Journal of Integrative Agriculture 2026, 25(2): 775–787

Available online at ScienceDirect

Journal of Integrative Agriculture



RESEARCH ARTICLE

Using mixed kernel support vector machine to improve the predictive accuracy of genome selection

Jinbu Wang[1], Wencheng Zong[1], Liangyu Shi[2], Mianyan Li[1], Jia Li[1, 3], Deming Ren[1, 4], Fuping Zhao[1], Lixian Wang[1#], Ligang Wang[1#]

[1] *Key Laboratory of Animal Genetics, Breeding and Reproduction (poultry) of Ministry of Agriculture and Rural Affairs, Institute of Animal Sciences, Chinese Academy of Agricultural Sciences, Beijing 100193, China*
[2] *School of Animal Science and Nutritional Engineering, Wuhan Polytechnic University, Wuhan 430023, China*
[3] *College of Animal Science and Technology, Beijing University of Agriculture, Beijing 102206, China*
[4] *College of Animal Science and Technology, Qingdao Agricultural University, Qingdao 266109, China*

Highlights

- We introduced mixed kernel functions and associated theories into genomic selection for the first time.
- We applied the Laplacian kernel and Cosine kernel functions to the support vector machine algorithm in the field of genomic selection for the first time.
- We proposed optimal weights for the mixed kernel support vector machine applied to traits with varying heritabilities. The SVR_GS algorithm exhibited robust performance in genomic selection.

Abstract

The advantages of genome selection (GS) in animal and plant breeding are self-evident. Traditional parametric models have disadvantage in better fit the increasingly large sequencing data and capture complex effects accurately. Machine learning models have demonstrated remarkable potential in addressing these challenges. In this study, we introduced the concept of mixed kernel functions to explore the performance of support vector machine regression (SVR) in GS. Six single kernel functions (SVR_L, SVR_C, SVR_G, SVR_P, SVR_S, SVR_L) and four mixed kernel functions (SVR_GS, SVR_GP, SVR_LS, SVR_LP) were used to predict genome breeding values. The prediction accuracy, mean squared error (MSE) and mean absolute error (MAE) were used as evaluation indicators to compare with two traditional parametric models (GBLUP, BayesB) and two popular machine learning models (RF, KcRR). The results indicate that in most cases, the performance of the mixed kernel function model significantly outperforms that of GBLUP, BayesB and single kernel function. For instance, for T1 in the pig dataset, the predictive accuracy of SVR_GS is improved by 10% compared to GBLUP, and by approximately 4.4 and 18.6% compared to SVR_G and SVR_S respectively. For E1 in the wheat dataset, SVR_GS achieves 13.3% higher prediction accuracy than GBLUP. Among single kernel functions, the Laplacian and Gaussian kernel functions yield similar results, with the Gaussian kernel function performing better. The mixed kernel function notably reduces the MSE and MAE when compared to all single kernel functions. Furthermore, regarding runtime, SVR_GS and SVR_GP mixed kernel functions run approximately three times faster than GBLUP in the pig dataset, with only a slight increase in runtime compared to the single kernel function model. In summary, the mixed kernel function model of SVR demonstrates speed and accuracy competitiveness, and the model such as SVR_GS has important application potential for GS.

Keywords: genome selection, machine learning, support vector machine, kernel function, mixed kernel function

1. Introduction

The genomic selection (GS) (Meuwissen *et al.* 2001) uses high-density single nucleotide polymorphism (SNP) markers covering the entire genome to estimate the breeding value of individuals. It assumes that at least one of these markers is in linkage disequilibrium with all the quantitative trait loci

Received 8 November 2023; Received in revised form 31 January 2024; Accepted 28 February 2024; Available online 29 March 2024
Jinbu Wang, E-mail: w18439393365@163.com; #Correspondence Lixian Wang, E-mail: iaswlx@263.net; Ligang Wang, E-mail: wangligang01@caas.cn

© 2026 CAAS. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co. Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/). Peer review under responsibility of Editorial Board of *Journal of Integrative Agriculture*.
doi: 10.1016/j.jia.2024.03.083

(QTLs) controlling the traits (Goddard and Hayes 2008). The first application of GS was in dairy cattle, where it greatly increased genetic gain (Schaeffer 2006; Hayes *et al.* 2009; García-Ruiz *et al.* 2016). After more than 20 years of development, GS has been widely used in animal husbandry and plant breeding programs (Heffner *et al.* 2009; Wolc *et al.* 2011; Lillehammer *et al.* 2013; Crossa *et al.* 2017). One of the key factors to improve the power and efficiency of GS is the statistical model used to predict the breeding value of candidate individuals. For this reason, many researchers currently aimed at improving existing models and developing new models to increase the prediction performance of GS (Montesinos-López *et al.* 2019). At present, parametric methods are most commonly used for genome selection, including genomic best linear unbiased prediction (GBLUP) (VanRaden 2008), single-step GBLUP (SSGBLUP) (Legarra *et al.* 2009), ridge regression (Whittaker *et al.* 2000), least absolute shrinkage and selection operator (LASSO) (Tibshirani 2011) and Bayesian models (Meuwissen *et al.* 2001; Yi and Xu 2008; Habier *et al.* 2011). However, with the commercialization of various SNP chips and the continuous reduction in the cost of sequencing, the number of markers is usually greater than the number of samples with phenotypic data ("large p Small n"). Highly correlated structure of SNP data, make model fitting face more and more problems (Gonzalez-Recio *et al.* 2014). Moreover, parametric methods often face limitations in their ability to handle nonlinear effects and high-dimensional genome information (epistasis, dominance, or genotype-environment interactions), resulting in significant computational demands (Varona *et al.* 2018). Many studies have shown that consideration of nonlinear effects may enhance the genomic predictive power of complex traits (Gianola *et al.* 2018). Therefore, new strategies should be explored to more accurately estimate genomic breeding value.

Machine learning (ML) is the study of computer algorithms that are automatically improved through experience, which includes a large number of learning algorithms, such as random forests (RF) (Breiman 2001), support vector machines (SVM) (Cortes and Vapnik 1995), boosting (Shrestha and Solomatine 2006), neural networks (LeCun *et al.* 2015), and so on. And different algorithms show different performance and advantages in different industries and applications. ML has found successful applications in diverse fields such as finance (Hansen and Borch 2021), medicine (Handelman *et al.* 2018), biology (Weiskittel *et al.* 2021), and industry (Bowler *et al.* 2022). Machine learning methods aim to improve predictive performance measures through repeated observational experience. They have no model specification and can capture hidden information from large datasets, which is attractive in genomic information environments where multiple complex relationships exist (Gianola *et al.* 2011). SVM, introduced by Vapnik (Cortes and Vapnik 1995) in 1995, are considered among the most advanced machine learning algorithms for classification and regression tasks across various domains (Long *et al.* 2011). Compared to other ML methods, SVM is powerful in identifying subtle patterns in complex datasets (Aruna and Rajagopalan 2013). Due to the diversity of kernel functions, SVM also has a powerful and flexible ability to process various data. To date, SVM have been used to analyze various complex biological datasets (Byvatov and Schneider 2003; Huang *et al.* 2018).

The kernel function enables the SVM algorithm to transform the actual problem into a high-dimensional feature space through nonlinear mapping, and construct a linear regression function in the high-dimensional space to realize the nonlinear regression in the original space, which can effectively avoid the "dimension disaster" problem. Therefore, based on different kernel methods, SVM can produce different results, and choosing the correct kernel will improve the efficiency of SVM (Ibrikci *et al.* 2012). Kernel functions can be divided into two categories: global kernel functions and local kernel functions. The global kernel function has global characteristics, allowing data points that are far apart to have an impact on the value of the kernel function, such as the Polynomial kernel function, which has strong generalization performance and weak learning ability. The local kernel function has locality, and only data points that are very close to each other are allowed to have an impact on the value of the kernel function, such as Gaussian kernel function, which has strong learning ability and weak generalization performance (Kung 2014). Therefore, SVM using a single kernel function has certain limitations. To overcome these limitations and achieve improved SVM performance, mixed kernel function is constructed by adjusting the weight ratio of the local and global kernel functions (Smits and Jordaan 2002). By combining the advantages of both local and global kernel functions, the mixed kernel function demonstrates favorable performance. This approach has been successfully applied in various domains, including industry (Gao *et al.* 2022), medicine (Zhao *et al.* 2019), energy (Tian *et al.* 2017), among others.

In this study, we introduced mixed kernel functions into the field of genome selection, and combining two local kernel functions (Gaussian kernel, Laplacian kernel) with two global kernel functions (Polynomial kernel, Sigmoid kernel). Inspired by the successful performance of the Cosine kernel in KcRR (An *et al.* 2021), we also introduced the Cosine kernel into the support vector machine regression (SVR). Our objective was to evaluate the performance of different types of kernel functions for genome selection and to identify the best performing mixed kernel model and determine the optimal weight values. We compared SVR with two traditional genome prediction models (GBLUP and BayesB), as well as two popular machine learning algorithm models (KcRR and RF). We used prediction accuracy and mean squared error (MSE) and mean absolute error (MAE) as evaluation indicators to evaluating the predictive ability of each model. Through this research, we demonstrated the unique and excellent performance of mixed kernel functions in genome selection.

2. Materials and methods

2.1. Genomic prediction comparison models

GBLUP GBLUP is one of the most widely used models in breeding work, as follows:

$$y=1\mu+Zg+e$$

where y is the vector of corrected phenotypes of genotyped

individuals. μ is the overall mean, 1 is a vector of 1 s, g is the vector of genomic breeding values, e is the vector of residuals, and Z is an incidence matrix allocating records to breeding values. The random effects and the residual error are assumed that t g~N(0,Gσ_g^2) and e~N(0,Iσ_e^2) , where G was the genomic relationship matrix, and σ_g^2 and σ_e^2 were the additive genetic variance and the residual variance.

BayesB BayesB assumes that most SNPs have no effect, while a few SNPs have effects and each has different effect variances. The Bayes B forecasting model can be expressed as:

$$y=\sum_{j=1}^{p} m_j\alpha_j+e$$

where y is the phenotype vector; m_j is the genotype of the jth site; α_j is the effect value of the jth site, and α_j obeys the normal distribution α_j~N(0,$\sigma_{\alpha j}^2$) . BayesB's assumptions for the prior distribution of $\sigma_{\alpha j}^2$ are: with probability π, $\sigma_{\alpha j}^2$=0; with probability 1-π, p($\sigma_{\alpha j}^2$)=x^{-2}(v,S), where π is artificially given. The value of π determines the proportion of markers with an effect value to the total number of markers.

Random forest (RF) Random forest uses voting or averaging of multiple decision trees to determine the classification or prediction value of new instances. A random forest is essentially a collection of decision trees, each slightly different from the others. Random forests reduce the risk of overfitting by averaging the predictions of many decision trees. Random forest regression can be written as follows:

$$y=\frac{1}{M}\sum_{m=1}^{M} t_m(\psi_m(y:X))$$

where y is the predicted value of the random forest regression and the predictor variable $t_m(\psi_m(y:X))$ is a decision tree. M is the number of decision trees included in the random forest.

Kernel ridge regression (KRR) KRR uses the kernel trick to map the data to a high-dimensional kernel space, and then builds a ridge regression model to make the data linearly separable in this kernel space. The KRR prediction model can be expressed as:

$$y(x_i)=k'(K+\lambda I)^{-1}\hat{y}$$

where K is the Gram matrix: $K_{ij}=K(x_i,x_j)=\Phi(x_i)\cdot\Phi(x_j)^T$, $k'=K(x_i,x_j)$ with j =1, 2, 3, …, n, n is the total number of samples in the training set; I is the identity matrix, and λ is the ridge regression parameter. In this study, we use KcRR, which uses a Cosine kernel to construct KRR.

SVR SVR is the application of SVM to deal with quantitative responses in regression. SVR transforms the input space into a high-dimensional space through the nonlinear transformation defined by the kernel function, and obtains the learning process of the regression function in this high-dimensional space. In this study, we used epsilon support vector regression, and the SVR model formula can be expressed as:

$$f(x)=\beta_0+h(x)^T\beta$$

where $h(x)^T$ is the kernel function, β is the weight vector, and β_0 is the bias. The smoothest function f(x) is sought by minimizing the square of the vector norm β, the error of the predicted value of each training data is at most equal to ε. To allow for outliers, data with prediction errors larger than ε are penalized using an ε-insensitive loss function. The SVR problem formalized as:

$$\min_{\beta_0,\beta}\frac{1}{2}\|\beta\|^2+C\sum_{i=1}^{n}V(y_i-f(x_i))$$

where

$$V_\varepsilon(r)=\begin{cases}0,\text{if } |r|<\varepsilon\\ |r|-\varepsilon,\text{otherwise}\end{cases}$$

where C is the regularization constant and $V_\varepsilon(r)$ is the ε-insensitive loss. Through a series of optimization processes, SVR can be written as:

$$f(x)=\sum_{i=1}^{m}(\hat{a}_i-\alpha_i)k(x,x_i)+b$$

where $k(x_i,x_j)=\Phi(x_i)^T\cdot\Phi(x_j)$ is the kernel function.

Kernel function The commonly used kernel functions are as follows:

(1) Linear kernel function:

$k(x\cdot y)=(x\cdot y)$

(2) Polynomial kernel function:

$k(x\cdot y)=[(x\cdot y)+c]^d$

where d is the dimension of the kernel function.

(3) Gaussian kernel function:

$$K(x,y)=\exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$$

where σ is the radial base radius.

(4) Sigmoid kernel function:

$K(x,y)=\tanh[v(x,y)+c]$

where v is called the slope and c is called the intercept. This study introduces Laplacian kernel and Cosine kernel,

(5) Laplacian kernel is defined as:

$K(x,y)=\exp[-\gamma\|x-y\|_1]$

where $\|x-y\|_1$ is the Manhattan distance between the input vectors.

(6) Cosine kernel is defined as:

$$K(x,y)=\frac{xy^T}{\|x\|\|y\|}$$

Mixed kernel function The common construction form of the Mixed kernel function is as follows:

$K_{mix}=\lambda K_{|oca|}+(1-\lambda)K_{g|oba|}$

where λ is a scale factor, which adjusts the proportion of the global kernel and the local kernel. When λ is 0, the function is a global kernel; when λ is 1, the function is a local kernel.

(1) Mixture of Gaussian kernel function and Sigmoid kernel function (SVR_GS):

$$K(x,y)=\lambda\exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)+(1-\lambda)\tanh[v(x,y)+c]$$

(2) Mixture of Gaussian kernel function and Polynomial kernel function (SVR_GP):

$$K(x,y)=\lambda\exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)+(1-\lambda)[(x\cdot y)+c]^d$$

(3) Mixture of Laplacian kernel function and Sigmoid kernel function (SVR_LS):

$K(x,y)=\lambda\exp(-\gamma\|x-y\|_1)+(1-\lambda)\tanh[v(x,y)+c]$

(4) Mixture of Laplacian kernel function and Polynomial kernel function (SVR_LP):

$K(x,y)=\lambda\exp(-\gamma\|x-y\|_1)+(1-\lambda)[(x\cdot y)+c]^d$

2.2. Genotype and phenotype data

In this study, the performance of different genomic prediction methods was evaluated using two datasets of pig and wheat.

The pig data was obtained from a pig farm operated

by the pig improvement company (PIC) (Cleveland *et al.* 2012). There are 3,534 samples genotyped by Illumina PorcineSNP60 chip and five traits. An imputed genotype based on a probability score was used to replace missing information. Phenotypes were corrected for fixed effects or were weighted progeny mean corrected phenotypes. The selection of study traits, T1, T2, T3, and T4, was based on the gradient distribution of heritability, with each trait having a heritability of T1=0.07, T2=0.16, T3=0.38, and T4=0.58, respectively. Quality control measures were applied, which included the removal of sites failing the Hardy-Weinberg equilibrium test ($P<10^{-6}$), SNPs with a deletion rate exceeding 5%, and SNPs with a minor allele frequency (MAF) less than 0.01. Due to some individuals lacking complete phenotypic data, the number of individuals varied for each trait. As a result, the final number of SNPs retained for T1, T2, T3, and T4 were 45,025, 45,441, 44,190, and 44,151, respectively, for use in this study.

The wheat dataset is a collection of 599 historical wheat lines from the International Maize and Wheat Improvement Center (CIMMYT) Global Wheat Program with the trait of grain yield (Crossa *et al.* 2010). The environments represented in these trials were grouped into four major agroclimatic regions, which are precisely defined and widely used by the CIMMYT global breeding program. This dataset includes phenotype, genotype and pedigree information. Each wheat line was genotyped using 1,447 diversity array technology (DArT) markers generated by Triticarte (Canberra, Australian Capital Territory; https://www.Triticarte.com). The DArT markers were recorded as present (1) or absent (0) for each line. Minor allele markers with a frequency less than 0.05 are removed, and missing genotypes are estimated with samples from the marginal distribution of marker genotypes, $x_j \sim \text{Bernoulli}(\hat{y}_j)$, where $\hat{y}_j$ is the estimated allele frequency computed from the non-missing genotypes. After quality control, 1,279 markers were retained.

2.3. Assessing predictive performance

In this study, 10-fold cross-validation was used to evaluate the predictive performance of each method. Each dataset is randomly divided into 10 parts, each sub-sample will be used as a test set, and the remaining 9 sub-samples will be used as a training set. Prediction accuracy was measured by the Pearson correlation coefficient between corrected phenotypes and predicted breeding values. The correlation coefficient is calculated as follows:

$$r=\frac{\text{cov}(y,f)}{\sqrt{\text{var}(y)*\text{var}(f)}}.$$

At the same time, we also use mean square error (MSE) and mean absolute error (MAE) to evaluate the performance of the model. MSE can consider both prediction accuracy and bias, and MAE can better reflect the actual situation of prediction value error. Their calculation formula is as follows:

$$\text{MSE}=\frac{1}{m}\sum_{i=1}^{m}(f_i-y_i)^2,\ \text{MAE}=\frac{1}{m}\sum_{i=1}^{m}|f_i-y_i|$$

where, m represents the number of test sets in cross-validation, f is the vector of predicted values, and y is the vector of observed values.

2.4. Method implementation

The GBLUP and BayesB methods were implemented using the 'rrBLUP' and 'BGLR' packages in the R statistical software, respectively, while machine learning algorithms (SVR, RF, KcRR) were implemented through the respective algorithm modules in the 'scikit-learn' (Pedregosa *et al.* 2011). Additionally, the mixed kernel function is a custom kernel function. In order to observe the performance changes of the mixed kernel function with weight changes and the performance of each method in parameter adjustment, we used default parameters and optimal parameters for each method to run the program (excluding GBLUP). The optimal parameters are determined using the grid search algorithm, coupled with 10-fold cross-validation to assess the performance of each set of hyperparameters. During this process, the dataset is partitioned into a training set and a validation set. The best parameters are identified based on the model's performance on the validation set.

In the parameter adjustment process, for BayesB, we set the 'nIter' to 50,000 and 'burnIn' to 5,000. For RF, parameter 'n_estimators' range from 100 to 1,000 in steps of 100, 'max_depth' value is from 10 to 100, with a step size of 10, and the 'None' is added. The specific tuning parameters for SVR and KcRR are as follows: for 'C' (and alpha in KcRR), we considered values of 0.1, 1, 5, 10, 15, and 20. For the gamma parameter, we use powers of 10 with negative integers as increments, setting the parameter values within the range of 1×10^{-1} to 1×10^{-10}. For specific kernel function specific parameters, 'coef0' in the Sigmoid kernel function also uses 1×10^{-1}–10^{-10}, while in the poly kernel function, the 'degree' parameter takes values from 1 to 10 with a step size of 1. In the case of the mixed kernel function, combinations were generated based on the aforementioned basic parameters. The weight parameter (λ) was set to take values of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9.

Finally, to evaluate the computational time required by each method, we conducted all benchmark tests on a single server equipped with 40 cores (Intel(R) Xeon(R) Gold 6148 CPU@2.40GHz) and 754 GB of memory. Running only a single job at a time on the server.

3. Results

3.1. Comparison of the performance of mixed kernel function with weight variation

In this study, we utilize the abbreviations SVR_Lin, SVR_G, SVR_P, SVR_S, SVR_L and SVR_C to represent the models of Linear kernel function, Gaussian kernel function, Polynomial kernel function, Sigmoid kernel function, Laplacian kernel function and Cosine kernel function of SVR, respectively. Similarly, we define each mixed kernel function models as follows: SVR_GS (mixing of Gaussian and Sigmoid kernel functions), SVR_GP (mixing of Gaussian Kernel and Polynomial Kernel functions), SVR_LS (mixing of Laplacian kernel and Sigmoid kernel functions), SVR_LP (mixing of Laplacian kernel and Polynomial kernel functions).

Figs. 1 and 2 illustrate the trend of prediction accuracy with increasing weights values for SVR_GS, SVR_GP, SVR_LS and SVR_LP under optimal parameters in pig and wheat

datasets. Detailed prediction accuracy results for these four mixed kernel functions under both optimal and default parameters can be found in Appendices A–D. For a clearer comparison of the results obtained by mixing the two kernel functions, we added results with mixed kernel function weights of 0 and 1 in Figs. 1 and 2. We found that mixing global and local kernel functions can enhance the predictive accuracy of a single-kernel, with particularly noticeable improvement compared to two global kernel functions. In this study, we defined the single kernel function which constitutes the mixed kernel function as the 'mother kernel function'. The results indicated that, except in E1 and T3, the prediction accuracy of SVR_GS and SVR_LS outperform the prediction accuracy of their mother kernel SVR across all traits in both datasets. For instance, for T1, the prediction accuracy of SVR_GS is approximately 4.4% higher than that of SVR_G and about 18.6% higher than that of SVR_S. In E3, the prediction accuracy of SVR_LS exhibit an improvement of approximately 2.8% compared to SVR_L and around 35.2% compared to SVR_S. Similarly, for the SVR_GP, its prediction accuracy exceeds that of its mother kernel function in T2 and E3, and for the SVR_LP, its prediction accuracy surpasses that of its mother kernel function in T3, T4, and E3.

In the wheat dataset, an intriguing result was observed: the prediction accuracy for SVR_GS and SVR_GP was equal to that of SVR_LS and SVR_LP when varying parameters were applied. For the pig dataset, the prediction accuracy of SVR_GS and SVR_GP are also very similar to prediction accuracy of SVR_LS and SVR_LP. Following these results, we organized the weights of the four mixed kernel functions to achieve the highest accuracy under default and optimal parameters in Table 1. Notably, for mixed kernel functions, as the weight λ approaches 0, the influence of the global kernel function on the mixed kernel function becomes more pronounced, while as λ approaches 1, the impact of the local kernel function becomes more significant. In the pig dataset, under optimal parameters, we observed that when mixed the Gaussian or Laplacian local kernels with the Sigmoid kernel function, the prediction accuracy of T1 and T2 generally exhibits an initial increase followed by a decrease. The prediction accuracy for T3 and T4 demonstrates an upward trend. When mixed the two local kernels with the Polynomial kernel function, T1 and T3 tend to show an increase in prediction accuracy, while T2 and T4 tend to decrease. Under default parameters, when Gaussian or Laplacian local kernel functions are mixed with Sigmoid kernel functions, the prediction accuracy of T1 generally shows a decreasing trend, while the prediction accuracy of T3 and T4 shows an increasing trend. When mixed with the Polynomial kernel function under default parameters, T1, T2, and T3 generally exhibit an increase in prediction accuracy,

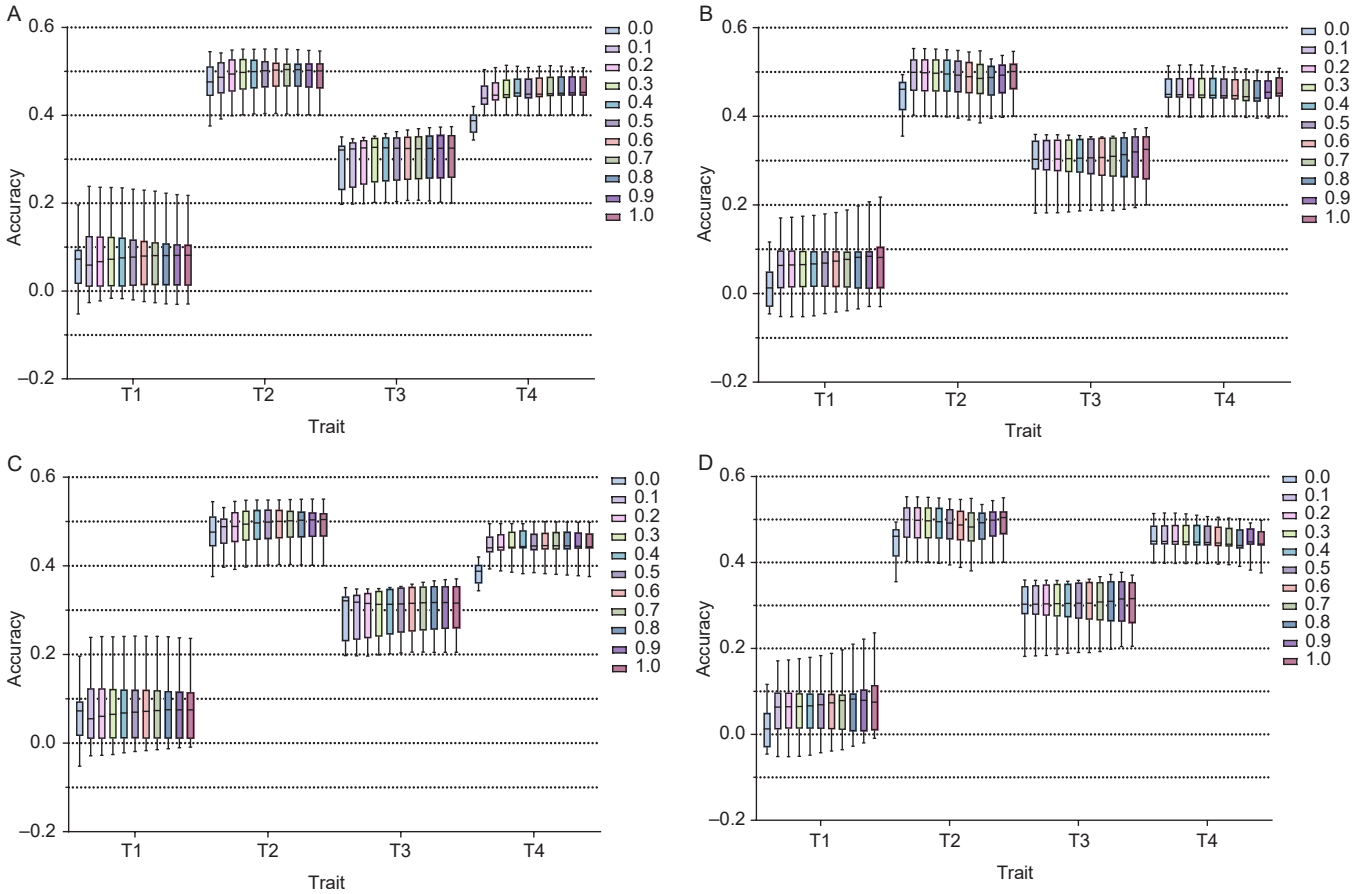


Fig. 1 Comparison of prediction accuracy of SVR_GS (A), SVR_GP (B), SVR_LS (C), and SVR_LP (D) with weight changes in pig dataset. For each boxplot, the middle line represents the median value, the bottom and top are the minimum and maximum, and the lower and upper of each box represent the first quartile and the third quartile value.

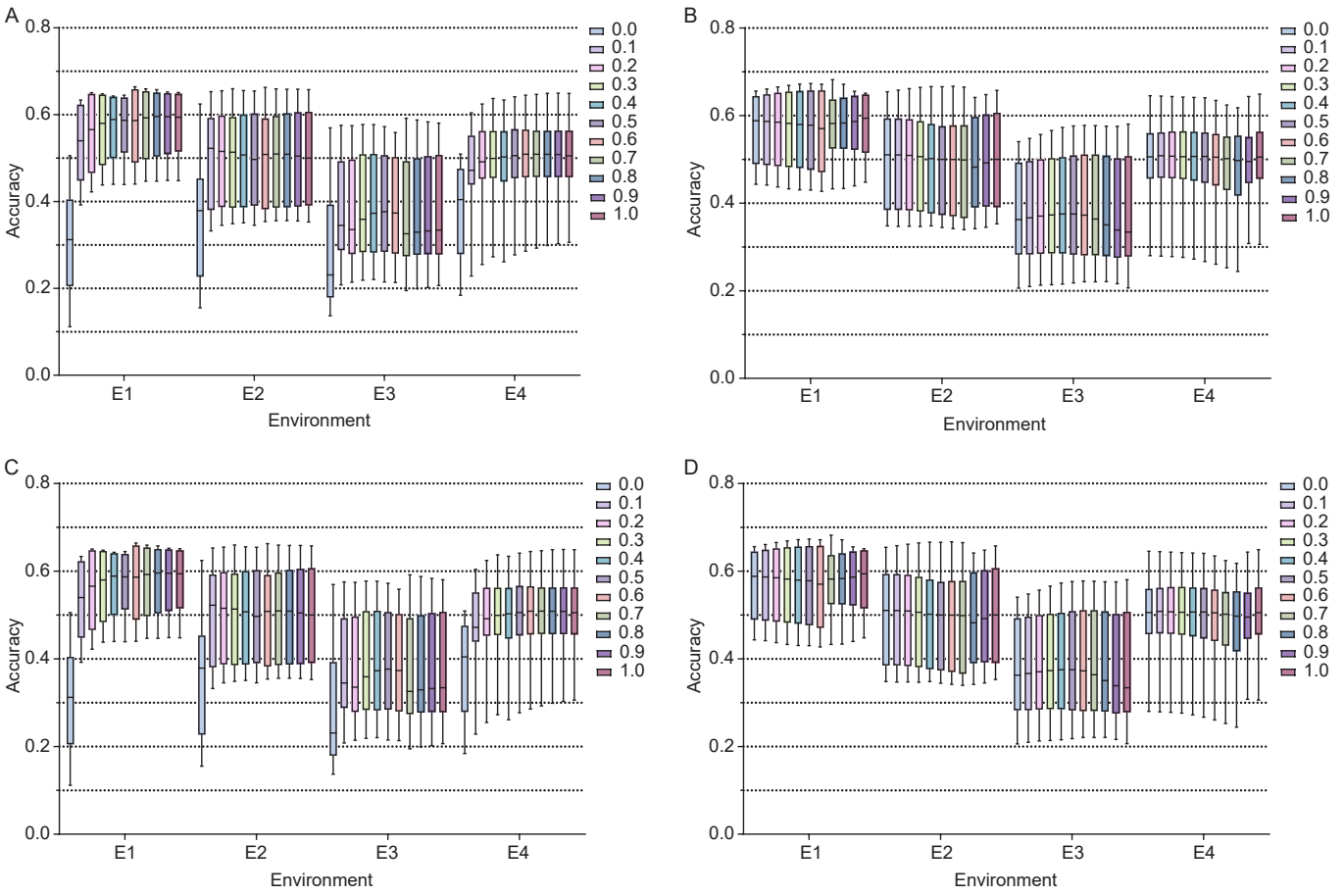


Fig. 2 Comparison of prediction accuracy of SVR_GS (A), SVR_GP (B), SVR_LS (C), and SVR_LP (D) with weight changes in wheat dataset.

Table 1 The weight values of four mixed kernel function models of SVR when achieving the highest accuracy in two datasets[1)]

Hyperparameters	Methods	T1	T2	T3	T4	E1	E2	E3	E4
Tuning	SVR_GS	0.4	0.5	0.9	0.9	0.9	0.8	0.4	0.9
	SVR_GP	0.9	0.1	0.9	0.1	0.9	0.1	0.6	0.2
	SVR_LS	0.5	0.8	0.9	0.4	0.9	0.8	0.4	0.9
	SVR_LP	0.9	0.1	0.9	0.1	0.9	0.1	0.6	0.2
Default	SVR_GS	0.1	0.5	0.9	0.9	0.9	0.9	0.9	0.9
	SVR_GP	0.9	0.9	0.9	0.1	0.1	0.1	0.6	0.2
	SVR_LS	0.1	0.8	0.9	0.9	0.9	0.9	0.9	0.9
	SVR_LP	0.9	0.9	0.9	0.1	0.1	0.1	0.6	0.2

[1)] T1, T2, T3 and T4 represents the four traits in the pig dataset, E1, E2, E3 and E4 represents the four environments in the wheat dataset.

while T4 tends to decrease. In the wheat dataset under default parameters, when mixed the Gaussian or Laplacian local kernels with the Sigmoid kernel function, prediction accuracy across all four environmental conditions generally displays a gradual increase. Conversely, when mixed with the Polynomial kernel function, prediction accuracy in all four environments tends to decreased. The results under optimal parameters do not have significantly difference with those under default parameters.

3.2. Comparison of prediction accuracy of different methods

In our research, the prediction performance of GBLUP, BayesB, KcRR, RF, and SVR in pig and wheat datasets were assessed. In the algorithm of SVR, six single-kernel SVR (SVR_Lin, SVR_G, SVR_P, SVR_S, SVR_C, SVR_L) and four mixed-kernel SVR (SVR_GS, SVR_GP, SVR_LS and SVR_LP) were performed. Fig. 3 shows the prediction performance of different methods in the two datasets. Because the prediction results of SVR_GS and SVR_LS, as well as SVR_GP and SVR_LP, are the same in the wheat dataset, we only compare the results of SVR_GS and SVR_GP to reduce the confounding of the data. When we solely concentrate on the performance of various kernel functions in SVR, we noted that the prediction results for SVR_L and SVR_G are the same in the wheat dataset. In the pig dataset, the prediction results

for these two kernel functions are similar, with the SVR_L outperforming the SVR_G in T1 and T2. In two datasets, for single kernel function, the SVR_G demonstrate strong performance, exhibiting the highest prediction accuracy among most traits, while the SVR_Lin consistently performs the least favorably. SVR_P generally outperform SVR_S, ranking second only to SVR_G. In the pig dataset, the SVR_C surpasses the SVR_P in T1 and T2, while in the wheat dataset, the SVR_P outperforms the SVR_C.

Compared to the two traditional parameter methods (GBLUP and BayesB), machine learning algorithms display remarkable performance. There was no significant difference between GBLUP and BayesB in both datasets. In the pig dataset, GBLUP and BayesB achieved the highest predictive accuracy in T2 and T4. Among them, the predictive accuracy of the six newly introduced kernel functions of SVR for T1 surpasses that of GBLUP and BayesB. Notably, the SVR_GS demonstrates the highest prediction accuracy among all models. Compared to GBLUP, the predictive accuracies of SVR_C, SVR_L, SVR_GS, SVR_GP, SVR_LS, and SVR_LP, were improved by 0.2, 9.4, 10, 2.5, 9.6, and 5.1%, respectively. Moreover, the prediction accuracy of SVR_GS is 69% higher than that of random forests. The RF algorithm exhibits the lowest prediction accuracy for T1 and the highest for T3. The prediction accuracy of the KcRR is lower than that of SVR for T1 but higher for T3. Exception of E3, the predictive accuracy of the six new kernel functions introduced by SVR in this study generally surpasses that of GBLUP and BayesB. For instance, in E1, SVR_L, SVR_C, SVR_GS, and SVR_GP achieve predictive accuracies that are 13.5, 3.8, 13.3, and 13.3% higher than GBLUP, respectively. In E4, SVR_L, SVR_GS, and SVR_GP achieve predictive

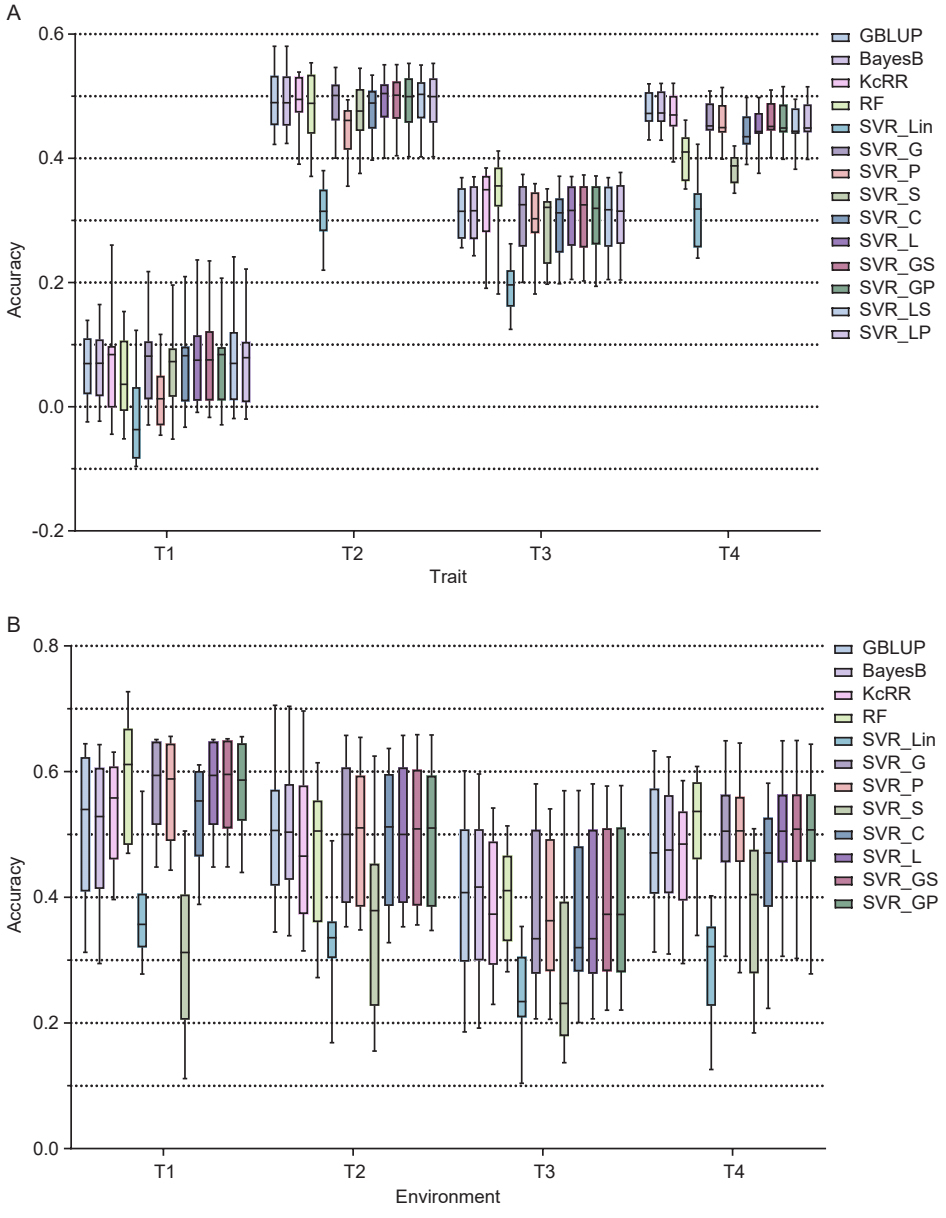


Fig. 3 Comparison of prediction accuracy of GBLUP, BayesB, KcRR, RF, SVR_Lin, SVR_G, SVR_P, SVR_S, SVR_C, SVR_L, SVR_GS, SVR_GP, SVR_LS and SVR_LP in pig (A) and wheat (B) datasets.

accuracies that are 5.9, 5.9, and 5.1% higher than BayesB, respectively. The prediction accuracy of KcRR is lower than that of the mixed kernel functions with SVR. In both datasets, the predictive accuracy of KcRR closely aligns with SVR_C, with KcRR generally outperforming SVR_C.

On the other hand, mean squared error (MSE) and mean absolute error (MAE) are used to evaluate the performance of different methods. As presented in Appendix A, in the pig dataset, for T1 and T2, the MSE and MAE of the two parameter models (GBLUP and BayesB) and the three machine learning algorithms do not exhibit significant differences, and the parameter models even yield lower MSE and MAE compared to the machine learning models. However, for T3 and T4, the MSE and MAE for the three machine learning algorithms are significantly lower than those for GBLUP. Among the machine learning algorithms, KcRR consistently achieves the lowest MSE and MAE, while RF demonstrates the worst performance, yielding the highest MSE and MAE values for T1, T2, and T4. As illustrated in Appendix B, across all four environments in wheat dataset, the MSE and MAE values for the machine learning algorithms are significantly lower than those for GBLUP. The results for BayesB and the machine learning algorithms are similar, but slightly higher than machine learning algorithms. RF achieves the lowest MSE and MAE in E1, E3, and E4, KcRR achieves the highest MSE and MAE in all environments. SVR yields MSE and MAE results significantly lower than KcRR and similar to RF. In two datasets, the MSE and MAE of SVR_Lin are notably high, while SVR_G consistently perform well, demonstrating the lowest MSE and MAE among single kernel functions. SVR_L generally exhibit slightly higher MSE and MAE compared to SVR_G, and SVR_C results also slightly surpass those of KcRR. Importantly, mixed kernel functions demonstrate reduction in MSE and MAE compared to all single kernel functions.

3.3. Comparison of computing time

To assess the computational performance of each method, we compared the program running times of BayesB under default parameters and three machine learning algorithms under optimal parameters. The running time of the four SVR mixed kernel function algorithms is the average time under 9 weights, and all the results are shown in Table 2. In the pig dataset, the single kernel function models of SVR (excluding SVR_L) exhibit the shortest computational times among different methods. RF algorithm demonstrates the longest running time, the average time for processing a single trait is approximately 31.4 h. BayesB requires roughly double the time of GBLUP under default parameters and can take more than 24 h when set with parameters 'nIter=50,000' and 'burnIn=5,000'. In the wheat dataset, GBLUP runs the fastest among different methods (except for SVR_L). The running times for each single kernel function model in SVR are slightly higher than those in GBLUP, while BayesB and RF still running the slowest among different methods. Notably, although SVR_L is a single kernel function, its runtime in the pig dataset is approximately ten times to that of other single kernel functions. However, in the wheat dataset, its runtime is the fastest among different methods, taking only about 3 s, which is roughly three times faster than GBLUP. SVR_C runs faster than KcRR in two datasets. Mixed kernel functions exhibit exceptional performance in both datasets, particularly SVR_GS and SVR_GP mixed kernel functions, running approximately three times faster than GBLUP in the pig dataset. When compared to single kernel functions, their computational times are only slightly higher, approximately 100 s more in the pig dataset, and even lower than some single kernel functions in the wheat dataset. Finally, it is worth noting that due to the relatively small size of the wheat dataset, it offers limited reference value in terms of program runtime.

3.4. Comparison of prediction accuracy under hyper-parameter tuning

For two datasets, Appendices A and B presents the predictive accuracy of all methods under optimal parameters, while Appendices C and D showcase their performance under default parameters. The optimal parameters for machine learning algorithms are organized in Appendices E–I. We found that using the optimal parameters could improve prediction accuracy significantly compared that using default parameters. For instance, in the pig dataset, for T1, KcRR improves by 67% compared to using default parameters when

Table 2 Computing time of different methods in two datasets (in seconds)[1)]

Methods	T1	T2	T3	T4	E1	E2	E3	E4
GBLUP	1,345.4	1,290.6	1,730.8	1,546.6	13.2	11.2	10.6	13.4
BayesB	2,224.8	2,326.3	2,498.2	2,520.7	49.3	44.3	53.5	47.7
KcRR	446.1	441.9	486.8	499.4	19.7	17.0	15.8	18.1
RF	49,231.8	79,044.7	179,159.4	144,776.2	359.3	172.4	56.9	452.0
SVR_Lin	316.8	316.8	369.2	377.7	26.3	25.6	27.9	25.9
SVR_G	293.0	285.9	321.9	342.1	13.0	14.5	11.9	13.0
SVR_P	292.7	271.5	314.3	338.7	22.2	23.8	19.7	23.2
SVR_S	288.3	314.9	317.2	323.2	10.8	11.7	9.9	9.3
SVR_C	324.2	327.2	347.3	372.3	15.3	16.6	12.0	12.3
SVR_L	2,991.5	2,926.4	3,683.4	3,862.8	3.5	3.7	3.5	3.4
SVR_GS	429.1	414.5	467.5	490.0	21.3	23.1	19.2	20.9
SVR_GP	429.1	440.9	491.0	491.4	30.3	29.7	29.6	27.9
SVR_LS	3,701.4	3,290.6	4,068.0	4,259.1	44.4	43.4	42.9	44.3
SVR_LP	3,798.8	3,484.1	4,208.2	4,434.4	43.7	42.9	43.8	44.5

[1)] T1, T2, T3 and T4 represents the four traits in the pig dataset, E1, E2, E3 and E4 represents the four environments in the wheat dataset.

using the optimal parameters. Using optimal parameters, SVR_C and SVR_L also present improvements of 33 and 43% of the prediction accuracy compare to using default parameters. SVR_GS and SVR_GP achieved enhancements of 18 and 53% of prediction accuracy, respectively. RF, among the machine learning algorithms, demonstrated the smallest improvement, with an average increase of 4% across the four traits. In the wheat dataset, KcRR experienced 3.4% increase in predictive accuracy when optimal parameters were used in E1, while SVR_G achieved an average improvement of 9% across all four environmental. SVR_GS exhibited an average improvement of 8.3% across the four environments. BayesB also improved its prediction accuracy due to the increased 'nIter' and 'burnIn' parameters, although the effect was not particularly significant. With the improvement of prediction accuracy in the optimal parameters of the model, MSE and MAE have been reduced, indicating that the model has achieved better fitting.

4. Discussion

Currently, there is a growing number of newly proposed models and methods being applied to genome selection, aiming to improve accuracy and accelerate the breeding improvement of animals and plants (Yin *et al.* 2020, 2023; An *et al.* 2021; Wang *et al.* 2023). Machine learning algorithms have demonstrated significant advantages in handling large datasets and complex genomic effects (Gonzalez-Recio and Forni 2011; Ogutu *et al.* 2011; Abdollahi-Arpanahi *et al.* 2020). However, certain studies have shown that the performance of machine learning algorithms is comparable or slightly inferior to that of parametric methods, particularly in cases only involving additive effects (Long *et al.* 2011; Srivastava *et al.* 2021). Therefore, the exploration of new algorithms remains necessary to enhance the performance of machine learning in genome selection.

The support vector machine in the machine learning algorithm shows great advantages because of the diversity of kernel functions and the convenience of processing high-dimensional data, and is widely used in the selection of animal and plant genomes. Liang *et al.* (2021) conducted research on Chinese Simmental beef cattle and found that compared to GBLUP, SVR achieved a 19.3% improvement in carcass weight, 14.5% in live weight, and 4.6% in eye muscle area. González-Camacho *et al.* (2018) reported that SVM with Linear kernel outperformed other machine learning methods and linear models in the genomic prediction of the rust resistance of wheat. Wang *et al.* (2022) found that for the total number of piglets born (TNB) in Chinese Yorkshire pigs, the accuracy of the SVR model was 19.0% higher than that of GBLUP. However, some studies have shown that the performance of the SVM algorithm may not be as high as other methods. Montesinos-López *et al.* (2019) discovered that the SVR model did not outperform MLP and TGBLUP in four out of seven datasets. Ornella *et al.* (2014) compared SVR, RF, reproducible kernel Hilbert space (RKHS), ridge regression, and Bayesian LASSO in different trait-environment combinations for maize and wheat in a wheat rust dataset. They found that RKHS and RF exhibited the best prediction performance. While SVR demonstrates advantages in genome selection, existing research often focuses on the Gaussian kernel function or optimal single kernel function model to represent the SVM algorithm. Zhao *et al.* (2020) studied four different SVM kernel functions in datasets for pigs and maize, concluding that SVM-RBF achieved higher prediction accuracy in the pig dataset, while SVM-linear exhibited higher prediction accuracy in the maize dataset. These findings suggest that a single kernel function may not perform well across all features or traits, making the selection of the appropriate kernel function challenging.

In this study, the mixed kernel function model of SVM, which has performed well in other fields, was introduced into genome selection, and the selection of weight parameters of different traits and the selection criteria of mixed kernel function suitable for different situations were proposed. First, we compare the performance of six single kernel functions (SVR_Lin, SVR_G, SVR_P, SVR_S, SVR_C, SVR_L) and four mixed kernel functions (SVR_GS, SVR_GP, SVR_LS, SVR_LP). And then, we compare the performance of SVR with two parameter methods (GBLUP, BayesB) and two machine learning algorithms (RF, KcRR), trying to find the kernel function that can be used in most traits or environments. In this research, 5 major categories of algorithms, 14 subcategories of algorithms, and 46 programs were used. It is hoped that it can provide a reference for the research of machine learning algorithms in subsequent genome selection.

4.1. Performance of SVM based on mixed kernel function in genome selection

As we know, both global genetic structures and local region correlations (interactions between specific genes or functional regions) could finally affect the traits. In general, global kernel functions could effectively capture the global genetic similarities, and offer a comprehensive reflection of relationships between different genotypes in the genome. Moreover, local kernel functions are better suited for identifying local associations and exhibit greater sensitivity to capturing specific nonlinear relationships within genomic information. Consequently, the mixture of global kernel functions with local kernel functions may enable the model to accommodate diverse structures within the data, ensuring a nuanced and comprehensive analysis.

However, for mixed functions, the prediction of its changing trend becomes challenge as the weight increases or decreases. This challenge arises due to the potential impact of diverse data structures, sample sizes, and complex genomic information, which can result in different outcomes. But based on our research (Table 1), several conclusions can be drawn regarding the optimization of weight parameters when mixed different kernel functions. When the local kernel function (Gaussian or Laplacian) is mixed with the Sigmoid kernel function, it is often advisable to initially select a weight parameter of 0.9. Specifically, for traits with high heritability (above 0.4), a weight of 0.9 is the primary choice. For traits with low to moderate heritability (below 0.4), a weight selection range of 0.1–0.5 is recommended. When the two

local kernel functions are mixed with the Polynomial kernel function, the optimal weight parameter selection is 0.1. In this case, for traits with high heritability, 0.1 is the preferred option. For traits with low to medium heritability, a choice of either 0.1 or 0.9 can be considered. Furthermore, we observed that the optimal weights for mixed kernel functions do not significantly differ between default parameters and optimal parameters. This indicates that even without parameter optimization, mixed kernel functions consistently perform well and robustly with default parameters. In most cases, the performance of mixed kernel function models surpasses that of their constituent single kernel functions. However, there are instances where the performance of the mixed kernel function falls short of its mother kernel functions. In such situations, even though the performance may be inferior to one of the mother kernel functions, the mixed kernel function model tends to approach the performance of the better single kernel function. From another perspective, this can be viewed as a significant enhancement of the underperforming kernel function.

4.2. Performance of different models in genomic selection

Pig and wheat datasets have been widely used in previous studies (Crossa *et al.* 2014; Zhao *et al.* 2020; An *et al.* 2021; Wang *et al.* 2023). Our results demonstrate that, in the majority of traits, machine learning algorithms outperform traditional parameter models in terms of prediction accuracy. In the pig dataset, the model experienced negative prediction accuracy in T1. T1 is a low heritability trait, which is relatively less more affected by none-genetic factors and has less variability in the dataset. When the machine learning model cannot learn effective information from genotype files to fit the phenotype, negative values may appear, which has also appeared in previous research (Wang *et al.* 2023). When we focus on SVR, the Linear kernel often exhibits poor performance, which may be attributed to the fact that linear models often cannot better capture complex effects, such as epistasis, dominance, or genotype-by-environment interactions. The Gaussian kernel function often performs well, and this could be attributed to its ability to map data effectively to a higher-dimensional space, enabling it to capture complex relationships within the data. As a result, it can better fit functions and yield improved results. In our research, we observed that the Gaussian and Laplacian kernel functions often produce similar results, with the Gaussian kernel function exhibiting shorter calculation times. This can be explained by the fact that both functions belong to radial basis kernel functions, but the Gaussian kernel function determines similarity through Euclidean distance calculation, whereas the Laplacian kernel function uses Manhattan distance. In the process of calculating the difference between vectors, the weights of the differences in each dimension are different, and the Gaussian kernel function often calculates faster due to the Euclidean distance calculation. But there are also studies showing that for high-dimensional data, Manhattan distance may be better than Euclidean distance (Aggarwal *et al.* 2001). The Cosine kernel utilizes cosine similarity, which measures the difference between two individuals based on the cosine value of the angle between their vectors in vector space (An *et al.* 2021). The Cosine kernel disregards the distance between samples, but rather treats them as directions in feature space. Therefore, the Gaussian kernel focuses more on the straight-line distance between samples, the Laplacian kernel focuses more on the sum of distances between samples in all directions, and the cosine kernel focuses more on the angle between samples. In both datasets, KcRR demonstrates relatively strong performance. However, the performance of the Cosine kernel function within SVR is not notably outstanding with inferior predictive accuracy to that of KcRR. This difference in performance may be attributed to the relatively more complex implementation logic of SVR compared to KRR. Nevertheless, SVR_C exhibits a faster computational speed than KcRR. BayesB demonstrated excellent accuracy in pig dataset, indicating that the prior assumption of BayesB is more suitable for fitting the genetic structure of quantitative traits (Clark *et al.* 2011; Sun *et al.* 2011). However, it is worth noting that the Bayesian method tends to lose its advantages as the number of QTLs increases (Daetwyler *et al.* 2010; Gonzalez-Recio and Forni 2011). Furthermore, the implementation process of the Bayesian method is relatively complex, leading to lower calculation efficiency and longer computation times. RF showed high accuracy in the wheat dataset, this may be related to the principle of its integration of many weak decision trees. Secondly, its prediction accuracy is mainly affected by the number of decision trees and the maximum depth. Therefore, this may because that we have given the random forest a very wide parameter range so that the random forest can find the optimal parameters. In the overall comparison of two datasets, machine learning algorithms demonstrate lower MSE and MAE compared to GBLUP. This may be attributed to the machine learning algorithms' superior ability to capture nonlinear relationships between genotypes and phenotypes. In contrast, GBLUP relies on a linear model, which might exhibit relatively poorer performance when dealing with intricate nonlinear relationships. This observation is further supported by the higher MSE and MAE obtained by the SVR_Lin. Although BayesB allows for nonlinear relationships, it may still have limitations in handling complex nonlinear patterns. Therefore, machine learning algorithms demonstrate better fitness to the data, especially when dealing with high-dimensional data and intricate data structures. The mixed kernel function could enhance the predictive accuracy of its mother kernels while reducing the predicted MSE and MAE.

4.3. Analysis of equal prediction results in kernel functions of SVR for wheat dataset

In our study, we interestingly find that in the wheat dataset, the prediction result of the SVR_GS was equal to the SVR_LS, and the SVR_GP was equivalent to the SVR_LP. In order to further clarify the reason, we fixed a single parameter for the SVR_G and SVR_L, and observed different prediction results, confirming that the two kernel functions are indeed distinct models. We also ran the four mixed kernel functions

on the maize dataset (Crossa *et al.* 2010) and found that they produced different prediction results. Therefore, we inferred the following possible reasons for this phenomenon: (1) Similarity of data features. The feature distribution and the relationship between features in the wheat data may be similar for the two mixture kernel functions. In other words, the two kernel functions are similar in mapping data to the feature space, which means that whether Gaussian kernel function and Polynomial kernel function are mixed, or Laplacian kernel function and Polynomial kernel function are mixed, both mixed kernel functions capture the same data patterns and relationships. (2) Similarity of parameter settings. The two mixed kernel functions use similar parameter settings. Similarity in the choice of weights, and hyperparameters of the kernel may cause two mixed kernels to behave identically, resulting in the same accuracy results. (3) Local optima of the training algorithm. Training kernel functions typically involves optimization problems. In some cases, the training algorithm may converge to a local optimum, leading to the same results regardless of the mixed kernel used. However, while the two mixed kernel functions may yield the same accuracy results, they differ in other aspects such as model complexity and the shape of the decision boundary. For instance, the Gaussian kernel function exhibits continuous and smooth variations in the original feature space, allowing it to capture features at different scales. On the other hand, the Laplacian kernel function demonstrates discontinuous and piecewise linear variations in the original feature space, tending to capture features near boundaries or decision surfaces.

4.4. Optimizing of machine learning hyperparameters

We observed that the default hyperparameters often yield suboptimal performance in the parameter tuning process of machine learning algorithms, which underscores the critical importance of hyperparameter tuning for model optimization. This result aligns with previous studies (Alves *et al.* 2021; Wang *et al.* 2022). There are significant limitations to grid search for optimal parameters, we may need to spend a lot of time exploring the optimal range of parameters. Although some studies have provided approximate parameter selection criteria (Cherkassky and Ma 2004), more often people rely on experience to determine the value of parameters. As the number of parameters and values increase, the computational resources and time required grow exponentially. For instance, in the case of mixed kernel functions, each single kernel function entails two parameters, and mixing two kernel functions results in a total of four parameters. If each parameter is tested with 10 values, the grid search process would involve running the program 10^5 times. This poses a significant challenge in terms of computational resources and time constraints. Moreover, there is no guarantee that the optimal parameters will be discovered within the predefined grid. Nevertheless, it is undeniable that tuning the parameters can lead to substantial improvements in the performance of machine learning methods compared to using default hyperparameters. Therefore, we can explore the application of intelligent optimization algorithms as alternatives to grid search, gradually replacing the traditional approach and enabling more efficient identification of optimal parameter configurations.

5. Conclusion

In this study, we compared the prediction accuracy, MSE and MAE results of 10 kinds of kernel functions of SVR with two parameter methods (GBLUP, BayesB) and two machine learning algorithms (RF, KcRR) in the prediction of genome breeding value. We found that the mixed kernel function model has both speed and accuracy advantages over GBLUP, BayesB, and has accuracy competitiveness to single kernel function in SVR of genome selection, with the SVR_GS mixed function shows excellent performance. We also found parameter tuning is critical for enhancing the performance of machine learning algorithms. When mix the local kernel functions (Gaussian or Laplacian) to the Sigmoid kernel function, the weight parameter of 0.9 is the primary choice for traits with high heritability (above 0.4) and the weight parameter of 0.1–0.5 is the primary choice for traits with low to moderate heritability (below 0.4). When mix the two local kernel functions with the Polynomial kernel function, the weight parameter of 0.1 is the primary choice for traits with high heritability and the weight parameter of 0.1 or 0.9 is the primary choice for traits with low to moderate heritability. Future research efforts could involve the utilization of intelligent optimization algorithms to further improve the performance. It is also important to note that different algorithms yield varied results depending on the dataset, and the specific circumstances should be taken into consideration.

Acknowledgements

This work was supported by the China Agriculture Research System of MOF and MARA, the National Natural Science Foundation of China (31872337 and 31501919), and the Agricultural Science and Technology Innovation Project, China (ASTIP-IAS02).

Declaration of competing interest

The authors declare that they have no conflict of interest.

Declaration of generative AI and AI-assisted technologies in the writing process

The authors declare that they did not use AI in the preparation and writing of this manuscript.

Ethical statements

This study does not require ethical approval, and the datasets used for analysis is publicly available.

Data availability

The pig dataset analyzed during the current study are available in the g3journal repository (http://www.g3journal.org/lookup/suppl/doi:10.1534/g3.111.001453/-/DC1). The wheat dataset analyzed during the current study are available in the genetics journal repository (https://academic.oup.com/

genetics/article/186/2/713/6063582).

Appendices associated with this paper are available at https://doi.org/10.1016/j.jia.2024.03.083

References

Abdollahi-Arpanahi R, Gianola D, Penagaricano F. 2020. Deep learning versus parametric and ensemble methods for genomic prediction of complex phenotypes. *Genetics Selection Evolution*, **52**, 12.

Aggarwal C C, Hinneburg A, Keim D A. 2001. On the surprising behavior of distance metrics in high dimensional space. In: Van den Bussche J, Vianu V, eds., *Database Theory-ICDT 2001*. Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 420–434.

Alves A A C, Espigolan R, Bresolin T, Costa R M, Fernandes Júnior G A, Ventura R V, Carvalheiro R, Albuquerque L G. 2021. Genome-enabled prediction of reproductive traits in Nellore cattle using parametric models and machine learning methods. *Animal Genetics*, **52**, 32–46.

An B, Liang M, Chang T, Duan X, Du L, Xu L, Zhang L, Gao X, Li J, Gao H. 2021. KCRR: A nonlinear machine learning with a modified genomic similarity matrix improved the genomic prediction efficiency. *Briefings in Bioinformatics*, **22**, 6.

Aruna S, Rajagopalan D S P. 2013. A novel SVM based CSSFFS feature selection algorithm for detecting breast cancer. *International Journal of Computer Applications*, **31**, 8.

Bowler A L, Pound M P, Watson N J. 2022. A review of ultrasonic sensing and machine learning methods to monitor industrial processes. *Ultrasonics*, **124**, 106776.

Breiman L. 2001. Random forests. *Machine Learning*, **45**, 5–32.

Byvatov E, Schneider G. 2003. Support vector machine applications in bioinformatics. *Applied Bioinformatics*, **2**, 67–77.

Cherkassky V, Ma Y. 2004. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Netw*, **17**, 113–126.

Clark S A, Hickey J M, van der Werf J H. 2011. Different models of genetic variation and their effect on genomic evaluation. *Genetics Selection Evolution*, **43**, 18.

Cleveland M A, Hickey J M, Forni S. 2012. A common dataset for genomic analysis of livestock populations. *Genes*, *Genomes*, *Genetics*, **2**, 429–435.

Cortes C, Vapnik V. 1995. Support-vector networks. *Machine Learning*, **20**, 273–297.

Crossa J, Campos G d l, Pérez P, Gianola D, Burgueño J, Araus J L, Makumbi D, Singh R P, Dreisigacker S, Yan J, Arief V, Banziger M, Braun H J. 2010. Prediction of genetic values of quantitative traits in plant breeding using pedigree and molecular markers. *Genetics*, **186**, 713–724.

Crossa J, Pérez P, Hickey J, Burgueño J, Ornella L, Cerón-Rojas J, Zhang X, Dreisigacker S, Babu R, Li Y, Bonnett D, Mathews K. 2014. Genomic prediction in CIMMYT maize and wheat breeding programs. *Heredity*, **112**, 48–60.

Crossa J, Pérez-Rodríguez P, Cuevas J, Montesinos-López O, Jarquín D, de Los Campos G, Burgueño J, González-Camacho J M, Pérez-Elizalde S, Beyene Y, Dreisigacker S, Singh R, Zhang X, Gowda M, Roorkiwal M, Rutkoski J, Varshney R K. 2017. Genomic selection in plant breeding: Methods, models, and perspectives. *Trends in Plant Science*, **22**, 961–975.

Daetwyler H D, Pong-Wong R, Villanueva B, Woolliams J A. 2010. The impact of genetic architecture on genome-wide evaluation methods. *Genetics*, **185**, 1021–1031.

Gao X, Jia B, Li G, Ma X. 2022. Calorific value forecasting of coal gangue with hybrid kernel function-support vector regression and genetic algorithm. *Energies*, **15**, 18.

García-Ruiz A, Cole J B, VanRaden P M, Wiggans G R, Ruiz-López F J, Van Tassell C P. 2016. Changes in genetic selection differentials and generation intervals in US Holstein dairy cattle as a result of genomic selection. *Proceedings of the National Academy of Sciences of the United States of America*, **113**, E3995–E4004.

Gianola D, Campos G, Gonzalez-Recio O, Long N, Okut H, Rosa G, Weigel K, Wu X L. 2018. Statistical learning methods for genome-based analysis of quantitative traits. *Conference*: *Machine Learning*.

Gianola D, Okut H, Weigel K A, Rosa G J. 2011. Predicting complex quantitative traits with Bayesian neural networks: A case study with Jersey cows and wheat. *BMC Genetics*, **12**, 87.

Goddard M, Hayes B. 2008. Genomic selection. *Journal of Animal Breeding and Genetics=Zeitschrift für Tierzüchtung und Züchtungsbiologie*, **124**, 323–330.

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

Gonzalez-Recio O, Forni S. 2011. Genome-wide prediction of discrete traits using Bayesian regressions and machine learning. *Genetics Selection Evolution*, **43**, 7.

Gonzalez-Recio O, Rosa G, Gianola D. 2014. Machine learning methods and predictive ability metrics for genome-wide prediction of complex traits. *Livestock Science*, **166**, 217–231.

Habier D, Fernando R L, Kizilkaya K, Garrick D J. 2011. Extension of the bayesian alphabet for genomic selection. *BMC Bioinformatics*, **12**, 186.

Handelman G S, Kok H K, Chandra R V, Razavi A H, Lee M J, Asadi H. 2018. eDoctor: Machine learning and the future of medicine. *Journal of Internal Medicine*, **284**, 603–619.

Hansen K B, Borch C. 2021. The absorption and multiplication of uncertainty in machine-learning-driven finance. *British Journal of Sociology*, **72**, 1015–1029.

Hayes B J, Bowman P J, Chamberlain A J, Goddard M E. 2009. Invited review: Genomic selection in dairy cattle: Progress and challenges. *Journal of Dairy Science*, **92**, 433–443.

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

Huang S, Cai N, Pacheco P P, Narrandes S, Wang Y, Xu W. 2018. Applications of support vector machine (SVM) learning in cancer genomics. *Cancer Genomics Proteomics*, **15**, 41–51.

Ibrikci T, Ustun D, Kaya I E. 2012. Diagnosis of several diseases by using combined kernels with Support Vector Machine. *Journal of Medical Systems*, **36**, 1831–1840.

Kung S Y. 2014. *Kernel Methods and Machine Learning*. Cambridge University Press, Cambridge.

LeCun Y, Bengio Y, Hinton G. 2015. Deep learning. *Nature*, **521**, 436–444.

Legarra A, Aguilar I, Misztal I. 2009. A relationship matrix including full pedigree and genomic information. *Journal of Dairy Science*, **92**, 4656–4663.

Liang M, Miao J, Wang X, Chang T, An B, Duan X, Xu L, Gao X, Zhang L, Li J, Gao H. 2021. Application of ensemble learning to genomic selection in Chinese simmental beef cattle. *Journal of Animal Breeding and Genetics*, **138**, 291–299.

Lillehammer M, Meuwissen T H, Sonesson A K. 2013. Genomic selection for two traits in a maternal pig breeding scheme. *Journal of Animal Science*, **91**, 3079–3087.

Long N, Gianola D, Rosa G J, Weigel K A. 2011. Application of support vector regression to genome-assisted prediction of quantitative traits. *Theoretical and Applied Genetics*, **123**, 1065–1074.

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

Montesinos-López O A, Martín-Vallejo J, Crossa J, Gianola D, Hernández-Suárez C M, Montesinos-López A, Juliana P, Singh R. 2019. A benchmarking between deep learning, support vector machine and bayesian threshold best linear unbiased prediction for predicting ordinal traits in plant breeding. *Genes*, *Genomes*, *Genetics*, **9**, 601–618.

Ogutu J O, Piepho H P, Schulz-Streeck T. 2011. A comparison of random forests, boosting and support vector machines for genomic selection. *BMC Proceedings*, **5** (Suppl 3), S11.

Ornella L, Pérez P, Tapia E, González-Camacho J M, Burgueño J, Zhang X, Singh S, Vicente F S, Bonnett D, Dreisigacker S, Singh R, Long N, Crossa J. 2014. Genomic-enabled prediction with classification algorithms. *Heredity* (Edinb), **112**, 616–626.

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay É. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

Schaeffer L R. 2006. Strategy for applying genome-wide selection

in dairy cattle. *Journal of Animal Breeding and Genetics*, **123**, 218–223.
Shrestha D L, Solomatine D P. 2006. Experiments with AdaBoost.RT, an improved boosting scheme for regression. *Neural Computation*, **18**, 1678–1710.
Smits G F, Jordaan E M. 2002. Improved SVM regression using mixtures of kernels. In: *Proceedings of the 2002 International Joint Conference on Neural Networks*. Institute of Electrical and Electronics Engineers, vol. **3**. pp. 2785–2790.
Srivastava S, Lopez B I, Kumar H, Jang M, Chai H H, Park W, Park J E, Lim D. 2021. Prediction of Hanwoo cattle phenotypes from genotypes using machine learning methods. *Animals* (Basel), **11**, 7.
Sun X, Habier D, Fernando R L, Garrick D J, Dekkers J C. 2011. Genomic breeding value prediction and QTL mapping of QTLMAS2010 data using Bayesian Methods. *BMC Proceedings*, **5** (Suppl 3), S13.
Tian Z, Li S, Wang Y, Wang X. 2017. Wind power prediction method based on hybrid kernel function support vector machine. *Wind Engineering*, **42**, 252–264.
Tibshirani R. 2011. Regression shrinkage and selection *via* the lasso: A Retrospective. *Journal of the Royal Statistical Society Series B* (Statistical Methodology), **73**, 273–282.
VanRaden P M. 2008. Efficient methods to compute genomic predictions. *Journal of Dairy Science*, **91**, 4414–4423.
Varona L, Legarra A, Toro M A, Vitezica Z G. 2018. Non-additive effects in genomic selection. *Frontiers in Genetics*, **9**, 78.
Wang K, Abid M A, Rasheed A, Crossa J, Hearne S, Li H. 2023. DNNGP, a deep neural network-based method for genomic prediction using multi-omics data in plants. *Molecular Plant*, **16**, 279–293.
Wang X, Shi S, Wang G, Luo W, Wei X, Qiu A, Luo F, Ding X. 2022. Using machine learning to improve the accuracy of genomic prediction of reproduction traits in pigs. *Journal of Animal Science and Biotechnology*, **13**, 60.
Weiskittel T M, Correia C, Yu G T, Ung C Y, Kaufmann S H, Billadeau D D, Li H. 2021. The trifecta of single-cell, systems-biology, and machine-learning approaches. *Genes* (Basel), **12**, 7.
Whittaker J C, Thompson R, Denham M C. 2000. Marker-assisted selection using ridge regression. *Genetics Research*, **75**, 249–252.
Wolc A, Stricker C, Arango J, Settar P, Fulton J E, O'Sullivan N P, Preisinger R, Habier D, Fernando R, Garrick D J, Lamont S J, Dekkers J C. 2011. Breeding value prediction for production traits in layer chickens using pedigree or genomic relationships in a reduced animal model. *Genetics Selection Evolution*, **43**, 5.
Yi N, Xu S. 2008. Bayesian LASSO for quantitative trait loci mapping. *Genetics*, **179**, 1045–1055.
Yin L, Zhang H, Tang Z, Yin D, Fu Y, Yuan X, Li X, Liu X, Zhao S. 2023. HIBLUP: An integration of statistical models on the BLUP framework for efficient genetic evaluation using big genomic data. *Nucleic Acids Research*, **51**, 3501–3512.
Yin L, Zhang H, Zhou X, Yuan X, Zhao S, Li X, Liu X. 2020. KAML: Improving genomic prediction accuracy of complex traits using machine learning determined parameters. *Genome Biology*, **21**, 146.
Zhao D, Liu H, Zheng Y, He Y, Lu D, Lyu C. 2019. Whale optimized mixed kernel function of support vector machine for colorectal cancer diagnosis. *Journal of Biomedical Informatics*, **92**, 103124.
Zhao W, Lai X, Liu D, Zhang Z, Ma P, Wang Q, Zhang Z, Pan Y. 2020. Applications of support vector machine in genomic prediction in pig and maize populations. *Frontiers in Genetics*, **11**, 598318.

Executive Editor-in-Chief Kui Li
Managing Editor Han Zhang