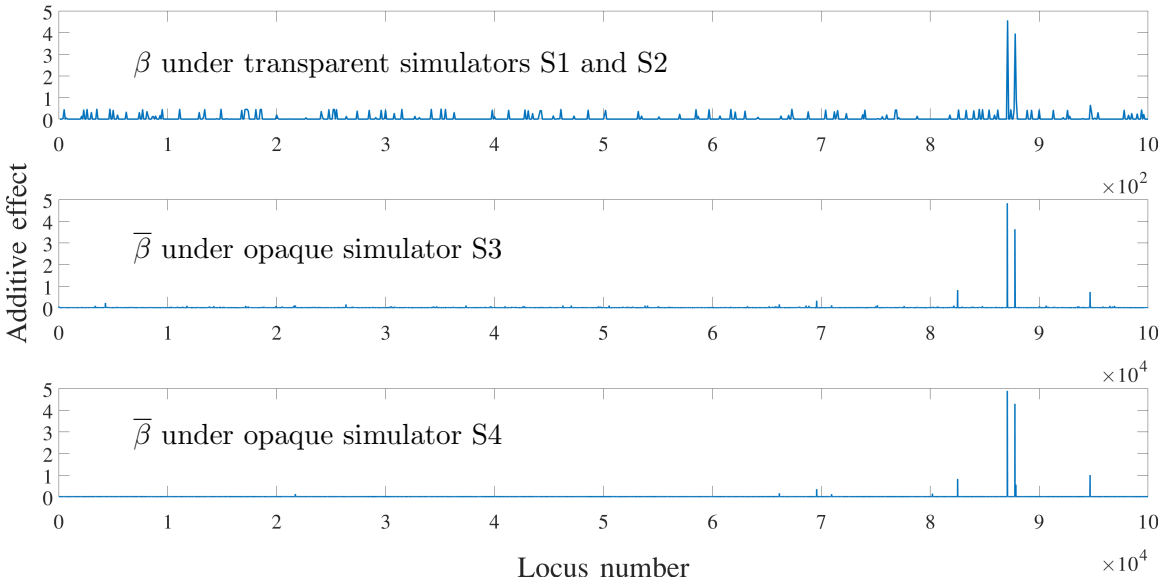


Figure 3.5: Assumed ground truth additive effects in the four simulators.

genomic prediction. Ridge regression estimates the allele effect vector as

$$\hat{\beta}_k^* = [G^\top G + kI_p]^{-1} G^\top P,$$

where I_p is the $p \times p$ identity matrix and $k \in [0, 1]$ is a parameter for balancing bias and variance. It is well known that [Marquaridt (1970)] the variance of $\hat{\beta}_k^*$ is a monotonically decreasing function of k, which becomes zero when $k = 0$ and the model reduces to the least square estimator. It has also been proven [Hoerl and Kennard (1970)] that the minimal mean square error is achieved for a positive k, which is less than that of the least square error estimator.

In our experiments, we calculated 10 scenarios of $\hat{\beta}_k^*$ with ten different k values from 0 and 1. All of these $\hat{\beta}_k^*$ vectors were provided to the LATB optimizer, whereas only the one with the minimal mean square error was used in CGS and LAS optimizers.

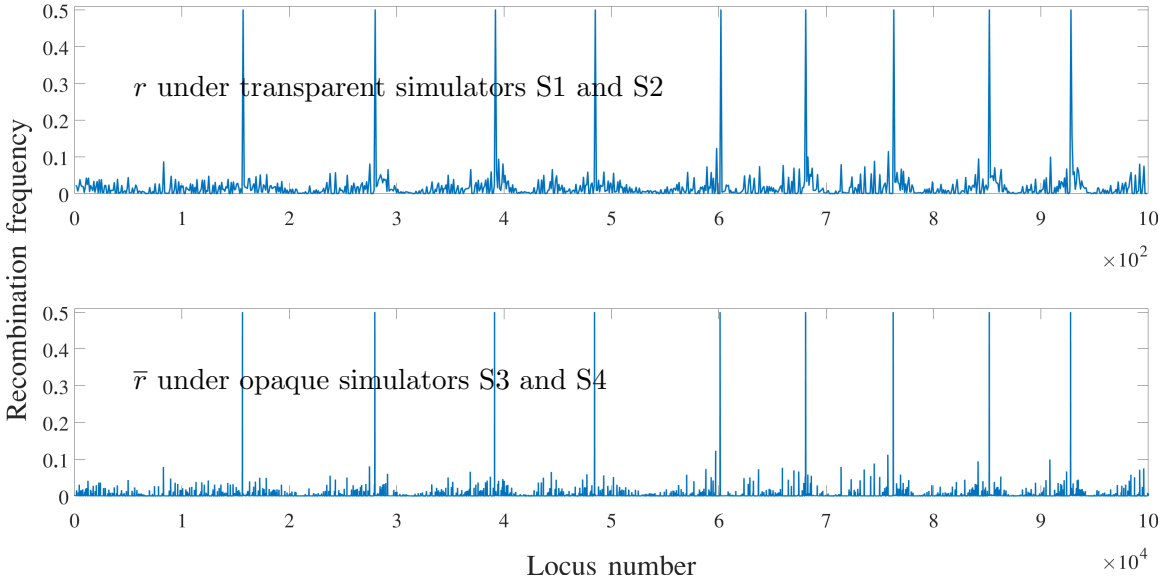


Figure 3.6: Assumed ground truth recombination frequencies in the four simulators.

3.4.2.2 PS optimizer

Selection decisions are based on the phenotypic performance. In each generation, 20 individuals with the highest phenotypes are selected and randomly mated to make 10 crosses, each producing 20 progeny.

3.4.2.3 CGS optimizer

Selection decisions are based on the genomic estimated breeding values (GEBVs), which are calculated using the β vector from ridge regression as $\text{GEBV}_i = \sum_j \beta_j (G_{i,1,j} + G_{i,2,j})$. Similar with the PS optimizer, in each generation, 20 individuals with the highest GEBVs are selected and randomly mated to make 10 crosses, each producing 20 progeny.

3.4.2.4 LAS optimizer

The same LAS algorithm from [Moeinizade et al. (2019)] was used in the experiment. In each generation, the algorithm anticipates the performance of progeny in the final generation and then searches for the best 10 crosses to make, each producing 20 progeny.

3.4.2.5 LATB optimizer

The same LATB algorithm from Section 3.3.3 was used in the experiment. In each generation, the algorithm selects the best 10 crosses, each producing 20 progeny.

3.4.3 Results

Figure 3.7 shows the genetic gain of four optimizers under four simulators, averaged over 500 independent simulations. We define genetic gain for generation t as the difference between the average phenotype of the population in generation t and that for the initial generation.

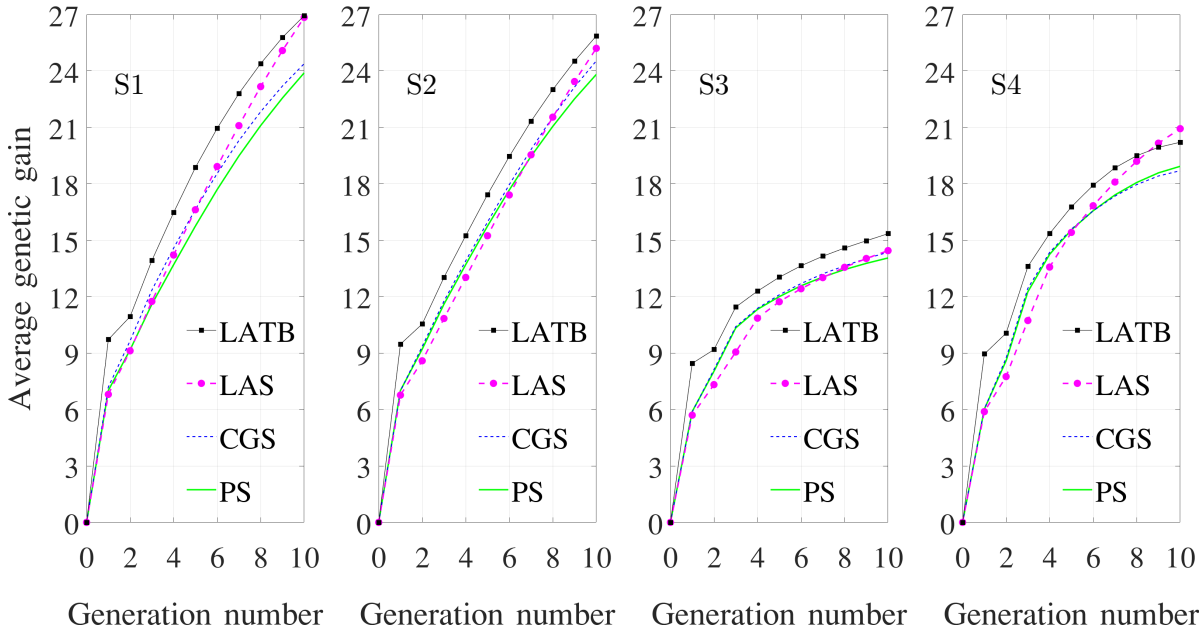


Figure 3.7: Genetic gains over ten generation, averaged over 500 independent simulation repetitions.

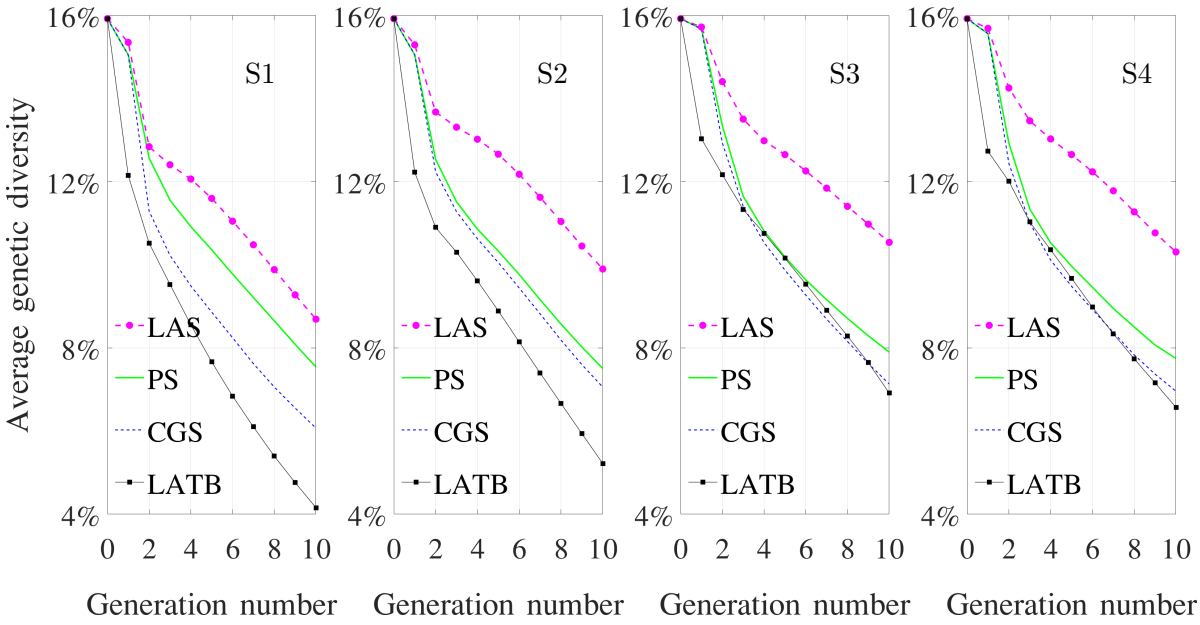


Figure 3.8: Genetic diversity over ten generation, averaged over 500 independent simulation repetitions.

Figure 3.8 shows the genetic diversity of four optimizers under four simulators, averaged over 500 independent simulations. We define genetic diversity as the average frequency of rare alleles over all genetic markers. If $G \in \mathbb{B}^{n,p,2}$ denotes the genotype of a population, then its genetic diversity is given by

$$\sum_i \frac{1}{n} \min \left\{ \sum_j \sum_c \frac{G_{i,j,c}}{2p}, 1 - \sum_j \sum_c \frac{G_{i,j,c}}{2p} \right\},$$

where $\sum_j \sum_c \frac{G_{i,j,c}}{2p}$ is the frequency of the allele coded as "1" at locus i, which may or may not be rarer than the variation coded as "0"; the min operator finds the frequency of the rare allele at locus i; the average of such values for all genetic markers gives the genetic diversity.

3.5 Discussions

3.5.1 Performance of four optimizers under four simulators

- **PS optimizer**
  - **S1 simulator:** The average genetic gain makes a big leap from the initial generation of inbred lines to F1. After that, a steady increase of genetic gains is maintained throughout subsequent generations, which are all hybrids. The genetic diversity falls gradually throughout the breeding process after bigger drops in the first couple of generations.
  - **S2 simulator:** The performances under S1 and S2 simulators are the same, since the knowledge of allele effects is not used by the PS optimizer.
  - **S3 simulator:** The increase in average genetic gain is dramatically slower than that under the S1 and S2 simulators, which is due to the more infinitesimal assumptions of the ground truth. Recombination events at the background genes partially offset and smooth out changes at the foreground markers. The loss of genetic diversity is also slower than that under the S1 and S2 simulators, but to a much less extent compared with the genetic gain, since genetic diversity is defined for the foreground markers only.
  - **S4 simulator:** The performances in both genetic gain and genetic diversity lie between S3 and S1/S2 simulators. This is intuitive because the dominance and epistatic effects would make the model less infinitesimal than S3.
- **CGS optimizer**
  - **S1 simulator:** The average genetic gain outperforms that of the PS optimizer throughout the breeding process. This is because CGS is able to use the knowledge of the true allele effects to filter out the noisy environmental effects and select the individuals with the highest genetic values. As a result of the more accurate selection,

the average genetic diversity is lost at an increasingly larger pace than the PS optimizer.

  - **S2 simulator:** CGS still outdoes PS in terms of both increasing genetic gain and losing genetic diversity over time, but to a reduced extent. This is because the optimizer no longer knows the true allele effects and has to use estimated effects to select crosses, which are inevitably less effective than those under the S1 simulator.
  - **S3 simulator:** Compared with PS, the average genetic gain follows almost the same trajectory before a slightly stronger finish in the tenth generation, but the average genetic diversity is lost noticeably faster. The estimated genetic effects of the partially observable genome was apparently not effective enough to filter out the random environmental effects, yet the side effect of selecting parents with similarly high genetic values still manages to manifest its erosion of genetic diversity over time.
  - **S4 simulator:** The assumed existence of non-additive effects makes it even harder to estimate the true genetic values of individuals. As a result, CGS leads to a slightly lower average genetic gain in the final generation than that of PS. Estimated allele effects appear to be more helpful for selecting genetically similar parents than outstanding ones, since the average genetic diversity is lost noticeably faster than that of PS.

- **LAS optimizer**
  - **S1 simulator:** LAS achieved a significantly higher genetic gain in the final generation than CGS while maintaining a significantly higher genetic diversity than PS. These observations are consistent with results in Moeinizade et al. (2019) using an S1 type of simulator. LAS was designed to maximize genetic gain at a specific deadline without performance requirements in intermediate generations; genetic diversity was maintained as a consequence of this long-term genetic gain oriented selection strategy.

- **S2 simulator:** Using imperfect estimation of allele effects, LAS barely outperforms CGS in terms of genetic gain. The reason of this disappointment is what made LAS outstanding under S1 simulator in the first place, which is its strategy to patiently accumulate favorable alleles from a diverse population of parents, some of which may be otherwise undesirable. When the allele effects turn out to be inaccurate, expected contributions of some crosses to genetic gains may fail to materialize. On the other hand, inaccurate and changing allele effect estimates lead to more diversified selections and a higher level of genetic diversity than under the S1 simulator.
- **S3 simulator:** LAS fails to outperform CGS in genetic gain, due to not only unreliable estimate of allele effects but also the effects of 99 background genes for every 1 observable genetic marker. Genetic diversity is still significantly higher than that of PS.
- **S4 simulator:** LAS shows a surprising superiority over CGS and PS, which is comparable with that under the S1 simulator in terms of genetic gain and even more so in genetic diversity. LAS benefits from the assumed existence of genetic interactions, which creates stronger signals of dominance and epistatic effects at isolated loci for the genomic prediction algorithm to pick up, enabling LAS to exhibit its strength in accumulating desirable alleles over time. In contrast, these signals may not be as helpful to CGS, because it aggregates the estimated genetic value at the individual level rather than marker level.

- **LATB optimizer**
  - **S1 simulator:** LATB outperforms LAS in terms of average genetic gain, but it also loses genetic diversity at a higher pace than all other optimizers. This is due to the selection strategy of LATB that reflects the breeding process model closely than LAS. We will discuss more differences between LAS and LATB in Section 3.5.2.

- **S2 simulator:** LATB is still accumulating genetic gain and losing genetic diversity faster than all other optimizers, but to a discounted extent due to imperfect allele effect estimates.
- **S3 simulator:** LATB widens its superiority over other optimizers percentage wise. Since LATB was designed with unreliable allele effects in mind, it makes crosses that are less sensitive to the accuracy of genomic prediction, which explains its improvement over LAS. The decline in genetic diversity over time is only slightly faster than that of CGS.
- **S4 simulator:** Similar statements can be made as under the S3 simulator. However, a noteworthy exception is that LAS outperforms LATB in the final two generations. Since LATB uses estimated allele effects more conservatively, it does not benefit as much as LAS from the amplified signals of dominance and epistatic effects.

3.5.2 Differences between LATB and LAS

First, when anticipating the consequences of crosses in the target generation, LAS assumes that all progeny will be randomly crossed with each other throughout the entire breeding process, whereas LATB explicitly expects that only the top performing progeny will be crossed with each other to produce the next generation. As a result, it is much more convenient to find the optimal set of crosses out of an enormous solution space to maximize the performance of the final generation under the LAS model, whereas the LATB model can only optimize within a relatively small subset of candidate crosses to achieve a comparable computational speed.

Second, LATB explicitly considers multiple estimates of allele effects using different prediction algorithms or parameters, which makes it less sensitive than LAS to the accuracy of the genomic prediction algorithm.

Third, the computational efficiency of LAS relies heavily on the built-in assumption of purely additive allele effects. In contrast, LATB will be compatible with more sophisticated genomic prediction algorithms that estimate both additive and non-additive allele effects, such as deep

learning models [Bellot et al. (2018)] and the recent algorithm for detecting epistatic effects [Ansarifar and Wang (2019)].

Fourth, computational experiment results suggest that LATB is more effective in increasing genetic gain, especially in early generations, whereas LAS maintains a higher level of genetic diversity.

3.5.3 Relative importance of prediction accuracy vs. selection strategy

Performances of CGS, LAS, and LATB with respect to PS are compared in Table 3.2 in terms of average genetic gain and genetic diversity in the final generation. These results suggest that selection strategy makes a greater difference than the accuracy of genomic prediction in GS.

Table 3.2: Performance comparison against PS in the final generation.

	Genetic Gain				Genetic Diversity			
	S1	S2	S3	S4	S1	S2	S3	S4
PS	0%	0%	0%	0%	0%	0%	0%	0%
CGS	2%	3%	2%	–1%	–19%	–6%	–9%	–10%
LAS	12%	6%	3%	10%	15%	31%	33%	33%
LATB	13%	9%	9%	7%	–45%	–31%	–12%	–15%

In terms of genetic gain, LAS has a much more impressive performance under S1 and S4 than under S2 and S3 simulators. In contrast, LATB is more robust: it outperforms PS by at least 7% even with imperfect genomic prediction and under the most opaque simulator. In terms of genetic diversity, LAS is the absolute winner whereas LATB makes the most compromise for genetic gain.

3.6 Conclusion

We have presented the look ahead trace back algorithm as a new selection strategy for GS and compared its performance with other state-of-the-art approaches under multiple transparent and opaque simulators. This study made three major contributions.

First, we proposed two designs of simulators for GS, transparent and opaque simulators, which represent different possibilities of nature. Most previous studies have used transparent simulators, assuming full knowledge of genotype information and purely additive allele effects. The opaque simulators were designed to capture more realistic and complex properties of nature by including partially observable genotype and non-additive genetic effects.

Second, we presented the LATB algorithm as a new optimizer for making GS decisions. This algorithm attempts to improve upon the LAS algorithm by anticipating the elimination of non-elite lines in each generations and by considering imperfect prediction of allele effects.

Third, we revealed the performances of four optimizers under four different simulators in comprehensive computational experiments. We not only demonstrated how differently an optimizer may behave under different simulators but also provided our interpretation for such behaviors. These results highlighted the importance of designing not only efficient optimizers for GS but also realistic simulators for training and evaluating the optimizers.

Our study is not without its limitations. For example, the design of the opaque simulator may not include all complex properties of nature. The environmental effects were simply assumed to follow a normal distribution and no genotype by environment interactions were explicitly incorporated. Moreover, we found it hard to determine which of the four simulators is the closest to the nature that we live in. Ultimate validation of a simulator's fidelity or an optimizer's performance requires actual experiments in nature, yet our computational results shed light on the robustness and vulnerability of different optimizers under different possibilities of nature.

Future studies should design more realistic simulators and use them to design and test more selection algorithms. Of particular interest to us is the combination of non-additive genomic prediction algorithms (such as machine learning based approaches) and the LATB algorithm.

3.7 References

Ansarifar, J. and Wang, L. (2019). New algorithms for detecting multi-effect and multi-way epistatic interactions. *Bioinformatics*, 35(24):5078–5085.

Bellot, P., de Los Campos, G., and Pérez-Enciso, M. (2018). Can deep learning improve genomic prediction of complex human traits? *Genetics*, 210(3):809–819.

Bhat, J. A., Ali, S., Salgotra, R. K., Mir, Z. A., Dutta, S., Jadon, V., Tyagi, A., Mushtaq, M., Jain, N., Singh, P. K., et al. (2016). Genomic selection in the era of next generation sequencing for complex traits in plant breeding. *Frontiers in Genetics*, 7:221.

Crossa, J., Perez, P., Hickey, J., Burgueno, J., Ornella, L., Cerón-Rojas, J., Zhang, X., Dreisigacker, S., Babu, R., Li, Y., et al. (2014). Genomic prediction in CIMMYT maize and wheat breeding programs. *Heredity*, 112(1):48–60.

Daetwyler, H. D., Hayden, M. J., Spangenberg, G. C., and Hayes, B. J. (2015). Selection on optimal haploid value increases genetic gain and preserves more genetic diversity relative to genomic selection. *Genetics*, 200(4):1341–1348.

Desta, Z. A. and Ortiz, R. (2014). Genomic selection: genome-wide prediction in plant improvement. *Trends in Plant Science*, 19(9):592–601.

Goddard, M. (2009). Genomic selection: prediction of accuracy and maximisation of long term response. *Genetica*, 136(2):245–257.

Goiffon, M., Kusmec, A., Wang, L., Hu, G., and Schnable, P. S. (2017). Improving response in genomic selection with a population-based selection strategy: optimal population value selection. *Genetics*, 206(3):1675–1682.

González-Camacho, J. M., Ornella, L., Pérez-Rodríguez, P., Gianola, D., Dreisigacker, S., and Crossa, J. (2018). Applications of machine learning methods to genomic selection in breeding wheat for rust resistance. *The Plant Genome*, 11(2).

Han, Y., Cameron, J. N., Wang, L., and Beavis, W. D. (2017). The predicted cross value for genetic introgression of multiple alleles. *Genetics*, 205(4):1409–1423.

Heffner, E. L., Jannink, J.-L., and Sorrells, M. E. (2011). Genomic selection accuracy using multifamily prediction models in a wheat breeding program. *The Plant Genome*, 4(1):65–75.

Heffner, E. L., Lorenz, A. J., Jannink, J.-L., and Sorrells, M. E. (2010). Plant breeding with genomic selection: gain per unit time and cost. *Crop Science*, 50(5):1681–1690.

Hickey, J. M., Chiurugwi, T., Mackay, I., Powell, W., Eggen, A., Kilian, A., Jones, C., Canales, C., Grattapaglia, D., Bassi, F., et al. (2017). Genomic prediction unifies animal and plant breeding programs to form platforms for biological discovery. *Nature genetics*, 49(9):1297.

Hoerl, A. E. and Hoerl, C. (1962). Application of ridge analysis to regression problems. *Chemical Engineering Progress*, 58(3):54–59.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Iizumi, T., Kotoku, M., Kim, W., West, P. C., Gerber, J. S., and Brown, M. E. (2018). Uncertainties of potentials and recent changes in global yields of major crops resulting from census-and satellite-based yield datasets at multiple resolutions. *PLOS One*, 13(9).

ISU (2020). Shoot apical meristem (sam) diversity panel genetic markers and map. https://iastate.figshare.com/s/374176500b04fd6f3729. [Online; accessed July-23-2020].

Karaman, E., Cheng, H., Firat, M. Z., Garrick, D. J., and Fernando, R. L. (2016). An upper bound for accuracy of prediction using gblup. *PloS One*, 11(8):e0161054.

Lorenzana, R. E. and Bernardo, R. (2009). Accuracy of genotypic value predictions for marker-based selection in biparental plant populations. *Theoretical and Applied Genetics*, 120(1):151–161.

Marquaridt, D. W. (1970). Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation. *Technometrics*, 12(3):591–612.

Meuwissen, T. and Goddard, M. (2010). Accurate prediction of genetic values for complex traits by whole-genome resequencing. *Genetics*, 185(2):623–631.

Meuwissen, T., Goddard, M., Hayes, et al. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, 9(7):2123–2133.

Ray, D. K., Mueller, N. D., West, P. C., and Foley, J. A. (2013). Yield trends are insufficient to double global crop production by 2050. *PloS one*, 8(6):e66428.

Shikha, M., Kanika, A., Rao, A. R., Mallikarjuna, M. G., Gupta, H. S., and Nepolean, T. (2017). Genomic selection for drought tolerance using genome-wide snps in maize. *Frontiers in Plant Science*, 8:550.

Singh, D., Wang, X., Kumar, U., Gao, L., Noor, M., Imtiaz, M., Singh, R. P., and Poland, J. (2019). High-throughput phenotyping enabled genetic dissection of crop lodging in wheat. *Frontiers in Plant Science*, 10:394.

Zhang, Z., Ersoz, E., Lai, C.-Q., Todhunter, R. J., Tiwari, H. K., Gore, M. A., Bradbury, P. J., Yu, J., Arnett, D. K., Ordovas, J. M., et al. (2010). Mixed linear model approach adapted for genome-wide association studies. *Nature Genetics*, 42(4):355–360.

CHAPTER 4. THE L-SHAPED SELECTION ALGORITHM FOR MULTI-TRAIT GENOMIC SELECTION

Authors: Fatemeh Amini [1], Guiping Hu [1], Ruoyu Wu [2], and Lizhi Wang [1]

[1] Department of Industrial and Manufacturing Systems Engineering, Iowa State University

[2] Department of Mathematics, Iowa State University

Modified from a manuscript published in *Genetics* journal

4.1 Abstract

Selecting for multiple traits as opposed to a single trait has become increasingly important in genomic selection. As one of the most popular approaches to multi-trait genomic selection (MTGS), index selection uses a weighted average of all traits as a single breeding objective. Although intuitive and effective, index selection is not only numerically sensitive but also structurally incapable of finding certain optimal breeding parents. This paper proposes a new selection method for MTGS, the L-shaped selection, which addresses the limitations of index selection by normalizing the trait values and using an L-shaped objective function to find optimal breeding parents. This algorithm has been proven to be able to find any Pareto optimal solution with appropriate weights. Two performance metrics have also been defined to quantify MTGS algorithms with respect to their ability to accelerate genetic gain and preserve genetic diversity. Computational experiments were conducted to demonstrate the improved performance of L-shaped selection over index selection.

4.2 Introduction

The effectiveness of genomic selection (GS) in accelerating genetic gain in plant and animal breeding programs [Meuwissen and Goddard (2010); Jannink et al. (2010); Rutkoski et al. (2016);

Desta and Ortiz (2014)] has motivated breeders to apply the technique for multiple traits, including yield, quality, and tolerance to biotic and abiotic stresses [Jia and Jannink (2012)]. Breeders are more likely to invest in genomic selection programs that are capable of improving multiple traits of a crop rather than a single trait throughout the generations [Bernardo (2002); Lynch et al. (1998)].

Existing approaches for multi-trait genomic selection (MTGS) include tandem selection, independent culling selection, and index selection. Tandem selection treats MTGS as the aggregation of multiple single-trait GS programs and selects for the traits sequentially [Burgess and West (1993)]. Independent culling selection sets a minimum threshold (i.e., culling levels) for each trait and only selects individuals that exceed the culling levels for all traits [Lorenzana and Bernardo (2009)]. Index selection converts MTGS to a single trait GS by using a linear combination of individual traits weighted by their importance as the breeding objective [Hazel and Lush (1942); Hazel (1943); Williams (1962)]. Recently, (Moeinizade et al., 2020) proposed a new algorithm for MTGS that maximizes one trait subject to the constraints that another trait falls within a desirable range.

Index selection has heretofore been a commonly used approach to MTGS due to its capability to select for multiple traits simultaneously and its flexibility to assign different weights according to the relative importance of the traits. In contrast, tandem selection can only select for one trait at a time, and independent culling selection may eliminate an otherwise high-performing individual due to its minor shortcoming in one trait [Lorenzana and Bernardo (2009)]. Index selection overcomes such limitations by taking an importance-weighted linear combination of all traits, giving breeders a wide range of trade-off options among the traits to choose from.

However, index selection also suffers from its own limitations, i.e., its numerical sensitivity and inability to find certain optimal selections. In this study, we propose a new approach to MTGS, which uses an L-shaped objective function (as opposed to the linear objective function used in index selection) to select optimal breeding parents that strike a balance among multiple traits with respect to their relative importance. This algorithm not only overcomes the two

limitations of index selection but also demonstrates superior performance with respect to both accelerating genetic gain and preserving genetic diversity.

The rest of the paper is organized as follows. In section 4.3, we formally formulate MTGS as a multi-objective optimization problem, introduce the L-shaped selection algorithm, and present the mathematical properties that allow it to overcome the limitations of index selection. Moreover, we define two metrics for assessing the performances of MTGS algorithms in terms of accelerating genetic gain and preserving genetic diversity. In section 4.4, we describe the computational experiments that we conducted to compare index selection and L-shaped selection methods. Finally, concluding remarks are made and future research directions are discussed in section 5.6.

4.3 Methods and Materials

Consider a breeding project that starts with an initial population of plant or animal individuals. A number of crosses are made in each generation to produce a new population of progeny in the next generation until a pre-defined deadline for the project. Suppose there are multiple traits that the breeders aim to improve through the breeding process. Under the following three simplifying assumptions, the focus of our study is to select the right individuals to make the right crosses in order to optimize all traits at the end of the breeding project.

Assumption 1 There is adequate and reliable genetic data, including genotype and recombination frequencies.

Assumption 2 All traits are largely determined by additive effects with negligible dominance or epistatic effects.

Assumption 3 Allele effects for all traits have been estimated sufficiently accurately and constant over the breeding process.

4.3.1 Problem definition

The objective of MTGS is to select a subset of breeding parents from a group of candidate individuals based on their genotype and estimated allele effects in order to maximize genetic gains with respect to multiple traits over a number of breeding generations. The following nomenclature will be used in this paper.

$\mathcal{I}$	set of candidate individuals of plants or animals for selection
$\mathcal{J}$	set of loci
$\mathcal{K}$	set of traits
$G_{i,j}$	genotype of individual $i \in \mathcal{I}$ at locus $j \in \mathcal{J}$
$\beta_{j,k}$	effect of allele $j \in \mathcal{J}$ on trait $k \in \mathcal{K}$
$v_{i,k}$	genetic value of individual $i \in \mathcal{I}$ on trait $k \in \mathcal{K}$: $v_{i,k} = \sum_j G_{i,j}\beta_{j,k}$
w_k	weight parameter that indicates the relative importance of trait $k \in \mathcal{K}$
x_i	binary variable indicating whether individual $i \in \mathcal{I}$ is selected ($x_i = 1$) or not ($x_i = 0$)
S	number of breeding parents to be selected

Without loss of generality, we assume that maximization (rather than minimization) is the direction of improvements for all traits. For a trait k that needs to be minimized, we can replace $v_{i,k}$ with $-v_{i,k}$ for all $i \in \mathcal{I}$ in a maximization model, which is equivalent to minimizing trait k. For traits whose values need to be contained within a desirable range, we can maximize the percentage of individuals in a population whose trait values fall within such a range.

With the above definitions and assumptions, the MTGS can be formulated as the following multi-objective optimization model:

$$\max_{x} \quad \sum_i x_i v_{i,k} \quad \forall k \in \mathcal{K} \tag{4.1}$$

$$\text{s. t.} \quad \sum_i x_i = S \tag{4.2}$$

$$x_i \in \{0,1\} \quad \forall i. \tag{4.3}$$

Here, the objective function (4.1) is the maximization of all traits of selected individuals. Constraint (4.2) requires that exactly S individuals be selected. Constraint (4.3) defines x_i as a binary variable for all $i \in \mathcal{I}$.

In multi-objective optimization, a feasible solution is called Pareto optimal if it is not dominated by any other feasible solution. Solution $\hat{x}$ dominates $\tilde{x}$ if $\hat{x}$ is no worse than $\tilde{x}$ in any trait and better in at least one:

$$\sum_i \hat{x}_i v_{i,k} \geq \sum_i \tilde{x}_i v_{i,k}, \forall k \in \mathcal{K} \text{ and } \sum_i \hat{x}_i v_{i,k} > \sum_i \tilde{x}_i v_{i,k}, \exists k \in \mathcal{K}.$$

Moreover, the ultimate goal of solving a multi-objective optimization problem is to find not a single Pareto optimal solution but all Pareto optimal solutions that represent the range of possible trade-offs among different traits, referred to as the Pareto frontier.

4.3.2 Index selection

As a widely used selection approach for MTGS, index selection solves a single objective optimization model that maximizes the weighted average of all traits:

$$\max_x \quad \sum_k w_k \sum_i x_i v_{i,k} \tag{4.4}$$

$$\text{s. t.} \quad \sum_i x_i = S \tag{4.5}$$

$$x_i \in \{0,1\} \qquad \forall i. \tag{4.6}$$

Here, the objective function (4.4) is the weighted average genetic value of all traits. When different weight parameters are used, index selection essentially searches for the convex efficient frontier, which is the subset of Pareto optimal solutions that are on the convex hull of the feasible region.

The following proposition shows that, when strictly positive weights are used for all traits, the optimal solution to index selection must be Pareto optimal to MTGS.

Proposition 1. *If solution $\hat{x}$ is optimal to (4.4)-(4.6) for some $w_k > 0, \forall k \in \mathcal{K}$, then $\hat{x}$ is Pareto optimal to (4.1)-(4.3).*

Proof. Suppose $\hat{x}$ is not Pareto optimal to (4.1)-(4.3). Then there exists a solution $\tilde{x}$ that dominates $\hat{x}$:

$$\sum_i \tilde{x}_i v_{i,k} \geq \sum_i \hat{x}_i v_{i,k}, \forall k \in \mathcal{K} \text{ and } \sum_i \tilde{x}_i v_{i,k} > \sum_i \hat{x}_i v_{i,k}, \exists k \in \mathcal{K}.$$

Since $w_k > 0, \forall k \in \mathcal{K}$, we have

$$\sum_k w_k \sum_i \tilde{x}_i v_{i,k} > \sum_k w_k \sum_i \hat{x}_i v_{i,k}.$$

This contradicts the assumption that $\hat{x}$ is optimal to (4.4)-(4.6). Therefore $\hat{x}$ must be Pareto optimal to (4.1)-(4.3). □

4.3.3 Two limitations of index selection

Despite the effectiveness of index selection in finding Pareto optimal solutions to MTGS, it suffers from two major limitations. First, numerical solutions to (4.4)-(4.6) may be sensitive to the units being used for the genetic effects of different traits. For example, when we try to maximize both plant height and grain yield with their respective weights, different solutions may result from simply changing the units used to measure the two traits from inches and bushels per acre to meters and tonnes per hectare.

The second limitation is the inability to discover all Pareto optimal solutions by using different values of weight parameters w_k due to the convexity of the objective function (4.4) and the potential non-convexity of the set of Pareto optimal solutions. To illustrate this point, consider the following example.

Example 1. *Suppose two individuals are to be selected from four to make a cross, then there are six candidate crosses. The genetic values of the four individuals and six crosses for two important traits are summarized in Tables 4.1 and 4.2. All six crosses are Pareto optimal since none of them is dominated by another. However, as shown in Figure 4.1, the index selection model* (4.4)-(4.6) *can only find three crosses (c_1, c_5, and c_6) that are on the efficient frontier of the six crosses in the v_1-v_2 space. The other three will never be selected no matter what non-negative weights w_1 and w_2 are used because they are dominated by the line segment between c_1 and c_5.*

Table 4.1: Genetic values of two traits for four individuals.

individual	$v_{i,k=1}$	$v_{i,k=2}$
i_1	0.05	0.33
i_2	0.22	0.22
i_3	0.30	0.15
i_4	0.44	0.06

Table 4.2: Genetic values of two traits for six crosses.

cross	individuals	$\sum_{i\in c} v_{i,k=1}$	$\sum_{i\in c} v_{i,k=2}$
c_1	(i_1, i_2)	0.27	0.55
c_2	(i_1, i_3)	0.35	0.48
c_3	(i_1, i_4)	0.49	0.39
c_4	(i_2, i_3)	0.52	0.37
c_5	(i_2, i_4)	0.66	0.28
c_6	(i_3, i_4)	0.74	0.21

4.3.4 L-shaped selection

The proposed L-shaped selection for MTGS can be formulated as the following optimization model:

$$\max_{x} \quad \min_{k} \frac{\sum_i x_i \tilde{v}_{i,k}}{w_k} \tag{4.7}$$

$$\text{s. t.} \quad \sum_i x_i = S \tag{4.8}$$

$$x_i \in \{0,1\} \quad \forall i. \tag{4.9}$$

Here, $\tilde{v}_{i,k} = \dfrac{v_{i,k} - \underline{v}_k}{\overline{v}_k - \underline{v}_k}$ is the normalized genetic value that falls within the range of $(0,1)$, where $\underline{v}_k$ and $\overline{v}_k$ are the lower and upper bounds of trait k, which can be obtained, respectively, as $\underline{v}_k = \min_i(v_{i,k} - \epsilon), \forall i \in \tilde{\mathcal{I}}$ and $\overline{v}_k = \max_i(v_{i,k} + \epsilon), \forall i \in \tilde{\mathcal{I}}$ for a large set of individuals $\tilde{\mathcal{I}}$; here ϵ is a small positive value to ensure that the normalized genetic value falls within $(0,1)$ and not on the boundaries.

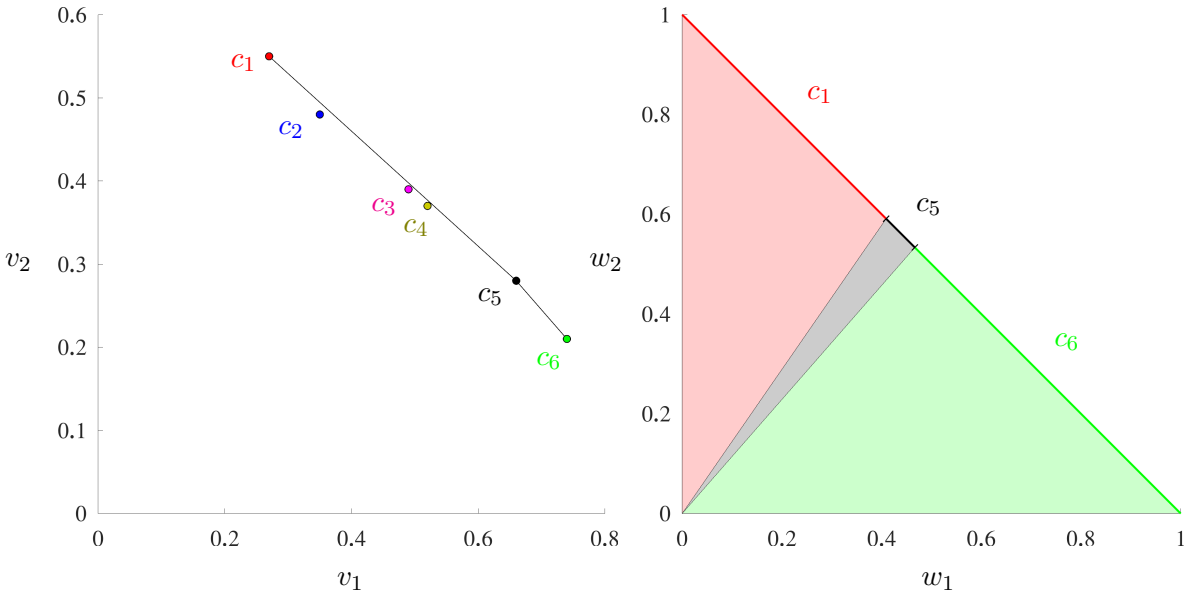


Figure 4.1: The left subfigure shows the six possible crosses in the v_1-v_2 space and the efficient frontier that index selection is able to find. The right subfigure shows that only c_1, c_5, and c_6 could be found to be Pareto optimal using index selection with different weights w_1 and w_2.

This new formulation was designed to address the two limitations discussed in section 4.3.3. The normalized genetic effect $\tilde{v}_{i,k}$ is independent of the measurement units being used, thus allowing the model to strike a balance among multiple traits in more meaningful terms. The following proposition demonstrates how model (4.7)-(4.9) addresses the second limitation.

Proposition 2. *If solution $\hat{x}$ is Pareto optimal to (4.1)-(4.3), then there exist $w_k > 0, \forall k \in \mathcal{K}$ such that $\hat{x}$ is optimal to (4.7)-(4.9).*

Proof. We claim that $w_k := \sum_i \hat{x}_i \tilde{v}_{i,k} > 0, \forall k \in \mathcal{K}$ are such that $\hat{x}$ is optimal to (4.7)-(4.9). Suppose not, then there exists a solution $\tilde{x}$ such that

$$\min_k \frac{\sum_i \tilde{x}_i \tilde{v}_{i,k}}{w_k} > \min_k \frac{\sum_i \hat{x}_i \tilde{v}_{i,k}}{w_k} = 1.$$

This means

$$\sum_i \tilde{x}_i \tilde{v}_{i,k} \geq w_k = \sum_i \hat{x}_i \tilde{v}_{i,k}, \forall k \in \mathcal{K} \text{ and } \sum_i \tilde{x}_i \tilde{v}_{i,k} > w_k = \sum_i \hat{x}_i \tilde{v}_{i,k}, \exists k \in \mathcal{K}.$$

Since $\sum_i \tilde{x}_i = S = \sum_i \hat{x}_i$, we have

$$\sum_i \tilde{x}_i v_{i,k} \geq \sum_i \hat{x}_i v_{i,k}, \forall k \in \mathcal{K} \text{ and } \sum_i \tilde{x}_i v_{i,k} > \sum_i \hat{x}_i v_{i,k}, \exists k \in \mathcal{K}.$$

This contradicts the assumption that $\hat{x}$ is Pareto optimal to (4.1)-(4.3). Therefore $\hat{x}$ must be optimal to (4.7)-(4.9) with $w_k := \sum_i \hat{x}_i \tilde{v}_{i,k} \geq 0, \forall k \in \mathcal{K}$. □

Proposition 2 is a guarantee that L-shaped selection is able to find any Pareto optimal solution to MTGS with appropriate weight parameters. To illustrate this desirable property, which index selection does not have, we solved Example 1 using L-shaped selection, and results are shown in Figure 4.2. The left subfigure illustrates that, in the case of $k = 2$, model (4.7)-(4.9) is trying to slide an L-shaped objective function (hence the name) along the direction from the origin towards (w_1, w_2) to the maximal extent while touching at least one solution with the L-shaped curve. The right subfigure shows the different Pareto optimal solutions that can be found using different combinations of weight parameters w_1 and w_2.

4.3.5 Performance measures of MTGS algorithms

In this section, we define two measures, namely Pareto optimality gap and diversity, to evaluate the performance of MTGS algorithms. The motivation is to assess the capability of an algorithm to produce progeny through the breeding process that are not only Pareto optimal but also representative of diverse trade-offs among different traits.

Suppose an MTGS algorithm was used in a number of breeding projects, each with a different sets of weight parameters. Let $\mathcal{I}^0$ denote the set of individuals produced in the final generation of all breeding projects combined, and let $\mathcal{I}^1$ denote a superset of $\mathcal{I}^0$, possibly also including individuals produced from all other competing algorithms. For any set of individuals $\hat{\mathcal{I}}$, we define $\mathcal{P}(\hat{\mathcal{I}})$ as the Pareto optimal subset of $\hat{\mathcal{I}}$:

$$\mathcal{P}(\hat{\mathcal{I}}) = \left\{ i : i \in \text{argmax}_{i \in \hat{\mathcal{I}}} \sum_{k \in \mathcal{K}} w_k v_{i,k}, \exists w_k > 0, \forall k \in \mathcal{K} \right\}.$$

The two performance measures are defined as follows:

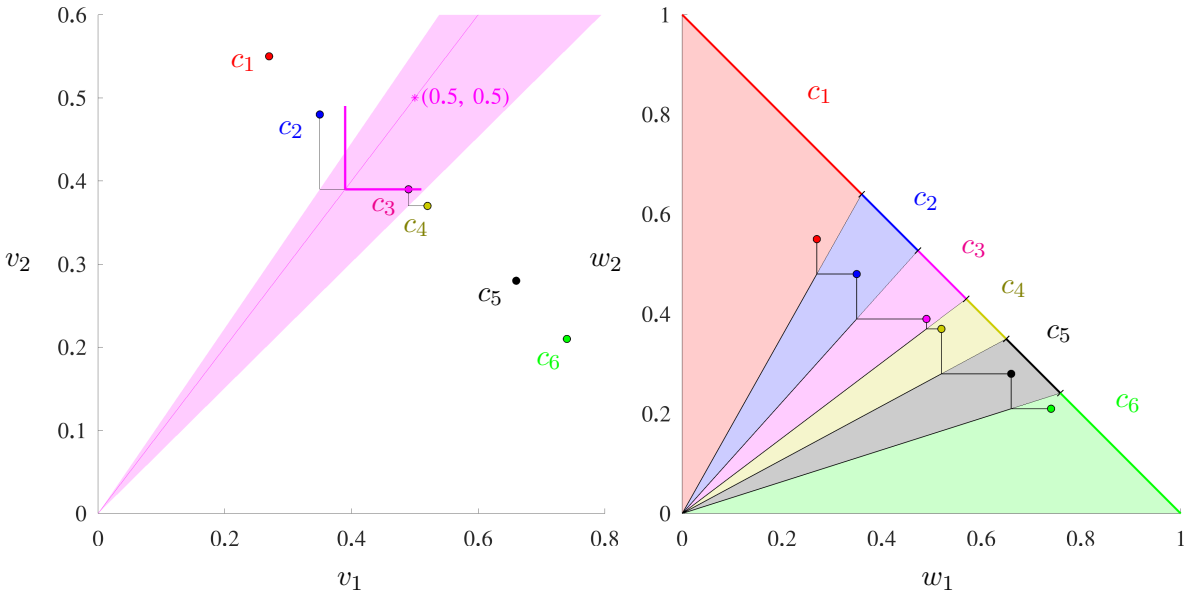


Figure 4.2: The left subfigure shows the six candidate solutions in the v_1-v_2 space and how L-shaped selection uses an L-shaped objective function to search for Pareto optimal solutions. As an example, c_3 is found to be optimal with equal weights on the two traits, since it allows the magenta-colored and L-shaped objective function to slide the furthest away from the origin towards the direction $(v_1 = w_1 = 0.5, v_2 = w_2 = 0.5)$. Moreover, c_3 is optimal for all weights inside the shaded (unbounded) triangle, the two edges of which cross the vertices of two L-shaped objective functions with one crossing c_2 and c_3 and the other crossing c_3 and c_4. The right subfigure shows that all six candidate solutions can be found to be Pareto optimal using L-shaped selection with different weights w_1 and w_2. The six candidate solutions are also plotted in the right subfigure in the w_1-w_2 space to illustrate how different regions of weights are determined.

- Pareto optimality gap of $\mathcal{P}(\mathcal{I}^0)$ against $\mathcal{P}(\mathcal{I}^1)$ is defined as

$$\sum_{i \in \mathcal{P}(\mathcal{I}^0)} \min_{i' \in \mathcal{P}(\mathcal{I}^1)} \min_{k \in \mathcal{K}} w_{i,k}(v_{i',k} - v_{i,k})_+,$$

  which measures the extent to which set $\mathcal{P}(\mathcal{I}^0)$ is dominated by $\mathcal{P}(\mathcal{I}^1)$. The term $(v_{i',k} - v_{i,k})_+ := \max\{v_{i',k} - v_{i,k}, 0\}$ detects any positive gap between individuals $i' \in \mathcal{P}(\mathcal{I}^1)$ and $i \in \mathcal{P}(\mathcal{I}^0)$ on trait k. Then, the term $\min_{k \in \mathcal{K}} w_{i,k}(v_{i',k} - v_{i,k})_+$ identifies the smallest weighted gap across all traits in order to determine the extent to which individual i is dominated by i'. Next, the term $\min_{i' \in \mathcal{P}(\mathcal{I}^1)} \min_{k \in \mathcal{K}} w_{i,k}(v_{i',k} - v_{i,k})_+$ identifies the individual i'

that dominates i by the least amount. This term is the Pareto optimality gap between individual i and $\mathcal{P}(\mathcal{I}^1)$, and the summation of which over all individuals in $\mathcal{P}(\mathcal{I}^0)$ gives the Pareto optimality gap of the set $\mathcal{P}(\mathcal{I}^0)$ against $\mathcal{P}(\mathcal{I}^1)$.

The Pareto optimality gap can be considered as the minimal amount of trait improvements in $\mathcal{I}^0$ necessary to break the dominance of $\mathcal{P}(\mathcal{I}^1)$ over any individual in $\mathcal{P}(\mathcal{I}^0)$. A noteworthy observation here is that the Pareto optimality gap between $i \in \mathcal{P}(\mathcal{I}^0)$ and $i' \in \mathcal{P}(\mathcal{I}^1)$ is zero if $v_{i',k'} = v_{i,k'}$ for one trait k' and $v_{i',k} > v_{i,k}$ for all others $k \in \mathcal{P} \backslash k'$. This may be counter-intuitive but is also defensible because, although i is dominated by i', it takes an arbitrarily small positive improvement in individual i on trait k' to break the dominance.

- Diversity of $\mathcal{P}(\mathcal{I}^0)$ is defined as

$$\sum_{k \in \mathcal{K}} \left[\frac{w_k}{|\mathcal{P}(\mathcal{I}^0)|} \sqrt{\sum_{i \in \mathcal{P}(\mathcal{I}^0)} \sum_{i' \in \mathcal{P}(\mathcal{I}^0)} (v_{i,k} - v_{i',k})^2} \right],$$

  which measures the weighted average Euclidean distance of all Pareto optimal solutions within set $\mathcal{P}(\mathcal{I}^0)$.

An ideal MTGS algorithm should be able to produce progeny with a small Pareto optimality gap (not being dominated by progeny produced from competing algorithms) and a large diversity (offering different trade-off options in traits).

4.4 Computational Experiments

We compared the performances of index selection and L-shaped selection with computational experiments using a maize data set considering two traits: plant height and ear diameter. No restrict assumptions are required in terms of the correlation between traits, i.e., they could be correlated, partially correlated or not correlated. Moreover, it should be mentioned that, as the purpose of recently published MT-LAS method Moeinizade et al. (2020) was different from this study, it has not been selected as a benchmark. MT-LAS is applicable when breeders are

interested in maximizing one trait and keeping the other trait in a desired range, however, in this paper, we aim at maximizing both traits.

4.4.1 Data Set

We used 200 maize inbred lines of 369 shoot apical meristem population distributed across the 10 chromosomes [ISU (2020)]. We extracted 1000 single nucleotide polymorphism (SNPs) out of the total of 1.4 million SNPs that were collected using genome-wide association study used by (Leiboff et al., 2015), merged with additional SNPs genotyped using tGBS [Schnable et al. (2013)], and those which were phased using Beagle [Browning and Browning (2009)]. Recombination rates in this population were estimated using the genetic map developed from the maize nested association mapping population. Genetic effects of the two traits were extracted from [Bernardo and Yu (2007)].

4.4.2 Breeding process

In each simulation of the breeding process, 200 individuals were randomly selected from the 369 inbred lines to form an initial population. In each of the subsequent generations, two individuals were selected using either index selection or L-shaped selection to produce 200 progeny in the next generation. The genetic values $v_{i,k}$ of all 200 individuals i for the two traits k in the fifth generation were used for performance analyses. Nine groups of this breeding process were simulated, each for a different set of weight parameters $w_1 \in \{0.1, 0.2, ..., 0.9\}$ and $w_2 = 1 - w_1$ with ten independent repetitions. The designed computational experiment platfrom is available on this *GitHub repository* link.

4.4.3 Results and Discussions

We compared the performances of index selection and L-shaped selection with respect to normalized plant heights and ear diameters of progeny in the final generation ($T = 5$), with both traits to be maximized. Figure 4.3 shows the aggregated results of 90 experiments (9 weights by

10 repetitions) using two algorithms; only those progeny that were Pareto optimal within each experiment were plotted. The x-axis and y-axis represent the normalized plant height and normalized ear diameter, respectively. It can be seen that L-shaped selection resulted in better-performing progeny in terms of both genetic gain and diversity.

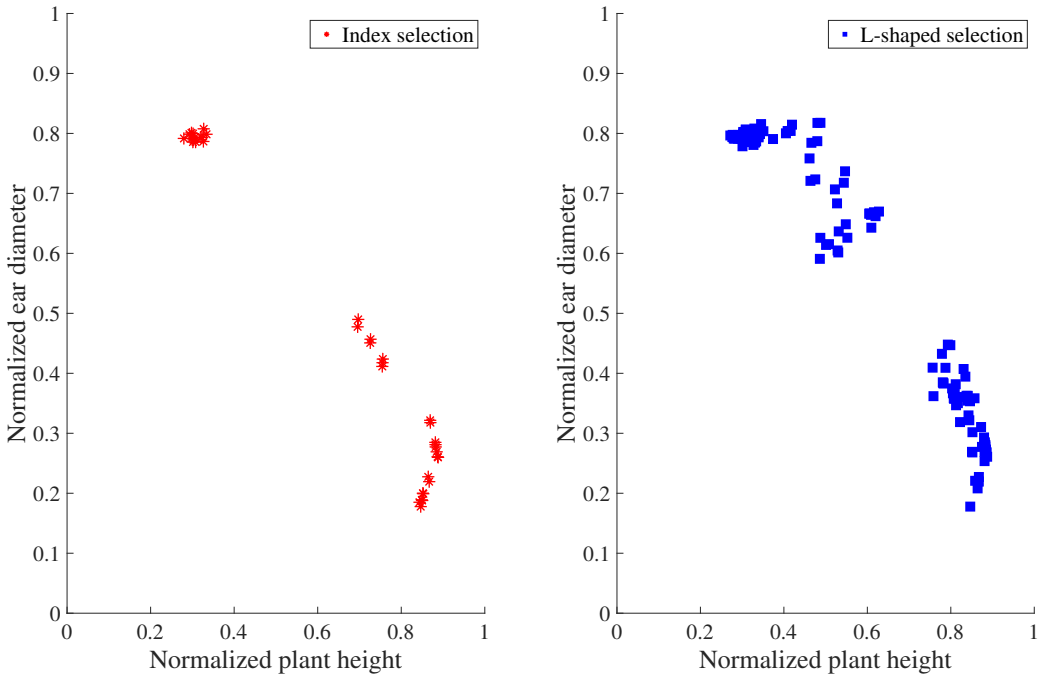


Figure 4.3: Performance of progeny in the final generation for index selection (left) and L-shaped selection (right).

The optimal Pareto frontier of Figure 4.3 is shown in Figure 4.4. It can be seen that, for all different sets of weight parameters $w_1 \in \{0.1, 0.2, ..., 0.9\}$ and $w_2 = 1 - w_1$, L-shaped selection outperforms index selection, since the optimal Pareto frontier for L-shaped selection not only dominates that for index selection but also is more diverse.

Table 4.3 shows the Pareto optimality gap and diversity of all progeny in the final generation. The Pareto optimality gap assesses the capability of a selection method in resulting in Pareto optimal progeny; the lower the gap, the more Pareto optimal the progeny. We assessed the Pareto

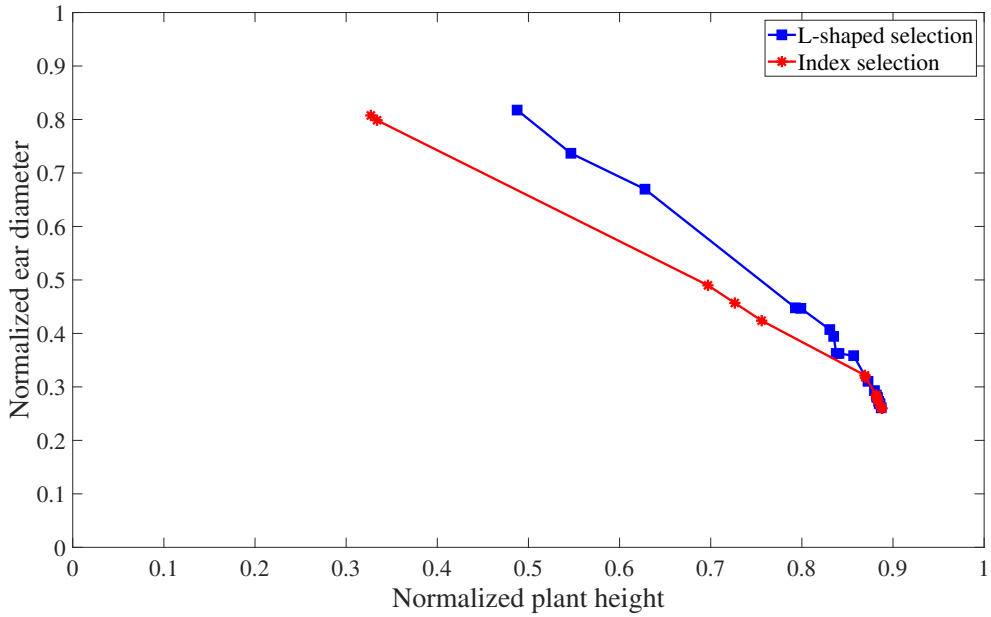


Figure 4.4: Pareto frontiers of progeny in the final generation for index selection and L-shaped selection.

optimality gap of index selection and L-shaped selection against the combined sets of progeny from the two approaches. As can be seen in Table 4.3, the Pareto optimality gap for L-shaped selection is zero, meaning that none of its Pareto optimal progeny was dominated by those from index selection. In contrast, index selection had a Pareto optimality gap of 0.2916, meaning that the Pareto optimal progeny from index selection was on average dominated by 0.2916 by those from L-shaped selection. The table also shows that progeny in the final generation produced using L-shaped selection were more diverse than those using index selection, in terms of representing different trade-offs between the two traits. These observations were consistent with results from Figure 4.3.

Table 4.3: Pareto optimality gap and diversity of index selection and L-shaped selection.

Metric / Method	Pareto optimality gap	Diversity
Index selection	0.2916	0.1933
L-shaped selection	0	0.2582

4.5 Conclusion

We presented the L-shaped selection for multi-trait genomic selection and demonstrated its improvements over the commonly used index selection in the capability to produce elite progeny with better genetic traits and higher diversity. Motivated by two limitations of index selection, the new approach made two major contributions to MTGS. First, L-shaped selection is robust against different measurement units used for multiple traits. Second, L-shaped selection guarantees to find all Pareto optimal solutions with appropriate weight parameters. Moreover, we introduced two metrics to quantify the capability of MTGS algorithms in accelerating genetic gain and preserving genetic diversity.

Besides theoretical contributions, L-shaped selection also outperformed index selection in computational experiments using a maize data set. The results demonstrate that L-shaped selection outperforms index selection in producing better-performing and more diverse population, considering plant height and ear diameter as two traits. However, the proposed L-shaped selection method is applicable to any breeding problem as long as appropriate genetic phenotypic data are available.

This study is not without limitations that could be addressed in follow-up research studies. Firstly, one can integrate the objective function in the L-shaped selection formulation in more sophisticated algorithms for genomic selection [Gorjanc et al. (2018); De Beukelaer et al. (2017); Goiffon et al. (2017); Moeinizade et al. (2020)] rather than a straightforward truncation selection as used in this study. Secondly, although we have considered only two traits in our study, the method applies to an arbitrarily large number of traits. A comprehensive case study with appropriate data from a large number of traits would be of interest for a future research project. Finally, it would be more realistic to consider dominance, epistatic, and environmental effects in calculating genetic trait values of individuals [Amini et al. (2021)], however, only additive effects have been considered in this study.

4.6 References

Amini, F., Franco, F. R., Hu, G., and Wang, L. (2021). The look ahead trace back optimizer for genomic selection under transparent and opaque simulators. *Scientific Reports*, 11(1):1–13.

Bernardo, R. (2002). *Breeding for quantitative traits in plants*.

Bernardo, R. and Yu, J. (2007). Prospects for genomewide selection for quantitative traits in maize. *Crop Science*, 47(3):1082–1090.

Browning, B. L. and Browning, S. R. (2009). A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. *The American Journal of Human Genetics*, 84(2):210–223.

Burgess, J. C. and West, D. R. (1993). Selection for grain yield following selection for ear height in maize. *Crop Science*, 33(4):cropsci1993.0011183X003300040006x.

De Beukelaer, H., Badke, Y., Fack, V., and De Meyer, G. (2017). Moving beyond managing realized genomic relationship in long-term genomic selection. *Genetics*, 206(2):1127–1138.

Desta, Z. A. and Ortiz, R. (2014). Genomic selection: genome-wide prediction in plant improvement. *Trends in Plant Science*, 19(9):592–601.

Goiffon, M., Kusmec, A., Wang, L., Hu, G., and Schnable, P. S. (2017). Improving response in genomic selection with a population-based selection strategy: optimal population value selection. *Genetics*, 206(3):1675–1682.

Gorjanc, G., Gaynor, R. C., and Hickey, J. M. (2018). Optimal cross selection for long-term genetic gain in two-part programs with rapid recurrent genomic selection. *Theoretical and Applied Genetics*, 131(9):1953–1966.

Hazel, L. N. (1943). THE GENETIC BASIS FOR CONSTRUCTING SELECTION INDEXES. *Genetics*, 28(6):476–490.

Hazel, L. N. and Lush, J. L. (1942). The efficiency of three methods of selection. *Journal of Heredity*, 33:393–399.

ISU (2020). Shoot apical meristem (sam) diversity panel genetic markers and map. https://iastate.figshare.com/s/374176500b04fd6f3729. [Online; accessed July-23-2020].

Jannink, J.-L., Lorenz, A. J., and Iwata, H. (2010). Genomic selection in plant breeding: from theory to practice. *Briefings in Functional Genomics*, 9(2):166–177.

Jia, Y. and Jannink, J.-L. (2012). Multiple-trait genomic selection methods increase genetic value prediction accuracy. *Genetics*, 192(4):1513–1522.

Leiboff, S., Li, X., Hu, H.-C., Todt, N., Yang, J., Li, X., Yu, X., Muehlbauer, G. J., Timmermans, M. C., Yu, J., et al. (2015). Genetic control of morphometric diversity in the maize shoot apical meristem. *Nature Communications*, 6(1):1–10.

Lorenzana, R. E. and Bernardo, R. (2009). Accuracy of genotypic value predictions for marker-based selection in biparental plant populations. *Theoretical and Applied Genetics*, 120(1):151–161.

Lynch, M., Walsh, B., et al. (1998). Genetics and analysis of quantitative traits.

Meuwissen, T. and Goddard, M. (2010). Accurate prediction of genetic values for complex traits by whole-genome resequencing. *Genetics*, 185(2):623–631.

Moeinizade, S., Kusmec, A., Hu, G., Wang, L., and Schnable, P. S. (2020). Multi-trait genomic selection methods for crop improvement. *Genetics*, 215(4):931–945.

Rutkoski, J., Poland, J., Mondal, S., Autrique, E., Pérez, L. G., Crossa, J., Reynolds, M., and Singh, R. (2016). Canopy temperature and vegetation indices from high-throughput phenotyping improve accuracy of pedigree and genomic selection for grain yield in wheat. *G3: Genes, Genomes, Genetics*, 6(9):2799–2808.

Schnable, P. S., Liu, S., and Wu, W. (2013). Genotyping by next- generation sequencing. *U.S. Patent Application*, No. 13/739:874.

Williams, J. (1962). The evaluation of a selection index. *Biometrics*, 18(3):375–393.

CHAPTER 5. APPLICATION OF THE TWO-LAYER WRAPPER-EMBEDDED FEATURE SELECTION METHOD TO IMPROVE GENOMIC SELECTION

Fatemeh Amini *, and Guiping Hu *

* Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript to be submitted to *Machine Learning with Applications* journal

5.1 Abstract

Breeders and crop scientists aim to improve the genetic gain, i.e., increasing the rate of genetic improvement, within a breeding population over time. The performance of a Genomic Selection (GS) algorithm is affected by multiple factors, such as, the performance of the genomic prediction methods, breeding parent selection strategy and, the evaluation platform. This paper aims at improving the genetic gain via incorporating a two-layer feature selection (FS) method in the breeding process. This two-layer FS method is a pre-processing wrapper-embedded FS method to enhance the prediction performance which ultimately improve the genetic gain. In addition, this paper proposed a comprehensive platform where the performances of several GS and genomic prediction algorithms can be tested. The proposed method and evaluation platform has been validated with a real Maize data set demonstrated that incorporating the proposed FS method in GS algorithms creates better-performing progeny throughout the generations. Moreover, the results confirm the superiority of incorporating two-layer FS method GS algorithms within multiple scenarios. The scenarios are built on the combination of different GS optimizer with different prediction methods within different nature simulators.

5.2 Introduction

Plant breeding relies on selection of parents over a number of generations to improve the crop performance. In the conventional breeding approach, elite parents are selected based on their trait/phenotype. This process requires a long period to achieve desired crop variety and it is less effective for some complex traits with low heritability Tuberosa (2012); Bhat et al. (2016). To address these challenges, Meuwissen et al. (2001) developed a new strategy called Genomic Selection (GS) in which breeding parent selection was based on marker/genotypic profile instead of the phenotype of individuals. The prediction model of GS integrates the genotypic and phenotypic data that further were used to estimate the GEBV for all breeding individuals from their genotypic data [Poland et al. (2012)]. GS offers number of merits over PS by reducing the breeding plant selection, increasing the efficiency of breeding process, and yield grain per unit of time and being environmentally insensitive [Rutkoski et al. (2011); Desta and Ortiz (2014); Goddard and Hayes (2007); Heffner et al. (2009); Jannink et al. (2010)]. The effectiveness of GS has been found to rely on the selection and mating strategy, and the accuracy of genomic prediction [Heffner et al. (2010)].

It is shown that an improved selection and mating strategy can provide room for higher and faster genetic gain. Several selection and mating strategies have been proposed to improve the crops genetic gain, such as conventional genomic selection Meuwissen et al. (2001), optimal haploid value (OHV) Daetwyler et al. (2015), and optimal population value (OPV) Goiffon et al. (2017). Furthermore, Moeinizade et al. (2019) proposed the look ahead selection (LAS) approach, which attempts to improve genetic gain by maximizing the probability of producing elite progeny by a target deadline. More recently Amini et al. (2021) proposed the look ahead trace back (LATB) algorithm, to further improve the performance of LAS in terms of genetic gain, especially with imperfect prediction of allele effects. In this paper, LAS, and LATB have been selected as the representatives of best-performing genomic selection algorithms.

Genomic Prediction (GP) used in GS is particularly well-suited for the prediction of quantitative traits controlled by many small-effect alleles Ribaut and Ragot (2007). It has been

used to identify highly parametric structures for modeling relationships between phenotypes and effects of hundreds or thousands of molecular markers [Meuwissen et al. (2001); González-Camacho et al. (2018)]. Several linear and nonlinear prediction methods have been adopted in the GS algorithms to improve the marker effects estimation accuracy. In this paper, the most common methods from both linear and nonlinear classes are selected to be used in the computational experiment. Ridge Regression from linear and Random Forest from nonlinear prediction method categories are adopted in this paper USAI et al. (2009); González-Recio and Forni (2011); Spindel et al. (2015).

A major challenge in using GP is relatively large number of markers (p) in comparison with limited number of phenotyped individuals [Meuwissen et al. (2001)]. This means that the datasets are underdetermined (also known as the $p >> n$ problem) and prone to overfitting due to the curse of dimensionality Whalen et al. (2020). Feature Selection (FS) has been introduced as a subdiscipline method in dimensionality reduction class to address the raised issue Whalen et al. (2020). The goal of FS method is to achieve the smallest, most powerful subset of features to not only reduce the computation time but also improve the prediction accuracy Huang and Wang (2006); Lin et al. (2015). It can be observed that the performance of a FS mechanism can be improved if it is carefully combined with another FS method. A hybrid FS method will further reduce the feature space and facilitate the design of a more efficient and accurate prediction model. Therefore, we have adopted the two-layer wrapper-embedded feature selection method capable of reducing the feature space while maintaining/improving the prediction accuracy in GP Amini and Hu (2021).

The contributions of this study can be summarized as follows. Firstly, it was the first time that a wrapper-embedded two-layer FS method has been integrated in the GS breeding cycles to improve the efficiency of GS algorithms. Secondly, a comprehensive comparison framework is designed in which the performance of multiple GS algorithms along with different prediction methods under different nature simulators in presence or absence of a FS method can be analyzed. Thirdly, despite most of the previous research that focused on the effectiveness of

incorporating a FS method on the prediction accuracy, we analyzed the FS effects on the genetic gain improvement.

The rest of the paper is organized as follows. Section 5.3 provides background on the mathematical model of the adopted two-layer FS method along with a brief description on benchmark prediction models, GS optimizers and nature simulators. Section 5.4 describes the case study and the simulation experiment settings. In section 5.5, the performance of different GS optimizers have been compared within several scenarios described in this section as well. Finally, section 5.6 concludes this study and suggests future research directions.

5.3 Materials and Methods

The ultimate breeding goal is assisting breeders to accelerate the crops genetic gain. However, due the time-consuming nature of the actual breeding process, a simulation platform can be designed to analyze the GS algorithms before application on the crop fields. Figure 5.1 demonstrate the proposed simulation and decision making platform inspired by Amini et al. (2021). In the designed simulation platform, the GS optimizer determines the crosses to make based on historical genotype (G^t) and phenotype data (P^t), in which a GP method is essential in the breeding parent selection. Additionally, we have included the two-layer FS method as a pre-process step to the GP methods to improve their performances. Then, nature determines the next generation genotype (G^{t+1}) as a result of the crosses and produces the next generation phenotype (P^{t+1}) as a result of the genotype and environment interactions, and the nature simulator attempts to mimic how nature works. In the following subsections, brief descriptions of each of the components of the simulation framework shown in figure 5.1, are discussed.

5.3.1 GS Optimizer

The word "optimizer" here refers to the selection approach each GS algorithm used to select elite breeding in each generation to construct the next generation. To illustrate the proposed method, two state-of-art GS optimizers, LAS and LATB, have been selected to be analyzed in

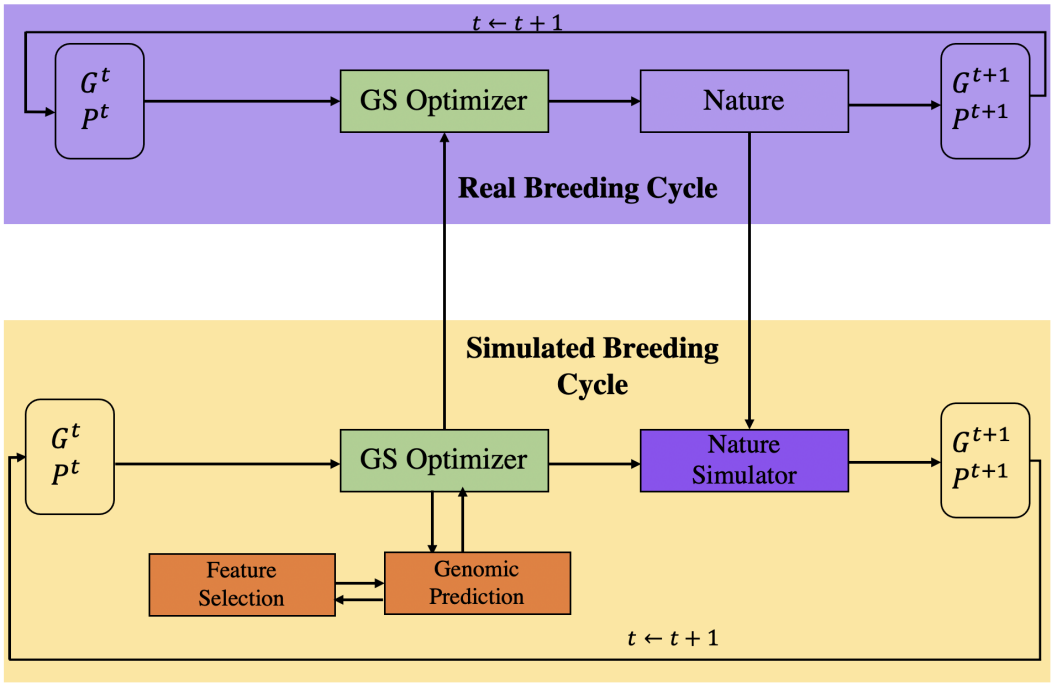


Figure 5.1: The proposed decision making platform

this paper. They have been selected due their outperformance in terms of genetic gain and genetic diversity in long-term horizon. Descriptions and summaries of both GS optimizers have been included below.

- Look Ahead Selection (LAS)

  LAS uses a heuristic model to maximize the expected GEBV of the best offspring in the terminal generation given a limited amount of resources. One of the contributions of LAS was to incorporate time and budget limitation in GS algorithms. It is highly sensitive to the final generation or the deadline of the breeding cycle and attempts to improve the genetic gain in the final generation of the breeding cycle while maintain genetic gain over early generations as well. LAS assumes that all progeny will be randomly crossed with each other throughout the entire breeding process and selects the best crosses out of the solution space to maximize the genetic gain in the final generation. Moreover, LAS is built on the

assumption of the existence of only additive effects while other non-additive effects have been seen in plants genetic decomposition Moeinizade et al. (2019).

- Look Ahead Trace Back (LATB) Selection

  LATB method proposed by (Amini et al., 2021) addressed the efficiency and compatibility challenges of LAS while maintaining its benefits. Similar to LAS, LATB also looks at the final generation determined by the breeders, then it simulates the breeding process toward the final generation, finds the best-performing progeny and traces back their ancestors in the current generation and declares them as the breeding parents. In the simulation process, it makes the pooling crosses only with well-performing parents, so it is more efficient than LAS. It is also compatible with more complex genomic prediction models in which, non-additive effects can be captured.

5.3.2 Genomic Prediction

As discussed earlier, each GS optimizer includes a genomics prediction method to predict the phenotype which further will be used for breeding parent selection by GS optimizer. Multiple linear and nonlinear prediction models can be adopted in the simulated breeding process, however, as the purpose of this study is to assess the performance of the GS optimizers with and without using the two-layer feature selection method, we selected two of the most popular methods, namely, Ridge Regression from the linear class and Random Forest from nonlinear class of prediction models. Brief descriptions of each of these prediction methods are included below.

- Linear - Ridge Regression

  Ridge Regression (RR) used to estimate the allele effect vector β to further predict individuals' phenotypes Hoerl and Hoerl (1962); Hoerl and Kennard (1970). Ridge Regression estimates the allele effect vector through Eq.(5.1).

$$\hat{\beta}_k^* = [G^\top G + kI_p]^{-1} G^\top P \tag{5.1}$$

where I_p is the $p \times p$ identity matrix, $k \in [0, 1]$ is a parameter for balancing bias and variance, G, and P represent the Genotype and phenotype information, respectively.

- Nonlinear - Random Forest

  Random Forest is an ensemble machine learning method that combines multiple decision trees which can generally reduce the variance of decision trees. As decision trees are computationally expensive and prone to overfitting, and tend to find local optima because they cannot go back after they have made a split, we turn to Random Forest which shows the power of combining many tree into one model. A Random Forest is a meta-estimator (i.e. it combines the result of multiple predictions) which aggregates many decision trees, with some helpful modifications as below:

  – The number of features that can be split on at each node is limited to a percentage of the total (which is known as a hyper-parameter in Random Forest). This ensures that the ensemble model does not rely too heavily on any individual feature, and makes appropriate use of all predictive features.

  – Each tree draws a random sample from the original data set when generating its splits, adding a further element of randomness that prevents overfitting.

  The above modifications can prevent the trees from being too highly correlated. Now the ensemble prediction is calculated by averaging the predictions of the all trees producing the final prediction Hastie et al. (2009).

$$\hat{P_i}^{final} = \frac{\sum_{t \in T} \hat{P}_i^t}{T} \tag{5.2}$$

In Eq.(5.2), T is the number of trees in the Random Forest. $\hat{P}_i^t$ is the predicted phenotype associated with individual i, predicted by tree t. $\hat{P_i}^{final}$ is the predicted value of sample i predicted by Random Forest.

5.3.3 Feature Selection

Reducing the feature space in high-dimensional data set such as genotype data will improve the prediction accuracy of genomic prediction. Therefore, in this paper, a two-layer wrapper-embedded method is incorporated in the simulation platform to improve the prediction accuracy of GP and further accelerate the genetic gain of GS algorithms. The adopted two-layer FS method consists of two layer, in the first layer, a Genetic Algorithm (GA) as a wrapper FS method has been adopted to come up with a subset of the informative SNP markers to predict the phenotype of individuals Cerrada et al. (2016); Liu et al. (2013). However, due to relatively large number SNP markers to number of individuals, a single wrapper method will not eliminate all irrelevant SNP marker. Therefore, a second layer is added to the first layer, in which an embedded method, namely Elastic Net (EN) is adopted to eliminate those remaining less informative SNP markers in the features space given in the first layer Park and Mazer (2018); Zou and Hastie (2005). Thus, the probability of having redundant/irrelevant SNP markers in the final FS model would decrease, preventing the prediction model to over fit. Fitness function in GA plays an important role since the individuals are evaluated based on it. The fitness function of the proposed FS method is defined as Eq.(5.3).

$$FF_{GA} = w_r * r_{RMSE} + w_p * R_p \tag{5.3}$$

w_r and w_p in Eq.(5.3) are the weights of the prediction error and the number of selected features, respectively, which satisfy the following conditions (Eqs.(2-3)).

$$w_r + w_p = 1 \tag{5.4}$$

$$w_r, w_p \geq 0 \tag{5.5}$$

r_{RMSE} is defined by the prediction error ($RMSE$) of the model divided by the average of response variable to demonstrate the error percentage. It is shown in Eq.(5.6).

$$r_{RMSE} = \frac{RMSE}{\overline{y}} \tag{5.6}$$

However, considering r_{RMSE} as the only performance metric of the prediction accuracy could be misleading. Therefore, it should be considered along with r_{fit} and *bias-variance* test.

The fraction of selected features is defined as R_p in Eq.(5.3), where f_p is a binary variable that denotes if feature p is included in a particular individual or not (Eq. 5.7).

$$0 \leq R_p = \frac{\sum_{p=1}^{P} f_p}{P} \leq 1, f_p \in \{0, 1\} \tag{5.7}$$

More details on this method can be found in Amini and Hu (2021).

5.3.4 Nature Simulators

The motivation behind considering a nature simulator is the fact that breeding process that is time-consuming, resource-intensive, and high-risk. So, it is challenging to design, validate, and train the algorithms directly during the actual breeding process Li et al. (2012); Moeinizade et al. (2020). Therefore, a nature simulator that mimics nature reasonably well becomes critical for training and evaluating the GS optimizers. In this study, we have analyzed two nature simulators, as transparent and opaques simulators described as below.

- The first one is the conventional transparent simulators used in GS algorithms Daetwyler et al. (2015); Goiffon et al. (2017); Moeinizade et al. (2019) in which almost all information is known to the optimizer meaning that the phenotype of individuals are predicted based on all genotype information assumed to be available.

- Second of all, however, all genotypic information may not be accessible in nature, so to have a more realistic representation of nature, we used an opaque simulator in which only partial information is observable to the simulator. The purpose of using an opaque simulator is to reveal how an optimizer might interact with an opaque nature rather than predict how nature will act. More details on this type of simulators are described in Amini et al. (2021).

5.4 Computational Experiments

We used a data set that consists of around 1.4 million SNPs (single nucleotide polymorphism) from the 369 maize inbred lines of shoot apical meristem (SAM) population distributed across the 10 maize chromosomes ISU (2020). In this paper, we extracted about 100,000 SNPs from this data set and simulated their phenotype by combining genetic and environmental effects. In each generation, 200 individuals were produced based on the specific GS method and the reproduction algorithm Goiffon et al. (2017). In the first generation 200 randomly selected individuals out of 369 inbred lines has been chosen. The purpose is to test the performance of GS algorithms using different initial breeding materials. The duration of the breeding process is set to be $T = 10$ generations. In each generation, updated genotype and phenotype data will be provided to the optimizer, which will then select 10 crosses from the current population. Then, 20 progeny from each cross has been made, so that a constant population size of 200 is maintained throughout the breeding process.

As for the opaque simulator, out of 100,000 genes, only 1000 genetic markers are used as the genotype information to be fed in the prediction model to estimate the phenotype or the additive effects. This simulator represents a nature in which an insufficient number of genetic markers are used, and little to no dominance effects or epistatic effects exist.

Moreover, in this study, as we have tuned the hyper-parameters of the adopted prediction methods, separately for each GS optimizer in each generation and adopted the best one for further calculation. The hyper-parameters associated with each prediction model are described as follows.

- **Ridge Regression**: It is well known that the variance of $\hat{\beta}_k^*$ in Eq.(5.1) is a monotonically decreasing function of k, which becomes zero when $k = 0$ and the model reduces to the least square estimator Marquaridt (1970). In our experiments, we tuned k using a grid search within the $\{0, 0.1, 0.2, ..., 1\}$ range.

- **Random Forest**: Although random forest as an ensemble model is less sensitive to the parameter changing, considering its large number of hyper-parameters, in this study, three

more important hyper-parameters of the random forest were tuned using a random search approach. Those are during each split: Number of trees, maximum depth of each tree, and maximum number of features considered in each split.

Moreover, the parameters of the adopted two-layer feature selection require to be tuned in the breeding process, for each GS optimizer in each generation. As for the first layer, there are no universal fixed parameters for GA and as they significantly affect the GA efficiency, they need to be generally tuned to specific problems. However, some of the GA parameters such as, initial population size and mutation rate have been tuned in this paper, limited computational capacity does not allow to expand the search grid to find the global optimal parameters. Therefore, a discrete range of search space has been defined for these two parameters and the best set were selected to be adopted in the final model. Furthermore, w_r and w_p in Eq. 5.3 also have been tuned using a discrete grid search from $[0, 1]$. The grid search space is defined in Table 5.1.

Table 5.1: Weights in fitness function

w_r	0.15	0.5	0.85	1
w_p	0.85	0.5	0.15	0

As for the second layer, the degree to which model complexity is penalized in Elastic Net is controlled by weighting terms α and ρ. As the outcome of the Elastic Net is affected by α and ρ, tuning them should be done within the learning process Chen et al. (2019); Park and Mazer (2018); Wei et al. (2019). In this paper, both of these parameters have been tuned in their feasible region, for $\alpha > 0$, and for $0 \leq \rho \leq 1$.

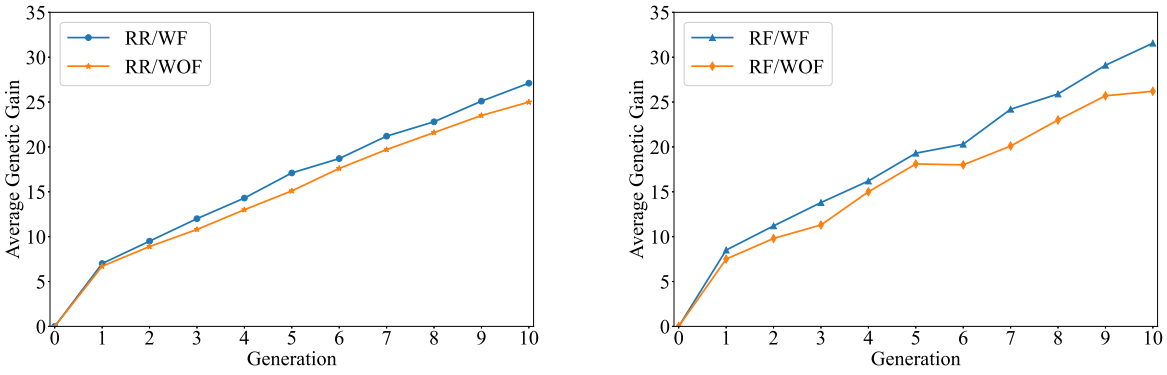
5.5 Results and Discussions

In this section, we analyzed the performances of LAS and LATB optimizers as the two best-performing GS methods, within several scenarios as follows. In the following figures, 5.2 - 5.5, the x-axis and y-axis demonstrate the generation number and the average genetic gain, respectively. The other abbreviations used in these figures are as follows: *RR: Ridge Regression,*

RF: Random Forest, WOF: Without using Feature Selection, WF: With Feature selection.
Moreover, the left component of each figure incorporates Ridge Regression as the prediction method in the breeding process and the right one use Random Forest. Therefore, the nature simulator, the GS optimizer and the prediction method are constant in each sub figures and the status of feature selection method is the only parameter changing.

Scenario 1: LAS performance in transparent simulator

In this scenario, the effect of using the two-layer feature selection method on the performance of LAS has been analyzed using different prediction methods within a transparent simulator. As can be seen in figure 5.2, LAS achieves higher genetic gain throughout the whole breeding process when the two-layer feature selection method is incorporated in all generation, regardless of the adopted prediction method. Furthermore, a fair comparison can be made on the performance of the prediction methods under similar circumstances. Comparing the the right and left sub figures of figure 5.2, the overall rate of improving genetic gain is higher with Random Forest as the GP method, e.g., it creates better-performing progeny in the final generation.(Refer to figure 5.6 in the Supplementary Data section for a comprehensive performance comparison.)

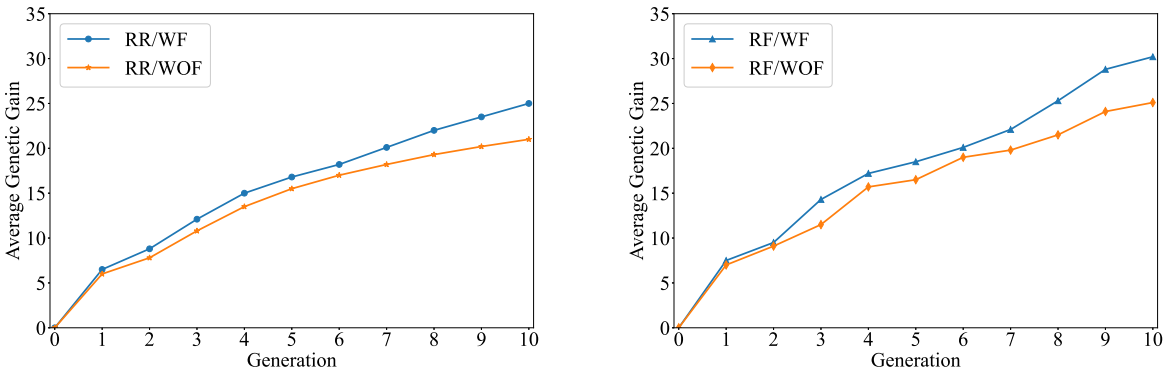


(a) (b)

Figure 5.2: Average genetic gain of LAS optimizer in **transparent simulator** using Ridge Regression (left), and Random Forest (right)

Scenario 2: LAS performance in opaque simulator

In this scenario, the adopted nature simulator has been changed to an opaque simulator, i.e., the effect of using the two-layer feature selection method on the performance of LAS has been analyzed using different prediction methods within an opaque simulator. Figure 5.3 demonstrates that, LAS achieves higher genetic gain in almost all generation using the two-layer feature selection method with both linear and nonlinear adopted prediction methods. Moreover, as the simulated breeding process proceeds to the end, the outperformance of adopting the two-layer FS method becomes more visible. Furthermore, a fair comparison can be made on the performance of the prediction methods under similar circumstances. Refer to figure 5.7 in the Supplementary Data section for analyzing the differences of prediction methods performances as well.



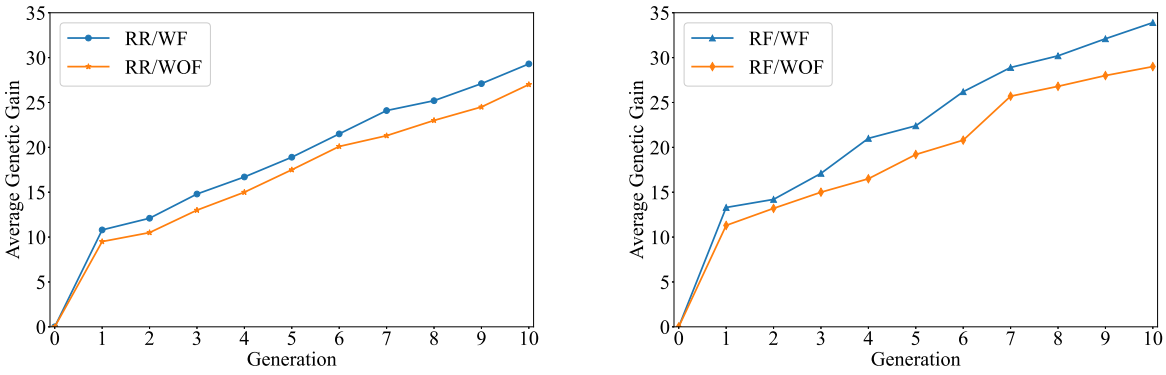
(a) (b)

Figure 5.3: Average genetic gain of LAS optimizer in **opaque simulator** using Ridge Regression (left), and Random Forest (right)

Scenario 3: LATB performance in transparent simulator

In this scenario, analyses have been conducted on LATB GS optimizer, i.e., the two-layer feature selection method is adopted within a transparent simulator to improve the performance of LATB in terms of the genetic gain. As can be seen in figure 5.4, using the two-layer FS method, LATB is capable of creating better-performing population throughout the breeding process when the two-layer feature selection method is incorporated regardless of the adopted prediction

method. However, a fair comparison can be made on the performance of the prediction methods under similar circumstances. (Refer to figure 5.8 in the Supplementary Data section for a comprehensive performance comparison.)



(a) (b)

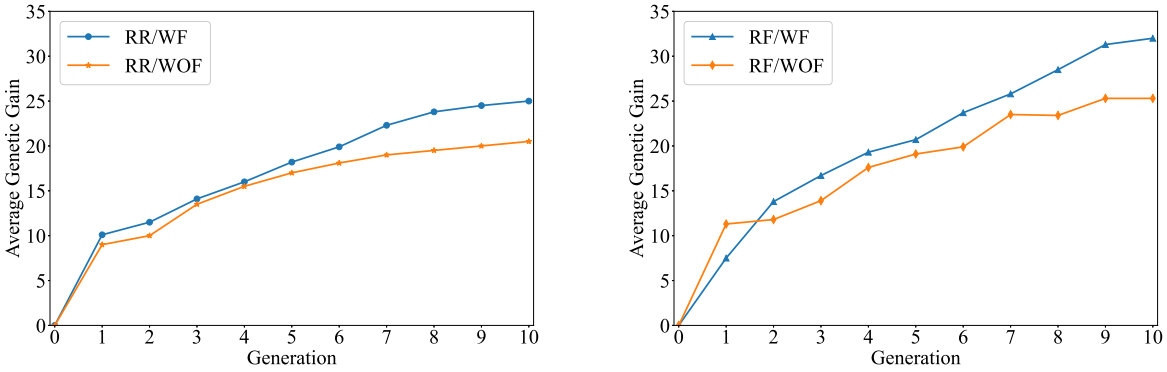
Figure 5.4: Average genetic gain of LATB optimizer in **transparent simulator** using Ridge Regression (left), and Random Forest (right)

Scenario 4: LATB performance in opaque simulator

In this scenario, the adopted nature simulator has been changed to an opaque simulator, i.e., the effect of using the two-layer feature selection method on the performance of LATB has been analyzed using different prediction methods within an opaque simulator. Figure 5.5 demonstrates that, LATB achieves higher genetic gain in almost all generations using the two-layer FS method with both linear and nonlinear prediction methods. Moreover, as the simulated breeding process proceeds to the end, the outperformance of adopting the two-layer FS method becomes more visible. Furthermore, a fair comparison can be made on the performance of the prediction methods under similar circumstances. Refer to figure 5.9 in the Supplementary Data section for analyzing the differences of prediction methods performances as well.

As the breeders perspective, the GS algorithm with higher genetic gain is preferable and based on the results, integrating Random Forest as the GP method along with the two-layer FS method within the LATB GS optimizer creates better-performing population. Moreover, the

benefit of the proposed comprehensive is not limited to select the best GS algorithm. It is capable of identifying the best GS algorithms considering the breeders feasible options and constraints. For instance, if a breeder considers one generation in the breeding cycle and uses LATB within and opaque simulator, Random Forest without using the FS method and Ridge Regression with the FS method would be the best options (Figure 5.9).



(a) (b)

Figure 5.5: Average genetic gain of LATB optimizer in **opaque simulator** using Ridge Regression (left), and Random Forest (right)

As a statistical point of view, it is important to quantitatively assess the effectiveness of the two-layer FS method on improving the genetic gain of GS algorithms. Therefore, a two-sample t-test has been designed in which, $H_o : \mu_{WF} = \mu_{WOF}$ vs $H_a : \mu_{WF} > \mu_{WOF}$, with significance level of $\alpha = 0.05$. μ_{WF} and μ_{WOF} stand for the average genetic gain in last generation with incorporating the two-layer FS method in the GS algorithms and not incorporating it, respectively. Table 5.2 summarizes the statistical differences between including and non including the two-layer FS method in GS algorithms in the last generation ($T = 10$). Based on the p-value resulted from the t-test, we can see that for all the GS optimizers and prediction methods within the both transparent and opaque breeding simulators, adopting the two-layer FS method within the GS algorithms is significantly effective on improving the performance of GS optimizers.

Table 5.2: Statistical differences of using two-layer FS method within the GS algorithms

GS Optimizer	Simulator	Prediction Method	p-value
LAS	Transparent	RR	$0.042 < \alpha = 0.05$
		RF	$0.035 < \alpha = 0.05$
	Opaque	RR	$0.032 < \alpha = 0.05$
		RF	$0.0315 < \alpha = 0.05$
LATB	Transparent	RR	$0.045 < \alpha = 0.05$
		RF	$0.03 < \alpha = 0.05$
	Opaque	RR	$0.029 < \alpha = 0.05$
		RF	$0.02 < \alpha = 0.05$

5.6 Conclusions

In this paper, we analyzed the effectiveness of adopting the two-layer wrapper-embedded feature selection method as a pre-processing step in the Genomic Selection algorithms to produce elite progeny, i.e., progeny with better genetic gain. The main motivation of this study was to decrease the feature space dimension of the genetic data to maintain the prediction accuracy in phenotype prediction while eliminating irrelevant and non-informative genetic data. This will further result in a better-performing population through several generations of the breeding process.

Moreover, the performance of the proposed two-layer feature selection method has been tested on two of the best-performing Genomic Selection optimizers such as, LAS and LATB, using linear and nonlinear prediction methods, within the transparent and opaque simulators. The results of the computational experiments on a Maize data set demonstrate that regardless of the type GS optimizer, prediction method, and the nature simulators, adopting the two-layer feature selection method within the GS algorithms produce better-performing progeny compare to not including it the algorithm. Furthermore, a two-test sample test confirms the significant outperformance of two-layer feature selection method in improving the genetic gain in long term.

This study is subject to a few limitations which suggest future research directions. Firstly, we have adopted a random search approach to tune the multiple hyper-parameters of the prediction

and FS method due to computational capacity limitation. Therefore, more complex approaches such as grid search over a wide range possible hyper-parameters could be addressed in the future studies that may improve the prediction accuracy and genetic gain. Secondly, although the effectiveness of the two-layer FS method on improving the crops genetic gain has been discussed in this paper, the effect of using the two-layer FS method on population genetic diversity should be addressed in the future studies. Finally, including other GS optimizers, such as conventional genomic selection in the comparison platform would confirm the capability of the two-layer FS method for optimizers without look ahead characteristics.

5.7 References

Amini, F., Franco, F. R., Hu, G., and Wang, L. (2021). The look ahead trace back optimizer for genomic selection under transparent and opaque simulators. *Scientific Reports*, 11(1):1–13.

Amini, F. and Hu, G. (2021). A two-layer feature selection method using genetic algorithm and elastic net. *Expert Systems with Applications*, 166:114072.

Bhat, J. A., Ali, S., Salgotra, R. K., Mir, Z. A., Dutta, S., Jadon, V., Tyagi, A., Mushtaq, M., Jain, N., Singh, P. K., et al. (2016). Genomic selection in the era of next generation sequencing for complex traits in plant breeding. *Frontiers in Genetics*, 7:221.

Cerrada, M., Zurita, G., Cabrera, D., Sánchez, R.-V., Artés, M., and Li, C. (2016). Fault diagnosis in spur gears based on genetic algorithm and random forest. *Mechanical Systems and Signal Processing*, 70:87–103.

Chen, W., Xu, C., Zou, B., Jin, H., and Xu, J. (2019). Kernelized elastic net regularization based on markov selective sampling. *Knowledge-Based Systems*, 163:57–68.

Daetwyler, H. D., Hayden, M. J., Spangenberg, G. C., and Hayes, B. J. (2015). Selection on optimal haploid value increases genetic gain and preserves more genetic diversity relative to genomic selection. *Genetics*, 200(4):1341–1348.

Desta, Z. A. and Ortiz, R. (2014). Genomic selection: genome-wide prediction in plant improvement. *Trends in Plant Science*, 19(9):592–601.

Goddard, M. and Hayes, B. (2007). Genomic selection. *Journal of Animal Breeding and Genetics*, 124(6):323–330.

Goiffon, M., Kusmec, A., Wang, L., Hu, G., and Schnable, P. S. (2017). Improving response in genomic selection with a population-based selection strategy: optimal population value selection. *Genetics*, 206(3):1675–1682.

González-Camacho, J. M., Ornella, L., Pérez-Rodríguez, P., Gianola, D., Dreisigacker, S., and Crossa, J. (2018). Applications of machine learning methods to genomic selection in breeding wheat for rust resistance. *The Plant Genome*, 11(2).

González-Recio, O. and Forni, S. (2011). Genome-wide prediction of discrete traits using bayesian regressions and machine learning. *Genetics Selection Evolution*, 43(1):7.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

Heffner, E. L., Lorenz, A. J., Jannink, J.-L., and Sorrells, M. E. (2010). Plant breeding with genomic selection: gain per unit time and cost. *Crop Science*, 50(5):1681–1690.

Heffner, E. L., Sorrells, M. E., and Jannink, J.-L. (2009). Genomic selection for crop improvement. *Crop Science*, 49(1):1–12.

Hoerl, A. E. and Hoerl, C. (1962). Application of ridge analysis to regression problems. *Chemical Engineering Progress*, 58(3):54–59.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Huang, C.-L. and Wang, C.-J. (2006). A ga-based feature selection and parameters optimizationfor support vector machines. *Expert Systems with applications*, 31(2):231–240.

ISU (2020). Shoot apical meristem (sam) diversity panel genetic markers and map. https://iastate.figshare.com/s/374176500b04fd6f3729. [Online; accessed July-23-2020].

Jannink, J.-L., Lorenz, A. J., and Iwata, H. (2010). Genomic selection in plant breeding: from theory to practice. *Briefings in Functional Genomics*, 9(2):166–177.

Li, X., Zhu, C., Wang, J., and Yu, J. (2012). Computer simulation in plant breeding. In *Advances in Agronomy*, volume 116, pages 219–264. Elsevier.

Lin, K.-C., Huang, Y.-H., Hung, J. C., and Lin, Y.-T. (2015). Feature selection and parameter optimization of support vector machines based on modified cat swarm optimization. *International Journal of Distributed Sensor Networks*, 11(7):365869.

Liu, S., Tai, H., Ding, Q., Li, D., Xu, L., and Wei, Y. (2013). A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. *Mathematical and Computer Modelling*, 58(3-4):458–465.

Marquaridt, D. W. (1970). Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation. *Technometrics*, 12(3):591–612.

Meuwissen, T., Goddard, M., Hayes, et al. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, 9(7):2123–2133.

Moeinizade, S., Kusmec, A., Hu, G., Wang, L., and Schnable, P. S. (2020). Multi-trait genomic selection methods for crop improvement. *Genetics*, 215(4):931–945.

Park, I. W. and Mazer, S. J. (2018). Overlooked climate parameters best predict flowering onset: Assessing phenological models using the elastic net. *Global change biology*, 24(12):5972–5984.

Poland, J. A., Endelman, J., Dawson, J., Rutkoski, J., Wu, S., Manes, Y., Dreisigacker, S., Crossa, J., Sánchez-Villeda, H., Sorrells, M., et al. (2012). Genomic selection in wheat breeding using genotyping-by-sequencing. *Plant Genome*, 5(3):103–113.

Ribaut, J.-M. and Ragot, M. (2007). Marker-assisted selection to improve drought adaptation in maize: the backcross approach, perspectives, limitations, and alternatives. *Journal of experimental botany*, 58(2):351–360.

Rutkoski, J. E., Heffner, E. L., and Sorrells, M. E. (2011). Genomic selection for durable stem rust resistance in wheat. *Euphytica*, 179(1):161–173.

Spindel, J., Begum, H., Akdemir, D., Virk, P., Collard, B., Redoña, E., Atlin, G., Jannink, J.-L., and McCouch, S. R. (2015). Genomic selection and association mapping in rice (oryza sativa): Effect of trait genetic architecture, training population composition, marker number and statistical model on accuracy of rice genomic selection in elite, tropical rice breeding lines. *PLOS Genetics*, 11(2):1–25.

Tuberosa, R. (2012). Phenotyping for drought tolerance of crops in the genomics era. *Frontiers in physiology*, 3:347.

USAI, M. G., GODDARD, M. E., and HAYES, B. J. (2009). Lasso with cross-validation for genomic selection. *Genetics Research*, 91(6):427–436.

Wei, C., Chen, J., Song, Z., and Chen, C.-I. (2019). Adaptive virtual sensors using snper for the localized construction and elastic net regularization in nonlinear processes. *Control Engineering Practice*, 83:129–140.

Whalen, I., Banzhaf, W., Al Mamun, H. A., and Gondro, C. (2020). *Evolving SNP Panels for Genomic Prediction*, pages 467–487. Springer International Publishing, Cham.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.

5.8 Appendix: Supplementary Data

In figures 5.6 - 5.9, the performances of genomic prediction method can be compared in similar circumstances.

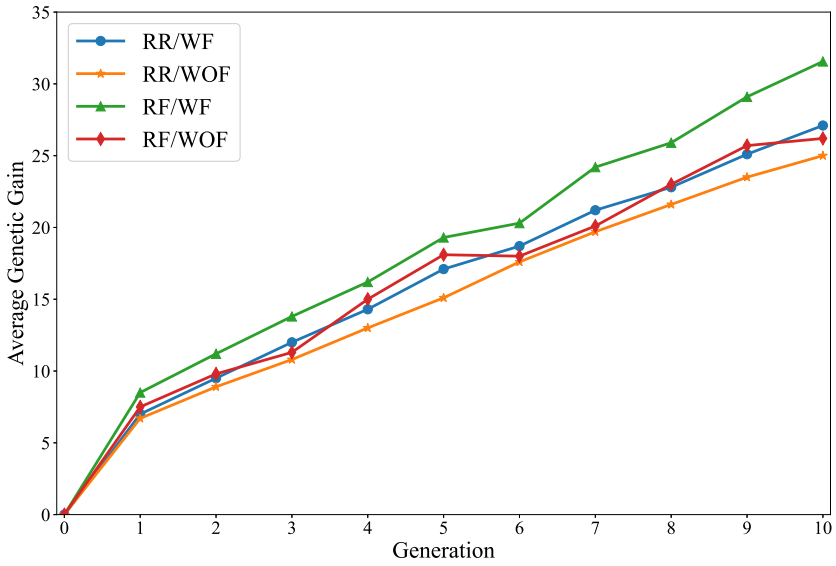


Figure 5.6: Average genetic gain using LAS optimizer in the **transparent simulator**

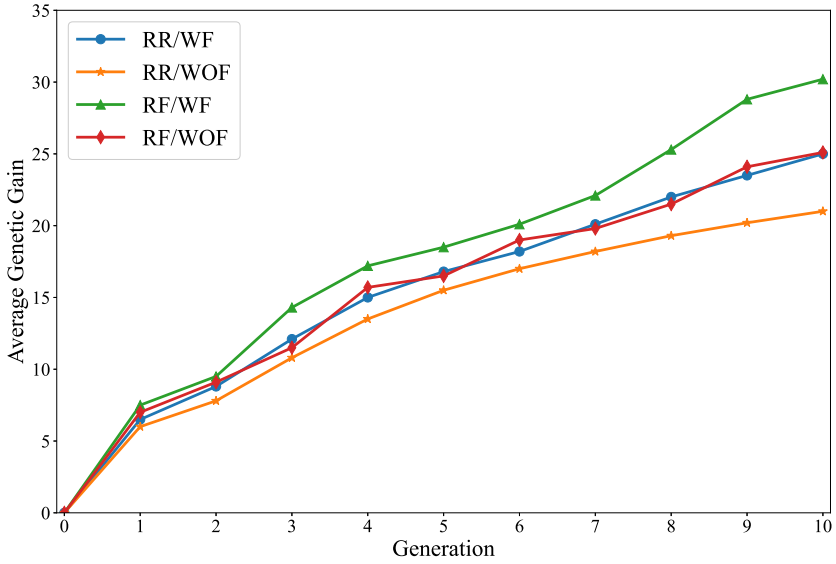


Figure 5.7: Average genetic gain using LAS optimizer in the **opaque simulator**

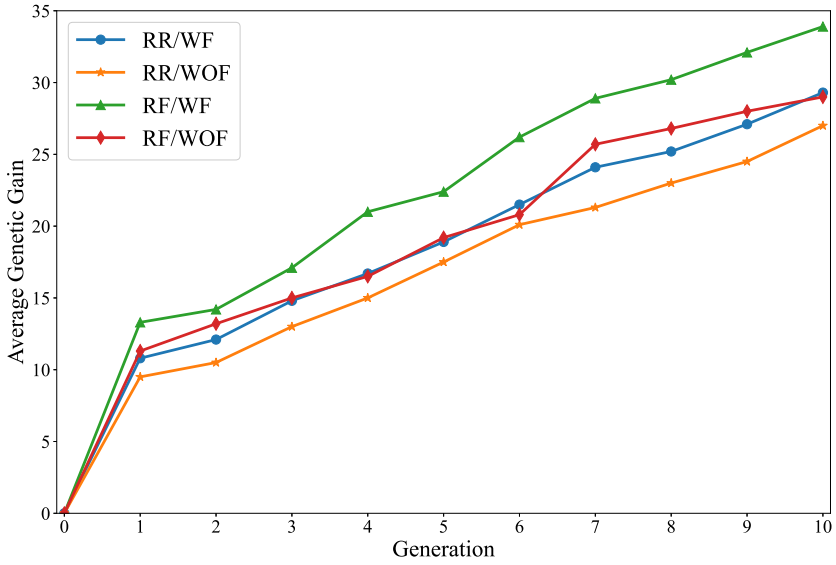


Figure 5.8: Average genetic gain using LATB optimizer in **transparent simulator**

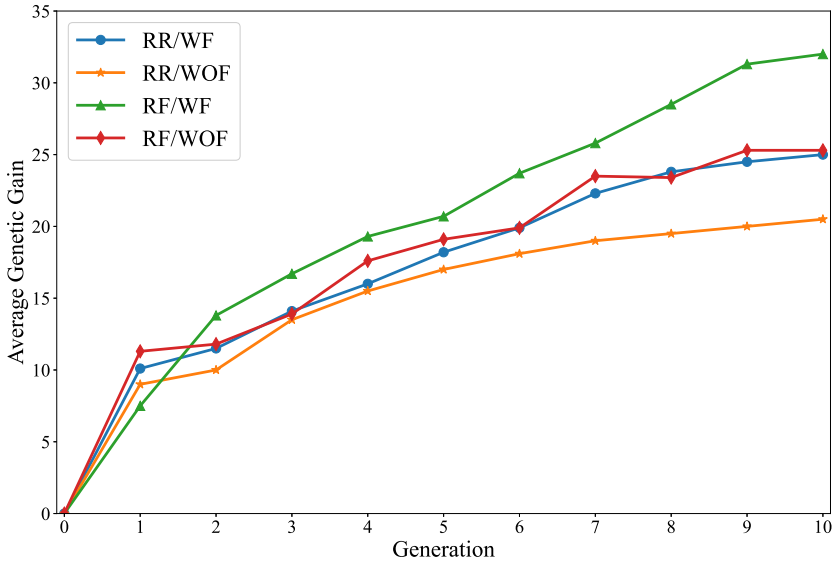


Figure 5.9: Average genetic gain using LATB optimizer in **opaque simulator**

CHAPTER 6. GENERAL CONCLUSION

This dissertation aims to address decision making challenges that breeders and analysts are facing, such as dealing with high-dimensional genetic data, producing better-performing population and creating a simulation platform to imitate the actual breeding process. They have been addressed via utilizing data analytics and optimization-based simulation approaches. The primary part of a breeding process is the breeding parents' selection in each generation, in a way that better-performing progeny could be produced in future generations. Genomic Selection algorithms have been designed to address this issue by focusing on improving the genetic gain and genetic diversity of crops throughout the generations. However, genomic selection approaches are influenced by multiple factors, such as the prediction accuracy in phenotype prediction, selection and mating strategy to maintain the population diversity while improving the genetic gain, considering single or multiple-traits at the same time. In this dissertation, we demonstrate how these factors affect the performance of genomic selection algorithm and develop optimization and machine learning techniques to enhance the crops genetic gain while maintaining genomic diversity.

The first paper developed a novel two-layer feature selection to address the curse of dimensionality problem in data set with high-dimensional feature space. This issue results in overfitting the prediction methods and significant cost on storing such data sets on huge databases. Therefore, feature selection (FS) methods should be adopted to select the best subset of salient features. Thus, in this paper, we combined a wrapper and an embedded FS method and propose a two-layer wrapper-embedded FS method for the first time. The first layer of this method incorporates Genetic Algorithm (GA) as the wrapper and Elastic Net (EN) as the embedded method and originally designed for regression problems. The main contribution of this study lies on the fitness function of the GA in the first layer of the proposed FS method that

integrates the rate of feature space reduction and prediction accuracy improvement, simultaneously and the user can determine the importance of each component using the relative weights that associated with them. Considering this fitness function, the two-layer FS method attempts to eliminate irrelevant and non-informative features while maintaining the prediction accuracy. This method has been compared with state-of-art FS method particularly on the diverse real genetic Maize inbred lines and the results demonstrated the outperformance of the proposed FS method both in terms of rate of feature space dimension reduction and prediction accuracy improvement.

The second paper introduced a new genomic selection (GS) algorithm built on a previous GS method, Look Ahead Selection (LAS), that aims at maximizing the expected GEBV of the best offspring in the terminal generation. However, in this study, we extended it to a new method called, Look Ahead Trace Back (LATB) algorithm in which the breeding parents of each generation are selected based on their performances in the final generation. This method creates a large pool of candidate crosses in each generation and simulates the breeding process upon each subset of them, then it identifies the elite progeny in the final generation and traces back to its ancestors in the current generation. In this case, a specific pair of parents may not be selected in the previous GS algorithms as the breeding parents since it does not perform well in the current generation. However, due the long-term perspective of LATB, it may select those parents if they were able to produce high-performing population in the final generation of the breeding process. Moreover, a new class of simulator, called opaque simulators, are designed in this paper in which the breeding processes are simulated on. These opaque simulators have the advantages of simulating the breeding process not knowing all genetic information of the population, while in previous transparent simulators, all the genetic data was observable to the simulator. The opaque simulators account for the nature uncertainty, thus the simulation result would be more robust in comparison with using transparent simulators.

The third paper developed the L-shaped selection method to improve the genetic gain in multi-trait genomic selection, when breeders are interested in improving multiple traits of a crop

rather than a single trait throughout the generations. Both contributions of this paper lie on the limitations of the common index selection that is being widely used in for multi-trait genetic improvements. The first contribution addresses the different traits measurement units that was ignored in the index selection. The L-shaped selection make the traits unit less via normalizing all desired traits to fall in same range, therefore, none of the traits are privileged because of their units. The only parameters that defines the relative importance of traits are the non-zero weights associated with each trait. The second contribution motivated by the convexity of the objective function of index selection, is a modified objective function in L-shaped selection method that is no longer essentially convex. This non-convexity enables the proposed method to capture all of the Pareto optimal solution, i.e., the elite individuals to be selected as the breeding parents. The L-shaped selection method has been compared with index selection using two traits with different measured units and multiple weight set alternatives for both traits. The results demonstrate the outperformance of the L-shaped selection in terms of genetic gain and diversity of the population in the final generation of the breeding process.

Finally, the last paper integrated the two-layer FS model proposed in the first paper, into multiple GS algorithms within the two different nature simulators. In this paper, we designed a comprehensive comparison platform that aims that analyzing the performance of adopting the two-layer FS method in terms of improving the genetic gain in different GS algorithm under different circumstances, such as different prediction methods and simulation platforms. Due to the computational capacity, we have selected the two best-performing GS algorithms, namely, LAS and LATB, and two commonly-used prediction methods in the genetic fields, namely, Ridge Regression from linear class, and Random Forest from non-linear class to assess the two-layer FS method in different situations. The performance of all combinations of mentioned GS algorithms and prediction methods with and without incorporating the FS method have been analyzed under transparent and opaque simulators, separately. The results on a real Maize inbred lines data set demonstrate that embedding the two-layer FS method, breeders are capable of producing

better-performing progeny not only in the final generation of the breeding process, but also in almost all mid-generations.

This dissertation is subject to some limitations which suggest future research directions. First, in the look ahead trace back selection, we assumed that there are no $G \times E$ effects attributed to the individuals' traits that can be included in the future studies. Second, although we introduced multiple versions of the opaque simulators to mimic the nature and ensure the robustness of the results, ultimate validation is required to identify the closest simulator to the nature. Third, whenever we tuned the hyper-parameters within the FS or GS algorithms, due to the computational limitation, a grid search within a constrained range of possible values were conducted. However, as the performance of these methods are influenced by their parameters, a more comprehensive search/tuning algorithm can be addressed in future research to further improve the performances of FS and GS methods. Finally, however, the L-shaped selection method for multi-trait genomic selection, it has been tested considering maximum of two/three positively correlated traits. Future simulations considering more traits negatively correlated are required to demonstrate the applicability of the proposed method on more realistic and sophisticated cases.