

Pitfalls and Remedies for Cross Validation with Multi-trait Genomic Prediction Methods

Daniel Runcie^{*,1} and Hao Cheng[†]

^{*}Department of Plant Sciences, University of California Davis, Davis, CA, USA, [†]Department of Animal Science, University of California Davis, Davis, CA, USA

ABSTRACT Incorporating measurements on correlated traits into genomic prediction models can increase prediction accuracy and selection gain. However, multi-trait genomic prediction models are complex and prone to overfitting which may result in a loss of prediction accuracy relative to single-trait genomic prediction. Cross-validation is considered the gold standard method for selecting and tuning models for genomic prediction in both plant and animal breeding. When used appropriately, cross-validation gives an accurate estimate of the prediction accuracy of a genomic prediction model, and can effectively choose among disparate models based on their expected performance in real data. However, we show that a naive cross-validation strategy applied to the multi-trait prediction problem can be severely biased and lead to sub-optimal choices between single and multi-trait models when secondary traits are used to aid in the prediction of focal traits and these secondary traits are measured on the individuals to be tested. We use simulations to demonstrate the extent of the problem and propose three partial solutions: 1) a parametric solution from selection index theory, 2) a semi-parametric method for correcting the cross-validation estimates of prediction accuracy, and 3) a fully non-parametric method which we call CV2*: validating model predictions against focal trait measurements from genetically related individuals. The current excitement over high-throughput phenotyping suggests that more comprehensive phenotype measurements will be useful for accelerating breeding programs. Using an appropriate cross-validation strategy should more reliably determine if and when combining information across multiple traits is useful.

KEYWORDS

Cross Validation
Genomic Prediction
Linear Model
Mixed Model
multi-trait

1

2 INTRODUCTION

3 Genomic Selection (GS) aims to increase the speed and accuracy of
4 selection in breeding programs by predicting the genetic worth of
5 candidate individuals or lines earlier in the selection process, or
6 for individuals that cannot be directly phenotyped ([Meuwissen](#)
7

8 *et al.* 2001; [Hayes et al.](#) 2009; [Crossa et al.](#) 2017). Genomic selec-
9 tion works by training statistical or Machine Learning models on
10 a set of completely phenotyped and genotyped individuals, and
11 then using the trained model to predict the genetic worth of un-
12 measured individuals. If the predictions are reasonably accurate,
13 selection intensity can be increased either because the population
14 size of candidate individuals is larger or their true genetic worth is
15 estimated more accurately.

16 Predictions of genetic values are usually based only on the geno-

Manuscript compiled: Friday 2nd August, 2019

¹Corresponding author: One Shield Ave, Davis, CA 95616 deruncie@ucdavis.edu

17 types or pedigrees of the new individuals. However predictions
18 can in some cases be improved by including measurements of
19 “secondary” traits that may not be of direct interest but are easier or
20 faster to measure (Thompson and Meyer 1986; Pszczola *et al.* 2013;
21 Lado *et al.* 2018). This is one goal of multi-trait genomic prediction.
22 Multi-trait prediction is most useful for increasing the accuracy of
23 selection on a single focal trait when that trait has low heritability,
24 the “secondary” traits have high heritability, and the genetic and
25 non-genetic correlations between the traits are large and opposing
26 (Thompson and Meyer 1986; Jia and Jannink 2012; Cheng *et al.*
27 2018). With the advent of cheap high-throughput phenotyping,
28 there is great interest in using measurements of early-life or easily
29 accessible traits to improve prediction of later-life or more expen-
30 sive traits, and multi-trait prediction models are attractive methods
31 for leveraging this information (Pszczola *et al.* 2013; Rutkoski *et al.*
32 2016; Fernandes *et al.* 2017; Lado *et al.* 2018).

33 A large number of genomic prediction methods are available,
34 and the best model varies across systems and traits (Heslot *et al.*
35 2012; de Los Campos *et al.* 2013). Due to their complexity and often
36 high-dimensional nature, genomic prediction methods are prone
37 to overfitting and require regularization to perform well on new
38 data. Therefore, comparing models based on their ability to fit
39 existing data (ex. with R^2) is unreliable; every candidate model
40 could explain 100% of the variation in a typical-size dataset.

41 Instead, prediction models are generally compared by cross-
42 validation (Meuwissen *et al.* 2001; Utz *et al.* 2000; Gianola and
43 Schon 2016). The basic idea of cross-validation is to separate the
44 model fitting and tuning process from the model evaluation pro-
45 cess by using separate datasets for each (Hastie *et al.* 2009). This
46 penalizes models that fit too closely to one data set at the expense
47 of generalization. In this way, cross-validation is meant to accu-
48 rately simulate the real-world usage of the model: predicting the
49 genetic values of un-phenotyped individuals; i.e. those not avail-
50 able during the model fitting process itself. Rather than requiring
51 new data *per se*, cross-validation works by splitting an existing
52 dataset into non-overlapping “training” and “testing” partitions,
53 fitting the candidate model to the former, and then evaluating it on
54 its accuracy at predicting the latter. Common measures of accuracy
55 include Pearson’s ρ or the square root of the average squared error
56 (RMSE) (Daetwyler *et al.* 2013). This process of splitting, training,
57 and predicting is typically repeated several times on the same
58 dataset to get a combined or averaged measure of accuracy across

59 different random partitions of the data.

Estimates of model accuracy by cross-validation are not perfect
(Hastie *et al.* 2009). They are subject to sampling error as are any
other statistic. They are also typically downwardly biased because
smaller training datasets are used for the cross-validation than in
the actually application of a model. However in typical cases, this
downward bias is the same for competing models and thus does
not impact model choice (Hothorn *et al.* 2005).

However, cross-validation can give upwardly biased estimates
of model accuracy when misused due to various forms of “data-
leakage” between the training and testing datasets, leading to
overly optimistic estimates of model performance (Kaufman *et al.*
2012). Several potential mistakes in cross-validation experiments
are well known:

- **Biased testing data selection.** The individuals in the model testing partitions should have the same distribution of genetic (and environmental) relatedness to the training population as individuals in the remaining target population (Amer and Banos 2010; Daetwyler *et al.* 2013). For example, if siblings or clones are present in the data, they should not be split between testing and training partitions unless siblings or clones of individuals in the training partition are also at the same frequency in the target population. Similarly, if the goal is to predict into a diverse breeding population, the cross-validation should not be performed only within one F2 mapping population.
- **Overlap between the testing and training datasets.** The observations used as testing data should be kept separate from the training data at all stages of the cross-validation procedure. For example, if data from individuals in the testing dataset are used to calculate estimated genetic values (EBVs) for model training, then the testing and training datasets are overlapping, even if the testing individuals themselves are excluded from model training (Amer and Banos 2010).
- **Pre-selection of features (e.g. markers) based on the full dataset before cross-validation.** All aspects of model specification and training that rely on the observed phenotypes should be performed only on the training partitions, without respect to the testing partition. For example, if a large number of candidate markers are available but only a portion will be included in the final model, the selection of markers (i.e. features) should be done using only the training partition of phenotypes and the selection itself should be repeated

101 each replicate of the cross-validation on each new training 142
102 dataset. If the feature selection is only done once on the whole 143
103 dataset before cross-validation begins, this can lead to biased 144
104 estimates of model accuracy (Hastie *et al.* 2009). 145

105 If these mistakes are avoided, cross-validation generally works 146
106 well for comparing among single-trait methods, and in some cases 147
107 for multi-trait methods. However, our goal in this paper is to 148
108 highlight a challenge with using cross-validation to choose be- 149
109 tween single-trait methods and multi-trait methods; specifically 150
110 multi-trait methods that use information from “secondary” traits 151
111 *measured on the target individuals* to inform the prediction of their 152
112 focal trait(s). In this case, standard cross-validation approaches 153
113 lead to biased results. As we discuss below, the source of bias is 154
114 not data leakage between the training and testing data *per se*, but 155
115 correlated errors with respect to the true genetic merit between 156
116 the secondary traits in the training data and the focal trait in the 157
117 testing data. Note that this issue only occurs when the multiple 158
118 traits are measured on the same individuals, and the traits share 159
119 non-genetic covariance. When traits are measured on different 160
120 individuals, the standard cross-validation approach is appropriate. 161

121 In the following sections, we first describe the opportunity of- 162
122 fered by multi-trait genomic prediction models in this setting, and 163
123 the challenge in evaluating them. We then develop a simulation 164
124 study that highlights the extent of the problem. Next, we pro- 165
125 pose three partial solutions that lead to fairly consistent model 166
126 selections between single and multi-trait models under certain sit- 167
127 uations. Finally, we draw conclusions on when this issue is likely 168
128 to arise and when it can be safely ignored. 169

129 GENERAL SETTING

130 Multi-trait genomic prediction is useful in two general settings: 172
131 1) When the overall value of an individual depends on each trait 173
132 simultaneously (ex. fruit number and fruit size) and these traits 174
133 are correlated, and 2) When a focal trait is difficult or expensive to 175
134 measure on every individual, but other correlated traits are more 176
135 readily available (Thompson and Meyer 1986; Pszczola *et al.* 2013;
136 Lado *et al.* 2018). While multi-trait models are clearly necessary
137 in the first setting, in the second the value of the secondary traits
138 depends on several factors including i) the repeatability of the focal
139 and secondary traits, ii) the correlations among the traits and the
140 *cause* of the correlations (i.e. genetic vs non-genetic), and iii) the
141 relative expenses of collecting data on each trait.

142 Here we focus on the goal of predicting a single focal trait using
143 information from both genetic markers (or pedigrees) and pheno-
144 typic information on other traits. Even within this context, there
145 are also two distinct prediction settings: 1) Predicting the focal trait
146 value for new individuals that are yet to be phenotyped for any
147 of the traits, and 2) Predicting the focal trait value for individuals
148 that have been partially phenotyped; phenotypic values for the
149 secondary traits are known and we wish to predict the individual’s
150 genetic value for the focal trait. These settings were described by
151 (Burgueño *et al.* 2012) as CV1 and CV2, respectively, although those
152 authors focused on multi-environment trials rather than single ex-
153 periments with multiple traits per individual. The same naming
154 scheme has since been extended to the more general multiple-trait
155 prediction scenarios (Lado *et al.* 2018).

156 The key difference between CV1 and CV2-style multi-trait pre-
157 diction is that in the former, the secondary traits help refine esti-
158 mates of the genetic values of *relatives* of the individuals we wish to
159 predict, while in the latter, the secondary traits provide information
160 directly about the genetics of the target individuals themselves.
161 This direct information on the target individuals is generally useful
162 (as we demonstrate below). However, it comes with a cost for the
163 *evaluation* of prediction accuracy by cross-validation. Since we do
164 not know the true genetic values for the testing individuals, we
165 must either use a model to estimate the genetic values or simply
166 use their phenotypic value as a proxy. Unfortunately, if we use
167 our genetic model to estimate these values, we are breaking the
168 independence between the testing and training data, and therefore
169 have biased estimates of cross-validation accuracy. On the other
170 hand, if we simply use the phenotypic values of the focal trait
171 as our predictand, these may be biased towards or away from
172 the true genetic values depending on the non-genetic correlation
173 between the focal and secondary traits. This leads to either over-
174 or under-estimation of the prediction accuracy of our multi-trait
175 models. In realistic scenarios, this can lead users to select worse
176 models.

177 MATERIALS AND METHODS

178 We used a simulation study to explore conditions when naive cross-
179 validation experiments as described above lead to sub-optimal
180 choices between single and multi-trait genomic prediction meth-
181 ods. Our simulations were designed to mimic the process of using
182 cross-validation to compare single and multi-trait models based

183 on their prediction accuracies. We repeated this simulation across
 184 scenarios with different genetic architectures for two traits: a single
 185 “focal” trait and a single “secondary” trait. Specifically, we modified
 186 the heritability and correlation structure of the two traits. These
 187 are the most important parameters for determining the relative
 188 efficiencies of single- and multi-trait prediction models (Thompson
 189 and Meyer 1986). Sample size and level of genomic relatedness
 190 will also affect the comparisons, but are likely to only quantita-
 191 tively (but not qualitatively) change the relative performances of
 192 the models and the accuracy of cross-validation.

To make our simulations realistic, we based them on genomic marker data from 803 lines from a real wheat breeding program (Lopez-Cruz *et al.* 2015). We downloaded the genomic relationship matrix \mathbf{K} based on 14,217 GBS markers from this population. We used this relationship matrix to generate a set of simulated datasets covering all combinations of the following parameters: the relative proportions of genetic and non-genetic variation for each trait ($h^2 = \{0.2, 0.6\}$), and the genetic and non-genetic correlations between the traits $\rho_g = \{0, 0.3, 0.6\}$, $\rho_R = \{-0.6, -0.4, -0.2, 0, 0.2, 0.4, 0.6\}$, drawing trait values for each simulation from multivariate normal distributions. In particular, we set:

$$\mathbf{Y} = \mathbf{U} + \mathbf{E}, \quad \mathbf{U} \sim \text{MN}(\mathbf{0}, \mathbf{K}, \mathbf{G}), \quad \mathbf{E} \sim \text{MN}(\mathbf{0}, \mathbf{I}_n, \mathbf{R})$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_{11} & \mathbf{g}_{12} \\ \mathbf{g}_{21} & \mathbf{g}_{22} \end{bmatrix} = \begin{bmatrix} h_1^2 & \rho_g h_1 h_2 \\ \mathbf{g}_{12} & h_2^2 \end{bmatrix} \quad (1)$$

$$\mathbf{R} = \begin{bmatrix} \mathbf{r}_{11} & \mathbf{r}_{12} \\ \mathbf{r}_{21} & \mathbf{r}_{22} \end{bmatrix} = \begin{bmatrix} (1 - h_1^2) & \rho_R \sqrt{(1 - h_1^2)(1 - h_2^2)} \\ \mathbf{r}_{12} & (1 - h_2^2) \end{bmatrix}$$

193 where $\text{MN}(\cdot)$ is the Matrix normal distribution, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2]$ are
 194 the phenotypic values for the two traits in the n individuals, $\mathbf{U} =$
 195 $[\mathbf{u}_1, \mathbf{u}_2]$ are the true genetic values for the two traits, and $\mathbf{E} =$
 196 $[\mathbf{e}_1, \mathbf{e}_2]$ are the true non-genetic deviations for the two traits. We
 197 repeated this process 500 times for each of the 42 combinations of
 198 the genetic architecture parameters. To improve the consistency of
 199 the simulations, we used the same draws from a standard-normal
 200 distribution for all 42 parameter combinations, but new draws for
 201 each of the 500 simulations.

After creating the 803 simulated individuals, we randomly divided them into a training partition and a testing partition. We arranged the rows of \mathbf{Y} so that the testing individuals were first,

and correspondingly partitioned \mathbf{K} into:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{nn} & \mathbf{K}_{no} \\ \mathbf{K}_{on} & \mathbf{K}_{oo} \end{bmatrix}. \quad (2)$$

202 Here and below, the subscript n refers to the testing partition (i.e.
 203 “new” individuals) and the subscript o refers to the training parti-
 204 tion (i.e. “old” individuals). We use the hat symbol ($\hat{\cdot}$) to denote
 205 parameter estimates or predictions.

206 We then fit single- and multi-trait linear mixed models to the
 207 training data and used these model fits to predict the genetic values
 208 for the focal trait (trait 1) in the testing partition.

Specifically, for the single-trait method we fit a univariate linear mixed model to the training data \mathbf{y}_{o1} :

$$\mathbf{y}_{o1} = \mu_1 + \mathbf{u}_{o1} + \mathbf{e}_{o1}, \quad \mathbf{u}_{o1} \sim \text{N}(0, \mathbf{g}_{11} \mathbf{K}_{oo}), \quad \mathbf{e}_{o1} \sim \text{N}(0, \mathbf{r}_{11} \mathbf{I}_{n_o}) \quad (3)$$

by Restricted Maximum Likelihood using the `remlmer` function of R package (Ziyatdinov *et al.* 2018) and extracted the BLUPs $\hat{\mathbf{u}}_{o1}$. Note: an expanded version of these derivations are provided in the Appendix. We then calculated predicted genetic values for the testing partition \mathbf{u}_{n1} as:

$$\hat{\mathbf{u}}_{n1}^{(1)} | \hat{\mathbf{u}}_{o1} = \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \hat{\mathbf{u}}_{o1}. \quad (4)$$

For the multi-trait model, we stacked the vectors of the two traits in the training dataset into the vector $\mathbf{y}_o = \begin{bmatrix} \mathbf{y}_{o1} \\ \mathbf{y}_{o2} \end{bmatrix}$ and fit:

$$\mathbf{y}_o = \boldsymbol{\mu} + \mathbf{u}_o + \mathbf{e}_o, \quad \mathbf{u}_o \sim \text{N}(\mathbf{0}, \mathbf{G} \otimes \mathbf{K}_{oo}), \quad \mathbf{e}_o \sim \text{N}(\mathbf{0}, \mathbf{R} \otimes \mathbf{I}_{n_o}) \quad (5)$$

209 using the `remlmer` function, extracted estimates $\hat{\boldsymbol{\mu}} = [\hat{\mu}_1^\top, \hat{\mu}_2^\top]^\top$,
 210 $\hat{\mathbf{G}}$, $\hat{\mathbf{R}}$, and BLUPs $\hat{\mathbf{u}}_o$.

To make predictions of the genetic values for the focal trait in the testing partition in the CV1 case without use of \mathbf{y}_{n2} , we calculated:

$$\hat{\mathbf{u}}_{n1}^{(2)} | \hat{\mathbf{u}}_{o1} = \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \hat{\mathbf{u}}_{o1} \quad (6)$$

211 which has the same form as for the single trait model, but the input
 212 BLUPs $\hat{\mathbf{u}}_{o1}$ are different.

To make predictions of the genetic values for the focal trait in the testing partition in the CV2 case, using the phenotypic observations of the secondary trait \mathbf{y}_{n2} , we used a two step method. First, we

estimated $\hat{\mathbf{u}}_o$ above based on both traits in the training data. Then we combined these estimates with the observed phenotypes of the testing data to calculate genetic predictions for the testing data:

$$\hat{\mathbf{u}}_{n1}^{(3)} | \mathbf{y}_{n2}, \hat{\mathbf{u}}_o = \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \hat{\mathbf{u}}_{o1} + \hat{\mathbf{g}}_{12} (\mathbf{K}^{-1})_{nn} (\hat{\mathbf{V}}_c)^{-1} (\mathbf{y}_{n2} - \hat{\boldsymbol{\mu}}_2 - \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \hat{\mathbf{u}}_{o2}), \quad (7)$$

where $\hat{\mathbf{V}}_c = \hat{\mathbf{g}}_{22} (\mathbf{K}^{-1})_{nn} + \hat{\mathbf{r}}_{22} \mathbf{I}_n$. This two-step method will be slightly less accurate than a one-step method that used \mathbf{y}_{n2} during the estimation of $\hat{\mathbf{u}}_o$, but is much easier to implement in breeding programs because no genotype or phenotype data of the evaluation individuals is needed during the model training stage.

We measured the accuracy of these three predictions by calculating the correlation between the prediction $\hat{\mathbf{u}}_{n1}^{(i)}$ and three predictands over the 500 simulations:

- \mathbf{u}_{n1} : The true genetic value.
- \mathbf{y}_{n1} : The phenotypic values of the testing individuals.
- $\hat{\mathbf{u}}_{n1}$: The estimated genetic values of the validation individuals using the full dataset (including \mathbf{y}_{n1}).

For the second accuracy measure that uses phenotypic values as predictands, we “corrected” the correlations by dividing by the true value of $\sqrt{h^2}$ to account for the larger variance of \mathbf{y}_{n1} relative to \mathbf{u}_{n1} . This impacts the denominator of the correlation (Daetwyler *et al.* 2013), but since it is the same across methods, does not impact their comparison.

As described below, we also simulated phenotypes for an additional set of individuals \mathbf{y}_x not included in either the validation or testing partitions. These individuals were selected to be close relatives of each of the validation partition individuals but experienced different micro-environments.

For each combination of genetic parameters, we declared the “best” prediction method to be the one with the highest average correlation with the true genetic values across the 500 simulations. Then we counted the proportion of the simulations in which this “best” method actually had the highest estimated accuracy when scored against \mathbf{y}_{n1} .

Data availability

Scripts for running all simulations and analyses described here are available at https://github.com/deruncie/multiTrait_crossValidation_scripts.

RESULTS

Although we ran simulations for two levels of heritability for the focal trait ($h_1^2 = \{0.2, 0.6\}$) we present results only for $h_1^2 = 0.2$. This is the “most-difficult” setting for prediction—when the heritability of the trait is low—but also the setting when we would expect the greatest benefit of using multi-trait models. Results for $h_1^2 = 0.6$ were qualitatively similar, but with higher overall prediction accuracies of all methods.

Accuracy of single and multi-trait methods in simulated data

With $h_1^2 = 0.2$ the true accuracy of prediction was moderate for all methods ($\text{cor}(\hat{\mathbf{u}}_{n1}, \mathbf{u}_{n1}) \sim 0.4 - 0.6$, Figure 1). Prediction accuracies for the single-trait method were constant across settings with different correlation structures because information from the secondary trait was not used.

The “standard” multi-trait model (i.e. CV1-style) that used phenotypic information only on the training partition slightly outperformed the single-trait model in some settings, more-so when the genetic and non-genetic correlations between traits were large and opposing and when the genetic determinacy of the secondary trait was high (Thompson and Meyer 1986). However it performed slightly worse whenever the genetic and residual correlations between traits were low. This was caused by inaccuracy in the estimation of the two covariance parameters ($\hat{\mathbf{g}}_{12}, \hat{\mathbf{r}}_{12}$). Neither multi-trait model performed worse than the single-trait model when the true \mathbf{G} and \mathbf{R} matrices were used Supplemental Figure 1, which we also verified by calculating the expected prediction accuracies analytically (See Appendix). In real data, multi-trait models require estimating more (co)variance parameters and therefore can show reduced performance when data are limited.

The CV2-style multi-trait method, which leverages additional phenotypic information on the secondary trait from the testing partition itself, showed dramatic improvements in prediction accuracy whenever genetic correlations among traits were large, irregardless of the non-genetic correlation between the traits. This is similar to the benefits seen by (Rutkoski *et al.* 2016) and (Lado *et al.* 2018). When the heritability of the secondary trait was high, the improvement in prediction accuracy was particularly dramatic (increasing to $\sim \rho = 0.6$). This is the potential advantage of incorporating secondary traits into prediction methods. However, the CV2 method also requires estimating \mathbf{G} and \mathbf{R} , and its performance was lower than the single-trait method whenever both genetic and residual

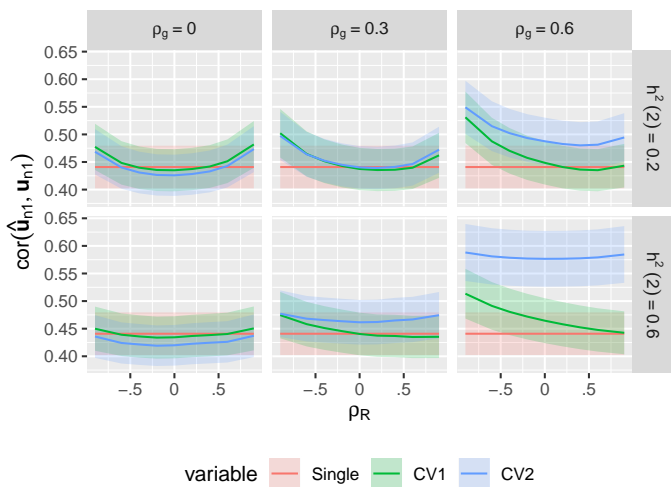


Figure 1 True prediction accuracy of single-trait and multi-trait prediction methods in simulated data. 500 simulations were run for each heritability of the secondary trait ($h_2^2 = \{0.2, 0.6\}$), and each combination of genetic and non-genetic correlation between the two traits ($\rho_g = \{0, 0.3, 0.6\}$, $\rho_R = \{-0.6, -0.4, -0.2, 0, 0.2, 0.4, 0.6\}$), all with $h_1^2 = 0.2$. For each simulation, we used 90% of the individuals as training to fit linear mixed models (either single or multi-trait), predicted the genetic values of the remaining validation individuals, and then measured the Pearson's correlation between the predicted ($\hat{\mathbf{u}}_{n1}$) and true (\mathbf{u}_{n1}) genetic values. In the CV1 method, we used only information on the training individuals to calculate $\hat{\mathbf{u}}_{n1}$. In the CV2 method, we used the training individuals to calculate $\hat{\mathbf{u}}_o$ and combined this with the observed phenotypes for the secondary trait on the validation individuals (\mathbf{y}_{n2}). Curves show the average correlation for each method across the 500 simulations. Ribbons show $\pm 1.96 \times SE$ over the 500 simulations.

287 correlations were low.

288 Therefore, multi-trait methods will not always be useful and
 289 it is important to test the relative performance of the different
 290 methods in real breeding scenarios. Unfortunately, we never know
 291 the true genetic values (\mathbf{u}_{n1}), and so must use proxy predictands to
 292 evaluate our methods in real data (Daetwyler *et al.* 2013; Legarra
 293 and Reverter 2018). In Figures 2A-B, we compare the prediction
 294 accuracies of the three methods using two candidate predictands:
 295 the observed phenotypic values (\mathbf{y}_{n1}) and estimated genetic values
 296 from a joint model fit to the complete dataset ($\hat{\mathbf{u}}_{n1}$).

297 Using the observed phenotypic values (\mathbf{y}_{n1}) as the predictand,
 298 the estimated accuracy of both the single-trait and CV1-style multi-
 299 trait prediction methods consistently under-estimated their true
 300 prediction accuracies. This is expected because in this setting
 301 80% of the phenotypic variation is non-genetic and cannot be
 302 predicted based on relatives alone. We therefore follow common
 303 practice to report a "corrected" estimate of the prediction accuracy
 304 by dividing by $\sqrt{\hat{h}^2}$ in Figure 2A. This correction factor itself must
 305 be estimated in real data, but when comparing models the same
 306 value of \hat{h}^2 should be used for each model so that differences in
 307 these estimates do not bias model selection.

308 In contrast, the estimated accuracy of the CV2-style multi-trait
 309 method varied dramatically across simulated datasets. We tended
 310 to overestimate the true accuracy when both genetic and non-
 311 genetic correlations were large and in the same direction, and
 312 dramatically underestimate the true accuracy when the two corre-
 313 lations were opposing. Importantly, there are situations where the
 314 CV2-style method appears to perform worse than the single-trait
 315 method based on \mathbf{y}_{n1} but actually performs better. Therefore, the
 316 observed phenotypic values are not reliable predictands to evalu-
 317 ate CV2-style methods when the intent is to estimate true genetic
 318 values and $\rho_R \neq 0$.

319 On the other hand, using estimated genetic values from a joint
 320 model fit to the complete dataset ($\hat{\mathbf{u}}_{n1}$) as the predictand led to
 321 dramatic over-estimation of the true prediction accuracy for all
 322 methods. This is also expected because the training data are used
 323 both to train the prediction model *and also* to create the testing
 324 dataset, a clear violation of the cross-validation rules that these
 325 datasets must be kept separate at all stages of the analysis. Again,
 326 the bias was most severe for the CV2-style method. Since this
 327 method is clearly invalid, we do not consider it further.

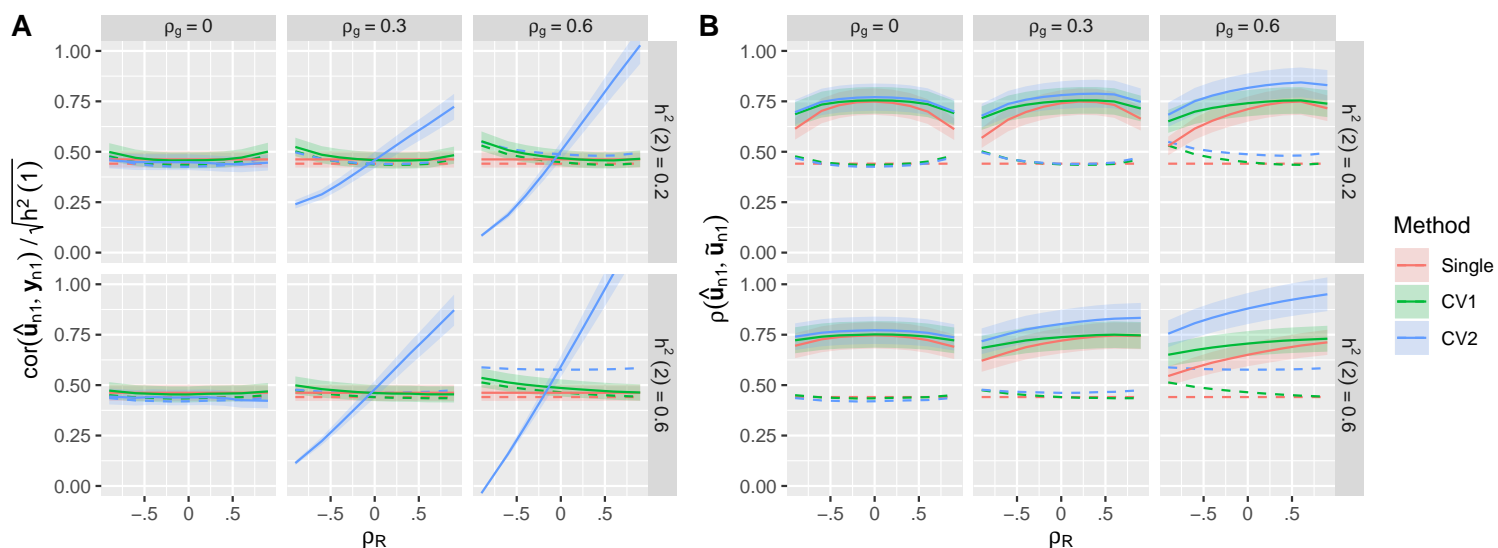


Figure 2 Estimated prediction accuracies based on candidate predictands. For the same set of simulations described in Figure 1, we estimated the prediction accuracies of the three methods using two different candidate predictands: **(A)** The observed phenotypic value y_{n1} for each training individual (with the correlation corrected by $1/\sqrt{h_1^2}$), or **(B)** An estimate of the genetic value of each training individual based on BLUPs calculated using the complete phenotype data (\hat{u}_{n1}). Solid lines in each panel show the average *estimated* accuracy for each method across the 500 simulations. Ribbons show $\pm 1.96 \times SE$ over the 500 simulations. Dotted lines show the average *true* accuracy from Figure 1.

328 Effects of predictand on model selection

329 To demonstrate the impact of biased estimates of model accuracy
 330 using y_{n1} on the effectiveness of model selection, we assessed in
 331 each simulation whether the single-trait or multi-trait methods
 332 had a higher *estimated* accuracy, and compared this result to the
 333 *true* difference in prediction accuracies in that simulation setting.

334 Figure 3 shows that selecting between the single-trait and CV1-
 335 style multi-trait models based on estimated accuracy using y_{n1}
 336 generally works well. Whenever one method is clearly better, we
 337 are able to choose that method $> 50\%$ of the time. But we never
 338 choose correctly $< 50\%$ of the time, even when the methods are
 339 approximately equivalent.

340 In contrast, when selecting between the single-trait and CV2-
 341 style multi-trait methods based on estimated accuracy using y_{n1} ,
 342 the differential bias in estimated accuracy between the two meth-
 343 ods frequently lead to sub-optimal model selection (Figure 3B).
 344 With opposing genetic and non-genetic covariances between the
 345 two traits, the better model was chosen $< 10\%$ of the time. In these
 346 situations, using y_{n1} to select a prediction method will obscure
 347 real opportunities to enhance prediction accuracy using multi-trait
 348 prediction models.

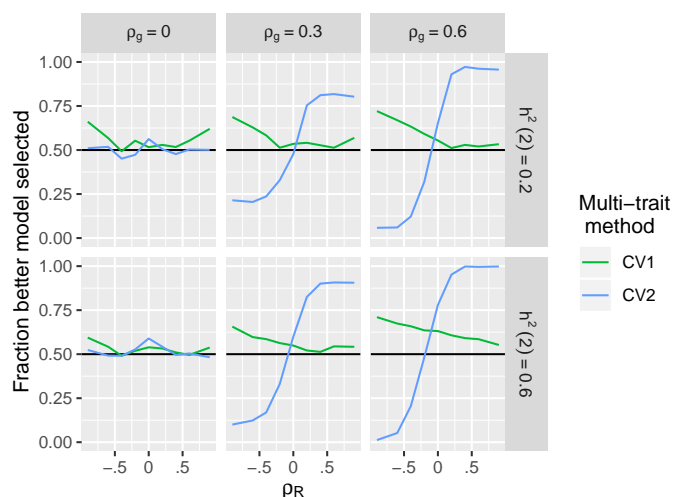


Figure 3 Impact of using phenotypic data to select between single-trait and multi-trait prediction methods. For each of the 500 simulations per genetic architecture described in Figure 1, we compared the estimated accuracy of a multi-trait prediction to the single-trait prediction. We then calculated the fraction of times that the selected model had higher average true accuracy in that setting (as shown in Figure 1).

Alternative estimates of multi-trait prediction accuracy

The CV2-style prediction method can be powerful because \mathbf{y}_{n2} provides information on the genetic value of the testing individuals themselves (through \mathbf{u}_{n2}), while \mathbf{y}_{n1} only provides indirect information on the genetic values of the testing individuals through the relatives. However, estimating prediction accuracy using \mathbf{y}_{n1} fails for the CV2-style prediction method because both the focal and secondary traits are observed on the same individual and therefore share the same non-genetic sources of variation. Since the CV2 method uses \mathbf{y}_{n2} , non-genetic deviations for the secondary trait \mathbf{e}_{n2} push $\hat{\mathbf{u}}_{n1}$ either towards or away from \mathbf{y}_{n1} depending on the estimated correlation \hat{r}_{12} . This either inflates or deflates the estimated accuracy, leading to incorrect model choices.

We now compare the effectiveness of three strategies for estimating cross-validation accuracy of CV2-style methods. To our knowledge, the second and third strategies are novel. Because the three methods have different data requirements, we implemented different experimental designs for each evaluation strategy.

Parametric estimate of accuracy. Our prediction $\hat{\mathbf{u}}_{n1}$ is similar to a selection index because it combines multiple pieces of information into a linear prediction. The accuracy of an index \mathbf{I} is: $cor_g(\mathbf{I}, \mathbf{y}) \sqrt{h_1^2}$, the genetic correlation between the index and phenotype multiplied by the heritability of the index (Falconer and Mackay 1996; Lopez-Cruz et al. 2019). Neither the genetic correlation nor the heritability can be directly observed, but we can estimate both as parameters of a multi-trait linear mixed model with the same form as (5). To be a valid cross-validation score, these parameters must be estimated with data only in the validation partition, rather than reusing estimates from model training. Since both model training and model evaluation equally require estimates of \mathbf{G} and \mathbf{R} , we divided the data 50:50 into training and validation partitions in each simulation, thus using 404 lines to train the prediction models and 403 lines to evaluate the prediction accuracy.

The parametric estimates of prediction accuracy for the CV2 method were less biased than the $cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{n1})$, the non-parametric estimates using \mathbf{y}_{n1} as a predictand (Figure 4A, compare to Figure 2). This led to more consistent model selections between the CV2 and single-trait methods (Figure 4B). However, the parametric approach still underestimated the accuracy of the CV2 method when the genetic and residual correlations were in

opposite directions, leading to model selection accuracies <50%. This negative bias was due to poor estimation of \mathbf{G} and \mathbf{R} for the selection indices, given the limited sample sizes remaining after the data were partitioned.

Semi-parametric estimate of accuracy. In principle, we can correct for the bias in the non-parametric accuracy estimate ($cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{n1})$) from the CV2-style method by calculating an adjustment factor based on the theoretical bias relative to the true accuracy ($cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1})$). This is similar to the semi-parametric accuracy estimates presented by (Legarra and Reverter 2018), and the “correction” of accuracy estimates by $1/\sqrt{h^2}$ used above to account for the difference in variance between \mathbf{y}_{n1} and \mathbf{u}_{n1} . As we derive in the Appendix, the difference between the true correlation from a CV2-style methods and its CV2 cross-validation estimate when a single secondary trait is used is:

$$\frac{\hat{\mathbf{g}}_{12}\mathbf{r}_{21}}{\sqrt{var(\hat{\mathbf{u}}_{n1}^{(3)})var(\mathbf{y}_{n1})}} \frac{tr(\mathbf{S}(\mathbf{K}^{-1})_{nn}\hat{\mathbf{V}}_c^{-1}\mathbf{K}_{nn})}{n-1}. \quad (8)$$

with \mathbf{V}_c defined above and $\mathbf{S} = \mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{n}$. The bias is a function of the the correlation among traits through the product $\hat{\mathbf{g}}_{12}\mathbf{r}_{21}$ (as the second term does not involve these parameters, and in most cases is ≈ 1), and is large and positive (i.e. accuracy is overestimated) when $\hat{\mathbf{g}}_{12}$ and \mathbf{r}_{12} are large and in the same direction, and large and negative (i.e. accuracy is underestimated) when these covariances are in opposite directions. Given this result, we can correct $cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{n1})$ by subtracting 8 from the estimated correlation, again corrected by $1/\sqrt{h^2}$ (Figure 5).

Clearly, the quality of this correction will depend on the accuracy of $\hat{\mathbf{g}}_{12}$ and \hat{r}_{12} as estimates of \mathbf{g}_{12} and \mathbf{r}_{12} . In Figure 5A, we show that the corrected correlation estimate has greatly reduced bias, particularly the dependence of the bias on the non-genetic covariance between the traits \mathbf{r}_{12} . However the correction is not perfect. Corrected accuracy estimates tend to overestimate the true accuracy. This over-estimation is caused by error in $\hat{\mathbf{G}}$ and $\hat{\mathbf{R}}$ as estimates of the true covariances: The correction factor is nearly perfect when the true covariance matrices are used in place of their estimates Supplemental Figure 2.

Using the semi-parametric accuracy estimates, we are more successful at selecting the best model over the range of genetic architectures (Figure 5B). The frequency of selecting the correct model rarely drops below 50% and is relatively constant with respect to the residual correlation between traits.

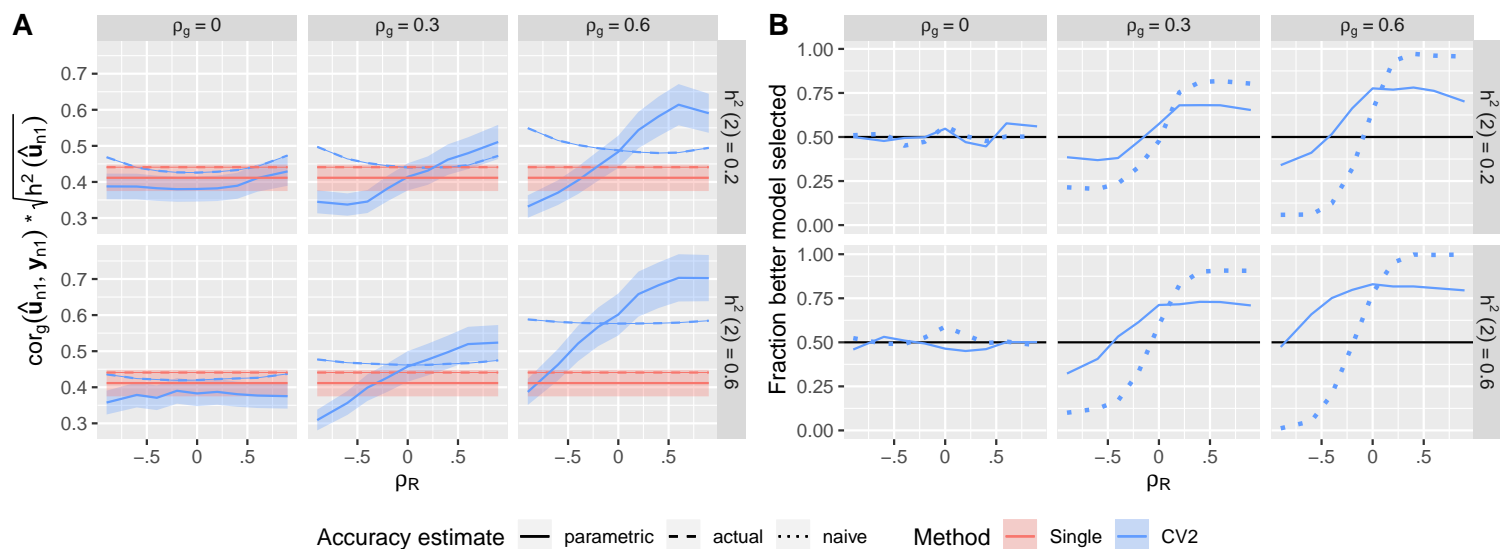


Figure 4 Parametric accuracy estimates. Estimated prediction accuracies and model selection accuracies for CV2-style methods using the parametric method. **(A)** Solid curves: estimates of prediction accuracy. Dashed curves: true prediction accuracy based on \mathbf{u}_{n1} . Dotted curves: estimated prediction accuracy using \mathbf{y}_{n1} from Figure 2A. Ribbons show $\pm 1.96 \times SE$ over the 500 simulations. **(B)** Solid curves: Fraction of the 500 simulations in which the better method (between CV2 and single-trait) for predicting the true genetic values was correctly selected. Dotted curve: model selection based on the naive prediction accuracy.

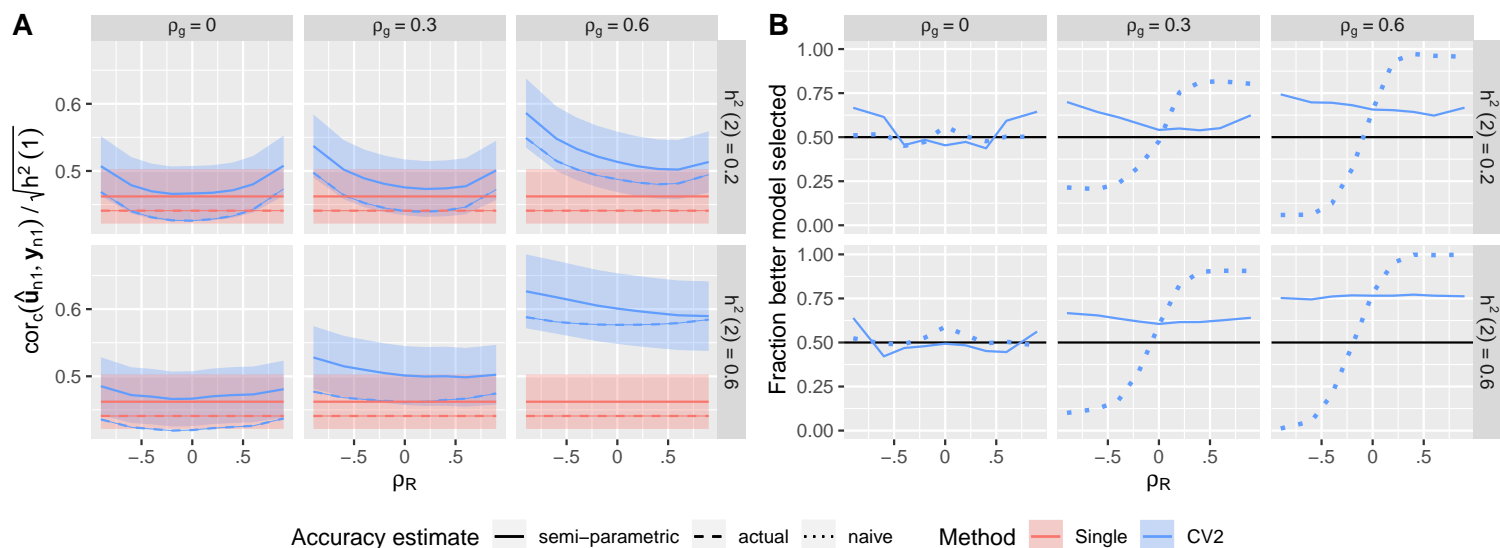


Figure 5 Semi-parametric accuracy estimates. Estimated prediction accuracies and model selection accuracies for CV2-style methods after semi-parametric correction. **(A)** Solid curves: corrected estimates of prediction accuracy. Dashed curves: uncorrected estimates of prediction accuracy based on \mathbf{y}_{n1} (mirroring Figure 3). Dotted curves: true prediction accuracy. Ribbons show $\pm 1.96 \times SE$ over the 500 simulations. **(B)** Solid curves: Fraction of the 500 simulations in which the better method (between CV2 and single-trait) for predicting the true genetic values was correctly selected. Dotted curve: model selection based on the naive un-corrected prediction accuracy.

418 **CV2* cross-validation strategy.** Since the biased estimate of pre- 460
419 diction accuracy for CV2-style methods is due to non-genetic cor- 461
420 relations between y_{n2} used for prediction and the predictand y_{n1} , 462
421 an alternative strategy, which we call CV2*, is to use phenotypic 463
422 information on close relatives of the testing individuals (y_{x1}) to 464
423 validate the model predictions in place of their own focal trait 465
424 phenotypes (y_{n1}). These “surrogate” validation individuals must 466
425 also be excluded from the model training and raised so that they 467
426 do not share the same non-genetic deviations as the testing indi- 468
427 viduals: $cor(\mathbf{e}_{x1}, \mathbf{e}_{n1}) = 0$. Therefore, $\hat{\mathbf{u}}_{x1}$ will not be artificially 469
428 pushed towards or away from \mathbf{u}_{x1} (measured on relatives) by y_{n2} 470
429 (measured on testing individuals), preventing this source of bias 471
430 in the estimated accuracy. 472

431 We implemented the CV2* cross-validation strategy in two 473
432 ways, simulating two different breeding schemes. 474

433 First, we considered the situation common in plant breeding 475
434 where inbred lines (i.e. clones) are tested, and each line is grown 476
435 in several plots in a field [Bernardo \(2002\)](#). Here, we can use one 477
436 set of clones for prediction (y_{n2}), and the other set of clones as 478
437 trait-1 surrogates (y_{x1}). Since they are clones, $\mathbf{u}_{x1} = \mathbf{u}_{n1}$ and y_{n2} 479
438 is just as good for predicting \mathbf{u}_{x1} as y_{x2} . Generally in this type of 480
439 experiment, replicate plots of each line will be combined prior to 481
440 analysis into a single line mean (or BLUP). But since we require 482
441 y_{n2} and y_{x1} to be recorded from separate individuals, each value 483
442 will have $2\times$ the residual variance because it is based on $1/2$ as 484
443 much data as the line means used for model training. Therefore, 485
444 in our simulations we drew two independent residual values for 486
445 each line in the validation partition, each with a variance of $2R$. 487
446 For these simulations, we used a 90:10 training:validation split. 488

447 Second, we considered the situation more common in animal 489
448 breeding where clones are not available. In this case, the best op- 490
449 tion for CV2* would be to select pairs of closely related individuals 491
450 to include in the training set; we use the first individual of the 492
451 pair as y_{n2} and the second as y_{x1} . To implement this strategy, we 493
452 again started with a validation partition of 10% of the lines. Then 494
453 for each line, we selected the most closely related remaining line 495
454 ($\arg \max_j \mathbf{K}_{ij}$ for validation line i) and held this additional set of 496
455 10% of the lines as y_{x1} . This left a training partition with only 80% 497
456 of the lines. The average genetic relatedness of validation partition 498
457 pairs in these simulations was 0.38. 499

458 Figure 6A shows that for the first setting with split clones, es- 500
459 timates of prediction accuracy for CV2-style predictions by CV2*

are vastly more accurate than the naive estimates based on y_{n1} ,
but they are slightly downwardly biased because of the increased
residual variance of y_{n1} and y_{x2} . Model selection works fairly
well across all settings when clones are used (Figure 6B, blue
lines), although with slightly lower success rates than for the semi-
parametric method. However, when we implementing the second
approach with nearest relatives (not clones), model selection was
rarely successful - we consistently chose the wrong model across
most simulation settings unless the genetic and residual correla-
tions were opposing. This is because the validation pairs were
too distantly related to provide any additional information on
genetic merit relative to individuals in the training partition. In-
terestingly, this method is relatively successful in the situations
where the parametric method fails (see Figure 4B), and so may be
complimentary.

475 DISCUSSION

476 Our study highlights a potential pitfall in using cross-validation to
477 estimate the accuracy of multi-trait genomic prediction methods.
478 When secondary traits are used to aid in the prediction of focal
479 traits and these secondary traits are measured on the individuals
480 to be tested, cross-validation evaluated against phenotypic obser-
481 vations can be severely biased and result in poor model choices.
482 Unfortunately, we rarely know the true genetic value of any indi-
483 vidual and therefore can only evaluate our models with phenotypic
484 data (since multi-trait-derived estimated genetic values are even
485 more severely biased as we demonstrated above (Figure 2B)). We
486 cannot find earlier discussions of this problem in the literature.
487 However a growing number of studies aim to use cheap or early-
488 life traits to improve predictions of genetic worth for individuals
489 in later-life traits (ex. [Pszczola et al. 2013](#); [Rutkoski et al. 2016](#); [Fer-
490 nandes et al. 2017](#); [Lado et al. 2018](#)). Therefore the issue is becoming
491 more important.

The problematic bias in the cross-validation-based accuracy
estimates is caused by non-genetic correlations between the pre-
dictors that we want to use (i.e. the secondary traits) and our best
predictand (the phenotypic value of the trait in the testing indi-
viduals) – non-genetic correlations between two traits measured
on the same individual are expected. However, in some cases this
correlation is zero by construction, and standard cross-validation
approaches can be valid. For example, in the original description
of the CV2 cross-validation method by ([Burgueño et al. 2012](#)), each

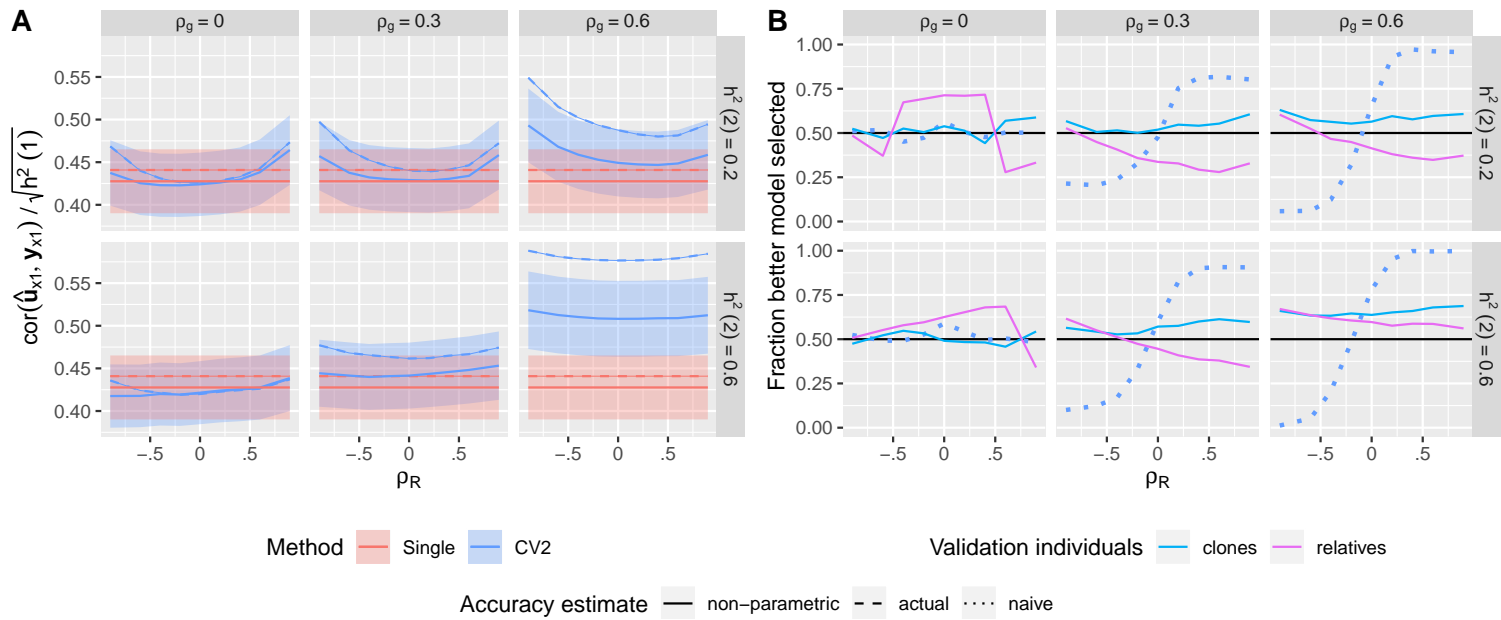


Figure 6 Non-parametric CV* accuracy estimates. Estimated prediction accuracies and model selection accuracies based on the phenotypic values of close relatives. **(A)** Solid curves: Estimated prediction accuracies of the CV2-style and Single-trait methods evaluated against y_{x1} using clones. Dashed curves: True prediction accuracies of each method. Ribbons show $\pm 1.96 \times SE$ over the 500 simulations. **(B)** Solid curves: Fraction of the 500 simulations in which the better method (between CV2 and single-trait) for predicting the true genetic values was correctly selected based on the phenotypes of relatives of the testing individuals. Dotted curve: Fraction of correct models selected based on the naive estimator.

501 trait was measured in a different environment. In this case, the
 502 traits were measured on different individuals and therefore did
 503 not share any non-genetic correlation. Also, CV1-style methods
 504 do not suffer from this problem because phenotypic information
 505 on the secondary traits *in the testing individuals* is not used for
 506 prediction. Similarly, this bias does not occur when the target of
 507 prediction is the phenotypic value itself (rather than the individ-
 508 ual's genetic value). For example, in medical genetics the aim is
 509 to predict whether or not a person will get a disease or not, not
 510 her genetic propensity to get a disease had she been raised in a
 511 different environment (ex [Spiliopoulou et al. 2015](#); [Dahl et al. 2016](#)).

512 We note that the common strategy of two-step genome selection:
 513 using single-trait methods to calculate estimated genetic values
 514 for each line:trait and then using these estimated genetic values
 515 as training (and validation) data, does not get around the prob-
 516 lem identified here. Using estimated genetic values instead of
 517 phenotypic values will tend to increase the genetic repeatability
 518 of the training and validation values, and therefore increase the
 519 overall prediction accuracy of all methods. But these estimated
 520 genetic values will still be biased by the non-genetic variation, and

521 the biases across traits will still be correlated by the non-genetic
 522 correlations. Therefore the same issue will arise.

523 Also, while we have used a GBLUP-like genomic prediction
 524 method for the analyses presented here, the same result will hold
 525 for any multi-trait prediction method that aims to use information
 526 from y_{n2} when there are non-genetic correlations with y_{n1} , i.e. any
 527 method that is evaluated with the CV2 cross-validation method
 528 on multiple traits measured on the same individual ([Calus and
 529 Veerkamp 2011](#); [Jia and Jannink 2012](#); [Fernandes et al. 2017](#)). This
 530 includes multi-trait versions of the Bayes Alphabet methods ([Calus
 531 and Veerkamp 2011](#); [Cheng et al. 2018](#)), or neural network or Deep
 532 Learning methods ([Montesinos-López et al. 2018](#)).

533 We presented three partial solutions to this problem, spanning
 534 from fully parametric to fully non-parametric.

535 The parametric solution relies on fitting a new multi-trait mixed
 536 model to the predicted values and the predictand, with the accu-
 537 racy estimated as the genetic correlation scaled by the heritability
 538 of the prediction. This solution is always available as long as the
 539 individuals in the validation partition have non-zero genomic re-
 540 latedness and the full dataset is large enough to estimate genetic

541 correlations in both training and validation partitions. However 583
542 it generally worked poorly in our simulations because \mathbf{G} and \mathbf{R} 584
543 were not estimated accurately. It may work better with very large 585
544 datasets. Also, because this parametric approach relies on the same 586
545 assumptions about the data (i.e. multivariate normality) as the 587
546 prediction model, it loses some of the guarantees of reliability that 588
547 completely non-parametric cross-validation methods can claim. 589

548 The semi-parametric solution aims to correct the non- 590
549 parametric correlation estimate for the bias caused by the non-null 591
550 residual correlation among traits. This correction factor is only 592
551 needed for CV2-style multi-trait prediction approaches, and is sim- 593
552 ilar to the approach of (Legarra and Reverter 2018) for single-trait 594
553 models. We show that this correction factor can work well, par- 595
554 ticularly if the covariances among traits are well estimated. We 596
555 only derived this correction method for prediction methods based 597
556 on linear mixed effect models with a single known genetic covari- 598
557 ance structure (i.e. GBLUP and RKHS-style methods with fixed 599
558 kernels), although the approximation $\frac{\hat{\mathbf{g}}_{12}\hat{\mathbf{r}}_{12}}{\sqrt{(\hat{\mathbf{u}})var(\mathbf{u})}}$ will probably 600
559 be approximately correct for other methods. However, when co- 601
560 variances are poorly estimated, the correction factor can still lead 602
561 to biased estimates of model accuracy. We are currently investigat- 603
562 ing whether Bayesian methods that sample over this uncertainty 604
563 can be useful, and will implement this method in JWAS (Cheng 605
564 *et al.* 2018). This method is semi-parametric, so also relies on dis- 606
565 tributional assumptions about the data and may fail when these 607
566 assumptions are not met.

567 As a third alternative, we proposed the CV2* cross-validation 608
568 method, a fully non-parametric approach for assessing CV2-style 609
569 multi-trait prediction accuracy. CV2* uses phenotypic values of 610
570 the focal trait from relatives of the testing individuals in place of 611
571 the phenotypic values of that trait from the testing individuals 612
572 themselves. If the close relatives are raised independently, they 613
573 will not share non-genetic variation, removing the source of bias in 614
574 the cross-validation estimate (Figure 6A). The CV2* method works 615
575 best when clones of the testing individuals are available. With 616
576 clones, secondary trait phenotypes of the testing individuals can 617
577 be used directly to predict focal trait genetic values of their clones 618
578 because the genetic values are identical. Replicates of inbred lines 619
579 are frequently used in plant breeding trials (Bernardo 2002). In 620
580 this case, all replicates should be held-out as a group from the 621
581 training data. Then the replicates can be partitioned again into 622
582 two sets; secondary trait phenotypes from one set can be incor-

porated into the genetic value predictions for the lines, and these
predictions evaluated against the phenotypic values of the other
set. To compare this estimate of CV2-style prediction accuracy to
the prediction accuracy for a single-trait method, the single-trait
method's predictions should be compared against the same set of
replicates of each line (i.e., not a joint average over all replicates
of the line as would be typical for single-trait cross-validation).
However, because of the separation of the replicates, each replicate
will have higher residual variance, which reduces the accuracy of
this method. Clones are less common outside of plant breeding,
so more distant relatives need to be used instead. In this case,
the estimated prediction accuracies of CV2-style methods will be
downwardly biased. In our simulations, despite relatively close
relatives for each validation line being available, this approach was
not successful.

In our simulations, the semi-parametric approach was the most
reliable, and the fully parametric approach the least reliable. How-
ever the fully parametric approach is always possible to implement
while our semi-parametric and non-parametric approaches may
not be possible depending on the prediction model used and the
structure of the experimental design.

604 CONCLUSIONS

We expect that multi-trait methods for genomic prediction carry
great promise to accelerate both plant and animal breeding. How-
ever there is a need to design better methods to evaluate and train
the prediction methods to ensure that models can be accurately
compared. We have presented and compared three contrasting
methods to evaluating multi-trait methods. Each of these methods
is preferred to naive cross-validation when secondary traits of the
target individuals are used to predict their focal traits. However
the methods can give contrasting answers for different datasets, so
careful consideration of which evaluation method to use is critical
when choosing among prediction methods.

616 ACKNOWLEDGMENTS

We would like to thank Erin Calfee and Graham Coop for sug-
gesting the CV2* method, Gustavo de los Campos for pointing us
towards the parametric approach, and helpful comments from two
anonymous reviewers.

HC's work is support by US Department of Agriculture, Agri-
culture and Food Research Initiative National Institute of Food

623 and Agriculture Competitive Grant No. 2018-67015-27957 DER
624 was supported by the United States Department of Agriculture
625 (USDA) National Institute of Food and Agriculture (NIFA), Hatch
626 project 1010469.

627 SUPPLEMENTAL FIGURES

628 **Supplemental Figure 1 Actual prediction accuracy of single-trait
629 and multi-trait prediction methods in simulated data when G
630 and R are known.** 500 simulations were run for each heritabil-
631 ity of the secondary trait ($h_2^2 = \{0.2, 0.6\}$), and each combina-
632 tion of genetic and non-genetic correlation between the two traits
633 ($\rho_g = \{0, 0.3, 0.6\}$, $\rho_R = \{-0.6, -0.4, -0.2, 0, 0.2, 0.4, 0.6\}$), all with
634 $h_1^2 = 0.2$. For each simulation, we used the 900 training individuals
635 to fit linear mixed models (either single or multi-trait) condition-
636 ing on the true values for **G** and **R**, predicted the genetic values
637 of the 100 testing individuals, and then measured the Pearson's
638 correlation between the predicted ($\hat{\mathbf{u}}_{n1}$) and true (\mathbf{u}_{n1}) genetic val-
639 ues. In the CV1 method, we used only information on the testing
640 individuals to calculate $\hat{\mathbf{u}}_{n1}$. In the CV2 method, we used the
641 training individuals to calculate $\hat{\mathbf{u}}_o$ and combined this with the
642 observed phenotypes for the secondary trait on the testing individ-
643 uals (\mathbf{y}_{n2}). Curves show the average correlation for each method
644 across the 500 simulations. Ribbons show $\pm 1.96 \times SE$ over the
645 500 simulations. Dashed lines show analytical calculations of the
646 expected correlation given one representative training:validation
647 data partition.

648 **Supplemental Figure 2 Estimated prediction accuracies and
649 model selection accuracies for single-trait and multi-trait pre-
650 diction methods after semi-parametric correction when G and
651 R are known.** Ribbons show $\pm 1.96 \times SE$ over the 500 simula-
652 tions. Dashed lines show the mean actual prediction accuracy:
653 $cor(\hat{\mathbf{u}}_{n1}, \mathbf{u}_{n1})$.

654 LITERATURE CITED

655 Amer, P. R. and G. Banos, 2010 Implications of avoiding overlap
656 between training and testing data sets when evaluating genomic
657 predictions of genetic merit. *Journal of Dairy Science* **93**: 3320–
658 3330.
659 Bernardo, R., 2002 *Breeding for Quantitative Traits in Plants*. Stemma
660 Press.

Burgueño, J., G. de Los Campos, K. Weigel, and J. Crossa, 2012 Ge-
661 nomic Prediction of Breeding Values when Modeling Genotype
662 \times Environment Interaction using Pedigree and Dense Molecular
663 Markers. *Crop Science* **52**: 707.
664
665 Calus, M. P. and R. F. Veerkamp, 2011 Accuracy of multi-trait
666 genomic selection using different methods. *Genetics Selection
667 Evolution* **43**: 26.
668
669 Cheng, H., R. Fernando, and D. Garrick, 2018 Jwas: Julia imple-
670 mentation of whole-genome analysis software. In *Proceedings
671 of the World Congress on Genetics Applied to Livestock Production*,
672 volume 11.
673
674 Crossa, J., P. Pérez-Rodríguez, J. Cuevas, O. Montesinos-López,
675 D. Jarquín, *et al.*, 2017 Genomic Selection in Plant Breeding:
676 Methods, Models, and Perspectives. *Trends in plant science* **22**:
677 961–975.
678
679 Daetwyler, H. D., M. P. L. Calus, R. Pong-Wong, G. de Los Cam-
680 pos, and J. M. Hickey, 2013 Genomic Prediction in Animals and
681 Plants: Simulation of Data, Validation, Reporting, and Bench-
682 marking. *Genetics* **193**: 347–365.
683
684 Dahl, A., V. Iotchkova, A. Baud, Å. Johansson, U. Gyllensten, *et al.*,
685 2016 A multiple-phenotype imputation method for genetic stud-
686 ies. *Nature Genetics* **48**: 466–472.
687
688 de Los Campos, G., J. M. Hickey, R. Pong-Wong, H. D. Daetwyler,
689 and M. P. L. Calus, 2013 Whole-Genome Regression and Predic-
690 tion Methods Applied to Plant and Animal Breeding. *Genetics*
691 **193**: 327–345.
692
693 Falconer, D. S. and T. F. C. Mackay, 1996 *Introduction to quantitative
694 genetics 4th edition*. Pearson, fourth edition edition.
695
696 Fernandes, S. B., K. O. G. Dias, D. F. Ferreira, and P. J. Brown,
697 2017 Efficiency of multi-trait, indirect, and trait-assisted genomic
698 selection for improvement of biomass sorghum. *TAG Theoretical
699 and applied genetics Theoretische und angewandte Genetik* **131**:
700 747–755.
701
702 Gianola, D., 2013 Priors in whole-genome regression: the bayesian
703 alphabet returns. *Genetics* **194**: 573–596.
704
705 Gianola, D. and C. C. Schon, 2016 Cross-Validation Without
706 Doing Cross-Validation in Genome-Enabled Prediction. *G3:
707 Genes | Genomes | Genetics* **6**: 3107–3128.
708
709 Hastie, T., R. Tibshirani, and J. Friedman, 2009 *The elements of statisti-
710 cal learning*. Data Mining, Inference, and Prediction, Springer-
711 Verlag New York, New York, second edition edition.
712
713 Hayes, B. J., P. J. Bowman, A. J. Chamberlain, and M. E. Goddard,

- 2009 Invited review: Genomic selection in dairy cattle: Progress
and challenges. *Journal of Dairy Science* **92**: 433–443.
- Heslot, N., H.-P. Yang, M. E. Sorrells, and J.-L. Jannink, 2012 Ge-
nomic Selection in Plant Breeding: A Comparison of Models. *Crop Science* **52**: 146–160.
- Hothorn, T., F. Leisch, A. Zeileis, and K. Hornik, 2005 The design
and analysis of benchmark experiments. *Journal of Computa-
tional and Graphical Statistics* **14**: 675–699.
- Jia, Y. and J.-L. Jannink, 2012 Multiple-Trait Genomic Selection
Methods Increase Genetic Value Prediction Accuracy. *Genetics*
192: 1513–1522.
- Kaufman, S., S. Rosset, C. Perlich, and O. Stitelman, 2012 Leakage
in data mining: Formulation, detection, and avoidance. *ACM*
Transactions on Knowledge Discovery from Data (TKDD) **6**:
15–21.
- Lado, B., D. Vázquez, M. Quincke, P. Silva, I. Aguilar, *et al.*, 2018
Resource allocation optimization with multi-trait genomic pre-
diction for bread wheat (<Emphasis Type="Italic">Triticum aes-
tivum</Emphasis> L.) baking quality. *TAG Theoretical and*
applied genetics Theoretische und angewandte Genetik **131**:
2719–2731.
- Legarra, A. and A. Reverter, 2018 Semi-parametric estimates of
population accuracy and bias of predictions of breeding values
and future phenotypes using the LR method. *Genetics Selection*
Evolution **50**: 659.
- Lopez-Cruz, M., J. Crossa, D. Bonnett, S. Dreisigacker, J. Poland,
et al., 2015 Increased Prediction Accuracy in Wheat Breeding
Trials Using a Marker × Environment Interaction Genomic Se-
lection Model. *G3: Genes | Genomes | Genetics* **5**: 569–582.
- Lopez-Cruz, M., E. Olson, G. Rovere, J. Crossa, S. Dreisigacker,
et al., 2019 Genetic image-processing using regularized selection
indices. *bioRxiv* .
- Meuwissen, T. H., B. J. Hayes, and M. E. Goddard, 2001 Prediction
of total genetic value using genome-wide dense marker maps.
Genetics **157**: 1819–1829.
- Montesinos-López, O. A., A. Montesinos-López, J. Crossa, D. Gi-
anola, C. M. Hernández-Suárez, *et al.*, 2018 Multi-trait, Multi-
environment Deep Learning Modeling for Genomic-Enabled
Prediction of Plant Traits. *G3: Genes | Genomes | Genetics* **8**:
g3.200728.2018–3840.
- Pszczola, M., R. F. Veerkamp, Y. de Haas, E. Wall, T. Strabel, *et al.*,
2013 Effect of predictor traits on accuracy of genomic breeding
values for feed intake based on a limited cow reference popula-
tion. *animal* **7**: 1759–1768.
- Rutkoski, J., J. Poland, S. Mondal, E. Autrique, L. G. Pérez, *et al.*,
2016 Canopy temperature and vegetation indices from high-
throughput phenotyping improve accuracy of pedigree and
genomic selection for grain yield in wheat. *G3: Genes, Genomes,
Genetics* **6**: 2799–2808.
- 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.
Human molecular genetics **24**: 4167–4182.
- Thompson, R. and K. Meyer, 1986 A review of theoretical aspects
in the estimation of breeding values for multi-trait selection.
Livestock Production Science **15**: 299–313.
- Utz, H. F., A. E. MELCHINGER, and C. C. Schön, 2000 Bias and
Sampling Error of the Estimated Proportion of Genotypic Vari-
ance Explained by Quantitative Trait Loci Determined From
Experimental Data in Maize Using Cross Validation and Valid-
ation With Independent Samples. *Genetics* **154**: 1839–1849.
- Ziyatdinov, A., M. Vazquez-Santiago, H. Brunel, A. Martinez-Perez,
H. Aschard, *et al.*, 2018 lme4qtl: linear mixed models with flexi-
ble covariance structure for genetic studies of related individuals.
BMC Bioinformatics p. btw080.

768 APPENDIX

769 Here, we derive the genomic predictions $\hat{\mathbf{u}}_{n1}$ given \mathbf{y} for the three prediction models that we use in the main text, and then evaluate the
 770 expected covariances between these predictions and the predictands \mathbf{u}_{n1} and \mathbf{y}_{n1} . We derive these relations for the more general situation
 771 with $p \geq 1$ “secondary” traits and a single “focal” trait.

We start with a phenotypic data matrix \mathbf{Y} with n individuals and $p + 1$ traits, where the first trait (first column of \mathbf{Y}) is the “focal” trait, and the other p traits are “secondary” traits. We first divide \mathbf{Y} into a training partition (“old” individuals) and a testing partition (“new” individuals), and arrange them with the testing partition first, so we can partition $\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_n \\ \mathbf{Y}_o \end{bmatrix} = \begin{bmatrix} \mathbf{y}_{n1} & \mathbf{Y}_{n2} \\ \mathbf{y}_{o1} & \mathbf{Y}_{o2} \end{bmatrix}$. We then work with stacked versions of these phenotype matrices: $\mathbf{y} = \text{vec}(\mathbf{Y})$, $\mathbf{y}_n = \text{vec}(\mathbf{Y}_n)$, $\mathbf{y}_o = \text{vec}(\mathbf{Y}_o)$. Our genetic model for \mathbf{y} is:

$$\begin{aligned} \mathbf{y} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{u} + \mathbf{e} \\ \boldsymbol{\beta} &= [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2]^\top \\ \mathbf{u} &\sim \text{N}(\mathbf{0}, \mathbf{G} \otimes \mathbf{K}) \\ \mathbf{e} &\sim \text{N}(\mathbf{0}, \mathbf{R} \otimes \mathbf{I}_n) \end{aligned}$$

where \mathbf{G} and \mathbf{R} are genetic and phenotypic covariance matrices for the $p + 1$ traits, and \mathbf{K} is the $n \times n$ genomic relationship matrix among the lines. For convenience below, we partition the following matrices as follows: We partition the trait vectors for the training individuals and covariance matrices between the “focal” (index 1) and “secondary traits” (index 2):

$$\begin{aligned} \mathbf{y}_o &= \begin{bmatrix} \mathbf{y}_{o1} \\ \mathbf{y}_{o2} \end{bmatrix}, \mathbf{u}_o = \begin{bmatrix} \mathbf{u}_{o1} \\ \mathbf{u}_{o2} \end{bmatrix}, \mathbf{e}_o = \begin{bmatrix} \mathbf{e}_{o1} \\ \mathbf{e}_{o2} \end{bmatrix}, \mathbf{X}_o\boldsymbol{\beta} = \begin{bmatrix} \mathbf{X}_{o1}\boldsymbol{\beta}_1 \\ \mathbf{X}_{o2}\boldsymbol{\beta}_2 \end{bmatrix} \\ \mathbf{G} &= \begin{bmatrix} \mathbf{g}_{11} & \mathbf{g}_{12} \\ \mathbf{g}_{21} & \mathbf{G}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{1\cdot} \\ \mathbf{G}_{2\cdot} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{\cdot 1} & \mathbf{G}_{\cdot 2} \end{bmatrix} \\ \mathbf{R} &= \begin{bmatrix} r_{11} & \mathbf{r}_{12} \\ \mathbf{r}_{21} & \mathbf{R}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{1\cdot} \\ \mathbf{R}_{2\cdot} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{\cdot 1} & \mathbf{R}_{\cdot 2} \end{bmatrix}, \end{aligned}$$

where scalars are normal text, vectors are bold-face lower case letters, and matrices are bold-face capital letters. Partitions for the testing individuals are similar. We also partition the genomic relationship matrix and its inverse between the training and testing individuals:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{nn} & \mathbf{K}_{no} \\ \mathbf{K}_{on} & \mathbf{K}_{oo} \end{bmatrix}, \mathbf{K}^{-1} = \begin{bmatrix} (\mathbf{K}^{-1})_{nn} & (\mathbf{K}^{-1})_{no} \\ (\mathbf{K}^{-1})_{on} & (\mathbf{K}^{-1})_{oo} \end{bmatrix}$$

772 Derivation of genomic predictions

Single trait predictions For the single-trait prediction, we begin by estimating \hat{g}_{11} , \hat{r}_{11} and $\hat{\boldsymbol{\beta}}_1$ by REML using only \mathbf{y}_{o1} . The joint distribution of \mathbf{u}_{n1} and \mathbf{y}_{o1} is:

$$\begin{bmatrix} \mathbf{u}_{n1} \\ \mathbf{y}_{o1} \end{bmatrix} \sim \text{N} \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{X}_{o1}\boldsymbol{\beta}_1 \end{bmatrix}, \begin{bmatrix} g_{11}\mathbf{K}_{nn} & g_{11}\mathbf{K}_{no} \\ g_{11}\mathbf{K}_{on} & g_{11}\mathbf{K}_{oo} + r_{11}\mathbf{I} \end{bmatrix} \right).$$

Let: $\mathbf{V}_{o1} = g_{11}\mathbf{K}_{oo} + r_{11}\mathbf{I}$. Therefore $E[\mathbf{u}_{n1}|\mathbf{y}_{o1}] = g_{11}\mathbf{K}_{no}\mathbf{V}_{o1}^{-1}(\mathbf{y}_{o1} - \mathbf{X}_{o1}\boldsymbol{\beta}_1)$, so our prediction is:

$$\hat{\mathbf{u}}_{n1}^{(1)} = \hat{g}_{11}\mathbf{K}_{no}\hat{\mathbf{V}}_{o1}^{-1}(\mathbf{y}_{o1} - \mathbf{X}_{o1}\hat{\boldsymbol{\beta}}_1). \quad (9)$$

To simplify, note that the joint distribution of \mathbf{u}_{o1} and \mathbf{y}_{o1} in the training data is:

$$\begin{bmatrix} \mathbf{u}_{o1} \\ \mathbf{y}_{o1} \end{bmatrix} \sim N \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{X}_{o1}\boldsymbol{\beta}_1 \end{bmatrix}, \begin{bmatrix} g_{11}\mathbf{K}_{oo} & g_{11}\mathbf{K}_{oo} \\ g_{11}\mathbf{K}_{oo} & g_{11}\mathbf{K}_{oo} + r_{11}\mathbf{I} \end{bmatrix} \right)$$

773 Therefore, $\hat{\mathbf{u}}_{o1}|\mathbf{y}_{o1} = \hat{g}_{11}\mathbf{K}_{oo}\hat{\mathbf{V}}_{o1}^{-1}(\mathbf{y}_{o1} - \mathbf{X}_{o1}\hat{\boldsymbol{\beta}}_1)$. Rearranging and plugging this in above simplifies to: $\hat{\mathbf{u}}_{n1}^{(1)} = \mathbf{K}_{no}\mathbf{K}_{oo}^{-1}\hat{\mathbf{u}}_{o1}$.

CV1-style multi-trait predictions For CV1-style multi-trait prediction, we begin by estimating $\hat{\mathbf{G}}$, $\hat{\mathbf{R}}$ and $\hat{\boldsymbol{\beta}}$ by REML using \mathbf{y}_o . The joint distribution of \mathbf{u}_{n1} and \mathbf{y}_o is:

$$\begin{bmatrix} \mathbf{u}_{n1} \\ \mathbf{y}_o \end{bmatrix} \sim N \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{X}_o\boldsymbol{\beta} \end{bmatrix}, \begin{bmatrix} g_{11}\mathbf{K}_{nn} & \mathbf{g}_{1\cdot} \otimes \mathbf{K}_{no} \\ \mathbf{g}_{\cdot 1} \otimes \mathbf{K}_{on} & \mathbf{G} \otimes \mathbf{K}_{oo} + \mathbf{R} \otimes \mathbf{I} \end{bmatrix} \right)$$

Let $\mathbf{V}_o = \mathbf{G} \otimes \mathbf{K}_{oo} + \mathbf{R} \otimes \mathbf{I}$. Therefore, $E[\mathbf{u}_{n1}|\mathbf{y}_o] = (\mathbf{g}_{1\cdot} \otimes \mathbf{K}_{no})\mathbf{V}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\boldsymbol{\beta})$, so our prediction is:

$$\hat{\mathbf{u}}_{n1}^{(2)} = (\hat{\mathbf{g}}_{1\cdot} \otimes \mathbf{K}_{no})\mathbf{V}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}). \quad (10)$$

As above, to simplify this expression, we form the joint distribution of \mathbf{u}_o and \mathbf{y}_o in the training data as:

$$\begin{bmatrix} \mathbf{u}_o \\ \mathbf{y}_o \end{bmatrix} \sim N \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{X}_o\boldsymbol{\beta} \end{bmatrix}, \begin{bmatrix} \mathbf{G} \otimes \mathbf{K}_{oo} & \mathbf{G} \otimes \mathbf{K}_{oo} \\ \mathbf{G} \otimes \mathbf{K}_{oo} & \mathbf{G} \otimes \mathbf{K}_{oo} + \mathbf{R} \otimes \mathbf{I} \end{bmatrix} \right)$$

774 Therefore, $\hat{\mathbf{u}}_{o1}|\mathbf{y}_o = (\hat{\mathbf{G}} \otimes \mathbf{K}_{oo})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}})$. Rearranging and plugging this in above simplifies to: $\hat{\mathbf{u}}_{n1}^{(2)} = \mathbf{K}_{no}\mathbf{K}_{oo}^{-1}\hat{\mathbf{u}}_{o1}$.

CV2-style multi-trait predictions For our CV2-style multi-trait prediction, we take a two-step approach. We first estimate $\hat{\mathbf{u}}_o$ from the training individuals and then supplement this with \mathbf{y}_{n2} from the testing individuals. The joint distribution of \mathbf{u}_{n1} , \mathbf{y}_{n2} and \mathbf{u}_o is:

$$\begin{bmatrix} \begin{bmatrix} \mathbf{u}_{n1} \\ \mathbf{y}_{n2} \end{bmatrix} \\ \mathbf{u}_o \end{bmatrix} \sim N \left(\begin{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{X}_2\boldsymbol{\beta}_2 \end{bmatrix} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{G} \otimes \mathbf{K}_{nn} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{22} \end{bmatrix} \otimes \mathbf{I}_{nn} & \mathbf{G} \otimes \mathbf{K}_{no} \\ \mathbf{G} \otimes \mathbf{K}_{on} & \mathbf{G} \otimes \mathbf{K}_{oo} \end{bmatrix} \right)$$

Conditional on a known value of \mathbf{u}_o from the training individuals, the distribution of $\begin{bmatrix} \mathbf{u}_{n1} \\ \mathbf{y}_{n2} \end{bmatrix}$ would be:

$$\begin{bmatrix} \mathbf{u}_{n1} \\ \mathbf{y}_{n2} \end{bmatrix} | \mathbf{u}_o \sim N \left(\begin{bmatrix} \mathbf{K}_{no}\mathbf{K}_{oo}^{-1}\mathbf{u}_{o1} \\ \mathbf{X}_2\boldsymbol{\beta}_2 + \mathbf{K}_{no}\mathbf{K}_{oo}^{-1}\mathbf{u}_{o2} \end{bmatrix}, (\mathbf{G} \otimes \mathbf{K}_{nn}) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{22} \end{bmatrix} \otimes \mathbf{I}_{nn} - \left[(\mathbf{G} \otimes \mathbf{K}_{no})(\mathbf{G}^{-1} \otimes \mathbf{K}_{oo}^{-1})(\mathbf{G} \otimes \mathbf{K}_{on}) \right] \right),$$

which simplifies to:

$$\begin{bmatrix} \mathbf{u}_{n1} \\ \mathbf{y}_{n2} \end{bmatrix} | \mathbf{u}_o \sim \mathbf{N} \left(\begin{bmatrix} \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \mathbf{u}_{o1} \\ \mathbf{X}_2 \boldsymbol{\beta}_2 + \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \mathbf{u}_{o2} \end{bmatrix}, \begin{bmatrix} \mathbf{g}_{11} (\mathbf{K}^{-1})_{nn}^{-1} & \mathbf{g}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1} \\ \mathbf{g}_{21} \otimes (\mathbf{K}^{-1})_{nn}^{-1} & \mathbf{G}_{22} \otimes (\mathbf{K}^{-1})_{nn}^{-1} + \mathbf{R}_{22} \otimes \mathbf{I}_{nn} \end{bmatrix} \right).$$

Let $\mathbf{V}_c = \mathbf{G}_{22} \otimes (\mathbf{K}^{-1})_{nn}^{-1} + \mathbf{R}_{22} \otimes \mathbf{I}_{nn}$. Now, conditioning on observed values of both \mathbf{u}_o from the training data and \mathbf{y}_{n2} from the testing data, the expectation of \mathbf{u}_{n1} would be:

$$\mathbf{E}[\mathbf{u}_{n1} | \mathbf{y}_{n2}, \mathbf{u}_o] = \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \mathbf{u}_{o1} + (\mathbf{g}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1}) \mathbf{V}_c^{-1} (\mathbf{y}_{n2} - \mathbf{X}_2 \boldsymbol{\beta}_2 - \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \mathbf{u}_{o2}).$$

Using this, we form our prediction as:

$$\hat{\mathbf{u}}_{n1}^{(3)} = \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \hat{\mathbf{u}}_{o1} + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1}) \hat{\mathbf{V}}_c^{-1} (\mathbf{y}_{n2} - \mathbf{X}_2 \hat{\boldsymbol{\beta}}_2 - \mathbf{K}_{no} \mathbf{K}_{oo}^{-1} \hat{\mathbf{u}}_{o2}), \quad (11)$$

775 where $\hat{\mathbf{u}}_{o1}$ and $\hat{\mathbf{u}}_{o2}$ are extracted from the calculation of $\hat{\mathbf{u}}_o$ for the CV1-style prediction. Plugging in the solutions for these values expands
776 to:

$$\begin{aligned} \hat{\mathbf{u}}_{n1}^{(3)} &= (\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no}) \hat{\mathbf{V}}_o^{-1} (\mathbf{y}_o - \mathbf{X}_o \hat{\boldsymbol{\beta}}) \\ &+ (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1}) \hat{\mathbf{V}}_c^{-1} (\mathbf{y}_{n2} - \mathbf{X}_2 \hat{\boldsymbol{\beta}}_2 - (\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no}) \hat{\mathbf{V}}_o^{-1} (\mathbf{y}_o - \mathbf{X}_o \hat{\boldsymbol{\beta}})). \end{aligned}$$

777 Expectations of prediction accuracy

Now, we evaluate the expected correlation between a random sample of pairs of elements from our three candidate predictions and the predictand \mathbf{y}_{n1} . We compare these expected correlations with the expected “true” correlations with \mathbf{u}_{n1} . Below, let $\text{var}(\mathbf{x})$ denote the variance of a random sample from a random vector \mathbf{x} ; $\text{cov}(\mathbf{x}, \mathbf{y})$ and $\text{cor}(\mathbf{x}, \mathbf{y})$ denote the covariance and correlation between a random sample of pairs of elements from \mathbf{x} and \mathbf{y} ; and $\text{Cov}(\mathbf{x}, \mathbf{y})$ denote the covariance matrix between vectors \mathbf{x} and \mathbf{y} . We use the following results:

$$\text{cor}(\mathbf{x}, \mathbf{y}) = \frac{\text{cov}(\mathbf{x}, \mathbf{y})}{\sqrt{\text{var}(\mathbf{x})\text{var}(\mathbf{y})}} = \frac{1}{n-1} \frac{(\mathbf{x} - \boldsymbol{\mu}_x)^\top (\mathbf{y} - \boldsymbol{\mu}_y)}{\sqrt{\text{var}(\mathbf{x})\text{var}(\mathbf{y})}} = \frac{1}{n-1} \frac{\mathbf{x}^\top \mathbf{S} \mathbf{y}}{\sqrt{\text{var}(\mathbf{x})\text{var}(\mathbf{y})}}$$

where $\mathbf{S} = \mathbf{I} - \frac{\mathbf{1}\mathbf{1}^\top}{n}$.

$$\mathbf{E}[\mathbf{x}^\top \mathbf{S} \mathbf{y}] = \text{tr}(\mathbf{S} \text{Cov}(\mathbf{x}, \mathbf{y})) + \boldsymbol{\mu}_x^\top \mathbf{S} \boldsymbol{\mu}_y = \text{tr}(\mathbf{S} \text{Cov}(\mathbf{x}, \mathbf{y}))$$

where $\text{tr}(\cdot)$ is the matrix trace, and $\boldsymbol{\mu}_x = \mathbf{0}$ and/or $\boldsymbol{\mu}_y = \mathbf{0}$. Therefore, the expected correlation between \mathbf{x} and \mathbf{y} is approximately:

$$\mathbf{E}[\text{cor}(\mathbf{x}, \mathbf{y})] \approx \frac{1}{n-1} \frac{\text{tr}(\mathbf{S} \text{Cov}(\mathbf{x}, \mathbf{y}))}{\sqrt{\mathbf{E}[\text{var}(\mathbf{x})]\mathbf{E}[\text{var}(\mathbf{y})]}}.$$

778 Our goal with cross-validation is to estimate $\text{cor}(\hat{\mathbf{u}}_{n1}, \mathbf{u}_{n1})$. Since we do not know \mathbf{u}_{n1} , we approximate the correlation with
779 $\text{cor}(\hat{\mathbf{u}}_{n1}, \mathbf{y}_{n1}) / \sqrt{h_1^2}$. The factor of $\sqrt{h_1^2}$ corrects the correlation for the larger variance of \mathbf{y}_{n1} relative to \mathbf{u}_{n1} . Otherwise, any differ-
780 ence between these two correlations must be due to their numerators: $\text{tr}(\mathbf{S} \text{Cov}(\hat{\mathbf{u}}_{n1}, \mathbf{u}_{n1}))$ and $\text{tr}(\mathbf{S} \text{Cov}(\hat{\mathbf{u}}_{n1}, \mathbf{y}_{n1}))$. Thus, for each of the
781 three prediction methods we compare these two numerators to evaluate the accuracy and bias in the approximation.

Single trait predictions The numerator of the expected correlation between $\mathbf{u}_{n1}^{(1)}$ and the true genetic values \mathbf{u}_{n1} is:

$$\begin{aligned} \text{tr}(\mathbf{S} \text{Cov}(\hat{\mathbf{u}}_{n1}^{(1)}, \mathbf{u}_{n1})) &= \text{tr} \left(\mathbf{S} \text{Cov}(\hat{\mathbf{g}}_{11} \mathbf{K}_{no} \hat{\mathbf{V}}_{o1}^{-1} (\mathbf{y}_{o1} - \mathbf{X}_{o1} \hat{\boldsymbol{\beta}}_1), \mathbf{u}_{n1}) \right) \\ &= \text{tr} \left(\hat{\mathbf{g}}_{11} \mathbf{S} \mathbf{K}_{no} \hat{\mathbf{V}}_{o1}^{-1} \text{Cov}(\mathbf{u}_{o1} + \mathbf{e}_{o1}, \mathbf{u}_{n1}) \right) \\ &= \text{tr} \left(\hat{\mathbf{g}}_{11} \mathbf{S} \mathbf{K}_{no} \hat{\mathbf{V}}_{o1}^{-1} (\mathbf{g}_{11} \mathbf{K}_{on}) \right) \\ &= \hat{\mathbf{g}}_{11} \mathbf{g}_{11} \text{tr} \left(\mathbf{S} \mathbf{K}_{no} \hat{\mathbf{V}}_{o1}^{-1} \mathbf{K}_{on} \right). \end{aligned}$$

where we assume that $\hat{\beta}_1 = \beta_1$ and $Cov(\mathbf{e}_{o1}, \mathbf{u}_{n1}) = \mathbf{0}$. The same result for the numerator of the expected correlation between $\mathbf{u}_{n1}^{(1)}$ and the observed phenotypic values \mathbf{y}_{n1} is:

$$\begin{aligned} tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(1)}, \mathbf{y}_{n1})) &= tr\left(\mathbf{S}Cov(\hat{g}_{11}\mathbf{K}_{no}\hat{\mathbf{V}}_{o1}^{-1}(\mathbf{y}_{o1} - \mathbf{X}_{o1}\hat{\beta}_1), \mathbf{y}_{n1})\right) \\ &= tr\left(\hat{g}_{11}\mathbf{S}\mathbf{K}_{no}\hat{\mathbf{V}}_{o1}^{-1}Cov(\mathbf{u}_{o1} + \mathbf{e}_{o1}, \mathbf{u}_{n1} + \mathbf{e}_{n1})\right) \\ &= tr\left(\hat{g}_{11}\mathbf{S}\mathbf{K}_{no}\hat{\mathbf{V}}_{o1}^{-1}(g_{11}\mathbf{K}_{on})\right) \\ &= \hat{g}_{11}g_{11}tr\left(\mathbf{S}\mathbf{K}_{no}\hat{\mathbf{V}}_{o1}^{-1}\mathbf{K}_{on}\right), \end{aligned}$$

782 where we additionally assume $Cov(\mathbf{u}_{o1}, \mathbf{e}_{n1}) = \mathbf{0}$ and $Cov(\mathbf{e}_{o1}, \mathbf{e}_{n1}) = \mathbf{0}$. Therefore, the numerators are the same, and $cor(\hat{\mathbf{u}}_{n1}^{(1)}, \mathbf{y}_{n1})/\sqrt{\hat{h}_1^2}$ is
783 a consistent estimator for $cor(\hat{\mathbf{u}}_{n1}^{(1)}, \mathbf{u}_{n1})$.

784 **CV1-style multi-trait predictions** The numerator of the expected correlation between $\mathbf{u}_{n1}^{(2)}$ and the true genetic values \mathbf{u}_{n1} is:

$$\begin{aligned} tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(2)}, \mathbf{u}_{n1})) &= tr(\mathbf{S}Cov((\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\beta}), \mathbf{u}_{n1})) \\ &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{n1})) \\ &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(g_{11} \otimes \mathbf{K}_{on})), \end{aligned}$$

785 again assuming $\hat{\beta} = \beta$ and now also $Cov(\mathbf{e}_o, \mathbf{u}_{n1}) = \mathbf{0}$. The same result for the numerator of the expected correlation between $\mathbf{u}_{n1}^{(2)}$ and the
786 observed phenotypic values \mathbf{y}_{n1} is:

$$\begin{aligned} tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(2)}, \mathbf{y}_{n1})) &= tr(\mathbf{S}Cov((\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{oo})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\beta}), \mathbf{y}_{n1})) \\ &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{oo})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{n1} + \mathbf{e}_{n1})) \\ &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{oo})\hat{\mathbf{V}}_o^{-1}(g_{21} \otimes \mathbf{K}_{on})), \end{aligned}$$

787 where we additionally assume $Cov(\mathbf{u}_o, \mathbf{e}_{n1}) = \mathbf{0}$ and $Cov(\mathbf{e}_o, \mathbf{e}_{n1}) = \mathbf{0}$. Therefore, the numerators are the same, and $cor(\hat{\mathbf{u}}_{n1}^{(2)}, \mathbf{y}_{n1})/\sqrt{\hat{h}_1^2}$ is a
788 consistent estimator for $cor(\hat{\mathbf{u}}_{n1}^{(2)}, \mathbf{u}_{n1})$.

CV2-style multi-trait predictions The numerator of the expected correlation between $\mathbf{u}_{n1}^{(3)}$ and the true genetic values \mathbf{u}_{n1} is:

$$\begin{aligned}
 tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1})) &= tr(\mathbf{S}[Cov\left((\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}) \right. \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}) \\
 &\quad \left. + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{y}_{n2} - \mathbf{X}_2\hat{\boldsymbol{\beta}}_2), \mathbf{u}_{n1}\right)]) \\
 &= tr(\mathbf{S}[Cov((\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}), \mathbf{u}_{n1}) \\
 &\quad - Cov((\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}), \mathbf{u}_{n1}) \\
 &\quad + Cov((\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{y}_{n2} - \mathbf{X}_2\hat{\boldsymbol{\beta}}_2), \mathbf{u}_{n1})]) \\
 &= tr(\mathbf{S}[(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{n1}) \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{n1}) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}Cov(\mathbf{u}_{n2} + \mathbf{e}_{n2}, \mathbf{u}_{n1})]) \\
 &= tr(\mathbf{S}[(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on}) \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on}) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{nn})]) \\
 &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on})) \\
 &\quad - tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on})) \\
 &\quad + tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{nn})),
 \end{aligned}$$

again assuming $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}$, $Cov(\mathbf{e}_o, \mathbf{u}_{n1}) = \mathbf{0}$, and $Cov(\mathbf{e}_{n2}, \mathbf{u}_{n1}) = \mathbf{0}$. From this, we can see the potential benefit of the CV2-style method:

$$\begin{aligned}
 tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1})) - tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(2)}, \mathbf{u}_{n1})) \\
 &= tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{nn})) - tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{on})) \\
 &= tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{nn} - (\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{on}))),
 \end{aligned}$$

789 which is generally (but maybe not necessarily) positive. This means that $cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1})$ is generally greater than $cor(\hat{\mathbf{u}}_{n1}^{(2)}, \mathbf{u}_{n1})$.

The same result for the numerator of the expected correlation between $\mathbf{u}_{n1}^{(3)}$ and the observed phenotypic values \mathbf{y}_{n1} is:

$$\begin{aligned}
 tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{n1})) &= tr(\mathbf{S}[Cov((\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}), \mathbf{u}_{n1} + \mathbf{e}_{n1}) \\
 &\quad - Cov((\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}), \mathbf{u}_{n1} + \mathbf{e}_{n1}) \\
 &\quad + Cov((\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{y}_{n2} - \mathbf{X}_2\hat{\boldsymbol{\beta}}_2), \mathbf{u}_{n1} + \mathbf{e}_{n1})]) \\
 &= tr(\mathbf{S}[(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{n1} + \mathbf{e}_{n1}) \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{n1} + \mathbf{e}_{n1}) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}Cov(\mathbf{u}_{n2} + \mathbf{e}_{n2}, \mathbf{u}_{n1} + \mathbf{e}_{n1})]) \\
 &= tr(\mathbf{S}[(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on}) \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on}) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{nn})]) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{r}_{21} \otimes \mathbf{I})]) \\
 &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on}) \\
 &\quad - tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_1 \otimes \mathbf{K}_{on}) \\
 &\quad + tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{nn}) \\
 &\quad + tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{r}_{21} \otimes \mathbf{I}_{nn}))),
 \end{aligned}$$

From this, we see that the numerator of the correlation $cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{n1})$ is not equal to that of $cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1})$:

$$tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{n1})) - tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1})) = tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{r}_{21} \otimes \mathbf{I}_{nn})).$$

790 If $p = 1$, then $\hat{\mathbf{g}}_{12}$ and \mathbf{r}_{12} are scalars and this excess covariance is approximately $n\hat{\mathbf{g}}_{12}\mathbf{r}_{12}$.

791 **CV2* approach** In our new CV2* cross-validation approach, we replace \mathbf{y}_{n1} with \mathbf{y}_{x1} —the phenotypes of a new set of individuals (x) that are
 792 relatives of the testing partition and were not part of the training partition. Let \mathbf{K}_{xx} be the genetic relationships among these n_x individuals,
 793 and \mathbf{K}_{xo} be their genetic relationships with the training partition. The numerator of the expected correlation $cor(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{x1}) / \sqrt{h_1^2}$ is:

$$\begin{aligned}
 tr(\mathbf{S}Cov(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{x1})) &= tr(\mathbf{S}[Cov((\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}), \mathbf{u}_{x1} + \mathbf{e}_{x1}) \\
 &\quad - Cov((\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{y}_o - \mathbf{X}_o\hat{\boldsymbol{\beta}}), \mathbf{u}_{x1} + \mathbf{e}_{x1}) \\
 &\quad + Cov((\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{y}_{n2} - \mathbf{X}_2\hat{\boldsymbol{\beta}}_2), \mathbf{u}_{x1} + \mathbf{e}_{x1})]) \\
 &= tr(\mathbf{S}[(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{x1} + \mathbf{e}_{x1}) \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}Cov(\mathbf{u}_o + \mathbf{e}_o, \mathbf{u}_{x1} + \mathbf{e}_{x1}) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}Cov(\mathbf{u}_{n2} + \mathbf{e}_{n2}, \mathbf{u}_{x1} + \mathbf{e}_{x1})]) \\
 &= tr(\mathbf{S}[(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{ox}) \\
 &\quad - (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{ox}) \\
 &\quad + (\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{xx})]) \\
 &= tr(\mathbf{S}(\hat{\mathbf{g}}_1 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{ox}) \\
 &\quad - tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\hat{\mathbf{G}}_2 \otimes \mathbf{K}_{no})\hat{\mathbf{V}}_o^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{ox}) \\
 &\quad + tr(\mathbf{S}(\hat{\mathbf{g}}_{12} \otimes (\mathbf{K}^{-1})_{nn}^{-1})\hat{\mathbf{V}}_c^{-1}(\mathbf{g}_{21} \otimes \mathbf{K}_{xx}))).
 \end{aligned}$$

794 If these new individuals are clones of the original testing set, then $\mathbf{K}_{xx} = \mathbf{K}_{nn}$, $\mathbf{K}_{ox} = \mathbf{K}_{on}$ and $tr(\mathbf{SCov}(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{y}_{x1})) = tr(\mathbf{SCov}(\hat{\mathbf{u}}_{n1}^{(3)}, \mathbf{u}_{n1}))$.
795 However, if clones are not available, then this equality will not hold.
796 Given these analytical results for the numerator of the expected correlations, we can estimate the correlation itself by calculating the
797 expected variances of $\hat{\mathbf{u}}_{n1}$ and \mathbf{u}_{n1} or \mathbf{y}_{n1} . We do not go through these calculations as they follow directly from the calculations given above.