Cross-Validation Without Doing Cross-Validation in Genome-Enabled Prediction

Daniel Gianola[*,†,‡,§,**,1] **and Chris-Carolin Schön**[§,**]
*Department of Animal Sciences, [†]Department of Dairy Science, and [‡]Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, Wisconsin 53706, and [§]Department of Plant Sciences, Technical University of Munich School of Life Sciences, D-85354 Freising, Germany and **Institute of Advanced Study, Technical University of Munich, D-85748 Garching, Germany
ORCID ID: 0000-0001-8217-2348 (D.G.)

ABSTRACT Cross-validation of methods is an essential component of genome-enabled prediction of complex traits. We develop formulae for computing the predictions that would be obtained when one or several cases are removed in the training process, to become members of testing sets, but by running the model using all observations only once. Prediction methods to which the developments apply include least squares, best linear unbiased prediction (BLUP) of markers, or genomic BLUP, reproducing kernels Hilbert spaces regression with single or multiple kernel matrices, and any member of a suite of linear regression methods known as "Bayesian alphabet." The approach used for Bayesian models is based on importance sampling of posterior draws. Proof of concept is provided by applying the formulae to a wheat data set representing 599 inbred lines genotyped for 1279 markers, and the target trait was grain yield. The data set was used to evaluate predictive mean-squared error, impact of alternative layouts on maximum likelihood estimates of regularization parameters, model complexity, and residual degrees of freedom stemming from various strengths of regularization, as well as two forms of importance sampling. Our results will facilitate carrying out extensive cross-validation without model retraining for most machines employed in genome-assisted prediction of quantitative traits.

KEYWORDS
cross-validation
genomic selection
genomic prediction
genomic BLUP
reproducing kernels
GenPred
Shared Data Resources

Whole-genome-enabled prediction, introduced by Meuwissen *et al.* (2001), has received much attention in animal and plant breeding (*e.g.*, Van Raden 2008; Crossa *et al.* 2010; Lehermeier *et al.* 2013), primarily because it can deliver reasonably accurate predictions of the genetic worth of candidate animals or plants at an earlier time in the context of artificial selection. It has also been suggested for prediction of complex traits in human medicine (*e.g.*, de los Campos *et al.* 2011; Makowsky *et al.* 2011; Vázquez *et al.* 2012; López de Maturana *et al.* 2014; Spiliopoulou *et al.* 2015)

An important contribution of Utz *et al.* (2000) and of Meuwissen *et al.* (2001) was implanting cross-validation (CV) in plant and animal breeding as a mechanism for comparing prediction models, typically multiple linear regressions on molecular markers. In retrospect, it is perplexing that the progression of genetic prediction models, *e.g.*, from simple "sire" or "family" models in the late 1960s (Henderson *et al.* 1959) to complex multivariate and longitudinal specifications (Mrode 2014), proceeded without CV, as noted by Gianola and Rosa (2015). An explanation, at least in animal breeding, is the explosion of best linear unbiased prediction, BLUP (Henderson 1973, 1984). The power and flexibility of the linear mixed model led to the (incorrect) belief that a bigger model is necessarily better, simply because of extra explanatory power from an increasing degree of complexity. However, a growing focus on predictive inference and on CV has altered such perception. A simple prediction model may produce more stable, and even better, results than a complex hierarchical model (Takezawa 2006; Wimmer *et al.* 2013), and the choice can be made via CV. Today, CV is a *sine qua non* part of studies comparing genome-assisted prediction methods.

Most often, CV consists of dividing a data set with n cases (each including a phenotypic measurement and a vector of genomic covariables) into a number of folds (K) of approximately equal size. Data in $K - 1$ folds are used for model training, and to effect predictions of phenotypes in the testing fold, given the realized values of the genomic covariables. The prediction exercise is repeated for each fold, and the

doi: 10.1534/g3.116.033381
Manuscript received May 23, 2016; accepted for publication July 28, 2016; published Early Online August 3, 2016.

[1]Corresponding author: Department of Animal Sciences, University of Wisconsin-Madison, 1675 Observatory Drive, Madison, WI 53706. E-mail: gianola@ansci.wisc.edu

overall results are combined; this is known as a K-fold CV (Hastie *et al.* 2009). A loss function, such as mean-squared error (MSE) or predictive correlation is computed to gauge the various predictive machines compared. However, the process must be repeated a number of times, with folds reconstructed at random (whenever possible) to obtain measures of CV uncertainty (*e.g.*, Okut *et al.* 2011; Tusell *et al.* 2014). CV is computationally taxing, especially when Bayesian prediction models with a massive number of genomic covariates and implemented via Markov chain Monte Carlo (MCMC) are involved in the comparison.

Stylized formulae (*e.g.*, Daetwyler *et al.* 2008) suggest that the expected predictive correlation ("accuracy") in genome-enabled prediction is proportional to training sample size (n). On intuitive grounds, more genetic variability ought to be spanned as a training sample grows, unless additional cases bring redundant information. With larger n, it is more likely that genomic patterns appearing in a testing set are encountered in model training. Although the formulae do not always fit real data well (Chesnais *et al.* 2016), it has been observed that a larger n tends to be associated with larger predictive correlations (Utz *et al.* 2000; Erbe *et al.* 2010).

Arguably, there is no better expectation than what is provided by a CV conducted under environmental circumstances similar to those under which the prediction machine is going to be applied. When n is small, the largest possible training set sample size is attained in a leave-one-out (LOO) CV, *e.g.*, Ober *et al.* (2015) with about 200 lines of *Drosophila melanogaster*. In LOO CV, $n-1$ cases are used for model training, to then predict the single out-of-sample case. Model training involves n implementations, each consisting of a training sample of size $n-1$ and a testing set of size 1.

It is not widely recognized that it is feasible to obtain CV results by running the model only once, which is well known for least-squares regression (*e.g.*, Seber and Lee 2003). Here, we show that this idea extends to other prediction machines, such as ridge regression (Hoerl and Kennard 1970), genome-based best linear unbiased prediction (GBLUP; Van Raden 2008), and reproducing kernel Hilbert spaces regression (RKHS; Gianola *et al.* 2006; Gianola and van Kaam 2008). It is also shown that the concept can be applied in a MCMC context to any Bayesian hierarchical model, *e.g.*, members of the "Bayesian alphabet" (Meuwissen *et al.* 2001; Gianola *et al.* 2009; Gianola 2013). This manuscript reviews available results for least-squares based CV, and shows how CV without actually doing CV can be conducted for ridge regression, BLUP of marker effects, GBLUP, and RKHS for any given kernel matrix. It is described how importance sampling can be used to produce Bayesian CV by running MCMC only once, which has great advantage in view of the intensiveness of MCMC computations. Illustrations are given by using a well characterized data set containing wheat grain yield as phenotype and 1279 binary markers as regressors, and the paper concludes with a *Discussion*. Most technical results are presented in a series of *Appendices*, to facilitate reading the main body of the manuscript.

CROSS-VALIDATION WITH ORDINARY LEAST-SQUARES

Setting

A linear model used for regressing phenotypes on additive codes at biallelic marker loci ($-1, 0, 1$ for *aa*, *Aa* and *AA*, respectively), such as in a genome-wide association study, is

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}+\mathbf{e}. \tag{1}$$

Here, $\mathbf{y}$ is the $n\times 1$ vector of phenotypic measurements, $\mathbf{X}=\{x_{ij}\}$ is the $n\times p$ marker matrix, and x_{ij} is the number of copies of a reference allele at locus j observed in individual i; $\boldsymbol{\beta}$ is a $p\times 1$ vector of fixed regressions on marker codes, known as allelic substitution effects. Phenotypes and markers are often centered; if an intercept is fitted, the model is expanded by adding β_0 as an effect common to all phenotypes. The residual vector is assumed to follow the distribution $\mathbf{e}\sim N(\mathbf{0},\mathbf{I}\sigma_e^2)$, where $\mathbf{I}$ is an $n\times n$ identity matrix, and σ_e^2 is a variance parameter common to all residuals.

The basic principles set here carry to the other prediction methods discussed in this paper. In this section, we assume $rank(\mathbf{X})=p<n$ so that ordinary least-squares (OLS) or maximum likelihood under the normality assumption above can be used. The OLS estimator of $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}}=(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$, and the fitted residual for datum i is $\hat{e}_i=y_i-\mathbf{x}_i'\hat{\boldsymbol{\beta}}$, where $\mathbf{x}_i'$ is the i^{th} row of $\mathbf{X}$. Assuming the model holds, $E(\hat{\boldsymbol{\beta}})=\boldsymbol{\beta}$, so the estimator is unbiased. A review of the pertinent principles is given in *Appendix A*, from which we extract results.

It is shown in *Appendix A* that the uncertainty of a prediction, as measured by variance, increases with p (model complexity), and decreases with the size of the testing set, n_{test}. Two crucial matters in genome-enabled prediction must be underlined. First, if the model is unnecessarily complex, prediction accuracy (in the MSE sense) degrades unless the increase in variance is compensated by a reduction in prediction bias. Second, if the training set is made large at the expense of the size of the testing set, prediction mean squared error will be larger than otherwise. The formulae of Daetwyler *et al.* (2008) suggest that expected prediction accuracy, as measured by predictive correlation (not necessarily a good metric; González-Recio *et al.* 2014), increases with n. However, the variability of the predictions would increase, as found by Erbe *et al.* (2010) in an empirical study of Holstein progeny tests with alternative CV layouts. Should one aim at a higher expected predictive correlation or at a more stable set of predictions at the expense of the former? This question does not have a simple answer.

Leave-one-out (LOO) cross-validation

LOO is often used when n is small and there is concern about the limited size of the training folds. Let $\mathbf{X}_{[-i]}$ be $\mathbf{X}$ with its i^{th} row ($\mathbf{x}_i'$) removed, so that its order is $(n-1)\times p$. Since

$$\mathbf{X}'\mathbf{X}=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i',$$

$$\mathbf{X}_{[-i]}'\mathbf{X}_{[-i]}=\mathbf{X}'\mathbf{X}-\mathbf{x}_i\mathbf{x}_i';\quad i=1,2,\ldots,n, \tag{2}$$

are $p\times p$ matrices. Likewise, if $\mathbf{y}_{[-i]}$ is $\mathbf{y}$ with its i^{th} element removed, the OLS right-hand sides in LOO are

$$\mathbf{X}_{[-i]}'\mathbf{y}_{[-i]}=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{y}-\mathbf{x}_iy_i=\mathbf{X}'\mathbf{y}-\mathbf{x}_iy_i \tag{3}$$

Making use of (81) in *Appendix B*, the least-squares estimator of $\boldsymbol{\beta}$ formed with the i^{th} observation deleted from the model is expressible as

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{[-i]}&=\left(\mathbf{X}_{[-i]}'\mathbf{X}_{[-i]}\right)^{-1}\mathbf{X}_{[-i]}'\mathbf{y}_{[-i]}\\&=\left[\left(\mathbf{X}'\mathbf{X}\right)^{-1}+\frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}_i'\left(\mathbf{X}'\mathbf{X}\right)^{-1}}{1-\mathbf{x}_i'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i}\right]\left(\mathbf{X}'\mathbf{y}-\mathbf{x}_iy_i\right).\end{aligned} \tag{4}$$

Employing *Appendix C*, the estimator can be written in the form

$$\hat{\boldsymbol{\beta}}_{[-i]}=\hat{\boldsymbol{\beta}}-\frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\hat{e}_i}{1-h_{ii}}, \tag{5}$$

where $h_{ii} = \mathbf{x}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i$ and $\hat{e}_i = y_i - \mathbf{x}_i'\hat{\boldsymbol{\beta}}$ is the fitted residual using all n observations in the analysis; the fitted LOO residual is

$$y_i - \mathbf{x}_i'\hat{\beta}_{[-i]} = \frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\beta}}}{1 - h_{ii}}. \tag{6}$$

Hence, the LOO estimator and prediction error can be computed directly from the analysis carried out with the entire data set: no need for n implementations.

Making use of (6), the realized LOO CV mean squared error of prediction is

$$\text{PMSE}(1) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\beta}}}{1-h_{ii}}\right)^2, \tag{7}$$

and the expected mean squared error of prediction is given by

$$\begin{aligned} E_{\mathbf{y}|\mathbf{X}}[\text{PMSE}(1)] &= \frac{1}{n}E\left[\sum_{i=1}^{n}\left(\frac{y_i - \mathbf{x}_i'\hat{\beta}}{1-h_{ii}}\right)^2\right] \\ &= \frac{1}{n}\left\{\delta'\mathbf{D}\delta + tr[\mathbf{D}Var(\mathbf{y} - \mathbf{X}\hat{\beta})]\right\} \end{aligned} \tag{8}$$

where $\mathbf{D} = \{(1-h_{ii})^{-2}\}$ is an $n \times n$ diagonal matrix. As shown in *Appendix A* the LOO expected PMSE gives an upper bound for the expected squared error in least-squares based CV. The extent of overstatement of the error depends on the marker matrix $\mathbf{X}$ (via the $h's$) and on the prediction biases δ_i. Hence, LOO CV represents a conservative approach, with the larger variance of the prediction resulting from the smallest possible testing set size ($n_{\text{test}} = 1$) If the prediction is unbiased, the $\delta -$ terms vanish, and it is clear that observations with $h -$ values closer to 1 contribute more to squared prediction error than those with smaller values, as the model is close to overfitting the former type of observations.

Leave-d-out cross-validation

The preceding analysis suggests that reallocation of observations from training into testing sets is expected to reduce PMSE relative to the LOO scheme. Most prediction-oriented analyses use $K -$ fold CV, where K is chosen arbitrarily (*e.g.*, $K = 2, 5, 10$) as mentioned earlier; the decision of the number of folds is usually guided by the number of samples available. Here, we address this type of scheme generically by removing d out of the n observations available for training, and declaring the former as members of the testing set. Let $\mathbf{X}_{[-d]}$ be $\mathbf{X}$ with d of its rows removed, and $\mathbf{y}_{[-d]}$ be the data vector without the d corresponding data points. As shown in *Appendix D*,

$$\hat{\boldsymbol{\beta}}_{[-d]} = \hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1}\hat{\mathbf{e}}_{[d]}, \tag{9}$$

where $\mathbf{H}_d = \mathbf{X}_{[d]}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'_{[d]}$; note that $(\mathbf{I}-\mathbf{H}_d)^{-1}$ does not always exist but can be replaced by a generalized inverse, and $\hat{\boldsymbol{\beta}}_{[-d]}$ will be invariant to the latter if $p < n - d$. Predictions are invariant with respect to the generalized inverse used. The similarity with (5) is clear: $(\mathbf{I}-\mathbf{H}_d)^{-1}$ appears in lieu of $1 - h_{ii}$, $\mathbf{X}'_{[d]}$ of order $p \times d$ contains (in columns) the rows of $\mathbf{X}$ being removed, and $\hat{\mathbf{e}}_{[d]} = \mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}$ are the residuals corresponding to the left out $d -$ *plet* obtained when fitting the model to the entire data set.

The error of prediction of the d phenotypes entering into the testing set is

$$\begin{aligned} \mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}_{[-d]} = \mathbf{y}_{[d]} - \mathbf{X}_{[d]}\Big\{&\hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1} \\ &\times\left(\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}\right)\Big\} = (\mathbf{I}-\mathbf{H}_d)^{-1}\hat{\mathbf{e}}_{[d]}. \end{aligned} \tag{10}$$

The mean-squared error of prediction of the d observations left out (testing set) becomes

$$\begin{aligned} \text{PMSE}(d) &= \frac{1}{d}\left(\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}_{[-d]}\right)'\left(\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}_{[-d]}\right) \\ &= \frac{1}{d}\hat{\mathbf{e}}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-2}\hat{\mathbf{e}}_{[d]}. \end{aligned} \tag{11}$$

The prediction bias obtained by averaging over all possible data sets is

$$E_{\mathbf{y}_{[d]},\mathbf{y}|,\mathbf{X},\mathbf{X}_{[d]}}\left(\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}\right) = \mathbf{I}\boldsymbol{\mu}_d - \mathbf{X}_{[d]}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\boldsymbol{\mu} = \boldsymbol{\delta}_d, \tag{12}$$

where $\boldsymbol{\mu}_d$ is a $d \times 1$ vector of true means of the distribution of observations in the testing sets. After algebra,

$$Var_{\mathbf{y}_{[d]},\mathbf{y}|,\mathbf{X},\mathbf{X}_{[d]}}\left(\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}\right) = (\mathbf{I} - \mathbf{H}_d)\sigma_e^2. \tag{13}$$

and

$$E_{\mathbf{y}_{[d]},\mathbf{y}|,\mathbf{X},\mathbf{X}_{[d]}}[\text{PMSE}(d)] = \frac{1}{d}\left\{\delta'_d(\mathbf{I}-\mathbf{H}_d)^{-2}\boldsymbol{\delta}_d + tr\left[(\mathbf{I}-\mathbf{H}_d)^{-1}\right]\sigma_e^2\right\}. \tag{14}$$

Observe that the term in brackets is a matrix counterpart of (76) in *Appendix A*, with $\mathbf{H}_d$ playing the role of h_{ii} in the expression. The two terms in the equation above represent the contributions and bias and (co) variance to expected squared prediction error.

The next section illustrates how the preceding logic carries to regression models with shrinkage of estimated allelic substitution effects ($\hat{\boldsymbol{\beta}}$).

CROSS-VALIDATION WITH SHRINKAGE OF REGRESSION COEFFICIENTS

BLUP of markers (ridge regression)

Assume again that phenotypes and markers are centered. Marker effects $\boldsymbol{\beta}$ are now treated as random variables and assigned the normal $N(\mathbf{0}, \mathbf{I}\sigma_\beta^2)$ distribution, where σ_β^2 is a variance component. The BLUP of $\boldsymbol{\beta}$ (Henderson 1975) is given by

$$\boldsymbol{\beta}^r = \left(\mathbf{X}'\mathbf{X} + \mathbf{I}\lambda\right)^{-1}\mathbf{X}'\mathbf{y}, \tag{15}$$

where $\lambda = \frac{\sigma_e^2}{\sigma_\beta^2}$ is a shrinkage factor taken as known. BLUP has the same mathematical form as the ridge regression estimator (Hoerl and Kennard 1970), developed mainly for tempering problems caused by colinearity among columns of $\mathbf{X}$ in regression models where $p < n$, and with all regression coefficients likelihood-identified. The solution vector $\boldsymbol{\beta}^r$ can also be assigned a Bayesian interpretation as a posterior expectation in a linear model with Gaussian residuals, and $N(\mathbf{0}, \mathbf{I}\sigma_\beta^2)$ used as prior distribution, with variance components known (Dempfle 1977; Gianola and Fernando 1986). A fourth view of $\boldsymbol{\beta}^r$ is as a penalized maximum likelihood estimator under an L_2 penalty (Hastie *et al.* 2009). Irrespective of its interpretation, $\boldsymbol{\beta}^r$ provides a "point statistic" of $\boldsymbol{\beta}$ for the $n < p$ situation. In BLUP, or in Bayesian inference, it is not a requirement that the regression coefficients are likelihood identified. There is one formula with four interpretations (Robinson 1991).

Given a testing set with marker genotype matrix $\mathbf{X}_{test}$, the point prediction of yet to be observed phenotypes is $\mathbf{X}_{test}\boldsymbol{\beta}^r$. We consider LOO CV because subsequent developments assume that removal of a single case has a mild effect on σ_e^2 and σ_β^2. This assumption is

reasonable for animal and plant breeding data sets where n is large, so removing a single observation should have a minor impact on, say, maximum likelihood estimates of variance components. If λ is kept constant, it is shown in *Appendix E* that

$$\boldsymbol{\beta}^r_{[-i]} = \boldsymbol{\beta}^r - \frac{\mathbf{C}^{-1}\mathbf{x}_i\hat{e}^r_i}{1-h^r_{ii}}, \tag{16}$$

where $\mathbf{C} = \mathbf{X}'\mathbf{X} + \mathbf{I}\lambda$, $h^r_{ii} = \mathbf{x}'_i\mathbf{C}^{-1}\mathbf{x}_i$ and $\hat{e}^r_i = y_i - \mathbf{x}'_i\boldsymbol{\beta}^r$ is the residual from ridge regression BLUP applied to the entire sample. A similar expression for leave-d-out cross-validation using the same set of variance components is also in *Appendix E*. If d/n is smaller and n is reasonably large, the error resulting from using variance components estimated from the entire data set should be small.

The error of predicting phenotype i is now $y_i - \mathbf{x}'_i\boldsymbol{\beta}^r_{[-i]}$ and is expressible as

$$y_i - \mathbf{x}'_i\boldsymbol{\beta}^r_{[-i]} = \frac{\left(y_i - \mathbf{x}'_i\boldsymbol{\beta}^r\right)}{1-h^r_{ii}}, \tag{17}$$

similar to that LOO OLS. Letting $\boldsymbol{\delta}_r = \{E(y_i - \mathbf{x}'_i\boldsymbol{\beta}^r)\}$ be a vector of prediction biases, $\mathbf{D}_r = \left\{(1-h^r_{ii})^{-2}\right\}$ and $\mathbf{M}_r = \mathbf{I} - \mathbf{X}\mathbf{C}^{-1}\mathbf{X}'$, the expected prediction MSE is

$$E_{\mathbf{y}|\mathbf{X},\sigma^2_e,\sigma^2_\beta}[\text{PMSE}_r(1)] = \frac{1}{n}\delta'_r\mathbf{D}_r\delta_r + \frac{\sigma^2_e}{n}tr\left[\mathbf{D}_r\mathbf{M}_r\left(\mathbf{X}\mathbf{X}'\lambda^{-1} + \mathbf{I}\right)\mathbf{M}_r\right]. \tag{18}$$

The first term is the average squared prediction bias, and the second is the prediction error variance. As $\sigma^2_\beta \to 0$, (18) tends to the least-squares $PMSE(1)$.

Genomic BLUP

Once marker effects are estimated as $\boldsymbol{\beta}^r$, a representation of genomic BLUP (GBLUP) for n individuals is the $n\times 1$ vector $\mathbf{g} = \mathbf{X}\boldsymbol{\beta}^r$ with its i^{th} element being $\hat{\mathbf{g}}_i = \mathbf{x}'_i\boldsymbol{\beta}^r$. In GBLUP, "genomic relationship matrices" are taken as proportional to $\mathbf{X}\mathbf{X}'$ (where $\mathbf{X}$ often has centered columns); various genomic relationship matrices are in, *e.g.*, Van Raden (2008), Astle and Balding (2009) and Rincent *et al.* (2014). Using (16) LOO GBLUP (*i.e.*, excluding case i from the training sample) is

$$\hat{\mathbf{g}}_{[-i]} = \mathbf{X}_{[-i]}\boldsymbol{\beta}^r_{[-i]} = \mathbf{X}_{[-i]}\boldsymbol{\beta}^r - \frac{\mathbf{X}_{[-i]}\mathbf{C}^{-1}\mathbf{x}_i\left(y_i - \mathbf{x}'_i\boldsymbol{\beta}^r\right)}{1-h_{ii}}. \tag{19}$$

The formula above requires finding $\boldsymbol{\beta}^r$, given λ; the procedure entails solving p equations on p unknowns and finding the inverse of $\mathbf{C}$ is impossible or extremely taxing when p is large. A simpler alternative based on the well-known equivalence between BLUP of marker effects and of additive genotypic value is used here.

If $\mathbf{g} = \mathbf{X}\boldsymbol{\beta}$ is a vector of marked additive genetic values and $\boldsymbol{\beta} \sim N(\mathbf{0}, \mathbf{I}\sigma^2_\beta)$, then $\mathbf{g} \sim N(\mathbf{0}, \mathbf{X}\mathbf{X}'\sigma^2_\beta)$. Many genomic relationship matrices are expressible as $\mathbf{G} = \frac{\mathbf{X}\mathbf{X}'}{c}$ for some constant c, so that $\mathbf{g} \sim N(\mathbf{0}, \mathbf{G}\sigma^2_g)$ and $\sigma^2_g = c\sigma^2_\beta$ is called "genomic variance" or "marked additive genetic variance" if $\mathbf{X}$ encodes additive effects; clearly, there is no loss of generality if $c = 1$ is used, thus preserving the λ employed for BLUP of marker effects. The model for the "signal" $\mathbf{g}$ becomes

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e} = \mathbf{g} + \mathbf{e}. \tag{20}$$

Letting $\mathbf{C} = \mathbf{I} + \mathbf{G}^{-1}\lambda$, then $BLUP(\mathbf{g}) = \hat{\mathbf{g}} = \{\hat{g}_i\} = \mathbf{C}^{-1}\mathbf{y}$ is GBLUP using all data points. *Appendix F* shows how LOO GBLUP and d-out GBLUP can be calculated indirectly from elements or blocks of $\mathbf{C}$, and elements of $\mathbf{y}$.

RKHS regression

In RKHS regression (Gianola *et al.* 2006; Gianola and van Kaam 2008), input variables, *e.g.*, marker codes, can be transformed nonlinearly, potentially capturing both additive and nonadditive genetic effects (Gianola *et al.* 2014a, 2014b), as further expounded by Jiang and Reif (2015) and Martini *et al.* (2016). When a pedigree or a genomic relationship matrix is used as kernel, RKHS yields pedigree-BLUP and GBLUP, respectively, as special cases (Gianola and de los Campos 2008; de los Campos *et al.* 2009, 2010).

The standard RKHS model is

$$\mathbf{y} = \mathbf{g} + \mathbf{e} = \mathbf{K}\boldsymbol{\alpha} + \mathbf{e}, \tag{21}$$

with $\mathbf{g} = \mathbf{K}\boldsymbol{\alpha}$ (and therefore $g_i = \mathbf{k}'_i\boldsymbol{\alpha}$); $\mathbf{K}$ is an $n\times n$ positive (semi)-definite symmetric matrix so that $\mathbf{K} = \mathbf{K}'$; $\boldsymbol{\alpha} = \mathbf{K}^{-1}\mathbf{g}$ when the inverse is unique, and $\boldsymbol{\alpha} \sim N(\mathbf{0}, \mathbf{K}^{-1}\sigma^2_\alpha)$. BLUP($\boldsymbol{\alpha}$) can be obtained by solving the system

$$\left(\mathbf{K}^2 + \mathbf{K}\frac{\sigma^2_e}{\sigma^2_\alpha}\right)\hat{\boldsymbol{\alpha}} = \mathbf{K}\mathbf{y}, \tag{22}$$

with solution (since $\mathbf{K}'\mathbf{K} = \mathbf{K}^2$ and $\mathbf{K}$ is invertible)

$$\hat{\boldsymbol{\alpha}} = \left(\mathbf{K} + \mathbf{I}\frac{\sigma^2_e}{\sigma^2_\alpha}\right)^{-1}\mathbf{y}. \tag{23}$$

The BLUP of $\mathbf{g}$ under a RKHS model is

$$\begin{aligned}\text{BLUP}_K(\mathbf{g}) &= \text{BLUP}_K(\mathbf{K}\boldsymbol{\alpha})\\ &= \mathbf{K}\left(\mathbf{K} + \mathbf{I}\frac{\sigma^2_e}{\sigma^2_\alpha}\right)^{-1}\mathbf{y} = \left(\mathbf{I} + \mathbf{K}^{-1}\lambda_K\right)^{-1}\mathbf{y},\end{aligned} \tag{24}$$

where K stands for "kernel," and $\lambda_K = \frac{\sigma^2_e}{\sigma^2_\alpha}$. Putting $\mathbf{C}^{-1}_K = \left(\mathbf{I} + \mathbf{K}^{-1}\lambda_{\text{RKHS}}\right)^{-1}$, the RKHS solution $\hat{\mathbf{g}}_K = \mathbf{C}^{-1}_K\mathbf{y}$ has the same form as $\text{BLUP}(\mathbf{g}) = \hat{\mathbf{g}} = \mathbf{C}^{-1}\mathbf{y}$, as given in the preceding section. Using *Appendix F*, it follows that

$$\tilde{\mathbf{g}}_{d,K} = \left(\mathbf{I} - \mathbf{C}^{dd}_K\right)^{-1}\left(\hat{\mathbf{g}}_{d,K} - \mathbf{C}^{dd}_K\mathbf{y}_d\right), \tag{25}$$

$$\tilde{\boldsymbol{\epsilon}}_{d,K} = \mathbf{y}_d - \tilde{\mathbf{g}}_{d,K} = \left(\mathbf{I} - \mathbf{C}^{dd}_K\right)^{-1}\left(\mathbf{y}_d - \hat{\mathbf{g}}_{d,K}\right), \tag{26}$$

and

$$\text{PMSE}_K(d) = \frac{\tilde{\boldsymbol{\epsilon}}'_{d,K}\tilde{\boldsymbol{\epsilon}}_{d,K}}{d}. \tag{27}$$

The previous expressions reduce to the LOO CV situation by setting $d = 1$.

BAYESIAN CROSS-VALIDATION

Setting

Many Bayesian linear regression on markers models have been proposed for genome-assisted prediction of quantitative traits (*e.g.*, Meuwissen *et al.* 2001; Heslot *et al.* 2012; de los Campos *et al.* 2013; Gianola 2013). All such models pose the same specification for the Bayesian sampling model (a linear regression), but differ in the prior distribution assigned

to allelic substitution effects. Implementation is often via MCMC, where computations are intensive even in the absence of CV; shortcuts and approximations are not without pitfalls. Is it possible to do CV by running an MCMC implementation only once? What follows applies both to LOO and $d-$ out CV situations as well as to any member of the Bayesian alphabet (Gianola *et al.* 2009; Gianola 2013)

Suppose some Bayesian model has been run with MCMC, leading to S samples collected from a distribution with posterior density $p(\boldsymbol{\theta}|\mathbf{y},H)$; here, $\boldsymbol{\theta}$ are all unknowns to be inferred and H denotes hyper-parameters arrived at in a typically subjective manner, *e.g.*, arbitrary variances in a four-component mixture distribution assigned to substitution effects (MacLeod *et al.* 2014). In CV, the data set is partitioned into $\mathbf{y}=(\mathbf{y}_{test},\mathbf{y}_{train})$, training and testing sets are chosen according to the problem in question, and Bayesian learning is based on the posterior distribution $[\boldsymbol{\theta}|\mathbf{y}_{train},H]$. Predictions are derived from the predictive distribution of the testing set data

$$p(\mathbf{y}_{test}|\mathbf{y}_{train},H)=\int p(\mathbf{y}_{test}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y}_{train},H)d\boldsymbol{\theta}; \tag{28}$$

the preceding assumes that $\mathbf{y}_{test}$ is independent of $\mathbf{y}_{train}$ given $\boldsymbol{\theta}$, a standard assumption in genome-enabled prediction. The point predictor chosen most often is the expected value of the predictive distribution

$$\begin{aligned}E(\mathbf{y}_{test}|\mathbf{y}_{train},H)&=\int\int \mathbf{y}_{test}p(\mathbf{y}_{test}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y}_{train},H)d\boldsymbol{\theta}d\mathbf{y}_{test}\\&=\int\int \mathbf{y}_{test}p(\mathbf{y}_{test},\boldsymbol{\theta}|\mathbf{y}_{train},H)d\boldsymbol{\theta}d\mathbf{y}_{test} \qquad (29)\\&=\int E(\mathbf{y}_{test}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y}_{train},H)d\boldsymbol{\theta} \qquad (30)\end{aligned}$$

In the context of sampling, representation (29) implies that one can explore the "augmented" distribution $[\mathbf{y}_{test},\boldsymbol{\theta}|\mathbf{y}_{train},H]$, and estimate $E(\mathbf{y}_{test}|\mathbf{y}_{train},H)$ by ergodic averaging of $\mathbf{y}_{test}$ samples. Representation (30) uses Rao-Blackwellization: if $E(\mathbf{y}_{test}|\boldsymbol{\theta})$ can be written in closed form, as is the case for regression models $(\mathbf{X}_{test}\boldsymbol{\beta})$, the Monte Carlo variance of an estimate of $E(\mathbf{y}_{test}|\mathbf{y}_{train},H)$ based on (30) is less than, or equal to, that of an estimate obtained with (29).

We describe Bayesian LOO CV, but extension to a testing set of size d is straightforward. In LOO, the data set is partitioned as $\mathbf{y}=(y_i,\mathbf{y}_{-i})$, $i=1,2,\ldots,n$, where y_i is the predictand and $\mathbf{y}_{-i}$ is the vector containing all other phenotypes in the data set. A brute force process involves running the Bayesian model n times, producing the posterior distributions $[\boldsymbol{\theta}|\mathbf{y}_{-i},H]$; $i=1,2,\ldots,n$. Since LOO CV is computationally formidable in an MCMC context, procedures based on drawing samples from $[\boldsymbol{\theta}|\mathbf{y},H]$ and converting these into realizations from $[\boldsymbol{\theta}|\mathbf{y}_{-i},H]$ can be useful (*e.g.*, Gelfand *et al.* 1992; Gelfand 1996; Vehtari and Lampinen 2002). Use of importance sampling, and of sampling importance resampling (SIR), algorithms for this purpose is discussed next. Cantet *et al.* (1992) and Matos *et al.* (1993) present early applications of importance sampling to animal breeding.

Importance sampling

We seek to estimate the mean of the predictive distribution of the left-out data point $E(y_i|\mathbf{y}_{-i},H)$. Since $e_i\sim N(0,\sigma_e^2)$ is independent of $\mathbf{y}_{-i}$, one has

$$E(y_i|\mathbf{y}_{-i},H)=\mathbf{x}_i'E(\boldsymbol{\beta}|\mathbf{y}_{-i},H). \tag{31}$$

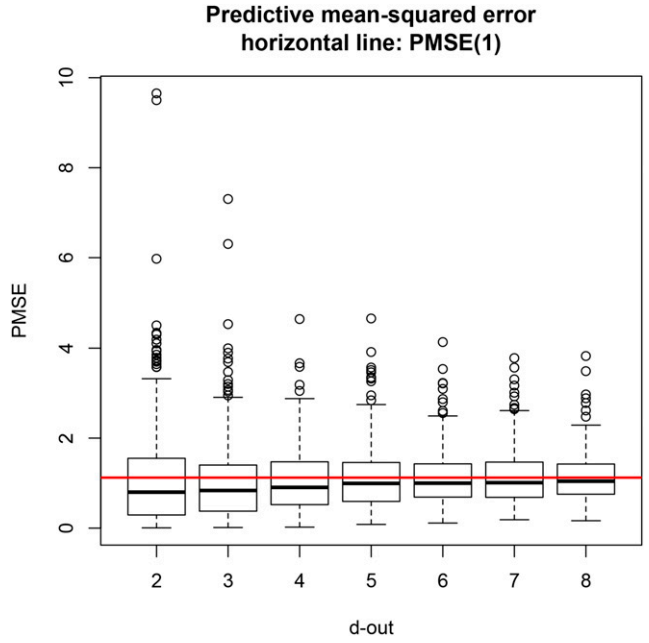

Figure 1 Predictive mean squared error (PMSE) of ordinary least-squares for seven cross-validation (CV) layouts, each replicated 300 times at random. Training sets had size 599 – d (d = 2,3,...,7, and 8). Horizontal line is PMSE for leave-one-out CV.

As shown in *Appendix G*

$$E(\boldsymbol{\beta}|\mathbf{y}_{-i},H)=\frac{E_{\boldsymbol{\beta}|\mathbf{y},H}[w_i(\boldsymbol{\beta})\boldsymbol{\beta}]}{E_{\boldsymbol{\beta}|\mathbf{y},H}[w_i(\boldsymbol{\beta})]},\ i=1,2,\ldots,n. \tag{32}$$

Here, $w_i(\boldsymbol{\beta})=\frac{p(\boldsymbol{\beta}|\mathbf{y}_{-i},H)}{p(\boldsymbol{\beta}|\mathbf{y},H)}$ is called an "importance sampling" weight (Gelfand *et al.* 1992; Albert 2009). Expression (32) implies that the posterior mean of $\boldsymbol{\beta}$ in a training sample can be expressed as the ratio of the posterior means of $w(\boldsymbol{\beta})\boldsymbol{\beta}$, and of $w(\boldsymbol{\beta})$ taken under a Bayesian run using the entire data set. It is shown in *Appendix F* that, given draws $\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}$ $(s=1,2,\ldots,S)$ from the full-posterior distribution, the posterior expectation can in equation (32) be estimated as

$$\hat{E}(\boldsymbol{\beta}|\mathbf{y}_{-i},H)=\sum_{s=1}^{S}w_{i,s}\boldsymbol{\beta}^{(s)};\ i=1,2,\ldots,n, \tag{33}$$

where

$$w_{i,s}=\frac{p^{-1}\left(y_i|\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}\right)}{\sum_{s=1}^{S}p^{-1}\left(y_i|\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}\right)};\ i=1,2,\ldots,n;\quad s=1,2,\ldots,S. \tag{34}$$

By making reference to (31), it turns out that a Monte Carlo estimate of the mean of the predictive distribution of datum i in the Bayesian LOO CV is given by

$$\hat{E}(y_i|\mathbf{y}_{-i},H)=\mathbf{x}_i'\sum_{s=1}^{S}w_{i,s}\boldsymbol{\beta}^{(s)};\ i=1,2,\ldots,n. \tag{35}$$

This type of estimator holds for any Bayesian linear regression model irrespective of the prior adopted, *i.e.*, it is valid for any member of the

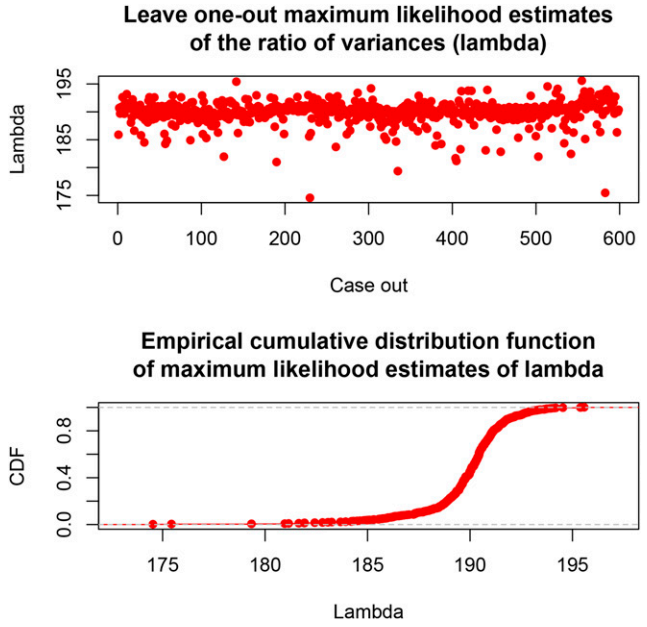

Figure 2 Maximum likelihood estimates of $\lambda = \frac{\sigma_e^2}{\sigma_\beta^2}$ for each of 599 leave-one-out settings; σ_e^2 = residual variance, σ_β^2 = variance of marker effects. The bottom panel gives the empirical distribution function of the estimates.

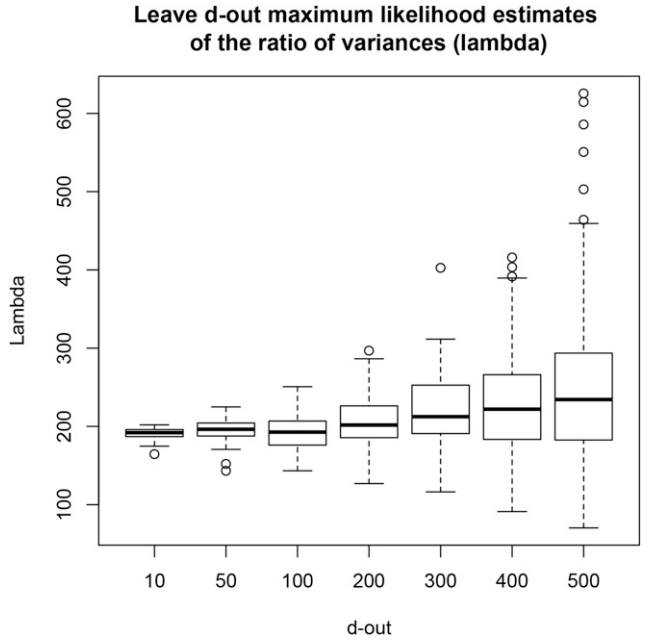

Figure 3 Maximum likelihood estimates of $\lambda = \frac{\sigma_e^2}{\sigma_\beta^2}$ for each of seven CV settings, each replicated 100 times at random. Training sets had size 599 − *d*; *d* = 10, 50, 100, 200, 300, 400, and 500.

"Bayesian alphabet" (Gianola *et al.* 2009; Gianola 2013). In $d-out$ CV, the prediction is

$$\hat{E}\left(\mathbf{y}_d|\mathbf{y}_{[-d]},H\right)=\mathbf{X}_d\sum_{s=1}^{S}w_d\boldsymbol{\beta}^{(s)}, \tag{36}$$

where

$$w_d=\frac{p^{-1}\left(\mathbf{y}_d|\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}\right)}{\sum_{s=1}^{S}p^{-1}\left(\mathbf{y}_d|\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}\right)}, \tag{37}$$

where $p\left(\mathbf{y}_d|\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}\right)=\prod_{j=1}^{d}p^{-1}\left(y_j|\boldsymbol{\beta}^{(s)},\sigma_e^{2(s)}\right)$ for j being a member of the $d-plet$ of observations forming the testing set.

The importance sampling weights are the reciprocal of conditional likelihoods; this specific mathematical representation can produce imprecise estimates of posterior expectations, especially if the posterior distribution with all data has much thinner tails than the posterior based on the training set. Vehtari and Lampinen (2002) calculate the "effective sample size" for a LOO CV as

$$S_{eff,i}=\frac{1}{\sum_{s=1}^{S}w_{i,s}^2}=\frac{1}{S\left[Var(w_i)+\bar{w}_i^2\right]}. \tag{38}$$

If all weights are equal over samples, the weight assigned to any draw is S^{-1}, and the variance of the weights is 0, yielding $S_{eff,i}=S$; on the other hand, $S_{eff,i}$ can be much smaller than S if the variance among weights is large, *e.g.*, when some weights are much larger than others.

The SIR algorithm described by Rubin (1988), Smith and Gelfand (1992) and Albert (2009) can be used to supplement importance sampling; SIR can be viewed as a weighted bootstrap. Let the sampled values and the (normalized) importance sampling weights be $\boldsymbol{\beta}^{(s)}$ and $w_{i,s}$, respectively, for $i=1,2,\ldots,n$ and $s=1,2,\ldots,S$. Then, obtain a resample of size S by sampling with replacement over $\boldsymbol{\beta}^{(1)},\boldsymbol{\beta}^{(2)},\ldots,\boldsymbol{\beta}^{(S)}$ with unequal probabilities proportional to $w_{i,1},w_{i,2},\ldots,w_{i,S}$, respectively, obtaining realizations $\boldsymbol{\beta}_{rep}^{(1)},\boldsymbol{\beta}_{rep}^{(2)},\ldots,\boldsymbol{\beta}_{rep}^{(S)}$. Finally, average realizations for estimating $E(\boldsymbol{\beta}|\mathbf{y}_{-i},H)$ in (31).

The special case of "Bayesian GBLUP"

The term "Bayesian GBLUP" is unfortunate but has become entrenched in animal and plant breeding. It refers to a linear model that exploits genetic or genomic similarity matrix among individuals (as in GBLUP), but where the two variance components are unknown and learned in a Bayesian manner. Prior distributions typically assigned to variances are scale inverted chi-square processes with known scale and degrees of freedom parameters (*e.g.*, Pérez and de los Campos 2014). The model is $\mathbf{y}=\mathbf{g}+\mathbf{e}$, with $\mathbf{e}\sim\mathbf{N}(0,\mathbf{I}\sigma_e^2)$; the hierarchical prior is $\mathbf{g}|\sigma_g^2\sim N(0,\mathbf{G}\sigma_g^2)$; $\sigma_g^2|S_g^2,\nu_g$ and $\sigma_e^2|S_e^2,\nu_e$, where the hyperparameters are $H=(S_g^2,\nu_g,S_e^2,\nu_e)$. A Gibbs sampler iteratively loops over the conditional distributions

$$\begin{aligned}\mathbf{g}|ELSE &\sim N\left(\hat{\mathbf{g}},\mathbf{V}_{g|ELSE}\right),\\ \sigma_g^2|ELSE &\sim \left(\mathbf{g}'\mathbf{G}^{-1}\mathbf{g}+S_g^2\nu_g\right)\chi_{n+\nu_g}^{-2},\\ \sigma_e^2|ELSE &\sim \left[(\mathbf{y}-\mathbf{g})'(\mathbf{y}-\mathbf{g})+S_e^2\nu_e\right]\chi_{n+\nu_e}^{-2}.\end{aligned} \tag{39}$$

Above, *ELSE* denotes the data plus H and all other parameters other than the ones being sampled in a specific conditional posterior distribution; χ_{df}^{-2} indicates the reciprocal of a draw from a central chi-square distribution on df degrees of freedom. The samples of $\mathbf{g}$ are drawn from a multivariate normal distribution of order n. Its mean, $\hat{\mathbf{g}}$, is GBLUP computed at the current state of the variance ratio, which varies at random from iteration to iteration; the covariance matrix is

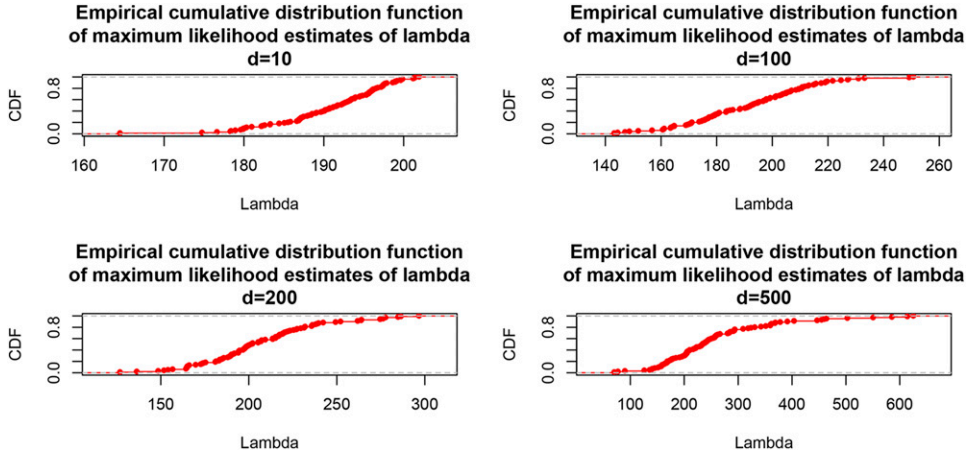

Figure 4 Empirical distribution function of maximum likelihood estimates of $\lambda = \frac{\sigma_e^2}{\sigma_\beta^2}$ for each of four CV settings each replicated 100 times at random; $\sigma_e^2 =$ residual variance, $\sigma_\beta^2 =$ variance of marker effects. Training set sizes were 599 – d; d = 10, 100, 200, and 500.

$\mathbf{V}_{g|ELSE} = \mathbf{C}^{-1}\sigma_e^2$. The current GBLUP is calculated as $\hat{\mathbf{g}} = \mathbf{C}^{-1}\mathbf{y} = (\mathbf{I} + \mathbf{G}^{-1}\lambda)^{-1}\mathbf{y}$; in this representation, $\mathbf{G}^{-1}$ must exist. If it does not, a representation of GBLUP that holds is

$$\hat{\mathbf{g}} = \mathbf{G}(\mathbf{G} + \mathbf{I}\lambda)^{-1}\mathbf{y} = \mathbf{B}\mathbf{y}, \tag{40}$$

where $\mathbf{B}$ is an $n \times n$ matrix of regression coefficients of $\mathbf{g}$ on $\mathbf{y}$. Likewise

$$\mathbf{V}_{g|ELSE} = \mathbf{G}(\mathbf{G} + \mathbf{I}\lambda)^{-1}\sigma_e^2 = \mathbf{B}\sigma_e^2. \tag{41}$$

Once the Gibbs sampler has been run and burn-in iterations discarded, S samples become available for posterior processing, with sample s consisting of $\{g_1^{(s)}, g_2^{(s)}, \ldots, g_n^{(s)}, \sigma_g^{2(s)}, \sigma_e^{2(s)}\}$. In a leave-$d$-out CV, the posterior expectation of g_j (the point predictor of the future phenotype of individual j) is estimated as

$$\hat{E}\left(g_j \middle| \mathbf{y}_{[-d]}, H\right) = \sum_{s=1}^{S} w_{j,s} g_2^{(s)}; j = 1, 2, \ldots, n, \tag{42}$$

where

$$w_{j,s} = \frac{p^{-1}\left(\mathbf{y}_d \middle| \mathbf{g}_d^{(s)}, \sigma_e^{2(s)}\right)}{\sum_{s=1}^{S} p^{-1}\left(\mathbf{y}_d \middle| \mathbf{g}_d^{(s)}, \sigma_e^{2(s)}\right)}; \quad i = 1, 2, \ldots, n; \quad s = 1, 2, \ldots, S, \tag{43}$$

and

$$p\left(\mathbf{y}_d \middle| \mathbf{g}_d^{(s)}, \sigma_e^{2(s)}\right) = \prod_{d \in Test} \frac{1}{\sqrt{2\pi\sigma_e^{2(s)}}} \exp\left[-\frac{\left(y_d - g_d^{(s)}\right)^2}{2\sigma_e^{2(s)}}\right], \tag{44}$$

with the product of the normal densities taken over members of the testing set. Sampling weights may be very unstable if d is large.

Data availability

The authors state that all data necessary for confirming the conclusions presented in the article are represented fully within the article.

EVALUATION OF METHODOLOGY

The methods described here were evaluated using the wheat data available in package BGLR (Pérez and de los Campos 2014). This data set is well curated and has also been used by, *e.g.*, Crossa *et al.* (2010), Gianola *et al.* (2011) and Long *et al.* (2011). The data originated in trials conducted by the International Maize and Wheat Improvement Center (CIMMYT), Mexico. There are 599 wheat inbred lines, each genotyped with 1279 DArT (Diversity Array Technology) markers and planted in four environments; the target trait was grain yield in environment 1. Sample size was then $n = 599$ with $p = 1279$ being the number of markers. These DArT markers are binary $(0, 1)$ and denote presence or absence of an allele at a marker locus in a given line. There is no information on chromosomal location of markers. The objective of the analysis was to illustrate concepts, as opposed to investigate a specific genetic hypothesis. The data set of moderate size allowed extensive replication and reanalysis under various settings.

LOO *vs.* leave-d-out CV: ordinary least-squares

The linear model had an intercept and regressions on markers 301 through 500 in the data file; markers were chosen arbitrarily. Here, $p = 201$ and $n = 599$, ensuring unique OLS estimates of substitution effects, *i.e.*, there was no rank deficiency in $\mathbf{X}$.

Seven CV layouts were constructed in which testing sets of sizes 2, 3,. . ., 7, or 8 lines were randomly drawn (without replacement) from the 599 inbred lines. Training set sizes decreased accordingly, *e.g.*, for $d = 7$, training sample size was 599 – 7= 592. Larger sizes of testing sets were not considered because $(\mathbf{I} - \mathbf{H}_d)$ in (9) became singular as d increased beyond that point. The training-testing process was repeated 300 times at random, to obtain an empirical distribution of prediction mean squared errors.

For LOO CV, regression coefficients were calculated using (5), and the predictive mean squared error was computed as in (7). For the leave-d-out CV, regressions and PMSE were computed with (9) and (11), respectively. Figure 1 shows that the median PMSE for leave-d-out CV was always smaller than the LOO PMSE (horizontal line), although it tended toward the latter as d increased, possibly due to the increasingly smaller training sample size. PMSE in LOO was 1.12, while it ranged from 0.80 to 1.04 for testing sets containing two or more lines. An increase in testing set size at the expense of some decrease in training sample size produced slightly more accurate but less variable predictions (less spread in the distribution of PMSE); this trend can be seen in the box plots depicted in Figure 1. Differences were small but LOO was always less accurate.

BLUP of marker effects

The developments for ridge regression or BLUP of marker effects depend on assuming that allocation of observations into testing sets, with a concomitant decrease in training set size, does not affect the ratio of variance components appreciably.

First, we examined consequences of removing each of the 599 lines at a time on maximum likelihood estimates (MLE) of marker (σ_β^2) and

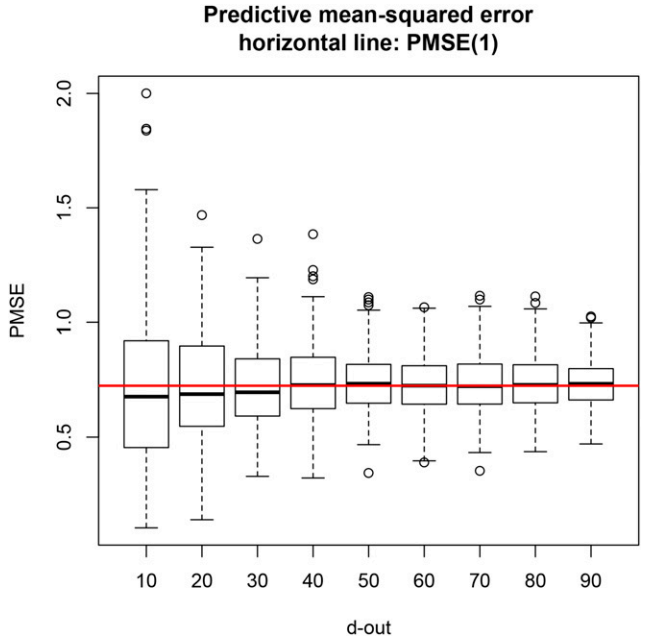

Figure 5 PMSE of BLUP of markers (ridge regression) for 300 testing sets in each of nine CV settings of sizes d = 10, 20, 30, 40, 50, 60, 70, 80, and 90; training set size was 599 − d.

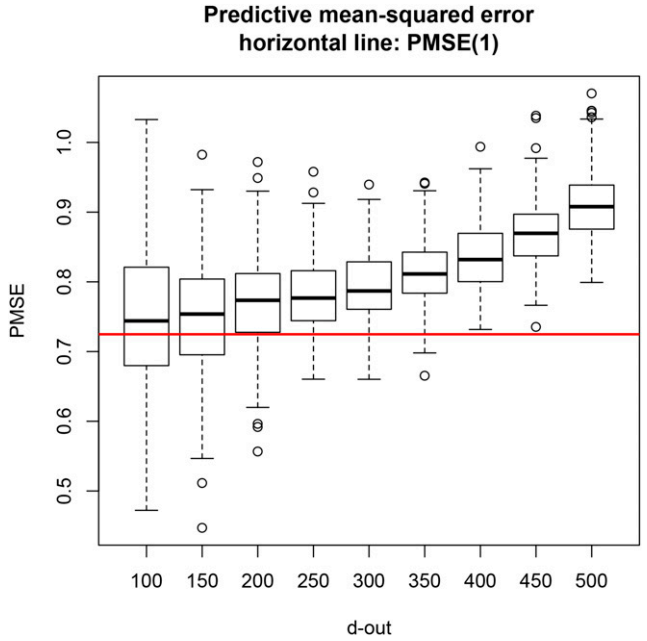

Figure 6 PMSE of BLUP of markers (ridge regression) for 300 testing sets in each of nine CV settings of sizes d = 100, 150, 200, 250, 300, 350, 400, 450, and 500; training set size was 599 − d.

residual (σ_e^2) variances. The model was as in (1), without an intercept (phenotypes were centered), assuming $\boldsymbol{\beta} \sim N(\mathbf{0}, \mathbf{I}\sigma_\beta^2)$ and $\mathbf{e} \sim N(\mathbf{0}, \mathbf{I}\sigma_e^2)$ were independently distributed, and with all 1279 markers used when forming $\mathbf{XX}'$. An eigen-decomposition of $\mathbf{XX}'$ coupled with the *R* function *optim* (G. de los Campos, personal communication) was used for computing MLE. It was assumed that convergence was always to a global maximum, as it was not practical to monitor the 599 implementations for convergence in each case. MLE of $\lambda = \frac{\sigma_e^2}{\sigma_\beta^2}$ were found by replacing the unknown variances by their corresponding estimates.

With all lines used in the analysis, $MLE(\lambda) = 189.9$. Figure 2 displays the 599 estimates of λ, and the resulting empirical cumulative distribution function when LOO was used. Removal of a single line produced MLE of λ ranging from 174.5 to 195.6 (corresponding to estimates of σ_β^2 spanning the range $5.1 - 5.7 \times 10^{-3}$); the 5 and 95 percentiles of the distribution of the LOO estimates of λ were 185.9 and 192.7, respectively. Model complexity (Ruppert *et al.* 2003; Gianola 2013) was gauged by evaluating the "effective number of parameters" as $p_{\text{eff}} = tr[\mathbf{X}(\mathbf{X}'\mathbf{X} + \mathbf{I}\lambda)^{-1}\mathbf{X}']$; the "effective degrees of freedom" are $\nu_e = n - p_{\text{eff}}$. For the entire data set with $n = 599$ and $p = 1279$, variation of λ from 174.5 to 195.6 was equivalent to reducing p_{eff} from 164.2 to 155.3, with ν_e ranging from 435.8 to 443.7. These metrics confirm that the impact of removing a single line from the training process was fairly homogeneous across lines.

Next, we excluded $d = 10, 50, 100, 200, 300, 400$, and 500 lines from the analysis, while keeping $p = 1279$ constant. The preceding was done by sampling with replacement the appropriate number of rows from the entire data matrix $(\mathbf{y}, \mathbf{X})$, and removing these rows from the analysis; the procedure was repeated 100 times at random for each value of d, to obtain an empirical distribution of the MLE. Figure 3 and Figure 4 depict the distributions of estimates. As d increased (training sample size decreased) the median of the estimates and their dispersion increased. Medians were 192.0 $(d = 10)$, 196.3 $(d = 50)$, 192.7 $(d = 100)$, 201.7 $(d = 200)$, 212.5 $(d = 300)$, 222.0 $(d = 400)$, and 234.3 $(d = 500)$. The increase of medians as training sample size decreased can be explained as follows: (a) stronger shrinkage (larger λ) must be exerted on marker effects to learn signal from 1279 markers as sample size decreases. (b) MLE of variance components have a finite sample size bias, which might be upwards for λ here; bias cannot be measured, so the preceding is conjectural.

In short, it appears that keeping λ constant in a LOO setting is reasonable; however, the estimated variance ratio was sensitive with respect to variation in training set size when 100 or more lines, *i.e.*, 15% or more of the total number of cases, were removed for model training.

To assess the impact on PMSE of number of lines removed from training and allocated to testing, 300 testing sets for each of $d = 10, 20, \ldots, 80, 90$ lines were formed by randomly sampling (with replacement) from the 599 lines. The regression model was trained on the remaining lines using the entire battery of 1279 markers with $\lambda = 190$. Figure 5 shows the distribution of PMSE for each of the layouts. A comparison with Figure 1 shows that the PMSEs for BLUP were smaller than for OLS; this was expected because, even though training set sizes were smaller than those used for OLS, BLUP predictions with $\lambda = 190$ are more stable and the model was more complex, since 1279 markers were fitted jointly. As testing set size increased, median PMSE was 0.68 $(d = 10)$ 0.70 $(d = 20)$ and 0.72–0.73 for the other testing set sizes. For LOO, PMSE was 0.72. As in the case of OLS, the distribution of PMSE over replicates became narrower as d grew. As anticipated, decreases in training set size produced a mild deterioration in accuracy of prediction (in an MSE sense) but generated a markedly less variable CV distribution. Testing sets of about 10% of all lines produced a distribution of PMSE with a similar spread to what was obtained with larger testing sets without sacrificing much in mean accuracy. We corroborated that attaining the largest possible training sample is not optimal if done at the expense of testing set size, because predictions are more variable.

Testing sets of size $d = 100$ to $d = 500$ (in increments of 50 lines) were evaluated as in the preceding case, again using 300 replicates for each setting and with $\lambda = 190$. Comparison of Figure 6 with Figure 5 indicates that a marked deterioration in PMSE ensued, which may be

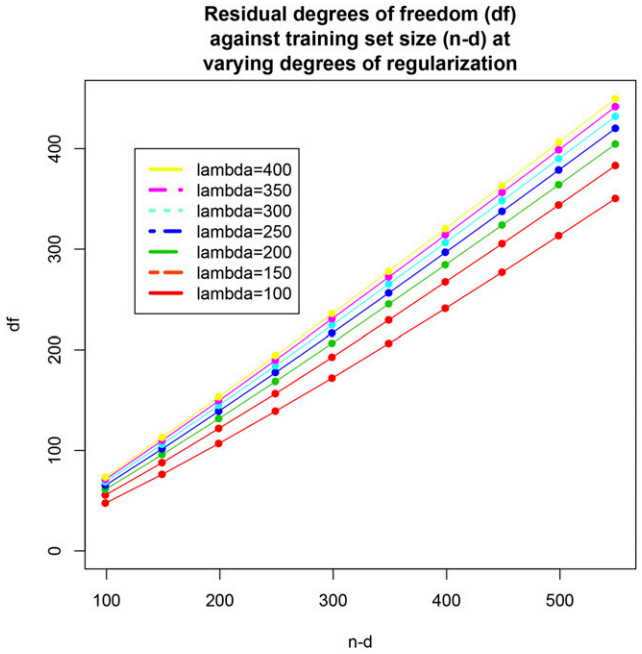

Figure 7 Effective residual degrees of freedom against training sets sizes ($n - d$ = 99, 149, 199, 249, 299, 349, 399, 449, 499, and 549) at selected values of the regularization parameter (λ = 100,150, 200, 250, 300, 350, and 400). Values are averages of 50 random replications.

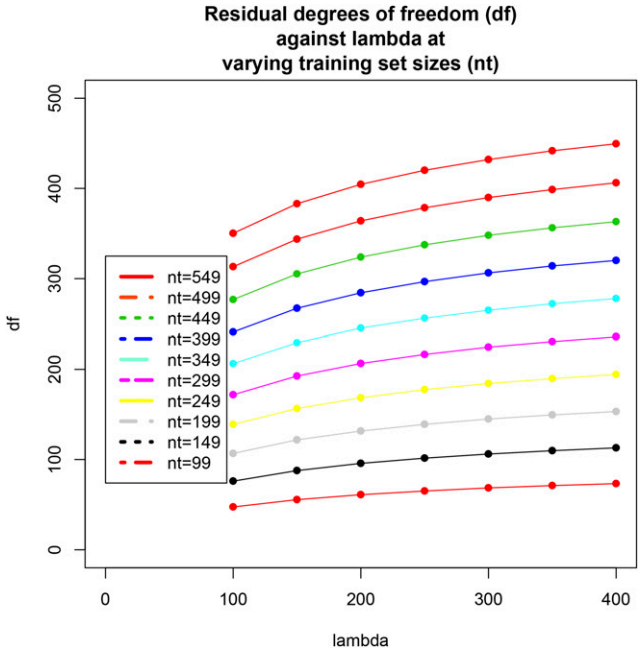

Figure 8 Effective residual degrees of freedom against the regularization parameter (λ = 100, 150, 200, 250, 300, 350, and 400) at various training sets sizes ($n - d$ = 99, 149, 199, 249, 299, 349, 399, 449, 499, 549). Values are averages of 50 random replications.

due to insufficient regularization or overfitting from the decrease in training set size. For example, a testing set of size 500 implies that the model with 1279 markers was trained using only 99 inbred lines. In this case, stronger regularization (shrinkage of regression coefficients toward 0) may be needed than what is effected by $\lambda = 190$. To examine whether overfitting or insufficient regularization was the source of degradation in PMSE, the effective residual degrees of freedom

$$\nu_e = n - d - tr\left[\mathbf{X}_{[-d]}\left(\mathbf{X}'_{[-d]}\mathbf{X}_{[-d]} + \mathbf{I}\lambda\right)^{-1}\mathbf{X}'_{[-d]}\right] \qquad (45)$$

were calculated for each of 70 combinations of $d = (50, 100, \ldots, 450, 500)$ and $\lambda = (100, 150, \ldots, 350, 400)$; each combination was replicated 50 times by sampling with replacement from $(\mathbf{y}, \mathbf{X})$ and extracting the appropriate number of rows. The ν_e values were averaged over the 50 replications. Figure 7 displays ν_e plotted against training set size ($n - d$ = 99, 149,. . .,499, 549): the impact of variations in λ on ν_e was amplified in absolute and relative terms as training set size increased. For instance, for $n - d = 99$, each observation in the training set contributed 0.48 and 0.74 residual degrees of freedom when λ varied from 100 to 400; when $n - d = 549$, the corresponding contributions were 0.64 and 0.82. Figure 8 shows how ν_e varied with λ for each of the training set sizes. Overfitting did not seem to be the cause of degradation in PMSE because the models "preserved" a reasonable number of degrees of freedom in each case considered.

These results reinforce the point that, in shrinkage-based methods, such as GBLUP or any member of the "Bayesian alphabet" (Gianola *et al.* 2009; Gianola 2013), there is an interplay between sample size, model complexity, and strength of regularization. The effective number of parameters in the training process is given by $n - d - \nu_e$, and here it varied from 25.64 $(n - d = 99, \lambda = 400)$ to 198.73 $(n - d = 549, \lambda = 100)$. Even though p =1279 markers were fitted, the model was not able to estimate beyond about 200 linear combinations of marker effects. This illustrates that the "prior" (*i.e.*, the distribution assigned to marker effects) matters greatly when $n \ll p$. In other words, there were $\sim$ 1079 estimates of marker effects that are statistical artifacts from regularization, and which should not be construed as sensible estimates of marker locus effects, as pointed out by Gianola (2013). Bayesian learning would gradually improve over time if n would grow faster than p, which seems unlikely given a tendency toward overmodeling as sequence and postgenomic data accrue.

Genomic BLUP

Standard GBLUP, $\hat{\mathbf{g}}(\lambda)$, of genotypic values of the 599 lines $(\hat{g}_i;\ i = 1, 2, \ldots, 599)$ was computed with $\lambda = 190$. Subsequently, $\hat{\mathbf{g}}_{[-i]}(\lambda)$ was obtained for each of the 599 lines, *i.e.*, the GBLUP of all lines after removing line i in the training process. Euclidean distances between $\hat{\mathbf{g}}(\lambda)$ and $\hat{\mathbf{g}}_{[-i]}(\lambda)$ were calculated as

$$d_i(\lambda) = \sqrt{\left(\hat{\mathbf{g}}(\lambda) - \hat{\mathbf{g}}_{[-i]}(\lambda)\right)'\left(\hat{\mathbf{g}}(\lambda) - \hat{\mathbf{g}}_{[-i]}(\lambda)\right)}; \qquad (46)$$

this metric measures the extent to which removal of line i influences model training. The minimum and maximum absolute distances were 3×10^{-4} and 1.082, respectively, and the coefficient of variation of distances was about 80%. An observation was deemed influential when $d_i(\lambda) \geqq 0.83$, the 99-percentile of the empirical distribution. Figure 9 (top panel) shows a scatter plot of the $d_i(\lambda)$; influential lines (28, 440, 461, 503, 559, and 580) correspond to points on top of the horizontal line. The relationship between the phenotype of the line excluded in the LOO CV is shown in the bottom panel: larger phenotypes tended to be associated with larger distances.

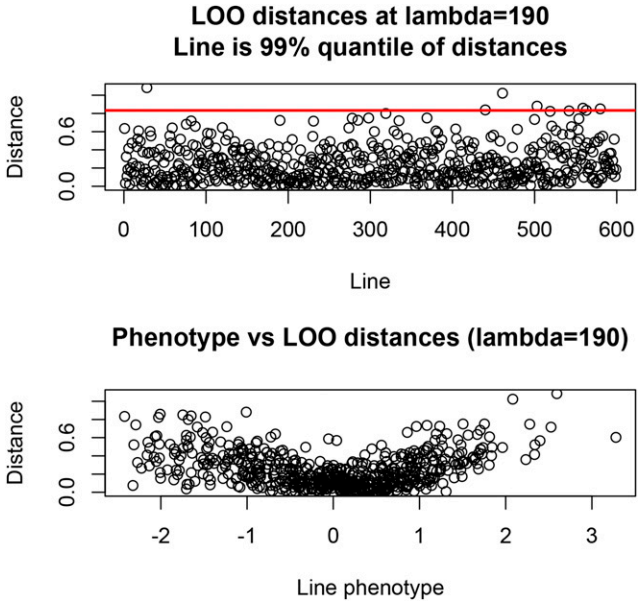

Figure 9 Euclidean distances between genomic BLUP (GBLUP) with all observations and leave-one-out (LOO) GBLUP, by line removed from training in the CV. Bottom panel depicts the association between phenotype left out in training and distances. The regularization parameter used was $\lambda = \frac{\sigma^2_\varepsilon}{\sigma^2_\beta} = 190$.

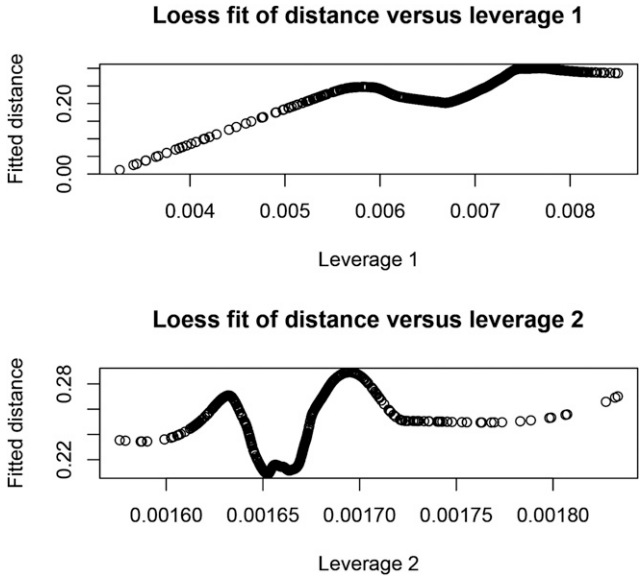

Figure 10 Nonparametric (LOESS) fits of Euclidean distance between GBLUP with all observations and LOO GBLUP on two measures of influence (leverage) a line has on model training. Leverage 1 is the average of absolute values of the regression of all lines on the phenotype of a given line; leverage 2 is the average of such regressions.

Using data from all lines, GBLUP is $\hat{\mathbf{g}} = \mathbf{C}^{-1}\mathbf{y}$; the influence of phenotype of line $j = 1, 2, \ldots, 599$ on GBLUP of line i is element ij of the matrix $\frac{\partial \hat{\mathbf{g}}}{\partial \mathbf{y}'} = \mathbf{C}^{-1}$. Observe that

$$\mathbf{C}^{-1} = Cov(\mathbf{g}, \mathbf{y}) Var^{-1}(\mathbf{y}) = \left(\mathbf{I} + \mathbf{G}^{-1}\lambda\right)^{-1} \tag{47}$$

is as a matrix of $n \times n$ regression coefficients; its i^{th} row contains the regression of the genotype of line i on the n phenotypes. A measure of overall influence (leverage) of line j is the average of values (or absolute values) of elements in column j of $\mathbf{C}^{-1}$. Clearly, leverages depend on relatedness structure and on λ but not on phenotypes. Figure 10 depicts plots of LOESS regressions (Cleveland 1979) of Euclidean distance between GBLUP calculated with all lines, and LOO GBLUP on two measures of leverage: the average of absolute values of $\mathbf{C}^{-1}$ over rows for each line (leverage 1), and the average of elements of $\mathbf{C}^{-1}$ over rows, by line (leverage 2). LOESS fits (span parameter equal to 0.50) indicated that leverage 1 informs about the impact of removing a specific line in LOO: the larger the leverage 1 of a line, the larger the effect of its removal from the training process.

RKHS

We built a kernel matrix $\mathbf{K}$ with typical element (ranging between 0 and 1)

$$k_{ij} = w_1 \exp\left[-h_1 \frac{d_{ij}}{\max(d_{ij})}\right] + w_2 \exp\left[-h_2 \frac{d_{ij}}{\max(d_{ij})}\right] + w_3 \exp\left[-h_3 \frac{d_{ij}}{\max(d_{ij})}\right]; \tag{48}$$

$d_{ij} = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j)$ for $i, j = 1, 2..., 599$. Here $\mathbf{x}_i$ is the 1279×1 vector of marker genotypes in line i; h_1, h_2, and h_3 are bandwidth parameters tuned to establish "global," "regional" and "local" similarities between individuals (as h increases, similarity decreases); w_1, w_2, and w_3 are weights assigned to the three sources of similarity, such that $0 < w_i < 1$ and $\sum_{i=1}^{3} w_i = 1$. We arbitrarily chose $h_1 = \frac{1}{2}$, $h_2 = 2$, and $h_3 = 4$, and $w_1 = 0.5$, $w_2 = 0.30$, and $w_3 = 0.20$. From $\mathbf{K}$ we created three additional kernels by placing $w_i = 1$ for $i = 1, 2, 3$, leading to matrices $\mathbf{K}_1$, $\mathbf{K}_2$, and $\mathbf{K}_3$. Mean off-diagonal elements of the four kernel matrices were 0.73 ($\mathbf{K}_1$), 0.29 ($\mathbf{K}_2$), 0.09 ($\mathbf{K}_3$), and 0.47 ($\mathbf{K}$); these values can be interpreted as correlations between pairs of individuals. Hence, $\mathbf{K}_1$ and $\mathbf{K}_3$ produced the highest and lowest degrees of correlation, respectively; complexity of the models increases from kernel 1 to kernel 3 because the fit to the data increases with h (de los Campos *et al.* 2009, 2010).

The four no-intercept RKHS models had the basic form

$$\mathbf{y} = \mathbf{K}\boldsymbol{\alpha} + \mathbf{e}; \boldsymbol{\alpha} \sim N\left(\mathbf{0}, \mathbf{K}^{-1}\sigma^2_\alpha\right); \mathbf{e} \sim N\left(\mathbf{0}, \mathbf{I}\sigma^2_e\right); Cov\left(\boldsymbol{\alpha}, \mathbf{e}'\right) = \mathbf{0}, \tag{49}$$

where $\mathbf{K}$ was either as in (48) or $\mathbf{K}_i = 1, 2, 3$. Variance components were estimated by maximum likelihood, producing as estimates of $\lambda = \frac{\sigma^2_\varepsilon}{\sigma^2_\alpha}$: $\lambda_1 = 0.16$, $\lambda_2 = 0.30$, $\lambda_3 = 0.32$, and $\lambda_K = 0.21$. The effective number of parameters was calculated (*e.g.*, kernel 1) as $p_1 = tr(\mathbf{I} + \mathbf{K}^{-1}\lambda_1)^{-1}$, yielding 224.5, 319.2, and 376.3 for kernel matrices 1, 2, and 3, respectively; for $\mathbf{K}$ the effective number of parameters fitted was 330.3. As expected, model complexity increased as the model became more "local."

We fitted the four RKHS models to all 599 lines, and conducted a LOO CV for each model. In the fitting process, the corresponding regularization parameter was employed; *e.g.*, for $\mathbf{K}_2$, $\lambda_2 = 0.30$. For each of the models, RKHS predictions of genotypic values were calculated as

$$\hat{\mathbf{g}}_i(\lambda_i, \mathbf{h}, \mathbf{w}) = \left(\mathbf{I} + \mathbf{K}_i^{-1}\lambda_i\right)^{-1}\mathbf{y}; \; i = 1, 2, 3, K. \tag{50}$$

The implicit dependence of predictions on bandwidths ($\mathbf{h}$) and weights ($\mathbf{w}$) is indicated in the notation above, but not used hereinafter. In LOO (line j left out in the training process), predictions and

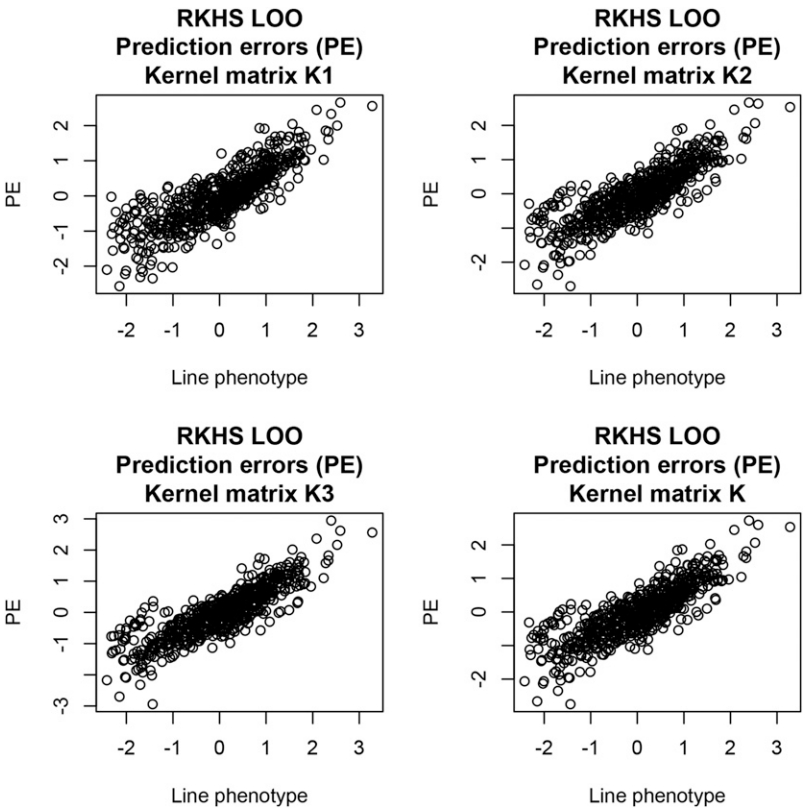

Figure 11 LOO prediction errors (testing set) of four reproducing kernel Hilbert spaces (RKHS) regression models against line phenotypes.

predictive mean-squared errors are calculated as for LOO GBLUP, that is,

$$\tilde{g}_{ij} = \left(1-c_i^{jj}\right)^{-1}\left(\hat{g}_{ij} - c_i^{jj}y_j\right);\ j = 1, 2, \ldots, n, \tag{51}$$

where c_i^{jj} is the j^{th} diagonal element of $(\mathbf{I} + \mathbf{K}_i^{-1}\lambda_i)^{-1}$, and

$$PMSE_i(1) = \frac{1}{n}\sum_{j=1}^{n}\left(y_j - \tilde{g}_{ij}\right)^2. \tag{52}$$

Predictive MSEs were 0.6795 ($\mathbf{K}_1$), 0.6446 ($\mathbf{K}_2$), 0.6555 ($\mathbf{K}_3$), and 0.6439 ($\mathbf{K}$); predictive correlations were 0.566, 0.597, 0.591, and 0.598. Differences between kernels with respect to the criteria used were nil, but the model combining three kernels conveying differing degrees of locality had a marginally smaller MSE and a slightly larger correlation. LOO prediction errors plotted against line phenotypes are shown in Figure 11 for the four kernels used. Prediction errors were larger in absolute value for lines with lowest and highest grain yields, suggesting that the model may benefit by accounting for possibly heterogeneous residual variances. $\mathbf{K}_2$ and $\mathbf{K}_3$ captured some substructure in the distribution of fitted residuals. The more global kernel ($\mathbf{K}_1$), arguably capturing mostly additive effects, did not suggest any substructure, which reemerged when the three kernels were combined into $\mathbf{K}$. The preceding exercise illustrates that predictive correlations and PMSE do not fully describe the performance of a prediction machine.

Bayesian GBLUP with known variance components

Bayesian GBLUP with known variances has a closed form solution: using all data, the posterior distribution is $\mathbf{g}|\mathbf{y}, \sigma_e^2, \sigma_g^2 \sim N(\hat{\mathbf{g}}, \mathbf{C}^{-1}\sigma_e^2)$, where $\hat{\mathbf{g}} = \mathbf{C}^{-1}\mathbf{y}$ and $\mathbf{C}^{-1} = (\mathbf{I} + \mathbf{G}^{-1}\lambda)^{-1}$. Set $\mathbf{G} = \mathbf{XX}'$, $\lambda = 190$, and $\sigma_e^2 = 0.54$; $\mathbf{X}$ is centered.

This problem was attacked (with Monte Carlo error) by drawing independent samples from the 599-variate normal posterior distribution; no MCMC is needed. Using (34), the importance sampling weights for LOO are

$$w_{i,s} = \frac{p^{-1}\left(y_i|g_i^{(s)}, 0.54\right)}{\sum_{s=1}^{S} p^{-1}\left(y_i|g_i^{(s)}, 0.54\right)};\quad i = 1, 2, \ldots, n;\quad s = 1, 2, \ldots, S, \tag{53}$$

where $g_i^{(s)}$ is sample s for line i; $\mathbf{g}^{(s)}$ is drawn from $N(\hat{\mathbf{g}}, \mathbf{C}^{-1}0.54)$. The importance sampling weight becomes

$$w_{i,s} = \frac{\exp\left[\frac{\left(y_i - g_i^{(s)}\right)^2}{1.08}\right]}{\sum_{s=1}^{S}\exp\left[\frac{\left(y_i - g_i^{(s)}\right)^2}{1.08}\right]}. \tag{54}$$

Observe that the likelihood that y_i confers to $g_i^{(s)}$ is inversely proportional to $w_{i,s}$. Hence, samples in which the phenotype removed (y_i) confers little likelihood to the g_i receive more weight.

We took S =15,000 independent samples from $N(\hat{\mathbf{g}}, \mathbf{C}^{-1}0.54)$. The effective number of weights per line, calculated with (38) ranged from 76.4 to 14,983.5; the median (mean) weight was 10,991.0 (9789.2), and the first and third quartiles of the distribution were 6826.1 and 14,983.5, respectively. On average, 1.54 independent samples were required for drawing an effectively independent LOO posterior sample. Figure 12

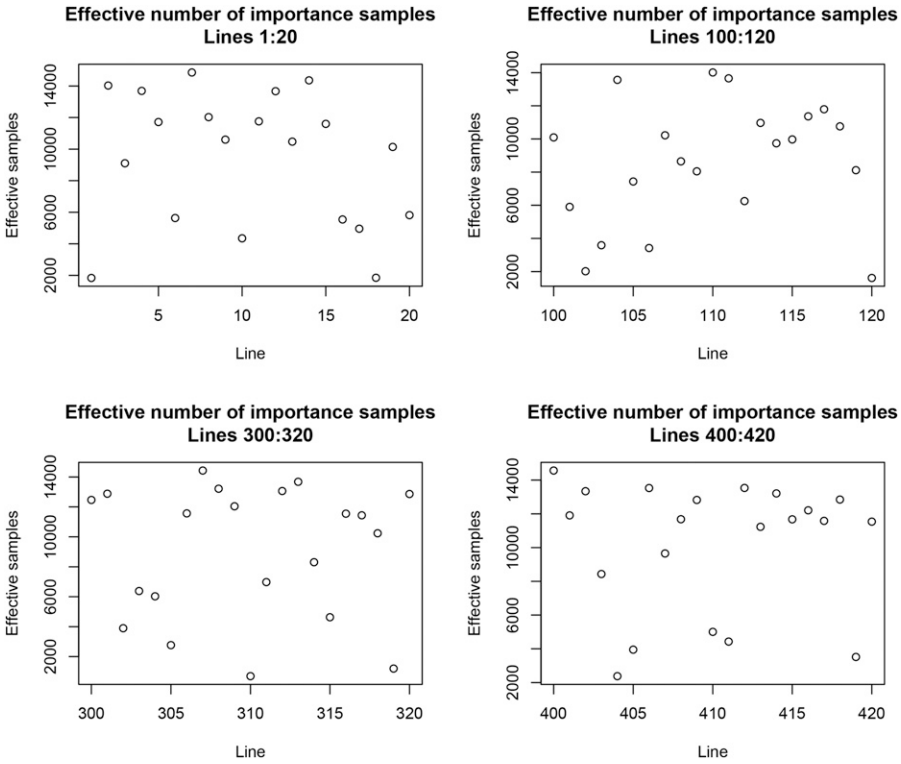

Figure 12 Effective number of importance samples for LOO Bayesian GBLUP (given the variances) for selected lines; 15,000 independent samples were drawn from the posterior distribution of genotypic values.

illustrates the variability among some arbitrarily selected lines of the mean number of importance weights. A phenotype having a small mean importance weight would be "surprising" with respect to the model. However, there are theoretical and numerical issues with the weights used here, a point retaken in the discussion.

Figure 13 shows that GBLUP using the entire sample of size n fitted closely. Correlations between predictors were 0.91 in all cases. Mean-squared errors were 0.40 (GBLUP, entire sample), 0.73 (Bayes LOO), and 0.73 (LOO GBLUP). The correlation between predictions and phenotypes was 0.81 for GBLUP (entire sample), 0.52 for LOO GBLUP, and 0.51 for Bayes LOO.

The importance sampling scheme worked well. Ionides (2008) suggested a modification of the importance ratio assigned to sample s and data point i, $r_{i,s} = p^{-1}(y_i|\boldsymbol{\beta}^{(s)}, \sigma_e^{2(s)})$, as follows

$$r'_{i,s} = \min\left(r_{i,s}, \sqrt{S}\,\bar{r}_i\right), \tag{55}$$

where $\bar{r}_i$ is the average (over samples) importance sample ratio for line i in the context of the wheat data set. Ionides (2008) argued that truncation of large importance sampling weights (TIS) would be less sensitive with respect to the importance sampling density function used (the posterior distribution using all data points in our case) than IS. We converted the IS weights into normalized TIS weights using rule (55) applied to the normalized IS weights. As mentioned earlier, the effective number of normalized IS weights ranged between 76.4 and 14983.5; for normalized TIS weights, the effective number spanned from 691.9 to 13,970. TIS produced more stable weights than standard IS. However, truncation of weights introduces a bias, which may affect predictive performance adversely (Vehtari *et al.* 2016). However, it may be that TIS made the weights "too homogeneous," thus creating a bias toward the posterior distribution obtained with all data points. If all weights were equal to a constant, IS or TIS would retrieve the full posterior distribution.

DISCUSSION

Cross-validation (CV) has become an important tool for calibrating prediction machines in genome-enabled prediction (Meuwissen *et al.* 2001), and is often preferred over resampling methods such as bootstrapping. It gives a means for comparing and calibrating methods and training sets (*e.g.*, Isidro *et al.* 2015). Typically, CV requires dividing data into training and testing folds, and the models must be run many times. An extreme form of CV is LOO; here if the sample has size n, each observation is removed in the training process, and labeled as a testing set of size 1. Hence, n different models must be run to complete a LOO CV.

Our paper presented statistical methodology aimed at enabling extensive CV in genome-enabled prediction using a suite of methods. These were OLS, BLUP on markers, GBLUP, RKHS, and Bayesian procedures. Formulae were derived that enable arriving at the predictions that would be obtained if one or more cases were to be excluded from the training process and declared as members of the testing set. In the cases of OLS, BLUP, GBLUP, and RKHS, and assuming that the ratio of variance components do not change appreciably from those that apply to the entire sample, the formulae are exact.

The deterministic formulae can also be applied in a multiple-kernel or multiple-random factors setting, given the variance components. For example, consider the bikernel RKHS regression (*e.g.*, de los Campos *et al.* 2010; Tusell *et al.* 2014)

$$\mathbf{y} = \mathbf{g}_P + \mathbf{g}_M + \mathbf{e} = \mathbf{K}_P\boldsymbol{\alpha}_P + \mathbf{K}_M\boldsymbol{\alpha}_M + \mathbf{e}. \tag{56}$$

Here, $\mathbf{g}_P = \mathbf{K}_P\boldsymbol{\alpha}_P$ is genetic signal captured by pedigree, and $\mathbf{g}_M = \mathbf{K}_M\boldsymbol{\alpha}_M$ is genetic signal captured by markers; $\boldsymbol{\alpha}_P$ and $\boldsymbol{\alpha}_M$ are unknown and independently distributed RKHS regression coefficients, and $\mathbf{K}_P$ and $\mathbf{K}_M$ are positive-definite similarity matrices. Suppose that $\mathbf{K}_P = \mathbf{A}$, *i.e.*, the numerator relationship matrix based on the assumption of additive inheritance, and that $\mathbf{K}_M$ is a Gaussian

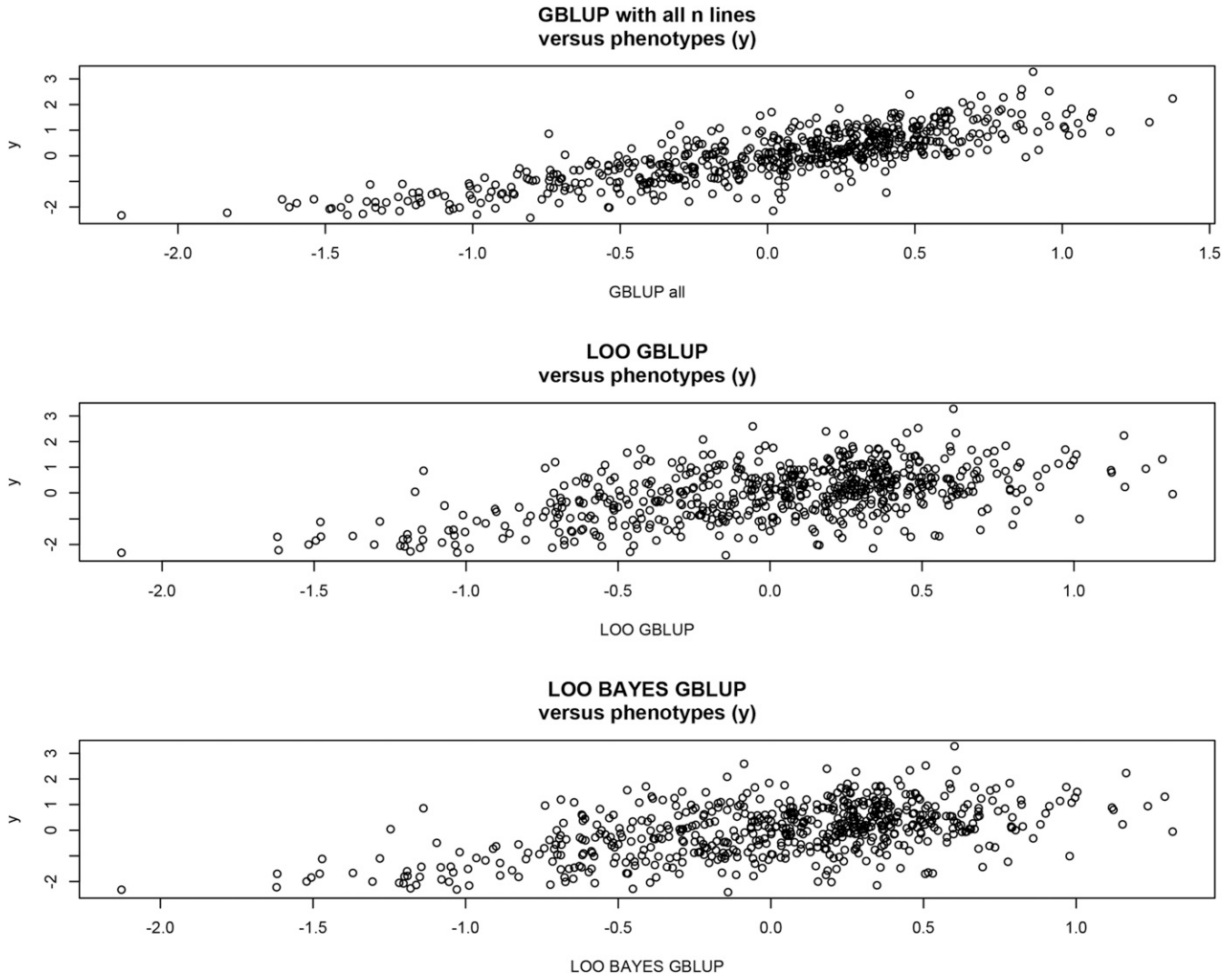

Figure 13 Associations between phenotypes and predictions (GBLUP with all $n = 599$ lines used in training; LOO GBLUP, leave-one-out genomic BLUP; LOO BAYES GBLUP, direct sampling from the posterior distribution of genotypic values of followed by importance sampling to obtained the LOO predictions.

kernel such as those employed earlier in the paper. The standard assumption for the RKHS regression coefficients is

$$\begin{bmatrix} \boldsymbol{\alpha}_P \\ \boldsymbol{\alpha}_M \end{bmatrix} \sim N\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{A}^{-1}\sigma^2_{\alpha_A} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_M^{-1}\sigma^2_{\alpha_M} \end{bmatrix} \right) \tag{57}$$

where $\sigma^2_{\alpha_A}$ and $\sigma^2_{\alpha_M}$ are variance components associated with kernels $\mathbf{A}$ and $\mathbf{K}_M$, respectively. Hence, $\mathbf{g}_P \sim N(\mathbf{0}, \mathbf{A}\sigma^2_{\alpha_A})$ and $\mathbf{g}_M \sim N(\mathbf{0}, \mathbf{K}_M\sigma^2_{\alpha_M})$. The BLUP of $\mathbf{g}_P$ and $\mathbf{g}_M$ can be found by solving the linear system

$$\begin{bmatrix} \mathbf{I} + \mathbf{A}^{-1}\lambda_A & \mathbf{I} \\ \mathbf{I} & \mathbf{I} + \mathbf{K}_M^{-1}\lambda_M \end{bmatrix} \begin{bmatrix} \hat{\mathbf{g}}_P \\ \hat{\mathbf{g}}_M \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ \mathbf{y} \end{bmatrix}, \tag{58}$$

where $\lambda_A = \dfrac{\sigma^2_e}{\sigma^2_{\alpha_A}}$ and $\lambda_M = \dfrac{\sigma^2_e}{\sigma^2_{\alpha_M}}$. Hence, the solutions satisfy

$$\hat{\mathbf{g}}_P = \left(\mathbf{I} + \mathbf{A}^{-1}\lambda_A\right)^{-1}\left(\mathbf{y} - \hat{\mathbf{g}}_M\right) = \mathbf{C}_A^{-1}\left(\mathbf{y} - \hat{\mathbf{g}}_M\right), \tag{59}$$

$$\hat{\mathbf{g}}_M = \left(\mathbf{I} + \mathbf{K}_M^{-1}\lambda_M\right)^{-1}\left(\mathbf{y} - \hat{\mathbf{g}}_P\right) = \mathbf{C}_M^{-1}\left(\mathbf{y} - \hat{\mathbf{g}}_P\right). \tag{60}$$

Applying the logic leading to (102) for the LOO situation, the preceding equations can be written as

$$\tilde{g}_{i,P} = \frac{1}{1 - c_A^{ii}}\left[\hat{g}_{i,P} - c_A^{ii}\left(y_i - \hat{g}_{i,M}\right)\right]; \tag{61}$$

$$\tilde{g}_{i,M} = \frac{1}{1 - c_M^{ii}}\left[\hat{g}_{i,M} - c_M^{ii}\left(y_i - \hat{g}_{i,P}\right)\right], \tag{62}$$

where $c_A^{ii}(c_M^{ii})$ are the i^{th} diagonal elements of $\mathbf{C}_A^{-1}(\mathbf{C}_M^{-1})$, respectively; $\hat{g}_{i,P}$ and $\hat{g}_{i,M}$ are the corresponding solutions to the system of equations (58). The prediction of the left-out phenotype is $\tilde{y}_i = \tilde{g}_{i,P} + \tilde{g}_{i,M}$.

The situation is different for Bayesian models solved by sampling methods such as MCMC. Typically, there is no closed form solution, except in some stylized situations, so sampling must be used. The Bayesian model must be run with the entire data set and posterior samples weighted, *e.g.*, via importance sampling, to convert realizations into draws pertaining to the posterior distribution that would result from using just the CV training set. Unfortunately, importance weights can be extremely variable. In (34), it can be seen that weights are

proportional to the reciprocal of likelihoods evaluated at the posterior samples, so if a data point (or a vector of data points) "left out" confers a tiny likelihood to the realized value of the parameter sampled, then the sample is assigned a very large weight; on the other hand, if the likelihood is large, the weight is small. This phenomenon produces a large variance among weights, which we corroborated empirically. Another view at the issue at stake is as follows: the importance weight is $w_i(\boldsymbol{\beta}) = \frac{p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H)}{p(\boldsymbol{\beta}|\mathbf{y}, H)}$; hence, if the posterior obtained with all data points has much thinner tails than the posterior density constructed by excluding one or more cases, the weights can "blow up."

The preceding problem would be exacerbated by including more than one observation in the testing set. Vehtari *et al.* (2016) examined seven data sets, and used "brute force" LOO MSE as a gold standard to examine the performance of various forms of importance sampling. TIS gave a better performance than standard importance sampling (IS) weights in two out of seven comparisons, with the standard method being better in three of the data sets; there were two ties. Vehtari *et al.* (2016) suggested another method called Pareto smoothed importance sampling (PSIS) that was better than IS in four of the data sets (two ties), and better than TIS in three of the analyses (two ties). Calculation of PSIS is involved, requiring several steps in our context: (a) compute IS ratios; (b) for each of the 599 lines, fit a generalized Pareto distribution to the 20% largest values found in (a); (c) replace the largest M IS ratios by expected values of the order statistics of the fitted generalized Pareto distribution, where $M = 0.20 \times S$; (d) for each line, truncate the new ratios. Clearly, this procedure does not lend itself to large scale genomic data, and the results of Vehtari *et al.* (2016), obtained with small data sets and simple models, are not conclusive enough. Additional research is needed to examine whether TIS, IS, or PSIS are better for genome-enabled prediction.

It is known (*e.g.*, Henderson 1973, 1984; Searle 1974) that the best predictor, that is, the function of the data with the smallest squared prediction error (MSE) under conceptual repeated sampling (*i.e.*, infinite number of repetitions over the joint distribution of predictands and predictors) is $E(\mathbf{g}|\mathbf{y}, \text{parameters})$. This property requires knowledge of the form of the joint distribution, and of its parameters. In the setting of the case study, under multivariate normality, and with a zero-mean model and known variance components, GBLUP is the best predictor. However, in CV, the property outlined above does not hold. One reason is that the data set represents a single realization of the conceptual scheme. Another reason is that incidence and similarity matrices change at random in CV, plus parameters are estimated from the data at hand. For example, if datum i is removed from the analysis, the training model genomic relationship matrix becomes $\mathbf{G}_{[-i,-i]}$, whereas, if observation j is removed, the matrix used is $\mathbf{G}_{[-j,-j]}$. Further, the entire data set is used in the CV, so yet-to-be observed data points appear in the training process at some point. The setting of best prediction requires that the structure of the data remains constant over repeated sampling, with the only items changing being the realized values of the data $(\mathbf{y})$, and of the unobserved genotypic values $(\mathbf{g})$. The CV setting differs from the idealized scheme, and expectations based on theory may not always provide the best effective guidance in predictive inference.

In conclusion, CV appears to be the best gauge for calibrating prediction machines. Results presented here provide the basis for conducting extensive cross-validation from results of a single run with all data. Future research should evaluate importance sampling schemes for more complex Bayesian models, *e.g.*, those using thick-tailed processes or mixtures as prior distributions. An important challenge is to make the procedures developed here computationally cost-effective, so that software for routine use can be developed.

ACKNOWLEDGMENTS

The Editor and two anonymous reviewers suggested useful comments concerning the structure of the manuscript. We are grateful to the Institut Pasteur de Montevideo, Uruguay, for providing office space and facilities to D.G. from January–March 2016. Part of this work was done while D.G. was a Hans Fischer Fellow at the Institute of Advanced Study, Technical University of Munich-München, Germany; their support is gratefully acknowledged. Research was partially supported by a United States Department of Agriculture (USDA) Hatch Grant (142-PRJ63CV) to D.G., and by the Wisconsin Agriculture Experiment Station.

LITERATURE CITED

Albert, J., 2009 Bayesian Computation with R, Ed. 2. Springer, New York.

Astle, W., and D. Balding, 2009 Population structure and cryptic relatedness in genetic association studies. Stat. Sci. 24: 451–471.

Cantet, R. J. C., R. L. Fernando, and D. Gianola, 1992 Bayesian inference about dispersion parameters of univariate mixed models with maternal effects: theoretical considerations. Genet. Sel. Evol. 24: 107–135.

Chesnais, J. P., T. A. Cooper, G. R. Wiggans, M. Sargolzaei, J. E. Pryce *et al.*, 2016 Using genomics to enhance selection of novel traits in North American dairy cattle. J. Dairy Sci. 99: 2413–2427.

Cleveland, W. S., 1979 Robust locally weighted regression and smoothing scatterplots. J. Am. Stat. Assoc. 74: 829–836.

Crossa, J., G. de los Campos, P. Pérez, D. Gianola, J. Burgueño *et al.*, 2010 Prediction of genetic values of quantitative traits in plant breeding using pedigree and molecular markers. Genetics 186: 713–724.

Daetwyler, H. D., B. Villanueva, and J. A. Woolliams, 2008 Accuracy of predicting the genetic risk of disease using a genome-wide approach. PLoS One 3: e3395.

de los Campos, G., D. Gianola, and G. J. M. Rosa, 2009 Reproducing kernel Hilbert spaces regression: a general framework for genetic evaluation. J. Anim. Sci. 87: 1883–1887.

de los Campos, G., D. Gianola, G. J. M. Rosa, K. A. Weigel, and J. Crossa, 2010 Semi-parametric genomic-enabled prediction of genetic values using reproducing kernel Hilbert spaces methods. Genet. Res. 92: 295–308.

de los Campos, G., D. Gianola, and D. A. B. Allison, 2011 Predicting genetic predisposition in humans: the promise of whole-genome markers. Nat. Rev. Genet. 11: 880–886.

de los Campos, G., J. M. Hickey, R. Pong-Wong, H. D. Daetwyler, and M. P. L. Calus, 2013 Whole-genome regression and prediction methods applied to plant and animal breeding. Genetics 193: 327–345.

Dempfle, L., 1977 Relation entre BLUP (best linear unbiased prediction) et estimateurs bayésiens. Ann. Genet. Sel. Anim. 9: 27–32.

Deng, C. Y., 2011 A generalization of the Sherman-Morrison-Woodbury formula. Appl. Math. Lett. 24: 1561–1564.

Erbe, M., E. C. G. Pimentel, A. R. Sharifi, and H. Simianer, 2010 Assessment of cross-validation strategies for genomic prediction in cattle. Book of Abstracts of the 9th World Congress of Genetics Applied to Livestock Production, Leipzig, Germany, p. S. 129.

Gelfand, A. E., 1996 Model determination using sampling-based methods, pp. 145–162 in Markov Chain Monte Carlo in Practice, edited by Gilks, W. R., S. Richardson, and D. J. Spiegelhalter. Chapman and Hall, London.

Gelfand, A., D. K. Dey, and H. Chang, 1992 Model determination using predictive distributions with implementation via sampling-based methods. Technical Report no. 462. Prepared under contract for the Office of Naval Research. Stanford University, Stanford, CA.

Gianola, D., 2013 Priors in whole genome regression: the Bayesian alphabet returns. Genetics 194: 573–596.

Gianola, D., and G. de los Campos, 2008 Inferring genetic values for quantitative traits non-parametrically. Genet. Res. 90: 525–540.

Gianola, D., and R. L. Fernando, 1986 Bayesian methods in animal breeding theory. J. Anim. Sci. 63: 217–244.

Gianola, D., and G. J. M. Rosa, 2015 One hundred years of statistical developments in animal breeding. Annu. Rev. Anim. Biosci. 3: 19–56.

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

Gianola, D., R. L. Fernando, and A. Stella, 2006 Genomic assisted prediction of genetic value with semi-parametric procedures. Genetics 173: 1761–1776.

Gianola, D., G. de los Campos, W. G. Hill, E. Manfredi, and R. L. Fernando, 2009 Additive genetic variability and the Bayesian alphabet. Genetics 187: 347–363.

Gianola, D., H. Okut, K. A. Weigel, and G. J. M. Rosa, 2011 Predicting complex quantitative traits with Bayesian neural networks: a case study with Jersey cows and wheat. BMC Genet. 12: 87. DOI: .10.1186/1471–2156–12–87

Gianola, D., K. A. Weigel, N. Krämer, A. Stella, and C. C. Schön, 2014a Enhancing genome-enabled prediction by bagging genomic BLUP. PLoS One 9(4): e91693.

Gianola, D., G. Morota, and J. Crossa, 2014b Genome-enabled prediction of complex traits with kernel methods: what have we learned? Proceedings of the 10th World Congress on Genetics Applied to Livestock Production, Vancouver, British Columbia, Canada. Available at: https://asas.org/wcgalp-proceedings.

González-Recio, O., G. J. M. Rosa, and D. Gianola, 2014 Machine learning methods and predictive ability metrics for genome-wide prediction of complex traits. Livest. Sci. 166: 217–231.

Hastie, T., R. Tibshirani, and J. Friedman, 2009 The Elements of Statistical Learning, Ed. 2. Springer, New York.

Henderson, C. R., 1973 Sire evaluation and genetic trends, Proceedings of the Animal Breeding and Genetics Symposium in Honor of Dr. Jay L. Lush. American Dairy Science Association & American Society of Animal Science, Champaign, IL, pp. 10–41.

Henderson, C. R., 1975 Best linear unbiased estimation and prediction under a selection model. Biometrics 31: 423–449.

Henderson, C. R., 1984 Application of Linear Models in Animal Breeding, University of Guelph, Canada.

Henderson, C. R., O. Kempthorne, S. R. Searle, and C. M. von Krosigk, 1959 Estimation of environmental and genetic trends from records subject to culling. Biometrics 15: 192–218.

Heslot, N., H. P. Yang, M. E. Sorrells, and J. L. Jannink, 2012 Genomic selection in plant breeding: a comparison of models. Crop Sci. 52: 146–160.

Hoerl, A. E., and R. W. Kennard, 1970 Ridge regression: applications to non-orthogonal problems. Technometrics 12: 69–82.

Ionides, E. L., 2008 Truncated importance sampling. J. Comput. Graph. Stat. 17: 295–311.

Isidro, J., J. L. Jannink, D. Akdemir, J. Poland, N. Heslot *et al.*, 2015 Training set optimization under population structure in genomic selection. Theor. Appl. Genet. 128: 145–158.

Jiang, Y., and J. C. Reif, 2015 Modeling epistasis in genomic selection. Genetics 201: 759–768.

Lehermeier, C., V. Wimmer, T. Albrecht, and H. J. Auinger, D. Gianola*et al.*, 2013 Sensitivity to prior specification in Bayesian genome-based prediction models. Stat. Appl. Genet. Mol. Biol. 12: 1–17.

Long, N., D. Gianola, G. J. M. Rosa, and K. A. Weigel, 2011 Marker-assisted prediction of non-additive genetic values. Genetica 139: 843–854.

López de Maturana, E., S. J. Chanok, A. C. Picornell, N. Rothman, J. Herranz *et al.*, 2014 Whole genome prediction of bladder cancer risk with the Bayesian LASSO. Genet. Epidemiol. 38: 467–476.

MacLeod, I. M., B. J. Hayes, C. J. Vander Jagt, K. E. Kemper, and M. Haile-Mariam*et al.*, 2014 *A Bayesian analysis to exploit imputed sequence variants for QTL discovery*. Proceedings of 10th World Congress of Genetics Applied to Livestock Production. Vancouver, British Columbia, Canada.

Makowsky, R., N. M. Pajewski, Y. C. Klimentidis, A. I. Vázquez, C. W. Duarte *et al.*, 2011 Beyond missing heritability: prediction of complex traits. PLoS Genet. 7: e1002051.

Martini, J. W. R., V. Wimmer, M. Erbe, and H. Simianer, 2016 Epistasis and covariance: how gene interaction translates into genomic relationship. Theor. Appl. Genet. 10.1007/s00122–016–2675–5.

Matos, C. A. P., C. Ritter, D. Gianola, and D. L. Thomas, 1993 Bayesian analysis of lamb survival using Monte Carlo numerical integration with importance sampling. J. Anim. Sci. 71: 2047–2054.

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.

Mrode, R., 2014 Linear Models for the Prediction of Animal Breeding Values, Ed. 3. CABI, Wallingford.

Ober, U., W. Huang, M. Magwire, M. Schlather, H. Simianer *et al.*, 2015 Accounting for genetic architecture improves sequence based genomic prediction for a *Drosophila* fitness trait. PLoS ONE 10: e0126880.

Okut, H., D. Gianola, G. J. M. Rosa, and K. A. Weigel, 2011 Prediction of body mass index in mice using dense molecular markers and a regularized neural network. Genet. Res. 93: 189–201.

Pérez, P., and G. de los Campos, 2014 Genome-wide regression and prediction with the BGLR statistical package. Genetics 198: 483–495.

Rincent, R., L. Moreau, H. Monod, E. Kuhn, A. E. Melchinger *et al.*, 2014 Recovering power in association mapping panels with variable levels of linkage disequilibrium. Genetics 197: 375–387.

Robinson, G. K., 1991 That BLUP is a good thing: the estimation of random effects. Stat. Sci. 6: 15–32.

Rubin, D. B., 1988 Using the SIR algorithm to simulate posterior distributions, pp. 395–402 in Bayesian Statistics 3, edited by Bernardo, J. M., M. H. DeGroot, D. V. Lindley, and A. F. M. Smith. Oxford University Press, Cambridge, MA.

Ruppert, D., M. P. Wand, and R. J. Carroll, 2003 Semiparametric Regression, Cambridge University Press, New York.

Searle, S. R., 1974 Prediction, mixed models and variance components, Reliability and Biometry, edited by Proschan, F., and R. J. Serfling. Society for Industrial and Applied Mathematics, Philadelphia.

Seber, G. A. F., and A. J. Lee, 2003 Linear Regression Analysis, Wiley-Blackwell, New York.

Smith, A. F. M., and A. E. Gelfand, 1992 Bayesian statistics without tears: a sampling-resampling perspective. Am. Stat. 46: 84–88.

Spiliopoulou, A., R. Nagy, M. L. Bermingham, J. E. Huffman, C. Hayward *et al.*, 2015 Genomic prediction of complex human traits: relatedness, trait architecture and predictive meta-models. Hum. Mol. Genet. 2015: 1–16.

Takezawa, K., 2006 Introduction to Non-Parametric Regression, Wiley-Interscience, Hoboken.

Tusell, L., P. Pérez-Rodríguez, S. Forni, and D. Gianola, 2014 Model averaging for genome-enabled prediction with reproducing kernel Hilbert spaces: a case study with pig litter size and wheat yield. J. Anim. Breed. Genet. 131: 105–115.

Utz, H. F., A. E. Melchingerm, and C. C. Schön, 2000 Bias and sampling error of the estimated proportion of genotypic variance explained by quantitative trait loci determined from experimental data in maize using cross validation and validation with independent samples. Genetics 154: 1839–1849.

Van Raden, P. M., 2008 Efficient methods to compute genomic predictions. J. Dairy Sci. 91: 4414–4423.

Vázquez, A. I., G. de los Campos, Y. C. Klimentidis, G. J. M. Rosa, D. Gianola *et al.*, 2012 A comprehensive genetic approach for improving prediction of skin cancer risk in humans. Genetics 192: 1493–1502.

Vehtari, A., and J. Lampinen, 2002 Bayesian model assessment and comparison using cross-validation predictive densities. Neural Comput. 14: 2439–2468.

Vehtari, A., A. Gelman, and J. Gabry, 2016 Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. arXiv: 1507.04544.

Wimmer, V., C. Lehermeier, T. Albrecht, H. J. Auinger, Y. Wang *et al.*, 2013 Genome-wide prediction of traits with different genetic architecture through efficient variable selection. Genetics 195: 573–587.

Communicating editor: D. J. de Koning

APPENDIX A: PREDICTION WITH LEAST-SQUARES

Following Seber and Lee (2003), consider out-of-sample prediction, and suppose that the distribution of residuals in left-out data (testing set) of size n_{test} is as in a training set of size n, with the training phenotypes represented as $\mathbf{y}$; residuals in training and testing sets are assumed to be mutually independent. In genome-enabled prediction, (1) is merely an instrumental model that may bear little resemblance with the state of nature, so we let the true distribution be $\mathbf{y}_{test} \sim N(\boldsymbol{\mu}, \mathbf{I}\sigma_e^2)$, allowing for the almost certain possibility that $\boldsymbol{\mu} \neq \mathbf{X}_{test}\boldsymbol{\beta}$, where $\mathbf{X}_{test}$ is the marker matrix in the testing set. In quantitative genetics, $\boldsymbol{\mu}$ is an unknown function of quantitative trait locus (QTL) genotypes and of their effects; the latter may not be additive, and dominance or epistasis may enter into the picture without contributing detectable variance.

Model training yields $\hat{\beta}$, and the point predictor of $\mathbf{y}_{test}$ is $\mathbf{X}_{test}\hat{\beta}$. The mean squared error of prediction (PMSE), conditionally on training data $(\mathbf{y}, \mathbf{X})$ and on $\mathbf{X}_{test}$ but averaged with respect of all possible testing sets of size, n_{test}, is

$$E(\text{PMSE}|\mathbf{y}, \mathbf{X}, \mathbf{X}_{test}) = \frac{1}{n_{test}} E_{\mathbf{y}_{test}|\mathbf{y},\mathbf{X},\mathbf{X}_{test}}\left[\left(\mathbf{y}_{test} - \mathbf{X}_{test}\hat{\beta}\right)'\left(\mathbf{y}_{test} - \mathbf{X}_{test}\hat{\beta}\right)\right]. \tag{63}$$

Define the conditional prediction bias as

$$\hat{\delta} = E_{\mathbf{y}_{test}|\mathbf{y},\mathbf{X},\mathbf{X}_{test}}\left(\mathbf{y}_{test} - \mathbf{X}_{test}\hat{\beta}|\mathbf{y}\right) = \boldsymbol{\mu} - \mathbf{X}_{test}\hat{\beta}. \tag{64}$$

Standard results on expectation of quadratic forms applied to (63) produce

$$E(\text{PMSE}|\mathbf{y}, \mathbf{X}, \mathbf{X}_{test}) = \frac{1}{n_{test}}\left\{(\boldsymbol{\mu} - \mathbf{X}_{test}\hat{\beta})'(\boldsymbol{\mu} - \mathbf{X}_{test}\hat{\beta}) + tr\left[Var\left(\mathbf{y}_{test}\right)\right]\right\} = \frac{1}{n_{test}}\left[\hat{\delta}'\hat{\delta} + n_{test}\sigma_e^2\right]. \tag{65}$$

Unconditionally, that is, by averaging over all possible sets of training (testing) data of size n (n_{test}) with marker configuration $\mathbf{X}$ ($\mathbf{X}_{test}$)

$$E(\text{PMSE}|\mathbf{X}, \mathbf{X}_{test}) = E_{\mathbf{y}|\mathbf{X},\mathbf{X}_{test}}[E(\text{PMSE}|\mathbf{y}, \mathbf{X}, \mathbf{X}_{test})] = \frac{1}{n_{test}}\left[E_{\mathbf{y}|\mathbf{X},\mathbf{X}_{test}}(\hat{\delta}'\hat{\delta}) + n_{test}\sigma_e^2\right], \tag{66}$$

where

$$E_{\mathbf{y}|\mathbf{X},\mathbf{X}_{test}}(\hat{\delta}'\hat{\delta}) = [\boldsymbol{\mu} - \mathbf{X}_{test}E(\hat{\beta})]'[\boldsymbol{\mu} - \mathbf{X}_{test}E(\hat{\beta})] + \sigma_e^2 tr\left[\mathbf{X}_{test}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{test}\right] = \delta'\delta + \sigma_e^2 tr\left[\mathbf{X}_{test}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{test}\right]. \tag{67}$$

Here, the expected prediction bias is

$$\boldsymbol{\delta} = \boldsymbol{\mu} - \mathbf{X}_{test}E(\hat{\beta}) = \left[\mathbf{I} - \mathbf{X}_{test}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right]\boldsymbol{\mu}, \tag{68}$$

after assuming for convenience of representation that $E(\mathbf{y}_{test}) = E(\mathbf{y}) = \boldsymbol{\mu}$, *i.e.*, that testing and training sets have the same size and unknown true mean $\boldsymbol{\mu}$; note that $\frac{\partial \mathbf{X}_{test}\hat{\beta}}{\partial \mathbf{y}'} = \mathbf{X}_{test}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is a matrix of "influences" describing how variation in training phenotypes affects predictions. Using (68) in (67), and this in (66), the expected prediction mean-squared error averaged over an infinite number of training and testing sets, but with fixed marker genotype matrices is

$$E(\text{PMSE}|\mathbf{X}, \mathbf{X}_{test}) = \frac{1}{n_{test}}\left\{\delta'\delta + \sigma_e^2 tr\left[\mathbf{X}_{test}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{test}\right] + n_{test}\sigma_e^2\right\} \tag{69}$$

If $\mathbf{X}_{test} = \mathbf{X}$, that is, in the special case where the same genotypes appear in the training and testing sets, $\mathbf{X}_{test}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'_{test} = \mathbf{H} = \{h_{ij}\}$ and $tr(\mathbf{X}_{test}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'_{test}) = p$, so the preceding equation becomes

$$E(\text{PMSE}|\mathbf{X}) = \frac{1}{n_{test}}\sum_{i=1}^{n}\delta_i^2 + \left(1 + \frac{p}{n_{test}}\right)\sigma_e^2, \tag{70}$$

with

$$\boldsymbol{\delta} = \{\delta_i\} = \left[\mathbf{I} - \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right]\boldsymbol{\mu} = (\mathbf{I} - \mathbf{H})\boldsymbol{\mu}, \tag{71}$$

and typical element $\delta_i = (1 - \sum_{j=1}^{n} h_{ij})\mu_i$. Expression (70) shows that the uncertainty of prediction, as measured by variance (second term) increases with p (model complexity) and decreases with n_{test} (equal to n here). The impact of increasing complexity on bias is difficult to discuss

in the absence of knowledge of QTL and their effects. When $p \rightarrow n$, the training data set fits better and better and the model increasingly copies both signal and error: predictions become increasingly poorer.

Note that

$$Var[(\mathbf{y} - \mathbf{X}\hat{\beta})] = \mathbf{M}\mathbf{M}\sigma_e^2 = (\mathbf{I} - \mathbf{H})\sigma_e^2, \tag{72}$$

where $\mathbf{M} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' = \mathbf{I} - \mathbf{H}$. Applying this result in (8)

$$E_{\mathbf{y}|\mathbf{X}}[\text{PMSE}(1)] = \frac{1}{n}\left[\delta'\mathbf{D}\delta + \sigma_e^2 tr(\mathbf{DM})\right], \tag{73}$$

where $\frac{1}{n}(\delta'\mathbf{D}\delta)$ is the average squared prediction bias, and $\frac{\sigma_e^2}{n} tr[\mathbf{DM}]$ is the average prediction error variance. Note that

$$\delta'\mathbf{D}\boldsymbol{\delta} = \sum_{i=1}^{n} \frac{\delta_i^2}{(1-h_{ii})^2}, \tag{74}$$

and that

$$tr(\mathbf{DM}) = tr(\mathbf{D} - \mathbf{DH}) = \sum_{i=1}^{n}\left(\frac{1}{(1-h_{ii})^2} - \frac{h_{ii}}{(1-h_{ii})^2}\right) = \sum_{i=1}^{n}\left[\frac{1-h_{ii}}{(1-h_{ii})^2}\right]. \tag{75}$$

Employing (74) and (75) in (73),

$$E_{\mathbf{y}|\mathbf{X}}[\text{PMSE}(1)] = \frac{1}{n}\left\{\sum_{i=1}^{n}\left[\frac{\delta_i^2}{(1-h_{ii})^2} + \sigma_e^2\sum_{i=1}^{n}\left(\frac{1-h_{ii}}{(1-h_{ii})^2}\right)\right]\right\} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{\delta_i^2 + \sigma_e^2(1-h_{ii})}{(1-h_{ii})^2}\right]. \tag{76}$$

Examine the difference (Δ_{MSE}) in expected prediction mean-squared error between the LOO procedure, as given in (76), and that from the "distinct testing set" layout given in (70), and set $n = n_{test}$. One obtains

$$\Delta_{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{\delta_i^2 + \sigma_e^2(1-h_{ii})}{(1-h_{ii})^2}\right] - \frac{1}{n}\left[\sum_{i=1}^{n}\delta_i^2 + (n+p)\sigma_e^2\right]. \tag{77}$$

Since $p = tr(\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}') = tr(\mathbf{H}) = \sum_{i=1}^{n} h_{ii}$,

$$\Delta_{MSE} = \frac{1}{n}\left\{\sum_{i=1}^{n}\left[\frac{\delta_i^2 + \sigma_e^2(1-h_{ii})}{(1-h_{ii})^2}\right] - \sum_{i=1}^{n}\left[\delta_i^2 + (1+h_{ii})\sigma_e^2\right]\right\} = \frac{1}{n}\left\{\sigma_e^2\sum_{i=1}^{n}\frac{h_{ii}^2}{1-h_{ii}} + \sum_{i=1}^{n}\frac{\delta_i^2 h_{ii}(2-h_{ii})}{(1-h_{ii})^2}\right\}. \tag{78}$$

The h_{ii} are bounded between 0 and 1, so the two terms in (78) are positive.

APPENDIX B: MATRIX ALGEBRA RESULT

Early derivations of the mixed model equations used for computing best linear unbiased estimation of fixed effects, and BLUP of random effects in mixed linear models used by Henderson's (*e.g.*, Henderson *et al.* 1959; Henderson 1975, 1984) made use of the Sherman-Morrison-Woodbury formula (Seber and Lee 2003). Moore-Penrose generalizations are in Deng (2011).

Assuming that the required inverses stated below exist, the following result holds

$$(\mathbf{A} + \mathbf{UBV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{UB}\left(\mathbf{B} + \mathbf{BVA}^{-1}\mathbf{UB}\right)^{-1}\mathbf{BVA}^{-1}. \tag{79}$$

A special case is when $\mathbf{B} = \mathbf{I}$, $\mathbf{U} = \pm\mathbf{u}$, $\mathbf{V} = \mathbf{v}'$, where $\mathbf{u}$ and $\mathbf{v}'$ denote column and row vectors; here

$$\left(\mathbf{A} + \mathbf{uv}'\right)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{uv}'\mathbf{A}^{-1}}{1 + \mathbf{v}'\mathbf{A}^{-1}\mathbf{u}}, \tag{80}$$

$$\left(\mathbf{A} - \mathbf{uv}'\right)^{-1} = \mathbf{A}^{-1} + \frac{\mathbf{A}^{-1}\mathbf{uv}'\mathbf{A}^{-1}}{1 - \mathbf{v}'\mathbf{A}^{-1}\mathbf{u}}. \tag{81}$$

APPENDIX C: LOO ORDINARY LEAST-SQUARES

In (4), we have

$$\hat{\boldsymbol{\beta}}_{[-i]} = \left(\mathbf{X}'_{[-i]}\mathbf{X}_{[-i]}\right)^{-1}\mathbf{X}'_{[-i]}\mathbf{y}_{[-i]} = \left(\mathbf{X}'\mathbf{X}-\mathbf{x}_i\mathbf{x}'_i\right)^{-1}\left(\mathbf{X}'\mathbf{y}-\mathbf{x}_i y_i\right) = \left[\left(\mathbf{X}'\mathbf{X}\right)^{-1} + \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}'_i\left(\mathbf{X}'\mathbf{X}\right)^{-1}}{1-\mathbf{x}'_i\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i}\right]\left(\mathbf{X}'\mathbf{y}-\mathbf{x}_i y_i\right). \tag{82}$$

Let the $n\times n$ regression "hat matrix" be $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$, with diagonal elements $h_{ii} = \mathbf{x}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i$; $i = 1, 2, \ldots, n$. Hence, (4) can be written as

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{[-i]} &= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i y_i + \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}'_i\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}}{1-h_{ii}} - \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}'_i\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i y_i}{1-h_{ii}} \\
&= \hat{\boldsymbol{\beta}} - \frac{(1-h_{ii})\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i y_i - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}'_i\hat{\boldsymbol{\beta}} + h_{ii}\left(\mathbf{X}'\mathbf{X}\right)_i^{-1}\mathbf{x}y_i}{1-h_{ii}} \\
&= \hat{\boldsymbol{\beta}} - \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\left(y_i-\mathbf{x}'_i\hat{\boldsymbol{\beta}}\right)}{1-h_{ii}} \\
&= \hat{\boldsymbol{\beta}} - \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\hat{e}_i}{1-h_{ii}}.
\end{aligned} \tag{83}$$

Using (83), the error of fitting phenotype y_i is then

$$y_i - \mathbf{x}'_i\hat{\boldsymbol{\beta}}_{[-i]} = y_i - \left[\mathbf{x}'_i\hat{\boldsymbol{\beta}} - \frac{\mathbf{x}'_i\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i\left(y_i-\mathbf{x}'_i\hat{\boldsymbol{\beta}}\right)}{1-h_{ii}}\right] = y_i - \mathbf{x}'_i\hat{\boldsymbol{\beta}} + \frac{h_{ii}\left(y_i-\mathbf{x}'_i\hat{\boldsymbol{\beta}}\right)}{1-h_{ii}} = \frac{y_i-\mathbf{x}'_i\hat{\boldsymbol{\beta}}}{1-h_{ii}}; \quad i = 1, 2, \ldots, n. \tag{84}$$

APPENDIX D: LEAVE-D-OUT ORDINARY LEAST-SQUARES

If d cases are removed from the training data set, $\mathbf{X}'_{[-d]}\mathbf{X}_{[-d]} = \mathbf{X}'\mathbf{X} - \mathbf{X}'_{[d]}\mathbf{X}_{[d]}$. In (79), put $\mathbf{A} = \mathbf{X}'\mathbf{X}$, $\mathbf{U} = \mathbf{X}'_{[d]}$, $\mathbf{B} = -\mathbf{I}$, and $\mathbf{V} = \mathbf{X}_{[d]}$, to obtain the coefficient matrix

$$\left(\mathbf{X}'\mathbf{X}-\mathbf{X}'_{[d]}\mathbf{X}_{[d]}\right)^{-1} = \left(\mathbf{X}'\mathbf{X}\right)^{-1} + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}\left[\mathbf{I}-\mathbf{X}_{[d]}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}\right]^{-1}\mathbf{X}_{[d]}\left(\mathbf{X}'\mathbf{X}\right)^{-1}, \tag{85}$$

for d being one of the $D = \binom{n}{d}$ possible $d-tuplets$. For example, if $n = 1000$, and $d = 100$ or 500, then $D = 6.39\times10^{139}$ and 2.71×10^{299}, respectively. Clearly, it is not feasible to consider all possible configurations of training and testing sets. Note that $[\mathbf{I}-\mathbf{X}_{[d]}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'_{[d]}]^{-1}$ will not exist for all combinations of n and d. For the leave-d-out least-squares estimator given to exist, it is needed that $n-d \geq p$.

The right-hand side vector in least-squares takes the form

$$\mathbf{X}'_{[-d,]}\mathbf{y}_{[-d]} = \sum_{i=1}^{n}\mathbf{x}_i y_i - \sum_{j\in d}\mathbf{x}_j y_d = \mathbf{X}'\mathbf{y} - \mathbf{X}'_{[d]}\mathbf{y}_{[d]}. \tag{86}$$

Let $\mathbf{H}_d = \mathbf{X}_{[d]}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'_{[d]}$ be the $d\times d$ part of the hat matrix $\mathbf{H}$. The OLS estimator, $\hat{\boldsymbol{\beta}}_{[-d]} = (\mathbf{X}'_{[-d,]}\mathbf{X}_{[-d,]})^{-1}\mathbf{X}'_{[-d,]}\mathbf{y}_{[-d]}$, can be written as

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{[-d]} &= \left\{\left(\mathbf{X}'\mathbf{X}\right)^{-1} + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}[\mathbf{I}-\mathbf{H}_d]^{-1}\mathbf{X}_{[d]}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right\}\left(\mathbf{X}'\mathbf{y}-\mathbf{X}'_{[d]}\mathbf{y}_{[d]}\right) \\
&= \hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}\mathbf{y}_{[d]} + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}[\mathbf{I}-\mathbf{H}_d]^{-1}\mathbf{X}_{[d]}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}[\mathbf{I}-\mathbf{H}_d]^{-1}\mathbf{X}_{[d]}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}\mathbf{y}_{[d]}.
\end{aligned} \tag{87}$$

Further,

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{[-d]} &= \hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}\mathbf{y}_{[d]} + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1}\mathbf{X}_{[d]}\hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)_{[d]}^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1}\mathbf{H}_d\mathbf{y}_{[d]} \\
&= \hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1}\left[(\mathbf{I}-\mathbf{H}_d)\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}} + \mathbf{H}_d\mathbf{y}_{[d]}\right] = \hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1}\left(\mathbf{y}_{[d]} - \mathbf{X}_{[d]}\hat{\boldsymbol{\beta}}\right) \\
&= \hat{\boldsymbol{\beta}} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'_{[d]}(\mathbf{I}-\mathbf{H}_d)^{-1}\hat{\mathbf{e}}_{[d]}.
\end{aligned} \tag{88}$$

APPENDIX E: LOO AND LEAVE-D-OUT BLUP OF MARKERS

As indicated in (15), finding BLUP of markers requires computing the coefficient matrix in the corresponding equations, and the vector of right hand sides. Following the reasoning used for least-squares

$$\mathbf{X}'\mathbf{X}+\mathbf{I}\lambda=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i'+\mathbf{I}\lambda \tag{89}$$

If observation $(y_i,\mathbf{x}_i')$ is to be predicted in a LOO CV, $\mathbf{X}_{[-i,]}'\mathbf{X}_{[-i,]}+\mathbf{I}\lambda=\mathbf{X}'\mathbf{X}+\mathbf{I}\lambda-\mathbf{x}_i\mathbf{x}_i'$ and $\mathbf{X}_{[-i,]}'\mathbf{y}_{[-i,]}=\mathbf{X}'\mathbf{y}-\mathbf{x}_i\mathbf{y}_i$, so

$$\begin{aligned}\boldsymbol{\beta}_{[-i]}^r&=\left(\mathbf{X}_{[-i,]}'\mathbf{X}_{[-i,]}+\mathbf{I}\lambda\right)^{-1}\mathbf{X}_{[-i,]}'\mathbf{y}_{[-i,]}=\left(\mathbf{X}'\mathbf{X}+\mathbf{I}\lambda-\mathbf{x}_i\mathbf{x}_i'\right)^{-1}\left(\mathbf{X}'\mathbf{y}-\mathbf{x}_i\mathbf{y}_i\right)\\&=\left[\left(\mathbf{X}'\mathbf{X}+\mathbf{I}\boldsymbol{\lambda}\right)^{-1}+\frac{\left(\mathbf{X}'\mathbf{X}+\mathbf{I}\boldsymbol{\lambda}\right)^{-1}\mathbf{x}_i\mathbf{x}_i'\left(\mathbf{X}'\mathbf{X}+\mathbf{I}\boldsymbol{\lambda}\right)^{-1}}{1-h_{ii}^r}\right]\left(\mathbf{X}'\mathbf{y}-\mathbf{x}_i\mathbf{y}_i\right),\end{aligned}$$

where $h_{ii}^r=\mathbf{x}_i'(\mathbf{X}'\mathbf{X}+\mathbf{I}\boldsymbol{\lambda})^{-1}\mathbf{x}_i$. Letting $\mathbf{C}^{-1}=(\mathbf{X}'\mathbf{X}+\mathbf{I}\boldsymbol{\lambda})^{-1}$, the preceding becomes

$$\begin{aligned}\boldsymbol{\beta}_{[-i]}^r&=\mathbf{C}^{-1}\mathbf{X}'\mathbf{y}-\mathbf{C}^{-1}\mathbf{x}_iy_i+\frac{\mathbf{C}^{-1}\mathbf{x}_i\mathbf{x}_i'\mathbf{C}^{-1}\mathbf{X}'\mathbf{y}}{1-h_{ii}^r}-\frac{\mathbf{C}^{-1}\mathbf{x}_i\mathbf{x}_i'\mathbf{C}^{-1}\mathbf{x}_iy_i}{1-h_{ii}^r}=\boldsymbol{\beta}^r-\mathbf{C}^{-1}\mathbf{x}_iy_i+\frac{\mathbf{C}^{-1}\mathbf{x}_i\mathbf{x}_i'\boldsymbol{\beta}^r}{1-h_{ii}^r}-\frac{\mathbf{C}^{-1}\mathbf{x}_ih_{ii}^ry_i}{1-h_{ii}^r}\\&=\boldsymbol{\beta}^r-\frac{(1-h_{ii}^r)\mathbf{C}^{-1}\mathbf{x}_iy_i-\mathbf{C}^{-1}\mathbf{x}_i\mathbf{x}_i'\boldsymbol{\beta}^r+h_{ii}^r\mathbf{C}^{-1}\mathbf{x}_iy_i}{1-h_{ii}^r}=\boldsymbol{\beta}^r-\frac{\mathbf{C}^{-1}\mathbf{x}_i\left(y_i-\mathbf{x}_i'\boldsymbol{\beta}^r\right)}{1-h_{ii}}=\boldsymbol{\beta}^r-\frac{\mathbf{C}^{-1}\mathbf{x}_i\hat{e}_i^r}{1-h_{ii}},\end{aligned} \tag{90}$$

where $\hat{e}_i^r=y_i-\mathbf{x}_i'\boldsymbol{\beta}^r$ is the residual from the ridge regression-BLUP analysis obtained with the entire sample.

For leave-d-out the algebra is as with least-squares, producing as coefficient matrix

$$\left(\mathbf{X}'\mathbf{X}+\mathbf{I}\lambda-\mathbf{X}_{[d]}'\mathbf{X}_{[d]}\right)^{-1}=\mathbf{C}^{-1}+\mathbf{C}^{-1}\mathbf{X}_{[d]}'\left[\mathbf{I}-\mathbf{X}_{[d]}\mathbf{C}^{-1}\mathbf{X}_{[d]}'\right]^{-1}\mathbf{X}_{[d]}\mathbf{C}^{-1}, \tag{91}$$

and the right-hand sides are as before $\mathbf{X}_{[-d,]}'\mathbf{y}_{[-d]}=\mathbf{X}'\mathbf{y}-\mathbf{X}_{[d]}'\mathbf{y}_{[d]}$. Letting $\mathbf{H}_{[d]}^r=\mathbf{X}_{[d]}\mathbf{C}^{-1}\mathbf{X}_{[d]}'$, algebra similar to the one producing (88) yields

$$\boldsymbol{\beta}_{[-d]}^r=\boldsymbol{\beta}^r-\mathbf{C}^{-1}\mathbf{X}_{[d]}'\left(\mathbf{I}-\mathbf{H}_d^r\right)^{-1}\hat{\mathbf{e}}_{[d]}^r, \tag{92}$$

where $\hat{\mathbf{e}}_{[d]}^r=\mathbf{y}_{[d]}-\mathbf{X}_{[d]}\boldsymbol{\beta}^r$ is the $d-$dimensional residual from ridge regression BLUP applied to the entire sample.

APPENDIX F: LOO AND LEAVE-D-OUT GBLUP

LOO GBLUP

In LOO CV, one seeks to estimate the mean of the predictive distribution of the left-out data point $E(y_i|\mathbf{y}_{-i},H)$. Since $y_i=\mathbf{x}_i'\boldsymbol{\beta}+e_i=g_i+e_i$ and $e_i\sim N(0,\sigma_e^2)$ is independent of $\mathbf{y}_{-i}$ one has

$$E(y_i|\mathbf{y}_{-i},H)=E\left(g_i+e_i|\mathbf{y}_{[-i]},H\right)=E\left(g_i|\mathbf{y}_{[-i]},H\right). \tag{93}$$

More generally, we are interested in predicting all n genetic values $\mathbf{g}=\{g_i\}$. Since our zero-mean BLUP is interpretable as the mean of the conditional or posterior distribution $[\mathbf{g}|\mathbf{y},\boldsymbol{\theta}]$ ($\boldsymbol{\theta}$ denotes σ_g^2 and σ_e^2 here), for $p(.|.)$ denoting a density function, it follows that

$$p(\mathbf{g}|\mathbf{y}_{-i},\boldsymbol{\theta})\propto p(\mathbf{y}_{-i}|\mathbf{g},\boldsymbol{\theta})p(\mathbf{g}|\boldsymbol{\theta})\propto\frac{p(\mathbf{y}|\mathbf{g},\boldsymbol{\theta})}{p(y_i|\mathbf{g},\boldsymbol{\theta})}p(\mathbf{g}|\boldsymbol{\theta}), \tag{94}$$

since $p(\mathbf{y}|\mathbf{g},\boldsymbol{\theta})=p(y_i|\mathbf{g},\boldsymbol{\theta})p(\mathbf{y}_{-i}|\mathbf{g},\boldsymbol{\theta})$, due to independence of residuals. The preceding posterior is Gaussian (given the dispersion parameters) because the prior and the sampling model are both Gaussian; hence, the mode and the median of this distributions are identical. Apart from an additive constant

$$\log\left[p(\mathbf{g}|\mathbf{y}_{-i},\ \boldsymbol{\theta})\right]=\log p(\mathbf{y}|\mathbf{g},\boldsymbol{\theta})+\log p(\mathbf{g}|\boldsymbol{\theta})-\log p(y_i|\mathbf{g},\boldsymbol{\theta}), \tag{95}$$

or, explicitly,

$$\log\left[p(\mathbf{g}|\mathbf{y}_{-i},\boldsymbol{\theta})\right]=-\frac{1}{2\sigma_e^2}(\mathbf{y}-\mathbf{g})'(\mathbf{y}-\mathbf{g})-\frac{1}{2\sigma_g^2}\mathbf{g}'\mathbf{G}^{-1}\mathbf{g}+\frac{1}{2\sigma_e^2}(y_i-g_i)^2=-\frac{1}{2\sigma_e^2}\left[(\mathbf{y}-\mathbf{g})'(\mathbf{y}-\mathbf{g})+\lambda\mathbf{g}'\mathbf{G}^{-1}\mathbf{g}-\left(y_i-\boldsymbol{\delta}_i'\mathbf{g}\right)^2\right] \tag{96}$$

where $\lambda=\frac{\sigma_e^2}{\sigma_g^2}$, and $\boldsymbol{\delta}_i^*$ is a $n\times1$ vector with a 1 in position i and 0's elsewhere. The gradient of the log-posterior with respect to $\mathbf{g}$ is

$$\frac{\partial}{\partial\mathbf{g}}\log\left[p(\mathbf{g}|\mathbf{y}_{-i},\boldsymbol{\theta})\right]=-\frac{1}{2\sigma_e^2}\left[-2(\mathbf{y}-\mathbf{g})+2\lambda\mathbf{G}^{-1}\mathbf{g}+2\boldsymbol{\delta}_i^*\left(y_i-\boldsymbol{\delta}_i^*\mathbf{g}\right)\right]; \tag{97}$$

note that $\boldsymbol{\delta}_i^{*'}\mathbf{g}=g_i$. Setting to zero yields the joint mode, *i.e.*, the *BLUP* of all n lines (but using $\mathbf{y}_{-i}$) as the solution to the linear system

$$(\mathbf{C}-\boldsymbol{\Delta}_i)\tilde{\mathbf{g}}_{[-i]}=\mathbf{y}-\boldsymbol{\delta}_i^*y_i, \tag{98}$$

where $\tilde{\mathbf{g}}_{[-i]}$ $(n\times 1)$ is the BLUP of $\mathbf{g}$ calculated with all observations other than i; $\mathbf{C}=\mathbf{I}+\mathbf{G}^{-1}\lambda$ is the coefficient matrix used for calculating $BLUP(\mathbf{g})=\hat{\mathbf{g}}=\{\hat{g}_i\}=\mathbf{C}^{-1}\mathbf{y}$ using all data points; $\boldsymbol{\delta}_i$ is as defined earlier, and $\boldsymbol{\Delta}_i=\boldsymbol{\delta}_i^*\boldsymbol{\delta}_i^{*'}$ is an $n\times n$ matrix of $0's$ except for a 1 in position (i,i). Observe that

$$\tilde{\mathbf{g}}_{[-i]}=\mathbf{C}^{-1}\mathbf{y}-\mathbf{C}^{-1}\boldsymbol{\delta}_i^*y_i+\mathbf{C}^{-1}\boldsymbol{\Delta}_i\tilde{\mathbf{g}}_{[-i]}=\hat{\mathbf{g}}-\mathbf{C}^{-1}\boldsymbol{\delta}_i^*y_i+\mathbf{C}^{-1}\boldsymbol{\Delta}_i\tilde{\mathbf{g}}_{[-i]}. \tag{99}$$

Further,

$$\mathbf{C}^{-1}\boldsymbol{\delta}_i^*y_i=\mathbf{c}^iy_i,\mathbf{C}^{-1}\boldsymbol{\Delta}_i\tilde{\mathbf{g}}_{[-i]}=\mathbf{c}^i\tilde{g}_i, \tag{100}$$

where $\mathbf{c}^i$ is the i^{th} column of $\mathbf{C}^{-1}=\{c^{ij}\}$, and $\tilde{g}_i$ is the i^{th} element of $\tilde{\mathbf{g}}_{[-i]}$. Thus

$$\tilde{\mathbf{g}}_{[-i]}-\mathbf{c}^i\tilde{g}_i=\hat{\mathbf{g}}-\mathbf{c}^iy_i. \tag{101}$$

Inspection of the preceding expression reveals that element i of $\tilde{\mathbf{g}}_{[-i]}$, that is, the genetic value of the observation left out in the LOO CV can be computed as

$$\tilde{g}_i=\frac{1}{1-c^{ii}}\left(\hat{g}_i-c^{ii}y_i\right);\quad i=1,2,\ldots,n. \tag{102}$$

If desired, the remaining $n-1$ elements of $\tilde{\mathbf{g}}_{[-i]}$ can be calculated recursively as

$$\tilde{g}_j=\hat{g}_j-c^{ji}\left(y_i-\tilde{g}_i\right);\quad j\neq i. \tag{103}$$

The observed predictive MSE is given by

$$PMSE(1)=\frac{1}{n}\sum_{i=1}^{n}\left[y_i-\frac{1}{1-c^{ii}}\left(\hat{g}_i-c^{ii}y_i\right)\right]^2=\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i-\hat{g}_i}{1-c^{ii}}\right)^2. \tag{104}$$

Leave-d-Out GBLUP

Consider next a leave-d-out setting. Assume that observations have been reordered such that phenotypes of members of the testing set are placed in the upper d positions in $\mathbf{y}$, so $\mathbf{y}'=[\mathbf{y}_d,\mathbf{y}_{[-d]}]$ and $\mathbf{g}'=[\mathbf{g}_d,\mathbf{g}_{[-d]}]$ Thus

$$\begin{aligned}\log\left[p\left(\mathbf{g}|\mathbf{y}_{[-d]},\boldsymbol{\theta}\right)\right]&=-\frac{1}{2\sigma_e^2}(\mathbf{y}-\mathbf{g})'(\mathbf{y}-\mathbf{g})-\frac{1}{2\sigma_g^2}\mathbf{g}'\mathbf{G}^{-1}\mathbf{g}+\frac{1}{2\sigma_e^2}(\mathbf{y}_d-\mathbf{g}_d)'(\mathbf{y}_d-\mathbf{g}_d)\\&=-\frac{1}{2\sigma_e^2}\left[(\mathbf{y}-\mathbf{g})'(\mathbf{y}-\mathbf{g})+\lambda\mathbf{g}'\mathbf{G}^{-1}\mathbf{g}-(\mathbf{y}_d-\boldsymbol{\Delta}_d\mathbf{g})'(\mathbf{y}_d-\boldsymbol{\Delta}_d\mathbf{g})\right],\end{aligned} \tag{105}$$

where $\boldsymbol{\Delta}_d=[\,\mathbf{I}_d\quad 0_{d,n-d}\,]$ is a $d\times n$ matrix partitioned into an identity $d\times d$ submatrix, and with a 0 as each element of the $d\times(n-d)$ partition. Hence

$$\frac{\partial}{\partial\mathbf{g}}\log\left[p\left(\mathbf{g}|\mathbf{y}_{[-d]},\boldsymbol{\theta}\right)\right]=-\frac{1}{2\sigma_e^2}\left[-2(\mathbf{y}-\mathbf{g})+2\lambda\mathbf{G}^{-1}\mathbf{g}+2\boldsymbol{\Delta}_d'\left(\mathbf{y}_d-\boldsymbol{\Delta}_d\mathbf{g}\right)\right] \tag{106}$$

Setting the vector of differentials to $\mathbf{0}$, and rearranging

$$\tilde{\mathbf{g}}_{[-d]}=\left(\mathbf{C}-\boldsymbol{\Delta}_d'\boldsymbol{\Delta}_d\right)^{-1}\left(\mathbf{y}-\boldsymbol{\Delta}_d'\mathbf{y}_d\right)=\left(\mathbf{C}-\boldsymbol{\Delta}_d'\boldsymbol{\Delta}_d\right)^{-1}\tilde{\mathbf{y}}_d, \tag{107}$$

where $\tilde{\mathbf{y}}_d$ is $\mathbf{y}$ with the d phenotypes in the testing set replaced by $0's$. Application of (79) produces

$$\left(\mathbf{C}-\boldsymbol{\Delta}_d'\boldsymbol{\Delta}_d\right)^{-1}=\mathbf{C}^{-1}-\mathbf{C}^{-1}\boldsymbol{\Delta}_d'\left(\mathbf{I}_d-\boldsymbol{\Delta}_d\mathbf{C}^{-1}\boldsymbol{\Delta}_d'\right)\boldsymbol{\Delta}_d\mathbf{C}^{-1} \tag{108}$$

Let now the inverse of the coefficient matrix for computing the BLUP of all individuals, or lines from all data points in $\mathbf{y}$, be partitioned as

$$\mathbf{C}^{-1}=\left(\mathbf{I}+\mathbf{G}^{-1}\lambda\right)^{-1}=\begin{bmatrix}\mathbf{C}^{dd} & \mathbf{C}^{d,-d}\\ \mathbf{C}^{-d,d} & \mathbf{C}^{-d,-d}\end{bmatrix}. \tag{109}$$

Then

$$\mathbf{C}^{-1}\boldsymbol{\Delta}_d'=\begin{bmatrix}\mathbf{C}^{dd} & \mathbf{C}^{d,-d}\\ \mathbf{C}^{-d,d} & \mathbf{C}^{-d,-d}\end{bmatrix}\begin{bmatrix}\mathbf{I}\\ \mathbf{0}_{n-d,d}\end{bmatrix}=\begin{bmatrix}\mathbf{C}^{dd}\\ \mathbf{C}^{-d,d}\end{bmatrix},\boldsymbol{\Delta}_d\mathbf{C}^{-1}\boldsymbol{\Delta}_d'=\begin{bmatrix}\mathbf{I}_d & \mathbf{0}_{d,n-d}\end{bmatrix}\begin{bmatrix}\mathbf{C}^{dd} & \mathbf{C}^{d,-d}\\ \mathbf{C}^{-d,d} & \mathbf{C}^{-d,-d}\end{bmatrix}\begin{bmatrix}\mathbf{I}\\ \mathbf{0}_{n-d,d}\end{bmatrix} \tag{110}$$

$$= \begin{bmatrix} \mathbf{I}_d & \mathbf{0}_{d,n-d} \end{bmatrix} \begin{bmatrix} \mathbf{C}^{dd} \\ \mathbf{C}^{-d,d} \end{bmatrix} = \mathbf{C}^{dd}, \tag{111}$$

$$\boldsymbol{\Delta}_d \mathbf{C}^{-1} = \begin{bmatrix} \mathbf{C}^{dd} & \mathbf{C}^{d,-d} \end{bmatrix} \tag{112}$$

Inspection of the form of (107) leads as BLUP predictor of phenotypes of individuals in the testing set to

$$\tilde{\mathbf{g}}_d = \left(\mathbf{I} - \mathbf{C}^{dd}\right)^{-1} \left(\hat{\mathbf{g}}_d - \mathbf{C}^{dd}\mathbf{y}_d\right), \tag{113}$$

with CV prediction error

$$\tilde{\boldsymbol{\epsilon}}_d = \mathbf{y}_d - \tilde{\mathbf{g}}_d = \left(\mathbf{I} - \mathbf{C}^{dd}\right)^{-1} (\mathbf{y}_d - \hat{\mathbf{g}}_d),$$

and realized predictive mean-squared error

$$\frac{\text{PMSE}(d) = \tilde{\boldsymbol{\epsilon}}_d^{'} \tilde{\boldsymbol{\epsilon}}_d}{d.} \tag{114}$$

If desired, the BLUP predictors of the remaining $n - d$ genotypes (based on $\mathbf{y}_{[-d]}$) can be calculated as

$$\tilde{\mathbf{g}}_{[-d]} = \hat{\mathbf{g}}_{[-d]} - \mathbf{C}^{-d,d}(\mathbf{y}_d - \tilde{\mathbf{g}}_d) \tag{115}$$

All preceding developments rest on assuming that exclusion of d observations does not modify λ in training sets in an appreciable manner.

APPENDIX G: CROSS-VALIDATION IMPORTANCE SAMPLING

The posterior expectation of the vector of marker effects is

$$E(\boldsymbol{\beta}|\mathbf{y}_{-i}, H) = \frac{\int \boldsymbol{\beta} p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H) d\boldsymbol{\beta}}{\int p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H) d\boldsymbol{\beta}} = \frac{\int \boldsymbol{\beta} \dfrac{p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H)}{p(\boldsymbol{\beta}|\mathbf{y}, H)} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}}{\int \dfrac{p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H)}{p(\boldsymbol{\beta}|\mathbf{y}, H)} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}} \tag{116}$$

$$= \frac{\int w_i(\boldsymbol{\beta}) \boldsymbol{\beta} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}}{\int w_i(\boldsymbol{\beta}) p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}} = \frac{E_{\boldsymbol{\beta}|\mathbf{y},H}[w_i(\boldsymbol{\beta})\boldsymbol{\beta}]}{E_{\boldsymbol{\beta}|\mathbf{y},H}[w_i(\boldsymbol{\beta})]}, i = 1, 2, \ldots, n. \tag{117}$$

Here, $w_i(\boldsymbol{\beta}) = \dfrac{p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H)}{p(\boldsymbol{\beta}|\mathbf{y}, H)}$ is an "importance sampling" weight (Albert, 2009). Bayes theorem yields

$$w_i(\boldsymbol{\beta}) = \frac{p(\boldsymbol{\beta}|\mathbf{y}_{-i}, H)}{p(\boldsymbol{\beta}|\mathbf{y}, H)} = \frac{\dfrac{p(\mathbf{y}_{-i}|\boldsymbol{\beta}, \sigma_e^2) p(\boldsymbol{\beta}|H)}{p(\mathbf{y}_{-i}|H)}}{\dfrac{p(\mathbf{y}|\boldsymbol{\beta}, \sigma_e^2) p(\boldsymbol{\beta}|H)}{p(\mathbf{y}|H)}} = \frac{p(\mathbf{y}|H)}{p(\mathbf{y}_{-i}|H)} \frac{1}{p(y_i|\boldsymbol{\beta}, \sigma_e^2)}, \tag{118}$$

and employing this form of $w_i(\boldsymbol{\beta})$ in (117) produces

$$E(\boldsymbol{\beta}|\mathbf{y}_{-i}, H) = \frac{\int \dfrac{p(\mathbf{y}|H)}{p(\mathbf{y}_{-i}|H)} \dfrac{1}{p(y_i|\boldsymbol{\beta}, \sigma_e^2)} \boldsymbol{\beta} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}}{\int \dfrac{p(\mathbf{y}|H)}{p(\mathbf{y}_{-i}|H)} \dfrac{1}{p(y_i|\boldsymbol{\beta}, \sigma_e^2)} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}} = \frac{\int \dfrac{1}{p(y_i|\boldsymbol{\beta}, \sigma_e^2)} \boldsymbol{\beta} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}}{\int \dfrac{1}{p(y_i|\boldsymbol{\beta}, \sigma_e^2)} p(\boldsymbol{\beta}|\mathbf{y}, H) d\boldsymbol{\beta}}. \tag{119}$$

The implication is that, given draws $\boldsymbol{\beta}^{(s)}$, $\sigma_e^{2(s)}$ $(s = 1, 2, \ldots, S)$ from the full-posterior distribution, the posterior expectation (119) can be estimated as

$$\hat{E}(\boldsymbol{\beta}|\mathbf{y}_{-i}, H) = \frac{\sum_{s=1}^{S} \frac{1}{p\left(y_i|\boldsymbol{\beta}^{(s)}, \sigma_e^{2(s)}\right)} \boldsymbol{\beta}^{(s)}}{\sum_{s=1}^{S} \frac{1}{p\left(y_i|\boldsymbol{\beta}^{(s)}, \sigma_e^{2(s)}\right)}} = \sum_{s=1}^{S} w_{i,s} \boldsymbol{\beta}^{(s)}; \quad i = 1, 2, \ldots, n, \tag{120}$$

where

$$w_{i,s} = \frac{p^{-1}\left(y_i|\boldsymbol{\beta}^{(s)}, \sigma_e^{2(s)}\right)}{\sum_{s=1}^{S} p^{-1}\left(y_i|\boldsymbol{\beta}^{(s)}, \sigma_e^{2(s)}\right)}; \quad i = 1, 2, \ldots, n; \quad s = 1, 2, \ldots, S. \tag{121}$$