Aus dem Institut für Pflanzenbau und Pflanzenzüchtung II
der Justus-Liebig-Universität Gießen
Professur für Biometrie und Populationsgenetik
Prof. Dr. Matthias Frisch

Predictive Modelling with Machine Learning in Plant Breeding

Dissertation zur Erlangung des akademischen Grades eines

Doktors der Agrarwissenschaften

- Dr. agr. -

im Fachbereich
Agrarwissenschaften, Ökotrophologie und Umweltmanagement der
Justus-Liebig-Universität Gießen

vorgelegt von

Philipp Georg Heilmann
aus Frankenthal (Pfalz)

Gießen, 2025

Contents

[1]P. G. Heilmann, M. Frisch, A. Abbadi, T. Kox, and E. Herzog (2023) Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP. *Frontiers in Plant Science* **14**:262.

[2]P. G. Heilmann, Y. F. Difabachew, M. Frisch, A. L. Moritz, A. Stahl, B. Wittkop, R. J. Snowdon, M. Koch, M. Kirchhoff, L. Cselényi, M. Wolf, J. Förster, and C. Zenke-Philippi (2024) Machine learning for prediction of resistance scores in wheat (*Triticum aestivum* L.). *Plant Breeding* **144**:192–205.

[3]P. G. Heilmann, E. Grosch, M. Frisch, M. Herrmann, S. Beuch, V. Kurra, M. Mascher, R. Avni, K. Oldach, I. Röhrs, A. Hanemann, R. R. Mehta, C. Reinbrecht, A. Serfling, A. Stahl, M. Stucke, A. Abbadi, T. Kox, and C. Zenke-Philippi. (2025) Haplotype-based autoencoders can reduce the dataset dimension and estimate haplotype block effects in different crop species. *BMC Bioinformatics* **26**(1):289

Abbreviations

BLUP	best linear unbiased prediction
DH	doubled haploid
DNA	deoxyribonucleic acid
GBLUP	genomic BLUP
GCA	general combining ability
GEBV	genomic estimated breeding value
LD	linkage disequilibrium
LGEBV	local genomic estimated breeding value
MAS	marker-assisted selection
ML	machine learning
QTL	quantitative trait locus
RFLP	restriction fragment length polymorphisms
RKHS	reproducing kernel Hilbert spaces
RR-BLUP	ridge regression BLUP
SCA	specific combining ability
SNP	single nucleotide polymorphism
SSR	simple sequence repeats

Chapter 1

General introduction

Breeding for improved crops

By 2050, the global population is projected to possibly exceed 10 billion people, creating increasing demands for food supplies and plant-based industry materials (FAO 2017, 2022; Tilman et al. 2011). The difficulty of this challenge is increased by climate change, which threatens to offset gains in agricultural productivity and complicate efforts to improve crop yields (FAO 2017, 2022). Beyond transforming food production systems, one crucial strategy is to breed high-yielding, more resilient crops. In particular, sustainable agricultural development calls for reducing reliance on agrochemicals that harm the environment (Foley et al. 2011; Garnett et al. 2013). To meet future needs, agricultural production must increase, but with fewer inputs and under increasingly variable climate conditions. One promising route for accelerating the improvement of crops is the integration of genomics, data science, and statistical modeling into plant breeding programs. In particular, genomic prediction and machine learning (ML) offer the potential to speed up breeding progress and increase selection accuracy, leading to improved crop varieties.

Plant breeding has played an important part in the agricultural advancement since the early 20th century, with particularly significant improvements during the Green Revolution of the 1960s and 1970s (Evenson and Gollin 2003). This era saw dramatic increases in crop productivity, largely due to the development of high-yielding varieties, improved agronomic practices, and the use of chemical inputs. A significant part of these improvements is attributed to structured breeding programs that systematically selected and crossed plants to achieve desired traits. At its core, a plant breeding program operates through iterative cycles of generating genetic

variation, selecting superior individuals, and recombining them to develop improved varieties. There are four main ways in which new varieties are bred (Becker 2011):

Clone breeding: Clonally propagated crops (e.g. potato, grapevine, many fruit trees) are multiplied vegetatively (via tubers, cuttings, etc.) rather than through seed. Initially, selected parents are crossed to create progeny. Due to the high heterozygosity often found in clonally propagated crops, the variation among the offspring is very high. Every individual resulting from this cross is already a potential variety, as clonal propagation preserves their genetic composition. Afterwards, promising offspring are evaluated in larger trials to identify the best individuals with regard to the target traits of the program. Segregation of the offspring after crossing is too large to apply backcrossing schemes. Even if it was applied, many clonally propagagted crops quickly suffer from inbreeding depression or have a high genetic self-incompatibility. This makes introgressing specific qualitative traits very hard and genetic gain through breeding very slow.

Line breeding: In line breeding, the target is to create completely homozygous genotypes. Typically, this way of breeding is applied to self-pollinated crops, such as wheat and soybean. They primarily fertilize themselves, leading to genetic uniformity through inbreeding. In breeding programs, genetic variation is initially introduced through controlled crosses between genetically distinct parents. As the first generation of crosses between homozygous lines is uniform, at least two generations of crosses are required to create genetic diversity and to scan the material for promising genotypes. Then, the segregating progeny are subjected to repeated cycles of selfing and selection over multiple generations to obtain homozygous inbred lines. Similarly, introgression of specific genes can be done by crossing a recurrent parent with a donor, followed by repeated backcrossing with the recurrent parent. For this, only progeny carrying the donor gene of interest are selected in each cycle while the genetic background of the recurrent parent is recovered. A major advancement in creating homozygous lines has been the development of doubled haploid (DH) technology, which enables the rapid production of completely homozygous lines from heterozygous individuals in a single generation. This technique bypasses multiple selfing generations and significantly accelerates the breeding cycle (Forster and Thomas 2005).

Population breeding: The target of population breeding is to improve the average performance of a population rather than selecting specific individuals. Crops in

which population breeding is commonly used include obligate outcrossers like fodder grasses. In mixed-mating plants like rye and faba bean, synthetic varieties are common, created by intercrossing a selected group of elite genotypes Crosses occur naturally within a managed population, rather than being planned between specific parents. The population is then gradually improved by selecting and keeping only superior individuals in the population while removing others. These varieties maintain a high level of genetic diversity, which provides adaptability but can also lead to variation in performance among individuals and across different environments. To ensure that desirable traits are preserved and improved over time, a breeding population must be maintained through recurrent selection and proper isolation to prevent genetic drift and outside contamination.

Hybrid breeding: Hybrid breeding aims to exploit heterosis by crossing two genetically distinct, homozygous inbred lines to produce F_1 progeny that are uniform and outperform their parents in traits such as yield. This strategy has been widely adopted for crops such as maize, sugar beet, rapeseed and rice, and has also been applied to species historically bred as lines, including wheat (Zhao et al. 2015) and barley (Bernhard et al. 2017). The initial step of the breeding process is the development of inbred parental lines through repeated selfing or DH technology. This is followed by field trials to evaluate hybrids based on these parental lines. Field trials allow the estimation of the general combining ability (GCA) of each parent, which is its average performance across multiple crosses. Based on these trials, parents showing the highest GCA are crossed again to obtain the most promising hybrids. From these hybrids, those are selected as experimental varieties that show the highest combination of GCA and specific combining ability (SCA), which is the performance of the specific cross that goes beyond the sum of the parental GCA (heterosis). To increase the heterosis effect systematically, breeders can split germplasm into distinct heterotic pools, optimizing hybrid performance by crossing lines from genetically distant groups (Melchinger and Gumber 1998; Krenzer et al. 2024). Hybrids with strong heterosis are particularly common in maize, where hybrid breeding has historically been easily applicable by removing the tassel of the female parent to prevent self-pollination. However, this approach is not feasible in many other crops, necessitating the implementation of technically challenging and time-consuming biotechnological male-sterility systems.

The variety type and reproductive system significantly affect the structure of a breeding program. For instance, self-pollinated crops allow for rapid fixation of traits

and are more suitable for genomic selection in early generations. In contrast, hybrid breeding requires comprehensive testing of parental lines and their combinations (i.e. GCA and SCA), making the prediction of hybrid performance a critical focus.

Over time, breeding programs have evolved from simple phenotypic selection to incorporate pedigree-based and, more recently, genomic information (Bernardo 1994, 2010; Lee et al. 2015). Today, successful breeding programs require accurate phenotyping, a broad genetic base, and tools for predicting genetic merit. The increasing availability of genomic data has opened new possibilities for accelerating genetic gain while reducing the cost and time required for selection.

The introduction of genetic markers

A transformation of plant breeding began with the availability and usage of genetic markers, providing a way to directly access the genetic variation underlying observable traits. Prior to the use of molecular tools, selection relied exclusively on phenotypic performance and pedigree records.

Marker-assisted selection

Initially, marker-assisted selection (MAS) emerged in the late 1980s and early 1990s, enabling selection based on molecular markers linked to desirable traits rather than solely on phenotypes (Lande and Thompson 1990). Molecular markers are DNA sequences that are polymorphic and heritable and their presence or absence can be used to track the inheritance of genomic regions associated with traits of interest. Early marker systems such as restriction fragment length polymorphisms (RFLPs), followed by simple sequence repeats (SSRs) and later single nucleotide polymorphisms (SNPs), enabled the construction of genetic linkage maps and the identification of quantitative trait loci (QTL) (Davis and DeNise 1998; Collard et al. 2005; Collard and Mackill 2008; Bernardo 2008). The idea is that if the position of a DNA marker on the chromosome is in close proximity to a gene or QTL affecting a trait, then selecting plants carrying the favourable marker allele should indirectly select for the desired trait.

MAS proved to be especially valuable for traits with simple genetic architecture, such as monogenic disease resistance or specific quality traits like dwarfism (Francia

et al. 2005). In many crops, MAS enabled the introgression of resistance alleles into elite germplasm, pyramiding of multiple resistance genes, and the elimination of undesirable linkage drag in backcross programs (Frisch and Melchinger 2001; Hospital 2001; Francia et al. 2005; Peng et al. 2014). The approach was also instrumental in accelerating early selection and reducing the breeding cycle time. For example, a breeder could use markers to screen seedlings for the presence of a disease resistance gene instead of running a full greenhouse trial to observe a possible resistance. The success of MAS demonstrated that molecular markers could be used not only for research but also as operational tools in applied breeding.

However, MAS encounters significant limitations when applied to complex quantitative traits such as yield. Identifying a few major QTL captures only a part of the genetic variance of complex traits. Many QTL with minor effects remained undetected, and those that were detected often had effects that varied across genetic backgrounds or environments (Beavis 1998; Melchinger et al. 1998; Xu 2003; Bernardo 2008). Moreover, MAS relies on tight linkage between the marker and the causal gene. Recombination can break the association over generations, reducing MAS accuracy if the markers are not very close to the actual genes. As a result, MAS has been most successful for traits governed by a few major genes while its use for polygenic traits has been limited. Furthermore, practical limitations in early molecular marker technologies, such as low marker density, high genotyping costs, and limited genome coverage, restricted the applicability of MAS in breeding programs (Young 1999; Collard and Mackill 2008). These shortcomings of MAS motivated a conceptual and methodological shift toward genome-wide approaches, which do not rely on the identification of individual QTL but instead use all marker information simultaneously to predict breeding values.

Genomic prediction

The shift away from MAS and towards genomic prediction was motivated by the limited success of MAS for improvement of complex traits in combination with the long known observation that quantitative genetic models based on Fisher's (1918) infinitesimal model fitted complex traits like yield quite well. The idea is that polygenic/quantitative traits are affected by many (theoretically infinite) genes, spread all across the genome, where most genes only have a small effect. Rather than focusing on individual QTL, researchers proposed incorporating all genome-wide marker data into prediction models, effectively capturing the many small genetic effects underlying quantitative traits. Initially, Bernardo (1994) showed that using markers

to calculate a genetic relationship matrix among individuals could improve the prediction of hybrid performance in maize. Later, Whittaker et al. (2000) suggested using ridge regression on all available markers simultaneously to predict quantitative traits. The concept of genomic prediction and subsequent genomic selection was formalised by Meuwissen et al. (2001).

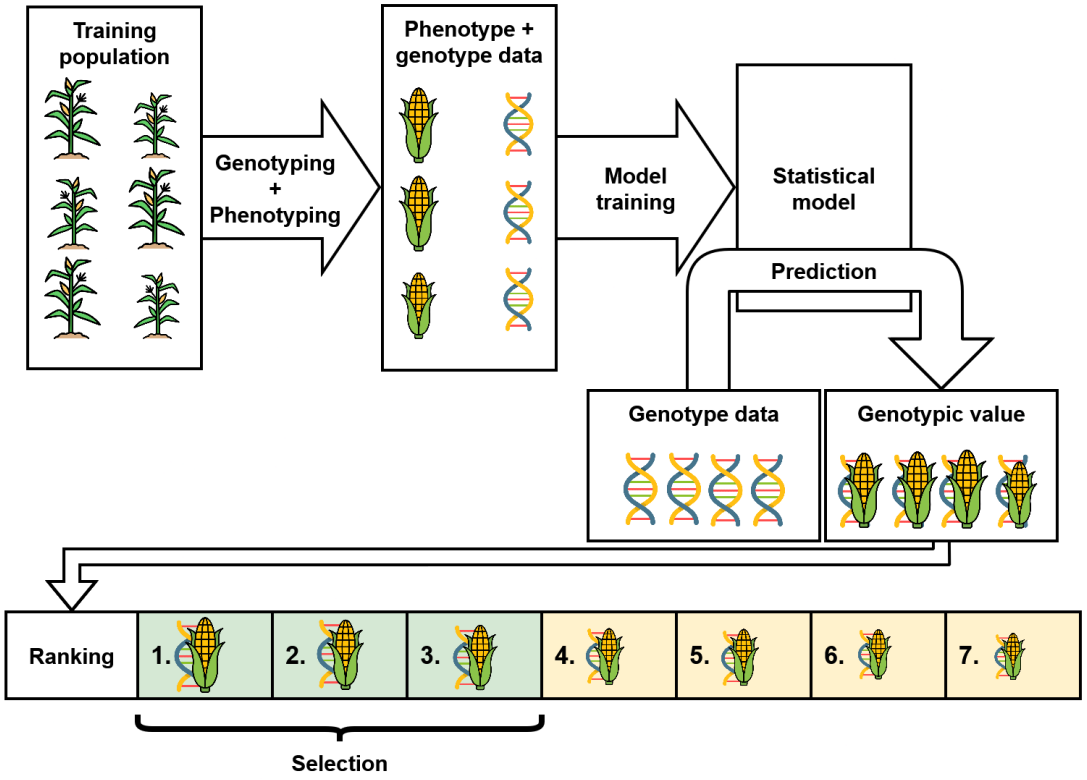

Figure 1.1. Illustration of the genomic prediction process for selection. Some individuals are evaluated in the field and then phenotyped and genotyped. A statistical model is trained using the phenotype and genotype data of the training population. This model is then used to predict the genotypic values of all individuals that have been genotyped. Individuals are then ranked based on their predicted genotypic value and the best are selected.

The typical process of genomic prediction begins with a training population that has been both genotyped and phenotyped for traits of interest (see Figure 1.1). Statistical models are then fitted to capture the relationship between genotypic and phenotypic data. These models can estimate effects for markers or individuals based on genomic relationships, depending on the method used. After training, the model can predict genotypic values for individuals that have only been genotyped.

In the context of inbred line selection, these predictions are often referred to as genomic estimated breeding values (GEBVs) and can be used to rank candidates for advancement. In hybrid breeding, genomic prediction is typically used to estimate GCA, which reflects the average performance of a line in hybrid combinations. This allows breeders to identify promising parental lines with either higher accuracy or by evaluating fewer hybrid crosses in field trials. Better and earlier selection leads to an increased genetic gain through shorter breeding cycles, as it enables earlier selection decisions by relying on genotypic data rather than waiting for phenotypic evaluations (Heffner et al. 2010). While statistical models and computer hardware have improved and genotyping costs for dense markers decreased over the years, the cost of phenotyping did not. This has made genomic prediction much more cost-efficient, leading to a widespread adoption in major plant breeding programs.

To make genomic predictions as accurate as possible, it is common to account for environmental influences and experimental design factors in the phenotypic data. Field trials are typically conducted across multiple environments (locations and/or years) with replications, so that the observed phenotype y can be decomposed into genetic and non-genetic components. A simplified mixed linear model might be:

$$y_{ij} = \mu + G_i + E_j + (G \times E)_{ij} + \epsilon_{ik}$$

where y_{ij} is the observed phenotype, G_i is the genotypic value of genotype i, Ej the environmental effect of environment j, and $(G \times E)_{ij}$ is the interaction effect of genotype i with environment j. The residual error ϵ_{ij} represents random variation in the data not explained by either environment or genotype. Depending on the actual field design, a model typically contains additional factors such as blocks and replications. Through this, it is possible to obtain so-called adjusted entry means. These are the average observed phenotypic values of a genotype without the effects of environments and their interactions with that genotype.

For genomic prediction, the adjusted entry means are then used in either a ridge regression best linear unbiased prediction (RR-BLUP) (Meuwissen et al. 2001) or genomic BLUP (GBLUP) (Bernardo 1994; VanRaden 2008). The key difference between RR-BLUP and GBLUP is in how genomic information is incorporated into the model. RR-BLUP is essentially a linear regression with all markers fitted simultaneously under a common shrinkage (ridge) penalty, assuming each marker contributes equally small variance. The genotypic value of an individual is then calculated as the sum of the effects of its markers. In contrast, GBLUP uses a genomic relationship

matrix derived from marker data to directly estimate genotypic values based on the realized genomic similarities among individuals. It has been shown that GBLUP and RR-BLUP can be mathematically equivalent (Habier et al. 2007; VanRaden 2008; Shen et al. 2013). The dimensions of the genomic relationship matrix are equivalent to the number of genotypes in the data. Since the number of genotypes is typically much smaller than the number of markers, the genomic relationship matrix used in GBLUP has a lower dimensionality than the marker matrix used in RR-BLUP. As a result, GBLUP is often computationally more efficient.

The RR-BLUP and GBLUP models have limitations that have led to the development of a multitude of different approaches used in plant breeding. One major limitation is the assumption of a purely additive genetic architecture. This simplification overlooks non-additive genetic contributions such as dominance and epistasis, which can have an influence on many agronomic traits, especially in hybrid crops. To address this, researchers have extended GBLUP to include additional genomic relationship matrices capturing dominance effects or epistatic interactions (Vitezica et al. 2013). These matrices are constructed analogously to the additive genomic relationship matrix but based on dominance or pairwise interaction terms between markers and then used as additional inputs in the model. Another limitation is the assumption of small marker effects with uniform variance. This homogeneity of shrinkage means they may perform worse if the genetic architecture deviates from the infinitesimal model (e.g. presence of multiple medium- or large-effect QTL). Some methods have been proposed to address this by splitting the uniform shrinkage parameter into marker-specific ridge parameters (Shen et al. 2013; Hofheinz and Frisch 2014).

Instead of deriving different relationship matrices from the marker data, it is also possible to exchange the marker data itself with different types of data, such as transcriptome (Zenke-Philippi et al. 2016, 2017; Knoch et al. 2021) or other omics data (Westhues et al. 2017). Haplotype blocks can also be an alternative to raw SNP data (Cuyabano et al. 2014; Jiang et al. 2018; Difabachew et al. 2023). These are segments of DNA on the chromosomes that tend to get inherited together (Gabriel et al. 2002), and can therefore be seen as a unit from a statistical point of view. Different block-building methods have been proposed, ranging from simple approaches using fixed-size windows based on a set number of SNPs or physical/genetic distance, to LD-based methods (Zhao et al. 2005), or more advanced approaches like HaploBlocker, which identifies subgroup-specific haplotype blocks based on shared allele

sequences (Pook et al. 2019). In theory, forming blocks allows models to capture local epistatic effects (Jiang et al. 2018).

At the same time, more flexible modelling approaches have been developed to overcome GBLUP's linear and additive constraints. Bayesian alphabet models such as BayesA, BayesB, and BayesCπ allow for variable shrinkage and sparsity in marker effects by imposing different prior distributions (Meuwissen et al. 2001; Habier et al. 2011). In theory, this makes them more suitable for traits controlled by a few major loci alongside many minor ones. Kernel methods like Reproducing Kernel Hilbert Space regression (RKHS) aim to model non-linear relationships between genotypes and phenotypes, potentially capturing additive, dominance, and epistatic effects in a unified framework (Gianola and van Kaam 2008).

In summary, the success of genomic prediction depends on three key aspects: the nature of the target trait, the type of input features used and the choice of the statistical model as the link between the inputs features and the target train (Heslot et al. 2012). Together, these elements determine the prediction accuracy, which in turn impacts the rate of genetic gain through more accurate selection.

Introduction to machine learning

The pursuit of higher predictive accuracy, especially for complex traits with non-linear genetic architectures influenced by genotype-by-environment interactions, has motivated the exploration of ML methods in genomic selection. ML, sometimes referred to as 'artificial intelligence', describes a class of algorithms that learn patterns from data without any assumptions about linearity and normality (Bishop 2006). They can handle high-dimensional input data and automatically learn interactions between markers without prior specification of model terms. This is typically done by iterating many times over a set of training data, where each iteration only slightly changes the model parameters along a gradient that reduces the model error.

ML has already transformed several scientific fields. In computer vision, deep convolutional neural networks have enabled breakthroughs in image classification and object detection, enabling applications from medical imaging to autonomous vehicles (Krizhevsky et al. 2012; Esteva et al. 2017; Tan and Le 2019; Spielberg et al. 2019). In natural language processing, transformer architectures have revolutionized machine translation and text understanding (Vaswani et al. 2017; Devlin

et al. 2019; Kumar 2024), which has led to the widespread adoption of language models like ChatGPT in daily life. In structural biology, the deep learning model AlphaFold achieved unprecedented accuracy in protein structure prediction (Jumper et al. 2021). This proven record of the ability to model complex relationships within data, which was previously impossible, has inspired interest in applying ML to plant breeding, particularly for complex traits and high-dimensional genomic data.

Common algorithms in machine learning

Tree-based ensemble models such as random forests (Breiman 2001) and gradient boosting machines (Friedman 2001) build multiple decision trees and aggregate their outputs to capture non-linear relationships and interactions between markers. These models are robust to overfitting and can perform implicit feature selection, making them attractive for noisy genomic datasets. Support vector machines (Cortes and Vapnik 1995; Drucker et al. 1996) find a hyperplane that best separates the data in a transformed feature space using kernel functions to model non-linear relationships. The primary applications of these models in plant breeding have so far been fairly standard genomic prediction approaches based on marker data, sometimes including environmental data (Azodi et al. 2019; Westhues et al. 2021; John et al. 2022; Gabur et al. 2022; Heinrich et al. 2023).

Artificial neural networks consist of layers of interconnected nodes, which resemble systems of linear equations passed through non-linear activation functions. They come in many forms and may incorporate a wide variety of internal algorithms to transform and process data, allowing them to approximate complex functions (LeCun et al. 2015). Neural networks that consist of many layers are referred to as deep neural networks and their field of research is called deep learning. These models are especially flexible in processing input data beyond the simple tabular form, i.e. data that is organised in the form of rows and columns of one table. In phenomics, this is used to process image and spectral data (Ubbens and Stavness 2017; Tross et al. 2024). Additionally, deep learning can be used for increasingly complex genotype by environment models, as the models are able to process 'multi-modal' data, i.e. datasets that consists of different types of data (Togninalli et al. 2023; Montesinos-López et al. 2024). A substantial part of the latest success stories of ML outside of plant breeding were achieved by deep learning. However, using marker data for genomic prediction typically is a case of tabular data usage, and while there is a broad and diverse array of neural networks, they often do not perform well with tabular data (Shwartz-Ziv and Armon 2022).

Machine learning for genomic prediction

Applying ML to genomic prediction, where inputs are typically SNP matrices, has proven challenging. A wide variety of different algorithms has been tested in studies on genomic prediction in plant breeding for different crops. A benchmarking study reported large differences in prediction accuracy across algorithms and species, with no algorithm consistently outperforming the others (Azodi et al. 2019). Similar patterns were observed in simulated animal and real maize data, where all tested algorithms achieved comparable prediction accuracy (Lourenco et al. 2024). In one study, fully connected and convolutional neural networks have been shown to outperform linear models only when epistatic effects are high (Zingaretti et al. 2020). Local convolutional networks performed well on very large simulated datasets or in scenarios with few epistatic QTLs, but underperformed for smaller simulated and real datasets (Pook et al. 2020). In contrast, no advantage of local convolutional networks over other algorithms was found in a comparative study involving simulated and real traits from *Arabidopsis*, maize, and soybean (John et al. 2022). In that study, BayesB consistently yielded the best performance on simulated data, while performance on real data varied by trait and species. Ensembles of multiple ML models were shown to outperform individual models in terms of prediction accuracy (Kick and Washburn 2023; Tomura et al. 2025).

Random forests have also been used in combination with variable selection methods for genomic prediction (Gabur et al. 2022; Heinrich et al. 2023). In one approach, markers from rapeseed and wheat datasets were selected based on their variable importance from a random forest model, with subsets of the top 100 and 1000 markers used as inputs in models for genomic prediction. In some cases, this led to better performance than RR-BLUP (Gabur et al. 2022). In another study, an incremental approach was used in which markers were added to a random forest model based on GWAS results, with the optimal number of markers chosen based on the prediction accuracy through cross-validation on the training set (Heinrich et al. 2023). The optimum number of markers was then included in the final model. This strategy often preserved prediction accuracy while reducing the number of SNPs, though no standard reference method was included in the comparison.

Other studies have combined the genetic data with environmental data to predict genotypes for specific environments. In one study, tree-based models outperformed linear random effect models in settings where environmental data were included

(Westhues et al. 2021). When different test set scenarios were used, neural networks showed better performance than GBLUP only when genotype, environment, and management data in the test set were not represented in the training data, but performed worse when such information was available during training (Washburn et al. 2021). In a similar context including genetic, environmental, and management inputs, BLUPs based on RKHS achieved the lowest average error, although they exhibited higher variance than neural networks (Kick et al. 2023). Additionally, the same study also showed that simple linear models did not perform much worse than other models, which is surprising, given the complex nature of the prediction task.

In summary, multiple comparative studies across diverse crops and traits, using different algorithms and data types, have shown that no single ML algorithm consistently outperforms others, and that standard models like RR-BLUP and GBLUP often remain competitive or superior in accuracy and speed (Azodi et al. 2019; Zingaretti et al. 2020; John et al. 2022; Lourenco et al. 2024). This aligns with the "No Free Lunch" theorem (Wolpert and Macready 1997), which is often cited in this context. It states that while some algorithms may perform well on specific tasks, no single algorithm performs best across all possible tasks. Despite the initial hopes for ML to achieve higher prediction accuracy than the standard methods, this does not seem to be the case.

One major obstacle that may be partially responsible for the lack of success of ML in plant breeding is the relatively small size of training datasets often used in plant breeding. As noted earlier, field-based phenotyping remains time-consuming and costly, resulting in datasets that are considered small by ML standards. Deep learning models, in particular, are known to be data-hungry and prone to overfitting if the provided data is limited and sparse (Brigato and Iocchi 2021). To make ML more viable for breeding applications, either new data generation strategies (e.g., high-throughput phenotyping, pooled datasets) or more data-efficient models are needed (Hayes et al. 2023). Another challenge is methodological: Many ML models used in plant breeding are directly adapted from other scientific fields without adjustment for biological context, data structure, or statistical requirements. Overcoming this will require specialised algorithm development and architectures that more closely fit the specificities of the field of plant breeding (Hayes et al. 2023). Thus, ML in plant breeding can still be considered to be in its early developmental stage, in which its full potential might not yet have been realized.

Objectives

Given the early stage of ML adoption in plant breeding, the primary goal of this thesis was to assess its potential for genomic prediction across its three key aspects: input features, modeling algorithms, and target traits. The work included comparisons across diverse species (including inbred and hybrid crops), various resistance traits and yield, and multiple approaches to data preprocessing, feature engineering, and model selection. The following concrete steps that were taken:

1. The initial phase of this investigation of ML-based genomic prediction aimed to compare various ML algorithms with GBLUP for predicting yield across five hybrid datasets from three crop species. The primary objectives were to establish a baseline by creating a minimalist ML model based solely on nominal parentage data against standard methods like GCA prediction or GBLUP, and subsequently compare these standards to ML algorithms utilizing full marker data during training.

2. In the second study, the aim was to extend the analysis to resistance traits. Using homozygous wheat lines, resistance scores for five fungal diseases were to be predicted. This work intended to incorporate a broader range of engineered features than the previous study, including haplotype blocks and autoencoder-extracted features as inputs. Additionally, the plan was to test multiple trait transformations (log-transformed, categorized) to evaluate the influence of target scaling alongside the untransformed resistance scores.

3. Building on the findings from the second study, the aim of the third study was to design a novel neural network architecture that combines both autoencoders and haplotype blocks to generate a new type of input feature. This architecture would then be tested on different datasets, including hybrids and inbred lines, with yield as the target trait. The first objective was to assess the novel method's ability to maintain high prediction accuracy while simultaneously reducing the data dimension. In a second step, the aim was to enhance the model to create a new way to estimate haplotype block effects beyond simply summing up individual marker effects.

Chapter 2

Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP[1]

[1]P. G. Heilmann, M. Frisch, A. Abbadi, T. Kox, and E. Herzog (2023) Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP. *Frontiers in Plant Science* **14**:262.

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 21 July 2023
DOI 10.3389/fpls.2023.1178902

OPEN ACCESS

EDITED BY
Ali M. Missaoui,
University of Georgia, United States

REVIEWED BY
Milind B. Ratnaparkhe,
ICAR Indian Institute of Soybean Research, India
Yanru Cui,
Hebei Agricultural University, China

*CORRESPONDENCE
Eva Herzog
Eva.Herzog@agrar.uni-giessen.de

RECEIVED 03 March 2023
ACCEPTED 26 June 2023
PUBLISHED 21 July 2023

CITATION
Heilmann PG, Frisch M, Abbadi A, Kox T and Herzog E (2023) Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP.
Front. Plant Sci. 14:1178902.
doi: 10.3389/fpls.2023.1178902

Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP

Philipp Georg Heilmann[1], Matthias Frisch[1], Amine Abbadi[2], Tobias Kox[2] and Eva Herzog[1]*

[1]Institute of Agronomy and Plant Breeding II, Justus Liebig University, Gießen, Germany, [2]NPZ Innovation GmbH, Holtsee, Germany

Testcross factorials in newly established hybrid breeding programs are often highly unbalanced, incomplete, and characterized by predominance of special combining ability (SCA) over general combining ability (GCA). This results in a low efficiency of GCA-based selection. Machine learning algorithms might improve prediction of hybrid performance in such testcross factorials, as they have been successfully applied to find complex underlying patterns in sparse data. Our objective was to compare the prediction accuracy of machine learning algorithms to that of GCA-based prediction and genomic best linear unbiased prediction (GBLUP) in six unbalanced incomplete factorials from hybrid breeding programs of rapeseed, wheat, and corn. We investigated a range of machine learning algorithms with three different types of predictor variables: (a) information on parentage of hybrids, (b) in addition hybrid performance of crosses of the parental lines with other crossing partners, and (c) genotypic marker data. In two highly incomplete and unbalanced factorials from rapeseed, in which the SCA variance contributed considerably to the genetic variance, stacked ensembles of gradient boosting machines based on parentage information outperformed GCA prediction. The stacked ensembles increased prediction accuracy from 0.39 to 0.45, and from 0.48 to 0.54 compared to GCA prediction. The prediction accuracy reached by stacked ensembles without marker data reached values comparable to those of GBLUP that requires marker data. We conclude that hybrid prediction with stacked ensembles of gradient boosting machines based on parentage information is a promising approach that is worth further investigations with other data sets in which SCA variance is high.

KEYWORDS

machine learning, stacked ensembles, gradient boosting, genomic prediction, general combining ability, specific combining ability, hybrid breeding, hybrid prediction

1 Introduction

Hybrid breeding programs have been a decade-long success story in corn, but are also increasingly implemented in crops that have previously been commercialized as homozygous line varieties, such as wheat (Schulthess et al., 2017), barley (Philipp et al., 2016) or rapeseed (Stahl et al., 2017). By implementing hybrid breeding, breeders hope to improve performance, resilience and yield stability of their varieties. For maximizing heterosis and hybrid performance, the hybrid breeding material is usually arranged in so-called heterotic groups of individuals with similar combining ability and heterotic response when crossed to individuals from genetically distinct germplasm groups (Melchinger and Gumber, 1998). Two heterotic groups used in a specific hybrid breeding program are referred to as a heterotic pattern. Breeding progress and establishment of novel heterotic patterns is based on constant selection for hybrid performance, heterosis and combining ability in test crosses between the parent groups. In most breeding programs, the number of potential hybrid combinations of the parental lines from the heterotic groups exceeds the number of hybrids that can be evaluated in field trials by far.

By estimating the general combining ability (GCA, Hallauer et al., 2010) of the parental components, the performance of the resulting hybrids can be predicted using the sum of both parental GCA values. GCA estimates can be obtained by testing only a part of all possible crosses of parental lines from two different genetic groups in the field. If heterotic patterns have been established, candidates for hybrid parents can be very efficiently identified with only one or a few testers from the opposite heterotic group due to the high accuracy and predominance of GCA variance over special combining ability (SCA) variance (Melchinger and Gumber, 1998). The GCA prediction approach is simple, yet in many breeding programs surprisingly precise. For decades, it has formed the backbone of successful hybrid breeding programs.

However, newly established hybrid breeding programs usually cannot rely on established heterotic patterns. These hybrid programs are often characterized by a predominance of SCA variance over GCA variance, which complicates GCA-based testing strategies. Due to the high costs of evaluating large numbers of potential hybrid combinations in the field, genetic bottlenecks in one or both parent germplasm groups, and unsuccessful crosses without viable offspring, testcross factorials in these hybrid programs are often highly unbalanced and consist only of a small fraction of all possible hybrid combinations between the parent groups. As a consequence, new prediction methods that enhance the accuracy of hybrid prediction in sparse unbalanced factorials with high relevance of SCA are continually sought after to increase the efficiency of selection in newly established hybrid breeding programs.

Genomic prediction models for hybrid performance are able to incorporate information of genome-wide marker data in addition to phenotypic estimates collected in the field. These genomic prediction models have been successfully used to predict the testcross performance of untested parental lines (Albrecht et al., 2011; Hofheinz et al., 2012). For parent groups with a high ratio of SCA over GCA variance, as frequently observed in newly established hybrid breeding programs, modifications of the genome-wide BLUP (GBLUP) model incorporating both GCA and SCA components have been shown to increase prediction accuracy over models considering additive GCA effects only (Technow et al., 2012; Technow et al., 2014).

The term machine learning (ML) summarizes a large number of comparatively new prediction methods in statistics, mathematics, and computer science (Domingos, 2012). These methods have gained a lot of popularity due to their proven ability to solve problems in many different fields of research more effectively than classical approaches (Butler et al., 2018; Abbas et al., 2019; Dargan et al., 2020), but have not yet been widely implemented in hybrid breeding programs. A common feature of ML algorithms is that they are able to model non-linear interactions, and thus find complex underlying patterns within data better than other algorithms (Bishop, 2006; Hastie et al., 2009). Each algorithm has a wide variety of parameters that have to be manually defined by the user, so-called hyperparameters. Thus, an important part of the application of ML is the search for the optimal hyperparameters, which is generally known as hyperparameter optimization (Probst et al., 2019). This process requires knowledge on ML and, depending on the task and data set, a lot of computational resources.

Among the most popular ML algorithms are decision-tree based methods. Most commonly used are gradient boosting (GB, Friedman, 2001), which consists of several decision trees trained after another, extreme gradient boosting (XGB, Chen and Guestrin, 2016) which is a computationally more efficient version of GB and specialized in handling sparse data, and Random Forests (RF, Breiman, 2001), where multiple decision trees are trained in parallel. There is also the field of deep learning centered around the application of artificial neural networks (ANN, Goodfellow et al., 2016) that gained major popularity in the recent years. A classic type of ML algorithm are support vector machines (SVM, Cortes and Vapnik, 1995), which were first introduced as a classification algorithm but have later been adapted to regression tasks (Hastie et al., 2009). Reproducing kernel hilbert spaces (RKHS, Perez and de los Campos, 2014) are similar to SVM and are additionally already quite common in plant breeding. A very simple and fast algorithm focused on filling out sparse matrices is matrix factorization (MF, Koren et al., 2009), most commonly used in recommender systems. Stacked ensembles (SE, Breiman, 1996; Van Der Laan et al., 2007) utilize the output of already existing models to train a new model on top. This model combines aspects of all models it incorporates.

Recent studies have started to investigate the potential of ML for tasks related to plant breeding. ML has been used for handling genotype-by-environment interactions in multi-environmental trials (Montesinos-López et al., 2018b; Gillberg et al., 2019; Washburn et al., 2021; Westhues et al., 2021), the identification of the optimal set of markers used for prediction (Li et al., 2018a; Gabur et al., 2022), phenomic prediction and image classification (Mohanty et al., 2016; Nagasubramanian et al., 2018; Cuevas et al., 2019; Nagasubramanian et al., 2019) as well as genomic prediction (Ma et al., 2018; Azodi et al., 2019; Banerjee et al., 2020; Montesinos-López et al., 2021). The majority of recently published studies rely on genomic data as the basis of their

predictions. Studies without genomic data usually incorporate other forms of complex data or prove the concept of a single specific algorithm without conducting a broad investigation of the potential of available algorithms (Montesinos-López et al., 2018a; Khaki and Wang, 2019; Khaki et al., 2020). To our knowledge, a comparison of ML methods under the same conditions as GCA-based hybrid prediction with real-life data from ongoing breeding programs has not yet been investigated.

Our goal was to investigate the suitability of the ML algorithms GB, RF, XGB, ANN, MF, RKHS, SVM, and an SE based on GB machines (GB-SE) for prediction of hybrid yield in six unbalanced factorials of different structure and size from hybrid breeding programs of rapeseed, wheat, and corn. In particular, our objectives were (i) to compare the prediction accuracy of ML algorithms based on hybrid parentage and phenotypic field data to classical GCA-based prediction, (ii) to test if the best ML algorithm from objective (i) can compete with marker-based predictions from a GBLUP model incorporating GCA and SCA components, (iii) to investigate if ML algorithms based on genotypic data or a combination of genotypic data and parentage information can outperform a GBLUP model incorporating GCA and SCA components, (iv) and to develop a user-friendly standardized procedure for hyperparameter optimization that is applicable in a wide range of hybrid breeding programs.

2 Material and methods

2.1 Software

All analyses were conducted in R 4.0.3 (R Core Team, 2022). For analysis of field data, GCA and SCA effects, GBLUP and implementation of the ML algorithms, we used the R packages 'lme4 1.1-31' (Bates et al., 2015), 'emmeans 1.7.3' (Lenth, 2021), 'sommer 4.2.0' (Covarrubias-Pazaran, 2016; Covarrubias-Pazaran, 2018), 'h2o 3.38.0.1' (LeDell et al., 2020; Chen et al., 2022), 'kernlab 0.9-31' (Karatzoglou et al., 2004), 'mlr 2.19.0' (Bischl et al., 2016), 'parallelMap 1.5.1' (Bischl et al., 2020), 'BGLR 1.1.0' (Perez and de los Campos, 2014), and 'recosystem 0.5' (Qiu et al., 2021), which are available from the Comprehensive R Archive Network (CRAN). Additionally, we used the package 'SelectionTools 21.3' which is freely available under http://population-genetics.uni-giessen.de/software/. For which algorithms the specific packages were used is described in detail below. A working R code example for tuning GB models with random grid search and building the GB-SE is provided for data set Co1 as PDF in the supplementary material, and as an R script under https://github.com/PGHeilmann/Minimalist-ML-frontiers.

2.2 Experimental data sets

We investigated six experimental data sets of hybrid yield which comprised incomplete factorials of two unbalanced parent groups. Descriptive statistics for the investigated factorials are summarized in Table 1. A graphical overview of the crossing matrices and the realized hybrid combinations for the six factorials is given in Figures S1, S2.

The phenotypic and genotypic data of rapeseed factorials Ra1 - Ra3 was provided by Norddeutsche Pflanzenzucht Hans-Georg Lembke KG. Factorial Ra1 consisted of 746 realized hybrids derived from two parent groups with 381 and 14 inbred lines. Factorial Ra2 consisted of 1621 realized hybrids derived from two parent groups with 756 and 24 inbred lines. Factorial Ra3 consisted of 1081 realized hybrids derived from two parent groups with 516 and 29 inbred lines. Phenotypic yield data was provided as adjusted entry means for each hybrid.

The phenotypic and genotypic data of wheat factorial Wh1 was published in Zhao et al. (2015) and Gowda et al. (2014). Factorial Wh1 consisted of 1604 realized hybrids derived from two parent groups with 120 and 15 inbred lines. Phenotypic yield data was provided as adjusted entry means for the hybrids in 11 environments. We calculated adjusted entry means for hybrid yield over environments with the mixed linear model $y_{ij} = \mu + g_i + e_j + \epsilon_{ij}$, where μ is the population mean, g_i is the fixed effect of the i -th hybrid genotype, e_j the random effect of the j-th environment, and ϵ_{ij} is the residual error. We did not include a genotype-by-environment interaction term, as the published data consisted of environment-specific adjusted entry means of the hybrids, which already included the replications within environments. We used the R packages 'lme4 1.1-31' (Bates et al., 2015) and 'emmeans 1.7.3'(Lenth, 2021) for fitting the model and calculating the adjusted entry means.

The phenotypic and genotypic data of corn factorial Co1 was published in Technow et al. (2014) and accessed through the R package 'sommer 4.2.0'(Covarrubias-Pazaran, 2016). Factorial Co1 consisted of 1254 hybrids derived from two parent groups with 123 and 86 inbred lines. Phenotypic yield data was provided as adjusted entry means for each hybrid.

The phenotypic and genotypic data of corn factorial Co2 was published in Schrag et al. (2018). Factorial Co2 consisted of 550 hybrids derived from two parent groups with 50 and 41 inbred lines. Phenotypic yield data was provided as adjusted entry means for each hybrid.

2.3 Pre-processing of genotypic marker data

Genotypic marker data was only available for factorials Ra1, Wh1, Co1 and Co2. For all four factorials, genotypic data consisted of single nucleotide polymorphisms (SNPs). The original marker data consisted of 52157 (Ra1), 1280 (Wh1), 35478 (Co1), and 37392 (Co2) SNP markers, respectively. Markers were removed from a data set if expected heterozygosity was below 10%, or if more than 1% of entries were missing. The remaining missing data was imputed using the mean of the respective marker. 'SelectionTools 21.3' was used for filtering and 'sommer 4.2.0' for imputing the data. For all data sets, it was checked that genetic markers evenly covered the whole genome. After pre-processing, 10880 (Ra1), 1264 (Wh1), 26069 (Co1) and 33666 (Co2) SNP markers remained for further analysis.

2.4 Linear model for GCA and SCA effects

The GCA of the parents GCA_{1i} and GCA_{2j}, and the SCA of the hybrids SCA_{ij} were predicted by BLUP with the mixed linear model

$$y_{ij} = \mu + GCA_{1i} + GCA_{2j} + SCA_{ij}$$

where y_{ij} is the (adjusted) treatment mean of the hybrid of the i-th parent from parent group 1 and the j -th parent from parent group 2, μ is the population mean, GCA_{1i} is the random GCA effect from the i-th parent from parent group 1, GCA_{2i} is the random GCA effect from the j-th parent from parent group 2, SCA_{ij} is the SCA effect. We also used this model for estimating the GCA and SCA variances. We used the R package 'lme4 1.1-31' (Bates et al., 2015) for fitting the model and predicting the GCA and SCA as well was the corresponding variance components.

To evaluate the relevance of GCA and SCA for hybrid yield in the single factorials, we calculated the sum $GCA_{1i} + GCA_{2j}$ for each realized hybrid, the Pearson correlations $r(GCA_{1i} + GCA_{2j}$, Hybrid yield) and $r(SCA_{ij}$, Hybrid yield) for all realized hybrids, and the proportion of the contribution of the SCA variance to the total genetic variance in the factorial $\tau = \sigma^2_{SCA}/(\sigma^2_{GCA_1} + \sigma^2_{GCA_2} + \sigma^2_{SCA})$ (Table 1).

2.5 Mixed model for GBLUP

Using the adjusted entry means for the hybrids from field trial analysis as phenotypic inputs y, we fitted a GBLUP model including GCA and SCA effects (Technow et al., 2014):

$$y = 1\beta_0 + Z_1 g_1 + Z_2 g_2 + Z_S s + e$$

where β_0 is a fixed intercept, Z_1 and Z_2 are the incidence matrices for the parents from parent groups 1 and 2, Z_S is the incidence matrix for the hybrids, g_1 and g_2 are vectors of random GCA effects from the parental lines from group 1 and 2, s is the vector of the random SCA effects for the hybrids, end e is the vector of residual errors. The genomic relationship matrices G_1 and G_2 for g_1 and g_2 were calculated as $G_1 = W_1 W_1' / c$ and $G_2 = W_2 W_2' / c$, where $w_{uv} = x_{uv} + 1 - 2p_v$ and $c = 2\sum p_v(1 - p_v)$, where u is the index of the parent, v is the index of the marker, x_{uv} the coding

TABLE 1 Descriptive statistics, variance components and proportion of SCA variance of the total variance τ for the six experimental data sets.

data set	Ra1	Ra2	Ra3	Wh1	Co1	Co2
No. of parents in group 1	381	756	516	120	123	50
No. of parents in group 2	14	24	29	15	86	41
Ratio group 1/group 2	27.2	31.5	17.8	8.0	1.4	1.2
No. of possible hybrids	5334	18144	14964	1800	10578	2050
No. of realized hybrids	746	1621	1081	1604	1254	550
Fraction of realized hybrids	14.0%	8.9%	7.2%	89.1%	11.9%	26.8%
Group 1: no. of crosses per parent						
Mean	2.0	2.1	2.1	13.4	10.2	11.0
Median	2.0	2.0	2.0	14.0	7.0	11.0
Range	1-6	2-5	1-6	3-15	2-55	3-26
Group 2: no. of crosses per parent						
Mean	53.3	67.5	37.3	106.9	14.6	13.4
Median	34.5	22.5	6.0	107.0	13.0	13.0
Range	2-140	2-247	1-222	91-117	1-99	2-42
Heterotic pools	No	No	No	No	Yes	Yes
$r(GCA_{1i} + GCA_{2j}$, Hybrid yield)	0.67	0.82	0.92	0.76	0.91	0.94
$r(SCA_{ij}$, Hybrid yield)	0.93	0.88	0.69	0.70	0.49	0.40
Variance components						
$\sigma^2_{GCA_1}$	0.516	5.50	12.95	0.048	43.21	73.05
$\sigma^2_{GCA_2}$	0.774	3.32	2.20	0.024	20.25	24.39
σ^2_{SCA}	2.663	9.35	5.90	0.051	17.47	15.78
$\tau = \sigma^2_{SCA}/(\sigma^2_{GCA1} + \sigma^2_{GCA2} + \sigma^2_{SCA})$	0.67	0.51	0.28	0.44	0.22	0.14

number of the genotype of parent u at marker locus v, i.e. -1 or 1, and p_v the allele frequency of the 1 allele in the respective parent group (Endelman and Jannink, 2012). The genomic relationship matrix S for s was calculated as the Kronecker product $G_1 \otimes G_2$ in accordance to Stuber and Cockerham (1966). We used the R package 'sommer 4.2.0' 2 (Covarrubias-Pazaran, 2016; Covarrubias-Pazaran, 2018) for fitting the GBLUP model and predicting hybrid yields.

2.6 ML algorithms

2.6.1 Input variables

For the ML algorithms, we investigated three different scenarios: prediction of hybrid yield without genotypic information, prediction of hybrid yield with genotypic marker data, and a combined set of variables. For prediction without genotypic information, we investigated two different sets of input variables. The parentage-based set of input variables consisted of the nominal parent factor levels, i.e. names or barcodes of the parent lines. For each hybrid, the only available information were the names of its parents. Thus, the original set of input variables consisted of only two variables, which for some algorithms were converted to binary variables via one-hot encoding.

For the second set of input variables, we again determined the two parents of each hybrid. The input variables consisted of the hybrid yields of each parent from the available crosses with all parents from the opposite parent group. Thus, this set of continuous input variables consisted of as many variables as the sum of the number of parents in the two parent groups. The hybrid yield in the response variable in a specific row of the data set was always deleted from the input variables in this row.

For prediction with genotypic marker data, we used the incidence matrix of the virtual hybrid genotypes for the pre-processed marker data coded with -1, 0, and 1 for homozygous for the first allele, heterozygous, and homozygous for the second allele as input variables.

For the combined set of variables, we merged the parentage-based set of input variables with the genotypic marker data and used all available information as input variables for the models.

2.6.2 Investigated ML algorithms for different sets of input variables

The algorithms MF, SVM, GB, RF, ANN, and GB-SE were investigated with the parentage and the hybrid yields input variable sets. The algorithms XGB, XGB-SE, RKHS and SVM were investigated with genotypic marker data as input variables.

2.6.3 Hyperparameters

A comprehensive overview over all the hyperparameter values considered for each ML algorithm is given in Table S1.

For GB, we tuned the number of decision trees (n_trees), the maximum depth of the trees (max_depth), the minimum number of observations per split (min_rows), the sample rate of observations per tree (sample_rate), and the number of bins for categorical variables (nbins_cats). We manually set the learning rate (learn_rate) to a constant value of 0.1 and used the default settings for all other hyperparameters.

For XGB, we essentially tuned the same hyperparameters as for GB, but used a fixed number of trees and tuned the learning rate. When XGB was only used with genotypic marker data, the hyperparameter nbins cats was removed. In this case the pruning parameter γ (gamma) was added instead. To handle the high dimensionality of the markers, random column sampling per tree (col sample rate by tree) was introduced.

Early stopping was used to reduce the computational time required.

For RF, we considered the same hyperparameters as for GB, with the exception that a learning rate does not exist for RF.

For ANN, we used the classical multilayer perceptron architecture for ANN (Goodfellow et al., 2016). We tuned the number of hidden layers and nodes within the hidden layers (hidden), the learning rate (rate), the number of iterations (epochs) and added input dropout (input_dropout) to some models. We used the rectified linear unit activation function for all models. The categorical variables of the parentage variable set were by default converted to binary variables by one-hot encoding.

For the GB-SE, we used ridge regression as the super learner and a certain optimal number of GB models selected by the grid search procedure and criteria described below as inputs.

The algorithms GB, XGB, RF, ANN and GB-SE were implemented using 'h2o 3.38.0.1' for R (LeDell et al., 2020; Chen et al., 2022).

For SVM, we tuned the hyperparameters ϵ (epsilon) and C (C). We considered three different kernels: linear, polynomial and radial. For the polynomial kernel, the degree of the polynomial was tuned. SVM was implemented using the R package 'kernlab 0.9-31' (Karatzoglou et al., 2004) with packages 'mlr 2.19.0' (Bischl et al., 2016) and 'parallelMap 1.5.1' (Bischl et al., 2020) for tuning and parallelization.

For RKHS, we used three gaussian kernels with bandwidth parameters 0.1, 0.5 and 2.5, respectively. We used the default function settings for all other parameters. RKHS was calculated using the package 'BGLR 1.1.0' [54]. RKHS did not require hyperparameter tuning.

For MF, we tuned the number of latent variables (dim), and used l1 regularization for some models during training, (costq_l1, costp_l1). The learning rate (lrate) was manually set to a constant value of 0.05 and the number of iterations (niters) to 500.

Categorical variables were encoded as numbers before being passed to the algorithm. MF was implemented using the package 'recosystem 0.5' (Qiu et al., 2021).

2.7 Random grid search

In order to determine the optimal set of hyperparameters for a given data set and ML algorithm, we performed a random grid search over a large hyperparameter space (Table S1) for every ML algorithm except RKHS, which did not require tuning. As a

stopping criterion, we set a maximum number of 50 models. Thus, for each ML algorithm that required tuning, 50 models were trained, each with a randomly chosen set of hyperparameters from the hyperparameter space. The performance of these 50 models within the gridsearch was evaluated with a 10-fold cross validation for the respective training set. We used the mean squared error (MSE) to evaluate the 50 hyperparameter combinations, since this metric was available for all algorithm implementations we used. The hyperparameter combination with the lowest MSE was considered optimal and used to predict the hybrid yields of the test set.

We used the 50 models created in the random grid search for GB to build the GB-SE. To choose the optimal number of models to include in the GB-SE, we used an iterative process where the best $5, 10, \ldots, 50$ models were included in the GB-SE, and thenevaluated with the Pearson correlation $r(y_{ij}, \hat{y}_{ij})$ between the observed and predicted yield in a 10-fold cross validation. The optimum number of models to include in the final GB-SE was chosen according to the highest value of $r(y_{ij}, \hat{y}_{ij})$.

2.8 Cross validation, pre-processing of training sets and test sets, and prediction of test set hybrids

All investigated prediction models and ML algorithms were tested in a cross validation procedure in order to evaluate the generalizability and stability of the predictions, and to evaluate the consistency of the applied grid search procedure for model selection. For cross validation, the respective factorial was randomly split into a training set consisting of 90% of the available hybrids, and a test set consisting of the remaining 10% of hybrids. This random split was repeated 100 times for each factorial. We removed all hybrids from the test set for which only one or none of the parents were represented in the training set as GCA estimation requires both parents to be available in the training set. If hybrid yields were used as input variables, all hybrid yields from the test set hybrids were removed from the training set.

After pre-processing, each of the investigated prediction models and ML algorithms was trained on the 100 training sets. The resulting models were used to predict the hybrid yields of the test set hybrids. For GCA prediction, the yield of the test set hybrids was predicted as $\hat{y}_{ij} = \mu + GCA_{1i} + GCA_{2j}$. For GBLUP and the ML algorithms, the yield of the test set hybrids $\hat{y}_{ij}$ was predicted with the respective prediction routines implemented in the R packages.

2.9 Evaluation criteria for model performance and comparisons across algorithms

To evaluate and compare model performance across prediction models and ML algorithms, we calculated the Pearson correlation $r(y_{ij}, \hat{y}_{ij})$ between the observed and predicted yield of the test set hybrids. This correlation is referred to as "prediction accuracy". For each method and factorial, we also compared the 20 best predicted hybrids to the 20 best observed hybrids and determined the percentage of overlap, since accurate identification of the best hybrids is more relevant to breeders than accurate predictions of low performing hybrids.

3 Results

3.1 Predictions based on parentage information and hybrid yields

We investigated two different sets of input variables without genotypic information: parentage information, and yields of all other realized crosses of the parents of a specific hybrid. For algorithms using the parentage information as input variables, we observed a wide range of median prediction accuracies for the different crops and factorials (Figure 1, red and grey boxplots). The lowest overall prediction accuracieswere observed for the factorials Ra1 and Ra2, the median prediction accuracies ranging between 0.14 and 0.45, and between 0.42 and 0.54, respectively. The factorials Ra3 and Wh1 resulted in intermediate median prediction accuracies between 0.68 and 0.75,and 0.69 and 0.72, respectively. The factorials Co1 and Co2 resulted in the highest median prediction accuracies, ranging between 0.80 and 0.87, and 0.89 and 0.91, respectively.

GB was the best single ML algorithm for most investigated factorials with the exception of Ra1 and Co1, where RF and SVM performed better. MF resulted in the lowest median prediction accuracies for all factorials except Ra1. None of the investigated single ML algorithms resulted in higher median prediction accuracies than classical GCA prediction (Figure 1, red vs. grey boxplots). The GB-SE increased median prediction accuracies in the factorials with the overall lowest prediction accuracies Ra1 and Ra2 from 0.39 to 0.45, and from 0.48 to 0.54, respectively. For all other factorials, the median prediction accuracy of the GB-SE was equivalent to or only marginally better than GCA prediction.

Algorithms using hybrid yields as input variables increased computation time in comparison to algorithms based on parentage, but resulted in slightly lower median prediction accuracies (Figure S3). The algorithm ANN did not converge for all factorials with this set of input variables.

3.2 Comparison of GCA prediction, GB-SE and GBLUP

GBLUP was only investigated for the factorials Ra1, Wh1, Co1 and Co2 with available marker data (Figure 1, blue boxplots). Median prediction accuracies of GBLUP were equivalent to GCA prediction and the GB-SE in the factorials Co1 and Co2. In factorial Wh1, GBLUP slightly improved median prediction accuracy from 0.71 with GCA and 0.72 with the GB-SE to 0.74. In factorial Ra1, GBLUP increased median prediction accuracy from 0.39 with GCA to 0.44, and resulted in an equivalent prediction accuracy as the GB-SE with 0.45. Neither ML algorithms nor GBLUP did reduce the variation of prediction accuracies across cross validation splits in

comparison to GCA prediction. Variation was generally low across cross validation splits for all investigated algorithms in factorials Wh1,Co1 and Co2, and largest in factorial Ra1 (Figure 1). For data sets Ra1, Ra2, Ra3, and Co1, GB-SE showed the highest percentage of overlap between the 20 best predicted and observed hybrids (Figure S9). For Wh1 and Co2, GBLUP showed the highest percentage of overlap. However, this percental overlap between the 20 best predicted and observed hybrids was generally similar for most algorithms with the exception of SVM and MF, which also performed poorly overall.

3.3 Effects of structure and composition of experimental data sets

The experimental data sets varied considerably in size, unbalancedness of the parent groups, percentage of realized hybrid combinations and relevance of GCA and SCA for hybrid yield (Table 1). As general trends, we observed that median prediction accuracies were high if the factorials relied on a heterotic pattern (Co1 and Co2), if percentage of realized hybrid combinations was high, i.e. if the factorial was almost complete (Wh1), if parent groups were more balanced, and if parentlines were represented in many hybrid crosses (Wh1, Co1 and Co2, compare Table 1 with Figure 1). Importantly, GBLUP and the GB-SE increased prediction accuracy considerably in comparison to GCA prediction for the factorials with the highest proportion of SCA variance in the total variance (Ra1 and Ra2). For these factorials, the correlation of SCA with hybrid yield was higher than the correlation of GCA with hybrid yield (Table 1), and overall prediction accuracy was low (Figure 1). For all other factorials, there was no improvement in comparison to classical GCA prediction.

3.4 Performance of ML algorithms with genotypic information

For data set Ra1 with the highest improvement of median prediction accuracy with ML algorithm GB-SE and GBLUP (Figure 1), we investigated four additional algorithms specifically tuned for marker data: XGB, RKHS, SVM and an XGB-SE (Figure 2), and a combined set of input variables including marker data and parentage information for the best ML algorithms, XGB-C and XGB-SE-C. With marker data only, XGB and the XGB-SE resulted in the highest prediction accuracies of 0.44 and 0.45, but did not improve prediction accuracy compared to the GB-SE based on parentage information and GBLUP. For the algorithms based on a combination of marker data and parentage information, XGB-C and XGB-SE-C, prediction accuracy was increased slightly to 0.46. The ensemble did not show any improvement over the single algorithm anymore. In spite of the marginal increase in prediction accuracy, the ML methods based on marker data or the combination of marker data and parentage information did not result in a higher percental overlap of the predicted and observed 20 best hybrids (Figure S9).

3.5 Tuning and importance of different hyperparameters for ML algorithms

The random grid search approach we used for model tuning yielded overall consistent results for the same data set across cross validation splits, as reflected in the heatmap in Figure 3 and Table S3. For most factorials, the same modelswere identified as the best and worst model in the majority of the cross validation splits. Where several models were identified as best, they were often

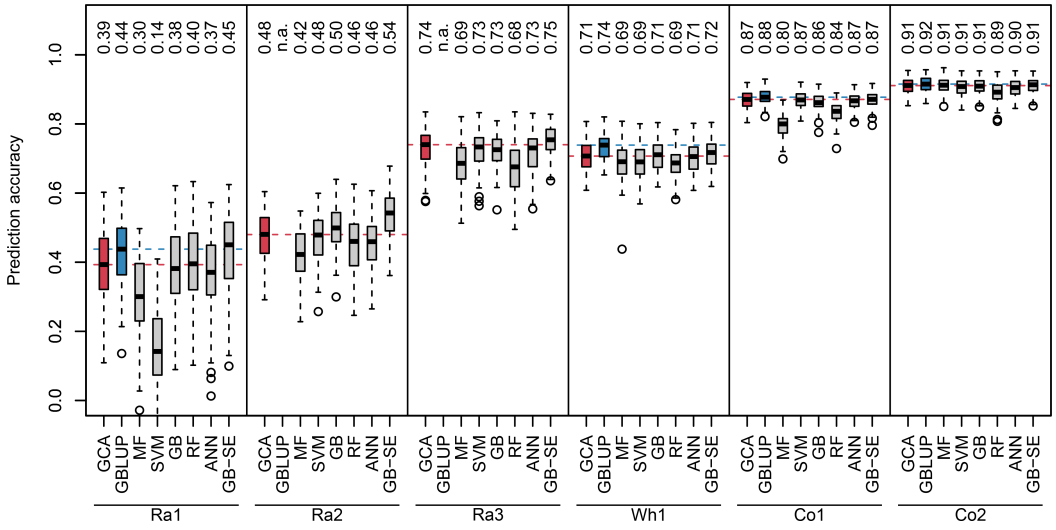

FIGURE 1
Boxplots of observed prediction accuracies in 100 cross validation splits for six different data sets (Ra1, Ra2, Ra3, Wh1, Co1, Co2) using eight different algorithms (GCA, GBLUP, MF, SVM, GB, RF, ANN, GB-SE). Median prediction accuracy is displayed above each boxplot. Median prediction accuracy of the baseline algorithms are represented by a dotted red (GCA) and blue (GBLUP) line. n.a., not available.

10.3389/fpls.2023.1178902

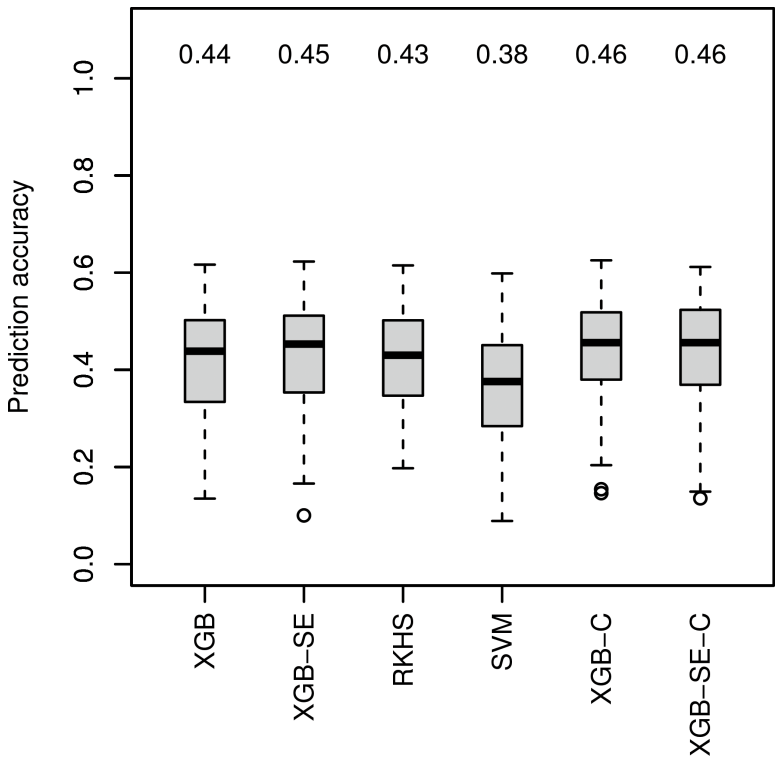

FIGURE 2
Comparison of observed prediction accuracies in 100 cross validation splits of marker-based ML algorithms for data set Ra1. Median prediction accuracy displayed above each boxplot. 'C' indicating models were trained on the combined variable set.

similar for the most important hyperparameters. As a stopping criterion for random grid search, we set a maximum number of 50 models, which in the case of GB also formed the basis for building the GB-SE. With the exception of factorials Co1 and Co2, which in most cases used less than 25 models, the GB-SE was for most factorials and cross validation splits build from 30 or more models (Figure S10). The factorials Ra2 and Ra3 even required 40-50 models in the majority of cross validation splits.

For evaluating the importance of different hyperparameters, we calculated correlations between hyperparameter levels and MSE (Table 2) and created ridge line plots for each hyperparameter level and the scaled MSE (Figures S4-S8). We also compared the best models for the different data sets from the random grid search approach (Table S2). For conciseness, we only looked at the hyperparameters for the most successful single ML algorithm GB, which also formed the basis for building the GB-SE. The hyperparameters with the greatest effect on the MSE were the number of bins for partitioning the data before determining the tree's best split point, and the maximum depth of the decision trees. As general trends, we observed that choosing an intermediate number of bins, ranging between 25-50% of the number of parents in the data set, resulted in the lowest MSE (Table S2). With respect to the maximum depth of the decision trees, we observed that the factorials Ra1-3 required deeper trees than the factorials Wh1, Co1 and Co2.

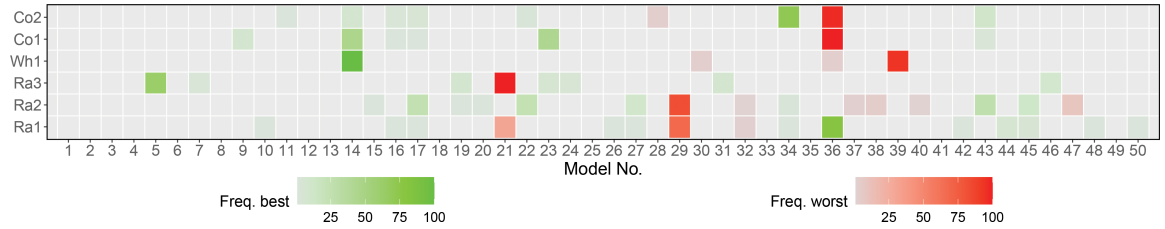

FIGURE 3
Selection of 50 models used in 100 cross validation splits for each data set using GB. Each model is corresponding to a unique random hyperparameter combination. Heatmap with green and red indicating how often a model was found to be the best or worst model over all iterations.

TABLE 2 Correlations between numerical hyperparameter levels and MSE of the models in the grid search.

Experiment	nbins_cats	max_depth	ntrees	sample_rate	min_rows
Ra1	0.62	-0.01	0.41	0.00	-0.08
Ra2	0.70	0.01	0.39	0.03	-0.09
Ra3	-0.33	0.00	0.21	-0.25	-0.11
Wh1	-0.38	0.06	0.07	-0.25	-0.34
Co1	-0.09	-0.03	0.19	-0.16	0.00
Co2	-0.22	-0.11	0.24	0.00	-0.18

4 Discussion

The present study is a case study exploring the potential of ML algorithms for prediction of hybrid yield in small unbalanced factorials. In accordance with the "No Free Lunch" theorem, implying that there is no single optimal algorithm for all problems and data sets, we tested a variety of single ML algorithms (MF, SVM, RF, GB, ANN, XGB, RKHS), as well as stacked ensembles of gradient boosting machines (GB-SE and XGB-SE) with optimized hyperparameters in comparison to the well-established approaches GCA prediction and GBLUP. The experimental data sets consisted of six unbalanced factorials from four different self-pollinating and outcrossing crops, and varied considerably in structure and size (Table 1, Figures S1 and S2). The parent groups were highly unbalanced with respect to the number of parent lines in the groups for some of the factorials, while the parent groups were almost balanced in size for other factorials, the ratios of group sizes ranging between 1.2-31.5. Moreover, the factorials were also unbalanced with respect to the number of crosses per parent line, which could range between 1 and 247 crosses per line. Some factorials such as Ra2 and Ra3 were very sparse, the percentage of realized hybrid combinations of all possible combinations being lower than 10%, while the factorial Wh1 was almost complete with almost 90% of realized hybrid combinations. The corn factorials Co1 and Co2 were created from the heterotic patterns of Flint and Dent (Technow et al., 2014; Schrag et al., 2018), while no heterotic pools were available for the wheat factorial Wh1 (Zhao et al., 2015) and the rapeseed factorials Ra1-3 (NPZ Innovation GmbH, personal communication). This is also reflected by the high contribution of SCA variance to total variance, and in the correlations of SCA and GCA with hybrid yield. As expected, we observed a higher relevance of SCA variance, and higher correlations of SCA with hybrid yield in the factorials for which no heterotic pools were available. Our initial research hypothesis was that ML algorithms might improve prediction accuracy in sparse, highly unbalanced factorials with a high relevance of SCA in comparison to GCA prediction based on a mixed linear model, and might even outperform GBLUP if genetic marker data is available.

4.1 Performance of ML algorithms based on parentage information and hybrid yields

We investigated three different scenarios, prediction without genotypic information, genomic prediction based on genotypic markers and a combination of both. In the first scenario, the only available information for predicting hybrid yields were the names of the hybrid parents, which we refer to as parentage information, and the hybrid yields from crosses with other parent lines from the factorial. For ANN, the parentage information was converted to binary variables indicating if a parent was used in a cross via one-hot encoding, all other algorithms used actual categorical variables with high cardinality. Using hybrid yields from crosses with other parent lines as predictors resulted in as many predictor variables as there were parent lines in the factorial, each variable being labeled with the name of a parent line and containing the information what yield was observed for the respective hybrid parents in a cross with this parent. Due to the unbalancedness and sparsity of the investigated factorials with respect to realized hybrid combinations, this led to predictor variables with a high percentage of missing values for rarely used parent lines, and numerous constant or almost constant variables for parent lines which were frequently used in crosses. Overall, the percentage of missing values in the predictor variables was way over 75% for most factorials in the cross validation splits, where all information from the validation set had to be removed from the training sets. As a consequence, some algorithms such as ANN did not converge for some of the factorials with this set of predictors, even if constant variables were dropped (Figure S3). For algorithms that converged, the use of hybrid yields as predictors considerably increased computation time, but never outperformed algorithms using parentage information only (Figure S3). We therefore focused our analysis on comparing ML algorithms based on parentage information to GCA prediction and GBLUP.

ML studies predicting yield without genotypic information have so far mostly focused on yield prediction for specific environments, using yield, parental and environmental information as well as management data as predictors. Shahhosseini et al. (2021) used GB, XGB and different ensembles for predicting corn yield of larger areas in the US corn belt with promising results. Khaki and Wang (2019) successfully designed a complex neural network for yield prediction in specific environments based on existing yield, environment and management data. The only other study using only parental information together with location and genetic clusters of the parents as input information to predict hybrid yield is the study of Khaki et al. (2020). However, this study was conducted using an exceptionally large data set of 294,128 hybrids provided by a large breeding company and a highly complex neural network approach. Babaie Sarijaloo et al. (2021) used the same dataset to compare different decision tree-based algorithms and

neural networks and found XGB to be the best, due to its regularization and handling of sparse data. This effect may be weaker in our study since data of Babaie Sarijaloo et al. (2021) was even sparser than ours, with only 4% of all possible combinations available. No other study has to our knowledge investigated the use of ML algorithms with parentage information in small unbalanced factorials.

4.2 Performance of ML algorithms based on parentage in comparison to GCA prediction and GBLUP

GB was the best single ML algorithm for all data sets except Ra1 (best algorithm RF) and Co1 (best algorithm SVM, Figure 1). While ANN excel at complex tasks such as processing images, text or speech, tree-based methods such as GB, XGB and RF often perform better in tasks with structured, tabular data (Shwartz-Ziv and Armon, 2022) and thus might be especially suitable for predicting unbalanced factorials with potentially extreme under-/overrepresentation of some parent lines. Additionally, due to the slow and stepwise fitting process, GB has a low tendency to overfit the data (James et al., 2013). In factorials with a high relevance of SCA for yield, the superior performance of GB might be due to the fact that every new tree in a GB machine will fit to the residual of the previous trees, thus potentially improving the accuracy of the SCA component of prediction in comparison to linear model approaches. However, in the present study none of the single ML algorithms MF, SVM, RF, GB, ANN could outperform classical GCA prediction based on a mixed linear model to a meaningful extent (Figure 1), irrespective of the investigated data set.

The GB-SE was the only ML algorithm which performed better than GCA prediction in the factorials with the overall lowest prediction accuracies Ra1 and Ra2, increasing median prediction accuracy from 0.39 to 0.45, and from 0.48 to 0.54, respectively (Figure 1). For all other data sets, the performance of the GB-SE was equal to or only marginally better than GCA prediction. GBLUP was only investigated for the four data sets with available genotypic information (Figure 1). GBLUP resulted in median prediction accuracies equivalent to GCA prediction and the GB-SE in data sets Co1 and Co2. In data sets Ra1 and Wh1, GBLUP increased median prediction accuracy from 0.39 to 0.44 and from 0.71 to 0.74 in comparison to GCA prediction, and resulted in approximately equivalent prediction accuracies as the GB-SE. When considering the best 20 hybrids only, GB-SE resulted in the highest percental overlap between the predicted and observed 20 hybrids for factorials Ra1, Ra2, Ra3 and Co1, and GBLUP resulted in the highest overlap for factorials Wh1 and Co2 (Figure S9). Thus, GB-SE and GBLUP had the highest ability to predict the top 20 hybrids correctly, but the differences between both methods were small in all investigated data sets. Neither ML algorithms nor GBLUP did reduce the variation of prediction accuracies across cross validation splits in comparison to GCA prediction. Variation was generally low across cross validation splits for all investigated prediction models in factorial Wh1, which was an almost complete factorial, and in factorials Co1 and Co2 with a more balanced representation of parent groups and single parent lines (Table 1). Both factors most likely resulted in less extreme random splits for cross validation, in which many parent lines from the validation set were under-represented in the training set. From these findings, we conclude that the applicability of single ML algorithms in small unbalanced factorials is limited. GB-SEs might be a viable alternative where GCA prediction and GBLUP result in low prediction accuracies, or where no genotypic data is available, but the performance apparently depends on the structure and composition of the investigated factorials.

4.3 Effects of structure and composition of experimental data sets

While it is difficult to draw generally valid conclusions from a limited number of very diverse experimental data sets, we observed some trends: For the two corn factorials Co1 and Co2, all algorithms yielded high prediction accuracies ranging between 0.80 and 0.91 (Figure 1). Neither the ML algorithms nor GBLUP outperformed GCA prediction. In contrast to rapeseed and wheat, hybrid breeding with selection for GCA in corn has been established for decades, resulting in the two genetically diverse Flint and Dent pools. If heterotic pools exist, hybrid performance of heterotic traits is to a large extent explained by GCA effects, which can be accurately estimated from crosses with few testers from the opposite pool. This is also reflected in the high correlations of GCA with hybrid yield of 0.91 and 0.94 that we observed for these data sets (Table 1). The wheat data set Wh1 is a published data set from a study with the objective of establishing a new heterotic pattern (Zhao et al., 2015). Even though the two parent groups consequently form no heterotic pools, prediction accuracy with GCA effects was high with 0.71 (Figure 1). GBLUP and the GB-SE yielded slightly higher and comparable prediction accuracies with 0.74 and 0.72, respectively (Figure 1). In comparison to the other data sets, Wh1 is the smallest and most complete factorial with a percentage of realized hybrid combinations of 89.1% of all possible combinations (Table 1), which were evaluated in 11 environments (Zhao et al., 2015). The correlation of GCA with grain yield was comparatively high with 0.76, and higher than the correlation of SCA with grain yield (Table 1). We conclude that in crops with established heterotic pools such as corn, or in almost complete factorials with highly accurate phenotypic data, the potential of ML algorithms and even GBLUP for increasing prediction accuracy is limited, and hybrid yield can be efficiently predicted with GCA effects.

In the rapeseed data sets Ra1-Ra3, the overall levels of prediction accuracy as well as the potential for improvement with more complex algorithms varied considerably (Figure 1). In data sets Ra1 and Ra2, even though overall prediction accuracy was low, the GB-SE increased prediction accuracy by 14.5% from 0.39 to 0.45, and by 12.9% from 0.48 to 0.54 in comparison to GCA. GBLUP could, due to the unavailability of genotypic data, only be investigated in Ra1, and resulted in a prediction accuracy of 0.44 that was almost equivalent to the value of the GB-SE. In contrast to the results of Ra1 and Ra2, prediction accuracies in factorial Ra3 were comparatively high with values of 0.74 for GCA and 0.75 for the GB-SE. The three rapeseed factorials were to our knowledge not based on a heterotic pattern of the

parent groups, but were pre-selected by the breeding company with the purpose to maximize hybrid yield (NPZ Innovation GmbH, personal communication). It is possible that parent groups for Ra3 were genetically more homogeneous within and more diverse between parent groups as was the case for Ra1 and Ra2. Factorials Ra1 and Ra2 are more sparse than Wh1 and Co2, with a percentage of realized hybrid combinations of 14.0% and 480 8.9% (Table 1), which might in part explain the overall low prediction accuracies. However, data set Ra3 is the sparsest of the investigated factorials, with a percentage of realized hybrid combinations of only 7.2%, and still resulted in prediction accuracies almost as high as the most complete factorial Wh1 (Figure 1). Moreover, Co1 with 11.9% of realized hybrid combinations is almost as sparse as Ra1 and Ra2 and still produced accurate GCA-based predictions and high prediction accuracies with the investigated ML algorithms.

4.4 Ratio of SCA to GCA effects

The major factors determining the overall level of prediction accuracy in the investigated factorials was τ describing the contribution of SCA variance to total variance, and the strength of the correlation of the SCA with hybrid yield. In the factorials Ra1 and Ra2 with overall low prediction accuracy, we observed very high correlations of hybrid yield with SCA of 0.93 and 0.88, while correlations of hybrid yield with GCA effects were lower in these data sets (Table 1). Conversely, in factorials with overall intermediate to high prediction accuracies in Figure 1, relevance of SCA variance and correlations of hybrid yield with SCA were always lower than correlations with GCA. This was most pronounced in the corn factorials Co1 and Co2 relying on the heterotic pattern of Flint and Dent. It also explains the high prediction accuracy observed in factorial Ra3 in comparison to factorials Ra1 and Ra2. In factorial Ra3, the correlation of hybrid yield with SCA amounted only to 0.69, while the correlation with GCA amounted to 0.92. The percentage of SCA variance in the total variance was low with 28% in factorial Ra3, and similar to the shares of 22% and 14% observed in factorials Co1 and Co2. In factorials Ra1 and Ra2, these proportions amounted to the highest with 67% and 51%of the total variance, respectively. The factorial Ra3 is exceptional, as in the absence of genetically diverse heterotic pools, hybrid performance is typically explained to a major extent by SCA. As the GB-SE was the best algorithm in factorials Ra1 andRa2 with high ratios of SCA variance to GCA variance, leading to considerable increases in prediction accuracy compared to GCA and equivalent values as GBLUP (Figure 1), we conclude that a potential field of application of ML algorithms in hybrid breeding programs is hybrid prediction in sparse unbalanced factorials with a high relevance of SCA effects.

4.5 Performance of ML algorithms with genotypic information

A major limitation of classical GCA prediction and the GB-SE based on parentage information compared to marker-based approaches is that hybrids for which only one or none of the parents have been tested before cannot be predicted. On the other hand, prediction accuracies are generally low for these so called type-1 and type-0 hybrids even with markers or more complex *omics* data (Zenke-Philippi et al., 2016, 511 Zenke-Philippi et al., 2017; Zhao et al., 2021). The potential to predict type-0 hybrids with marker-based ML in sufficiently large data sets remains to be investigated.

GBLUP did not increase prediction accuracy in comparison to classical GCA prediction based on phenotypic data and a mixed linear model in factorials Co1 and Co2 (Figure 1). In contrast, in factorials Ra1 and Wh1 GBLUP increased prediction accuracy from 0.39 to 0.44, and from 0.71 to 0.74 in comparison to GCA prediction. In all four factorials with available marker data, GBLUP resulted in comparable prediction accuracies as the GB-SE (Figure 1). From this, we conclude that a state-of-the-art GBLUP model is a perfectly suitable and efficient tool for predicting yield performance in sparse factorials with a high relevance of SCA. The GB-SE is a viable alternative to GBLUP, and can save the costs for genotyping the parent lines.

We expected that using marker data or a combination of marker data and parentage information with ML algorithms might further increase prediction accuracy in factorial Ra1, as genetic markers should contain much more detailed information on similarities between parent lines than parentage information alone, and might pick up non-additive effects and relationships between parent lines that are beyond the scope of the GBLUP model. ML algorithms should in theory be able to exploit also non-linear relationships and interactions between markers. We investigated four additional algorithms specifically tuned for marker data: XGB, RKHS, SVM and an XGB-SE (Figure 2). XGB and the XGB-SE resulted in the highest prediction accuracies of0.44 and 0.45, underlining again the superiority of GB machines for hybrid prediction in sparse unbalanced factorials.

Xu et al. (2021) have shown that combining phenotypic data of the parent lines with *omics* data improves the prediction accuracy. Similarly, other studies have combined genomic and pedigree information to outperform GBLUP (Sood et al., 2020). We therefore also considered a combined set of input variables including both marker data and parentage information for Ra1 and the best ML algorithms, XGB and XGB-SE. The prediction accuracies of the algorithms XGB-C and XGB-SE improved only slightly to 0.46 for both the single model and the ensemble (Figure 2). We therefore assume that with the investigated factorials, the ML algorithms mainly exploit the same information, i.e. the genetic similarity of the parents, and little additional information is gained when combining parentage information with marker data.

Other recent studies have observed that ML algorithms sometimes outperform the classical methods of genomic prediction based on mixed-model equations, but with an overall high variance in prediction accuracy across different traits and data sets (Azodi et al., 2019; Montesinos-López et al., 2019). Among the many ML algorithms available, those based on decision trees show promising results in many scenarios. Banerjee et al. (2020) found that tree-based algorithms such as RF, GB and XGB outperform classical genomic prediction algorithms. Westhues et al. (2021) used GB and XGB to

improve yield prediction accuracy for genotypes across locations and years. Abdollahi-Arpanahi et al. (2020) also found that GB performs well when non-additive effects are important. Liang et al. (2021) achieved higher accuracies for three different animal data sets combining three different algorithms into an SE. Azodi et al. (2019) benchmarked several different algorithms on six different data sets with several traits each and also found that algorithm performance varied between data sets but an ensemble of different algorithms performed consistently equal or better than classical linear approaches. Yan et al. (2021) advocate the use of LightGBM, which is a more efficient type of GB due to building trees leave-wise instead of depth-wise. In their study, LightGBM also outperformed ridge regression BLUP for soybean and rice data. These results are in accordance with our finding that GB-SEs are more successful than single ML algorithms. However, in our study the use of XGB and an XGB-SE with genetic markers considerably increased computation time, but did not improve prediction accuracy compared to the GB-SE with parentage information (compare Figure 2 with Figure 1). It is possible that the factorials were too small for effective pattern recognition in marker data, as the number of predictors was far higher than the number of hybrids. In factorial Ra1, 10880 markers were available after pre-processing for only 756 hybrids. In the case of comparatively small sparse factorials the positive effects of more detailed genetic information could be counter-balanced by increased dimensionality and noise, resulting in overfitting and spurious associations.

4.6 Hyperparameter optimization with random grid search

Selecting the best possible set of hyperparameters is crucial for maximizing prediction accuracy for a given data set. We used random grid search to test different hyperparameter combinations over a large hyperparameter space. In contrast to cartesian grid search, which tests all possible combinations of different values for each hyperparameter, the random grid search samples uniformly from the set of all possible hyperparameter combinations and stops when a user-specified criterion, e.g. a fixed number of models, is met. In most cases, random grid search performs as well as cartesian grid search while considerably reducing the computation time (Bergstra and Bengio, 2012). The frequency of how often a model was chosen as best or worst in a grid search for GB with parentage information is shown in Figure 3. For most of the investigated factorials, the best and worst grid search models accumulated on one or very few models. When considering the information given in Table S2, we can see that in the case where a few models have equal share in being the best, these models are similar in their hyperparameters. For example, the best combinations for factorial Ra2 all share the same value for the number of bins and an overall small sample rate. When only one or very few similar hyperparameter combinations repeatedly performed best in all cross validation splits of a given factorial, we assumed that these were very close to a theoretical 'optimum' hyperparameter combination for this scenario. If the selected hyperparameter combinations of the best models were far away from the optimum, we should observe more variation in the selected sets of hyperparameters across cross validation splits. We therefore conclude that the random grid search approach is a user-friendly, efficient and consistent approach for the identification of a suitable set of hyperparameters for all factorials investigated in the present study. We expect it to perform well in a wide range of crops and scenarios.

The fifty tuned GBs from the random grid search also formed the basis for building the GB-SE. We observed that some GB-SEs used all 50 models of the grid search (Figure S10). Thus, models that perform badly on their own might add some value when used within an ensemble, and a reduction in the number of investigated models for the random grid search might impair prediction accuracy. More efficient grid search methods such as hyperband search (Li et al., 2018b) and Bayesian optimization (Snoek et al., 2012) exist but have not yet been implemented in the software we used. Implementations of these might reduce the required computation time or find even better hyperparameter combinations. Galli et al. (2022) used an automated model training process to choose the best of 50 neural networks. This approach is even easier to apply than random grid search, with the downside that the hyperparameter space is always predefined and cannot be modified. Liang et al. (2022) propose a 'tree-structured Parzen estimator' that automatically tunes the hyperparameters. However, random grid search also performed well in this study.

4.7 Importance of different hyperparameters for ML algorithms

We focused our investigation on the hyperparameters for GB, which was the most successful single ML algorithm, and formed the basis for building the GB-SE. The most important hyperparameters with the greatest effect on reducing the MSE in GB with parentage information were the number of bins for partitioning the data before determining the tree's best split point, and the depth of trees. Binning of factor levels for GB is also known as 'histogram-based GB'. Table 2 shows that for Ra1and Ra2 the number of bins is strongly correlated with the error. Setting a value for the number of bins that is smaller than the number of factor levels will group factor levels together into the specified number of groups (bins). For unordered nominal predictor variables as the parentage information investigated in the present study, these groups are somewhat arbitrary. Nevertheless, it seems that the number of bins has a large impact on the generalization error rate (Malohlava and Candel, 2022). As a tendency, reducing the number of bins for factors with high cardinality adds randomness to the splits in the decision trees, which seems to increase the generalizability of the model, while selecting a higher number of bins increases model fit to the training data and can lead to overfitting. When taking the ridge line plots in Figure S4 into account, it seems that the relationship between the number of bins and the error is not linear. Too many as well as too few bins both increase the error. Selecting a medium number of bins had a positive effect on the prediction accuracy, especially for data sets Ra1 and Ra2 with overall low prediction accuracy. This is also reflected in Table S2,

where we listed the hyperparameters for the best models investigated in the random grid search. As trees for GB always need to be trained sequentially, model training is slow in comparison to other models such as RF, for which trees can be trained in parallel. As a desirable side effect, binning will considerably speed up model training. We therefore recommend tuning of the number of bins if many parents, i.e. many factor levels, have to be included in the model in order to increase efficiency and performance of GB. It remains to be investigated if ordered binning, e.g. by clustering genetically similar parents together, can further increase prediction accuracy.

With respect to the depth of the decision trees, we observed that the factorials Ra1-3 required deeper trees than the factorials Wh1, Co1 and Co2 (Table S2). Usually the depth of the trees is related to the complexity of the task. We observed high accuracies of GCA effects for factorials Wh1, Co1 and Co2, indicating that hybrid performance can be accurately represented by the additive parent GCA effects. Consequently, fewer splits per tree are also needed in GB. For the rapeseed factorials Ra1 and Ra2 with a high relevance of SCA for hybrid yield, deeper trees might be able to capture non-additive SCA effects. We saw little room for reducing the computation time of the random grid search by a more limited search space when initially looking for the best model for the investigated factorials, since hyperparameters from the upper and the lower end of our search space were considered best in some cases.

4.8 Implementation of machine learning algorithms in hybrid breeding programs

The required know-how and computational power is a major obstacle for the implementation of more complex ML algorithms in hybrid breeding programs. In particular, feature selection, model implementation in languages such as Python or Julia, the tuning of hyperparameters and the composition of ensembles of models are tasks which may seem daunting for breeders without solid background in data science. Our study has shown that successful prediction of hybrid yield does not necessarily rely on very large datasets and expert-designed models. Here, we present a fairly automated random grid search approach for building GB-SEs with the parentage information as a predictor, implemented in a user-friendly software package with an R interface. The well-documented Rcode for our procedure is available in the supplementary material, and can be tested with the publicly available experimental data sets investigated in our study. Many breeders are already familiar with analysis of field trials and genomic selection in R, and can thus easily adapt the code for their own breeding programs and purposes.

We expect that ML algorithms will only be widely implemented in practical breeding programs if they offer advantages over well-established prediction models such as GCA prediction and GBLUP in real-life data sets from ongoing breeding programs. In the present study, we observed potential for ML algorithms in comparatively small, sparse unbalanced factorials with high relevance of SCA effects. The GB-based SE with parentage information increased prediction accuracy considerably in these specific factorials in comparison to classical GCA prediction, and resulted in equivalent prediction accuracies in comparison to GBLUP for all other investigated factorials. Moreover, the random grid search approach for tuning the basic GBs delivered consistent sets of hyperparameters across cross-validation splits in reasonable computation time. In comparison to GBLUP and other marker-based approaches, the simple use of parents as predictors can also save the cost of genotyping. We therefore expect that our suggested procedure is applicable in wide range of crops and breeding programs, and can be considered as an alternative to GBLUP.

For the present study, we decided to test the ML algorithms in crops of major commercial importance, because high-quality genotypic and phenotypic data was easily accessible, and we were able to compare long-established hybrid breeding programs with high relevance of GCA with newly established programs with high relevance of SCA. However, we do not expect that large hybrid breeding programs with well-established genomic selection pipelines represent the main field of application for our method. A recent review on hybrid breeding from the perspective of commercial breeders has pointed out that breeding targets shift and change over time, and that older cultivars and selection candidates form an important secondary breeding pool that might contain useful variation for new traits of interest (Steeg et al., 2022). One potential application of the GB-SE based on parentage information might be to pre-screen older field trials without *ad-hoc* available genotypic data and pre-select interesting candidates for such novel breeding targets for future testing. The same paper has also pointed out that on a world-wide level about 6000 plants are currently under cultivation. Only about 50 of those are currently bred as hybrids, exploiting hybrid vigor and other advantages for both commercial breeders and consumers. This suggests great future potential for establishment of hybrid breeding also in crops of minor commercial importance. New hybrid breeding programs have been established in crops such as guava, onion, eggplant, potato, triticale etc. Such newly set-up hybrid programs are often characterized by a high relevance of SCA. Many of these minor crops with high relevance for food diversity are bred in small breeding programs with very low budget, restricted number of field plots, and very limited staff. Not for all of these breeding programs high-throughput genotyping is readily available at the moment. For these breeding programs, the GB-SE with parentage information could provide a viable short-term alternative to genomic prediction.

Data availability statement

Publicly available datasets were analyzed in this study. The phenotypic and genotypic data of experiments Co1 and Co2 and the phenotypic data for experiment Wh1 have been published as Supplementary Materials to the research articles cited in the material and methods section. The genotypic data for experiment

Wh1 is available at https://datadryad.org/stash/dataset/doi:10.5061/dryad.461nc. Data sets Ra1-3 were provided by NPZ Innovation GmbH and are confidential.

Author contributions

EH and PH conceived the study. AA and TK provided the phenotypic and genotypic data of experiments Ra1-3. PH compiled the phenotypic and genotypic data of experiments Wh1, Co1 and Co2 from public data sources. PH pre-processed and analyzed the phenotypic and genotypic data, fitted the prediction models and conducted the predictions. PH and EH wrote the manuscript with further input from MF. All authors contributed to the article and approved the submitted version.

Funding

Funding for this study was provided by the German Federal Ministry of Education and Research (BMBF) grant 031B0890 (BreedPatH). The publication was funded by the Open Access Publication Fund of the Justus Liebig University Giessen.

Conflict of interest

Authors AA and TK were employed by the company NPZ Innovation GmbH.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpls.2023.1178902/full#supplementary-material

References

Abbas, Q., Ibrahim, M. E., and Jaffar, M. A. (2019). A comprehensive review of recent advances on deep vision systems. *Artif. Intell. Rev.* 52, 39–76. doi: 10.1007/s10462-018-9633-3

Abdollahi-Arpanahi, R., Gianola, D., and Peñagaricano, F. (2020). Deep learning versus parametric and ensemble methods for genomic prediction of complex phenotypes. *Genet. Selection Evol.* 52, 12. doi: 10.1186/s12711-020-00531-z

Albrecht, T., Wimmer, V., Auinger, H.-J., Erbe, M., Knaak, C., Ouzunova, M., et al. (2011). Genome-based prediction of testcross values in maize. *Theor. Appl. Genet.* 123, 339–350. doi: 10.1007/s00122-011-1587-7

Azodi, C. B., Bolger, E., McCarren, A., Roantree, M., de los Campos, G., and Shiu, S.-H. (2019). Benchmarking parametric and machine learning models for genomic prediction of complex traits. *G3: Genes Genomes Genet.* 9, 3691–3702. doi: 10.1534/g3.119.400498

Babaie Sarijaloo, F., Porta, M., Taslimi, B., and Pardalos, P. M. (2021). Yield performance estimation of corn hybrids using machine learning algorithms. *Artif. Intell. Agric.* 5, 82–89. doi: 10.1016/j.aiia.2021.05.001

Banerjee, R., Marathi, B., and Singh, M. (2020). Efficient genomic selection using ensemble learning and ensemble feature reduction. *J. Crop Sci. Biotechnol.* 23, 311–323. doi: 10.1007/s12892-020-00039-4

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *J. Stat. Software* 67, 1-48. doi: 10.18637/jss.v067.i01

Bergstra, J., and Bengio, Y. (2012). Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* 13, 281–305.

Bischl, B., Lang, M., Kotthoff, L., Schiffner, J., Richter, J., Studerus, E., et al. (2016). Mlr: machine learning in r. *J. Mach. Learn. Res.* 17, 1–5.

Bischl, B., Lang, M., and Schratz, P. (2020) *Parallelmap: unified interface to parallelization back-ends r package version 1.5.0*. Available at: https://cran.r-project.org/package=parallelMap.

Bishop, C. M. (2006). *Pattern recogniton and machine learning* (New York, USA: Springer-Verlag New York).

Breiman, L. (1996). Stacked regressions. *Mach. Learn.* 24, 49–64. doi: 10.1023/A:1018046112532

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1010933404324

Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O., and Walsh, A. (2018). Machine learning for molecular and materials science. *Nature* 559, 547–555. doi: 10.1038/s41586-018-0337-2

Chen, T., and Guestrin, C. (2016). "Xgboost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (New York, NY, USA: Association for Computing Machinery). doi: 10.1145/2939672.2939785

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., et al. (2022) *Xgboost: extreme gradient boosting r package version 1.6.0.1*. Available at: https://CRAN.R-project.org/package=xgboost.

Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20, 273–297. doi: 10.1007/bf00994018

Covarrubias-Pazaran, G. (2016). Genome assisted prediction of quantitative traits using the r package sommer. *PloS One* 11, 1–15. doi: 10.1371/journal.pone.0156744

Covarrubias-Pazaran, G. (2018). Software update: moving the r package sommer to multivariate mixed models for genome-assisted prediction. *biorxv*. 354639. doi: 10.1101/354639

Cuevas, J., Montesinos-López, O., Juliana, P., Guzmán, C., Pérez-Rodríguez, P., González-Bucio, J., et al. (2019). Deep kernel for genomic and near infrared predictions in multi-environment breeding trials. *G3: Genes Genomes Genet.* 9, 2913–2924. doi: 10.1534/g3.119.400493

Dargan, S., Kumar, M., Ayyagari, M. R., and Kumar, G. (2020). A survey of deep learning and its applications: a new paradigm to machine learning. *Arch. Comput. Methods Eng.* 27, 1071–1092. doi: 10.1007/s11831-019-09344-w

Domingos, P. (2012). A few useful things to know about machine learning. *Commun. ACM* 55, 78–87. doi: 10.1145/2347736.2347755

Endelman, J. B., and Jannink, J.-L. (2012). Shrinkage estimation of the realized relationship matrix. *G3: Genes Genomes Genet.* 2, 1405–1413. doi: 10.1534/g3.112.004259

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Ann. Stat* 29, 1189–1232. doi: 10.1214/aos/1013203451

Gabur, I., Simioniuc, D. P., Snowdon, R. J., and Cristea, D. (2022). Machine learning applied to the search for nonlinear features in breeding populations. *Front. Artif. Intell.* 5. doi: 10.3389/frai.2022.876578

Galli, G., Sabadin, F., Yassue, R. M., Galves, C., Carvalho, H. F., Crossa, J., et al. (2022). Automated machine learning: a case study of genomic "image-based" prediction in maize hybrids. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.845524

Gillberg, J., Marttinen, P., Mamitsuka, H., and Kaski, S. (2019). Modelling gxe with historical weather information improves genomic prediction in new environments. *Bioinformatics* 35, 4045–4052. doi: 10.1093/bioinformatics/btz197

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning* (Cambridge, MA, USA: MIT Press). Available at: http://www.deeplearningbook.org.

Gowda, M., Zhao, Y., Würschum, T., Longin, C. F., Miedaner, T., Ebmeyer, E., et al. (2014). Relatedness severely impacts accuracy of marker-assisted selection for disease resistance in hybrid wheat. *Heredity* 112, 552–561. doi: 10.1038/hdy.2013.139

Hallauer, A. R., Carena, M. J., and Miranda Filho, J. B. (2010). *Quantitative genetics in maize breeding* (New York, USA: Springer).

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning* (New York, USA: Springer). doi: 10.1007/978-0-387-84858-7

Hofheinz, N., Borchardt, D., Weissleder, K., and Frisch, M. (2012). Genome-based prediction of test cross performance in two subsequent breeding cycles. *Theor. Appl. Genet.* 125, 1639–1645. doi: 10.1007/s00122-012-1940-5

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning* (New York, USA: Springer). doi: 10.1007/978-1-4614-7138-7

Karatzoglou, A., Hornik, K., Smola, A., and Zeileis, A. (2004). Kernlab - an s4 package for kernel methods in r. *J. Stat. Software* 11, 1–20. doi: 10.18637/jss.v011.i09

Khaki, S., Khalilzadeh, Z., and Wang, L. (2020). Predicting yield performance of parents in plant breeding: a neural collaborative filtering approach. *PLoS One* 15, e0233382. doi: 10.1371/journal.pone.0233382

Khaki, S., and Wang, L. (2019). Crop yield prediction using deep neural networks. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.00621

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer* 42, 30–37. doi: 10.1109/MC.2009.263

LeDell, E., Gill, N., Aiello, S., Fu, A., Candel, A., Click, C., et al. (2020). *h2o: r interface for the "h2o" scalable machine learning platform r package version 3.32.0.3*.

Lenth, R. (2021) *Emmeans: estimated marginal means, aka least-squares means r package version 1.2.3*. Available at: https://cran.r-project.org/package=emmeans.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., and Talwalkar, A. (2018b). Hyperband: a novel bandit-based approach to hyperparameter optimization. *J. Mach. Learn. Res.* 18, 1–52. doi: 10.3389/fgene.2018.00237

Li, B., Zhang, N., Wang, Y.-G., George, A. W., Reverter, A., and Li, Y. (2018a). Genomic prediction of breeding values using a subset of snps identified by three machine learning methods. *Front. Genet.* 9. doi: 10.3389/fgene.2018.00237

Liang, M., An, B., Li, K., Du, L., Deng, T., Cao, S., et al. (2022). Improving genomic prediction with machine learning incorporating tpe for hyperparameters optimization. *Biology*. 11, 1647. doi: 10.3390/biology11111647

Liang, M., Chang, T., An, B., Duan, X., Du, L., Wang, X., et al. (2021). A stacking ensemble learning framework for genomic prediction. *Front. Genet.* 12. doi: 10.3389/fgene.2021.600040

Ma, W., Qiu, Z., Song, J., Li, J., Cheng, Q., Zhai, J., et al. (2018). A deep convolutional neural network approach for predicting phenotypes from genotypes. *Planta* 248, 1307–1318. doi: 10.1007/s00425-018-2976-9

Malohlava, M., and Candel, A. (2022). *Gradient boosting machine with h2o* (Mountain View, CA, USA: H2O.ai Inc). Available at: https://docs.h2o.ai/h2o/latest-stable/h2o-docs/booklets/GBMBooklet.pdf.

Melchinger, A. E., and Gumber, R. K. (1998). "Chap. 3," in *Overview of heterosis and heterotic groups in agronomic crops* (Madison, WI, USA: John Wiley & Sons, Ltd), 29–44.

Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7. doi: 10.3389/fpls.2016.01419

Montesinos-López, O. A., Martín-Vallejo, J., Crossa, J., Gianola, D., Hernández-Suárez, C. M., Montesinos-López, A., et al. (2019). A benchmarking between deep learning, support vector machine and bayesian threshold best linear unbiased prediction for predicting ordinal traits in plant breeding. *G3: Genes Genomes Genet.* 9, 601–618. doi: 10.1534/g3.118.200998

Montesinos-López, O. A., Montesinos-López, A., Crossa, J., Gianola, D., Hernández-Suárez, C. M., and Martín-Vallejo, J. (2018a). Multi-trait, multi-environment deep learning modeling for genomic-enabled prediction of plant traits. *G3: Genes Genomes Genet.* 8, 3829–3840. doi: 10.1534/g3.118.200728

Montesinos-López, O. A., Montesinos-López, A., Crossa, J., Montesinos-López, J. C., Mota-Sanchez, D., Estrada-González, F., et al. (2018b). Prediction of multiple-trait and multiple-environment genomic data using recommender systems. *G3: Genes Genomes Genet.* 8, 131–147. doi: 10.1534/g3.117.300309

Montesinos-López, O. A., Montesinos-López, A., Pérez-Rodríguez, P., Barrón-López, J. A., Martini, J. W., Fajardo-Flores, S. B., et al. (2021). A review of deep learning applications for genomic selection. *BMC Genomics* 22, 19. doi: 10.1186/s12864-020-07319-x

Nagasubramanian, K., Jones, S., Sarkar, S., Singh, A. K., Singh, A., and Ganapathysubramanian, B. (2018). Hyperspectral band selection using genetic algorithm and support vector machines for early identification of charcoal rot disease in soybean stems. *Plant Methods* 14, 86. doi: 10.1186/s13007-018-0349-9

Nagasubramanian, K., Jones, S., Singh, A. K., Sarkar, S., Singh, A., and Ganapathysubramanian, B. (2019). Plant disease identification using explainable 3d deep learning on hyperspectral images. *Plant Methods* 15, 98. doi: 10.1186/s13007-019-0479-8

Perez, P., and de los Campos, G. (2014). Genome-wide regression and prediction with the bglr statistical package. *Genetics* 198, 483–495. doi: 10.1534/genetics.114.164442

Philipp, N., Liu, G., Zhao, Y., He, S., Spiller, M., Stiewe, G., et al. (2016). Genomic prediction of barley hybrid performance. *Plant Genome* 9, plantgenome 2016–02. doi: 10.3835/plantgenome2016.02.0016

Probst, P., Boulesteix, A.-L., and Bischl, B. (2019). Tunability: importance of hyperparameters of machine learning algorithms. *J. Mach. Learn. Res.* 20, 1–32.

Qiu, Y., Cortes, D., Lin, C.-J., Juan, Y.-C., Chin, W.-S., Zhuang, Y., et al. (2021). *Recosystem: recommender system using matrix factorization*.

R Core Team (2022). *R: a language and environment for statistical computing* (Vienna, Austria: R Foundation for Statistical Computing).

Schrag, T. A., Westhues, M., Schipprack, W., Seifert, F., Thiemann, A., Scholten, S., et al. (2018). Beyond genomic prediction: combining different types of omics data can improve prediction of hybrid performance in maize. *Genetics* 208, 1373–1385. doi: 10.1534/genetics.117.300374

Schulthess, A. W., Zhao, Y., and Reif, J. C. (2017). *Genomic selection in hybrid breeding* (Cham: Springer International Publishing), 149–183.

Shahhosseini, M., Hu, G., Huber, I., and Archontoulis, S. V. (2021). Coupling machine learning and crop modeling improves crop yield prediction in the us corn belt. *Sci. Rep.* 11, 1–15. doi: 10.1038/s41598-020-80820-1

Shwartz-Ziv, R., and Armon, A. (2022). Tabular data: deep learning is not all you need. *Inf. Fusion* 81, 84–90. doi: 10.1016/j.inffus.2021.11.011

Snoek, J., Larochelle, H., and Adams, R. P. (2012). "Practical bayesian optimization of machine learning algorithms," in *Advances in neural information processing systems*, F. Pereira and C.J. Burges and L. Bottou and K.Q. Weinberger (Red Hook, NY, USA: Curran Associates Inc), 25, 2951–2959. Available at: https://proceedings.neurips.cc/paper_files/paper/2012/file/05311655a15b75fab86956663e1819cd-Paper.pdf.

Sood, S., Lin, Z., Caruana, B., Slater, A. T., and Daetwyler, H. D. (2020). Making the most of all data: combining non-genotyped and genotyped potato individuals with hblup. *Plant Genome* 13, e20056. doi: 10.1002/tpg2.20056

Stahl, A., Pfeifer, M., Frisch, M., Wittkop, B., and Snowdon, R. J. (2017). Recent genetic gains in nitrogen use efficiency in oilseed rape. *Front. Plant Sci.* 8, 963. doi: 10.3389/fpls.2017.00963

Steeg, E., Struik, P., Visser, R., and Lindhout, P. (2022). Crucial factors for the feasibility of commercial hybrid breeding in food crops. *Nat. Plants* 8, 1–11. doi: 10.1038/s41477-022-01142-w

Stuber, C. W., and Cockerham, C. C. (1966). Gene effects and variances in hybrid populations. *Genetics* 54, 1279–1286. doi: 10.1093/genetics/54.6.1279

Technow, F., Riedelsheimer, C., Schrag, T. A., and Melchinger, A. E. (2012). Genomic prediction of hybrid performance in maize with models incorporating dominance and population specific marker effects. *Theor. Appl. Genet.* 125, 1181–1194. doi: 10.1007/s00122-012-1905-8

Technow, F., Schrag, T. A., Schipprack, W., Bauer, E., Simianer, H., and Melchinger, A. E. (2014). Genome properties and prospects of genomic prediction of hybrid performance in a breeding program of maize. *Genetics* 197, 1343–1355. doi: 10.1534/genetics.114.165860

Van Der Laan, M. J., Polley, E. C., and Hubbard, A. E. (2007). "Super learner," in *Statistical applications in genetics and molecular biology*. 6, 25. doi: 10.2202/1544-6115.1309

Washburn, J. D., Cimen, E., Ramstein, G., Reeves, T., O'Briant, P., McLean, G., et al. (2021). Predicting phenotypes from genetic, environment, management, and historical data using cnns. *Theor. Appl. Genet.* 134, 3997–4011. doi: 10.1007/s00122-021-03943-7

Westhues, C. C., Mahone, G. S., da Silva, S., Thorwarth, P., Schmidt, M., Richter, J. C., et al. (2021). Prediction of maize phenotypic traits with genomic and environmental predictors using gradient boosting frameworks. *Front. Plant Sci.* 12. doi: 10.3389/fpls.2021.699589

Xu, Y., Zhao, Y., Wang, X., Ma, Y., Li, P., Yang, Z., et al. (2021). Incorporation of parental phenotypic data into multi-omic models improves prediction of yield-related traits in hybrid rice. *Plant Biotechnol. J.* 19, 261–272. doi: 10.1111/pbi.13458

Yan, J., Xu, Y., Cheng, Q., Jiang, S., Wang, Q., Xiao, Y., et al. (2021). Lightgbm: accelerated genomically designed crop breeding through ensemble learning. *Genome Biol.* 22, 271. doi: 10.1186/s13059-021-02492-y

Zenke-Philippi, C., Frisch, M., Thiemann, A., Seifert, F., Schrag, T., Melchinger, A. E., et al. (2017). Transcriptome-based prediction of hybrid performance with unbalanced data from a maize breeding programme. *Plant Breed.* 136, 331–337. doi: 10.1111/pbr.12482

Zenke-Philippi, C., Thiemann, A., Seifert, F., Schrag, T., Melchinger, A. E., Scholten, S., et al. (2016). Prediction of hybrid performance in maize with a ridge regression model employed to dna markers and mrna transcription profiles. *BMC Genomics* 17, 262. doi: 10.1186/s12864-016-2580-y

Zhao, Y., Li, Z., Liu, G., Jiang, Y., Maurer, H. P., Würschum, T., et al. (2015). Genome-based establishment of a high-yielding heterotic pattern for hybrid wheat breeding. *Proc. Natl. Acad. Sci. U.S.A.* 112, 15624–15629. doi: 10.1073/pnas.1514547112

Zhao, Y., Thorwarth, P., Jiang, Y., Philipp, N., Schulthess, A. W., Gils, M., et al. (2021). Unlocking big data doubled the accuracy in predicting the grain yield in hybrid wheat. *Sci. Adv.* 7, eabf9106. doi: 10.1126/sciadv.abf9106

frontiers

Supplementary Material: Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP

1 SUPPLEMENTARY TABLES AND FIGURES

1.1 Figures

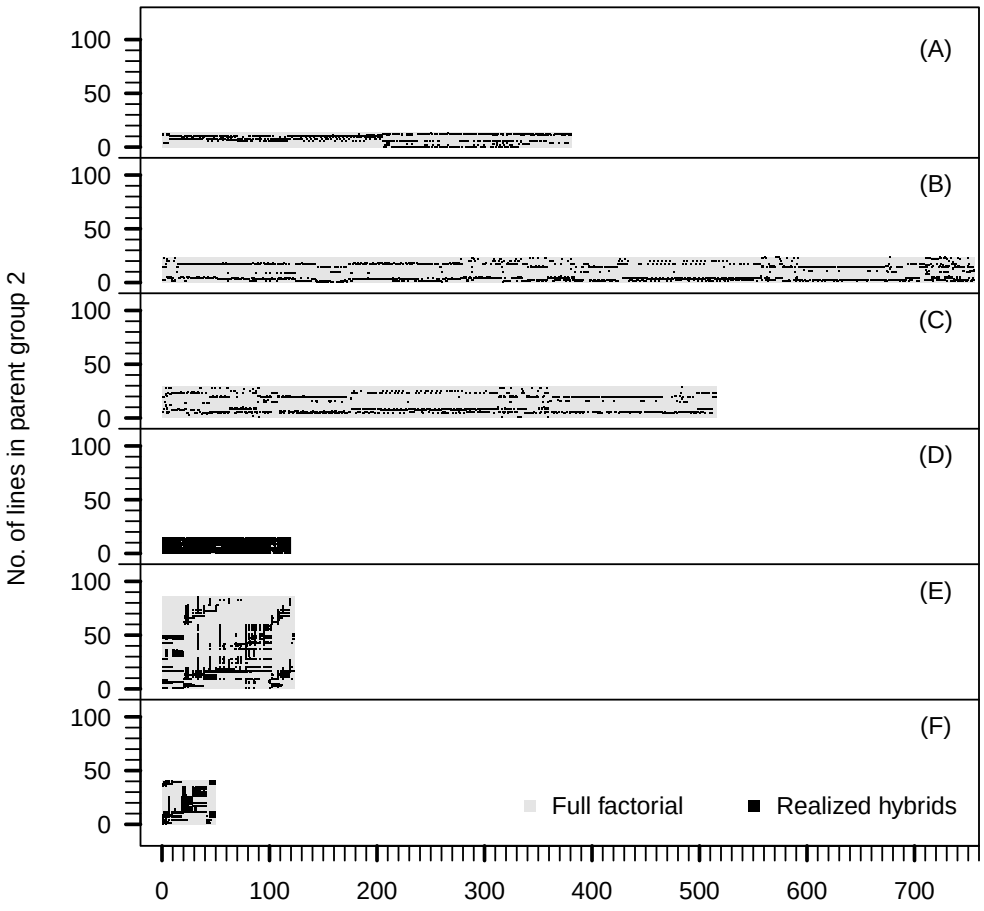

Figure S1. Crossing matrices for every experimental dataset indicating the size of the parent groups as well as the number of realized hybrid combinations. Black tiles represent realized hybrid crosses. The factorials are displayed on the same scales to highlight the differences between datasets with regard to size, unbalancedness in size of parent groups and sparsity. Datasets: **(A)**: Ra1, 14 × 381; **(B)**: Ra2, 24 × 756; **(C)**: Ra3, 29 × 516; **(D)**: Wh1, 15 × 120; **(E)**: Co1, 86 × 123; **(F)**: Co2, 50 × 41.

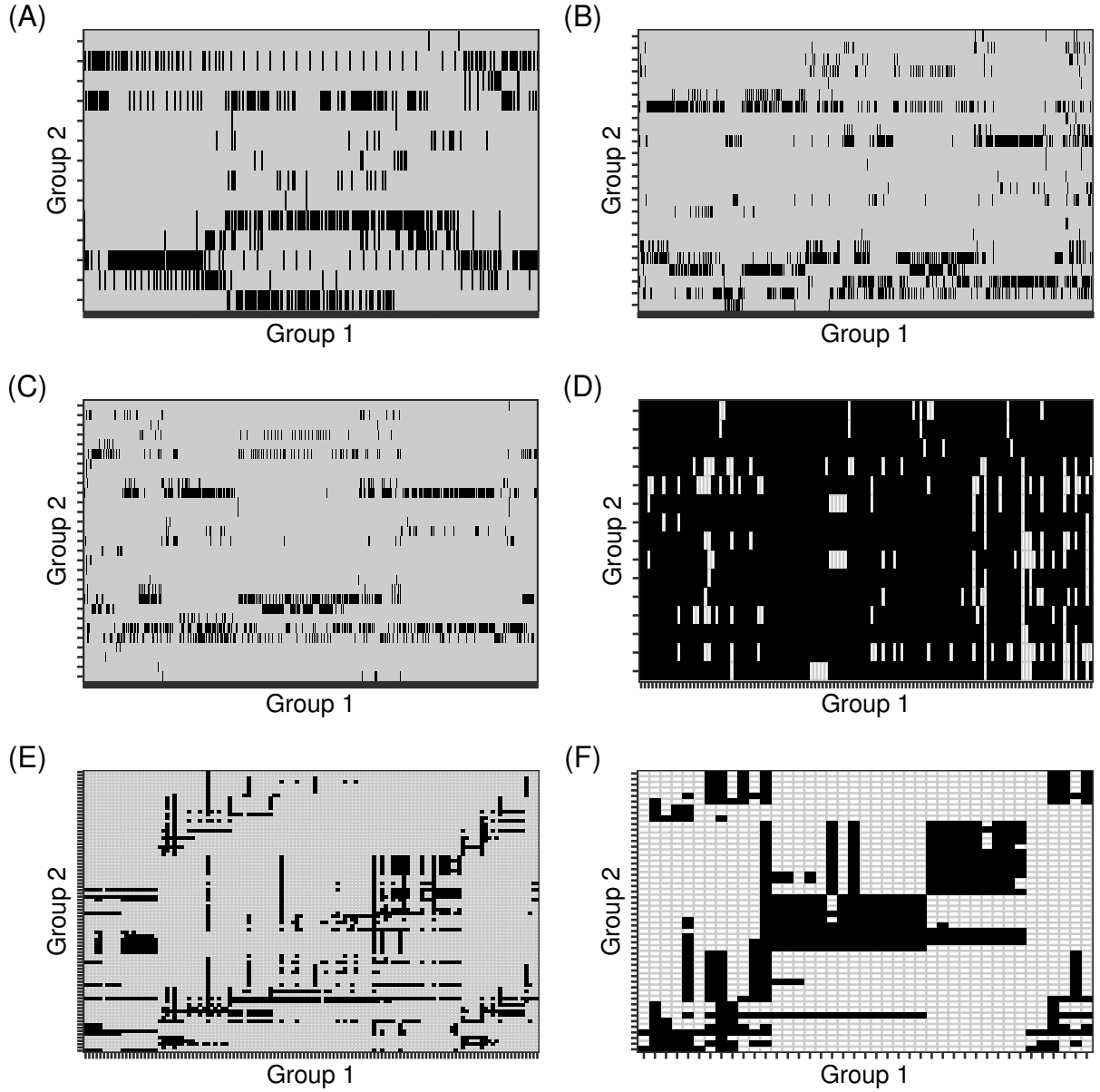

Figure S2. Crossing matrices of the individual datasets indicating the sparsity/completeness of the factorial. Black tiles represent realized crosses. Tick marks on both axes represent the lines from the respective group. Enlarged representation of Figure S1. Datasets: **(A)**: Ra1, 14 × 381; **(B)**: Ra2, 24 × 756; **(C)**: Ra3, 29 × 516; **(D)**: Wh1, 15 × 120; **(E)**: Co1, 86 × 123; **(F)**: Co2, 50 × 41.

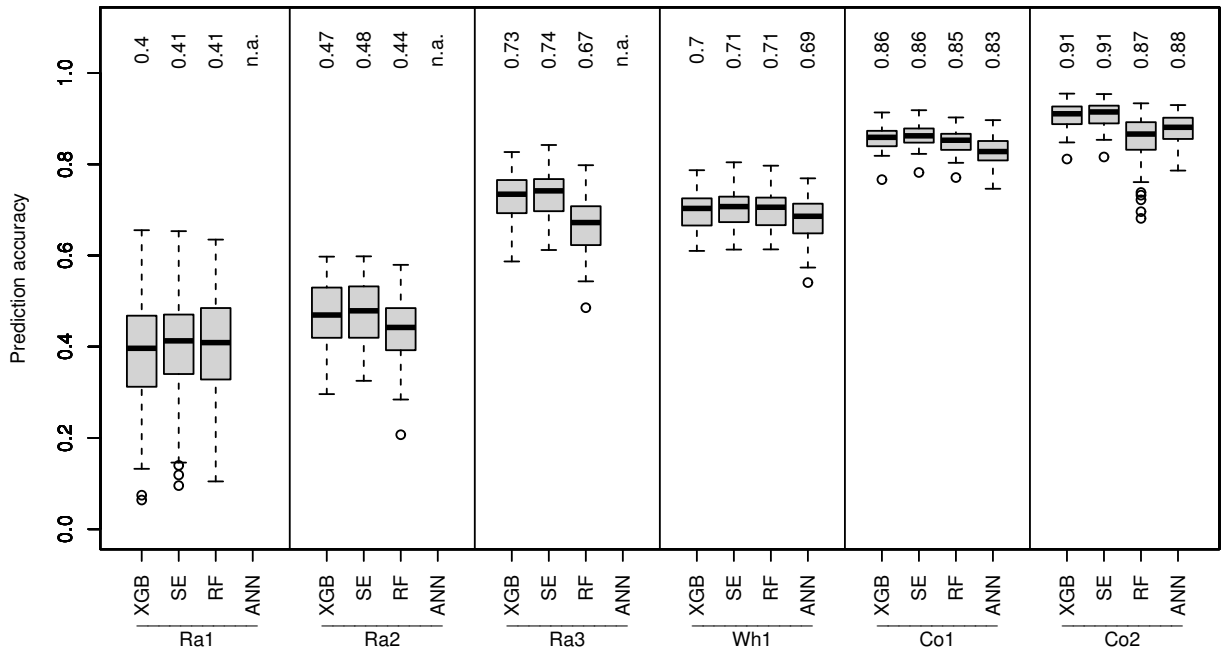

Figure S3. Boxplots of observed prediction accuracies for six different datasets (Ra1, Ra2, Ra3, Wh1, Co1, Co2) using four different methods (XGB, SE, RF, ANN) and yield of all other realized crosses of the parents of a specific hybrid as input features. Median prediction accuracy is displayed above each boxplot. As an alternative to the parentage information approach which uses parent names as input variables, we used the yield of all other realized crosses of the parents of a specific hybrid as the input features to predict its yield. This set of input features increased computation time, but never outperformed prediction with nominal parent factor levels. Results of ANN missing (n.a.) for datasets Ra1, Ra2 and Ra3 because models did not converge during training.

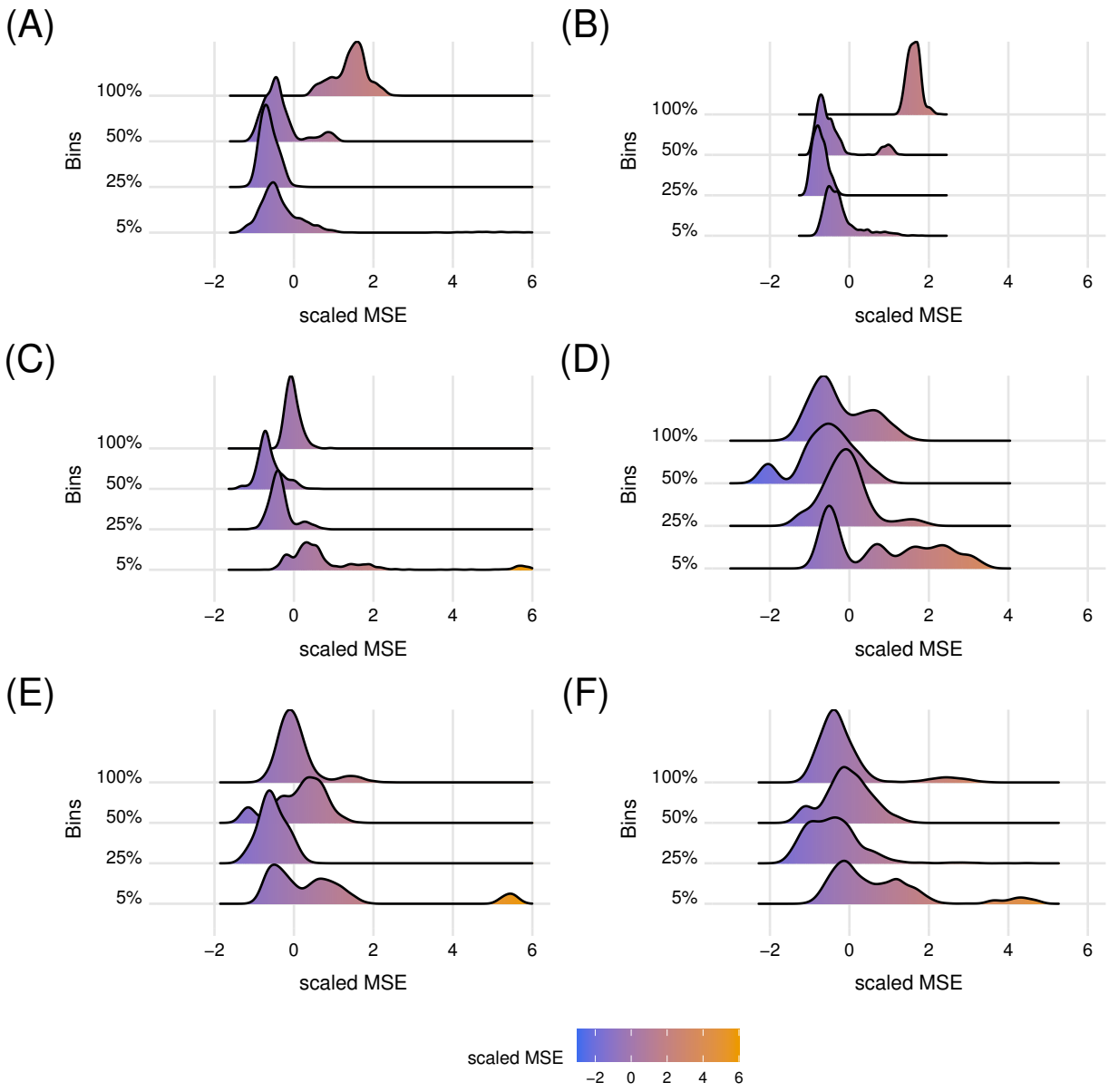

Figure S4. Distribution of the scaled mean squared error (MSE) per hyperparameter level of number of bins used for categories. Error of every grid search model was scaled by subtracting the mean and dividing it by the standard deviation of the respective cross validation split. Datasets: **(A)**: Ra1, **(B)**: Ra2, **(C)**: Ra3, **(D)**: Wh1, **(E)**: Co1, **(F)**: Co2

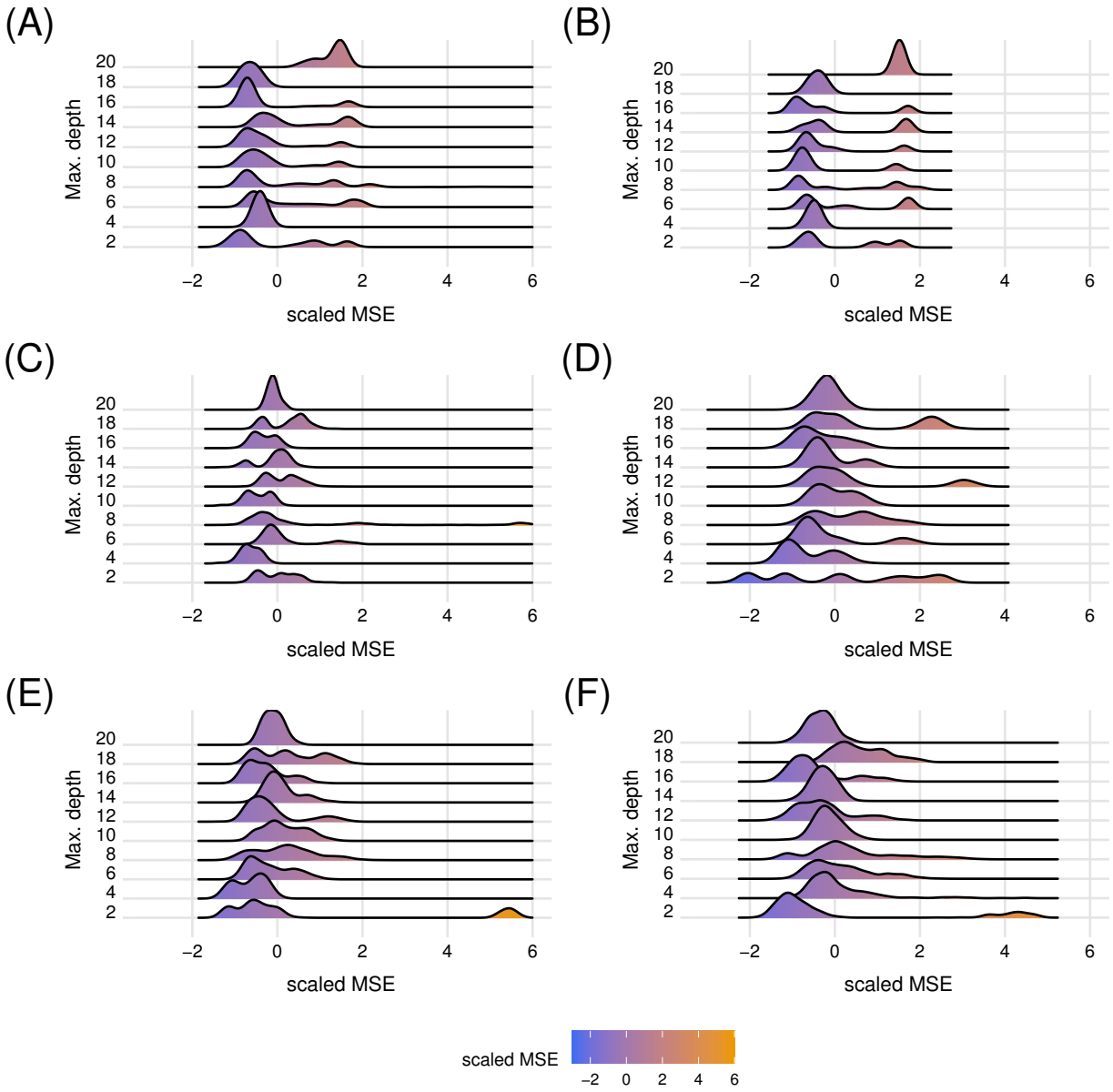

Figure S5. Distribution of the scaled mean squared error (MSE) per hyperparameter level of maximum depth of trees. Error of every grid search model was scaled by subtracting the mean and dividing it by the standard deviation of the respective cross validation split. Datasets: **(A)**: Ra1, **(B)**: Ra2, **(C)**: Ra3, **(D)**: Wh1, **(E)**: Co1, **(F)**: Co2

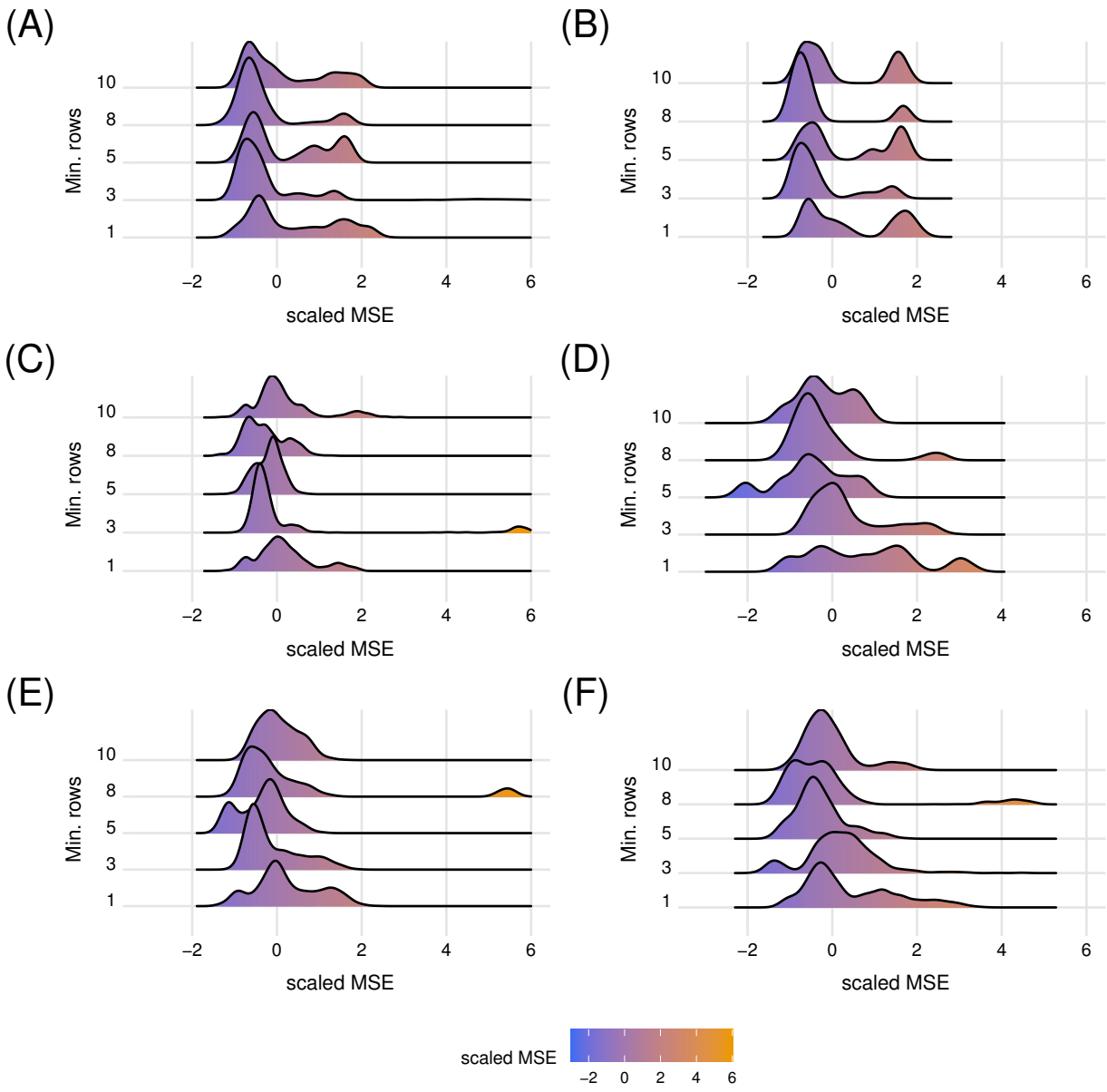

Figure S6. Distribution of the scaled mean squared error (MSE) per hyperparameter level of minimum number of rows required for further splitting. Error of every grid search model was scaled by subtracting the mean and dividing it by the standard deviation of the respective cross validation split. Datasets: **(A)**: Ra1, **(B)**: Ra2, **(C)**: Ra3, **(D)**: Wh1, **(E)**: Co1, **(F)**: Co2

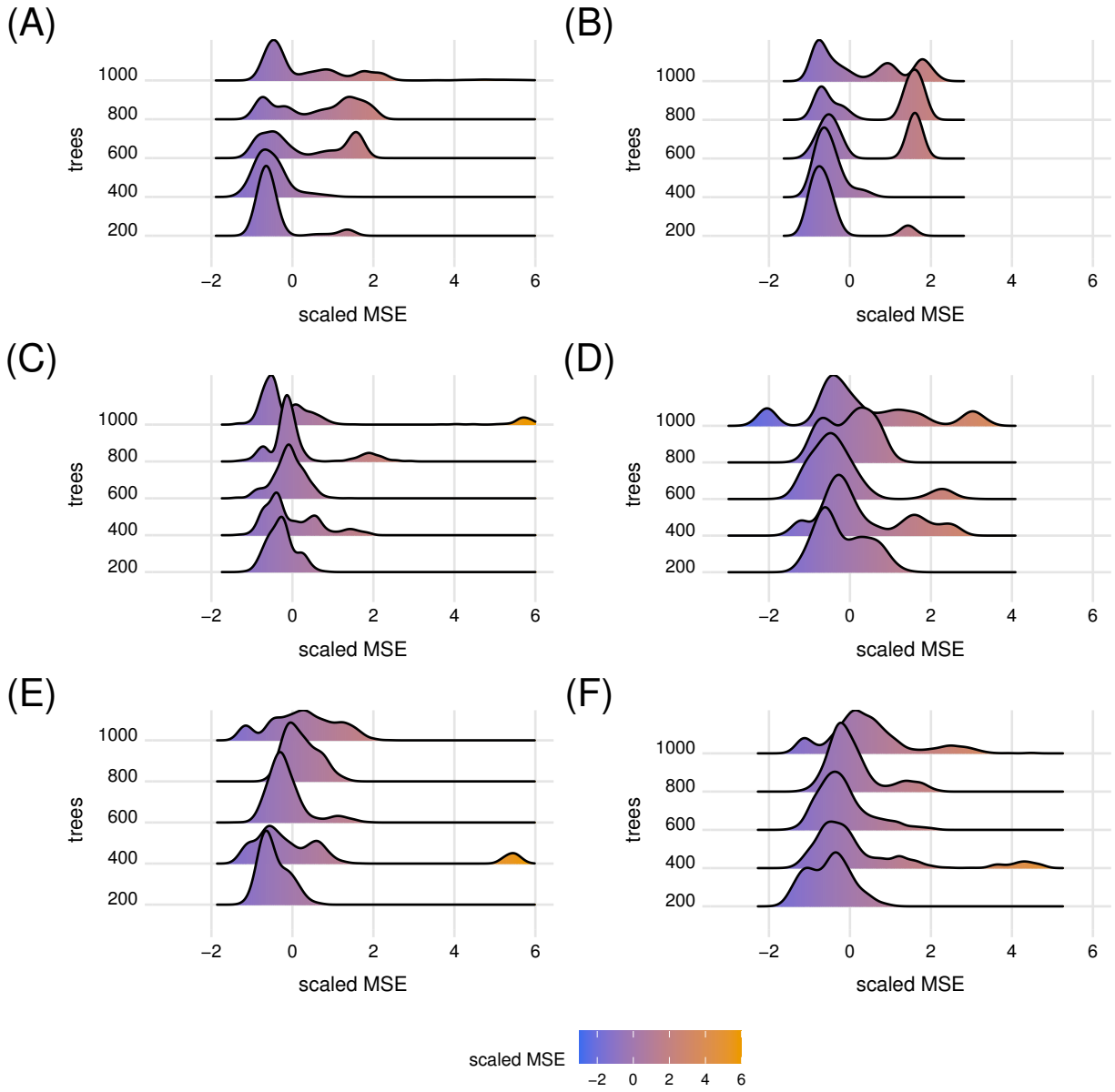

Figure S7. Distribution of the scaled mean squared error (MSE) per hyperparameter level of number of trees used to train a model. Error of every grid search model was scaled by subtracting the mean and dividing it by the standard deviation of the respective cross validation split. Datasets: **(A)**: Ra1, **(B)**: Ra2, **(C)**: Ra3, **(D)**: Wh1, **(E)**: Co1, **(F)**: Co2

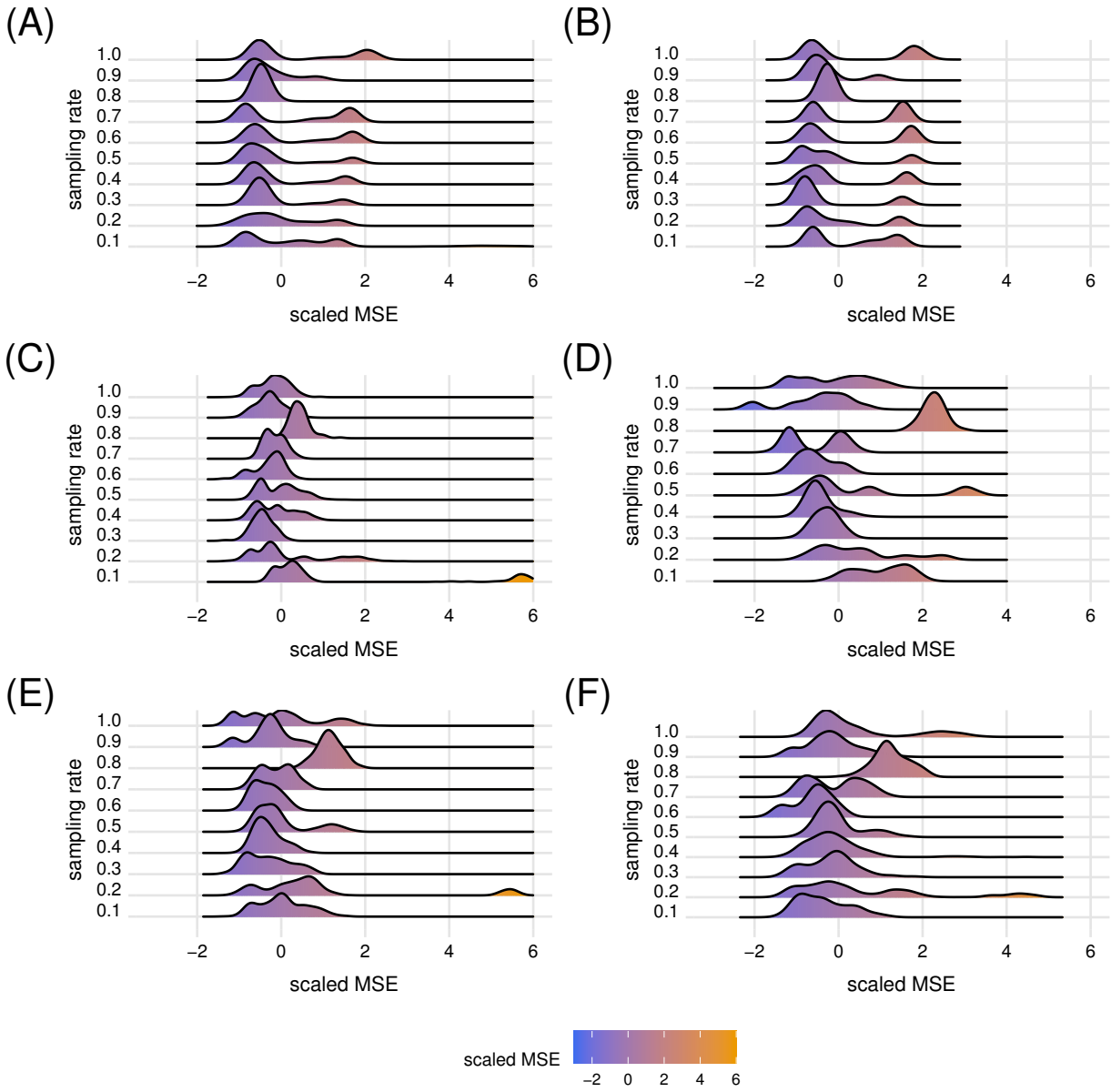

Figure S8. Distribution of the scaled mean squared error (MSE) per hyperparameter level of the row sampling rate. Error of every grid search model was scaled by subtracting the mean and dividing it by the standard deviation of the respective cross validation split. Datasets: **(A)**: Ra1, **(B)**: Ra2, **(C)**: Ra3, **(D)**: Wh1, **(E)**: Co1, **(F)**: Co2

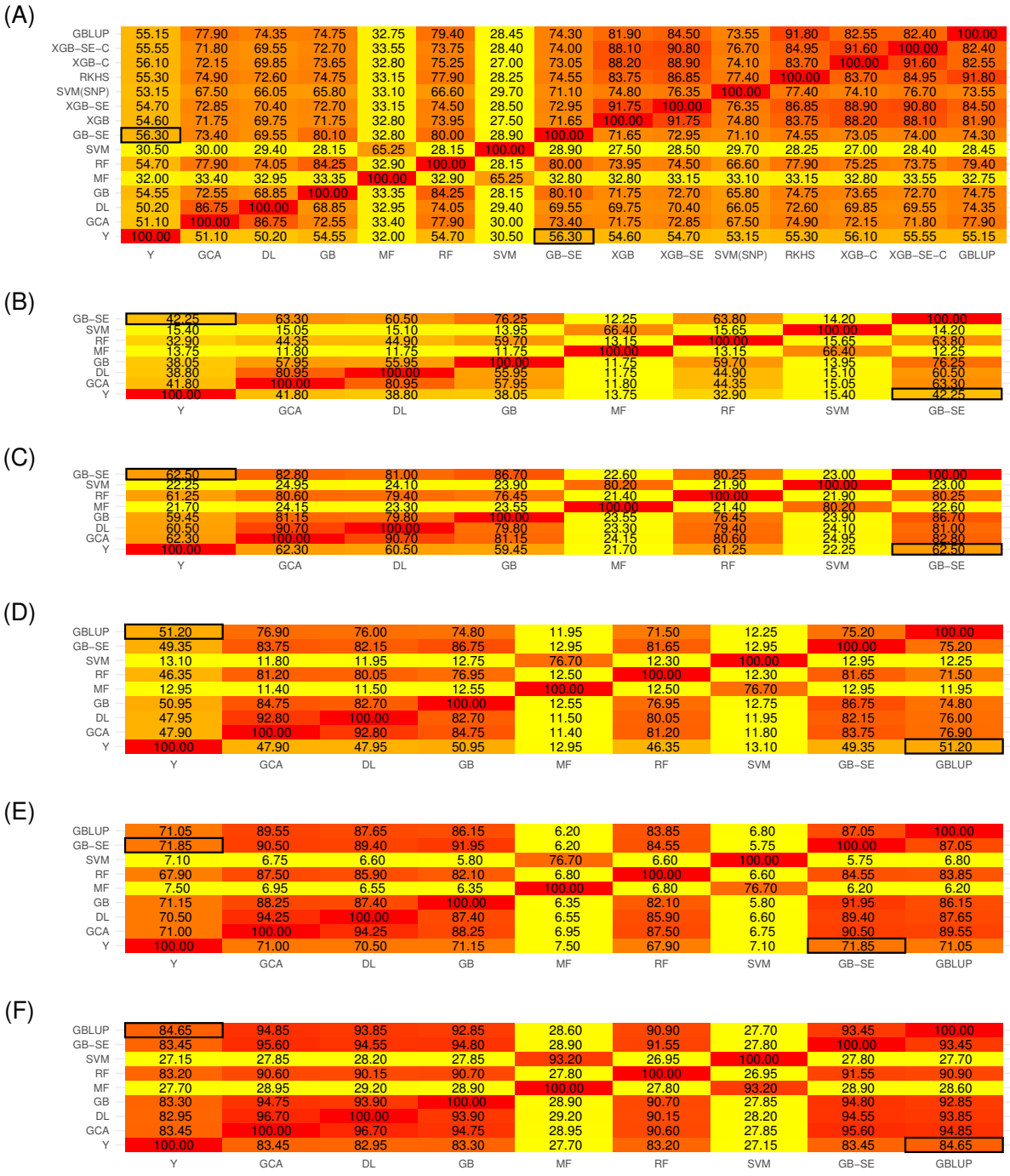

Figure S9. Heatmap indicating the overlap for the best 20 hybrids between two prediction methods. Numbers display the percentage of hybrids occuring in the top 20 of both methods according to predicted yield. 'Y' represents the true top 20 highest yielding hybrids. Black boxes indicating the highest overlap between a prediction method and the true top 20. Datasets: **(A)**: Ra1, **(B)**: Ra2, **(C)**: Ra3, **(D)**: Wh1, **(E)**: Co1, **(F)**: Co2

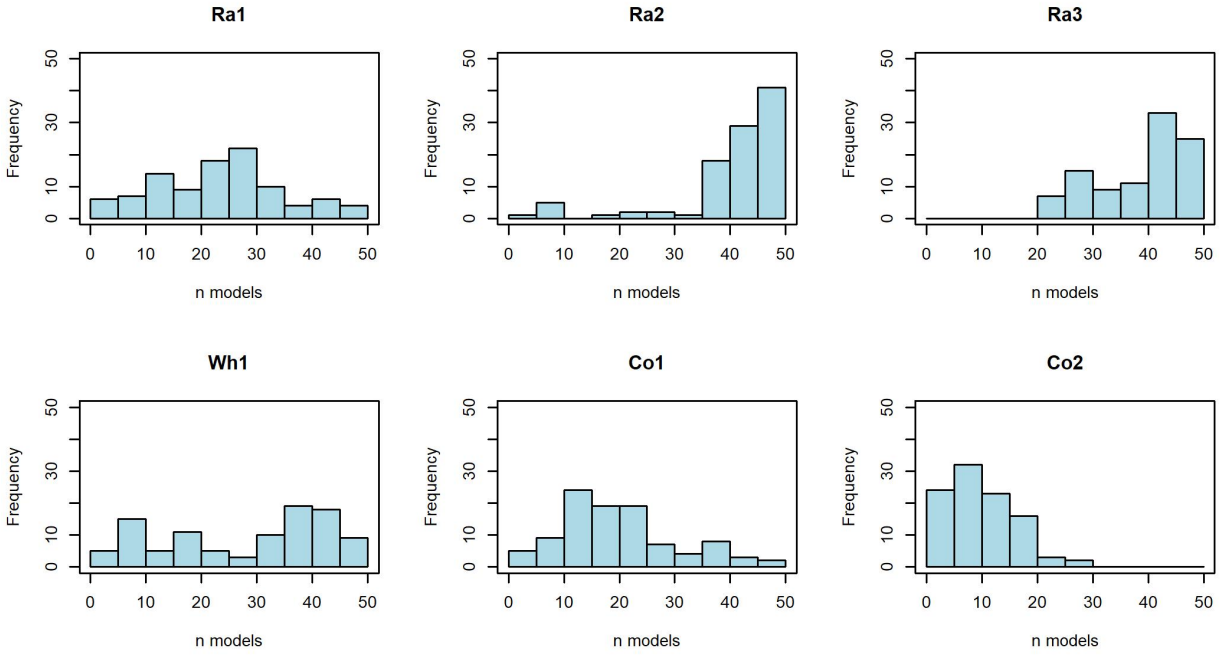

Figure S10. Optimum number of models to include in a GB-SE for each dataset. Every bar indicates the frequency of how often a certain number of models was considered the optimum number of models to be included in the final GB-SE.

1.2 Tables

Table S1. Overview of all possible hyperparameters considered in the grid search. Names of the listed hyperparameters correspond to the argument names used in h2o. A short description of each hyperparameter and their common names can be found in the materials and methods section of the paper.

Hyperparameter	Gridparameters
GBM & RF	
n_trees	$(200, 400, \ldots, 1000)$
max_depth	$(2, 4, \ldots, 20)$
min_rows	$(1, 3, 5, 8, 10)$
sample_rate	$(0.1, 0.2, \ldots, 1.0)$
nbins_cats	$(N_{pl} \cdot 0.05, N_{pl} \cdot 0.25, N_{pl} \cdot 0.5, N_{pl})$†
learn_rate ‡	0.1
XGB	
n_trees	(5000)
max_depth	$(2, 4, \ldots, 20)$
min_rows	$(1, 5, 10)$
sample_rate	$(0.2, 0.4, \ldots, 1.0)$
col_sample_rate_by_tree	$(0.2, 0.4, \ldots, 1.0)$
gamma	$(0, 5, 10, 20)$
learn_rate	$(0.01, 0.05, 0.1)$
nbins_cats	$(N_{pl} \cdot 0.05, N_{pl} \cdot 0.25, N_{pl} \cdot 0.5, N_{pl})$†
ANN	
hidden	$64, 128, (32, 32), (64, 32), (128, 64), (64, 64, 64)$
epochs	$(2, 4, 10, 20)$
input_dropout	$(0, 0.2)$
rate	$(0.001, 0.0005, 0.0001)$
SVM	
C	$(2^2, 2^3, \ldots, 2^9)$
epsilon	$(0.00, 0.01, \ldots, 0.20)$
kernel	linear, polynomial, radial
degree	$(2, 3, \ldots, 9)$
MF	
dim	$(4, 8, \ldots, 20)$
niter	500
costp_l1	$(0, 0.01)$
costq_l1	$(0, 0.01)$
lrate	0.05

† N_{pl} is the sum of all parental lines in the training set ‡ Only for GB

Table S2. GB hyperparameter combinations for each dataset with no. times best ≥ 5. A model was considered the best if it achieved the lowest error within a grid search. Only grid search runs with nominal parent information were considerd.

Dataset	Model No.	max_depth	min_rows	nbins_cats	ntrees	sample_rate	no. times best
Ra1	36	2	8	5%	400	0.2	80
	44	18	3	50%	600	0.7	6
	45	10	3	25%	600	0.5	5
Ra2	43	8	8	25%	200	0.2	26
	17	16	8	25%	200	0.4	24
	22	16	8	25%	200	0.3	23
	27	8	3	25%	400	0.5	10
	45	10	3.	25%	600	0.5	12
Ra3	5	4	8	50%	600	0.6	62
	46	14	10	50%	400	0.2	9
	19	10	8	50%	1000	0.3	8
	31	10	8	50%	800	0.2	8
	23	4	5	25%	400	1.0	6
Wh1	14	2	5	50%	1000	0.9	100
Co1	14	2	5	50%	1000	0.9	45
	23	4	5	25%	400	1.0	42
	9	4	1	25%	400	0.3	8
Co2	34	2	3	25%	200	0.6	71
	43	8	8	25%	200	0.2	11
	14	2	5	50%	1000	0.9	8

Table S3. Complete overview of the count of all best performing GB models for each dataset. Numbers indicate the times a model was considered the best, i.e. achieved the lowest mean square error within a grid search. Models without any case of being considered the best were removed from the table. Only grid search runs with nominal parent information were considered.

Model No.	Ra1	Ra2	Ra3	Wh1	Co1	Co2
5	0	0	62	0	0	0
7	0	0	3	0	0	0
9	0	0	0	0	8	0
10	1	0	0	0	0	0
11	0	0	0	0	0	1
14	0	0	0	100	45	8
15	0	1	0	0	0	0
16	1	0	0	0	1	3
17	1	24	0	0	1	4
19	0	1	8	0	0	0
20	0	1	0	0	0	0
22	0	23	0	0	0	2
23	0	0	6	0	42	0
24	0	0	4	0	0	0
26	1	0	0	0	0	0
27	1	10	0	0	0	0
31	0	0	8	0	0	0
34	1	2	0	0	0	71
36	80	0	0	0	0	0
42	1	0	0	0	0	0
43	0	26	0	0	3	11
44	6	0	0	0	0	0
45	5	12	0	0	0	0
46	0	0	9	0	0	0
48	1	0	0	0	0	0
50	1	0	0	0	0	0

1.3 Code Example

```
### Heilmann et al. 2023
# Example procedure to conduct a Gradient Boosting Machine grid search
# and form Stacked Ensembles on the basis of the results using h2o
# and a corn dataset from Technow et al. 2014

# Load required packages
library(sommer)
library(h2o)

# Start the h2o cluster
h2o.init()

# Function to restart after each iteration to prevent memory cluttering
restart <- function(nodes = 4){
  h2o.shutdown(F)
  h2o.init(nthreads = nodes)
  Sys.sleep(1)  # Sometimes needs some time to start up
}

# Load the dataset
data("DT_technow")

# Categorical variables are required to be of type 'factor'
DT_technow$dent   <- as.factor(DT_technow$dent)
DT_technow$flint  <- as.factor(DT_technow$flint)
DT_technow$hy     <- as.factor(DT_technow$hy)

### Hyperparameter search space for grid search
# All of these hyperparameter levels will be considered in the random selection
# of hyperparameter combinations used in the random grid search.
# Can be extended or modified

params <- list(ntrees      = seq(200,1000,200),     # Number of trees to include
               max_depth   = seq(2, 20, 2),         # Max. depth of each tree
               min_rows    = c(1,3,5,8,10),         # Min. rows required for split
               sample_rate = seq(0.1, 1.0, 0.1),    # Fraction of rows sampled per tree
               nbins_cats  = NA)                    # gets replaced later

search_criteria <- list(strategy   = "RandomDiscrete", # Set for random grid search
                        max_models = 10,               # Train 50 models
                        seed       = 2102)             # Seed for reproducibility

# Creates list 'idx.list' with indices for 100 random splits
# Split ratio was 90% training set to 10% test set
# Splits can be generated again by setting the seed and sending
# the command again. This is to ensure somewhat reproducible results.

seed.vec <- 2017
idx.list <- list()
set.seed(seed.vec)

# Each iteration creates a random split, stores it in a list
for (x in 1:100) {
  idx.list[[x]] <- sample(x    = 1:nrow(DT_technow),
                          size = round(nrow(DT_technow)*0.9))
}


# Seeds for the random processes within the ML algorithm
# Not to be confused with seed set for choosing
# the hyperparameters.
# Setting this seed does not provide 100% reproducibility
# but results are closer to each other
set.seed(123123)
lseed <- sample(1:10000, 100)
```

```
gbm_cors  <- c()     # To store gbm prediction accuracy
ens_cors  <- c()     # To store ensemble prediction accuracy
grid_list <- list()  # To store grid details

for (i in 1:100) {
  # Get the indicies for the training set in this iteration
  idx   <- idx.list[[i]]

  # Assign training set by index
  Train <- DT_technow[idx, ]

  # Hence, removing the training set indices leaves the test set
  Test  <- DT_technow[-idx, ]

  # Check if any parental line are contained in the test set but not the
  # training set. If such is the case, remove them from the training set
  # Both parental lines need to be available in the training set for GCA
  # to work

  if (!all(Test$dent %in% Train$dent)) {
    Test <- Test[Test$dent %in% Train$dent, ]
  }
  if (!all(Test$flint %in% Train$flint)) {
    Test <- Test[Test$flint %in% Train$flint, ]
  }

  # Dataframes need to be transformed into h2o.frames for further use
  Train.h2o <- as.h2o(Train[c("GY", "flint", "dent")])
  Test.h2o  <- as.h2o( Test[c("GY", "flint", "dent")])

  # We set the nbins_cats hyperparameter space according
  # to the number of factor levels, i.e. parental lines, in our
  # training set. Depending on the train/test split, this may change
  n_cat <- length(levels(Train$flint)) + length(levels(Train$dent))

  # Modify the hyperparameter search space
  # We use 100%, 50%, 25% or 5% of factor levels
  params$nbins_cats <- round(c(n_cat, n_cat * .5, n_cat * .25, n_cat * .05))

  # Run the grid search
  # Depending on the computational capacity and threads used, this may take a while
  # Using the standard metric 'mean residual deviance' is identical to MSE for this task
  gbm_grid <-
    h2o.grid( algorithm       =  "gbm",                  # Algorithm for grid search
              y               = "GY",                    # Target variable
              x               = c("flint", "dent"),      # Predictor variables
              training_frame  = Train.h2o,               # Data, must be in h2o format
              grid_id         = paste0("gbm_grid_", i),  # Name of the grid
              nfolds          = 10,                      # 10-fold CV
              seed            = lseed[i],                # seed for random procedures
              hyper_params    = params,                  # predefined hyperparameters
              search_criteria = search_criteria,         # predefined criteria
              keep_cross_validation_predictions = TRUE   # save CV - required for SE
    )

  # Get best grid search model
  best_model <- h2o.getModel(gbm_grid@model_ids[[1]])

  # Make predictions with best model, transform h2o.frane to dataframe
  gbm_preds <- as.data.frame(h2o.predict(best_model, Test.h2o))[, ]

  # Store accuracy in vector
  gbm_cors  <- c(gbm_cors, cor(Test$GY, gbm_preds, method = "pearson"))
```

```r
  # Visualize predicted compared to observed yield
  plot(
    Test$GY,
    gbm_preds,
    pch = 20,
    col = "black",
    xlim = c(min(Test$GY),   max(Test$GY)),
    ylim = c(min(gbm_preds), max(gbm_preds))
  )

  # Store grid details and ranking in a list
  grid_list[[i]] <- as.data.frame(gbm_grid@summary_table)


  ### This finds the optimum number of models for the
  ### Stacked Ensemble and predicts yield using the best

  # Create dataframe to store number of models
  # and prediction accuracy
  ensemble_eval <- data.frame()

  # Iterate over sequence of Top 5, 10, ..., 50 models
  for (x in seq(5,50,5)) {
    ensemble <-
      h2o.stackedEnsemble(
        y                    = "GY",                                # target
        x                    = c("flint", "dent"),                  # predictors
        metalearner_algorithm = "glm",                              # super learner
        metalearner_params    = list(lambda_search = T, alpha = 0), # Ridge Regression
        metalearner_nfolds    = 10,                                 # 10-fold CV
        training_frame        = Train.h2o,                          # in h2o format
        base_models           = c(unlist(gbm_grid@model_ids[1:x]))  # model names
      )

    # Save the r2 as a measure of prediction accuracy here
    r2 <- ensemble@model$metalearner_model@model$cross_validation_metrics@metrics$r2
    r2 < as.vector(r2)

    # Store no. of models included and accuracy in dataframe
    ensemble_eval = rbind(ensemble_eval,c(x,r2) )
  }

  # Choose no. of models with highest accuracy
  ens_model_num <- ensemble_eval[which.max(ensemble_eval[,2]),1]
  # n_models <- c(n_models,ens_model_num)

  # Build Stacked Ensemble with best no. of models
  ensemble <-
    h2o.stackedEnsemble(
      y                    = "GY",
      x                    = c("flint", "dent"),
      metalearner_algorithm = "glm",
      metalearner_params    = list(lambda_search = T, alpha = 0),
      metalearner_nfolds    = 10,
      training_frame        = Train.h2o,
      base_models           = c(unlist(gbm_grid@model_ids[1:ens_model_num]))
    )

  # Make predictions with best ensemble, transform h2o.frane to dataframe
  ens_preds <- as.data.frame(h2o.predict(ensemble, Test.h2o))[, ]

  # Store accuracy in vector
  ens_cors <- c(ens_cors, cor(Test$GY, ens_preds, method = "pearson"))
}
```

Chapter 3

Machine learning for prediction of resistance scores in wheat (*Triticum aestivum* L.)[1]

[1]P. G. Heilmann, Y. F. Difabachew, M. Frisch, A. L. Moritz, A. Stahl, B. Wittkop, R. J. Snowdon, M. Koch, M. Kirchhoff, L. Cselényi, M. Wolf, J. Förster, and C. Zenke-Philippi (2024) Machine learning for prediction of resistance scores in wheat (*Triticum aestivum* L.). *Plant Breeding* **144**:192–205.

Plant Breeding

WILEY

Plant Breeding

ORIGINAL ARTICLE OPEN ACCESS

Machine Learning for Prediction of Resistance Scores in Wheat (*Triticum aestivum* L.)

Philipp Georg Heilmann[1] | Yohannes Fekadu Difabachew[1] | Matthias Frisch[1] | Anna Luise Moritz[2] | Andreas Stahl[3] | Benjamin Wittkop[2] | Rod J. Snowdon[2] | Michael Koch[4] | Martin Kirchhoff[5] | László Cselényi[6] | Markus Wolf[7,8] | Jutta Förster[8] | Carola Zenke-Philippi[1]

[1]Institute of Agronomy and Plant Breeding II, Justus Liebig University, Gießen, Germany | [2]Institute of Agronomy and Plant Breeding I, Justus Liebig University, Gießen, Germany | [3]Institute for Resistance Research and Stress Tolerance, Julius Kühn Institute, Quedlinburg, Germany | [4]Deutsche Saatveredelung AG, Lippstadt, Germany | [5]Nordsaat Saatzucht GmbH, Langenstein, Germany | [6]W. von Borries-Eckendorf GmbH & Co. KG, Leopoldshöhe, Germany | [7]German Seed Alliance GmbH, Holtsee, Germany | [8]Saaten-Union Biotec GmbH, Leopoldshöhe, Germany

Correspondence: Carola Zenke-Philippi (biometry.popgen@uni-giessen.de)

Received: 7 February 2024 | **Revised:** 12 July 2024 | **Accepted:** 10 October 2024

Funding: This research was supported by the German Federal Ministry of Food and Agriculture, Grant/Award number: FKZ 2818403A18.

Keywords: cross-validation | genomic prediction | machine learning | wheat

ABSTRACT

Machine learning methods were shown to improve the prediction accuracies of genomic prediction of resistance scores compared to methods like RR-BLUP, which were originally designed for metric rather than ordinal response values. We conducted a cross-validation study with 361 wheat genotypes evaluated for five fungal diseases. Our objective was to compare the prediction accuracy and the ability to identify the most resistant genotypes of 19 genomic prediction approaches. Each approach consisted of a different combination of prediction method (RR-BLUP, an alternative method with heterogeneous marker variances, Bayesian generalized linear regression with an ordinal response, support vector machine, gradient boosting machine and random forest), predictor (single SNP markers, LD-based haplotype blocks, 250 variables generated with an autoencoder and SNPs identified with incremental feature selection) and response value (untransformed and logit-transformed resistance scores). In our dataset, RR-BLUP was consistently among the methods with the largest prediction accuracies and the best abilities to identify resistant genotypes in four of five investigated traits. However, in *P. triticina*, using gradient boosting machine and random forest instead of RR-BLUP increased the prediction accuracy from 0.64 to 0.71, indicating that machine learning methods may have an advantage over linear models in genomic prediction. We also found that even though there was a positive correlation between the prediction accuracy and Cohen's κ, a measure to judge how well the most resistant genotypes can be identified, the correlation is not perfect and a large value for the prediction accuracy does not necessarily translate into an equally large κ value.

1 | Introduction

In the last two decades, genomic prediction (Meuwissen, Hayes, and Goddard 2001), which aims at predicting the phenotypic value of an individual from its genotypic data, has increasingly replaced phenotypic selection. The advantage is that only a part of all the genotypes in the breeding population have to be phenotyped or, even better, that phenotypic data that are already

Abbreviations: BGLR, Bayesian generalized linear regression; GBLUP, genomic best linear unbiased prediction; GBM, gradient boosting machine; GWAS, genome-wide association study; LD, linkage disequilibrium; RF, random forest; RMLA, estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components; RMSE, square root of the mean square error; RR-BLUP, ridge-regression best linear unbiased prediction; SNP, single nucleotide polymorphism; SVM, support vector machine; SVR, support vector machine regression.

Philipp Georg Heilmann and Yohannes Fekadu Difabachew contributed equally to this work.

available can be used for the predictions. For the remaining genotypes, only marker data are needed. This is especially beneficial for the evaluation of resistance traits, which is time consuming and expensive. The genomic prediction approach has three components: (1) the form of the genotypic data that are used as predictors, (2) the type of the response values (metric values, percentages and values on an ordinal or nominal scale) and (3) the statistical model that links predictor and response. All of these components have an influence on the prediction accurary, defined as Pearson's correlation between the observed and predicted phenotypic values.

The last component, the statistical model, is the one that receives the most attention in studies on genomic prediction. Ridge regression best linear unbiased prediction (RR-BLUP) and Bayesian methods are among the standard methods for genomic prediction (Wang et al. 2018). While RR-BLUP assumes homogeneous marker variances, most Bayesian methods (Meuwissen, Hayes, and Goddard 2001) as well as another method called "estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components" (RMLA) (Hofheinz and Frisch 2014) allow for heterogeneous marker variances. This might be a better fit for the oligogenic nature of resistance traits because the effects of some markers may be large while those of most others may be close to zero (Hofheinz and Frisch 2014). RR-BLUP, methods from the Bayesian alphabet, and RMLA were developed for the prediction of metric response values with single SNP markers. A possible alternative for the prediction of ordinal response values is Bayesian generalized linear regression as implemented in the R package BGLR (Pérez and de los Campos 2014). More recently, machine learning methods such as support vector machine (SVM), gradient boosting machine (GBM) and random forest (RF) have been used for genomic prediction of resistance scores (Azodi et al. 2019; John et al. 2022; Jones et al. 2023; Ornella et al. 2012; Ornella et al. 2014; Tomar et al. 2021). Machine learning methods are non-parametric and can be applied to metric or ordinal response values without any assumption on the underlying distribution. They also allow to reframe the prediction problem as a classification in which not the observed or predicted resistance scores of a genotype are used as response values but rather its assignment to the "top" or "flop" class (González-Camacho et al. 2018).

Other attempts to improve the prediction accuracy of genomic prediction address the predictors of the model by using haplotype blocks (Difabachew et al. 2023; Weber et al. 2023) or autoencoder features (Islam et al. 2023) as predictors instead of single SNPs or by using subsets of SNPs determined with feature selection (Heinrich et al. 2023; Li et al. 2018). Haplotype blocks group adjacent SNPs on the chromosomes together based on different criteria such as linkage disequilibrium (LD), a fixed number of markers, a fixed physical or genetic distance on the chromosome, or algorithms that aim to create haplotype block libraries that are as representative of the whole set of markers as possible (Pook et al. 2019). When haplotype blocks are used as input variables in RR-BLUP, they are able to capture local epistatic effects (Jiang, Schmidt, and Reif 2018). Autoencoder features are extracted from the encoding layer of an autoencoder. Autoencoders are unsupervised neural networks, in which the input variables are also the targets of the model output (Goodfellow, Bengio, and Courville 2016). In between the input and the output layers is at least one hidden layer with fewer nodes than input variables. These layers function as a bottleneck where the input variables are mapped to a lower dimensional representation (encoding). The number of dimensions can be selected by the user. The model then reconstructs the original input variables from this representation (decoding) in the output layer. To minimize the reconstruction loss, the model learns to preserve as much information of the original variables in the hidden layer as possible (Kramer 1991). Once the "optimal" encoding model is found, the encoded data are used as input variables in a genomic prediction model. In a study in rice, this reduction in dimensionality preserved most of the prediction accuracy while it reduced the computation time considerably (Islam et al. 2023). For feature selection, a genome-wide association study is performed to identify markers that are associated with the trait. The optimum number of markers to be used in the prediction is then determined by cross-validation and the final model is fit accordingly. The results on whether feature selection increases the prediction accuracy compared to the full set of SNPs are contradictory (Heinrich et al. 2023; Li et al. 2018).

Apart from ignoring that the response values are not normally distributed and using RR-BLUP or other methods for metric data anyway, researchers have the option to transform the response so that it better fits the normality assumption. The goal here is to avoid potentially biased results when methods that were originally intended for use with normally distributed data are applied to data on an ordinal scale (Montesinos López et al. 2015). Additionally, when marker effects are estimated with methods like RR-BLUP with an additive model, the additivity of effects can lead to genomic estimates of the genotypic value (GEGVs) that are outside of the original scale, that is, smaller than 0 or larger than 9 on a 0–9 scale. The GEGVs then have no direct translation into meaningful resistance scores. The logit transformation addresses both of these issues. It is intended to achieve a normal distribution of the data (Lesaffre, Rizopoulos, and Tsonaka 2007) and shrinks the score values at both ends of the scale so that GEGVs below or above the limits of the scale are avoided.

We designed this study in order to evaluate the potential of machine learning methods for genomic prediction not only for single SNP markers but also for alternative input features, precisely haplotype blocks and autoencoder features and for subsets of SNP markers determined with feature selection. In order to compare these newer methods with established approaches, we also included Bayesian generalized linear regression and the use of logit-transformed response values. In particular, our objectives were to compare (1) the prediction accuracy of different prediction approaches, including machine-learning methods, and (2) the ability of these approaches to identify the genotypes with the smallest resistance scores with a reference scenario (RR-BLUP with single SNP markers) for the prediction of resistance to five different fungal diseases in a panel of 361 German elite winter wheat lines.

2 | Materials and Methods

2.1 | Phenotypic Data

We evaluated the resistances against *Puccinia triticina* (brown rust), *Fusarium graminearum*, *Septoria tritici*, *Blumeria graminis* (mildew) and *Puccinia striiformis* (yellow rust) of 378

elite wheat lines at three locations in Germany (Böhnshausen, Sachsen-Anhalt; Hovedissen, Nordrhein-Westfalen; Leutewitz, Sachsen) in 2020. Resistances were scored on a 1–9 scale in observation plots in one replication at one (*S. tritici*), two (*F. graminearum*, *P. triticina*), or three locations (*B. graminis*, *P. striiformis*). In case there was more than one location, the arithmetic mean of the two or three observations was used as the resistance score. In order to improve the readability of the manuscript, we use only the name of the disease instead of the full term for the trait, for example, "*S. tritici*" instead of "*S. tritici* resistance score."

2.2 | Genotypic Data

All wheat lines were genotyped with the 25k Illumina iSelect SNP array (SGS TraitGenetics, Gatersleben, Germany). All SNP markers with more than two recorded alleles, more than 10% missing values and an expected heterozygosity of $< 5\%$ as well as all individuals with more than 10% missing marker information were excluded from the analysis. As a result, 16,667 SNP markers and 361 genotypes remained for further analysis. Missing marker data were imputed with BEAGLE (Browning, Zhou, and Browning 2018). We used this dataset for all further calculations. There was no population structure in the dataset (Figure S1).

2.3 | Genomic Prediction Methods

We used genomic prediction based on linear models and machine learning algorithms to evaluate genomic prediction accuracy and efficiency for resistance traits. We used RR-BLUP (Meuwissen, Hayes, and Goddard 2001), RMLA (Hofheinz and Frisch 2014) and Bayesian generalized linear regression (BGLR) with an ordinal response (Pérez and de los Campos 2014). RR-BLUP was technically implemented using a transformation to an animal model (Shen et al. 2013). In order to obtain more robust results in case singular design matrices occur during the cross-validations, we used method 2 of Nazarian and Gezan 2016. The method is available in our software package SelectionTools: https://www.uni-giessen.de/de/fbz/fb09/institute/pflbz2/population-genetics/software. RR-BLUP is considered a standard genomic prediction method in plant and animal breeding programs as it provides stable prediction results (Clark and van der Werf 2013; VanRaden 2008) and is therefore, together with single SNP markers as predictors and resistance scores as response values, treated as the reference in this study.

We also used three supervised machine learning algorithms: support vector regression (SVR)/SVM, GBM and RF. Hyperparameter optimization was performed for all algorithms. SVR is a special case of SVM that is used for metric response values (Drucker et al. 1996). We used a radial basis function as the kernel and tuned the `cost`, the error margin (`margin`) and the influence reach of the individual data points (`sigma`). GBM and RF are both based on ensembles of decision trees (Breiman 2001; Friedman 2001). For GBMs, decision trees are trained in a consecutive order, each tree based on the previous one. For RFs, multiple trees are trained in parallel, each based on a different subset of the training data. The final prediction of the RF model is the average of the predictions of all trees. For both algorithms, we tuned the number of trees used by the model (`ntrees`), the random column sampling rate (`mtry`) and the minimum data points required for a split (`min_n`). We manually set a learning rate of 0.001 for GBM. Default settings were used for all other hyperparameters.

As an alternative, we treated the prediction of resistance scores as a classification task. We used SVM and GBM to predict whether a line was resistant, that is, had a resistance score y smaller than or equal to the 10% quantile Q_{10}, or not. For classification, we used a linear kernel for the SVM and only tuned the `cost` and `margin`. Learning rate for GBM was increased to 0.01 and `min_n` was manually set to 1.

We used a two-step procedure to optimize the hyperparameters for SVR, RF and GBM. The procedure was the same for all algorithms, only the hyperparameters changed (Table 1). We used a 5-fold cross-validation based on the training set to evaluate the hyperparameters. The metric used for evaluation was the square root of the mean square error (RMSE). First, we trained 10 models with hyperparameter combinations based on a maximum entropy grid (Kuhn and Frick 2024; Shewry and Wynn 1987). The essential idea of the maximum entropy grid is to sample points (i.e., combinations of hyperparameters) that cover the hyperparameter space as well as possible, which ensures that the grid search explores a broad range of hyperparameter combinations. Since the points are sampled, they vary between replications. The range of the hyperparameters is shown in Table 1. We used the results of the grid search to initialise an iterative Bayesian optimization, training 10 more models (Snoek, Larochelle, and Adams 2012). Based on the error distribution of the initial maximum entropy grid points, a Bayesian optimization approach can sample and test new combinations from the most promising

TABLE 1 | Overview of hyperparameter ranges considered during tuning.

Hyperparameter	Regression	Classification
RF		
ntrees	(200, 1000)	—
mtry	(0.01, 0.33)	—
min_n	(1, 20)	—
GBM		
ntrees	(50, 500)	(500, 2000)
mtry	(0.01, 0.2)	(0.01, 0.8)
min_n	2, 40)	—
SVR/SVM		
cost	(−10, 5)	(−10, 5)
margin	(0, 0.2)	(0, 0.2)
sigma	(−10, 0)	—

Note: Names of the listed hyperparameters correspond to the argument names used in the software.

regions of the hyperparameter space more quickly. The hyperparameter combination of the model with the smallest RMSE was used to train the final model. The optimization of the hyperparameters for classification was performed analogously, except that some of the parameter ranges in the grid were changed and Cohen's κ was used as the evaluation metric.

2.4 | Feature Engineering

In addition to the complete set of SNP markers, we used three alternative sets of predictors. For the first set, we constructed haplotype blocks based on linkage disequilibrium (LD), which can be measured by r^2 (Zhao et al. 2005). Pairwise LD values were calculated for all SNP markers on each chromosome. SNP markers were added to the left or to the right of a haplotype block as long as the average r^2 between all pairs of SNPs within a block was greater than $t = 0.7$. In order to be able to apply RR-BLUP and RMLA to multi-allelic haplotype block data, the design matrix $\mathbf{Z}$ was re-parameterized (Difabachew et al. 2023).

For the second alternative set of predictors, we extracted the outputs of the encoding layer of an autoencoder. Our autoencoder consisted of five fully connected hidden layers. The layers consisted of [4000,1000, 250,1000, 4000] nodes. The input and output layers consisted of as many nodes as there were predictor variables. The output of the centre layer, consisting of 250 nodes, was treated as the encoding and extracted after model training. We used a rectified linear unit activation function in the hidden layers and applied batch normalization to the outputs of all hidden layers except for the encoding layer. Our data consisted only of homozygous inbred lines with no heterozygous markers present after filtering. Therefore, the markers could be encoded in a binary format, represented by 0 and 1. This allowed for the use of a sigmoid activation function in the output layer. We used binary crossentropy as the loss function and Adam as the optimizer (Kingma and Ba 2015) and trained the autoencoder for 100 epochs.

The third alternative set of predictors was determined by feature selection with a RF model based on GWAS (Heinrich et al. 2023). Analogous to the grid search in the hyperparameter optimization, we conducted a 5-fold cross-valiation on the training set. First, a GWAS was conducted and the markers were ranked according to their p values. Next, RF models were trained, starting with only the most important markers and then incrementing the number of markers in a iterative procedure in steps of 50 from 100 to 1000, of 100 from 1001 to 5000 and of 1000 beyond 5000 markers. The number of markers that resulted in the largest prediction accuracy was determined as the optimum number and the marker set was then used to train another RF model on the complete training set in order to predict the phenotypic values in the validation set. Default settings were used in all RF models. The distribution of the number of SNPs selected by the feature selection procedure is shown in Figure S2.

2.5 | Response Values

For the regression approaches, we used either the resistance scores y or the logit-transformed resistance scores $y^* = \text{logit}\left(\frac{y}{10}\right) = \ln\left(\frac{1}{1-\frac{y}{10}}\right)$ (Lesaffre, Rizopoulos, and Tsonaka 2007) as response values. The division by 10 was necessary because the logit transformation can only be applied to values in the interval (0,1). For the classification methods, the observations y were transformed into two classes: Individuals in the "top" class had a y below or equal to the 10% quantile Q_{10}, and individuals in the "flop" class had a y above the 10% quantile Q_{10}.

2.6 | Prediction Approaches

We define a prediction "approach" as the combination of prediction method, predictor and response values. The name of each approach consists of three elements, divided by a hyphen. The first element is the prediction method: ridge regression BLUP (`RR-BLUP-...`), estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components (`RMLA-...`), Bayesian generalized linear regression with an ordinal response (`BGLR-...`), support vector regression (`SVR-...`), support vector machine (`SVM-...`), gradient boosting machine (`GBM-...`) and random forest (`RF-...`). The predictors can either be SNPs, indicated by `...-SNP-...` as the second element of the approaches, haplotype blocks, indicated by `...-HAP-...`, the autoencoder output, indicated by `...-AEN-...`, or a set of SNP markers determined by feature selection (`...-FS-...`). The last element of each approach is the type of the response value: The use of untransformed values y is indicated by `...-...-0` in the name of the approach, the use of logit-transformed values y^* is indicated by `...-...-1`. Classified values are denoted by `...-...-c`. For example, the approach with the name `SVR-AEN-0` means that a support vector regression was applied on the autoencoder data with the untransformed resistance scores as the response values.

2.7 | Evaluation of the Prediction Approaches

Each prediction approach was evaluated in 200 cross-validation runs. In each of the 200 runs, the dataset was randomly divided into a training set with 289 genotypes (80%) and a validation set with 72 genotypes (20%). The same splits into training and validation set were used for all sets of predictors and algorithms. When predicting ordinal values, the prediction accuracy $r(y,\hat{y})$ was calculated as the correlation between the actual phenotypic values y and the predicted phenotypic values $\hat{y}$ in the validation set. The predicted logit-transformed resistance scores $\hat{y}^*$ were transformed back to $\hat{y}$ and the prediction accuracy was then calculated as $r(y,\hat{y})$.

Cohen's κ (Cohen 1960; Fielding and Bell 1997) as a measure for the agreement between observed and predicted class can be calculated from the confusion matrix for the class assignment (Table 2) as $\kappa = \frac{p_o - p_e}{1-p_e}$ with $P_0 = \frac{\text{tp}+\text{tn}}{n}$ (the proportion of agreement between the observed and predicted values) and $P_e = \frac{\text{tp}+\text{fn}}{n} \times \frac{\text{tp}+\text{fp}}{n} + \frac{\text{fp}+\text{tn}}{n} \times \frac{\text{fn}+\text{tn}}{n}$ (the expected agreement by random chance) (Montesinos López, Montesinos López, and Crossa 2022). The values for κ range from -1 to 1 where $\kappa = 1$ for perfect agreement and $\kappa \leq 0$ for agreement only by random

chance (González-Camacho et al. 2018). The assignment of the observed values y to the "top" or "flop" class was based on the 10% quantile Q_{10}. Individuals in the "top" class had a y smaller than or equal to Q_{10}, and individuals in the "flop" class had a y greater than Q_{10}. This assignment led to different numbers of individuals in the "top" and "flop" classes for the different diseases (Table 3). To account for the different numbers n_{top}, an individual was assigned to the "top" class of the predictions $\hat{y}$ if its predicted value $\hat{y}$ was among the n_{top} individuals with the smallest $\hat{y}$ values for this disease and to the "flop" class otherwise. For the classification approaches `SVM-SNP-C` and `GBM-SNP-C`, the observed values in the confusion matrix resulted from the assignment of the genotypes to the "top" and "flop" classes by the algorithm. A "good" prediction can mean that (a) the prediction accuracy is high and (b) a prediction approach is able to correctly identify the genotypes with extreme resistance scores, that is, the ones that are most interesting for selection decisions, which would be reflected in a κ value of at least 0.3 to 0.5 (Kuhn and Johnson 2013).

The efficiency of an algorithm was evaluated as the mean of the computation time required for one cross-validation run.

2.8 | Software and Hardware

We used R 4.2.2 (R Core Team 2022) for all calculations except the autoencoders, which where calculated using Python 3.10 (Van Rossum and Drake 2009). The adjusted entry means of the genotypes were estimated using "ASReml-R 4.1.0.110" (Butler et al. 2017). Haplotype blocks were built and RR-BLUP and RMLA were calculated using the R package "SelectionTools 22.1." BGLR was calculated using "BGLR" version 1.1.0 (Pérez and de los Campos 2014). SVR was calculated using the package "kernlab 0.9-30" (Karatzoglou, Smola, and Hornik 2022). For RF, we used "ranger 0.16.0" (Wright and Ziegler 2017). GBMs were trained using "lightgbm 3.3.5" (Shi et al. 2023). Maximum entropy grids were constructed using "dials 1.2.0" (Kuhn and Frick 2024) and Bayesian optimization was based on "tune 1.2.1" (Kuhn 2024). We used "parsnip 1.2.1" (Kuhn and Vaughan 2024) and "tidymodels 1.2.0" (Kuhn and Wickham 2020) as wrapper packages to access all the machine learning-related packages. Autoencoders were built using "tensorflow 2.10.0" (Abadi et al. 2015). Missing marker data were imputed with "BEAGLE 5.4" (Browning, Zhou, and Browning 2018). "plink 1.90b6.12" (Chang et al. 2015; Purcell and Chang 2018) was used for recoding the data into VCF format and conducting the GWAS for incremental feature selection.

All calculations were performed on four Intel Xeon Platinum processors 8276 (28 × 2.20 GHz) with 1 TB DDR4 RAM each and 112 kernels in total. For the ML methods, a maximum of 50 kernels was used at the same time. Due to technical limitations on the package side, it was not possible to run one iteration of SVR or SVM on multiple threads. To keep comparability between machine learning algorithms, we ran 50 instances of SVR at the same time and divided the runtime by 50. This way, 50 cores could be used for training.

TABLE 2 | Confusion matrix for a classification problem with two classes.

		Predicted values		
		Top	**Flop**	**∑**
Observed values	Top	`tp`	`fn`	`tp + fn`
	Flop	`fp`	`tn`	`fp + tn`
	∑	`tp + fp`	`fn + tn`	n

Abbreviations: `fn`, number of false negatives; `fp`, number of false positives; n, total number of individuals; `tn`, number of true negatives; `tp`, number of true positives.

3 | Results

3.1 | Phenotypic Values

The observed resistance scores covered only a part of the available range from 1 to 9. *F. graminearum* had the smallest range with observed scores between 3 and 7. The proportion of individuals assigned to the "top" class ranged from 12% in *S. tritici* to 49% in *P. striiformis* (Table 3). An illustration of the distribution of the phenotypic data in one particular validation set can be found in the Supporting Information (Figure S3).

3.2 | Prediction Accuracy of Different Prediction Approaches

All results presented in this section are shown in Figure 1. The overall level of the prediction accuracy was determined by the trait. The reference prediction approach `RR-BLUP-SNP-0`,

TABLE 3 | Summary statistics for the five resistance scores.

Trait	Min	Q_{10}	$Z = Q_{50}$	Q_{90}	Max	n_{top} $(y \leq Q_{10})$	n_{flop} $(y > Q_{10})$
S. tritici	1.00	1.00	2.00	3.00	6.00	42	319
B. graminis	1.00	1.50	2.00	3.50	5.50	119	242
P. triticina	1.00	1.00	2.00	3.75	7.50	66	295
P. striiformis	1.00	1.00	1.33	3.50	6.50	177	184
F. graminearum	3.00	4.00	4.50	5.50	7.00	86	275

Note: The last two columns show how many of the $n = 361$ individuals have a phenotypic value y below/equal to or above the 10% quantile.

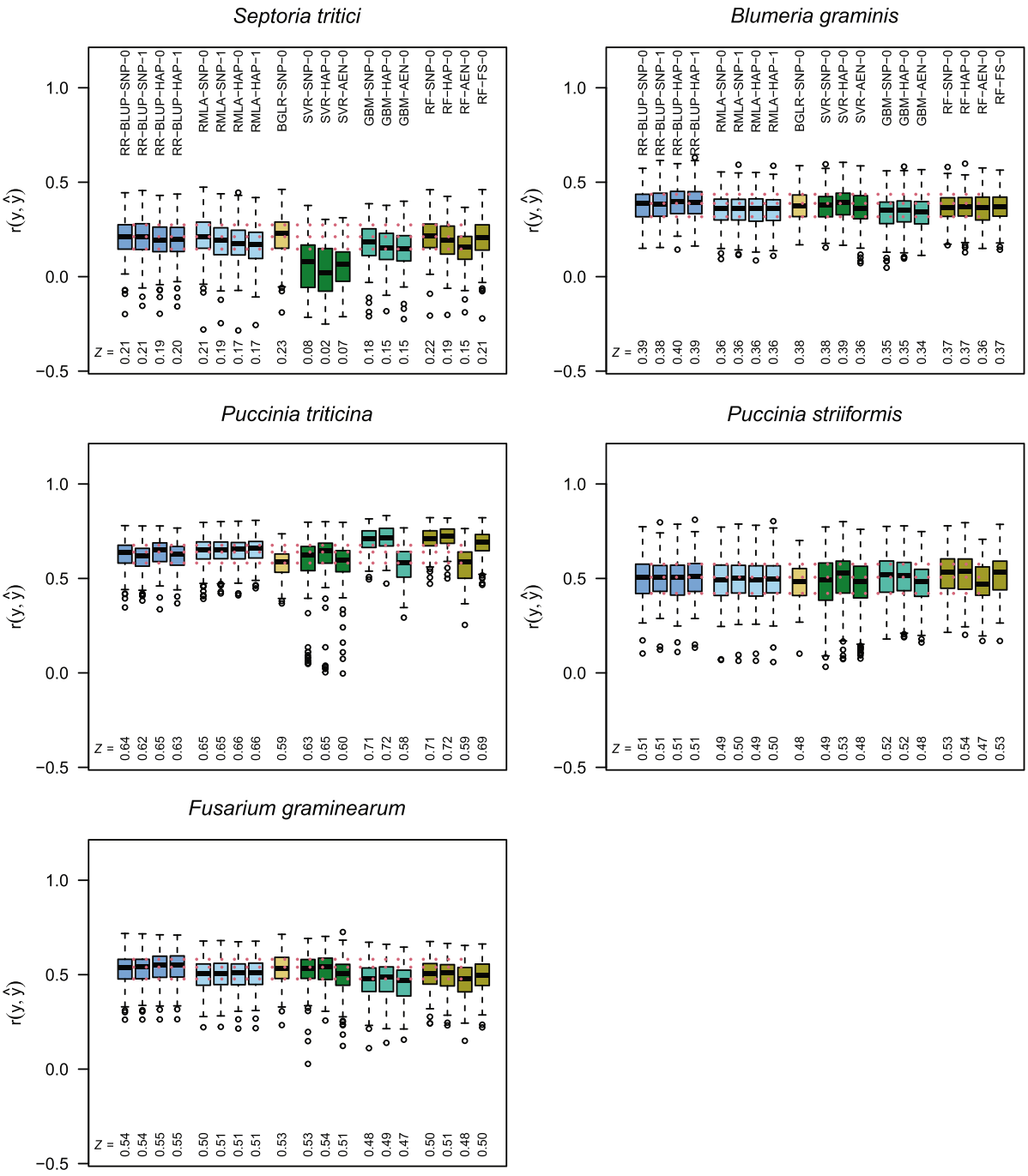

FIGURE 1 | Prediction accuracies for genomic prediction of resistance scores for *S. tritici*, *B. graminis*, *P. triticina*, *P. striiformis* and *F. graminearum* with different prediction approaches. The boxplots show the correlations $r(y, \hat{y})$ between the observed phenotypic values y and the predicted phenotypic values $\hat{y}$ in the validation set for 200 cross-validation runs. Predictions were made with methods ridge regression BLUP (`RR-BLUP-…`), estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components (`RMLA-…`), Bayesian generalized linear regression (`BGLR-…`), support vector regression (`SVR-…`), gradient boosting machine (`GBM-…`) and random forest (`RF-…`). Predictors were either the full set of 16,667 SNP markers (`…-SNP-…`), haplotype blocks based on linkage disquilibrium (`…-HAP-…`), 250 autoencoder features (`…-AEN-…`), or SNP markers identified by feature selection (`…-FS…-`). The response values were either the untransformed resistance scores (`…-…-0`) or the logit-transformed resistance scores (`…-…-1`). Red dotted lines: quartiles from RR-BLUP with 16,667 SNPs (reference). *Z*: median of the correlations $r(y, \hat{y})$ in the 200 cross-validation runs. [Color figure can be viewed at wileyonlinelibrary.com]

RR-BLUP with SNP markers as predictors and the untransformed resistance scores as the response, resulted in medians of $r(y, \hat{y})$ from 0.21 in *S. tritici* to 0.64 in *P. triticina*.

In *S. tritici*, medians of the prediction accuracy ranged from 0.19 to 0.21 in the RR-BLUP approaches and from 0.17 to 0.21 in the RMLA approaches, with the smaller values in the approaches

that used haplotype blocks as predictors. BGLR-SNP-0 had a median of 0.23, the largest value that was observed in this trait. Medians for SVR-SNP-0 and SVR-HAP-0 were 0.08 and 0.02, respectively, while the SVR approach with autoencoder features as predictors, SVR-AEN-0, had a median of 0.07. Medians for the GBM approaches ranged from 0.15 for GBM-HAP-0 to 0.18 for GBM-SNP-0. Medians for the random forest approaches were between 0.15 when autoencoder features were used as predictors (RF-AEN-0) and 0.22 when single SNPs were used instead (RF-SNP-0).

In *B. graminis*, all medians of the correlations $r(y,\hat{y})$ were between 0.34 and 0.40. The largest median, 0.40, was observed with approach RR-BLUP-HAP-0, and the smallest values of 0.34 and 0.35 with the GBM approaches. The medians of the other approaches were in between.

The largest prediction accuracies of all traits were observed in *P. triticina*. The reference approach RR-BLUP-SNP-0 had a median of 0.64, with medians of the other RR-BLUP approaches ranging from 0.62 to 0.65. Medians of the RMLA approaches were 0.65 with untransformed and 0.66 with logit-transformed response values. The median of BGLR-SNP-0 was 0.59. The medians of the SVR approaches ranged from 0.60 for autoencoder features as predictors (SVR-AEN-0) to 0.65 for haplotype blocks (SVR-HAP-0). Medians of the GBM and RF approaches were similar: 0.71 for SNPs as predictors (GBM-SNP-0 and RF-SNP-0), 0.72 for haplotype blocks (GBM-HAP-0 and RF-HAP-0) and 0.58 and 0.59 for autoencoder features (GBM-AEN-0 and RF-AEN-0, respectively). The random forest approach with incremental feature selection (RF-FS-0) was in between with a median of 0.69.

All medians of the prediction accuracies in *P. striiformis* were in the range between 0.47 (for appraoch RF-AEN-0) and 0.53 (approaches SVR-HAP-0, RF-SNP-0 and RF-FS-0). The median of the reference, RR-BLUP-SNP-0, was 0.51 in this case.

In *F. graminearum*, the reference approach RR-BLUP-SNP-0 resulted in a median of the prediction accuracies of 0.54, as did the corresponding approach with haplotype blocks. When logit-transformed response values were used instead, the medians of the prediction accuracies increased to 0.55. RMLA approaches resulted in medians of 0.50 with single SNPs and untransformed response values (RMLA-SNP-0) and 0.51 otherwise. The median of approach BGRL-SNP-0 was 0.53. Among the machine learning methods, the SVR approaches had the largest medians with 0.54 for SVR-HAP-0 and 0.53 for SVR-SNP-0. The smallest medians were observed in the GBM and RF approaches with values of 0.48 for GBM-SNP-0 and RF-AEN-0 and 0.47 for GBM-AEN-0. The remaining RF approaches resulted in medians of 0.50 or 0.51.

3.3 | Identification of the Most Resistant Genotypes

Figure 2 visualizes the results presented in this section. When Cohen's κ was used to evaluate the approaches for how well they were able to identify the most resistant genotypes, the overall level of the κ values was again dependent on the trait.

In *S. tritici*, the reference approach RR-BLUP-SNP-0 had a median of 0.11, as did the corresponding approach with haplotype blocks, RR-BLUP-HAP-0. Using logit-transformed response values led to medians of 0.13 in the RR-BLUP approaches. A similar pattern could be observed in the RMLA approaches, with medians of 0.07 for RMLA-SNP-0 and 0.08 for RMLA-HAP-0 and 0.10 and 0.11 for RMLA-SNP-1 and RMLA-HAP-1, respectively. BGLR-SNP-0 had a median of 0.10. The medians were 0.01 for SVR-SNP-0 and 0.13 and SVR-HAP-0. The use of autoencoder features led to a median of 0.04 in SVR-AEN-0 and the classification approach SVM-SNP-c resulted in a median of 0.02. In the GBM approaches based on regression, the medians ranged between 0.09 for GBM-HAP-0 and 0.13 for GBM-AEN-0. The classification approach GBM-SNP-c had a median of -0.03. The medians of the RF approaches were 0.11 for RF-SNP-0, RF-HAP-0 and RF-FS-0 and 0.10 for RF-AEN-0.

In *B. graminis*, all medians were between 0.21 and 0.23, with the exception of the classification approaches SVM-SNP-c and GBM-SNP-c with median of 0.05 and 0.11, respectively.

In *P. triticina*, the reference RR-BLUP-SNP-0 had a median of 0.16. The medians of the remaining RR-BLUP approaches ranged from 0.15 to 0.18 and showed more variation than the reference. The RMLA approaches resulted in medians of 0.22 for RMLA-SNP-1 and 0.20 for the others. BGRL-SNP-0 had a median of 0.16. The medians of the SVR approaches were either 0.15 or 0.16, with a smaller median of 0.04 for the classification approach SVM-SNP-c. GBM-SNP-0 and GBM-HAP-0 resulted in medians of 0.25. Smaller medians of 0.16 and 0.11 were observed for GBM-AEN-0 and GBM-SNP-c. The pattern for the random forest approaches was similar, with medians of 0.25, 0.28 and 0.15 for approaches RF-SNP-0, RF-HAP-0 and RF-AEN-0, respectively. Approach RF-FS-0 was in between with a median of 0.20.

In *P. striiformis*, the range of the κ values was smaller than for the other traits. The RR-BLUP approaches resulted in medians of 0.30 (for RR-BLUP-SNP-0) to 0.33 (for RR-BLUP-HAP-1). All RMLA approaches had medians of 0.28. The median of BGLR-SNP-0 was 0.31. SVR-HAP-0 had a median of 0.31, compared to medians of 0.28 and 0.25 in the other SVR/SVM approaches. The medians of the GBM approaches were between 0.25 and 0.30. The medians of RF-SNP-0 and RF-HAP-0 were 0.28 and 0.30, respectively, compared to medians of 0.25 for RF-AEN-0 and 0.28 for RF-FS-0.

The overall level of the κ values was highest in *F. graminearum*. All RR-BLUP and RMLA approaches as well as BGLR-SNP-0 had medians of 0.36. The largest median for this trait, 0.38, was observed for approach SVR-SNP-0. The medians of SVR-HAP-0, SVR-AEN-0 and SVM-SNP-c were 0.36, 0.31 and 0.30, respectively. The medians of the GBM regression approaches ranged from 0.26 to 0.31 and the median of the classification approach GBM-SNP-c was 0.24. The medians of the RF approaches ranged from 0.28 to 0.31.

3.4 | Correlation Between r and κ

Figure 3 visualizes the relationship between the prediction accuracy $r(y,\hat{y})$ and Cohen's κ. The means of both measures

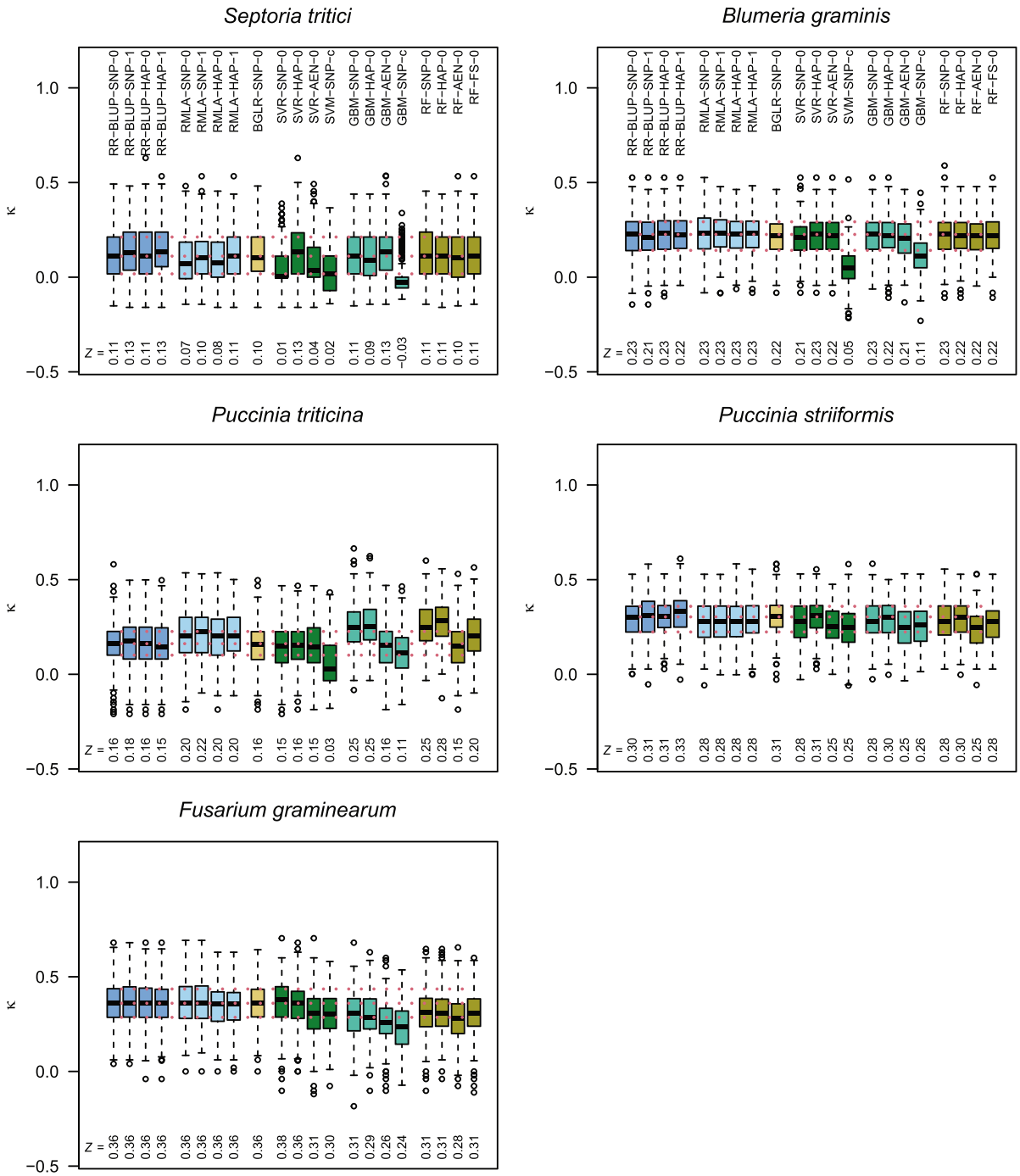

FIGURE 2 | Cohen's κ for genomic prediction of resistance scores for *S. tritici*, *B. graminis*, *P. triticina*, *P. striiformis*, and *F. graminearum* with different prediction approaches. The boxplots show the κ values for the agreement between the assignment to the "top" class (y or $\hat{y}$ equal to or below the 10% quantile Q_{10}) and the "flop" class (y or $\hat{y}$ greater than the 10% quantile Q_{10}) in the validation set for 200 cross-validation runs. Predictions were made with methods ridge regression BLUP (`RR-BLUP-...`), estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components (`RMLA-...`), Bayesian generalized linear regression (`BGLR-...`), support vector regression (`SVR-...`), support vector machine (`SVM-...`), gradient boosting machine (`GBM-...`), and random forest (`RF-...`). Predictors were either the full set of 16,667 SNP markers (`...-SNP-...`), haplotype blocks based on linkage disquilibrium (`...-HAP-...`), 250 autoencoder features (`...-AEN-...`), or SNP markers identified by feature selection (`...-FS...-`). The response values were either the untransformed resistance scores (`...-....-0`), the logit-transformed resistance scores (`...-....-1`), or classifications based on the 10% quantile Q_{10} (`...-....-c`). Red dotted lines: quartiles from RR-BLUP with 16,667 SNPs (reference). Z: median of the κ values in the 200 cross-validation runs. [Color figure can be viewed at wileyonlinelibrary.com]

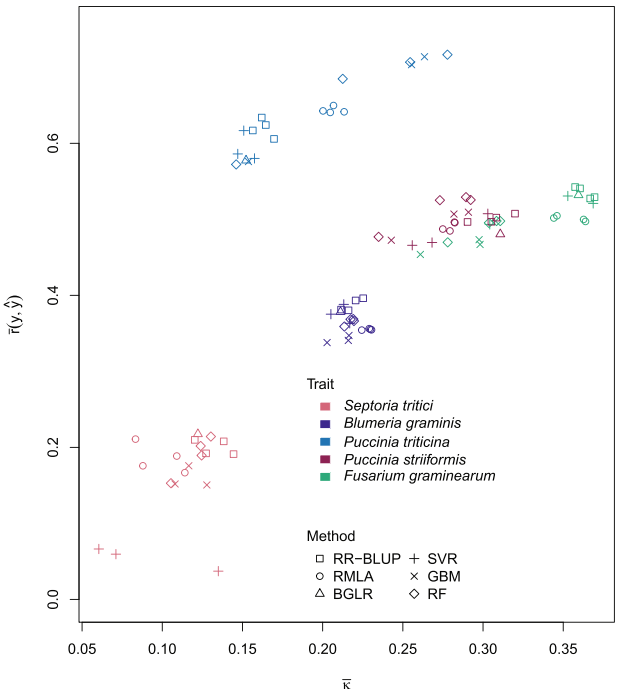

FIGURE 3 | Mean values of correlations $r(y, \hat{y})$ between the observed phenotypic values y and the predicted phenotypic values $\hat{y}$ in the validation set and of Cohen's κ for 200 cross-validation runs. Displayed are the values for the resistance scores for *S. tritici*, *B. graminis*, *P. triticina*, *P. striiformis* and *F. graminearum* for different prediction approaches. Predictions were made with methods ridge regression BLUP (`RR-BLUP`), estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components (`RMLA`), Bayesian generalized linear regression (`BGLR`), support vector regression (`SVR`), gradient boosting machine (`GBM`) and random forest (`RF`). Predictors were either the full set of 16,667 SNP markers, haplotype blocks based on linkage disquilibrium, 250 autoencoder features, or SNP markers identified by feature selection. The phenotypic values used as response values were either the untransformed resistance scores or the logit-transformed resistance scores. Different predictors and response values are not visualized. [Color figure can be viewed at wileyonlinelibrary.com]

showed a correlation across the traits. *P. triticina*, which had the largest mean prediction accuracies of around 0.60, had mean κ values of 0.12 to 0.28. Larger mean κ values of between 0.23 and 0.37 were observed in *P. striiformis* and *F. graminearum* together with smaller mean prediction accuracies of around 0.50. Within the traits, a linear relationship between $r(y, \hat{y})$ and κ could be observed in *P. triticina* and, to a smaller extent, in *P. striiformis* and *F. graminearum*, but not in *S. tritici* and *B. graminis*.

3.5 | Computation Times

The computation times of all approaches can be found in Table 4. RR-BLUP with all 16,667 single SNP markers as predictors was the fastest method with a computation time of 0.68 second per individual cross-validation run on average. However, SVR with autoencoder features was faster when considering the averaged runtime of the parallelization. RMLA and BGLR had a computation time that was about twice as long for the same set of predictors. The computation times of the machine learning methods with single SNP markers were longer with 1.65 min for SVR (averaged), 5.02 min for GBM regression and 3.78 min for RF. When haplotype blocks instead of single SNPs were used, computation times of both RR-BLUP and RMLA increased compared to single SNP markers. An increase of the computation time was also observed for GBM regression, RF and SVR. The use of autoencoder features as predictors in the machine learning methods reduced their computation times to around 1 min or less. The computation time of SVM (averaged) was about 2.87 min and longer compared to that of SVR with single SNP markers. In contrast, the GBM classification took more than twice as long per run as the corresponding regression approach. Computation time was much longer for SVR and SVM compared to other approaches when considering the runtime of the individual cross-validation runs. SVR took 82 min with SNPs and 132 min with haplotype blocks. The autoencoder-based approach (`SVR-AEN-0`) was closer to the other machine learning approaches with 6.36 minutes of individual computation time. SVM took slightly longer than the SNP-based approach `SVR-SNP-0`.

4 | Discussion

4.1 | Prediction Accuracy of Different Prediction Approaches

4.1.1 | Trait

The overall level of the prediction accuracy was determined by the trait (Figure 1). Prediction accuracies between the different approaches varied less for *B. graminis*, *P. striiformis* and *F. graminearum* and more for *S. tritici*, the trait with the smallest overall prediction accuracies with medians around 0.20, and *P. triticina*, the trait with the largest overall prediction accuracies with medians around 0.60 or greater (Figure 1).

The wheat lines in this study are either registered elite varieties or genotypes that are already close to registration. They have therefore been bred for resistance against a variety of pathogens which is reflected in the distribution of the phenotypic values: The observations only cover part of the available scale from 1 to 9 and the larger values, indicating less resistance, are relatively rare (Table 3 and Figure S3). Small prediction accuracies could therefore be at least partially due to the low variation in the response values. In order to obtain reliable results for the genomic predictions, other authors suggest a training set of diverse lines which is continually updated with new breeding material and which can be phenotyped once per season (Juliana et al. 2017).

4.1.2 | Prediction Method

Predictions made with RMLA resulted in similar prediction accuracies as predictions made with RR-BLUP in most cases (Figure 1), even though the genetic architecture of resistance traits is made up of major and minor genes and should, in theory, be captured better by a prediction model like RMLA that allows for heterogeneous marker variances (Hofheinz and Frisch 2014).

TABLE 4 | Computation times in minutes for the different prediction approaches.

Prediction method	Computation time (in minutes)
RR-BLUP-SNP-0	0.68
RR-BLUP-SNP-1	0.68
RR-BLUP-HAP-0	4.97
RR-BLUP-HAP-1	4.74
RMLA-SNP-0	1.45
RMLA-SNP-1	1.45
RMLA-HAP-0	2.10
RMLA-HAP-1	2.10
BGLR-SNP-0	1.33
SVR-SNP-0	1.65 (82.51)
SVR-HAP-0	2.65 (132.53)
SVR-AEN-0	0.13 (6.36)
SVM-SNP-c	2.87 (143.44)
GBM-SNP-0	5.02
GBM-HAP-0	7.2
GBM-AEN-0	0.95
GBM-SNP-c	12.8
RF-SNP-0	3.78
RF-HAP-0	4.89
RF-AEN-0	0.75
RF-FS-0	2.14

Note: The table contains the average values of 200 cross-validation runs for all five traits. For SVR/SVM, due to parallelization, we provide the averaged time per run for 200 cross-validation runs and the time required for a single run in brackets ( ). Predictions were made with methods ridge regression BLUP (RR-BLUP-...), estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components (RMLA-...), Bayesian generalized linear regression (BGLR-...), support vector regression (SVR-...), support vector machine (SVM-...), gradient boosting machine (GBM-...) and random forest (RF-...). Predictors were either the full set of 16,667 SNP markers (...-SNP-...), haplotype blocks based on linkage disquilibrium (...-HAP-...), 250 autoencoder features (...-AEN-...), or SNP markers identified by feature selection (...-FS...-). The response values were either the untransformed resistance scores (...-...-0), the logit-transformed resistance scores (...-...-1), or classifications based on the 10% quantile Q_{10} (...-...-c).

Other studies on genomic prediction of rust in wheat found that Bayesian methods, which also allow for heterogeneous marker variances, are not necessarily superior to RR-BLUP or genomic BLUP (GBLUP) for the prediction of resistance scores in empirical datasets (Tehseen et al. 2021; Mahmood et al. 2022) even though simulation studies predict that they should be (Meher, Rustgi, and Kumar 2022). A study on both empirical and simulated datasets found the same discrepancy between properties of the methods that should result in better prediction accuracies in theory—and do in simulated datasets—and the actual performance in real-life data (John et al. 2022).

Bayesian generalized linear regression with ordinal response values (approach BGLR-SNP-0) also led to correlations $r(y,\hat{y})$ that were mostly similar to those of RR-BLUP-SNP-0, except for *P. triticina*, in which the values were smaller (Figure 1). This was true regardless of the distribution of the phenotypic values in the validation set. The use of a method specifically designed for ordinal response values therefore did not result in greater prediction accuracies than the use of methods designed for metric response values.

For the machine learning approaches, we did not observe larger prediction accuracies than for the reference approach except for GBM-SNP-0, GBM-HAP-0, RF-SNP-0 and RF-HAP-0 in *P. triticina*. Since we showed in another study that haplotype blocks also led to larger prediction accuracies in this trait compared to single SNPs (Difabachew et al. 2023), we hypothesize that local epistatic effects that can be incorporated by haplotype blocks and machine learning methods, but not by RR-BLUP with single SNPs, may play a role here (Jiang, Schmidt, and Reif 2018; Momen et al. 2018). The prediction accuracies for SVR with single SNP markers (SVR-SNP-0) were generally in the range of those for the corresponding RR-BLUP approach (RR-BLUP-SNP-0), with a difference in the medians of 0.02 at most, except for *S. tritici*. Predictions made with method RF mostly had medians that were 0.01 to 0.04 points greater than those for the corresponding GBM approaches (Figure 1). Only for *P. triticina*, the medians were similar for GBM and RF. Our results partially confirm and partially contradict the results of others. For example, RF resulted in larger prediction accuracies compared to RR-BLUP in the prediction of *P. striiformis* (Tomar et al. 2021) and *F. graminearum* (Rutkoski et al. 2012). In a recent simulation study on genomic prediction with machine learning methods, SVM, RF and GBM showed larger prediction accuracies in a dataset with clear population structure but not in a dataset in which population structure was absent (Jones et al. 2023). The latter corresponds to our dataset (Figure S1), possibly explaining the equal performance of linear and machine learning genomic prediction approaches in four of the five traits in our study. In an extensive study spanning six crops with mostly quantitative traits that compared the prediction accuracy of RR-BLUP, Bayes A and B, Bayesian LASSO, Baysian ridge regression, SVR with linear and nonlinear kernels, gradient tree boosting, artificial neural networks and convolutional neural networks, the results were similar to ours: No single genomic prediction method performed best in all crop/trait combinations, and RR-BLUP was close to the method with the largest prediction accuracy in most cases (Azodi et al. 2019). The same result was found in another study on a simulated animal dataset and three real-life datasets for maize (Lourenço et al. 2024). Our study confirms these findings for resistance traits in wheat.

4.1.3 | Predictor

Replacing single SNP markers with haplotype blocks led to mostly similar prediction accuracies for the corresponding methods, with only small decreases or increases (Figure 1). It has to be noted that there are other possibilities for defining haplotype blocks. In this study, haplotype blocks were built

based on an LD threshold of $r^2 > 0.7$. Other thresholds as well as other methods like building blocks based on a fixed number of markers, fixed window sizes in cM or kilobases on the chromosome, or haplotype block libraries created with the R package HaploBlocker (Pook et al. 2019) are alternative options which have already been investigated in greater detail for this dataset (Difabachew et al. 2023) and others (Weber et al. 2023) and have been shown to increase prediction accuracy in some but not in all cases.

Using autoencoder features as predictors in the machine learning methods resulted in medians of the prediction accuracies that were either similar to or smaller than those of the other approaches, regardless of the method they were used in (Figure 1). Their use led to a reduction in the computation time compared to other predictors for the machine learning methods (Table 4). However, since the computation of the autoencoder features also needs time and the prediction accuracy is generally decreased compared to other predictors, their use as inputs for the machine learning methods was not advantageous in our dataset. More complex studies (Islam et al. 2023) demonstrate the feasibility of preserving prediction accuracy with a reduced set of autoencoder features. We found larger prediction accuracies for GBM and RF than for RR-BLUP with single SNP markers in *P. triticina*, albeit with longer computation times (Table 4). Further research is required to find an easily applicable way to use the autoencoder while maintaining the prediction accuracy and thus save a lot of computation time.

When sets of markers determined by feature selection were used as predictors in a random forest prediction approach (`RF-FS-0`), prediction accuracies were similar to those obtained with the full set of SNPs in nearly all cases (Figure 1), even though the distributions of the numbers of selected SNPs were different between the traits (Figure S2). The findings from other authors in this respect are contradictory: Some found substantial increases with incremental feature selection compared to using the full set of SNPs (Heinrich et al. 2023) while the results of others are similar to ours (Li et al. 2018). We conclude that while feature selection can be beneficial in some cases, further research is needed to determine under which circumstances exactly it can improve the prediction accuracy.

4.1.4 | Response Values

When logit-transformed resistance scores (approaches `RR-BLUP-SNP-1`, `RR-BLUP-HAP-1`, `RMLA-SNP-1` and `RMLA-HAP-1`) were used as response variables instead of the untransformed resistance scores (approaches `RR-BLUP-SNP-0`, `RR-BLUP-HAP-0`, `RMLA-SNP-0` and `RMLA-HAP-0`), differences between the prediction accuracies were small with a maximum of 0.02 points in the medians of the prediction accuracies of the corresponding approaches (Figure 1). We conclude that the logit transformation could successfully address the problem of GEGVs outside the interpretable range and yields predictions with a similar accuracy to those obtained with untransformed data in our dataset. However, it did not improve the predictions by a change in the distribution of the response values. These findings are supported by a study on *P. striiformis* infection in wheat in which the use of logarithmic, boxcox and square root transformations on the observed data did not result in consistent increases in the prediction accuracies obtained with RR-BLUP (Merrick et al. 2022).

4.2 | Identification of the Most Resistant Genotypes

Overall, κ should have a value between 0.3 and 0.5 for acceptable agreement between the classes (Kuhn and Johnson 2013), indicating that an approach is able to identify the most resistant genotypes. We found values in this range only for *F. graminearum*. In the other traits, the κ values were usually smaller.

The patterns for the comparisons between the κ values of the regression approaches in terms of the prediction methods, predictors and response values were the same as for the prediction accuracy (Figure 2). The use of alternative prediction methods, predictors and logit-transformed response values led to medians of the κ values that were either smaller than or similar to the reference approach `RR-BLUP-SNP-0`. The only exception was *P. triticina*, with an increase for GBM and RF from a median of the κ values of 0.16 for `RR-BLUP-SNP-0` to 0.25 for `GBM-SNP-0`, `GBM-HAP-0` and `RF-SNP-0` and 0.28 for `RF-HAP-0`. Autoencoder features as predictors led to smaller κ values in most cases in comparison to `RR-BLUP-SNP-0` (Figure 2). We could not confirm the superiority of SVM for the identification of superior genotypes that was found in 16 wheat datasets (Ornella et al. 2014).

In most studies on genomic prediction, only the prediction accuracy $r(y,\hat{y})$ is reported. However, while a large value for the prediction accuracy indicates that the predictions are accurate on average, this is different from the correct identification of the most resistant genotypes, which are the ones that are interesting for selection. Ideally, a prediction approach would yield both large κ values as well as have a large prediction accuracy. We found a positive correlation between the means of the prediction accuracy $r(y,\hat{y})$ and the means of κ across the traits (Figure 3). Apart from the smaller range of the κ values, these findings are mostly similar to those for rust resistance in wheat (Ornella et al. 2014; González-Camacho et al. 2018). However, both measures must be considered together when the suitability of a method identify superior genotypes is evaluated: In *P. triticina*, the prediction accuracies were largest for all traits, with mean values around 0.6, while the means of the κ values were between 0.12 and 0.28. In contrast, the mean prediction accuracies in *F. graminearum* were around 0.5, but the means of the κ values were all greater than 0.25 (Figure 3). Our findings show that even if κ and $r(y,\hat{y})$ are positively correlated, a large prediction accuracy does not automatically translate into a κ value that is sufficient for the selection of superior genotypes.

4.3 | Summary

A good genomic prediction model is supposed to extract the relevant information from the genotypic data while simultaneously dealing with the noise which comes from other factors. Linear models like RR-BLUP make simplifying assumptions in this situation, particularly when they include only additive effects, like

in our study. The questions then become if there are additional patterns in the genotypic data that cannot be captured by linear models and if machine learning methods are able to find these patterns. In our dataset, RR-BLUP was consistently among the methods with the largest prediction accuracies and the best abilities to identify resistant genotypes in four of the five investigated traits. Compared to machine learning methods, RR-BLUP is implemented in most genomic prediction software. It is easy to apply without the need for hyperparameter tuning and consequently very fast. Additionally, the resulting marker effects are easy to interpret and understand. However, we found substantial increases in the prediction accuracies and κ values compared to the reference approach `RR-BLUP-SNP-0` in *P. triticina*, indicating that investing the additional effort to fine-tune such a method may be worth it. We also found that even though there was a positive correlation between the prediction accuracy and Cohen's κ, a measure to judge how well the most resistant genotypes can be identified, the correlation is not perfect and a large value for the prediction accuracy does not necessarily translate into an equally large κ value. This shows that the prediction accuracy should not be the only measure that is used to select a "good" genomic prediction method.

Author Contributions

Matthias Frisch, Rod Snowdon and Andreas Stahl conceived the study. Michael Koch, Martin Kirchhoff, László Cselényi, Markus Wolf and Jutta Förster collected the field data and genotypic data. Anna Moritz, Andreas Stahl, Benjamin Wittkop and Matthias Frisch evaluated the field data. Yohannes Difabachew carried out the genomic predictions with RR-BLUP, RMLA and BGLR. Philipp Heilmann carried out the genomic predictions with SVR/SVM, GBM and RF. Philipp Heilmann, Yohannes Difabachew and Carola Zenke-Philippi wrote the manuscript. All authors read and approved the final manuscript.

Acknowledgments

The project was funded by the Federal Ministry of Food and Agriculture (BMEL) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support programme (FKZ 2818403A18). Open Access funding enabled and organized by Projekt DEAL.

Conflicts of Interest

Michael Koch is employed by Deutsche Saatveredelung AG. Martin Kirchhoff was employed by Nordsaat Saatzucht GmbH and is employed by Nordzucker AG. László Cselényi is employed by W. von Borries-Eckendorf GmbH & Co. KG. Markus Wolf is employed by German Seed Alliance GmbH. Jutta Förster is employed by Saaten-Union Biotec GmbH. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Data Availability Statement

The genotypic and phenotypic data as well as the scripts used for this study can be downloaded from https://github.com/czp-jlu/resistance.

References

Abadi, M., A. Agarwal, P. Barham, et al. 2015. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems." https://www.tensorflow.org/. Software available from tensorflow.org.

Azodi, C. B., E. Bolger, A. McCarren, M. Roantree, G. de los Campos, and S.-H. Shiu. 2019. "Benchmarking Parametric and Machine Learning Models for Genomic Prediction of Complex Traits." *G3: Genes, Genomes, Genetics* 9, no. 11: 3691–3702.

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

Browning, B. L., Y. Zhou, and S. R. Browning. 2018. "A One-Penny Imputed Genome From Next Generation Reference Panels." *American Journal of Human Genetics* 103: 338–348.

Butler, D. G., B. R. Cullis, A. R. Gilmour, B. G. Gogel, and R. Thompson. 2017. *ASReml-R Reference Manual Version 4*. Hemel Hempstead, HP1 1ES, UK: VSN International Ltd. https://asreml.kb.vsni.co.uk/knowledge-base/asreml_r_documentation/.

Chang, C. C., C. C. Chow, LCAM Tellier, S. Vattikuti, S. M. Purcell, and J. J. Lee. 2015. "Second-Generation PLINK: Rising to the Challenge of Larger and Richer Datasets." *GigaScience* 4, no. 1: s13742–015.

Clark, S. A., and J. van der Werf. 2013. "Genomic Best Linear Unbiased Prediction (gBLUP) for the Estimation of Genomic Breeding Values." edited by C. Gondro, J. van der Werf, and B. Hayes, *Genome-Wide Association Studies and Genomic Prediction*. Totowa, NJ: Humana Press, pp. 321–330.

Cohen, J. 1960. "A Coefficient of Agreement for Nominal Scales." *Educational and Psychological Measurement* 20, no. 1: 37–46.

Difabachew, Y. F., M. Frisch, A. L. Langstroff, et al. 2023. "Genomic Prediction With Haplotype Blocks in Wheat." *Frontiers in Plant Science* 14: 1168547.

Drucker, H., C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. 1996. "Support Vector Regression Machines." edited by M. C. Mozer, M. Jordan, and T. Petsche, *Advances in Neural Information Processing Systems*, Vol. 9. MIT Press, pp. 155–161. https://proceedings.neurips.cc/paper_files/paper/1996/file/d38901788c533e8286cb6400b40b386d-Paper.pdf.

Fielding, A. H., and J. F. Bell. 1997. "A Review of Methods for the Assessment of Prediction Errors in Conservation Presence/Absence Models." *Environmental Conservation* 24, no. 1: 38–49.

Friedman, J. H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." *Annals of Statistics* 29: 1189–1232.

González-Camacho, J. M., L. Ornella, P. Pérez-Rodríguez, D. Gianola, S. Dreisigacker, and J. Crossa. 2018. "Applications of Machine Learning Methods to Genomic Selection in Breeding Wheat for Rust Resistance." *Plant Genome* 11, no. 2: 170104.

Goodfellow, I., Y. Bengio, and A. Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Heinrich, F., T. M. Lange, M. Kircher, F. Ramzan, A. O. Schmitt, and M. Gültas. 2023. "Exploring the Potential of Incremental Feature Selection to Improve Genomic Prediction Accuracy." *Genetics Selection Evolution* 55, no. 1: 78.

Hofheinz, N., and M. Frisch. 2014. "Heteroscedastic Ridge Regression Approaches for Genome-Wide Prediction With a Focus on Computational Efficiency and Accurate Effect Estimation." *G3: Genes, Genomes, Genetics* 4, no. 3: 539–546.

Islam, T., C. Kim, H. Iwata, H. Shimono, and A. Kimura. 2023. "DeepCGP: A Deep Learning Method to Compress Genome-Wide Polymorphisms for Predicting Phenotype of Rice." *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 20, no. 3: 2078–2088.

Jiang, Y., R. H. Schmidt, and J. C. Reif. 2018. "Haplotype-Based Genome-Wide Prediction Models Exploit Local Epistatic Interactions Among Markers." *G3: Genes, Genomes, Genetics* 8, no. 5: 1687–1699.

John, M., F. Haselbeck, R. Dass, et al. 2022. "A Comparison of Classical and Machine Learning-Based Phenotype Prediction Methods on Simulated Data and Three Plant Species." *Frontiers in Plant Science* 13: 932512.

Jones, D., R. Fornarelli, M. Derbyshire, M. Gibberd, K. Barker, and J. Hane. 2023. "The Pursuit of Genetic Gain in Agricultural Crops Through the Application of Machine-Learning to Genomic Prediction." *Frontiers in Genetics* 14: 1186782.

Juliana, P., R. P. Singh, P. K. Singh, et al. 2017. "Genomic and Pedigree-Based Prediction for Leaf, Stem, and Stripe Rust Resistance in Wheat." *Theoretical and Applied Genetics* 130: 1415–1430.

Karatzoglou, A., A. Smola, and K. Hornik. 2022. "kernlab: Kernel-Based Machine Learning Lab." https://CRAN.R-project.org/package%3Dkernlab. R package version 0.9-30.

Kingma, D. P., and J. Ba. 2015. "Adam: A Method for Stochastic Optimization." In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* edited by Y. Bengio, and Y. LeCun. https://arxiv.org/abs/1412.6980.

Kramer, M. A. 1991. "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks." *AIChE Journal* 37, no. 2: 233–243.

Kuhn, M. 2024. "tune: Tidy Tuning Tools." https://CRAN.R-project.org/package%3Dtune. R package version 1.2.1.

Kuhn, M., and H. Frick. 2024. "dials: Tools for Creating Tuning Parameter Values." https://CRAN.R-project.org/package%3Ddials. R package version 1.2.1.

Kuhn, M., and K. Johnson. 2013. *Applied Predictive Modeling*. New York, NY: Springer.

Kuhn, M., and D. Vaughan. 2024. "parsnip: A Common API to Modeling and Analysis Functions." https://CRAN.R-project.org/package%3Dparsnip. R package version 1.2.1.

Kuhn, M., and H. Wickham. 2020. "tidymodels: A Collection of Packages for Modeling and Machine Learning Using Tidyverse Principles." https://www.tidymodels.org.

Lesaffre, E., D. Rizopoulos, and R. Tsonaka. 2007. "The Logistic Transform for Bounded Outcome Scores." *Biostatistics* 8, no. 1: 72–85.

Li, B., N. Zhang, Y.-G. Wang, A. W. George, A. Reverter, and Y. Li. 2018. "Genomic Prediction of Breeding Values Using a Subset of SNPs Identified by Three Machine Learning Methods." *Frontiers in Genetics* 9: 237.

Lourenço, V. M., J. O. Ogutu, R. A. P. Rodrigues, A. Posekany, and H.-P. Piepho. 2024. "Genomic Prediction Using Machine Learning: A Comparison of the Performance of Regularized Regression, Ensemble, Instance-Based and Deep Learning Methods on Synthetic and Empirical Data." *BMC Genomics* 25, no. 1: 152.

Mahmood, Z., M. Ali, J. I. Mirza, et al. 2022. "Genome-Wide Association and Genomic Prediction for Stripe Rust Resistance in Synthetic-Derived Wheats." *Frontiers in Plant Science* 13: 788593.

Meher, P. K., S. Rustgi, and A. Kumar. 2022. "Performance of Bayesian and BLUP Alphabets for Genomic Prediction: Analysis, Comparison and Results." *Heredity* 128, no. 6: 519–530.

Merrick, L. F., D. N. Lozada, X. Chen, and A. H. Carter. 2022. "Classification and Regression Models for Genomic Selection of Skewed Phenotypes: A Case for Disease Resistance in Winter Wheat (Triticum aestivum L.)."*Frontiers in Genetics* 13: 835781.

Meuwissen, T. H. E., B. J. Hayes, and M. E. Goddard. 2001. "Prediction of Total Genetic Value Using Genome-Wide Dense Marker Maps." *Genetics* 157: 1819–1829.

Momen, M., A. A. Mehrgardi, A. Sheikhi, et al. 2018. "Predictive ability of Genome-Assisted Statistical Models Under Various Forms of Gene Action." *Scientific Reports* 8: 12309.

Montesinos López, O. A., A. Montesinos López, and J. Crossa. 2022. *Multivariate Statistical Machine Learning Methods for Genomic Prediction*. Cham: Springer.

Montesinos López, O. A., A. Montesinos López, P. Pérez-Rodríguez, G. de los Campos, K. Eskridge, and J. Crossa. 2015. "Threshold Models for Genome-Enabled Prediction or Ordinal Categorical Traits in Plant Breeding." *G3: Genes, Genomes, Genetics* 5: 291–300.

Nazarian, A., and S. A. Gezan. 2016. "GenoMatrix: A Software Package for Pedigree-Based and Genomic Prediction Analyses on Complex Traits." *Journal of Heredity* 107, no. 4: 372–379.

Ornella, L., P. Pérez, E. Tapia, et al. 2014. "Genomic-Enabled Prediction With Classification Algorithms." *Heredity* 112: 616–626.

Ornella, L., S. Singh, P. Perez, et al. 2012. "Genomic Prediction of Genetic Values for Resistance to Wheat Rusts." *The Plant Genome* 5: 136–148.

Pérez, P., and G. de los Campos. 2014. "Genome-Wide Regression and Prediction With the BGLR Statistical Package." *Genetics* 198, no. 2: 483–495.

Pook, T., M. Schlather, G. de Los Campos, M. Mayer, C. C. Schön, and H. Simianer. 2019. "HaploBlocker: Creation of Subgroup-Specific Haplotype Blocks and Libraries." *Genetics* 212, no. 4: 1045–1061.

Purcell, S., and C. Chang. 2018. "Plink v1.90b6.12." https://www.cog-genomics.org/plink/1.9/.

R Core Team. 2022. "R: A Language and Environment for Statistical Computing." Vienna, Austria. https://www.R-project.org.

Rutkoski, J., J. Benson, Y. Jia, G. Brown-Guedira, J.-L. Jannink, and M. Sorrells. 2012. "Evaluation of Genomic Prediction Methods for Fusarium Head Blight Resistance in Wheat." *Plant Genome* 5: 51–61.

Shen, X., M. Alam, F. Fikse, and L. Rönnegård. 2013. "A Novel Generalized Ridge Regression Method for Quantitative Genetics." *Genetics* 193, no. 4: 1255–1268.

Shewry, M. C., and H. P. Wynn. 1987. "Maximum Entropy Sampling." *Journal of Applied Statistics* 14, no. 2: 165–170.

Shi, Y., G. Ke, D. Soukhavong, et al. 2023. "lightgbm: Light Gradient Boosting Machine." https://CRAN.R-project.org/package%3Dlightgbm. R package version 3.3.5.

Snoek, J., H. Larochelle, and R. P. Adams. 2012. "Practical Bayesian Optimization of Machine Learning Algorithms." edited by F. Pereira, C. J. Burges, L. Bottou, and K. Q. Weinberger, *Advances in Neural Information Processing Systems*, Vol. 25. Curran Associates, Inc., pp. 2951–2959.

Tehseen, M. M., Z. Kehel, C. P. Sansaloni, et al. 2021. "Comparison of Genomic Prediction Methods for Yellow, Stem, and Leaf Rust Resistance in Wheat Landraces From Afghanistan." *Plants* 10: 558.

Tomar, V., G. S. Dhillon, D. Singh, et al. 2021. "Evaluations of Genomic Prediction and Identification of New Loci for Resistance to Stripe Rust Disease in Wheat (Triticum aestivum L.)."*Frontiers in Genetics* 12: 710485.

Van Rossum, G., and F. L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace. https://api.semanticscholar.org/CorpusID:61259041.

VanRaden, P. M. 2008. "Efficient Methods to Compute Genomic Predictions." *Journal of Dairy Science* 91, no. 11: 4414–4423.

Wang, X., Y. Xu, Z. Hu, and C. Xu. 2018. "Genomic Selection Methods for Crop Improvement: Current Status and Prospects." *Crop Journal* 6, no. 4: 330–340.

Weber, S. E., M. Frisch, R. J. Snowdon, and K. P. Voss-Fels. 2023. "Haplotype Blocks for Genomic Prediction: A Comparative Evaluation in Multiple Crop Datasets." *Frontiers in Plant Science* 14: 1217589.

Wright, M. N., and A. Ziegler. 2017. "ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R." *Journal of Statistical Software* 77, no. 1: 1–17.

Zhao, H., D. Nettleton, M. Soller, and J. C. M. Dekkers. 2005. "Evaluation of Linkage Disequilibrium Measures Between Multi-Allelic Markers as Predictors of Linkage Disequilibrium Between Markers and QTL." *Genetics Research* 86, no. 1: 77–87.

Supporting Information

Additional supporting information can be found online in the Supporting Information section.

Machine learning for prediction of resistance scores in wheat (*Triticum aestivum* L.) - Supplementary figures

Philipp Georg Heilmann | Yohannes Fekadu Difabachew | Matthias Frisch | Anna Luise Moritz | Andreas Stahl | Benjamin Wittkop | Rod J Snowdon | Michael Koch | Martin Kirchhoff | László Cselényi | Markus Wolf | Jutta Förster | Carola Zenke-Philippi

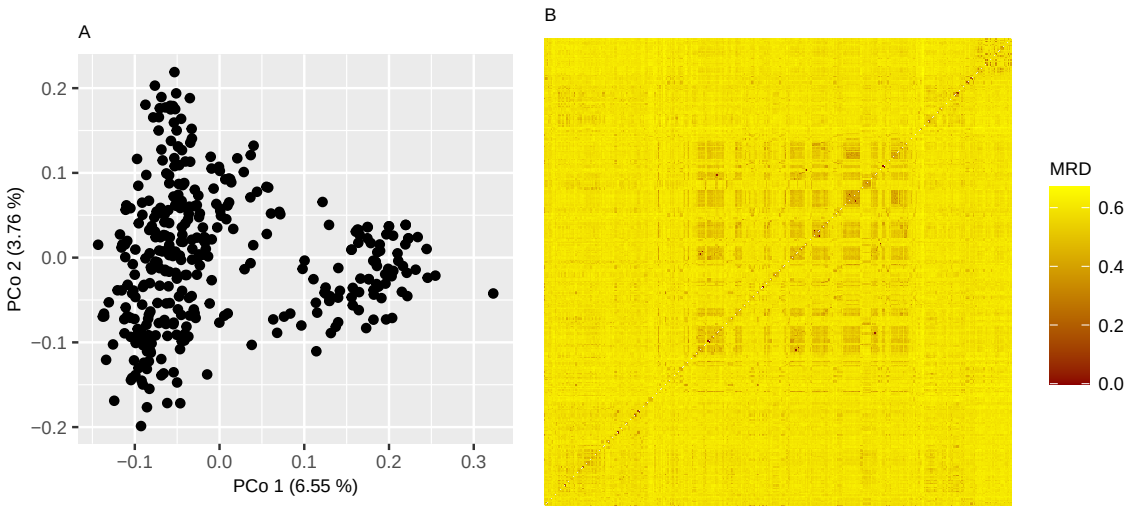

FIGURE S1 (A) Principal coordinate analysis based on the pairwise modified Roger's distances (MRD) and (B) heatmap showing the distances between the 361 elite winter wheat lines.

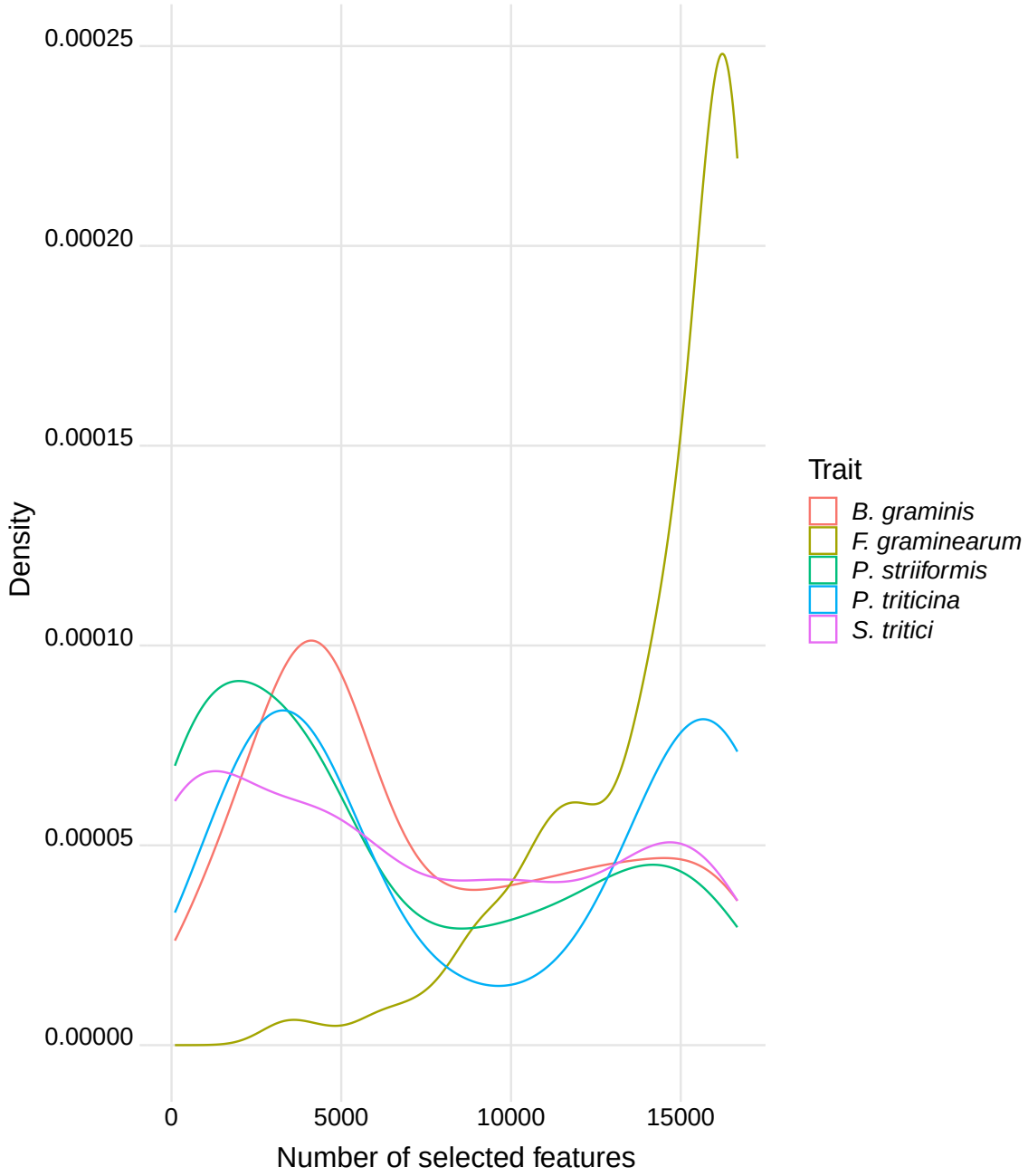

FIGURE S2 Distribution of the number of selected features by method `RF-FS-0` for the five different traits.

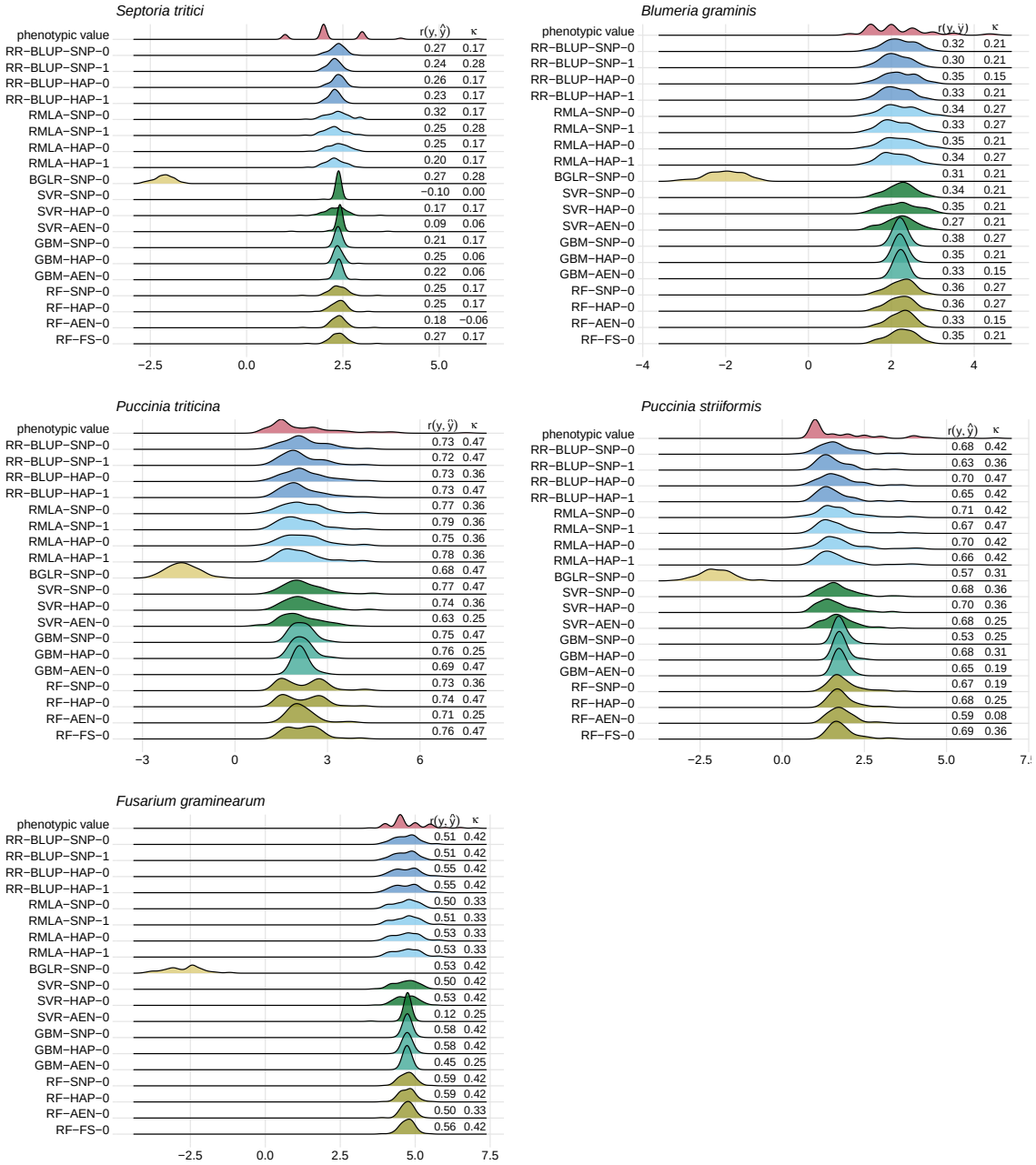

FIGURE S3 Distributions of observed phenotypic values and predicted phenotypic values of resistance scores for *S. tritici*, *B. graminis*, *P. triticina*, *P. striiformis*, and *F. graminearum* with different prediction approaches in the validation set (72 genotypes) in cross-validation run 126. Predictions were made with methods ridge regression BLUP (`RR-BLUP-...`), estimation of the error and genetic variance components with restricted maximum likelihood and partitioning according to ANOVA variance components (`RMLA-...`), Bayesian generalized linear regression (`BGLR-...`), support vector regression (`SVR-...`), gradient boosting machine (`GBM-...`), and random forest (`RF-...`). Predictors were either the full set of 16,667 SNP markers (`...-SNP-...`), haplotype blocks based on linkage disquilibrium (`...-HAP-...`), or 250 autoencoder features (`...-AEN-...`), or SNP markers identified by feature selection (`...-FS...-`). The response values were either the untransformed resistance scores (`...-...-0`) or the logit-transformed resistance scores (`...-...-1`).

Chapter 4

Haplotype-based autoencoders can reduce the dataset dimension and estimate haplotype block effects in different crop species[1]

[1]P. G. Heilmann, E. Grosch, M. Frisch, M. Herrmann, S. Beuch, V. Kurra, M. Mascher, R. Avni, K. Oldach, I. Röhrs, A. Hanemann, R. R. Mehta, C. Reinbrecht, A. Serfling, A. Stahl, M. Stucke, A. Abbadi, T. Kox, and C. Zenke-Philippi. (2025) Haplotype-based autoencoders can reduce the dataset dimension and estimate haplotype block effects in different crop species. *BMC Bioinformatics* **26**(1):289

BMC Bioinformatics

RESEARCH Open Access

Haplotype-based autoencoders can reduce the dataset dimension and estimate haplotype block effects in different crop species

Philipp Georg Heilmann[1*], Emanuel Grosch[1], Matthias Frisch[1], Matthias Herrmann[2], Steffen Beuch[3], Vivek Kurra[4], Martin Mascher[5], Raz Avni[5], Klaus Oldach[6], Ina Röhrs[7], Anja Hanemann[8], Raja Ram Mehta[4], Carsten Reinbrecht[9], Albrecht Serfling[10], Andreas Stahl[10], Marco Stucke[9], Amine Abbadi[11], Tobias Kox[11] and Carola Zenke-Philippi[1]

*Correspondence:
Philipp Georg Heilmann
philipp.g.heilmann@agrar.uni-giessen.de

Full list of author information is available at the end of the article

Abstract

Background In plant breeding, many studies currently investigate the application of machine learning (ML) to genomic prediction, hoping for an improvement in prediction accuracy compared to standard models like Genomic Best Linear Unbiased Prediction (GBLUP). However, ML algorithms require much higher computational resources. This study aims to reduce the computational requirements and speed up training time by developing a novel autoencoder architecture inspired by haplotype blocks. Our approach incorporates prior knowledge on genetic linkage, inspired by haplotype block building, into the autoencoder architecture, resulting in a new encoded variable per haplotype block. We further modified our model into a semi-supervised version by adding available yield information. We used features extracted from the autoencoder's block layer as inputs for Random Forest and GBLUP models to predict the yield of hybrid and inbred crops.

Results Genomic prediction based on the extracted features maintained prediction accuracies equal to using the original marker data, even with a variable reduction of up to 98% and significantly reduced computation time. Prediction accuracy of the supervised component was in some cases equal to and in some lower than the prediction accuracy achieved using GBLUP. Effects estimated for haplotype block variants using our new method showed a high correlation to the blockwise sum of marker effects, which is the current standard approach for haplotype block effects. Correlation between the two block effect estimation approaches was very low for some blocks, which might indicate the incorporation of non-linear effects by the autoencoder.

Conclusions Our approach introduces a new perspective on processing haplotype blocks for genomic prediction, potentially providing more flexible modelling opportunities without the use of multiple binary dummy variables for each block variant. Additionally, training time for ML models may be significantly reduced by using the reduced feature sets generated using our method. By adding the semi-supervised component, the model is able to estimate values similar to marker effects

for each block on yield. In future work, this may provide a new way of quantifying the importance of haplotype blocks for selection and breeding.

Keywords Plant breeding, Machine learning, Genomic prediction, Neural networks, Autoencoders, Dimensionality reduction, Haplotype blocks

Introduction

With the introduction of marker-based prediction methods [1–3] and the development of high-throughput genotyping, plant breeding has increasingly complemented phenotyping with genotyping, allowing the selection and evaluation of untested genotypes. Methods such as genomic best linear unbiased predictor (GBLUP) provide robust frameworks for predicting the phenotypic values of traits based on genetic data. As field trials are expensive, accurately identifying the most promising genotypes greatly increases their efficiency and maximises the utility gained from the resources available. Therefore, research in plant breeding is heavily focused on developing new methods with higher prediction accuracies. To this day however, the original GBLUP proves to still be competitive and remains a highly reliable standard method that has not yet been consistently outperformed by other methods [4, 5].

Recently, machine learning (ML) has attracted attention in the field of plant breeding. The term ML encompasses a variety of algorithms that typically learn iteratively from training data. These algorithms can theoretically approximate any underlying function that describes the relationship between predictor variables and target variables within a dataset. Due to this inherent ability to also model nonlinear interactions, researchers hoped to find algorithms that would outperform GBLUP (a linear model by design). However, despite initial optimism, no ML approach has yet been shown to outperform traditional methods consistently in genomic prediction. While many studies demonstrate cases in which ML outperforms older approaches, such as GBLUP or Ridge Regression BLUP (RR-BLUP), its success appears to be species-, trait- and dataset-specific [4–10]. As the application of ML in plant breeding is still in its initial stage, this does not mean that ML cannot outperform classical methods, but rather that we need to develop more specific models and architectures for plant breeding [11].

However, ML algorithms have drawbacks that make them harder to train compared to linear mixed models for genomic prediction. While GBLUP generally uses a $n \times n$ genomic relationship matrix based on single nucleotide polymorphism (SNP) markers, where n is the number of genotypes, ML methods typically use the raw marker data, resulting in datasets of much higher dimensionality. Combined with the necessity of training many models for hyperparameter tuning and internal cross-validation [12], ML models often require much more training time and strong computer hardware. This slows down the application of ML in scientific studies compared to straightforward linear models. To address these issues, an initial study of 361 German winter wheat genotypes [8] was conducted, where we investigated the use of different sets of input variables, e.g. haplotype blocks and autoencoder features, to improve the training efficiency and increase the prediction accuracy of ML algorithms.

Haplotype blocks are said to have the potential to reduce the dimensionality of the dataset [13] and capture local epistatic effects [14]. To use blocks as features in prediction, every unique combination of marker alleles within a block (i.e. each variant of a block) is encoded using variables that represent the number of copies of each block

variant present in the genotype. When we used haplotype-based blocks as input features, we found that the prediction accuracy was comparable to or slightly higher than that of the full set of SNP markers for all algorithms for all traits except one [8]. However, due to the presence of many block variants, the haplotype-based blocks increased the dimension of the dataset rather than decreasing it [15], which further increased computation time.

As a second set of features, the output of the encoding layer of an autoencoder was used. Autoencoders are unsupervised neural networks optimized to encode a set of input features into a lower dimensional space in such a way that it is possible to reconstruct the original data from the encoded state [16]. Although this approach significantly reduced computation time, prediction accuracy declined for all traits and algorithms, presumably because the learned features were less informative than the original SNP markers [8].

Another study [17] proposed an alternative approach using a series of autoencoders instead of one autoencoder with fully connected layers. With this method, a fixed number of adjacent markers were grouped together and used as inputs for small autoencoders, which were then combined in a second stage to generate lower-dimensional representations for genomic prediction. This way, the prediction accuracy remained stable while the data dimension was reduced. The approach of grouping together a fixed number of adjacent markers is somewhat reminiscent of a very similar approach for building haplotype-based blocks using fixed window sizes. However, studies show that a fixed window size approach is usually not the best way of building blocks as prediction accuracy based on those is usually lower compared to predictions using blocks based on linkage disequilibrium (LD) [15, 18].

In this study, we build upon the method proposed in [17] by incorporating prior knowledge of the genetic linkage structure into the autoencoder structure. Instead of grouping markers based on a fixed window size, we define blocks using pairwise LD. Markers within a block are connected locally within the autoencoder, resulting in an architecture that more accurately reflects the underlying genetic structure. This approach enables each haplotype block to be reduced to a single unit in the encoding layer, i.e. a single variable after extraction, leading to a guaranteed dimensionality reduction compared to other haplotype block-based approaches.

Haplotype stacking has been proposed as a possible application for haplotype blocks in breeding [19]. However, this requires the estimation of effects for each block variant. An estimation method for effects and variances of haplotype blocks was first proposed in Voss-Fels et al.[19]. Effects of block variants are defined as the sum of the individual marker effects that belong to the same block. This approach has been adopted in other studies [20, 21]. Adding an output node of a target trait to our autoencoder architecture enables the model to estimate block variant effects by quantifying the contribution of a variant to the trait as the product of the outputs of the block layer and their respective weights. This provides a novel approach to estimating block variant effects beyond summing up individual effects. Additionally, it solves the problem of handling unseen block variants, which frequently occur in breeding programmes as newly generated material or crosses often contain variants that did not exist in the initial population.

The overall goal of this study was to develop an autoencoder-based method that integrates LD-based haplotype blocks into its architecture. By incorporating genetic linkage

information into the model, we aimed to improve the efficiency of dimensionality reduction for genomic prediction. Specifically, we aimed to: (1) characterise the features generated by the autoencoder during unsupervised training and evaluate their usefulness for genomic prediction; and (2) explore the potential of semi-supervised learning as an alternative approach for estimating the effects of variants within haplotype blocks.

Materials and methods

Materials

We applied our methodology to five datasets in total. Two of these datasets consisted of inbred/double haploid lines while the remaining three consisted of hybrids. All of the datasets contained yield data and genetic markers. We filtered the data, removing genotypes with more than 60% missing markers. Furthermore, we removed markers consisting of more than two alleles, with 10% or more missing values, or with an expected heterozygosity below 5%. If the remaining markers contained missing values, we imputed them using BEAGLE [22].

Dataset **Ra1** was provided by Norddeutsche Pflanzenzucht Hans-Georg Lembke KG and consisted of 746 rapeseed hybrids. After filtering 10,939 markers remained. Dataset **Mz1** was published previously [23] and accessed through the R package 'sommer 4.2.0' [24, 25]. It consisted of 1,254 maize hybrids. Genetic data was provided for the parental lines and 28,803 markers remained after filtering. Dataset **Mz2** is based on the data provided for the 2022 maize prediction competition hosted by the 'genomes 2 fields' project [26] and publicly available via the projects' website [27]. After preprossessing, it consisted of 4,689 maize hybrids and 267,818 markers for their respective parents. Dataset **Wh1** consisted of wheat lines which were generated from a factorial crossing design within the project MultiResistGS. After preprocessing, 293 lines and 17,221 markers remained. Dataset **Ot1** consisted of oat lines collected during the project FUGE, also from factorial crosses. After preprocessing, 234 lines and 29,457 markers remained. More information on the datasets is provided in Supplementary Table S1 and S2, and Supplementary Fig. S2.

For every dataset, adjusted entry means were either provided for each genotype or were calculated using an appropriate mixed linear model, factoring in environments and local field design. The specific mixed linear model design depended on the dataset.

Feature engineering

The haplotype blocks used in this study were built using the LD between adjacent markers with a threshold of 0.7 required for a marker to be assigned to a block. Pairwise LD between all markers was based on the r^2 measure [28]. We set a tolerance threshold of 1 marker within a block to be below this threshold.

The autoencoder that was used in this study was based on the generated haplotype blocks, as inputs are only connected locally within the borders of these blocks. Figure 1 displays a schematic illustration of an example autoencoder with this architecture. The unsupervised part of the autoencoder was used for feature engineering. It consisted of 4 to 5 layers, depending on the block size. Unsupervised learning refers to models that process input features without regard to any target trait Y. The input layer consisted of as many units as there were markers assigned to blocks. Blocks that only consisted of a single marker were not considered blocks and the markers were subsequently removed

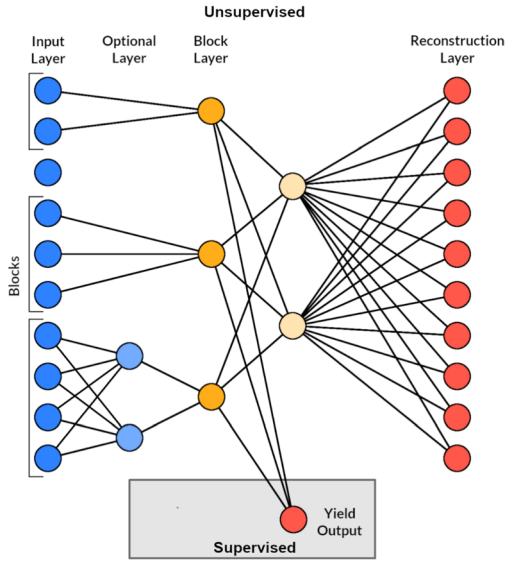

Fig. 1 Illustration of the autoencoder architecture

from the data input. The input layer was locally connected to either all units of an optional hidden layer or to the block layer. If a block contained ≥ 4 markers, an additional hidden layer was added between the input and the block layer to introduce additional non-linearity and modeling capabilities ("optional layer" in Fig. 1). The optional hidden layer consisted of $\lfloor \frac{n}{2} \rfloor$ units, where n was the size of the block. The output of those $\lfloor \frac{n}{2} \rfloor$ units was passed on to the block layer through a leaky ReLU activation function with a negative slope of 0.1. Local connections were limited to markers within the same block, resulting in one single unit per block in the block layer. The block layer was then fully connected to a hidden layer with 1000 units using the same leaky ReLU activation function as before. The hidden layer was connected to the reconstruction layer, which acted as the output of our unsupervised model. This layer returned the reconstructed marker data, which included markers that were previously removed due to a lack of block assignment. As the markers were encoded as -1, 0, and 1 to represent homozygosity for the minor allele, heterozygosity, and homozygosity for the major allele, respectively, the reconstruction layer used a tanh activation function that returned values in the range from -1 to 1.

The autoencoder was trained for 100 epochs. Model parameters were optimized using the Adam optimizer with a learning rate of 0.001 and the mean squared error (MSE) as the loss function. After training, the outputs of the block layer were extracted (first reduction step, AE1). As those were highly correlated, a second dimensionality reduction step (AE2) was added. The block layer output of the first autoencoder was used as the input for a new autoencoder with the same architecture which was then again trained for 100 epochs. Since it was not possible to form LD-based blocks in this scenario, blocks that showed a pairwise correlation >0.7 were grouped together to form 'meta blocks'. This is analogous to [17], where the same procedure was also applied a second time to the already reduced data. Model training was only needed once before the actual cross-validation, since the features could be used in all subsequent cross-validation runs as no phenotypic value was required at this stage.

As an additional reference, principal component analysis (PCA) was used to create principle components as input features, as this is a common and widespread method.

Genomic prediction

The first model we used to evaluate the influence of dimensionality reduction was a GBLUP. We used two different models for fitting the GBLUP, depending on the type of the crop. For datasets consisting of $i = 1, ..., n$ homozygous lines we used the model

$$\mathrm{y} = 1\beta_0 + \mathrm{Zu} + \mathrm{e}. \tag{1}$$

The vector y is the response vector with the phenotypic observations of the $i = 1, ..., n$ inbred lines, 1 is an $n \times 1$ column vector of ones, β_0 is a fixed intercept, u is a vector of normally distributed random genotypic values with $\mathrm{E(u)} = 0$ and $\mathrm{cov(u)} = \mathrm{G}\sigma^2_A$, 0 is an $n \times 1$ column vector of zeroes, and Z is the design matrix for the genotypic values. In models where y contains the adjusted treatment means of a field trial, Z is the identity matrix I_n. e is a vector of randomly distributed residuals with $\mathrm{E(e)} = 0$ and $\mathrm{cov(e)} = \mathrm{I}_n\sigma^2_R$. I_n is an identity matrix and σ^2_R is the residual variance.

For hybrids, we used a model that included the general combining ability (GCA) and specific combining ability (SCA) as described in [29] and applied in [23]:

$$\mathrm{y} = 1\beta_0 + \mathbf{Z_1 u_1} + \mathbf{Z_2 u_2} + \mathbf{Z_s u_s} + \mathrm{e}. \tag{2}$$

The vectors y, 1, β_0, and e are defined as stated above, this time for the $i = 1, ..., n$ hybrids. $\mathbf{u_1}$ and $\mathbf{u_2}$ are vectors of normally distributed random GCA effects of parents 1 and 2, respectively, with $\mathrm{E}(\mathbf{u_1}) = \mathrm{E}(\mathbf{u_2}) = 0$, $\mathrm{cov}(\mathbf{u_1}) = \mathbf{G_1}\sigma^2_1$, and $\mathrm{cov}(\mathbf{u_2}) = \mathbf{G_2}\sigma^2_2$, $\mathbf{u_s}$ is a vector of normally distributed random SCA effects associated with the specific combinations of parents 1 and 2, with $\mathrm{E}(\mathbf{u_s}) = 0$ and $\mathrm{cov}(\mathbf{u_s}) = \mathbf{G_s}\sigma^2_s$, and the design matrices $\mathbf{Z_1}$, $\mathbf{Z_2}$, and $\mathbf{Z_s}$ link the observations in y to the corresponding GCA and SCA effects.

The genomic relationship matrices G, $\mathbf{G_1}$ and $\mathbf{G_2}$ were calculated using method I of VanRaden [30]. $\mathbf{G_s}$ is defined as the Kronecker product $\mathbf{G_1} \otimes \mathbf{G_2}$. The variance components were estimated by restricted maximum likelihood (REML).

As an additional reference to evaluate the influence of dimensionality reduction in the context of ML, we used Random Forests [31]. We trained Random Forest models based on 20 different sets of hyperparameters, generated using a maximum entropy grid [32, 33]. Each set of hyperparameters was evaluated using a 5-fold cross-validation based on the training set. We tuned the number of trees ([100, 1000]), the column sampling rate ($[\frac{ncol}{100}, \frac{ncol}{3}]$) and the minimum observations required for an additional split ([1, 20]). We then created a stacked ensemble based on the models trained for every hyperparameter combination [34]. We used the LASSO algorithm [35] as the super learner for the new model.

Estimation of haplotype block effects

Baseline block effects were calculated using the sum of individual marker effects belonging to the same block [19].

To estimate block effects using the autoencoder, the block layer was directly connected to an additional unit that returned a prediction for yield ("yield output" in Fig. 1), which added a supervised part to the autoencoder. While both supervised and unsupervised

parts of the model were optimized simultaneously, the supervised part was only optimised on the basis of observations included in the training set while the unsupervised part of the model was optimised on the full data set. This combination of supervised and unsupervised, or labelled and unlabelled, data is known as semi-supervised learning [36].

For the semi-supervised form, the autoencoder used a custom loss function

$$\mathrm{MSE}(X, \hat{X}) - \mathrm{cor}(y, \hat{y}) + (\mathrm{MSE}(y, \hat{y}) + \lambda ||w_Z||^2) \tag{3}$$

where X represents the true markers and $\hat{X}$ the reconstructed markers, y represents the true yield and $\hat{y}$ the predicted yield, and λ is the ridge parameter of the squared ℓ_2-norm of the parameters of the block layer w_Z. The term $\mathrm{MSE}(X, \hat{X})$ is the loss function for the unsupervised part of the autoencoder. It represents the difference between the output of the autoencoder, which are the reconstructed markers, and the true markers that were used as inputs. By minimizing this error, the model learns to form a meaningful latent representation of the marker data in the block layer. Minimizing the negative correlation between the observed and predicted yield is equal to maximizing the same correlation. It is important that the rankings of observed and predicted phenotypes match, especially for the best observed genotypes, as these are the ones selected in a breeding program. Since the correlation is scale-free, the term $(\mathrm{MSE}(y, \hat{y}) + \lambda ||w_Z||^2)$ is introduced which minimizes the MSE between predicted and observed trait values. This ensures that predictions are on the same scale as the observed values. Similar to the ridge parameter in methods like RR-BLUP, $\lambda ||w_Z||^2$ leads to a shrinkage of the parameters connecting the block layer and the supervised part. This shrinks the parameters of blocks with lesser importance for the final prediction. In addition, the constraint encourages the model to rely more on the shared block representation, thereby stabilizing training and reducing the risk of overfitting in the supervised part. We set λ to 0.001 for every dataset.

Evaluation

A Mantel test [37] was used to assess the autoencoders ability to capture the genomic information in a lower dimensional space. A Mantel test compares the similarity between the distance matrices of the target matrices, in our case the genomic relationship matrices generated from the full SNP data and both reduced feature sets. Genomic relationship matrices were calculated according to method I of VanRaden [30].

To assess the suitability of our data for genomic prediction, we used GBLUP and RF based on the full marker data as the reference models. The features from the first and second reduction step were rescaled to be between -1 and 1 to compute the genomic relationship matrices used in GBLUP as this format was required by the software. The RF model used untransformed features.

For cross-validation, 100 training and test sets based on 80%/20% of the data were created and tested with each combination of method (GBLUP and RF) and feature set (full SNPs, reduction step 1 and reduction step 2). Additionally, 100 training and test sets were created for data sets of hybrid crops (Ra1, Mz1, Mz2) for which the test sets only consisted of T0 hybrids, i.e. hybrids where both parental lines did not appear in the training set. The prediction accuracy of a model was defined as the correlation between the observed and predicted yield. Prediction accuracy and computation time were recorded for every cross-validation run.

Table 1 Absolute and relative dimension of the full set of SNPs (SNP), the output of the first autoencoder (AE1) and the second reduction step (AE2) for all datasets. Additionally, amounts of markers with no block assignment are listed

	SNP	AE1	AE2	Unassigned SNPs
Ra1	10939 (100%)	1458 (13.33%)	177 (1.62%)	1472
Mz1	28803 (100%)	4140 (14.37%)	534 (1.85%)	14061
Mz2	267818 (100%)	34643 (12.94%)	3819 (1.43%)	106500
Wh1	17221 (100%)	2353 (13.66%)	427 (2.48%)	5806
Ot1	29457 (100%)	2890 (9.81%)	432 (1.47%)	3090

Table 2 Mantel test of similarity between genomic relationship matrices derived from the full set of SNPs (SNP), the output of the first autoencoder (AE1) and the second reduction step (AE2) for all datasets. Results are provided in the form: *r*-value (*p*-value). The *r*-value stands for the correlation and ranges from -1 to 1 where 1 is perfect correlation

	SNP vs AE1	SNP vs AE2	AE1 vs AE2
Ra1	0.7559 (<0.001)	0.7176 (<0.001)	0.9669 (<0.001)
Mz1	0.9988 (<0.001)	0.9834 (<0.001)	0.9886 (<0.001)
Mz2	0.9499 (<0.001)	0.9348 (<0.001)	0.9855 (<0.001)
Wh1	0.7907 (<0.001)	0.8159 (<0.001)	0.9807 (<0.001)
Ot1	0.5322 (<0.001)	0.5392 (<0.001)	0.9419 (<0.001)

To explore the potential of semi-supervised autoencoders for genomic prediction and variant effect estimation, the prediction accuracy of the semi-supervised version of the autoencoder was compared to the standard GBLUP model using the same initial 100 training and test sets. Additionally, the haplotype block variant effects derived from the autoencoder were compared to the sum of the single marker effects belonging to the same block. The marker effects were derived from the GBLUP. As GBLUP and RR-BLUP are mathematically equivalent [38, 39], it is possible to transform the random effects of the genotypes into marker effects.

Software

Autoencoders were computed using Python 3.12 and PyTorch 2.5.1 [40] using an NVIDIA®Quadro RTX™4000 GPU. All other calculations were done using R 4.3.3 [41] on two Intel®Xeon®Platinum 8276 CPU. Marker filtering and haplotype block building was done using routines which we programmed in the C and R programming languages. Mantel test was done using the R package 'vegan' [42]. Genomic relationship matrices and GBLUP were calculated using 'sommer 4.2.0' [24, 25], for RF we used the 'tidymodels' framework [43] with 'ranger 0.16.0' [44]. Due to its large size, GBLUP for dataset Mz2 was calculated using ASReml 4.2.0.267 [45].

Results

Dimensionality reduction for genomic prediction

The magnitude of the dimension reduction is shown in Table 1. In general, the dimension of the features was reduced to 9-15% of the original dimension, where all datasets fell in the range of 13-14% with the exception of Ot1 (∼10%). After the second reduction step, only around 1-3% of the original dimensionality remained.

A Mantel test was used to compare the similarity between all the genomic relationship matrices (Table 2). All genomic relationship matrices had a significant *r*-value (*p*-value < 0.001) which indicated a high correlation between all tested matrices. The Mantel test

for comparing the full set of SNPs to the first autoencoder features showed high variability between the datasets. While the two maize datasetes showed very high similarity (values above 0.9), the oat dataset showed much lower similarity (around 0.5). The same pattern could be observed for the comparison of the SNPs to the features of the second autoencoder. The *r*-values generally showed the same tendencies as the previous test, with only small differences. Features AE1 and AE2 seemed to result in very similar genomic relationship matrices as the *r*-values were very high for all datasets (>0.9).

Results of the cross-validation for hybrids are shown in Fig. 2. For hybrids, the reduction in the data dimensionality did not lead to a decrease in prediction accuracy. In the cross-validation scenario where parents of the hybrids in the test set also appeared in the training set (T1/T2 hybrids), prediction accuracy was equivalent for different feature sets within algorithms. In the cases of Ra1 and Mz1, prediction accuracy was also equivalent between algorithms while RF performed slightly worse for dataset Mz2. In the case of the T0-scenario, prediction accuracies within algorithms showed slightly more variability between feature sets. The changes in the prediction accuracy were still very minor, with the only exception being GBLUP for Ra1, where prediction accuracy improved from 0.1 to 0.14 and 0.2 with the first and second reduction step. This was not observed for RF for the same dataset. Similar results were observed for the inbred datasets (Fig. 2). For lines, the prediction accuracy was equal between the two algorithms and across feature sets. Prediction accuracies were generally higher than when principle components were used as inputs (Figure S3).

The GBLUP was much faster to compute and computation time was mostly unaffected by the change in the data dimension (Table 3). The average computation time for one RF run, including grid search, was drastically reduced in both steps. While training the RF models with the full set of SNPs took several minutes minimum for all algorithms, even up to more than two hours per run in the case of Mz2, computation time of three of the

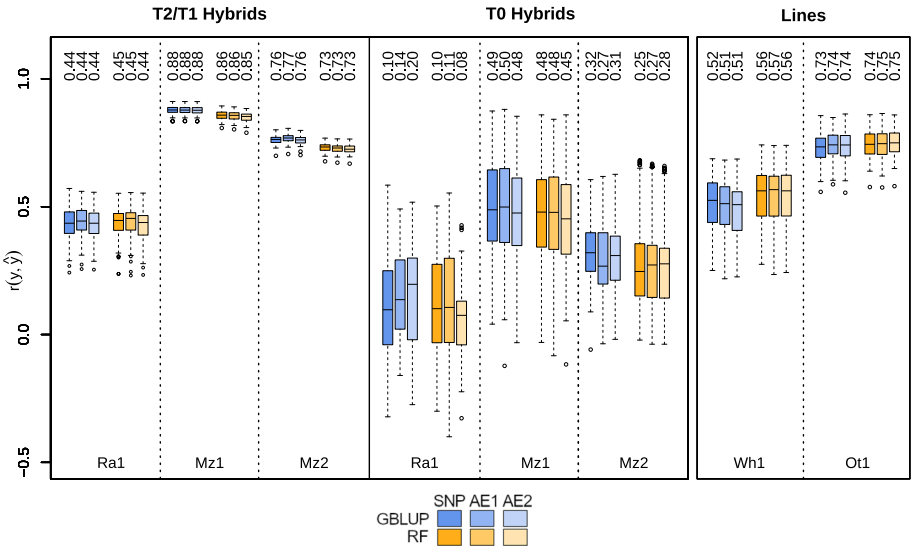

Fig. 2 Prediction accuracy of all cross-validation runs. Blue boxplots indicate GBLUP and orange indicate RF, wheares the reduced saturation indicates reduced feature sets. Numbers above the boxplots indicate the median prediction accuracy. Left shows T2/T1 hybrids, i.e. the 100 regular cross-validation runs where either one (T1) or both (T2) parental lines of the hybrids in the test set were available in the training set. The center shows 100 cross-validation runs where parental lines of hybrids in the test set were removed from the training set (T0). Plot on the right shows prediction accuracy for the cross-validation of the line datasets

Table 3 Mean computation time in minutes. GBLUP and RF refer to the methods used. SNP, AE1 and AE2 refer to the feature set used with each method and represent the full set of SNP markers (SNP), the output of the first autoencoder (AE1) and the second reduction step (AE2)

	GBLUP SNP	GBLUP AE1	GBLUP AE2	RF SNP	RF AE1	RF AE2
Ra1	0.23	0.08	0.09	4.19	0.94	0.20
Mz1	0.10	0.49	0.53	21.86	2.84	0.93
Mz2	7.37	7.22	9.77	162.49	34.32	4.35
Wh1	>0.01	>0.01	>0.01	2.64	0.57	0.18
Ot1	>0.01	>0.01	>0.01	3.98	0.79	0.21

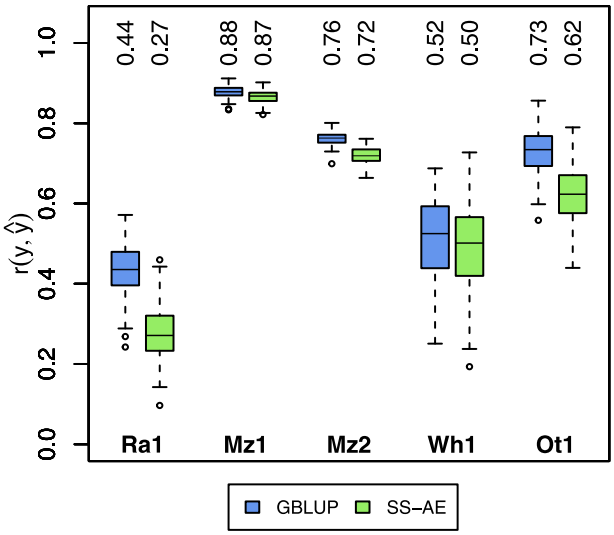

Fig. 3 Prediction accuracy of all cross-validation runs for all datasets. Green boxplots indicate the semi-supervised autoencoder (SS-AE), blue boxplots indicate GBLUP. Numbers above the boxplots indicate the medians of the prediction accuracy. For hybrids, training and test sets were identical to the T2/T1 scenario

datasets (Ra1, Ot1, Wh1) was reduced to under one minute after the first reduction step, and Mz1 requiring under one minute after the second reduction step. For dataset Mz2, a reduction from more than two hours to 30 to 4 min after the first and second reduction steps could be observed. Computation time after the second reduction step was even faster than the GBLUP for the same dataset with the full SNP data.

Semi-supervised estimation of block effects

We observed varying differences in the prediction accuracy between the semi-supervised autoencoder and the GBLUP (Fig. 3). Generally, the semi-supervised autoencoder performed worse than the GBLUP. While prediction accuracy for both methods was equal for datasets Mz1 and, to some degree, Mz2 and Wh1, the prediction accuracy of the autoencoder was lower compared to GBLUP for datasets Ra1 and Ot1.

We compared the haploblock variant effects estimated by the semi-supervised autoencoder to the marker effects estimated by GBLUP, summed up for every block. Figure 4 shows the distribution of the correlation between those variant effects for each block. The figure is exemplary and based on the effects estimated for the first cross-validation run for every dataset. The overall tendency was towards a positive correlation between both methods. However, different patterns in the distribution of the correlations could be observed. While Ra1 showed a tendency towards having more observations in the tails of the distribution, it also showed some similarity to a left-skewed distribution. The

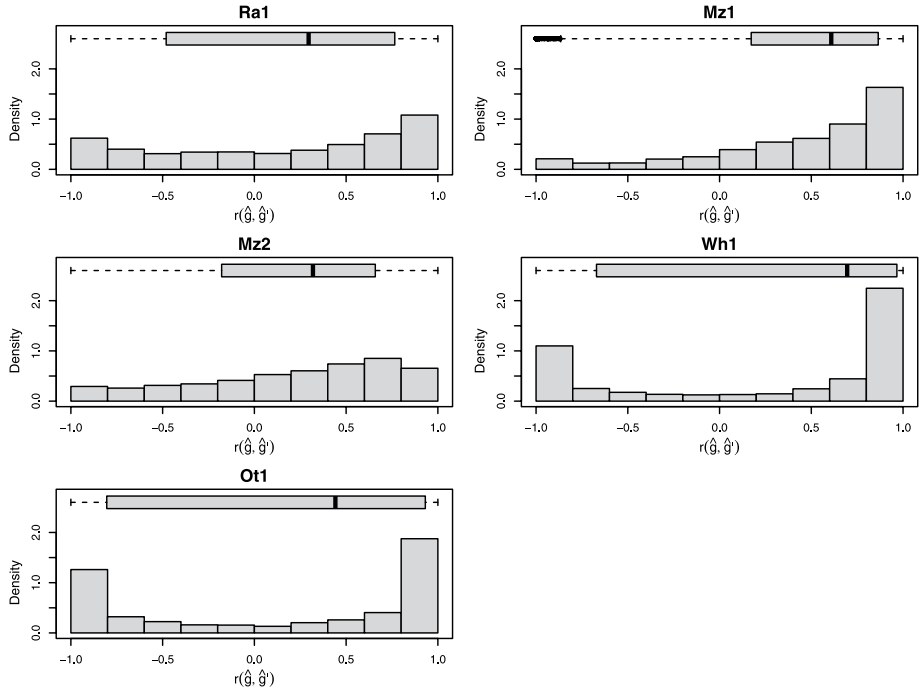

Fig. 4 Histograms and boxplots showing the distribution of correlation between effects of encoded blocks ($\hat{g}$) and summed up marker effects ($\hat{g}'$). For each block individually, correlation was recorded between autoencoder-based variant effects and marker effects summed up for every variant

distributions for Mz1 and Mz2 showed a clear left-skewed distribution with relatively few of the observed correlations falling below 0 and with each median correlation above 0.5. For the line datasets Wh1 and Ot1, the distribution showed a u-shape with most of the blocks falling into either a nearly perfect positive or negative correlation, with a stronger tendency towards a positive correlation.

A direct comparison of block variant effects based on the autoencoder and the block-wise sum of the marker effects is shown in Fig. 5. The observed values were scaled for both approaches due to the absence of an intercept in the autoencoder, resulting in larger effect sizes compared to those generated using GBLUP. The overall trends in effects were comparable, but the distribution of block variant effects was more symmetrical around 0 for single markers. For the encoded effects, there was a stronger tendency toward positive effects (above 0), while fewer negative effects, especially strong negative effects, were observed. Effects from the autoencoder showed some extreme positive values, relative to which most other effects appeared relatively small. Marker-based effects were more evenly distributed across all blocks, with fewer extreme values.

Discussion

Autoencoders for dimensionality reduction

In this study, we developed a novel autoencoder architecture based on the concept of haplotype-based blocks to compress high dimensional genetic data while preserving its core information content. Our first goal was to assess how much relevant information is retained within the encoded features extracted from the autoencoder and to determine their usefulness as inputs for prediction models. The Mantel test on the different feature sets showed a high similarity between the genomic relationship matrices generated with

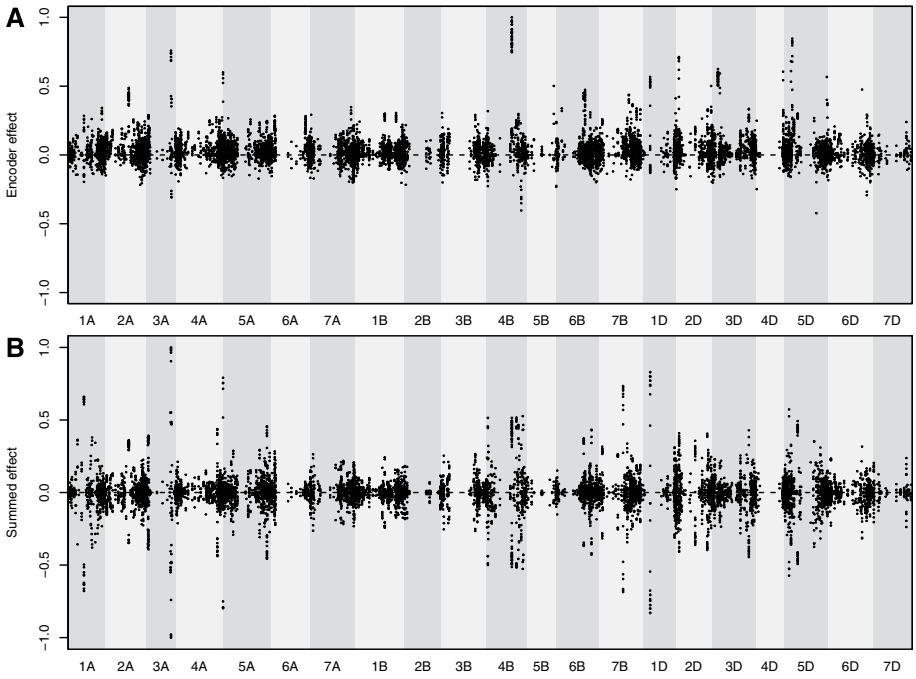

Fig. 5 Comparison of effects estimated for each haplotype block variant using the semisupervised autoencoder **A** to the block-wise sum of effects estimated using a GBLUP **B**. Estimation is based on the first cross-validation run for dataset Wh1. Differently highlighted areas indicate different chromosomes. Position on the x-axis indicates physical distance between blocks. The y-axis was scaled for both methods individually through dividing all effects by the largest effect of the respective method, thus scaling all effects from -1 to 1

those features (Table 2). While all genomic relationship matrices were significantly correlated with each other, the *r*-value between full and reduced data was comparatively low for dataset Ot1. For the maize datasets, the similarity was nearly perfect between full and reduced data. We conclude from this that the majority of genetic relationships contained within the marker data are retained during dimensionality reduction. Additionally, the genomic relationship matrices based on the first and second reduction steps were highly similar in all cases. This implies that reducing the already compressed data further does not result in the loss of any more information regarding the relationship structure in our data. This is reaffirmed by the consistency in the observed prediction accuracy for all datasets and algorithms. Compared to a widespread dimensionality reduction method, PCA, prediction accuracies of RF based on the autoencoder features were also higher (Figure S3). In all cases, prediction accuracy remained largely unaffected by the change in the dimensionality (Fig. 2). Through the dimensionality reduction of approximately 85 to 90% in the initial step and up to 98% in the second step (Table 1), computation time for ML algorithms was drastically reduced (Table 3). Computation time for GBLUP was unaffected as the dimension of the genomic relationship matrices depends only on the number of genotypes tested. Fluctuations in computation time may be related to a higher workload on the server side or other factors beyond the scope of this study.

In summary, we generated features in which each haplotype block is represented by a single variable, and the variables representing the blocks contain most, if not all, of the information from the full marker data. As our model guarantees dimension reduction, it overcomes the typical problem of an increased dataset dimension encountered with haplotype blocks [15]. This may help to avoid the potential problem of decreased

prediction accuracy due to multicollinearity when there are many block variants [46]. We conclude that locally connecting markers belonging to the same blocks based on flanking LD in an autoencoder is an effective way to reduce the dimensionality of genetic data. Our method therefore solves the problems faced in an earlier study [8], where autoencoders did not seem to be useful when using a standard architecture based on fully connected layers, as the prediction accuracy declined drastically. By reducing the number of features in the dataset training models for RF and other ML algorithms takes much less time and uses fewer computational resources. This can be advantageous in many situations, such as when computation time or model size are critical factors or related to costs (e.g. if rented cloud computing space is used or local space is limited), or during hyperparameter tuning, which is an integral part of ML [12]. A smaller model could be used to quickly find optimal regions in a very large hyperparameter space or create large ensembles of models. This is important as ensembles of models often perform better than individual models [7, 47, 48], especially when the ensemble is made up of diverse models [49]. The additional cost of training the unsupervised autoencoder is comparatively low as it only has to be trained once and does not require phenotypic measurements. Similar to the popular ML approach of transfer learning [50], a fully trained model can be stored and fine-tuned for each new cycle of a breeding program using newly generated data or for modeling new traits closely related to the original task in its semi-supervised form. This process allows the model to continually improve and adapt to the evolving germplasm within the breeding program.

Autoencoders are rarely used for genomic prediction in plant breeding. Existing studies focus on 2- and 3-dimensional [51, 52] or hyperspectral data [53, 54]. Tross et al. [54] used autoencoders to reduce the dimension of hyperspectral data, where some autoencoder features also showed similarity to known genes influencing leaf phenotypes. To our knowledge, the study that comes closest to ours is Islam et al [17]. Our findings regarding prediction accuracy are in accordance with the findings in their [17] study. However, our method has some key differences. The most notable is that we used a single autoencoder to train the full model, whereas Islam et al [17] used a series of small autoencoders. The equivalent to this would be training an individual autoencoder for each block, which our method can relatively easy be adapted to. However, as we have fully connected layers in the decoder part and also include the markers without block assignment in the output layer, we argue that having a single autoencoder comes with the advantage of the model learning the interconnectedness between blocks and all markers. This can be important as one possible reason for markers not being assigned to a block is that their physical position is not determined correctly and therefore actual existing blocks are broken up. We assume that in our case, a unit in the block layer would also be a good predictor for markers with wrong physical positioning and this information is therefore also accounted for in the model. Secondly, Islam et al. [17] chose to one-hot encode markers, therefore quadrupling the amount of input variables. The advantage is that their approach is not limited to biallelic SNP markers. However, as most markers are biallelic, we believe that this increases computation time unnecessarily in most cases.

Haplotype block effects

Our second goal was to enhance our model to create a novel approach of estimating the effects of haplotype block variants that goes beyond simply adding up the local effects of

markers. For this, we extended our model to include a yield output layer which was used only for those training samples that were in the training set, while the unsupervised part was using all of the data. This concept is called semi-supervised learning [36] and to our knowledge, this approach has not ben utilized in plant breeding before. The advantage of semi-supervised learning is that it uses all of the data in the model. In supervised models, genetic data without phenotypic data is usually excluded completely during model training. This is important because it is often hypothesised that datasets in plant breeding might be too small for ML to work in general [4, 11, 55]. By maximising data usage, breeders can get the most out of the available data.

Comparing GBLUP and semi-supervised autoencoders, the results showed that GBLUP was performing better, albeit by a smaller (Mz1, Mz2, Wh1) or larger (Ra1, Ot1) margin (Fig. 3). One possible explanation is that we used the same approach for every dataset. Typically, the number of epochs, the learning rate, and architectural details such as the size of the hidden layer before the output layer are adapted for each dataset. In our study we prioritised creating a general framework and prove the feasibility of a novel method of estimating haplotype block effects instead of working on fine-tuning the model for every dataset as this would have unnecessarily increased the complexity of the methodology. It is reasonable to assume that prediction accuracy would further improve with dataset-specific parameters.

Some authors argue that Neural Networks may not be suitable for estimating marker effects when used for genomic prediction [56]. However, by incorporating prior knowledge about genetic linkage into our model architecture and a direct connection between block and yield output layer, we ensure that the effects are, by design, in the same unit as the target variable. The effects derived from the autoencoders and the block-wise sums of marker effects [19] were often highly correlated (Fig. 4), especially for datasets Wh1 and Ot1. This agreement between the methods confirms that the autoencoder effects are plausible and meaningful. However, since there were also many cases where the correlation was lower, it appears that the autoencoder identified alternative regions with important contribution to estimating the final yield. This may indicate areas where non-additive effects may play a role. It has been shown that haplotype blocks capture local epistatic effects to some degree [14] although with varying success [57]. As we optimize our model towards two objectives (yield and the reconstructed marker data) and pass the inputs through multiple layers with non-linear activation functions, our model is designed in a way that could help capture local epistatic effects. While this was not directly tested here, future work could explore whether the model can also capture more complex non-linear interactions beyond local epistasis, such as global epistatic effects. Training on a combination of labelled and unlabelled data may also act as a form of regularisation, leading to better model generalisation [58]. Our approach therefore provides an alternative method of estimating the effects of haplotype blocks directly on the variant level instead of relying on estimating effects for markers first. This could be used for proposed methods like "haplotype stacking", where the goal is to bring favourable combinations of blocks together [19, 21], as this is currently relying on block-wise sums of marker effects.

Limitations

A current limitation of our model is that, due to the use of a (-1, 0, 1) marker encoding, some haplotype block variants are effectively lost for hybrid datasets. This encoding does not take into account on which chromosome an allele is positioned. The following example (Figure S1) illustrates the problem: The cross between two homozygous parental lines where one has the block variant CCAC and the other CAAA would result in a block variant encoded as (-1, 0, 1, 0). Another cross between two different homozygous parents with the variants CAAC and CCAA would equally result in a hybrid with marker data (-1, 0, 1, 0). As a result, our models assigns the same effect to some block variants that are functionally different. This problem does not occur in the case when lines are predicted as they are homozygous.

While this is a constraint when estimating block effects for hybrids, this problem can be circumvented in future studies. A potential solution would be to provide a tuple of the two alleles (i.e. two variables) instead of a single binary variable. This would effectively double the size of the input and unsupervised output layers, while the principle of the architecture remains unchanged. The idea of representing SNPs with two codes, typically one for additive and one for dominance effects, has long been used in quantitative genetics and genomic prediction [30]. Recent implementations in efficient, multilocus mixed models [59, 60] demonstrate the continued relevance of this approach for GWAS. A similar extension based on design or genomic relationship matrices could also be applied in our framework by using two separate matrices to represent the two parental haplotypes in a block, thereby addressing the problem of ambiguous encodings. Adjusting our method would be required if a diverse set of genotypes with a high rate of heterozygosity is used. Any aspect related to our first objective was not affected as the primary goal was dimensionality reduction while maintaining a stable prediction accuracy. Therefore, the actual values of the features are not interpreted.

Conclusion

Our method overcomes the limitations of autoencoders that were encountered in previous studies, where prediction accuracy decreased when features generated using an autoencoder where used as inputs in another model. The novel haplotype-based autoencoder successfully compresses high-dimensional genetic data while preserving genetic relationships, as shown by the high similarity in relationship information and consistent prediction accuracy compared to the standard GBLUP model. This dimensionality reduction significantly decreases computation time for ML algorithms, making them advantageous for scenarios where computational efficiency is important or extensive hyperparameter tuning is required. Additionally, the semi-supervised learning approach offers a new way of estimating the effects of haplotype blocks that goes beyond simply adding up the individual marker effects for each block.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-025-06323-w.

Supplementary file 1.

Author contributions

PGH conceived and conducted the study and prepared the manuscript. EG analysed the field data for oat. MH and CZP conceived the design for the project in which the oat data were generated. MH carried out the crossings and selfings

for oat. MH, MM and RA collected the genotypic data for oat. SB, VK, KO, and IR collected the field data for oat. AH, RM, CR, ASe, and MS carried out the factorial crossings for wheat. AH, RRM, CR, and MS collected the field data for wheat. AA and TK conceived the design for the project in which the data for rapeseed were generated. MF and ASt conceived the design for the project in which the wheat data were generated.

Funding
Open Access funding enabled and organized by Projekt DEAL. The project SEEH in which part of the oat data was generated is supported by funds of the Federal Ministry of Agriculture, Food and Regional Identity (BMLEH) based on a decision of the parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the Federal Organic Farming Scheme (FKZ 2822OE188). The projects FUGE and MultiResistGS in which part of the oat data and the wheat data were generated were supported by funds of the Federal Ministry of Agriculture, Food and Regional Identity (BMLEH) based on a decision of the parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support programme (FKZ 28AINO2A20 and FKZ 28A8411A18). Ra1 data were generated as part of the Project PROGReSs (BMBF, FKZ 031A297A-J), which was funded by the German Federal Ministry for Education and Research (BMBF).

Data availability
The datasets Ot1 and Wh1 were originally generated within the research projects SEEH, FUGE, and MultiResistGS. Datasets Ot1 and Wh1, together with the R and Python scripts used in this study, are available in the GitHub repository: https://github.com/PGHeilmann/Haploencoder. Dataset Mz1 is accessible through the R package "sommer" and was originally published in Technow et al. (2014). The dataset Mz2 was obtained from the Genomes to Fields (G2F) initiative and is publicly available through the official project website: https://www.genomes2fields.org/resources. Dataset Ra1 was provided by NPZ Innovation GmbH for internal use and restrictions apply to the availability of these data.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
Steffen Beuch is employed by Nordsaat Saatzucht GmbH. Vivek Kurra and Raja Mehta are employed by Saatzucht Bauer GmbH & Co. KG. Ina Röhrs is employed by Landbauschule Dottenfelderhof e.V. Klaus Oldach is employed by KWS Lochow GmbH. Anja Hanemann is employed by Saatzucht Josef Breun GmbH & Co. KG. Carsten Reinbrecht and Marco Stucke are employed by Saatzucht Streng-Engelen GmbH & Co. KG. Amine Abbadi and Tobias Kox are employed by NPZ Innovation GmbH. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential Conflict of interest.

Author details
[1]Institute of Agronomy and Plant Breeding II, Justus Liebig University, Heinrich-Buff-Ring 26-32, 35392 Giessen, Germany
[2]Research on Agricultural Crops, Federal Research Centre for Cultivated Plants, Institute for Breeding, Julius Kühn Institute, Rudolf-Schick-Platz 3a, 18190 Sanitz, Germany
[3]Saatzucht Granskevitz, Nordsaat Saatzucht GmbH, Granskevitz 3, 18569 Schaprode, Germany
[4]Saatzucht Bauer GmbH & Co. KG, Landshuter Str. 3a, 93083 Obertraubling, Germany
[5]Leibniz Institute of Plant Genetics and Crop Plant Research (IPK) Gatersleben, Corrensstraße 3, 06466 Seeland, Germany
[6]KWS Lochow GmbH, Ferdinand-von-Lochow-Str. 5, 29303 Bergen, Germany
[7]Research & Breeding Dottenfelderhof, Landbauschule Dottenfelderhof e.V., Dottenfelderhof 1, 61118 Bad Vilbel, Germany
[8]Saatzucht Josef Breun GmbH & Co. KG, Amselweg 1, 91074 Herzogenaurach, Germany
[9]Saatzucht Streng-Engelen GmbH & Co. KG, Aspachhof, 97215 Uffenheim, Germany
[10]Institute for Resistance Research and Stress Tolerance, Julius Kühn Institute, Erwin-Baur-Str. 27, 06484 Quedlinburg, Germany
[11]NPZ Innovation GmbH, Hohenlieth-Hof, 24363 Holtsee, Germany

Received: 4 July 2025 / Accepted: 7 November 2025
Published online: 02 December 2025

References

1. Bernardo R. Genetic Models for Predicting Maize Single-Cross Performance in Unbalanced Yield Trial Data. Crop Science. 1995;35(1) https://doi.org/10.2135/cropsci1995.0011183X003500010026x.
2. Whittaker JC, Thompson R, Denham MC. Marker-assisted selection using ridge regression. Genet Res. 2000;75(2):249–52. https://doi.org/10.1017/S0016672399004462.
3. Meuwissen THE, Hayes BJ, Goddard ME. Prediction of total genetic value using genome-wide dense marker maps. Genetics. 2001;157:1819–29. https://doi.org/10.1093/genetics/157.4.1819.
4. Azodi CB, Bolger E, McCarren A, Roantree M, de los Campos G, Shiu SH. Benchmarking Parametric and Machine Learning Models for Genomic Prediction of Complex Traits. G3: Genes, Genomes, Genetics. 2019 11;9(11):3691–3702. https://doi.org/10.1534/g3.119.400498.

5. Zingaretti LM, Gezan SA, Ferrão LFV, Osorio LF, Monfort A, Muñoz PR, et al. Exploring deep learning for complex trait genomic prediction in polyploid outcrossing species. Front Plant Sci. 2020. https://doi.org/10.3389/fpls.2020.00025.
6. John M, Haselbeck F, Dass R, Malisi C, Ricca P, Dreischer C, et al. A comparison of classical and machine learning-based phenotype prediction methods on simulated data and three plant species. Front Plant Sci. 2022. https://doi.org/10.3389/fpls.2022.932512.
7. Heilmann PG, Frisch M, Abbadi A, Kox T, Herzog E. Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP. Front Plant Sci. 2023. https://doi.org/10.3389/fpls.2023.1178902.
8. Heilmann PG, Difabachew YF, Frisch M, Moritz AL, Stahl A, Wittkop B, et al. Machine learning for prediction of resistance scores in wheat (*Triticum aestivum* L.). Plant Breed. 2024;144(2):192–205. https://doi.org/10.1111/pbr.13235.
9. Lourenco VM, Ogutu JO, Rodrigues RA, Posekany A, Piepho HP. Genomic prediction using machine learning: a comparison of the performance of regularized regression, ensemble, instance-based and deep learning methods on synthetic and empirical data. BMC Genom. 2024;25(1):152. https://doi.org/10.1186/s12864-023-09933-x.
10. Montesinos-López A, Montesinos-López OA, Ramos-Pulido S, Mosqueda-González BA, Guerrero-Arroyo EA, Crossa J, et al. Artificial intelligence meets genomic selection: comparing deep learning and GBLUP across diverse plant datasets. Front Genet. 2025. https://doi.org/10.3389/fgene.2025.1568705.
11. Hayes BJ, Chen C, Powell O, Dinglasan E, Villiers K, Kemper KE, et al. Advancing artificial intelligence to help feed the world. Nat Biotechnol. 2023;41:1188–9. https://doi.org/10.1038/s41587-023-01898-2.
12. Probst P, Boulesteix AL, Bischl B. Tunability: importance of hyperparameters of machine learning algorithms. J Mach Learn Res. 2019;20(53):1–32.
13. Cuyabano BC, Su G, Lund MS. Genomic prediction of genetic merit using LD-based haplotypes in the Nordic Holstein population. BMC Genom. 2014;15(1):1171. https://doi.org/10.1186/1471-2164-15-1171.
14. Jiang Y, Schmidt RH, Reif JC. Haplotype-based genome-wide prediction models exploit local epistatic interactions among markers. G3: Genes, Genomes, Genetics. 2018;8(5):1687–99. https://doi.org/10.1534/g3.117.300548.
15. Difabachew YF, Frisch M, Langstroff AL, Stahl A, Wittkop B, Snowdon RJ, et al. Genomic prediction with haplotype blocks in wheat. Front Plant Sci. 2023. https://doi.org/10.3389/fpls.2023.1168547.
16. Goodfellow I, Bengio Y, Courville A. Deep Learning. Cambridge, MA.: MIT Press; 2016. Available from: http://www.deeplearningbook.org.
17. Islam T, Kim CH, Iwata H, Shimono H, Kimura A. DeepCGP: a deep learning method to compress genome-wide polymorphisms for predicting phenotype of rice. IEEE/ACM Trans Comput Biol Bioinf. 2023;20(3):2078–88. https://doi.org/10.1109/TCBB.2022.3231466.
18. Weber SE, Frisch M, Snowdon RJ, Voss-Fels KP. Haplotype blocks for genomic prediction: a comparative evaluation in multiple crop datasets. Front Plant Sci. 2023. https://doi.org/10.3389/fpls.2023.1217589.
19. Voss-Fels KP, Stahl A, Wittkop B, Lichthardt C, Nagler S, Rose T, et al. Breeding improves wheat productivity under contrasting agrochemical input levels. Nature Plants. 2019;5(7):706–14. https://doi.org/10.1038/s41477-019-0445-5.
20. Villiers K, Voss-Fels KP, Dinglasan E, Jacobs B, Hickey L, Hayes BJ. Evolutionary computing to assemble standing genetic diversity and achieve long-term genetic gain. Plant Genome. 2024;17(2):e20467. https://doi.org/10.1002/tpg2.20467.
21. Tong J, Tarekegn ZT, Jambuthenne D, Alahmad S, Periyannan S, Hickey L, et al. Stacking beneficial haplotypes from the Vavilov wheat collection to accelerate breeding for multiple disease resistance. Theor Appl Genet. 2024;137(12):274. https://doi.org/10.1007/s00122-024-04784-w.
22. Browning BL, Zhou Y, Browning SR. A one-penny imputed genome from next-generation reference panels. Am J Hum Genet. 2018;103(3):338–48. https://doi.org/10.1016/j.ajhg.2018.07.015.
23. Technow F, Schrag TA, Schipprack W, Bauer E, Simianer H, Melchinger AE. Genome properties and prospects of genomic prediction of hybrid performance in a breeding program of maize. Genetics. 2014;8(197):1343–55. https://doi.org/10.1534/genetics.114.165860.
24. Covarrubias-Pazaran G. Genome assisted prediction of quantitative traits using the R package sommer. PLoS ONE. 2016;11:1–15.
25. Covarrubias-Pazaran G. Software update: Moving the R package sommer to multivariate mixed models for genome-assisted prediction. biorxv. 2018
26. Lima DC, Washburn JD, Varela JI, Chen Q, Gage JL, Romay MC, et al. Genomes to fields 2022 maize genotype by environment prediction competition. BMC Res Notes. 2023;16(1):148. https://doi.org/10.1186/s13104-023-06421-z.
27. Genomes to Fields Initiative.: Resources. Accessed 20 May 2025. https://www.genomes2fields.org/resources/.
28. Zhao H, Nettleton D, Soller M, Dekkers JCM. Evaluation of linkage disequilibrium measures between multi-allelic markers as predictors of linkage disequilibrium between markers and QTL. Genet Res. 2005;86(1):77–87. https://doi.org/10.1017/S001667230500769X.
29. Stuber CW, Cockerham CC. Gene effects and variances in hybrid populations. Genetics. 1966;54(6):1279–86. https://doi.org/10.1093/genetics/54.6.1279.
30. VanRaden PM. Efficient methods to compute genomic predictions. J Dairy Sci. 2008;91(11):4414–23. https://doi.org/10.3168/jds.2007-0980.
31. Breiman L. Random forests. Mach Learn. 2001;10(45):5–32. https://doi.org/10.1023/A:1010933404324.
32. Shewry M, Wynn HP. Maximum entropy sampling. J Appl Stat. 1987;14:165–70.
33. Kuhn M, Frick H.: dials: Tools for Creating Tuning Parameter Values. R package version 1.2.1. Available from: https://CRAN.R-project.org/package=dials.
34. Wolpert DH. Stacked generalization. Neural Netw. 1992;5(2):241–59. https://doi.org/10.1016/S0893-6080(05)80023-1.
35. Tibshirani R. Regression shrinkage and selection via the lasso. J Roy Stat Soc: Ser B (Methodol). 1996;58(1):267–88.
36. Amini MR, Usunier N. Learning with Partially Labeled and Interdependent Data. Springer Cham; 2015.
37. Mantel N. The detection of disease clustering and a generalized regression approach. Can Res. 1967;27(2):209–20 (https://aacrjournals.org/cancerres/article-pdf/27/2_Part_1/209/2382183/cr0272p10209.pdf).
38. Shen X, Alam M, Fikse F, Rönnegård L. A novel generalized ridge regression method for quantitative genetics. Genetics. 2013;193(4):1255–68. https://doi.org/10.1534/genetics.112.146720 (https://academic.oup.com/genetics/article-pdf/193/4/1255/42118481/genetics1255.pdf.).

39. de Vlaming R, Groenen PJF. The current and future use of ridge regression for prediction in quantitative genetics. Biomed Res Int. 2015;2015:143712. https://doi.org/10.1155/2015/143712.
40. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. In: PyTorch: an imperative style, high-performance deep learning library. Red Hook, NY, USA: Curran Associates Inc.; 2019.
41. R Core Team.: R: A Language and Environment for Statistical Computing. Vienna, Austria. Available from: https://www.R-project.org/.
42. Oksanen J, Simpson GL, Blanchet FG, Kindt R, Legendre P, Minchin PR, et al.: vegan: Community Ecology Package. R package version 2.6-10. Available from: https://CRAN.R-project.org/package=vegan.
43. Kuhn M, Wickham H.: tidymodels: a collection of packages for modeling and machine learning using tidyverse principles. Available from: https://www.tidymodels.org.
44. Wright MN, Ziegler A. Ranger: a fast implementation of random forests for high dimensional data in C++ and R. J Stat Softw. 2017;77(1):1–17. https://doi.org/10.18637/jss.v077.i01.
45. The VSNi Team.: asreml: Fits Linear Mixed Models using REML. R package version 4.2.0.267.
46. Matias FI, Galli G, Correia Granato IS, Fritsche-Neto R. Genomic prediction of autogamous and allogamous plants by snps and haplotypes. Crop Sci. 2017;57(6):2951–8. https://doi.org/10.2135/cropsci2017.01.0022.
47. Liang M, Chang T, An B, Duan X, Du L, Wang X, et al. A stacking ensemble learning framework for genomic prediction. Front Genet. 2021. https://doi.org/10.3389/fgene.2021.600040.
48. Kick DR, Washburn JD. Ensemble of best linear unbiased predictor, machine learning and deep learning models predict maize yield better than each model alone. In silico Plants. 2023 09;5(2):diad015. https://doi.org/10.1093/insilicoplants/diad015.
49. Tomura S, Wilkinson MJ, Cooper M, Powell O. Improved genomic prediction performance with ensembles of diverse models. G3 Genes|Genomes|Genetics. 2025 03;15(5):jkaf048. https://doi.org/10.1093/g3journal/jkaf048.
50. Lu J, Behbood V, Hao P, Zuo H, Xue S, Zhang G. Transfer learning using computational intelligence: A survey. Knowledge-Based Systems. 2015;80:14–23. 25th anniversary of Knowledge-Based Systems. https://doi.org/10.1016/j.knosys.2015.01.010.
51. de Villiers HAC, Otten G, Chauhan A, Meesters L. Autoencoder-based 3D representation learning for industrial seedling abnormality detection. Comput Electron Agric. 2023. https://doi.org/10.1016/j.compag.2023.107619.
52. Jurado-Ruiz F, Rousseau D, Botia JA, Aranzana MJ. GenoDrawing: an autoencoder framework for image prediction from SNP markers. Plant Phenom 2023, https://doi.org/10.34133/plantphenomics.0113.
53. Powadi A, Jubery TZ, Tross MC, Schnable JC, Ganapathysubramanian B. Disentangling genotype and environment specific latent features for improved trait prediction using a compositional autoencoder. Front Plant Sci. 2024. https://doi.org/10.3389/fpls.2024.1476070.
54. Tross MC, Grzybowski MW, Jubery TZ, Grove RJ, Nishimwe AV, Torres-Rodriguez JV, et al. Data driven discovery and quantification of hyperspectral leaf reflectance phenotypes across a maize diversity panel. Plant Phenome J. 2024;7(1):e20106. https://doi.org/10.1002/ppj2.20106.
55. Montesinos-López OA, Montesinos-López A, Pérez-Rodríguez P, Barrón-López JA, Martini JWR, Fajardo-Flores SB, et al. A review of deep learning applications for genomic selection. BMC Genom. 2021;22:19. https://doi.org/10.1186/s12864-020-07319-x.
56. Ubbens J, Parkin I, Eynck C, Stavness I, Sharpe AG. Deep neural networks for genomic prediction do not estimate marker effects. Plant Genome. 2021;14(3):e20147. https://doi.org/10.1002/tpg2.20147.
57. Da Y, Liang Z, Prakapenka D. Multifactorial methods integrating haplotype and epistasis effects for genomic estimation and prediction of quantitative traits. Front Genet. 2022. https://doi.org/10.3389/fgene.2022.922369.
58. van Engelen JE, Hoos HH. A survey on semi-supervised learning. Mach Learn. 2020;109(2):373–440. https://doi.org/10.1007/s10994-019-05855-6.
59. Wang JT, Chang XY, Zhao Q, Zhang YM. FastBiCmrMLM: a fast and powerful compressed variance component mixed logistic model for big genomic case-control genome-wide association study. Brief Bioinform 2024 06;25(4):bbae290. https://doi.org/10.1093/bib/bbae290.
60. Wang J, Chen Y, Shu G, Zhao M, Zheng A, Chang X, et al. Fast3VmrMLM: a fast algorithm that integrates genome-wide scanning with machine learning to accelerate gene mining and breeding by design for polygenic traits in large-scale GWAS datasets. Plant Commun. 2025;6(7):101385. https://doi.org/10.1016/j.xplc.2025.101385.

Publisher's Note

Supplementary File 1: Haplotype-based Autoencoders Can Reduce the Dataset Dimension and Estimate Haplotype Block Effects in Different Crop Species

Supplementary Tables and Figures

Table S1 Genotypes generated for each cross between parental lines for dataset **Ot1**. Parent 1 was used as a resistance donor for *Fusarium spec.*. Parent 2 was a genotype with superior potential for yield. The remaining genotypes in the dataset were the parental lines themselves.

	Parent1					
Parent 2	Jaak	Keely	Odal	PGL228	Puhti	Zorro
	n					
Apollon	5		3		3	8
Armani		12			6	2
DCAs2PGL-253inH817					21	
Delfin	8	7	2		12	
HSH_PanFläti				19	21	
Max		7	15		9	18
Sy12/1					22	
Sy3512				4		
Symphony	2		7		4	7

Table S2 Genotypes generated for each cross between parental lines for Dataset **Wh1**. Parent 1 was a resistance donor for either *Zymoseptoria tritici*, *Pyrenophora tritici repentis*, or *Fusarium spec.*. Parent 2 was an elite breeding line.

	Parent1				
Parent 2	20812.2.2	XX41	XX45	Stb19/Lorikeet	HTRI1410
	n				
Asory	15			16	38
Informer	1			16	9
Kamerad	29			21	1
LG Initial	23	9	4	41	11
LG Mocca	22			20	17

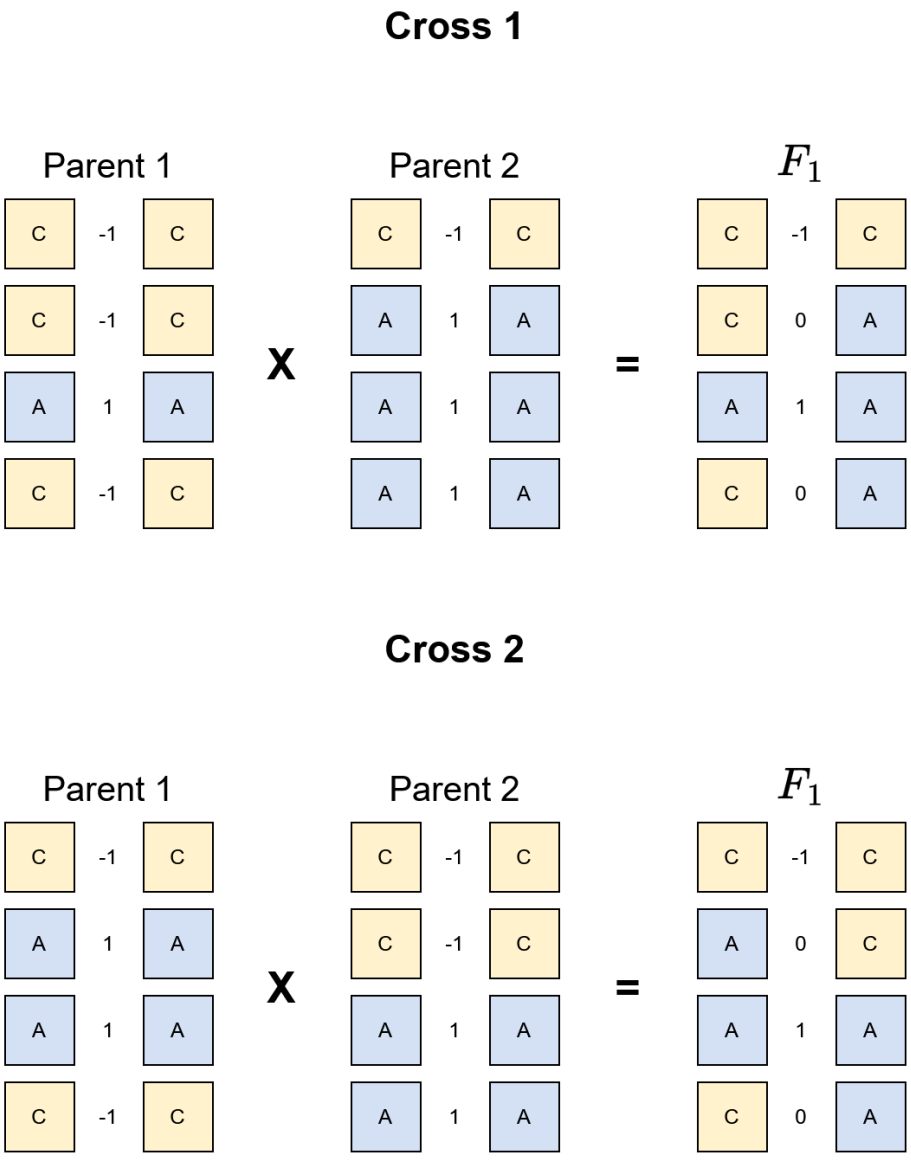

Fig. S1 Illustration showing the encoding problem encountered with hybrids. An example of haplotype block variants containing different alleles and their respective encodings of two genotypes resulting from crosses between different parents is displayed. Although the two genotypes have identical encodings, the block variants differ. This illustrates the issue that different block variants may be treated as the same variant using the current encoding in our model.

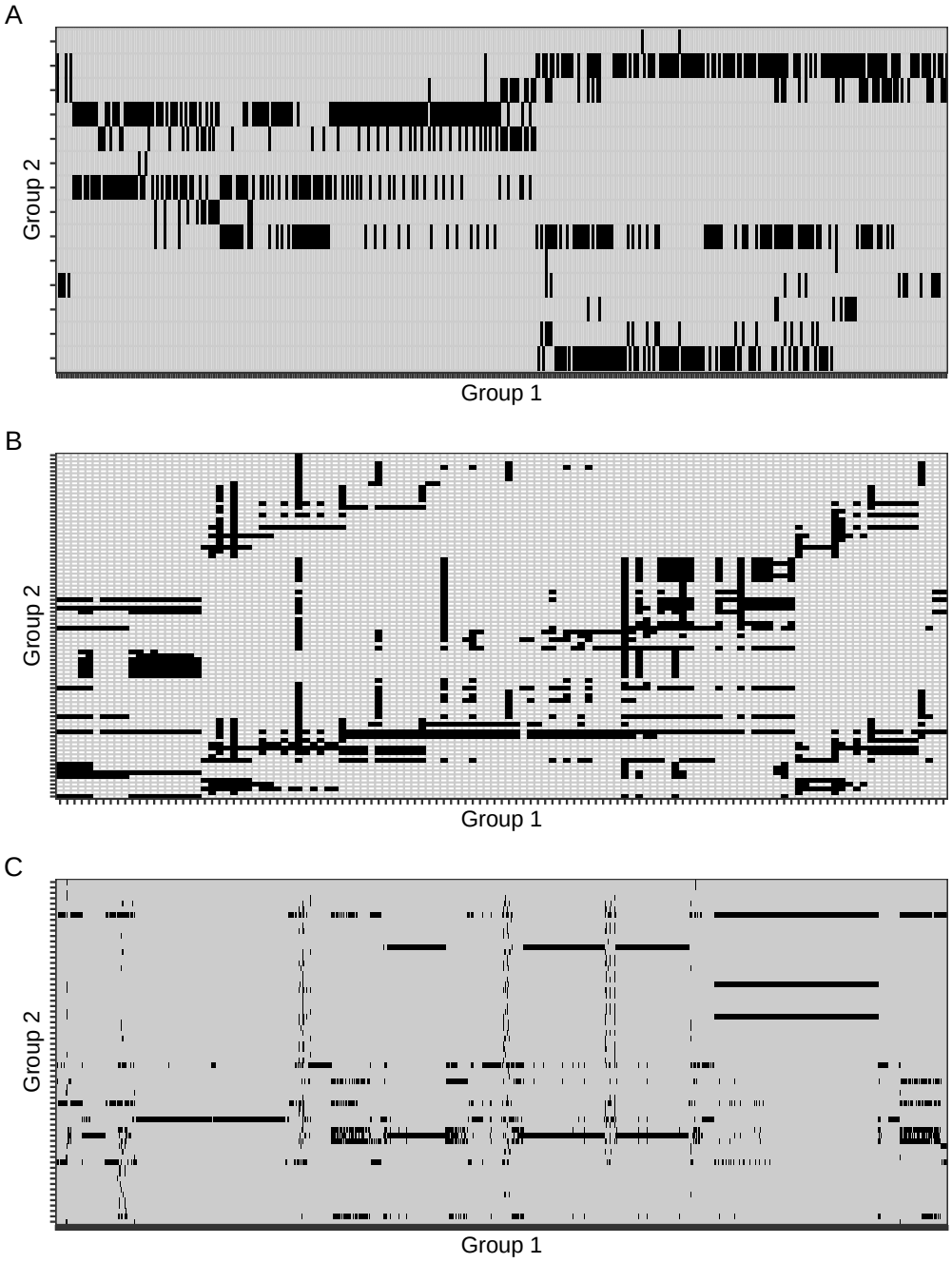

Fig. S2 Crossing matrices of the datasets indicating the sparsity/completeness of the factorial. Black tiles represent realized crosses. Tick marks on both axes represent the parents from the respective group. **A**: Ra1, 14 × 381; **B**: Mz1, 86 × 123; **C**: Mz2, 64 × 2100

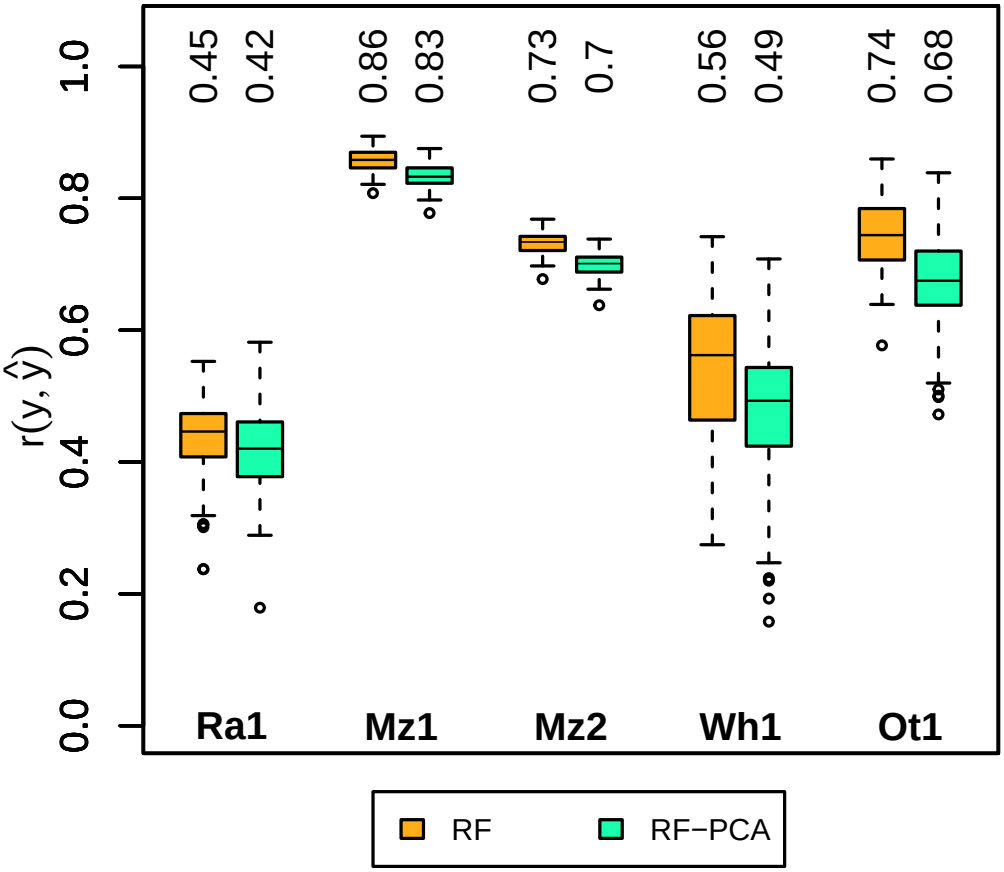

Fig. S3 Prediction accuracy of all cross-validation runs for all datasets, comparing RF based on the full SNPs and RF based on principal components of a PCA. Median prediction accuracy displayed above boxplots. RF on SNPs corresponds to the results presented in the main body of the study. When principle components were used as inputs, all principle components were included.

Chapter 5

General discussion

Genomic prediction in plant breeding is influenced by many factors. While field-related aspects such as experiment size, design, and cross planning affect data quality, this thesis focuses on the computational side of the process, following the completion of field trials. Three main components that determine the success of genomic prediction from a computational perspective are: (1) the selection and preparation of input features for the model; (2) the choice of the predictive model connecting inputs to outputs; (3) the combination of traits and crops that represent the target output.

These aspects are heavily interconnected and interdependent. Different models may be more suitable for predicting certain traits, and certain input features may be more effective for this purpose or for increasing the computational speed of prediction models. In the previous chapters, various models were compared using different input features to predict yield and resistance traits in different species. This chapter provides a general discussion of each of these three aspects in the context of the results presented in this thesis and relevant literature. An outlook beyond the current state of ML is provided that addresses different problems of ML in plant breeding in the *status quo*.

The influence of input features on genomic prediction

The selection and preprocessing of input features is a critical part of ML for genomic prediction and makes up a substantial part of the overall time spent on a ML

project (Kuhn and Johnson 2019; Verdonck et al. 2024). As diverse data types become more widely available, breeders must decide which information to include and how to encode it for prediction. This can be challenging, as datasets may contain tens of thousands of markers and environmental data recorded at up to the minute level, image data, transcriptome data, management data, and more. Adding more variables p to a dataset with a constant number of observations n can lead to a 'small n, high p' scenario, resulting in long training times and potential overfitting, sometimes described as the 'curse of dimensionality' (Hastie et al. 2001). ML algorithms are particularly prone to this, because they can even learn patterns in the predictors that are caused by random experimental errors. Such overfitted models fail to make accurate predictions when predicting new data without the same random pattern (Hastie et al. 2001). Therefore, engineering new, more informative features or reducing feature dimensionality while retaining most information can be beneficial in terms of both speed and generalisation (Kuhn and Johnson 2019; Verdonck et al. 2024).

A minimalist pedigree approach as an alternative to marker data

Since the introduction of ML into plant breeding was fairly recent, it has been seen as an extension to the latest genomic methods. However, the initial study presented in this thesis first considered methods that have long been used in plant breeding practice, such as models based on pedigree information, before exploring more complex feature engineering approaches. These simpler models formed the basis of plant breeding before molecular markers became widely available, and the aim of the study was to evaluate their utility as minimalist baselines for ML. The idea was to investigate whether ML is able to model genetic relatedness based on nominal parentage records alone, i.e. the names of the parental lines as variables, and to accurately predict yield on that basis.

The results of Heilmann et al. (2023) show that ML models trained solely on nominal parentage features often performed well in predicting hybrid yield. In several cases, prediction accuracies were comparable to those achieved using GBLUP with SNP marker data. Only a few other studies have used parentage data in this form (Khaki and Wang 2019; Khaki et al. 2020; Sarijaloo et al. 2021). These studies rely on a large dataset from the 'Syngenta Crop Challenge' (Syngenta 2021), mainly

focusing on predicting yield for genotypes on an environment level. However, as genotype data was not provided with this dataset, none of the studies include a comparison of parentage-based performance with actual genomic models.

The implications of this finding are particularly relevant for breeding programs that may not yet use genotyping or have only been doing so for a short period of time. While many small to medium-sized breeding companies have extensive historical datasets spanning decades of phenotypic evaluations, they often lack genotypic information for older varieties. Consequently, when companies switch to genomic selection, these datasets tend to be underutilised. However, our results suggest that this data could still be useful if it were included in the training of ML models, thereby extracting value from unused data. The large amount of historical data helps to provide a suitable training set for ML algorithms. As this method takes a minimalist approach, it lowers the technical and financial barriers to implementing ML-based prediction. This could be adopted in settings with limited resources, or to bridge the first years after introducing genotyping while the amount of genomic data is insufficient for accurate predictions.

Haplotype blocks

Haplotype blocks are sometimes used as input features for genomic prediction as an alternative to individual SNPs. The idea of this approach is that adjacent markers in strong LD tend to be inherited together. In theory, representing these regions as haplotypes rather than independent SNPs can capture local epistatic effects and may better reflect the underlying genetic architecture of complex traits (Jiang et al. 2018). While there are also non-LD-based approaches to building haplotype blocks, some studies suggest that LD-based blocks are generally better suited for genomic prediction than those defined by physical distance or fixed window sizes (Difabachew et al. 2023; Weber et al. 2023).

Several studies have reported that haplotype-based models can improve prediction accuracy compared to single-marker approaches, particularly when causal variants are in LD with specific haplotype blocks (Gabriel et al. 2002; Cuyabano et al. 2014; Jiang et al. 2018). In the study presented in Chapter 4 (Heilmann et al. 2024), ML models using haplotype blocks performed similarly to SNP-based models in terms of prediction accuracy, and in some cases, slightly better.

One drawback often observed with haplotypes is the increase in dimensionality of the input data (Difabachew et al. 2023). This phenomenon was also observed in Heilmann et al. (2024). Each block can represent multiple allele combinations, requiring one-hot encoding of several variants per block. Consequently, model training time increased, especially for computationally intensive algorithms.

Since haplotype blocks as inputs produced similar or sometimes better results than SNPs, they can be a suitable choice as inputs if computation time is of lesser importance. Depending on the dataset and the number of variants of haplotype blocks, ML can be hampered by haplotype blocks without any guaranteed benefit.

Input features with reduced feature dimensionality

Given that haplotype blocks increased the number of features and thus computation time for ML models, alternative ways to reduce genomic feature dimensionality were explored. Two additional approaches were investigated in Heilmann et al. (2024): variable selection and autoencoder-based feature compression. The goal was to make ML training faster and less prone to overfitting by condensing the genotype data while maintaining prediction accuracy. Based on the results of this investigation and findings regarding haplotype blocks, a new deep learning architecture was developed that incorporates the idea of LD-based haplotype blocks into autoencoders. This combined the autoencoder's guaranteed dimensionality reduction with the performance of haplotype blocks, creating a new set of input features that are equal to or better than the original in terms of prediction accuracy while reducing computational time.

Variable selection

One straightforward approach to dimensionality reduction is to select the most important SNP markers and discard the rest. Feature selection is a common ML technique used to remove noisy or redundant variables. However, applying it to genomic prediction might not be useful since the infinitesimal model assumes many loci with small effects. Therefore, omitting most markers theoretically violates this assumption.

Several studies have applied variable selection using ML in plant breeding (Azodi et al. 2019; Gabur et al. 2022; Heinrich et al. 2023). Heilmann et al. (2024) used the

approach described in Heinrich et al. (2023), in which SNPs are ranked by importance based on a genome-wide association study. Starting with the top-ranked SNPs, more markers are iteratively added to a random forest model. Cross-validation is used on the training set to identify the optimal number of SNPs to retain based on the lowest mean squared error.

Results of the genomic prediction were mostly equal to using a random forest with the full set of SNPs, with only one case showing a notable decline in prediction accuracy. Median training time was reduced by approximately 20%, suggesting that feature selection can help reduce computational demands. However, caution is advised, as this did not work in all cases. It could be assumed that the more markers influence a trait, the longer variable selection takes, since more markers have to be tested to find the optimum number to include. If more effort is required for variable selection, the advantage of reduced computation time for the actual model is nullified.

Autoencoders

Instead of elimination, another approach to dimensionality reduction is to compress features into a lower-dimensional representation. In order to achieve this, a special neural network called autoencoder was used as a non-linear alternative to principal component analysis to encode genotype data (Hinton and Salakhutdinov 2006). An autoencoder is a neural network consisting of multiple layers, where the intermediate layers are typically of smaller dimension than the input and the output. The model is then optimized in such a way that the input data is passed through the lower-dimensional layers and then reconstructed again from this compressed state. Taking the output of the middle hidden layer as a condensed latent representation creates a set of new variables that are abstract representations of the original markers. This compression could address the curse of dimensionality by reducing the number of variables passed into the prediction model, speeding up computation. Given the reconstruction error is sufficiently low, the compressed representation should theoretically contain all the information also contained in the raw data and would therefore lead to a similar prediction accuracy.

While the use of autoencoders in plant breeding is relatively rare, they have been more common in phenomics-related studies (Jurado-Ruiz et al. 2023; Powadi et al. 2024; Tross et al. 2024) to compress or process hyperspectral data and images,

only one study has used autoencoders for dimensionality reduction and subsequent genomic prediction (Islam et al. 2023). In this study, autoencoders were successful at reducing the data dimensionality while keeping the prediction accuracy stable However, results from Heilmann et al. (2024) in Chapter 4 were different. There, features generated by autoencoders helped drastically speeding up computation time of ML algorithms, but at the cost of prediction accuracy. In most cases, algorithms performed a lot worse when using autoencoder features as inputs.

Autoencoders are neural networks that reconstruct original input data from a compressed state. Apart from this characteristic, there is no pre-defined architecture for building an autoencoder. The number and type of layers that the autoencoder consists of are chosen by the user. There are also different approaches on how to use autoencoders for dimensionality reduction. While Islam et al. (2023) trained a lot of small autoencoders, each on only a small window of adjacent markers, Heilmann et al. (2024) trained one autoencoder on the whole data, using fully connected layers between all inputs and units of the next layer. While one autoencoder instead of many small autoencoders is better suited to capture global correlations, the predictions based on these features showed low prediction accuracy. Finding more suitable architectures for autoencoders could help creating a small set of input features that result in improved computation speed compared to models using the original data while maintaining the same level of prediction accuracy.

Haplotype-based autoencoders

To address the limitations of the autoencoder described in Heilmann et al. (2024), Heilmann et al. (2025) proposed a combination of the above ideas: LD-based haplotype blocks as the basis for autoencoders. For five datasets from different crops, marker data was partitioned into LD-based blocks. Then, the units in the input layer of an autoencoder were only locally connected to the next layer, i.e. only markers that belonged to the same block shared a connection in the next layer. All units belonging to the same block were eventually connected to a single unit which was referred to as 'block layer'. This addressed the problem of dimensionality increase using one-hot encoded haplotype blocks, as extracting the outputs of the block layer guarantees one variable per block and thus a reduction in dimensionality. Prediction accuracy remained stable after compression, and remained stable even after a second compression was applied to the features that had already been reduced. Relationship matrices derived from the full SNP data were similar to those

from the compressed data, indicating that information contained within the genome was mostly preserved.

This approach was inspired by Islam et al. (2023), as the way autoencoders are built in their study resemble a fixed window size approach in haplotype blocks. As mentioned earlier, LD-based haplotypes blocks are usually more stable and perform better when used as input features in genomic prediction compared to blocks built using fixed distance or window size methods. The method presented here could be seen as an improvement over Islam et al. (2023) as one autoencoder could be used to train on all data, whereas Islam et al. (2023) used many small autoencoders. Additionally, the new method did not require an arbitrary choice or tuning of window size, as it relied on patterns found within the data. In human genomics, Taş et al. (2024) developed LD-aware autoencoders. They achieved very low reconstruction errors of genetic data, also by using many small autoencoders, but per haplotype block. While this approach is very similar to the one presented here, they did not extract any features to use for genomic prediction, so the usefulness of this approach for genomic prediction is not clear. As their study used more than six million genetic markers, a single autoencoder would likely not have been functional. Generally, a single large autoencoder requires more memory and is slower to train, while smaller autoencoders can be faster if parallelized efficiently but may miss global patterns across the full marker set (Mirsky et al. 2018; Ausmees and Nettelblad 2022; Nawaz et al. 2024). In scenarios where data dimensionality is too high for computer memory or processing too slow, an ensemble of smaller autoencoders is necessary. In this case, the approach presented here could likely be modified into smaller autoencoders, resembling the approach of Taş et al. (2024). As a compromise, data could be partitioned chromosome-wise to still be able to capture genetic patterns while requiring smaller computer memory.

Building a haplotype-based autoencoder using genomic information illustrates how a domain-specific architecture design can improve the applicability of ML. This method did not exist in the literature previously and represents a new contribution on how to integrate genomic information into ML models for plant breeding.

Overall, the influence of the input variables on the training process and the prediction accuracy was quite strong. Raw marker data can be seen as the gold standard, as there were few cases in which other input variables resulted in better prediction accuracy. ML is especially affected by the number of variables used

during model training. Time required to train a model may increase drastically with an increase in input dimension. Typically, many models have to be trained for hyperparameter tuning, cross-validation and the formation of model ensembles (Cawley and Talbot 2010; Probst et al. 2019). This can make studies and projects involving ML time-consuming and tiresome, especially during an initial 'exploration' phase. Speeding up computation time while maintaining genomic information and prediction accuracy allows for testing broader hyperparameter spaces, creating more diverse models in a shorter time or larger ensembles of models in the same amount of time. Additionally, GBLUP and RR-BLUP have a rarely addressed problem. Both typically employ a genomic relationship matrix in their practical software implementations. This relationship matrix always has the dimension $n \times n$, where n is the number of genotypes. If n is large, the dimensions of the relationship matrix increases in both row and column direction. During model fitting of dataset Mz2 in Heilmann et al. (2025), consisting of about 2000 parental lines, common packages used for GBLUP like 'sommer' (Covarrubias-Pazaran 2016) were not computationally able to process the relationship matrix. Hence, it was necessary to switch to ASRemL (The VSNi Team 2023), a licensed software. Nevertheless, training an ensemble of random forests based on the twice-compressed dataset was faster than the GBLUP. Even without deeper insight, it is reasonable to assume that breeding companies may have similar or larger datasets, or will have them at some point as they accumulate more data each year. As GBLUP/RR-BLUP apparently also face limitations based on dimensionality, ML could be an alternative for 'high n, low p' scenarios.

The influence of algorithm choice and hyperparameters on genomic prediction

Results of Heilmann et al. (2023) and Heilmann et al. (2024) showed notable differences between some algorithms and a high similarity between others. For instance, support vector machines tended to perform worse than other models, and sometimes the decrease in prediction accuracy was relatively strong. Heslot et al. (2012) reported similar observations, in which support vector machines had the lowest performance among several methods tested. In the study of Azodi et al. (2019), support vector machines also tended to perform worse than other algorithms, showing high

inaccuracy for certain traits. Support vector machines generally seemed more unstable than tree-based algorithms. They sometimes performed extremely poorly, with prediction accuracies close to 0. Even then, there were some traits in Heilmann et al. (2024) for which support vector machines outperformed other ML algorithms. This is a typical example of the 'No Free Lunch' theorem (Wolpert and Macready 1997), which states that there is no single algorithm that performs best at every task. Some algorithms excel at some tasks but perform poorly at others. Since plant breeding is a very diverse field, with a strong differences between different predictors and different traits, some algorithms perform better in certain cases. Tree-based algorithms produced more consistent prediction accuracies and were usually comparable to GBLUP and in some cases even better (Heilmann et al. 2023, 2024, 2025), an observation that can also often be found in literature (Heslot et al. 2012; Blondel et al. 2015; Azodi et al. 2019).

Heilmann et al. (2023) tested random grid search-based hyperparameter tuning. Across 100 cross-validation runs, some hyperparameter combinations appeared much more often than others as the optimal choice based on the grid search. These optimal hyperparameter combinations differed between datasets. This indicates that each algorithm has a certain 'set' of hyperparameter combinations or 'area' in the hyperparameter space where it performs best for each crop. The somewhat consistent results also imply that the influence of the random partitioning into training and test sets during cross-validation did not have a strong influence on the result of the grid search. Often, the effect of hyperparameters is not discussed in detail, but hyperparameter tuning itself is typically seen as a mandatory part of successful ML models (Probst et al. 2019; Pérez-Enciso and Zingaretti 2019). We did not find that the Bayesian optimisation approach used in Heilmann et al. (2024) yielded better results than the less sophisticated random grid search used in Heilmann et al. (2023).

A practical alternative to extensive hyperparameter tuning is to build ensembles of models, i.e. training a new model on the basis of a set of models with different hyperparameters or even different algorithms altogether. These stacked ensembles are typically better than the individual models they consist of (Wolpert 1992), have been successfully implemented in plant breeding studies (Liang et al. 2021; Heilmann et al. 2023; Kick and Washburn 2023; Tomura et al. 2025), and tend to work better the more diverse the models included are (Page 2018; Tomura et al. 2025).

The influence of crop and trait on genomic prediction

The effectiveness of genomic prediction via ML is also influenced by the characteristics of the trait being predicted, the crop, and the specific population under investigation. In the study on hybrids (Heilmann et al. 2023), the ensemble model of the minimalist ML approach worked better than the standard methods when the ratio of SCA to GCA was high and the yield itself was highly correlated to SCA, whereas no differences could be observed when the yield was determined by GCA. The GCA reflects additive genetic variance and can be effectively modeled using linear models, whereas SCA involves more complex interactions such as dominance and epistasis, making it harder to capture. (Sprague and Tatum 1942). This could indicate that ML in general or ensembles of models specifically can be an effective tool for such scenarios. Even among the same crop, strong differences were observed, e.g. among the three rapeseed datasets used in this study. This suggests that the composition of the population may be equally influential as the crop itself, which should be investigated further.

Differences were also observed between five fungal diseases in wheat (Heilmann et al. 2024), in which the overall level of prediction accuracy was determined by the trait. For some traits, RR-BLUP was the best, whereas for others, tree-based methods performed better. Tree-based methods were also among the best algorithms in some other studies (Rutkoski et al. 2012; Tomar et al. 2021). Differences in prediction accuracy between methods within a trait were much smaller than the differences of the average prediction accuracy between traits. It is not possible to answer why the differences are observable, but two main influences could be: (a) traits were measured on an ordinal scale from 1–9, but the full range was not used as most measurements fell towards the lower end of the scale for some traits, and (b) the genetic architecture between resistances is different. It could be possible that there were a few loci with larger effects in addition to many small effects for *Puccina triticina*, which ML algorithms should naturally be able to deal with. However, when resistance was polygenic and the phenotype data noisy, the ML models struggled. However, this is speculation as more field trials and a deeper investigation into the traits would be necessary. The genetic architecture and other influences on resistance to diseases are complex interactions and it is not trivial to characterize

those underlying mechanisms (Derbyshire et al. 2024). On the other hand, the high imbalance in the distribution of some traits due to the low infection rate may have made it challenging for algorithms to identify genetic patterns associated with actual resistance.

The findings described in the two studies also lead up to a similar conclusion as before: There is no one unified approach to successful predictions. The success of ML is determined to some degree by traits and crops, but the composition of the population or the sparsity level of the factorial crosses used as the basis for prediction could be even more important, as observed with the three rapeseed datasets.

For every key aspect of the success of genomic prediction, no single prediction method could be recommended as superior under all circumstances, consistent with the 'No Free Lunch' theorem. This is also a widely observed phenomenon in the literature and one of the ongoing problems ML faces in plant breeding (Azodi et al. 2019; González-Camacho et al. 2018; Abdollahi-Arpanahi et al. 2020; Kick and Washburn 2023; Jones et al. 2023). Model training can be made faster, but no method guarantees a high prediction accuracy. Conversely, most of the methods tested in the studies presented here, as well as in other works (Azodi et al. 2019; González-Camacho et al. 2018; Abdollahi-Arpanahi et al. 2020; Kick and Washburn 2023; Jones et al. 2023), seem to perform reasonably well for a wide range of datasets, with the exception of support vector machines.

An outlook beyond the current state of ML in plant breeding

At present, the performance of ML methods in genomic selection is often on par with standard methods like GBLUP but not significantly better in most cases (Azodi et al. 2019; Abdollahi-Arpanahi et al. 2020; Montesinos-López et al. 2021). This has been somewhat disappointing relative to early expectations. The results shown in Chapters 2, 3, and 4 support this: While there were scenarios and traits in which an optimized ML model or an ensemble outperformed GBLUP or RR-BLUP, there were also many cases in which gains in prediction accuracy were minor or absent. Moreover, due to hyperparameter tuning and computational requirements, ML models often required more time for training and were harder to interpret than linear models.

However, rather than viewing this as a dead-end, it could be seen as an incentive to change how ML is applied in breeding. There is no definite answer on how to do this. Suggestions include increased cooperation and data pooling (Hayes et al. 2023), more thorough investigations and scientific practice (Messina et al. 2025), and the integration of crop growth models into ensembles (Cooper et al. 2025). Especially the second notion of scientific practice seems important as many recent studies lack quality and contain methodological errors (Leukel et al. 2025). For example, some studies present novel deep learning architectures applied to plant breeding problems but do not conduct more than one cross-validation (Wang et al. 2023; Xie et al. 2024a; Li et al. 2024b,a; Montesinos-López et al. 2025; Xie et al. 2024b). While those deep learning approaches may be methodologically sound, their performance in plant breeding cannot be evaluated reliably from a single cross-validation run, even if it is a 10-fold cross-validation. Results may vary depending on the partitioning of the data into training and test sets, particularly in smaller or unbalanced datasets (Cawley and Talbot 2010). Studies with only one cross-validation run therefore make it harder to compare models and synthesise knowledge from the available pool of literature.

In general, the greatest advantage of ML is its flexibility. Various data types, such as genetic markers, pedigrees, environmental variables, sensor data and images can be incorporated into a single predictive framework. In the case of deep learning, the user also designs the internal architecture of the neural network. Classical linear models do not offer the same flexibility. The haplotype-based autoencoder from Heilmann et al. (2025) is an example for this: It is a ML architecture based on linkage information in the data, which allows the generation of new features that (1) improve the computation speed when used in other models and (2) provide estimates for haplotype block effects. While much research is still conducted on haplotype blocks and their effects (Villiers et al. 2024; Tong et al. 2024), the standard method is still to sum up the individual marker effects as initially proposed in Voss-Fels et al. (2019). This seems counterintuitive if the aim is to account for non-linear epistatic interactions. The new model proposed in Heilmann et al. (2025) effectively addresses this issue and also demonstrates that ML can be used for purposes other than prediction. Although ML algorithms are often described as a black box procedures due to their lack of interpretability, this does not necessarily have to be the case. Interpretable ML is an ongoing topic of research and direct or indirect effect estimation is possible (Murdoch et al. 2019; Azodi et al. 2020; Talukder et al. 2020).

Conclusions

A typical statement found in the introduction and/or conclusion of a study on ML might read something like this: 'ML algorithms do not consistently outperform methods such as GBLUP'. Despite its frequent use, this statement offers limited practical insight. Using the same logic, one could also argue that GBLUP does not consistently outperform ML algorithms. Both statements are true, as one implies the other, but neither suggests a recommended course of action. However, while the second version of this statement is rarely said or written, the first appears frequently and is often interpreted pessimistically: It is not worth trying ML because it is not better than GBLUP. This is a misconception, as the only real conclusion one can derive from the initial statement is that arbitrarily choosing a model without prior thought or testing may result in a suboptimal model choice with regard to prediction accuracy.

Seasoned researchers and plant breeders have likely witnessed several anticipated revolutions in their field that then turned out to be exaggerated, only for the next trend to come around a few years later (Bernardo 2016). This is surprising, since the story of plant breeding is a success story that is based on small, incremental gains that accumulate year by year, and their true magnitude only becomes apparent in hindsight. For example, the yield of wheat has consistently been improved over the last six decades, but only by an average of around 1% per year (Voss-Fels et al. 2019; Tadesse et al. 2019). Initially, 1% may not seem much, but the accumulated gain over the years is substantial. ML may take a course similar to many trends in plant breeding before, as it has been exaggerated by some and seen as a disappointment by others so far. Many studies are published every year trying to improve the existing models and working on proper procedures for their implementation. Custom ML solutions for breeding, as well as broader use of ML to exploit new data types (drones, sensors, genomics, transcriptomics) into a single framework, will likely improve with a better understanding of the underlying mechanisms that influence the success of ML. The reason for this optimism is the broad success of ML in many other scientific areas. The unique flexibility of deep learning in particular offers a distinct advantage over linear models, hence it seems inevitable that deep learning methods will be adopted in at least some area of plant breeding. Ongoing research will increase model interpretability in future in cases where interpretation is necessary. As initially stated, no model always works best, but it is also not necessary to rely

on a single "best" model. Task-specific models could be designed that excel in very specific situations. Since these scenarios recur in breeding programs with every generation, specific models will also be helpful.

Developing such solutions will likely require time and will consist of the combined work of many studies, each contributing and advancing the methodology. These advances, combined with validation and incorporation of breeder knowledge, will likely lead to better predictions, effect estimations and, ultimately, better crops.

Chapter 6

Summary

Genomic prediction, originally proposed as a solution to the limitations of marker-assisted selection for complex traits, has become the standard for estimating breeding values in both inbred and hybrid crops. While linear models such as GBLUP and RR-BLUP remain effective in many cases, especially when assuming an additive genetic architecture, recent years have seen a growing interest in applying machine learning (ML) methods to overcome some of their constraints, including their limited capacity to model non-additive effects and nonlinear interactions. This thesis explored the influence of three key aspects on the success of genomic prediction: The choice of input features, the statistical model used, and the target trait or crop.

In terms of input features, marker data was compared to minimalist parentage-based models, haplotype blocks, and features generated using autoencoders. It was shown that even simple ML models using parentage-based information can rival marker-based GBLUP under certain conditions, which holds potential for small breeding programs with large amounts of historical, but ungenotyped, records. At the same time, dimensionality reduction techniques, especially a novel haplotype-based autoencoder that was developed during this thesis, were introduced as a means to compress genomic data while preserving prediction accuracy and succesfully accelerated model training.

Concerning the model aspect, a variety of ML algorithms were benchmarked using different approaches for hyperparameter tuning. Although no single model

outperformed others across all traits and crops, ensemble approaches typically performed better than the individual models they were based on. Support vector machines seemed to be relatively unstable when compared to other ML based algorithms, such as tree-based models.

Finally, results showed that the accuracy of the genomic predictions was strongly dependent on differences between traits, crops with different breeding schemes, and different populations. For hybrids, ML performed well when SCA was more important for determining the hybrid yield than GCA. Large differences were observed for different fungal diseases in wheat, while differences among methods for the same disease were relatively similar.

While ML has not yet provided a significant improvement over traditional methods in many scenarios, its flexibility and potential for multi-modal data integration remain promising. The development of plant breeding-specific model architectures, such as haplotype-based autoencoders, may represent a more promising path than the general application of standard ML models.

Kapitel 7

Zusammenfassung

Die genomische Vorhersage, die ursprünglich als Lösung für die Einschränkungen der markergestützten Selektion bei polygenen Merkmalen entwickelt wurde, ist zum Standard für die Schätzung von Zuchtwerten sowohl bei Inzucht- als auch bei Hybridsorten geworden.

Während lineare Modelle wie GBLUP und RR-BLUP in vielen Fällen nach wie vor effektiv sind, insbesondere wenn sie von einer additiven genetischen Architektur ausgehen, hat in den letzten Jahren das Interesse an der Anwendung von Machine Learning (ML) Methoden zugenommen. Die Hoffnung hierbei ist, dass so einige Beschränkungen der linearen Modelle überwunden werden können. Dazu gehört unter anderem ihre begrenzte Fähigkeit, nicht-additive Effekte und nicht-lineare Interaktionen zu modellieren. In dieser Arbeit wurde der Einfluss dreier Schlüsselaspekte auf den Erfolg genomischer Vorhersagen untersucht: die Wahl der Prädiktoren, das verwendete statistische Modell und das Zielmerkmal der Kulturart.

In Bezug auf Prädiktoren wurden Markerdaten mit nominalskalierten Elterninformationen, Haplotypblöcken und autencodergenerierten latenten Variablen verglichen. Es wurde gezeigt, dass selbst einfache ML-Modelle, die lediglich die Namen der Eltern verwenden, unter bestimmten Bedingungen ähnliche Vorhersagegenauigkeiten wie markerbasierte GBLUP-Modelle erzielen. Dies kann bei kleinen Zuchtprogrammen mit großen Mengen historischer Stammbauminformationen, für die keine Markerdaten vorliegen, genutzt werden. Gleichzeitig wurden Techniken zur Dimensionsreduktion als Mittel zur Komprimierung genomischer Daten bei gleichzeitiger Wahrung der Vorhersagegenauigkeit eingeführt, wodurch das Modelltraining beschleunigt wurde. Dazu gehört insbesondere ein neuartiger haplotypbasierter Autoencoder, der im Rahmen dieser Arbeit entwickelt wurde.

Auf der Modellebene wurden eine Reihe von ML-Algorithmen mit verschiedenen Ansätzen für die Suche nach geeigneten Hyperparametern verglichen. Obwohl kein Modell über alle Merkmale und Kulturen hinweg das beste war, erzielten Ensemble-Ansätze in der Regel bessere Vorhersagegenauigkeiten als die einzelnen Modelle, auf denen sie basierten. Support Vector Machines schienen im Vergleich zu anderen ML-basierten Algorithmen, wie tree-based Modellen, relativ instabil zu sein.

Die Vorhersagegenauigkeit der genomischen Vorhersagemodelle wurde stark vom jeweils betrachteten Merkmal, der Kulturart und Besonderheiten in der genetischen Zusammensetzung des jeweiligen Datensatzes beeinflusst. Bei Hybriden schnitt ML gut ab, wenn die SCA für die Bestimmung des Hybridertrags wichtiger war als die GCA. Große Unterschiede wurden für verschiedene Pilzerkrankungen bei Weizen beobachtet, während die Unterschiede zwischen den Methoden für dieselbe Krankheit relativ gering waren.

Obwohl ML in vielen Szenarien noch keine signifikante Verbesserung gegenüber herkömmlichen Methoden darstellt, sind seine Flexibilität und sein Potenzial für die multimodale Datenintegration weiterhin vielversprechend. Die Entwicklung von pflanzenzüchtungsspezifischen Modellarchitekturen, wie z. B. Haplotypbasierte Autoencoder, könnte ein vielversprechenderer Weg sein als die allgemeine Anwendung von Standard-ML-Modellen.

Chapter 8

Literature

R. Abdollahi-Arpanahi, D. Gianola, and F. Peñagaricano. Deep learning versus parametric and ensemble methods for genomic prediction of complex phenotypes. *Genetics Selection Evolution*, 52(1):12, Feb 2020. doi: 10.1186/s12711-020-00531-z.

K. Ausmees and C. Nettelblad. A deep learning framework for characterization of genotype data. *G3: Genes, Genomes, Genetics*, 12(3):jkac020, 01 2022. doi: 10.1093/g3journal/jkac020.

C. B. Azodi, E. Bolger, A. McCarren, M. Roantree, G. de los Campos, and S.-H. Shiu. Benchmarking parametric and machine learning models for genomic prediction of complex traits. *G3: Genes, Genomes, Genetics*, 9(11):3691–3702, 11 2019. doi: 10.1534/g3.119.400498.

C. B. Azodi, J. Tang, and S.-H. Shiu. Opening the black box: Interpretable machine learning for geneticists. *Trends in Genetics*, 36(6):442–455, 2020. doi: 10.1016/j.tig.2020.03.005.

W. D. Beavis. QTL analyses: power, precision, and accuracy. In A. H. Paterson, editor, *Molecular Dissection of Complex Traits*, pages 145–162. CRC Press, Boca Raton, 1998.

H. Becker. *Pflanzenzüchtung*. UTB, Stuttgart, 04 2011. ISBN 9783838535586. doi: 10.36198/9783838549507.

R. Bernardo. Prediction of maize single-cross performance using RFLPs and information from related hybrids. *Crop Science*, 34(1): cropsci1994.0011183X003400010003x, 1994. doi: 10.2135/cropsci1994.0011183X003400010003x.

R. Bernardo. Molecular markers and selection for complex traits in plants: Learning from the last 20 years. *Crop Science*, 48(5):1649–1664, 2008. doi: 10.2135/cropsci2008.03.0131.

R. Bernardo. *Breeding for Quantitative Traits in Plants*. Stemma Press, Woodbury, Minnesota, 2010.

R. Bernardo. Bandwagons i, too, have known. *Theoretical and Applied Genetics*, 129(12):2323–2332, Dec 2016. doi: 10.1007/s00122-016-2772-5.

T. Bernhard, W. Friedt, K. P. Voss-Fels, M. Frisch, R. J. Snowdon, and B. Wittkop. Heterosis for biomass and grain yield facilitates breeding of productive dual-purpose winter barley hybrids. *Crop Science*, 57(5):2405–2418, 2017. doi: 10.2135/cropsci2016.10.0872.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

M. Blondel, A. Onogi, H. Iwata, and N. Ueda. A ranking approach to genomic selection. *PLOS ONE*, 10(6):1–23, 06 2015. doi: 10.1371/journal.pone.0128570.

L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

L. Brigato and L. Iocchi. A close look at deep learning with small data. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2490–2497, 2021. doi: 10.1109/ICPR48806.2021.9412492.

G. C. Cawley and N. L. C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(70):2079–2107, 2010.

B. C. Collard and D. J. Mackill. Marker-assisted selection: an approach for precision plant breeding in the twenty-first century. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 363(1491):557–572, 2008. doi: 10.1098/rstb.2007.2170.

B. C. Y. Collard, M. Z. Z. Jahufer, J. B. Brouwer, and E. C. K. Pang. An introduction to markers, quantitative trait loci (QTL) mapping and marker-assisted selection for crop improvement: The basic concepts. *Euphytica*, 142(1):169–196, Jan 2005. doi: 10.1007/s10681-005-1681-5.

M. Cooper, S. Tomura, M. J. Wilkinson, O. Powell, and C. D. Messina. Breeding perspectives on tackling trait genome-to-phenome (G2P) dimensionality using ensemble-based genomic prediction. *Theoretical and Applied Genetics*, 138(7): 172, Jul 2025. doi: 10.1007/s00122-025-04960-6.

C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, Sep 1995. doi: 10.1007/BF00994018.

G. Covarrubias-Pazaran. Genome assisted prediction of quantitative traits using the R package sommer. *PLoS ONE*, 11:1–15, 2016.

B. C. Cuyabano, G. Su, and M. S. Lund. Genomic prediction of genetic merit using LD-based haplotypes in the Nordic Holstein population. *BMC Genomics*, 15(1): 1171, Dec 2014. doi: 10.1186/1471-2164-15-1171.

G. P. Davis and S. K. DeNise. The impact of genetic markers on selection. *Journal of Animal Science*, 76(9):2331–2339, 09 1998. doi: 10.2527/1998.7692331x.

M. C. Derbyshire, T. E. Newman, W. J. W. Thomas, J. Batley, and D. Edwards. The complex relationship between disease resistance and yield in crops. *Plant Biotechnology Journal*, 22(9):2612–2623, 2024. doi: 10.1111/pbi.14373.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding, 2019. URL `https://arxiv.org/abs/1810.04805`.

Y. F. Difabachew, M. Frisch, A. L. Langstroff, A. Stahl, B. Wittkop, R. J. Snowdon, M. Koch, M. Kirchhoff, L. Cselényi, M. Wolf, J. Förster, S. Weber, U. J. Okoye, and C. Zenke-Philippi. Genomic prediction with haplotype blocks in wheat. *Frontiers in Plant Science*, 14, 2023. doi: 10.3389/fpls.2023.1168547.

H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In M. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1996.

A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, Feb 2017. doi: 10.1038/nature21056.

R. E. Evenson and D. Gollin. Assessing the impact of the green revolution, 1960 to 2000. *Science*, 300(5620):758–762, 2003. doi: 10.1126/science.1078710.

FAO. *The future of food and agriculture – Trends and challenges*. Food and Agriculture Organization of the United Nations, Rome, 2017. ISBN 978-92-5-109551-5.

FAO. *The future of food and agriculture – Drivers and triggers for transformation*. Food and Agriculture Organization of the United Nations, Rome, 2022. ISBN 978-92-5-136639-4. doi: 10.4060/cc0959en.

R. A. Fisher. The correlation between relatives on the supposition of mendelian inheritance. *Transactions of the Royal Society of Edinburgh*, 52:399–433, 1918.

J. A. Foley, N. Ramankutty, K. A. Brauman, E. S. Cassidy, J. S. Gerber, M. Johnston, N. D. Mueller, C. O'Connell, D. K. Ray, P. C. West, C. Balzer, E. M. Bennett, S. R. Carpenter, J. Hill, C. Monfreda, S. Polasky, J. Rockström, J. Sheehan, S. Siebert, D. Tilman, and D. P. M. Zaks. Solutions for a cultivated planet. *Nature*, 478(7369):337–342, Oct 2011. doi: 10.1038/nature10452.

B. P. Forster and W. T. B. Thomas. *Doubled Haploids in Genetics and Plant Breeding*, chapter 3, pages 57–88. John Wiley & Sons, Ltd, 2005. ISBN 9780470650301. doi: 10.1002/9780470650301.ch3.

E. Francia, G. Tacconi, C. Crosatti, D. Barabaschi, D. Bulgarelli, E. Dall'Aglio, and G. Valè. Marker assisted selection in crop plants. *Plant Cell, Tissue and Organ Culture*, 82(3):317–342, Sep 2005. doi: 10.1007/s11240-005-2387-z.

J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

M. Frisch and A. E. Melchinger. Marker-assisted backcrossing for simultaneous introgression of two genes. *Crop Science*, 41(6):1716–1725, 2001. doi: 10.2135/cropsci2001.1716.

S. B. Gabriel, S. F. Schaffner, H. Nguyen, J. M. Moore, J. Roy, B. Blumenstiel, J. Higgins, M. DeFelice, A. Lochner, M. Faggart, S. N. Liu-Cordero, C. Rotimi, A. Adeyemo, R. Cooper, R. Ward, E. S. Lander, M. J. Daly, and D. Altshuler. The structure of haplotype blocks in the human genome. *Science*, 296(5576): 2225–2229, 2002.

I. Gabur, D. P. Simioniuc, R. J. Snowdon, and D. Cristea. Machine learning applied to the search for nonlinear features in breeding populations. *Frontiers in Artificial Intelligence*, Volume 5 - 2022, 2022. doi: 10.3389/frai.2022.876578.

T. Garnett, M. C. Appleby, A. Balmford, I. J. Bateman, T. G. Benton, P. Bloomer, B. Burlingame, M. Dawkins, L. Dolan, D. Fraser, M. Herrero, I. Hoffmann, P. Smith, P. K. Thornton, C. Toulmin, S. J. Vermeulen, and H. C. J. Godfray. Sustainable intensification in agriculture: Premises and policies. *Science*, 341(6141):33–34, 2013. doi: 10.1126/science.1234485.

D. Gianola and J. B. C. H. M. van Kaam. Reproducing kernel Hilbert spaces regression methods for genomic assisted prediction of quantitative traits. *Genetics*, 178(4):2289–2303, 2008.

J. M. González-Camacho, L. Ornella, P. Pérez-Rodríguez, D. Gianola, S. Dreisigacker, and J. Crossa. Applications of machine learning methods to genomic selection in breeding wheat for rust resistance. *The Plant Genome*, 11(2):170104, 2018. doi: 10.3835/plantgenome2017.11.0104.

D. Habier, R. L. Fernando, and J. C. M. Dekkers. The impact of genetic relationship information on genome-assisted breeding values. *Genetics*, 177(4):2389–2397, 12 2007. doi: 10.1534/genetics.107.081190.

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

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

B. J. Hayes, C. Chen, O. Powell, E. Dinglasan, K. Villiers, K. E. Kemper, and L. T. Hickey. Advancing artificial intelligence to help feed the world. *Nature Biotechnology*, 41:1188–1189, 2023. doi: 10.1038/s41587-023-01898-2.

E. L. Heffner, A. J. Lorenz, J.-L. Jannink, and M. E. Sorrells. Plant breeding with genomic selection: Gain per unit time and cost. *Crop Science*, 50(5):1681–1690, 2010. doi: 10.2135/cropsci2009.11.0662.

P. G. Heilmann, M. Frisch, A. Abbadi, T. Kox, and E. Herzog. Stacked ensembles on basis of parentage information can predict hybrid performance with an accuracy comparable to marker-based GBLUP. *Frontiers in Plant Science*, 14, 2023. doi: 10.3389/fpls.2023.1178902.

P. G. Heilmann, Y. F. Difabachew, M. Frisch, A. L. Moritz, A. Stahl, B. Wittkop, R. J. Snowdon, M. Koch, M. Kirchhoff, L. Cselényi, M. Wolf, J. Förster, and

C. Zenke-Philippi. Machine learning for prediction of resistance scores in wheat (*Triticum aestivum* L.). *Plant Breeding*, 144(2):192–205, 2024. doi: 10.1111/pbr.13235.

P. G. Heilmann, E. Grosch, M. Frisch, M. Herrmann, S. Beuch, V. Kurra, M. Mascher, R. Avni, K. Oldach, I. Röhrs, A. Hanemann, R. R. Mehta, C. Reinbrecht, A. Serfling, A. Stahl, M. Stucke, A. Abbadi, T. Kox, and C. Zenke-Philippi. Haplotype-based autoencoders can reduce the dataset dimension and estimate haplotype block effects in different crop species. *BMC Bioinformatics*, 26(1):289, Dec 2025. doi: 10.1186/s12859-025-06323-w.

F. Heinrich, T. M. Lange, M. Kircher, F. Ramzan, A. O. Schmitt, and M. Gültas. Exploring the potential of incremental feature selection to improve genomic prediction accuracy. *Genetics Selection Evolution*, 55(1):78, Nov 2023. doi: 10.1186/s12711-023-00853-8.

N. Heslot, H. Yang, M. E. Sorrells, and J.-L. Jannink. Genomic selection in plant breeding: a comparison of models. *Crop Science*, 52(1):146–160, 2012.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006. doi: 10.1126/science.1127647.

N. Hofheinz and M. Frisch. Heteroscedastic ridge regression approaches for genome-wide prediction with a focus on computational efficiency and accurate effect estimation. *G3: Genes, Genomes, Genetics*, 4(3):539–546, 03 2014. doi: 10.1534/g3.113.010025.

F. Hospital. Size of donor chromosome segments around introgressed loci and reduction of linkage drag in marker-assisted backcross programs. *Genetics*, 158(3): 1363–1379, 2001.

T. Islam, C. H. Kim, H. Iwata, H. Shimono, and A. Kimura. DeepCGP: A deep learning method to compress genome-wide polymorphisms for predicting phenotype of rice. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20 (3):2078–2088, 2023. doi: 10.1109/TCBB.2022.3231466.

Y. Jiang, R. H. Schmidt, and J. C. Reif. Haplotype-based genome-wide prediction models exploit local epistatic interactions among markers. *G3: Genes, Genomes, Genetics*, 8(5):1687–1699, 2018. doi: 10.1534/g3.117.300548.

M. John, F. Haselbeck, R. Dass, C. Malisi, P. Ricca, C. Dreischer, S. J. Schultheiss, and D. G. Grimm. A comparison of classical and machine learning-based phenotype prediction methods on simulated data and three plant species. *Frontiers in Plant Science*, 13, 2022. doi: 10.3389/fpls.2022.932512.

D. Jones, R. Fornarelli, M. Derbyshire, M. Gibberd, K. Barker, and J. Hane. The pursuit of genetic gain in agricultural crops through the application of machine-learning to genomic prediction. *Frontiers in Genetics*, Volume 14 - 2023, 2023. doi: 10.3389/fgene.2023.1186782.

J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, Aug 2021. doi: 10.1038/s41586-021-03819-2.

F. Jurado-Ruiz, D. Rousseau, J. A. Botia, and M. J. Aranzana. GenoDrawing: An autoencoder framework for image prediction from SNP markers. *PLANT PHENOMICS*, 5, NOV 3 2023. doi: 10.34133/plantphenomics.0113.

S. Khaki and L. Wang. Crop yield prediction using deep neural networks. *Frontiers in Plant Science*, Volume 10 - 2019, 2019. doi: 10.3389/fpls.2019.00621.

S. Khaki, Z. Khalilzadeh, and L. Wang. Predicting yield performance of parents in plant breeding: A neural collaborative filtering approach. *PLOS ONE*, 15(5): 1–13, 05 2020. doi: 10.1371/journal.pone.0233382.

D. R. Kick and J. D. Washburn. Ensemble of best linear unbiased predictor, machine learning and deep learning models predict maize yield better than each model alone. *in silico Plants*, 5(2):diad015, 09 2023. doi: 10.1093/insilicoplants/diad015.

D. R. Kick, J. G. Wallace, J. C. Schnable, J. M. Kolkman, B. Alaca, T. M. Beissinger, J. Edwards, D. Ertl, S. Flint-Garcia, J. L. Gage, C. N. Hirsch, J. E. Knoll, N. de Leon, D. C. Lima, D. E. Moreta, M. P. Singh, A. Thompson, T. Weldekidan, and J. D. Washburn. Yield prediction through integration of genetic, environment, and management data through deep learning. *G3: Genes, Genomes, Genetics*, 13(4):jkad006, 01 2023. doi: 10.1093/g3journal/jkad006.

D. Knoch, C. R. Werner, R. C. Meyer, D. Riewe, A. Abbadi, S. Lücke, R. J. Snowdon, and T. Altmann. Multi-omics-based prediction of hybrid performance in canola. *Theoretical and Applied Genetics*, 134(4):1147–1165, Apr 2021. doi: 10.1007/s00122-020-03759-x.

D. Krenzer, M. Frisch, K. Beckmann, T. Kox, C. Flachenecker, A. Abbadi, R. Snowdon, and E. Herzog. Simulation-based establishment of base pools for a hybrid breeding program in winter rapeseed. *Theoretical and Applied Genetics*, 137(1): 16, Jan 2024. doi: 10.1007/s00122-023-04519-3.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

M. Kuhn and K. Johnson. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. Chapman and Hall/CRC, New York, 2019. ISBN 9781315108230. doi: 10.1201/9781315108230.

P. Kumar. Large language models (LLMs): survey, technical frameworks, and future challenges. *Artificial Intelligence Review*, 57(10):260, Aug 2024. doi: 10.1007/s10462-024-10888-y.

R. Lande and R. Thompson. Efficiency of marker-assisted selection in the improvement of quantitative traits. *Genetics*, 124(3):743–756, 03 1990. doi: 10.1093/genetics/124.3.743.

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

J. Lee, J. H. Chin, S. N. Ahn, and H.-J. Koh. *Brief History and Perspectives on Plant Breeding*, pages 1–14. Springer Netherlands, Dordrecht, 2015. ISBN 978-94-017-9996-6. doi: 10.1007/978-94-017-9996-6_1.

J. Leukel, L. Scheurer, and T. Zimpel. Overinterpretation of evaluation results in machine learning studies for maize yield prediction: A systematic review. *Computers and Electronics in Agriculture*, 230:109892, 2025. doi: 10.1016/j.compag.2024.109892.

J. Li, Z. He, G. Zhou, S. Yan, and J. Zhang. DeepAT: A deep learning wheat phenotype prediction model based on genotype data. *Agronomy*, 14(12), 2024a. doi: 10.3390/agronomy14122756.

J. Li, D. Zhang, F. Yang, Q. Zhang, S. Pan, X. Zhao, Q. Zhang, Y. Han, J. Yang, K. Wang, and C. Zhao. TrG2P: A transfer-learning-based tool integrating multi-trait data for accurate prediction of crop yield. *Plant Communications*, 5(7), Jul 2024b. doi: 10.1016/j.xplc.2024.100975.

M. Liang, T. Chang, B. An, X. Duan, L. Du, X. Wang, J. Miao, L. Xu, X. Gao, L. Zhang, J. Li, and H. Gao. A stacking ensemble learning framework for genomic prediction. *Frontiers in Genetics*, Volume 12 - 2021, 2021. doi: 10.3389/fgene.2021.600040.

V. M. Lourenco, J. O. Ogutu, R. A. Rodrigues, A. Posekany, and H.-P. Piepho. Genomic prediction using machine learning: a comparison of the performance of regularized regression, ensemble, instance-based and deep learning methods on synthetic and empirical data. *BMC genomics*, 25(1):152, 2024. doi: 10.1186/s12864-023-09933-x.

A. E. Melchinger and R. K. Gumber. *Overview of Heterosis and Heterotic Groups in Agronomic Crops*, chapter 3, pages 29–44. John Wiley & Sons, Ltd, 1998. ISBN 9780891186045. doi: 10.2135/cssaspecpub25.c3.

A. E. Melchinger, H. F. Utz, and C. C. Schön. Quantitative trait locus (QTL) mapping using different testers and independent population samples in maize reveals low power of QTL detection and large bias in estimates of QTL effects. *Genetics*, 149(1):383–403, 1998.

C. Messina, J. Garcia-Abadillo, O. Powell, S. Tomura, A. Zare, B. Ganapathysubramanian, and M. Cooper. Toward a general framework for AI-enabled prediction in crop improvement. *Theoretical and Applied Genetics*, 138(7):151, Jun 2025. doi: 10.1007/s00122-025-04928-6.

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

Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai. Kitsune: An ensemble of autoencoders for online network intrusion detection. *CoRR*, abs/1802.09089, 2018. doi: 10.48550/arXiv.1802.09089.

O. A. Montesinos-López, A. Montesinos-López, P. Pérez-Rodríguez, J. A. Barrón-López, J. W. R. Martini, S. B. Fajardo-Flores, L. S. Gaytan-Lugo, P. C. Santana-Mancilla, and J. Crossa. A review of deep learning applications for genomic selection. *BMC Genomics*, 22(1):19, Jan 2021. doi: 10.1186/s12864-020-07319-x.

A. Montesinos-López, O. A. Montesinos-López, S. Ramos-Pulido, B. A. Mosqueda-González, E. A. Guerrero-Arroyo, J. Crossa, and R. Ortiz. Artificial intelligence meets genomic selection: comparing deep learning and GBLUP across diverse plant datasets. *Frontiers in Genetics*, Volume 16 - 2025, 2025. doi: 10.3389/fgene.2025.1568705.

O. A. Montesinos-López, M. Chavira-Flores, Kismiantini, L. Crespo-Herrera, C. Saint Piere, H. Li, R. Fritsche-Neto, K. Al-Nowibet, A. Montesinos-López, and J. Crossa. A review of multimodal deep learning methods for genomic-enabled prediction in plant breeding. *Genetics*, 228(4):iyae161, 11 2024. doi: 10.1093/genetics/iyae161.

W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019. doi: 10.1073/pnas.1900654116.

A. Nawaz, S. S. Khan, and A. Ahmad. Ensemble of autoencoders for anomaly detection in biomedical data: A narrative review. *IEEE Access*, 12:17273–17289, 2024. doi: 10.1109/ACCESS.2024.3360691.

S. E. Page. *The model thinker: What you need to know to make data work for you*. Hachette UK, 2018.

T. Peng, X. Sun, and R. H. Mumm. Optimized breeding strategies for multiple trait integration: I. minimizing linkage drag in single event introgression. *Molecular Breeding*, 33(1):89–104, Jan 2014. doi: 10.1007/s11032-013-9936-7.

T. Pook, M. Schlather, G. de los Campos, M. Mayer, C. C. Schoen, and H. Simianer. Haploblocker: Creation of subgroup-specific haplotype blocks and libraries. *Genetics*, 212(4):1045–1061, 05 2019. doi: 10.1534/genetics.119.302283.

T. Pook, J. Freudenthal, A. Korte, and H. Simianer. Using local convolutional neural networks for genomic prediction. *Frontiers in Genetics*, Volume 11 - 2020, 2020. doi: 10.3389/fgene.2020.561497.

A. Powadi, T. Z. Jubery, M. C. Tross, J. C. Schnable, and B. Ganapathysubramanian. Disentangling genotype and environment specific latent features for improved trait prediction using a compositional autoencoder. *Frontiers in Plant Science*, 15, 2024. doi: 10.3389/fpls.2024.1476070.

P. Probst, A.-L. Boulesteix, and B. Bischl. Tunability: Importance of hyperparameters of machine learning algorithms. *Journal of Machine Learning Research*, 20 (53):1–32, 2019.

M. Pérez-Enciso and L. M. Zingaretti. A guide on deep learning for complex trait genomic prediction. *Genes*, 10(7), 2019. doi: 10.3390/genes10070553.

J. Rutkoski, J. Benson, Y. Jia, G. Brown-Guedira, J.-L. Jannink, and M. Sorrells. Evaluation of genomic prediction methods for fusarium head blight resistance in wheat. *The Plant Genome*, 5(2), 2012. doi: 10.3835/plantgenome2012.02.0001.

F. B. Sarijaloo, M. Porta, B. Taslimi, and P. M. Pardalos. Yield performance estimation of corn hybrids using machine learning algorithms. *Artificial Intelligence in Agriculture*, 5:82–89, 2021. doi: 10.1016/j.aiia.2021.05.001.

X. Shen, M. Alam, F. Fikse, and L. Rönnegård. A novel generalized ridge regression method for quantitative genetics. *Genetics*, 193(4):1255–1268, 04 2013. doi: 10.1534/genetics.112.146720.

R. Shwartz-Ziv and A. Armon. Tabular data: Deep learning is not all you need. *Information Fusion*, 81:84–90, 2022. doi: 10.1016/j.inffus.2021.11.011.

N. A. Spielberg, M. Brown, N. R. Kapania, J. C. Kegelman, and J. C. Gerdes. Neural network vehicle models for high-performance automated driving. *Science Robotics*, 4(28):eaaw1975, 2019. doi: 10.1126/scirobotics.aaw1975.

G. F. Sprague and L. A. Tatum. General vs. specific combining ability in single crosses of corn. *Agronomy Journal*, 34(10):923–932, 1942. doi: 10.2134/agronj1942.00021962003400100008x.

Syngenta. Syngenta crop challenge in analytics, 2021. URL `https://www.ideaconnection.com/syngenta-crop-challenge/challenge.php/`.

W. Tadesse, M. Sanchez-Garcia, S. G. Assefa, A. Amri, Z. Bishaw, F. C. Ogbonnaya, and M. Baum. Genetic gains in wheat breeding and its role in feeding the world, 2019.

A. Talukder, C. Barham, X. Li, and H. Hu. Interpretation of deep learning in genomics and epigenomics. *Briefings in Bioinformatics*, 22(3):bbaa177, 08 2020. doi: 10.1093/bib/bbaa177.

M. Tan and Q. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 09–15 Jun 2019.

G. Taş, T. Westerdijk, E. Postma, P. M. A. G. Consortium, J. H. Veldink, A. Schönhuth, and M. Balvert. Computing linkage disequilibrium aware genome embeddings using autoencoders. *Bioinformatics*, 40(6):btae326, 05 2024. doi: 10.1093/bioinformatics/btae326.

The VSNi Team. *asreml: Fits Linear Mixed Models using REML*, 2023. R package version 4.2.0.267.

D. Tilman, C. Balzer, J. Hill, and B. L. Befort. Global food demand and the sustainable intensification of agriculture. *Proceedings of the National Academy of Sciences*, 108(50):20260–20264, 2011. doi: 10.1073/pnas.1116437108.

M. Togninalli, X. Wang, T. Kucera, S. Shrestha, P. Juliana, S. Mondal, F. Pinto, V. Govindan, L. Crespo-Herrera, J. Huerta-Espino, R. P. Singh, K. Borgwardt, and J. Poland. Multi-modal deep learning improves grain yield prediction in wheat breeding by fusing genomics and phenomics. *Bioinformatics*, 39(6):btad336, 05 2023. doi: 10.1093/bioinformatics/btad336.

V. Tomar, G. S. Dhillon, D. Singh, R. P. Singh, J. Poland, A. A. Chaudhary, P. K. Bhati, A. K. Joshi, and U. Kumar. Evaluations of genomic prediction and identification of new loci for resistance to stripe rust disease in wheat (*Triticum aestivum* L.). *Frontiers in Genetics*, Volume 12 - 2021, 2021. doi: 10.3389/fgene.2021.710485.

S. Tomura, M. J. Wilkinson, M. Cooper, and O. Powell. Improved genomic prediction performance with ensembles of diverse models. *G3 Genes—Genomes—Genetics*, 15(5):jkaf048, 03 2025. doi: 10.1093/g3journal/jkaf048.

J. Tong, Z. T. Tarekegn, D. Jambuthenne, S. Alahmad, S. Periyannan, L. Hickey, E. Dinglasan, and B. Hayes. Stacking beneficial haplotypes from the Vavilov wheat collection to accelerate breeding for multiple disease resistance. *Theoretical and Applied Genetics*, 137(12):274, Nov 2024. doi: 10.1007/s00122-024-04784-w.

M. C. Tross, M. W. Grzybowski, T. Z. Jubery, R. J. Grove, A. V. Nishimwe, J. V. Torres-Rodriguez, G. Sun, B. Ganapathysubramanian, Y. Ge, and J. C. Schnable. Data driven discovery and quantification of hyperspectral leaf reflectance phenotypes across a maize diversity panel. *The Plant Phenome Journal*, 7(1):e20106, 2024. doi: 10.1002/ppj2.20106.

J. R. Ubbens and I. Stavness. Deep plant phenomics: A deep learning platform for complex plant phenotyping tasks. *Frontiers in Plant Science*, Volume 8 - 2017, 2017. doi: 10.3389/fpls.2017.01190.

P. M. VanRaden. Efficient methods to compute genomic predictions. *Journal of Dairy Science*, 91(11):4414–4423, 2008.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

T. Verdonck, B. Baesens, M. Óskarsdóttir, and S. vanden Broucke. Special issue on feature engineering editorial. *Machine Learning*, 113(7):3917–3928, Jul 2024. doi: 10.1007/s10994-021-06042-2.

K. Villiers, K. P. Voss-Fels, E. Dinglasan, B. Jacobs, L. Hickey, and B. J. Hayes. Evolutionary computing to assemble standing genetic diversity and achieve long-term genetic gain. *The Plant Genome*, 17(2):e20467, 2024. doi: 10.1002/tpg2.20467.

Z. G. Vitezica, L. Varona, and A. Legarra. On the additive and dominant variance and covariance of individuals within the genomic selection scope. *Genetics*, 195 (4):1223–1230, 12 2013. doi: 10.1534/genetics.113.155176.

K. P. Voss-Fels, A. Stahl, B. Wittkop, C. Lichthardt, S. Nagler, T. Rose, T.-W. Chen, H. Zetzsche, S. Seddig, M. Majid Baig, A. Ballvora, M. Frisch, E. Ross, B. J. Hayes, M. J. Hayden, F. Ordon, J. Leon, H. Kage, W. Friedt, H. Stützel, and R. J. Snowdon. Breeding improves wheat productivity under contrasting agrochemical input levels. *Nature Plants*, 5(7):706–714, Jul 2019. doi: 10.1038/s41477-019-0445-5.

K. Wang, M. A. Abid, A. Rasheed, J. Crossa, S. Hearne, and H. Li. DNNGP, a deep neural network-based method for genomic prediction using multi-omics data in plants. *Molecular Plant*, 16(1):279–293, 2023. doi: 10.1016/j.molp.2022.11.004.

J. D. Washburn, E. Cimen, G. Ramstein, T. Reeves, P. O'Briant, G. McLean, M. Cooper, G. Hammer, and E. S. Buckler. Predicting phenotypes from genetic, environment, management, and historical data using CNNs. *Theoretical and Applied Genetics*, 134(12):3997–4011, Dec 2021. doi: 10.1007/s00122-021-03943-7.

S. E. Weber, M. Frisch, R. J. Snowdon, and K. P. Voss-Fels. Haplotype blocks for genomic prediction: a comparative evaluation in multiple crop datasets. *Frontiers in Plant Science*, Volume 14 - 2023, 2023. doi: 10.3389/fpls.2023.1217589.

C. Westhues, G. Mahone, S. Silva, P. Thorwarth, M. Schmidt, J.-C. Richter, H. Simianer, and T. Beissinger. Prediction of maize phenotypic traits with genomic and environmental predictors using gradient boosting frameworks. *Frontiers in Plant Science*, 12, 11 2021. doi: 10.3389/fpls.2021.699589.

M. Westhues, T. A. Schrag, C. Heuer, G. Thaller, H. F. Utz, W. Schipprack, A. Thiemann, F. Seifert, A. Ehret, A. Schlereth, M. Stitt, Z. Nikoloski, L. Willmitzer, C. C. Schön, S. Scholten, and A. E. Melchinger. Omics-based hybrid prediction in maize. *Theoretical and Applied Genetics*, 130(9):1927–1939, Sep 2017. doi: 10.1007/s00122-017-2934-0.

J. C. Whittaker, R. Thompson, and M. C. Denham. Marker-assisted selection using ridge regression. *Genetical Research*, 75(2):249–252, 2000. doi: 10.1017/S0016672399004462.

D. Wolpert and W. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997. doi: 10.1109/4235.585893.

D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992. doi: 10.1016/S0893-6080(05)80023-1.

Z. Xie, L. Weng, J. He, X. Feng, X. Xu, Y. Ma, P. Bai, and Q. Kong. PNNGS, a multi-convolutional parallel neural network for genomic selection. *Frontiers in Plant Science*, Volume 15 - 2024, 2024a. doi: 10.3389/fpls.2024.1410596.

Z. Xie, X. Xu, L. Li, C. Wu, Y. Ma, J. He, S. Wei, J. Wang, and X. Feng. Residual networks without pooling layers improve the accuracy of genomic predictions. *Theoretical and Applied Genetics*, 137(6):138, May 2024b. doi: 10.1007/s00122-024-04649-2.

S. Xu. Theoretical basis of the Beavis effect. *Genetics*, 165(4):2259–2268, 2003.

N. D. Young. A cautiously optimistic vision for marker-assisted breeding. *Molecular Breeding*, 5:505–510, 1999.

C. Zenke-Philippi, A. Thiemann, F. Seifert, T. Schrag, A. E. Melchinger, S. Scholten, and M. Frisch. Prediction of hybrid performance in maize with a ridge regression model employed to DNA markers and mRNA transcription profiles. *BMC Genomics*, 17(1):262, Mar 2016. doi: 10.1186/s12864-016-2580-y.

C. Zenke-Philippi, M. Frisch, A. Thiemann, F. Seifert, T. Schrag, A. E. Melchinger, S. Scholten, and E. Herzog. Transcriptome-based prediction of hybrid performance with unbalanced data from a maize breeding programme. *Plant Breeding*, 136(3): 331–337, 2017. doi: 10.1111/pbr.12482.

H. Zhao, D. Nettleton, M. Soller, and J. C. M. Dekkers. Evaluation of linkage disequilibrium measures between multi-allelic markers as predictors of linkage disequilibrium between markers and QTL. *Genetical Research*, 86(1):77–87, 2005. doi: 10.1017/S001667230500769X.

Y. Zhao, Z. Li, G. Liu, Y. Jiang, H. P. Maurer, T. Würschum, H.-P. Mock, A. Matros, E. Ebmeyer, R. Schachschneider, E. Kazman, J. Schacht, M. Gowda, C. F. H. Longin, and J. C. Reif. Genome-based establishment of a high-yielding heterotic pattern for hybrid wheat breeding. *Proceedings of the National Academy of Sciences*, 112(51):15624–15629, 2015. doi: 10.1073/pnas.1514547112.

L. M. Zingaretti, S. A. Gezan, L. F. V. Ferrão, L. F. Osorio, A. Monfort, P. R. Muñoz, V. M. Whitaker, and M. Pérez-Enciso. Exploring deep learning for complex trait genomic prediction in polyploid outcrossing species. *Frontiers in Plant Science*, 11, 2020. doi: 10.3389/fpls.2020.00025.

Acknowledgments

I would like to express my sincere gratitude to my academic supervisor, Prof. Dr. Matthias Frisch, for his continuous support, many suggestions and his advice during my thesis work.

I am also grateful to my colleagues at the Department of Biometry and Population Genetics for the pleasant working atmosphere and their help in all circumstances, especially to Dr. Carola Zenke-Philippi for her valuable feedback and for proof-reading this manuscript.

Many thanks to Prof. Dr. Rod Snowdon for being my second supervisor.

Finally, I wish to thank my partner, Sarina Geisler, and my family – Susanne, Gerald, and Max – who have always provided support during difficult times.

Eidesstattliche Erklärung

Ich erkläre: Ich habe die vorgelegte Dissertation selbständig und ohne unerlaubte fremde Hilfe und nur mit den Hilfen angefertigt, die ich in der Dissertation angegeben habe.

Alle Textstellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen sind, und alle Angaben, die auf mündlichen Auskünften beruhen, sind als solche kenntlich gemacht.

Bei den von mir durchgeführten und in der Dissertation erwähnten Untersuchungen habe ich die Grundsätze guter wissenschaftlicher Praxis, wie sie in der „Satzung der Justus-Liebig-Universität Gießen zur Sicherung guter wissenschaftlicher Praxis“ niedergelegt sind, eingehalten.

Gießen, 22. Juli 2025 Philipp Georg Heilmann