Plant Methods

METHODOLOGY

Open Access

Improved genomic prediction using machine learning with Variational Bayesian sparsity

Qingsen Yan[5], Mario Fruzangohar[2], Julian Taylor[2*], Dong Gong[3], James Walter[4], Adam Norman[4], Javen Qinfeng Shi[1] and Tristan Coram[4]

Abstract

Background Genomic prediction has become a powerful modelling tool for assessing line performance in plant and livestock breeding programmes. Among the genomic prediction modelling approaches, linear based models have proven to provide accurate predictions even when the number of genetic markers exceeds the number of data samples. However, breeding programmes are now compiling data from large numbers of lines and test environments for analyses, rendering these approaches computationally prohibitive. Machine learning (ML) now offers a solution to this problem through the construction of fully connected deep learning architectures and high parallelisation of the predictive task. However, the fully connected nature of these architectures immediately generates an over-parameterisation of the network that needs addressing for efficient and accurate predictions.

Results In this research we explore the use of an ML architecture governed by variational Bayesian sparsity in its initial layers that we have called VBS-ML. The use of VBS-ML provides a mechanism for feature selection of important markers linked to the trait, immediately reducing the network over-parameterisation. Selected markers then propagate to the remaining fully connected feed-forward components of the ML network to form the final genomic prediction. We illustrated the approach with four large Australian wheat breeding data sets that range from 2665 lines to 10375 lines genotyped across a large set of markers. For all data sets, the use of the VBS-ML architecture improved genomic prediction accuracy over legacy linear based modelling approaches.

Conclusions An ML architecture governed under a variational Bayesian paradigm was shown to improve genomic prediction accuracy over legacy modelling approaches. This VBS-ML approach can be used to dramatically decrease the parameter burden on the network and provide a computationally feasible approach for improving genomic prediction conducted with large breeding population numbers and genetic markers.

Keywords Machine learning, Genomic prediction, Linear mixed models, Bayesian, Variational inference, Feature selection

*Correspondence:
Julian Taylor
julian.taylor@adelaide.edu.au
Full list of author information is available at the end of the article

Introduction

Genomic Selection (GS), through genomic prediction, has proven to be a useful tool for achieving rapid genetic gain in livestock and plant breeding programmes. Since its inception in [1] genomic prediction approaches have mostly focused on using hierarchical linear models for assessing the relative genetic merit of lines for phenotypic traits of interest, with various prediction accuracies developed from these models [2]. Historically, these approaches were piecemeal, estimating proxy QTL effects using simple marker regression scans of the whole genome [1]. This was quickly extended to using the complete set of genetic markers in linear based models by considering the marker effects as random effects variables with various distributional properties [3–5]. When the number of markers became large, penalization methods such as the ones used in [3] and [4], became a useful tool for mechanistically pushing small marker random effects to zero and giving rise to various Bayesian variable selection methods [6, 7]. Once the number of markers became routinely larger than the number of individuals being studied, computationally efficient methods were developed that re-dimensionalised the genetic marker information in the models into an additive genomic relationship matrix (GRM) that allowed direct prediction of the relative performance of lines [8–12]. In more complex experimental scenarios, such as plant breeding programmes the inclusion of a dense GRM in a one-stage linear model can be computationally challenging due to requirements to involve the GRM in iterative parameter estimation algorithms [13, 14]. As the number of breeding lines and the number of testing environments increases the model becomes computationally cumbersome to solve [13] and modern computing techniques such as matrix algebra parallelisation [15–17] and machine learning (ML) approaches [18, 19] have become common place in GS research.

ML has now been widely adopted in crop and livestock agriculture when there is sufficient data complexity or computationally difficult tasks that require undertaking [20]. In the context of genomic prediction of agricultural traits, various deep learning techniques have been researched to understand their potential to improve prediction accuracy over legacy modelling approaches [18, 19]. These approaches use the complete set of genetic markers spanning the genome as input features to a neural network and the output, a trait of interest, is optimally learned through the network using various computationally intensive statistical modelling techniques. In most ML based genomic prediction cases the deep learning architecture has consisted of a type of artificial neural network, called a multi-layer perceptron (MLP), due to its ability to learn a high level of abstraction from the complex connection between the phenotype and genotype data [21–25]. In crops, such a maize and wheat, where grain yield or end use quality traits are highly quantitative in nature the use of MLP networks has also been shown to improve genomic prediction accuracy over more conventional ML approaches, such as convolutional neural networks [25–27].

Neural networks can be potentially complex and fixed architectural aspects of the network, such as the number of layers and the number of nodes (or neurons) within a layer, can be tuned in various ways to optimise the learning potential of the network [28]. In situations where the network becomes highly over-parameterised, various dropout techniques have been proposed for reducing the computational burden through the reduction of less important connections between layers [29, 30]. Historically, these techniques were based on random dropout of neurons or weights between layers [31, 32] but quickly expanded to more distributional based methods [33, 34] that include the use of regularizers such as the L1 or Lasso. Extensions of these regularizaton techniques are now focussing on using appropriate Bayesian hierarchical priors [35] to conduct variable selection of important markers in the initial stages of the network [36].

The objectives of this study were to evaluate the accuracy of a cutting edge ML based approach for conducting genomic prediction that involves the variational Bayesian sparsity (VBS) technique derived in [35] and L1-regularization for reducing the over-paramaterisation burden on the proposed MLP deep learning network. We have called this VBS-ML and to illustrate the effectiveness of the approach we conducted VBS-ML genomic prediction of grain yield collected from a large wheat breeding panel phenotyped for four years and genotyped with a high quality set of SNP markers. We compared the results of the newly proposed ML deep learning network with a more naive ML network as well as a more classical genomic prediction using linear mixed models (LMMs) and Bayesian regression methods BayesA and BayesB. In nearly all cases the VBS-ML network showed a marked improvement in genomic prediction accuracy compared to the naive ML network or other approaches. In addition, the genetic marker features selected from a given year or combined years were also shown to more accurately predict subsequent years compared to other prediction methods used in this research. This suggests the VBS-ML approach may potentially be a useful genomic selection tool for plant breeding programmes.

Material and methods

Plant material and phenotype data

Plant material used within this study consists of early and advanced generation breeding lines from within Australian

Grain Technologies' wheat breeding programmes. This material was spread across four field trials in the years 2014, 2016, 2017 and 2018. The 2014 field trial contained early and advanced breeding lines, comprised of material adapted for southern Australia (early and advanced lines) as well as material adapted for western and eastern Australia (advanced lines). A total of 10,375 genotypes were included in the trial and were planted in non-replicated randomised design with randomised grid checks (1 check per 11 plots). Further details of this trial can be found in [13]. Trials in 2016, 2017 and 2018 contained advanced breeding lines adapted to southern Australia. A total of 2869, 2869 and 2665 genotypes were included in the 2016, 2017 and 2018 trials respectively, with each of these trials planted in a completely randomised design with partial replication at a 1.25x level. All trials were sown as small-scale yield plots of 3 m^2 at Roseworthy, South Australia (−34.52, 138.69), and managed according to best local practices. Phenotype data was collected as plot level grain yield from a mechanical small-scale plot harvester.

Genotype data

In this research we used a whole genome set of genetic markers from a custom 20K SNP Affymetrix array that spanned the 21 chromosomes of the wheat genome. These markers are known to be of high quality and have been used as the basis for several published grains research articles [13, 14, 37]. To simplify the usage of the markers, we have used imputed markers only where missing alleles have been imputed using the *k*-NN nearest neighbor algorithm developed in [38] and used extensively in [39] [see 42, norm17]. For all methods below we define the genetic marker matrix for a set of lines in any given year as $\boldsymbol{M} = [\boldsymbol{m}_1 \dots \boldsymbol{m}_p]$ of dimension r lines by p markers spanning the complete 21 wheat chromosomes.

Adjusted yield derivation

Preceding genomic prediction using linear based models and ML we derive an adjusted yield prediction for each set of lines within a year using a spatial LMM that partitions and estimates genetic and non-genetic sources of variation. We specify this model in a general manner to allow for independent modelling of each trial conducted at Roseworthy. Let $\boldsymbol{y}_e = [y_{e1}, \dots, y_{en}]$ be the n raw yield observations from a field trial within a year. The LMM was then of the form

$$\boldsymbol{y}_e = \mathbf{1}_n\mu + \boldsymbol{Z}_e\boldsymbol{u} + \boldsymbol{Z}_g\boldsymbol{g} + \boldsymbol{e}, \tag{1}$$

where μ is the fixed grand mean parameter and $\mathbf{1}_n$ is an n length vector of ones. The $\boldsymbol{u}$ is a vector of random effects partitioned as $\boldsymbol{u} = [\boldsymbol{u}_1^T \dots \boldsymbol{u}_s^T]^T$ with conformably partitioned indicator matrix $\boldsymbol{Z}_e = [\boldsymbol{Z}_1 \dots \boldsymbol{Z}_s]$. This partitioning is typically the result of including multiple random effect terms in the LMM that are required to account for non-genetic sources of variation, such as design induced effects or non-linear variation that may be occurring across the Rows or Range of the experiments. The complete set of random effects follow the distribution $\boldsymbol{u} \sim N(\mathbf{0}, \boldsymbol{G})$ where $\boldsymbol{G} = \oplus_{i=1}^{s} \boldsymbol{G}_i$ where $\oplus$ is the so-called direct sum structure that generates a block diagonal matrix and $\boldsymbol{G}_i$ is typically simplified to $\sigma_{ei}^2 \boldsymbol{I}_{n_i}$. Similarly the residual error term, $\boldsymbol{e}$, was partitioned to $\boldsymbol{e} = [\boldsymbol{e}_1^T \dots \boldsymbol{e}_t^T]^T$ and distributed as $\boldsymbol{e} \sim N(\mathbf{0}, \boldsymbol{R})$ where $\boldsymbol{R} = \oplus_{i=1}^{t} \boldsymbol{R}_i$. Here, $\boldsymbol{R}_i = \sigma_i^2 \boldsymbol{\Sigma}_i^{(ro)} \otimes \boldsymbol{\Sigma}_i^{(ra)}$ containing a paramaterization for a separable AR1 by AR1 (AR1 = autoregressive process of order 1) correlation process that adequately captures the similarity of the observations across distinct Range and Rows of the experimental design for the *it*th trial within a year. The final term on the right hand side of (1), contained a vector of genetic effects, $\boldsymbol{g}$, of length r with an associated indicator matrix $\boldsymbol{Z}_g$ that assigns the line to the appropriate yield plot in the experiment. The genetic effects capture the underlying genetic variation of yield across the breeding population around the experimental average μ. The distribution of the effects are assumed to be $\boldsymbol{g} \sim N(\mathbf{0}, \sigma_g^2 \boldsymbol{I}_r)$ where σ_g^2 is the genetic variance.

Empirical best linear unbiased predictors (eBLUPs) of the genetic effects $\tilde{\boldsymbol{g}}$ were then extracted from the fitted LMM and generalized heritabilities were calculated using [40]. The techniques of [41] were then used to conduct a de-regression to derive adjusted yield values for each line, namely

$$y_i = \hat{\mu} + \frac{\tilde{g}_i}{1 - PEV_i/\hat{\sigma}_g^2}, \qquad i = 1, \dots, r. \tag{2}$$

where $\tilde{\boldsymbol{g}}_i$ and PEV_i are the eBLUP and prediction error variance of the ith line respectively and $\hat{\sigma}_g^2$ is the Residual Maximum Likelihood (REML) estimate [see 42] of the genetic variance.

Linear genomic prediction

We define a general form for the linear genomic prediction model of the adjusted yield, namely

$$\boldsymbol{y} = \mathbf{1}_n\mu^* + \boldsymbol{M}\boldsymbol{q} + \boldsymbol{e}^* \tag{3}$$

where $\mathbf{1}_n\mu^*$ is the grand mean and $\boldsymbol{q}$ is a p length vector of additive marker effects with a distribution that varies depending on the method used for prediction. In this reduced model the residual errors, $\boldsymbol{e}^*$, are used to account for all non-additive genetic variation and have distribution $\boldsymbol{e}^* \sim N(\mathbf{0}, \sigma^2 \boldsymbol{I}_r)$.

LMM genomic prediction

For cases where the distribution of the marker effects are assumed to have an unconditional Gaussian of the form $\boldsymbol{q} \sim N(\boldsymbol{0}, \sigma_a^2 \boldsymbol{I}_p)$, (3) can be defined as a genomic prediction LMM. If $p >> r$ then it is computationally convenient to re-write the LMM as

$$\boldsymbol{y} = \boldsymbol{1}_r \mu^* + \boldsymbol{a} + \boldsymbol{e}^* \tag{4}$$

where $\boldsymbol{a}$ is an r length vector of additive line effects with distribution $\boldsymbol{a} \sim N(\boldsymbol{0}, \sigma_a^2 \boldsymbol{G}_a)$. Here, σ_a^2 is the additive genetic variance, $\boldsymbol{G}_a = \boldsymbol{M}\boldsymbol{M}^T/s$ represents an $r \times r$ additive relationship matrix reflecting the marker based relationships between the lines and is scaled using $s =$ trace $(\boldsymbol{M}\boldsymbol{M}^T)/r$ [43].

Using the techniques of [44], (4) can be solved and the genomic predictions can be immediately written as

$$\tilde{\boldsymbol{y}} = \boldsymbol{1}_r \hat{\mu}^* + \tilde{\boldsymbol{a}}$$

where

$$\hat{\mu}^* = (\boldsymbol{1}_r^T \boldsymbol{H}^{-1} \boldsymbol{1})^{-1} \boldsymbol{1}_r^T \boldsymbol{H}^{-1} \boldsymbol{y}$$
$$\tilde{\boldsymbol{a}} = \boldsymbol{G}_a \boldsymbol{H}^{-1} (\boldsymbol{y} - \boldsymbol{1}_r \hat{\mu}^*)$$

where $\boldsymbol{H} = \sigma^2 \boldsymbol{I}_r + \sigma_a^2 \boldsymbol{G}_a$. Typically, σ^2 and σ_a^2 are replaced by their Residual Maximum Likelihood (REML) estimates and $\tilde{\boldsymbol{a}}$ becomes an empirically based additive genomic prediction of the adjusted yield.

BayesA and BayesB genomic prediction

BayesA and BayesB are a form of hierarchical Bayesian regression based on the linear model (3). In this model we now consider additional structure for the marker effects, $\boldsymbol{q} = (q_1, \ldots, q_p)$ such that the ith marker effect has a distribution of the form

$$q_i \mid \sigma_i^2, \pi \sim \begin{cases} 0 & \text{probability} \quad \pi \\ N(0, \sigma_i^2) & \text{probability} \quad 1 - \pi \end{cases}$$
$$\sigma_i^2 \mid \nu, s_q^2 \sim \chi^{-2}(\nu, s_q^2)$$

where $\chi^{-2}(\nu, s_q^2)$ represents a scaled inverse chi-square distribution with ν degrees of freedom and scale parameter s_q^2, or equivalently an $\Gamma^{-1}(\nu/2, s_q^2 \nu/2)$. After integrating over the marker variances, $\sigma_i^2, i = 1, \ldots, p$, we can obtain marginal marker effects of the form

$$q_i \mid \pi \sim \begin{cases} 0 & \text{probability} \quad \pi \\ t(0, s_q^2, \nu) & \text{probability} \quad 1 - \pi \end{cases} \qquad i = 1, \ldots, p.$$

BayesB considers the complete structure derived here and BayesA is a special case of BayesB where $\pi = 0$. In both cases the non-zero marginal effects have a t-distribution with ν degrees of freedom and scale parameter s_q^2 reflecting the requirement to capture the important positive and negative marker effects and shrink negligible effects close to zero. The spike and slab prior of the marginal marker effects ensures BayesB acts like a feature selection method and consequently provides a useful comparison to the ML feature selection method outlined in the next section.

ML genomic prediction

Based on its previous successful use in genomic prediction we have chosen to use an MLP-based machine learning scheme. The MLP is a densely connected network used in deep learning and is a typical feed-forward neural network that does not assume a particular structure of the input features [25]. We investigated the use of MLP architecture presented in Fig. 1. The MLP consisted of input layer that correspond to a fixed number of neurons where the complete set of neurons denote a set of SNP marker features

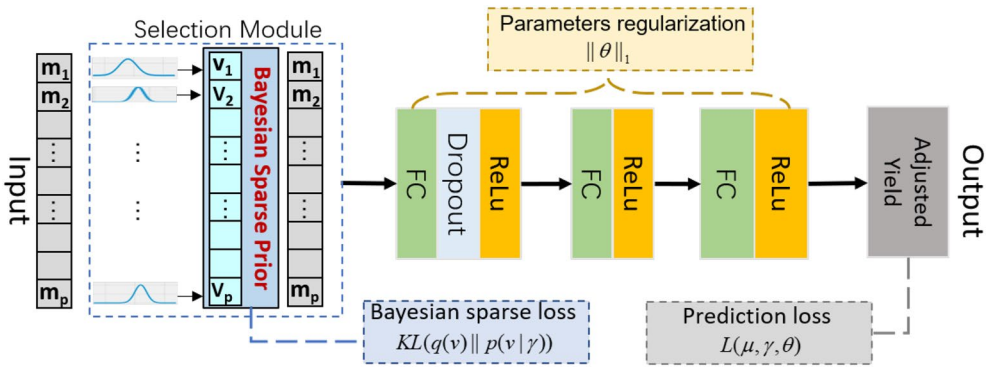

Fig. 1 The proposed variational Bayesian ML architecture that includes the initial feature selection module and additional hidden layers in the prediction module

from a row of $\boldsymbol{M}$. The array of hidden layers then capture non-linear features from the output of the previous layers were each of the hidden layers may consist of varying number of neurons. Hidden Layers are usually fully connected (FC) between neurons with each connection given its own weight parameter. The output layer receives the outputs of the last hidden layer and provides the prediction, in this case a prediction of adjusted grain yield. The weights of the whole network are parameters that require learning from the training set and their estimates determine the effectiveness of each neurons contribution towards the final prediction. In the MLP architecture presented in Fig. 1, two major sub-networks are proposed including a feature selection module (for marker selection) and a prediction module (for result estimation). For each module, the component of the MLP model is described in more detail in the sections below along with a derivation of the objective function governing the complete network optimisation. For ease of notation we have considered a single sample in Fig. 1 with recognition this network is applied to all training samples.

Feature selection module

Let $\boldsymbol{m} = [m_1, \ldots, m_p]^T$ represent a p length vector of genetic marker input features for a line or sample in the dataset. As shown in Fig. 1, we introduce a feature selection module to adaptively select the important genetic markers. Within this feature selection module we introduce a hidden selection layer with an output defined by the model

$$\boldsymbol{x} = \boldsymbol{m} \odot \boldsymbol{\nu} \tag{5}$$

where $\odot$ denotes the element wise multiplication and $\boldsymbol{\nu} = [\nu_1, \ldots, \nu_p]^T$ is a vector of weights. The weights $\boldsymbol{\nu}$ will be learned during network optimisation, and to enforce the sparsity of the selection, we assume they will be governed by a hierarchical sparse prior distribution. Before outlining the methodological details of this hierarchical prior and the associated learning objectives, we introduce the remainder of the neural network structures and operations.

Prediction module

After the selection module we then utilise further layers of an MLP (see Fig. 1) to refine the feature representation and prediction. Let $\boldsymbol{w}_j$ be a p length vector of weights for the connections between the complete set of outputs from the first hidden layer to the jth neuron in the second hidden layer. The output for the jth neuron is then

$$z_{1j} = \text{ ReLU } \left(b_0 + \sum_{s=1}^{p} w_{js} x_s \right) \tag{6}$$

where b_0 was the bias for the first hidden layer and ReLU denotes the the rectifier linear unit activation function. For a full set of connections between layers, (6) can be generalized to become

$$\boldsymbol{z}_1 = \text{ ReLU } (b_0 + \boldsymbol{W}_1 \boldsymbol{x})$$

where $\boldsymbol{x}$ is the p length vector of outputs from the first hidden layer and $\boldsymbol{W}_1$ is a $(n_1 \times p)$ matrix with jth row $\boldsymbol{w}_j$. Given an arbitrary k fully connected hidden layers the output from the kth hidden layer can be immediately written as

$$\boldsymbol{z}_k = \text{ ReLU } (b_{k-1} + \boldsymbol{W}_k \boldsymbol{z}_{k-1}).$$

where b_{k-1} and $\boldsymbol{z}_{k-1}$ is the bias and output from the $k-1$ hidden layer and $\boldsymbol{W}_k$ are the $(n_k \times n_{k-1})$ matrix of weights. For the data sets used in this research we trained models using an MLP containing a prediction module with $k = 3$ FC hidden layers where $(n_1, n_2, n_3) = (256, 128, 1)$ with the last layer as the output layer. We utilized one dropout layer after the first layer.

Bayesian sparse prior for featue selection and objective function

Following [35], we assume the hierarchical sparse prior distribution for the feature selection weights $\boldsymbol{\nu}$ is of the form

$$p(\boldsymbol{\nu} \mid \boldsymbol{\gamma}) = \prod_{i=1}^{p} p(\nu_i \mid \gamma_i) = \prod_{i=1}^{p} N(\nu_i \mid 0, \gamma_i)$$
$$p(\boldsymbol{\gamma}) = \prod_{i=1}^{p} p(\gamma_i) = \prod_{i=1}^{p} \mathcal{U}(\gamma_i \mid a, b).$$

Here, $p(\boldsymbol{\nu} \mid \boldsymbol{\gamma})$ is the sparse prior for $\boldsymbol{\nu}$ conditioned on the hyperparameters $\boldsymbol{\gamma}$ and $\mathcal{U}(\gamma_i \mid a, b)$ is a uniform hyperprior with range hyperparameters $[a, b]$. The hyperparameters $\boldsymbol{\gamma}$ will be estimated during the optimisation of the network.

For notational simplicity, we use $\mathcal{D}$ to indicate all the data samples (including both $\boldsymbol{M}$ and $\boldsymbol{y}$) available for training and let $\boldsymbol{\theta}$ be the network parameters defined as the complete set of network weights with the exception of the feature selected weights, $\boldsymbol{\nu}$. Under a Bayesian paradigm, we require the ability to learn the unknown variables or parameters $\boldsymbol{\nu}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\theta}$ from the given data $\mathcal{D}$ through an appropriate formulation of the posterior distribution $p(\boldsymbol{\nu}, \boldsymbol{\gamma}, \boldsymbol{\theta} \mid \mathcal{D})$. Directly estimating this posterior is difficult. To make progress, we formulate the learning task with the variational Bayesian approach outlined in [33]. Firstly, we define a variational posterior for $\boldsymbol{\nu}$ as $q(\boldsymbol{\nu}) = \prod_i q(\nu_i)$ such that

$$q(\nu_i) = \mathcal{N}(\mu_i, \alpha_i \mu_i), \tag{7}$$

where μ_i and $\alpha_i \mu_i$ are the mean and variance of the variational posterior, respectively. With (7), the proposed variational Bayesian learning task can be represented as

$$\min_{\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\alpha},\boldsymbol{\theta}} \mathrm{KL}(q(\boldsymbol{\nu},\boldsymbol{\gamma},\boldsymbol{\theta}) \,||\, p(\boldsymbol{\nu},\boldsymbol{\gamma},\boldsymbol{\theta} \mid \mathcal{D})), \tag{8}$$

where KL denotes the Kullback–Leibler (KL) divergence, $\boldsymbol{\mu}$ is used to parameterise the distribution corresponding to $\boldsymbol{\nu}$, $p(\boldsymbol{\nu},\boldsymbol{\gamma},\boldsymbol{\theta} \mid \mathcal{D})$ is the joint posterior distribution of the parameters given data, and $q(\boldsymbol{\nu},\boldsymbol{\gamma},\boldsymbol{\theta})$ is the corresponding variational joint posterior of $p(\boldsymbol{\nu},\boldsymbol{\gamma},\boldsymbol{\theta} \mid \mathcal{D})$.

We assume $\boldsymbol{\theta}$ and $\boldsymbol{\nu}$ in the prior distribution are independent and this allows the joint posterior to be re-formulated as

$$p(\boldsymbol{\nu},\boldsymbol{\gamma},\boldsymbol{\theta}) = p(\boldsymbol{\nu},\boldsymbol{\gamma})p(\boldsymbol{\theta}).$$

By using the variational posterior to approximate the true posterior, the objective in (8) can be re-formulated as the variational lower bound (VLB) of the marginal likelihood [45] over the data, namely

$$\min_{\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\alpha},\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\theta}) - \mathrm{KL}(q(\boldsymbol{\nu}) \,||\, p(\boldsymbol{\nu} \mid \boldsymbol{\gamma})) - \mathrm{KL}(q(\boldsymbol{\gamma}) \,||\, p(\boldsymbol{\gamma})) - \mathrm{KL}(q(\boldsymbol{\theta}) \,||\, p(\boldsymbol{\theta})),$$

where $\mathcal{L}$ denotes the expected log-likelihood and absorbs the loss term for optimally fitting the data. [35] shows that under a uniform hyperprior for $\boldsymbol{\gamma}$, $\mathrm{KL}(q(\boldsymbol{\gamma}) \,||\, p(\boldsymbol{\gamma}))$ does not depend on $\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\alpha}$ or $\boldsymbol{\theta}$ and can be safely ignored. Assuming $p(\boldsymbol{\theta})$ is a known Laplacian distribution of the form $Laplacian(\boldsymbol{\theta} \mid 0, \lambda_\theta)$ with hyperparameters λ_θ, we can now reduce the VLB to

$$\min_{\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\alpha},\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\theta}) - \lambda_\theta \|\boldsymbol{\theta}\|_1 - \mathrm{KL}(q(\boldsymbol{\nu}) \,||\, p(\boldsymbol{\nu} \mid \boldsymbol{\gamma})),$$

where $\lambda_\theta \|\boldsymbol{\theta}\|_1$ is the derived regularizer of $\boldsymbol{\theta}$. Using the results from [35] as well as a mean absolute error loss function we can then derive a final objective function for jointly estimating feature selection weights $\boldsymbol{\nu}$ and the network parameters $\boldsymbol{\theta}$, namely

$$\mathcal{L}_{\mathrm{obj}} = \sum_{i=1}^{r} |y_i - \tilde{y}_i|/n + \lambda_\theta \|\boldsymbol{\theta}\|_1 + 0.5 \sum_{i=1}^{p} \log(1 + \alpha_i^{-1}), \tag{9}$$

where y_i is the adjusted yield and $\tilde{y}_i$ is the predicted yield for the ith sample. The final term on the right hand side of (9) can be viewed as the variational Bayesian sparsity regularization term to encourage sparsity of the feature selection weights across the p dimensions. The estimated means for the posterior of $\boldsymbol{\nu}$, $\boldsymbol{\mu}$, will be used as the actual sparse weights for feature selection and the parameters α_i, $i = 1,\ldots,p$ control the sparsity of these weights. This derived component then acts as a regularizer for the weights $\boldsymbol{\nu}$ where, for example, if $\alpha_i^{-1} \to 0$ during training, then the corresponding weight ν_i and the associated feature/marker m_i from any given $\boldsymbol{m}$ can potentially be ignored in the subsequent processes of the neural network. After optimisation, a set of sparse weights are have been automatically and adaptively learned. For this reason the feature selection does not need a manually set threshold. This complete ML approach we have called VBS-ML.

Computations and Benchmarking

The LMM was fitted using the flexible LMM R package ASReml-R [46] available in the R statistical computing environment [47] and commercially available from VSNi at https://vsni.co.uk/software/asreml-r. For computational efficiency we incorporated the genetic marker relationship matrix of the lines through the special function `vm()` in the random model formula.

BayesA and BayesB models were computationally fitted using the BGLR R package [48] freely available in the R statistical computing environment [47]. Due to the intractability of the posterior density of the parameters for both hierarchical models, BGLR uses a numerically based Gibbs sampling algorithm. BGLR also assumed some additional structure of some hyperparameters that included $\pi \sim Beta(\pi_0, p_0)$ where we have assumed the probability of marker inclusion to be $1 - \pi_0 = 0.05$ and p_0 is sufficiently large to ensure $E(\pi) = \pi_0$. Additionally, we have assumed $\nu = 4$ and s_q^2 is assumed to be distributed $s_q^2 \sim \Gamma(s, r)$ where $s = 1.1$ and solved for the rate parameter based on an attributed $R^2 = 50\%$ (R-squared) for the linear predictor $\boldsymbol{M\tilde{q}}$. Other MCMC numerical attributes such as number of total iterations, burn in number of iterations and thinning were set at default values.

For the ML networks we used the Pytorch [49] package available in the Python software environment [50] where we assumed a batch size of 512 and $1e^5$ epoch. We used an ADAM optimiser and a cosine annealing learning rate adjustment strategy with a learning rate of $1e^{-4}$ and a weight decay of $5e^{-4}$. We set λ_θ as $1e^{-3}$. For the ADAM optimisation we used β_1=0.9, β_2=0.99. Our network had four fully connected layers and three residual blocks.

For computational benchmarking, we focussed on computational timings for conducting analyses of the 2014 and 2016 data sets only. The 2017 and 2018 data sets are very similar in size to 2016 and would generate redundant information. For the linear genomic prediction approaches we used an Oracle cloud instance (OCI) with 16 OCPU and 256 Gb RAM. For the ML networks, we used an OCI consisting of 12 OCPU with 72Gb RAM and a NVIDIA Tesla P100 with 3584 cores.

Model validation and accuracy

We randomly partitioned the complete data set into training and validation data sets four times. For each split we used a training data set containing 90% of the samples and a validation data set with the remaining 10% of the samples. Training and testing data sets did not overlap. For each split, the models were trained on the training data set only and the accuracy of the genomic prediction was assessed on the validation set.

There has been some recent discussion on the sole use of Pearsons correlation for asessing accuracy [51, 52] when regularization or feature selection methods are used for genomic prediction. For this reason, we have used a combination of Pearsons correlation and relative accuracy (RE) where, for n samples, the RE was defined as

$$RE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \tilde{y}_i|/y_i$$

The use of the observed value in the denominator of each of the elements provides a mechanism to scale the error according to the size of the observations that are being predicted. This RE provides an easily intepretable average proportional difference between the predicted and observed values.

Results

Figure 2 presents the distribution of the adjusted yield values for each of the years. The plot indicates the large differences in average yield across the lines over the years used in this research even though they were similarly located. The variation of the adjusted yield values for 2014 and 2016 were similar with reduced variation in 2017 and 2018. The broad sense heritabilities for each of the years indicate yield is under strong genetic influence across the set of varieties in each year. This suggests there are definitive underlying mechanisms for the changes in yield between varieties and these can be modelled using genomic prediction.

Linear genomic prediction approaches achieve similar accuracy

For each of the data sets, Table 1 presents the mean relative errors from each of the genomic prediction methods conducted using four random cross-validation splits with 90% training data and 10% validation data. Additionally, to visually gauge the accuracy variation, Fig. 3 presents the relative error across the complete set of lines for each genomic prediction method by year combination for split 1 only. Table 1 and Fig. 3 indicate, across most splits and years, the linear regression approaches LMM, BayesA and BayesB produced very similar results with BayesA and BayesB slightly outperforming LMM. Notably, for the 2016, 2017 and 2018 data sets, Table 1 indicates that the Bayesian regression approaches only produced negligible improvements or no improvement at all over the legacy LMM approach potentially indicating that using a smaller number of lines may impact the ability for these hierarchical models to improve the accuracy of the prediction.

VBS-ML improves relative accuracy over all other approaches

Table 1 and Fig. 3 definitively show that the VBS-ML approach achieves the lowest relative error compared to all other approaches used here. This reduction occurred even though the number of markers used in the prediction component of the VBS-ML network was reduced through feature selection by up to 98%. These relative

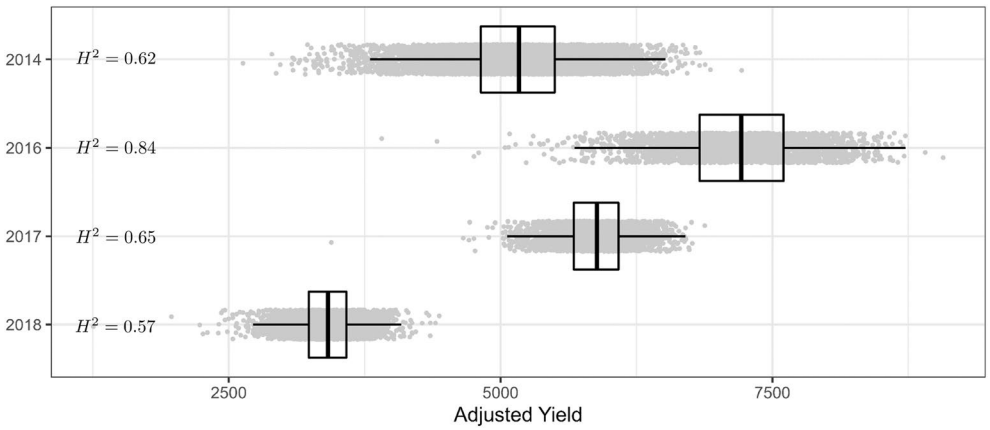

Fig. 2 Distribution of the derived adjusted yield across the set of lines for Roseworthy data sets from 2014, 2016, 2017 and 2018. Generalized broad sense heritabilities H^2 are given on the left hand side of the plot

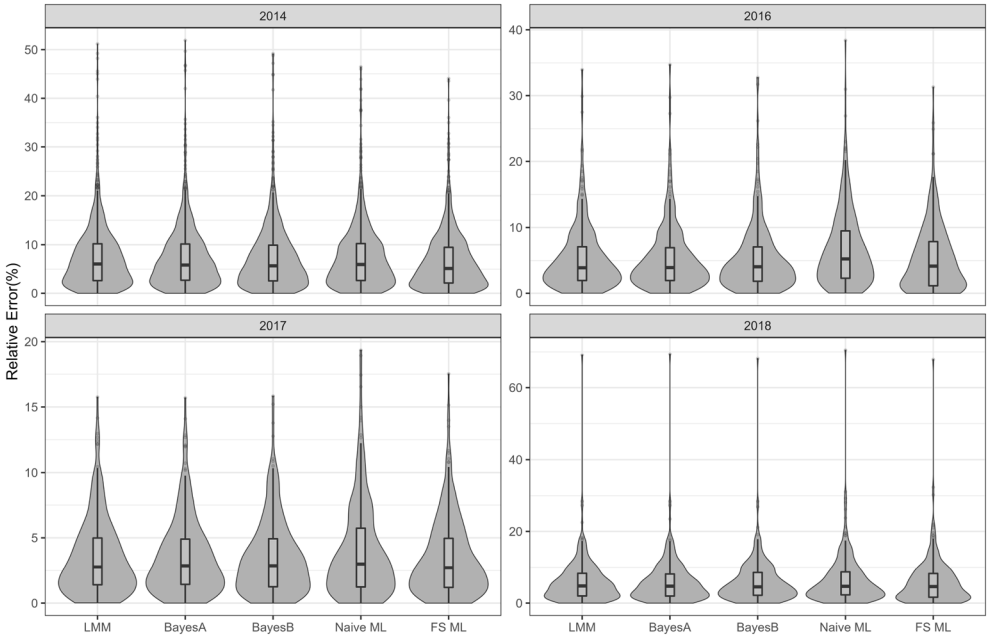

Fig. 3 Relative error prediction accuracy for genomic prediction methods LMM, BayesA, BayesB, Naive-ML and VBS-ML conducted on split 1 of the Roseworthy data sets from 2014, 2016, 2017 and 2018

Table 1 For each of the data sets, the mean relative errors (%) from each of the genomic prediction methods conducted using four random cross-validation splits with 90% training data and 10% validation data. The number of feature selected markers for VBS-ML is given in parentheses

Year	Methods	1	2	3	4	Ave.
2014	LMM	7.38	7.21	7.44	7.31	7.34
	BayesA	7.28	7.16	7.37	7.25	7.27
	BayesB	7.20	7.09	7.36	7.21	7.22
	Naive-ML	7.61	7.56	8.31	7.61	7.77
	VBS-ML (354)	6.49	7.08	7.58	7.04	7.05
2016	LMM	5.36	5.00	5.11	5.23	5.18
	BayesA	5.36	5.02	5.15	5.20	5.18
	BayesB	5.30	5.02	5.07	5.28	5.16
	Naive-ML	6.67	6.28	6.59	6.10	6.41
	VBS-ML (409)	5.20	4.89	4.94	5.04	5.02
2017	LMM	3.53	3.37	3.63	3.45	3.50
	BayesA	3.54	3.38	3.64	3.48	3.51
	BayesB	3.51	3.43	3.65	3.43	3.51
	Naive-ML	3.99	4.16	4.18	3.77	4.03
	VBS-ML (315)	3.48	3.26	3.54	3.26	3.39
2018	LMM	5.94	4.98	5.80	5.94	5.67
	BayesA	5.94	4.99	5.77	5.94	5.66
	BayesB	5.98	5.06	5.63	5.97	5.66
	Naive-ML	6.39	5.70	6.20	6.23	6.13
	VBS-ML (385)	5.89	4.98	5.16	5.86	5.47

error reductions are close to 0.2% for VBS-ML compared to LMM, BayesA, BayesB and between 0.6% and 1.4% for VBS-ML compared to the Naive-ML approach. Additionally, Fig. 3 indicates the VBS-ML tends to have a higher relative error peak closer to zero with thinner tails generated from the larger relative errors. Table 1 also indicates that, on average, for all splits and years, the Naive-ML approach was definitively the poorest performing genomic prediction approach compared to all others. In many cases the relative error increase using the Naive-ML were > 1% for some splits and this equates to a considerable difference on the scale of the response. For example, from split 3 in 2014 a relative error increase of 0.87% using Naive-ML compared to LMM equates to a 44 kg/ha increase in the average differences between the predicted and observed yield values. Figure 3 also indicates that, compared to other methods, the distribution of relative errors for Naive-ML tends to have a smaller peak further away from zero and a fatter tail. This skewness is especially prevalent for the relative errors in 2016 where there were dramatic differences between Naive-ML and other approaches.

VBS-ML slightly improves correlation

Table 2 presents the Pearsons correlations of the predicted vs the observed values of grain yield for each of the genomic prediction methods conducted on each data set from four random cross validation data splits. To complement the table, Fig. 4 presents the correlation of the predicted vs observed grain yield values obtained from all genomic prediction methods conducted using split 1 of each data set. Table 2 indicates that, on average, VBS-ML generated similar correlation to the linear regression methods for the 2016 and 2018 data sets. For the 2014 and 2017 data sets VBS-ML managed to slightly improve over these approaches. This is especially evident in the 2014 correlation plot in Fig. 4 where there appears to be a broader and stronger relationship. The table also indicates, across all data sets, the linear regression approaches achieved a very similar correlation. This similarity is also highlighted in Fig. 4 where the median values and distribution of the predicted values is similar from all three prediction methods. Further to the discussion of relative error, Table 2 and Fig. 4 indicate the Naive-ML genomic prediction method for each year had substantially reduced correlations in 2016, 2017 and 2018

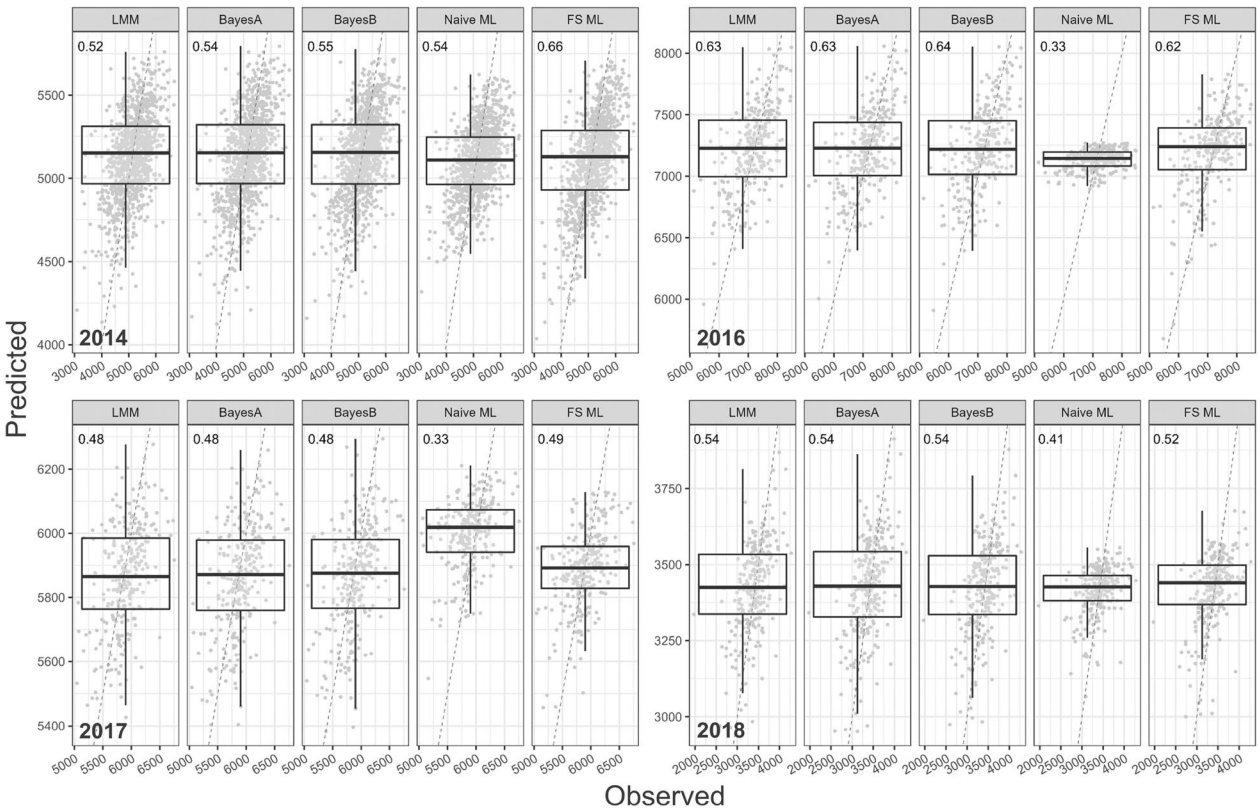

Fig. 4 Relative error prediction accuracy for genomic prediction methods LMM, BayesA, BayesB, Naive-ML and VBS-ML conducted on split 1 of the Roseworthy data sets from 2014, 2016, 2017 and 2018

Table 2 For each of the data sets, the Pearson correlation between the observed and predicted grain yield from each of the genomic prediction methods conducted using four random cross-validation splits with 90% training data and 10% validation data. The number of feature selected markers for VBS-ML is given in parentheses

Year	Methods	1	2	3	4	Ave
2014	LMM	0.52	0.49	0.50	0.47	0.50
	BayesA	0.54	0.50	0.51	0.48	0.51
	BayesB	0.55	0.51	0.51	0.49	0.51
	Naive-ML	0.54	0.41	0.39	0.42	0.44
	VBS-ML (354)	0.66	0.50	0.47	0.52	0.54
2016	LMM	0.63	0.56	0.65	0.57	0.60
	BayesA	0.63	0.56	0.65	0.58	0.61
	BayesB	0.64	0.55	0.65	0.57	0.60
	Naive-ML	0.33	0.24	0.41	0.33	0.33
	VBS-ML (409)	0.62	0.53	0.68	0.56	0.60
2017	LMM	0.48	0.52	0.53	0.52	0.51
	BayesA	0.48	0.52	0.51	0.51	0.51
	BayesB	0.48	0.51	0.51	0.53	0.51
	Naive -ML	0.33	0.38	0.40	0.52	0.41
	VBS-ML (315)	0.49	0.55	0.54	0.60	0.54
2018	LMM	0.54	0.54	0.46	0.48	0.51
	BayesA	0.54	0.54	0.47	0.47	0.50
	BayesB	0.54	0.54	0.49	0.46	0.51
	Naive-ML	0.41	0.25	0.32	0.37	0.34
	VBS-ML (385)	0.52	0.50	0.57	0.44	0.51

and for 2017 it also induced a mean shift in the predicted values.

Feature selected markers show useful predictive properties

The connectivity of the breeding lines between years 2016 and 2018 allows us to further verify the effectiveness of the proposed selection module for genomic prediction. After conducting ML genomic prediction independently in 2016 and 2017, we used the feature selected markers from each of the years to train an MLP to predict adjusted grain yield in future years. Table 3 shows the relative error genomic prediction accuracy of an MLP where feature selected markers from 2016 are used to predict adjusted grain yield in 2017 and 2018 and where feature selected markers from 2017 are used to predict adjusted grain yield in 2018. Comparing this table to the relative error prediction accuracies in Table 1 indicates that using an MLP consisting of feature selected markers from 2016 to predict 2017 adjusted grain yield managed to outperform all genomic prediction methods, except for VBS-ML, conducted on 2017 data. A similar result was observed from the prediction of 2018 adjusted grain yield from feature selected markers in 2016 with improved accuracy from VBS-ML when using only 2018 data. When an MLP, consisting of 2017 feature selected markers, was used to predict 2018 adjusted grain yield data, the relative error slightly improved over the Naive ML approach using 2018 data but was outperformed by all other genomic prediction methods used with the 2018 data.

VBS-ML shows efficiency over LMM for large data sets

Figure 5 presents the computational timings for the analysis methods conducted in the OCIs. The inflated computational time of the LMM in 2014 is due to the ASReml-R 4.1 version used in this research that can only conduct the LMM optimisation on one core of the 16 available in OCI. The large reduction in LMM computational time for 2016 is due to the large reduction in size of the relationship matrix being used in the optimisation. BayesA and BayesB are clearly the most computationally efficient

Table 3 Mean relative error prediction accuracy (%) for genomic prediction of an MLP that used feature selected markers from one year to predict adjusted grain yield in future years

Approach	1	2	3	4	Ave.
2016 $\Longrightarrow$ 2017	3.49	3.27	3.55	3.27	3.40
2016 $\Longrightarrow$ 2018	5.90	5.13	5.17	5.84	5.51
2017 $\Longrightarrow$ 2018	6.32	5.83	5.82	6.02	6.00

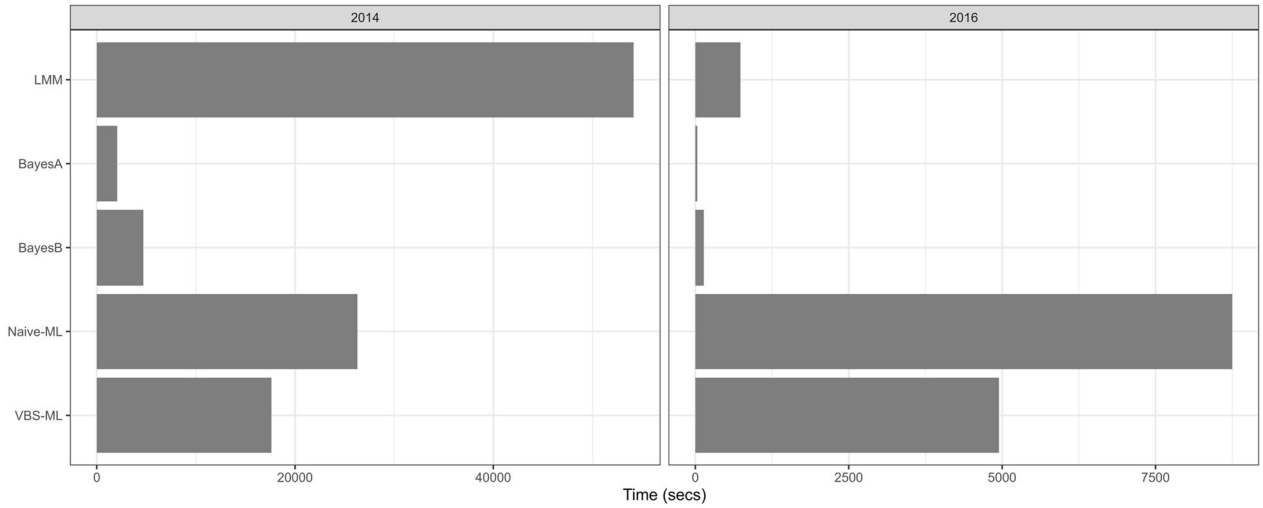

Fig. 5 Computational timings of each of the analysis methods for split 1 of the 2014 and 2016 data. Timings are in seconds of elapsed CPU or GPU time

analysis methods for both data sets. Both approaches utilised all 16 cores available in the OCI suggesting that the MCMC approach implemented in the software is highly parallelized.

For the ML networks, although an OCI consisting of 12 OCPU with 72Gb RAM was available for use in tandem with the NVIDIA Tesla P100, only one CPU with 20% of the available CPU RAM was needed to analyse the 2014 data. For the smaller data set in 2016 the linear prediction analysis approaches computationally outpaced the VBS-ML and Naive-ML analysis methods. For the much larger data set in 2014, the Naive-ML is 2× faster than the LMM approach and the VBS-ML is 3× more efficient. Although highly parallelized through the Tesla multi-core GPU, the ML approaches were not as efficient as multi-core CPU BayesA and BayesB.

Discussion

This research focussed on improving prediction accuracy in large scale genomic prediction problems using an MLP architecture consisting of a feature selection module governed by variational Bayesian sparsity inference. For all data sets analysed in this study the number of genetic markers exceeded the number of samples. Consequently, the incorporation of the feature selection module in the initial stages of the ML architecture provided clear benefits through dramatically reducing the number of important markers and the burden of over-paramaterisation on the network. Further reductions in the over-parameterisation were achieved through the use of an L_1 penalty on the weights of the network across the hidden layers. The VBS-ML approach was shown to improve genomic prediction accuracy over linear based legacy genomic prediction approaches such as LMM, BayesA and BayesB as well as the naive MLP without the feature selection module. In addition, we showed the feature selection of markers obtained from one year could be used to train an MLP for the following years data and produce a competitive accuracy that would usually outperform legacy based approaches trained on the year that was being predicted.

The VBS-ML analysis approach can be considered to be an embedded feature selection approach that ensures redundant SNP markers are removed and the markers with the highest association in each linkage disequilibrium grouping are retained [53, 54]. This suggests this approach would be broadly applicable to other traits beyond grain yield where polygenicity or genetic complexity varies. Additionally, the feature selection properties of the VBS-ML can be considered to provide *explainability* of the prediction through identification of important contributing markers [55].

The VBS inference governing the initial layer of the MLP architecture is akin to the application of variable selection in more traditional regression problems where the objective is optimisation of a non-concave penalized likelihood [56]. Specifically, the resulting log penalty that is derived from the hierarchical modelling of the feature selection weights resembles log penalties derived in various variable selection studies [57, 58]. This penalty is well known for generating sparse solutions when it is applied to coefficients associated with a large set of covariates and a similar result was observed with the feature selection weights associated with the VBS-ML method used in this research. Penalties of this type have the so-called oracle property described by [56] that ensures strong

sparsity without loss of accuracy for non-zero weights or coefficients over more traditional estimation approaches. In this case the penalty is inferred by the distributional hierarchy of the weights but this also suggests other non-hierarchical oracle type penalties, such as the extended penalty class in [58] could be used in the initial layers of the MLP. This is now being explored and is a subject of further research.

In this study we focussed on improving the additive component of genomic prediction using ML. We note that [36] used a comparative Bayesian variable selection type ML architecture that attempted to incorporate epistatic features but had limited success in improving genomic prediction over more standard approaches. [59] used strong regularization of a small number of ML network weights in an approximate Bayesian setting to ensure over-parameterisation of the network was reduced and slightly improved genomic prediction accuracy through estimation of additive components of epistasis. We are now exploring the use of a novel VBS-ML approach to efficiently incorporate and select important non-additive features that will include epistatic as well higher order features that would not usually be modelled through legacy approaches.

Conclusion

The novel VBS-ML method discussed in this research provides a computationally feasible approach for undertaking genomic prediction modelling when the data contains large numbers of lines phenotyped and genotyped across a large set of genetic markers. This approach is of particular relevance to the plant breeding community where there has been a sizeable increase in the germplasm sets being used for genomic analyses [13, 14] and current analysis software limitations are being reached. The high parallelisation of the ML predictive task will require plant breeding organizations to acquire appropriate computational infrastructure as well as analytically integrate the VBS-ML into their plant breeding pipelines. If this is achieved, this research indicates that VBS-ML may be a useful avenue for improved genomic prediction accuracy, allowing plant breeders to accelerate their breeding cycles and continue to increase rates of genetic gain.

Acknowledgements
The authors gratefully acknowledge members of the Biometry Hub in the School of Agriculture, Food and Wine for various minor contributions to the manuscript. We also would like to thank members of the Australian Machine Learning Institute for the initial engagements in this research.

Author contributions
QY, DG and MF conducted the analyses for this research. MF, QY, DG, JW and JT drafted the manuscript. JT, JW, DG, JQS, AN and TC provided overarching supervision as well strategic direction of the research. All authors reviewed and approved the manuscript.

Funding
The authors gratefully acknowledge the Grains Research Development Council (GRDC) for their funding and support of this research through the grant AGT9177537.

Availability of data and materials
The phenotype and genotype datasets for the 2014 Roseworthy trial are publicly available from the Supplmentary material of the article https://link.springer.com/article/10.1007/s00122-017-2975-4 and also available for direct download from https://doi.org/10.25909/23949333.v1. The phenotype and genotype datasets for the 2016, 2017 and 2018 Roseworthy trials are under commercial IP arrangements and not publicly available at this point. The python VBS-ML implementation is publicly available and downloadable from the GitHub repository, https://github.com/mfruzan/VBS-ML-Genomic-Prediction. In addition the R scripts to conduct the ASreml-R and BGLR analyses are also available along with information about the 2014 split identification data that was used to generate the results in this manuscript.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
All authors agree with the submission of this manuscript to Plant Methods.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Australian Institute for Machine Learning, University of Adelaide, Adelaide, Australia. [2]School of Food, Agriculture and Wine, University of Adelaide, Adelaide, Australia. [3]School of Computer Science and Engineering, The University of New South Wales, Sydney, Australia. [4]Australian Grains Technologies, Roseworthy, Australia. [5]School of Computer Science, Northwestern Polytechnical University, Xi'an, China.

Received: 15 September 2022 Accepted: 22 August 2023
Published online: 02 September 2023

References

1. 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.
2. Estaghvirou SBO, Ogutu JO, Schulz-Streeck T, Knaak C, Ouzunova M, Gordillo A, Piepho H-P. Evaluation of approaches for estimating the accuracy of genomic prediction in plant breeding. BMC Genomics. 2013. https://doi.org/10.1186/1471-2164-14-860.
3. Xu S. Estimating polygenic effects using markers of the entire genome. Genetics. 2003;164:789–801. https://doi.org/10.1093/genetics/163.2.789.
4. Zhang YM, Xu S. A penalized maximum likelihood method for estimating epistatic effects of QTL. Heredity. 2005;95:96–104. https://doi.org/10.1038/sj.hdy.6800702.
5. Verbyla AP, Cullis BR, Thompson R. The analysis of QTL by simultaneous use of the of the full linkage map. Theor Appl Genet. 2007;116:95–111. https://doi.org/10.1007/s00122-007-0650-x.
6. Hayes BJ, Bowman PJ, Chamberlain AJ, Goddard ME. Invited review: genomic selection in dairy cattle: progress and challenges. J Dairy Sci. 2009;92(2):433–43. https://doi.org/10.3168/jds.2008-1646.
7. Habier D, Fernando RL, Kizilkaya K, Garrick DJ. Extension of the Bayesian alphabet for genomic selection. BMC Bioinfor. 2011. https://doi.org/10.1186/1471-2105-12-186.
8. Villanueva B, Pong-Wong R, Fernandez J, Toro MA. Benefits from marker-assisted selection under an additive polygenic genetic model1. J Animal Sci. 2005;83(8):1747–52. https://doi.org/10.2527/2005.8381747x.

9. Meuwissen T. Genomic selection : marker assisted selection on a genome wide scale. J Animal Breed Genet. 2007;124(6):321–2. https://doi.org/10.1111/j.1439-0388.2007.00708.x.
10. VanRaden P. Genomic measures of relationship and inbreeding. INTERBULL bull. 2007;37:33.
11. VanRaden PM. Efficient methods to compute genomic predictions. J Dairy Sci. 2008;91(11):4414–23. https://doi.org/10.3168/jds.2007-0980.
12. Verbyla AP, Taylor JD, Verbyla KL. RWGAIM: an efficient high dimensional random whole genome average (QTL) interval mapping approach. Genet Res. 2012;94:291–306. https://doi.org/10.1017/s0016672312000493.
13. Norman A, Taylor J, Tanaka E, Telfer P, Edwards J, Martinant J-P, Kuchel H. Increased genomic prediction accuracy in wheat breeding using a large Australian panel. Theor Appl Genet. 2017;130(7):1–13. https://doi.org/10.1007/s00122-017-2975-4.
14. Norman A, Taylor J, Edwards J, Kuchel H. Optimising genomic selection in wheat: effect of marker density, population size and population structure on prediction accuracy. G3 Genes Genomes Genet. 2018;8(9):2889–99. https://doi.org/10.1534/g3.118.200311.
15. De Coninck A, Kourounis D, Verbosio F, Schenk O, De Baets B, Maenhout S, Fostier J. Towards parallel large-scale genomic prediction by coupling sparse and dense matrix algebra. In: 2015 23rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, 2015. 10.1109/PDP.2015.94
16. Covarrubias-Pazaran G. Genome assisted prediction of quantitative traits using the R package Sommer. PLoS ONE. 2016;11:1–15. https://doi.org/10.1371/journal.pone.0156744.
17. Garrick DJ, Garrick DP, Golden B. An introduction to BOLT software for genetic and genomic evaluations. 2018
18. Azodi CB, Bolger E, Mccarren A, Roantree M, de los Campos G, Shiu S-H. Benchmarking parametric and machine learning models for genomic prediction of complex traits. Genes Genomes Genet. 2019;9(11):3691–702. https://doi.org/10.1534/g3.119.400498.
19. Montesinos-López OA, Montesinos-López A, Pérez-Rodríguez P, Barrón-López JA, Martini JWR, Fajardo-Flores SB, Gaytan-Lugo LS, Santana-Mancilla PC, Crossa J. A review of deep learning applications for genomic selection. BMC Genomics. 2021. https://doi.org/10.1186/s12864-020-07319-x.
20. Meshram V, Patil K, Meshram V, Hanchate D, Ramkteke SD. Machine learning in agriculture domain: a state-of-art survey. Artif Intell Life Sci. 2021;1:100010. https://doi.org/10.1016/j.ailsci.2021.100010.
21. Patterson J, Gibson A. Deep learning: a practitioner's approach, 1st edn. O'Reilly Media, Inc., Sebastopol. 2017.
22. Montesinos-López A, Montesinos-López OA, Gianola D, Crossa J, Hernández-Suárez CM. Multi-environment genomic prediction of plant traits using deep learners with dense architecture. G3 Genes Genomes Genet. 2018;8(12):3813–28. https://doi.org/10.1534/g3.118.200740.
23. González-Camacho JM, Ornella L, Pérez-Rodríguez P, Gianola D, Dreisigacker S, Crossa J. Applications of machine learning methods to genomic selection in breeding wheat for rust resistance. Plant Genome. 2018;11(2):170104. https://doi.org/10.3835/plantgenome2017.11.0104.
24. Montesinos-López OA, Montesinos-López JC, Singh P, Lozano-Ramirez N, Barrón-López A, Montesinos-López A, Crossa J. A multivariate Poisson deep learning model for genomic prediction of count data. Genes Genomes Genet. 2020;10(11):4177–90. https://doi.org/10.1534/g3.120.401631.
25. Sandhu KS, Lozada DN, Zhang Z, Pumphrey MO, Carter AH. Deep learning for predicting complex traits in spring wheat breeding program. Front Plant Sci. 2021;11:2084. https://doi.org/10.3389/fpls.2020.613325.
26. Sandhu K, Patil SS, Pumphrey M, Carter A. Multitrait machine- and deep-learning models for genomic selection using spectral information in a wheat breeding program. Plant Genome. 2021;14(3):1. https://doi.org/10.1002/tpg2.20119.
27. Sandhu K, Aoun M, Morris C, Carter A. Genomic selection for end-use quality and processing traits in soft white winter wheat breeding program with machine and deep learning models. Biology. 2021;10(7):689. https://doi.org/10.3390/biology10070689.
28. Stathakis D. How many hidden layers and nodes? Int J Remote Sens. 2009;30(8):2133–47. https://doi.org/10.1080/01431160802549278.
29. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res. 2014;15(56):1929–58.
30. Labach A, Salehinejad H, Valaee S. Survey of dropout methods for deep neural networks. arXiv. 2019. https://doi.org/10.48550/ARXIV.1904.13310.
31. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I. Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. CoRR. 2012. https://doi.org/10.48550/arXiv.1207.0580.
32. Wan L, Zeiler M, Zhang S, Le Cun Y, Fergus R. 2013. Regularization of neural networks using DropConnect. Proceedings of the 30th International Conference on Machine Learning. Proceedings of Machine Learning Research. 28:1058–1066
33. Kingma DP, Salimans T, Welling M. Variational dropout and the local reparameterization trick. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2. NIPS'15. 2575–2583. MIT Press, Cambridge, MA, USA. 2015. arXiv1506.02557
34. Gal Y, Hron J, Kendall A. Concrete dropout. NIPS. 2017. https://doi.org/10.48550/arXiv.1705.07832.
35. Liu Y, Dong W, Zhang L, Gong D, Shi Q. Variational bayesian dropout with a hierarchical prior. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019;7117–7126. 10.1109/cvpr.2019.00729
36. van Bergen GHH, Duenk P, Albers CA, Bijma P, Calus MPL, Wientjes YCJ, Kappen HJ. Bayesian neural networks with variable selection for prediction of genotypic values. Genet Select Evol. 2020. https://doi.org/10.1186/s12711-020-00544-8.
37. Telfer P, Edwards J, Taylor J, Able JA, Kuchel H. A multi-environment framework to evaluate the adaptation of wheat (Triticum Aestivum) to heat stress. Theor Appl Genet. 2022. https://doi.org/10.1007/s00122-021-04024-5.
38. Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, Botstein D, Altman RB. Missing value estimation methods for DNA microarrays. Bioinformatics. 2001;17(6):520–5. https://doi.org/10.1093/bioinformatics/17.6.520.
39. Rutkoski JE, Poland J, Jannink J-L, Sorrells ME. Imputation of unordered markers and the impact on genomic selection accuracy. G3 Genes Genomes Genet. 2013;3(3):427–39. https://doi.org/10.1534/g3.112.005363.
40. Cullis BR, Smith AB, Coombes NE. On the design of early generation variety trials with correlated data. J Agric Biol Environ Stat Comput. 2006;11:381–93. https://doi.org/10.1198/108571106x154443.
41. Garrick DJ, Taylor JF, Fernando RL. Deregressing estimated breeding values and weighting information for genomic regression analyses. Genet Select Evol. 2009;41(1):55. https://doi.org/10.1186/1297-9686-41-55.
42. Patterson HD, Thompson R. Recovery of interblock information when block sizes are unequal. Biometrika. 1971;58:545–54. https://doi.org/10.1093/biomet/58.3.545.
43. Forni S, Aguilar I, Misztal I. Different genomic relationship matrices for single-step analysis using phenotypic, pedigree and genomic information. Genet Select Evol. 2011;43(1):1. https://doi.org/10.1186/1297-9686-43-1.
44. Henderson CR. Estimation of variance and covariance components. Biometrics. 1953;9:226–52. https://doi.org/10.2307/3001853.
45. Jordan MI, Ghahramani Z, Jaakkola TS, Saul LK. An introduction to Variational methods for graphical models. Mach Learn. 1999;37(2):183–233. https://doi.org/10.1023/A:1007665907178.
46. Butler DG, Cullis BR, Gilmour AR, Gogel BJ, Thompson R, ASReml-R. Reference manual (version 4). Wollongong: University of Wollongong; 2018.
47. R Core Team. Language and environment for statistical computing. Vienna: R Foundation for Statistical Computing; 2021.
48. Perez P, de los Campos G. Genome-wide regression and prediction with the BGLR statistical package. Genetics. 2014;198(2):483–95. https://doi.org/10.1534/genetics.114.164442.
49. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S. Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. 2019.
50. Van Rossum G, Drake FL. Python 3 Reference manual. Scotts Valley: CreateSpace; 2009.
51. Gianola D, Schön C-C. Cross-validation without doing cross-validation in genome-enabled prediction. G3 Genes Genomes Genet. 2016;6(10):3107–28. https://doi.org/10.1534/g3.116.033381.

52. Waldmann P. On the use of the pearson correlation coefficient for model evaluation in genome-wide prediction. Front Genet. 2019. https://doi.org/10.3389/fgene.2019.00899.
53. Tadist K, Najah S, Nikolov NS, Mrabti F, Zahi A. Feature selection methods and genomic big data: a systematic review. J Big Data. 2019. https://doi.org/10.1186/s40537-019-0241-0.
54. Pudjihartono N, Fadason T, Kempa-Liehr AW, O'Sullivan JM. A review of feature selection methods for machine learning-based disease risk prediction. Front Bioinform. 2022. https://doi.org/10.3389/fbinf.2022.927312.
55. Tong H, Nikoloski Z. Machine learning approaches for crop improvement: leveraging phenotypic and genotypic big data. J Plant Physiol. 2021;257:153354. https://doi.org/10.1016/j.jplph.2020.153354.
56. Fan J, Li RZ. Variable selection via nonconcave penalized likelihood and its oracle properties. J Am Statist Assoc. 2001;96:1348–60. https://doi.org/10.1198/016214501753382273.
57. Mazumder R, Friedman JH, Hastie T. Sparsenet: Coordinate descent with nonconvex penalties. J Am Statist Assoc. 2011;106(495):1125–38. https://doi.org/10.1198/jasa.2011.tm09738.
58. Taylor JD, Verbyla AP, Cavanagh C, Newberry M. Variable selection in mixed models using an extended class of penalties. Australia New Zealand J Statist. 2012;54:427–49. https://doi.org/10.1111/j.1467-842X.2012.00687.x.
59. Waldmann P. Approximate Bayesian neural networks in genomic prediction. Genet Select Evol. 2018. https://doi.org/10.1186/s12711-018-0439-1.

Publisher's Note