ORIGINAL ARTICLE

Journal of
Animal Breeding and Genetics   **WILEY**

Impact of multi-output and stacking methods on feed efficiency prediction from genotype using machine learning algorithms

Mónica Mora[1,2] 🄳 | **Pablo González**[3] 🄳 | **José Ramón Quevedo**[3] 🄳 |
Elena Montañés[3] 🄳 | **Llibertat Tusell**[2] 🄳 | **Rob Bergsma**[4] 🄳 | **Miriam Piles**[2] 🄳

[1]Departamento de Ciencia Animal, Universidad Politècnica de València, Valencia, Spain

[2]Animal Breeding and Genetics, Institute of Agrifood Research and Technology (IRTA), Barcelona, Spain

[3]Artificial Intelligence Centre, University of Oviedo, Gijón, Spain

[4]Topigs Norsvin Research Center, Beuningen, Netherlands

Correspondence
Mónica Mora, Departamento de Ciencia Animal, Universidad Politècnica de València, Valencia, Spain.
Email: momofe@etsiamn.upv.es

Abstract

Feeding represents the largest economic cost in meat production; therefore, selection to improve traits related to feed efficiency is a goal in most livestock breeding programs. Residual feed intake (RFI), that is, the difference between the actual and the expected feed intake based on animal's requirements, has been used as the selection criteria to improve feed efficiency since it was proposed by Kotch in 1963. In growing pigs, it is computed as the residual of the multiple regression model of daily feed intake (DFI), on average daily gain (ADG), backfat thickness (BFT), and metabolic body weight (MW). Recently, prediction using single-output machine learning algorithms and information from SNPs as predictor variables have been proposed for genomic selection in growing pigs, but like in other species, the prediction quality achieved for RFI has been generally poor. However, it has been suggested that it could be improved through multi-output or stacking methods. For this purpose, four strategies were implemented to predict RFI. Two of them correspond to the computation of RFI in an indirect way using the predicted values of its components obtained from (i) individual (multiple single-output strategy) or (ii) simultaneous predictions (multi-output strategy). The other two correspond to the direct prediction of RFI using (iii) the individual predictions of its components as predictor variables jointly with the genotype (stacking strategy), or (iv) using only the genotypes as predictors of RFI (single-output strategy). The single-output strategy was considered the benchmark. This research aimed to test the former three hypotheses using data recorded from 5828 growing pigs and 45,610 SNPs. For all the strategies two different learning methods were fitted: random forest (RF) and support vector regression (SVR). A nested cross-validation (CV) with an outer 10-folds CV and an inner threefold CV for hyperparameter tuning was implemented to test all strategies. This scheme was repeated using as predictor variables different subsets with an increasing number (from 200 to 3000) of the most informative SNPs identified with RF. Results showed that the highest prediction performance was achieved with 1000 SNPs, although the stability of feature selection was poor (0.13

points out of 1). For all SNP subsets, the benchmark showed the best prediction performance. Using the RF as a learner and the 1000 most informative SNPs as predictors, the mean (SD) of the 10 values obtained in the test sets were: 0.23 (0.04) for the Spearman correlation, 0.83 (0.04) for the zero–one loss, and 0.33 (0.03) for the rank distance loss. We conclude that the information on predicted components of RFI (DFI, ADG, MW, and BFT) does not contribute to improve the quality of the prediction of this trait in relation to the one obtained with the single-output strategy.

KEYWORDS

artificial intelligence, multi-trait, regression problem, residual feed intake, SNPs, stacking

1 | INTRODUCTION

The impact of rising feeding costs and the need to find new ways to reduce emissions to the environment by making sustainable use of resources make improving feed efficiency (FE) one of the most important objectives of meat production. Koch et al. (1963) proposed a measurement of FE known as residual feed intake (RFI). This trait is defined for each animal as the difference between its actual and expected feed intake, which is estimated based on its feed requirements for maintenance of physiological functions and weight gain. Thus, RFI is estimated as the residual of a linear regression model of daily feed intake (DFI) on average daily gain (ADG), backfat thickness (BFT), and metabolic weight (MW). However, the availability of an effective measure for RFI can be difficult and expensive due to the need for individual measurements of feed intake for all animals. Consequently, predicting FE without measuring feed intake is a challenge in many breeding programs.

With the availability of high-density nucleotide polymorphism (SNP) data, several genomic models can be used to evaluate selection candidates to improve FE. For this purpose, machine learning (ML) methods are an appealing alternative since they do not require assumptions about the genetic determinism underlying a trait and they can be implemented when the number of parameters is much larger than the number of observations as is the case of the genomic analysis.

Machine learning methods have been successfully used in livestock and plant breeding to predict important economic traits using dense molecular markers as predictors. Some examples are milk yield in dairy cattle (Long et al., 2011) and reproductive traits in pigs (Wang et al., 2022). For the prediction of RFI, ML methods have been reported to be suitable when subsets of the most informative SNPs are used as predictor variables. Tusell et al. (2020) and Piles et al. (2021) predicted FE using different sources of phenotypic and genotypic information as well as different algorithms for SNP selection. In these studies, as well as in most genomic selection research, predictive models only use single phenotypes, building an independent model for each target variable and ignoring the relationship among them. Contrary to single-output models, multi-output models predict all the target variables simultaneously exploiting possible existing genetic correlations. A review of the state of the art of multi-output regression problems is presented by Borchani et al. (2015). The main benefits of multi-output methods over single-output methods are (i) less computation requirements since a single estimator is built, (ii) more efficient use of information since multi-output methods exploit possible dependencies between different target variables in addition to relationships among the features and the targets (Blockeel et al., 2000; Struyf & Džeroski, 2006), and (iii) a potential improvement of the predictive performance in comparison to single-output methods when multiple targets are predicted simultaneously under the existence of relationships among target variables (Burnham et al., 1999; Han et al., 2012).

Inspired by multi-label classification approaches, Spyromitros-Xioufis et al. (2012) proposed another algorithm for multi-target regression: the stacking method. At the training phase, it consists of a two-stage process. In the first stage, multiple single-output models are fitted (base models), one per each of the target variables. In the second stage, the outputs of the first stage are used as inputs to fit another single-output regression model called meta-model. During this last stage, the algorithm learns the optimal combination of the predictions of the base models to improve the prediction quality of the meta-model. In some cases, the meta-model can also include the inputs of the base models as predictor variables. At the prediction phase, the algorithm returns the outputs of the meta-model. Stacking methods have been shown to increase the predictive performance by 7.70% compared with GBLUP (a widely used method for genomic selection), which is very relevant for animal and plant breeding (Liang et al., 2021).

The goal of this work was to exploit the advantages of multi-output and stacking methods to find the best strategy to predict the individual RFI from the genotype in a pig population. For this purpose, we used several metrics to evaluate the prediction performance of RFI obtained from two general strategies. In the first one, the RFI was obtained as the residual of the regression model of predicted DFI on predicted ADG, BFT, and MW, all of them obtained (i) from individual predictions from the genotype (multiple single-output strategy) or (ii) from simultaneously predictions from the genotype (multi-output strategy). In the second one, the RFI was predicted directly (iii) using the individual predictions of DFI, ADG, BFT, and MW and the genotypes as predictors of RFI (stacking strategy) or (iv) using only the genotypes as predictors of RFI (single-output strategy). This last strategy was used as the benchmark.

2 | MATERIALS AND METHODS

2.1 | Animals

All the animals were born and raised in two specific pathogen-free nucleus farms for a terminal sire line of Topigs Norsvin (Vught). One of the farms was located in the Netherlands and the other one in France. Semen exchange between the two farms was frequent. Both nucleus farms were equipped with IVOG feeding stations (INSENTEC) that daily registered individual feed intake for all pigs. The pigs were fed ad libitum with a commercially available diet until the end of the performance test. Data on 5828 male pigs were available.

2.2 | Phenotypes

The test period was from 68 to 155 days (31–130 kg of body weight) of the median. Daily feed intake (DFI, g/day) was calculated as the total feed intake during this period divided by the number of days elapsed. Average daily gain (ADG, g/day) was calculated as the difference in body weight recorded at the beginning and the end of the test divided by the number of days elapsed. Only records from animals that started the test period between 50 and 105 days of age and remained on the test period between 60 and 120 days were retained for the analyses. Back fat thickness (BFT) was measured ultrasonically on live animals (US-fat in mm) at the end of the test period. Metabolic weight (MW, g) was calculated as $MW = \frac{(W_{start} + W_{end})^{0.75}}{2}$, where W_{start} and W_{end} are the body weight at the beginning and the end of the test period, respectively. All these records were pre-adjusted by environmental effects fitting a linear model which included the fixed effects of age at the start of the test (Age, covariate), duration of the performance test (Length, covariate), and the combination of farm and batch (FarmBatch, 46 levels). The FarmBatch effect resulted from the combination of 2 farms and 2-month period batches. Only FarmBatch levels with more than 10 records each were retained for the analysis. Thus, the model to adjust the records for animal i-th, level of FarmBatch j-th, and trait kth can be written as:

$$y_{ijk} = \text{FarmBatch}_{jk} + \beta_{1k} \times \text{Age}_i + \beta_{2k} \times \text{Length}_i + e_{ijk},$$

where y_{ijk} is the individual record with k = DFI, ADG, BFT, and MW; β_{1k} is the regression coefficient of trait k on age; β_{2k} is the regression coefficient of trait k on Length, and e_{ijk} is the residual term. All the other terms are defined above. The adjusted records were obtained as the residuals from the corresponding model for each trait. These linear models were fitted using the $lm()$ function of R (R Core Team, 2022).

2.3 | Genotypes

Animals were genotyped using the Illumina Porcine SNP60 BeadChip (Illumina Inc.). Assuming an additive allele substitution effect, genotypes were arbitrarily coded to 0, 1, and 2 for the homozygote for the minor allele (i.e., aa), heterozygote (i.e., Aa) and the other homozygote (i.e., AA), respectively. SNPs with a call rate lower than 0.90 and with a minor allele frequency lower than 0.05 were removed. Animals with a call rate lower than 0.90 and parent–offspring pairs that showed Mendelian inconsistencies were rejected. After this quality control, the number of SNPs remaining for further analysis was 45,610 and the number of animals was 5708. Then, SNPs were binarized to 0|0 for the homozygote for the minor allele (coded as 0), 1|0 for the heterozygote (coded as 1) and 1|1 for the other homozygote (coded as 2), thus the number of features was doubled. This step was carried out to not consider the genotypes coding as a magnitude (see Figure 1).

2.4 | Model fitting and evaluation

A nested resampling (see Figure 2) was implemented to obtain reliable performance estimates for the learners and to quantify the generalization ability of the model.

This method consists of two nested resampling loops. In the outer resampling loop, a 10-fold cross-validation (CV) was carried out by randomly dividing the dataset into 10 groups of equal size. One group was used as an
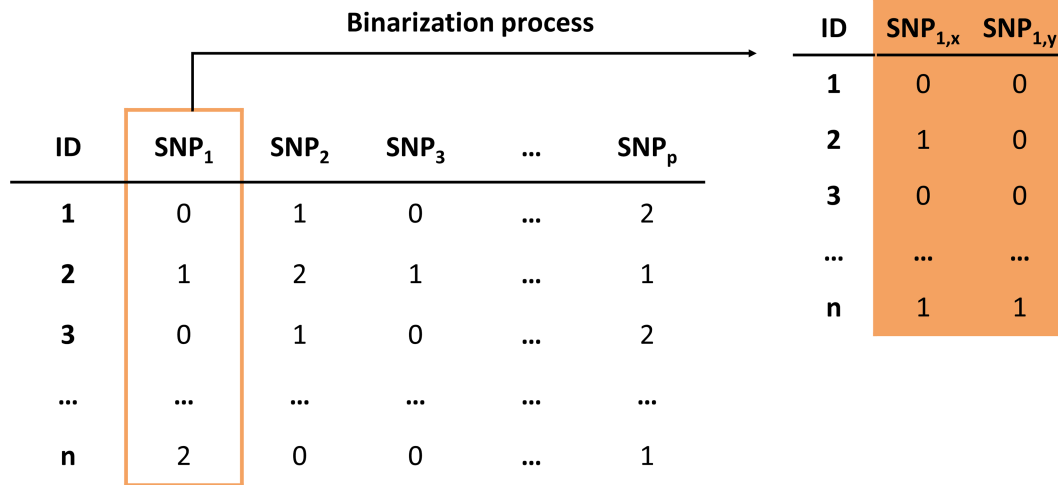
Binarization process

ID	SNP$_1$	SNP$_2$	SNP$_3$	...	SNP$_p$
1	0	1	0	...	2
2	1	2	1	...	1
3	0	1	0	...	2
...	...	...	...	...	...
n	2	0	0	...	1

ID	SNP$_{1,x}$	SNP$_{1,y}$
1	0	0
2	1	0
3	0	0
...	...	...
n	1	1

FIGURE 1 Schema of the binarization process for one SNP. In columns: ID (unique animal identification, from 1 until n animals) and SNP (single-nucleotide polymorphism, from 1 until p). 0: genotype for the homozygote for the minor allele (i.e., aa), 1: genotype for the heterozygote (i.e., Aa) and 2: genotype for the other homozygote (i.e., AA).
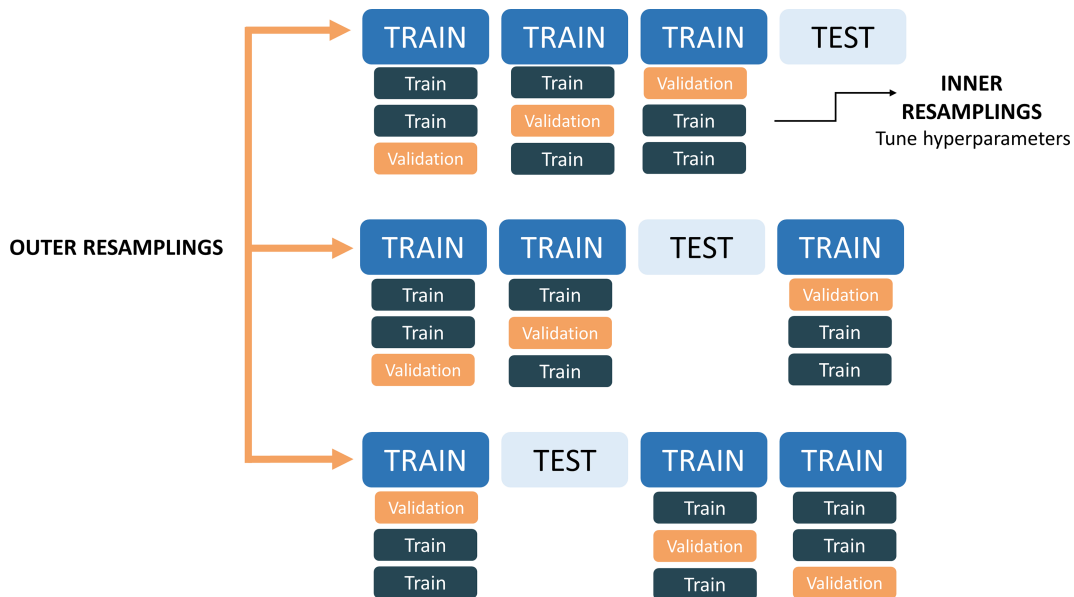

FIGURE 2 Nested k-fold cross-validation diagram.

outer test set and the remaining nine groups were used as an outer training set. The train set was divided into three-folds using an inner CV. One fold is used as a validation set and the rest as an inner train set. The evaluation of this inner CV will be used in a grid-search procedure to find the optimum hyperparameters of the machine learning systems. The grid-search method was implemented with the *GridSearchCV()* function from the scikit-learn package (Fabian Pedregosa Gaël Varoquaux et al., 2011). This is a process by which a model is built and evaluated with every combination (discrete grid) of the manually specified subset of values for each hyperparameter. The optimal hyperparameter values were chosen based on the mean absolute error on the validation set of this inner CV.

The outer CV was repeated 10 times, each one with a different group of data used as a test set, resulting in a total of 10 pairs of training/test sets. Within each outer training set, RFI was estimated as the residual of a phenotypic linear regression of the observed values of DFI on ADG, BFT, and MW. Thus, for animal *i*-th:

$$DFI_i = \beta_1 \times ADG_i + \beta_2 \times BFT_i + \beta_3 \times MW_i + RFI_i.$$

The linear model was fitted using the *ols()* function from the stats models package in Python (Seabold & Perktold, 2010). The model fitted in each outer training set was used to compute RFI in each outer testing set. In addition, in each outer training set, feature selection was carried out selecting subsets of the most informative features

—WILEY—

(i.e., SNPs) in a regression performed with Random Forest (RF) (explained in the "Learner" section below). Machine learning algorithms were tested with different subsets of data that contained an increasing number (from 200 to 3000 by 200) of the most informative SNPs carefully and strategically selected by RF. The process of feature selection was carried out independently for each outer CV iteration and for each variable (DFI, ADG, BFT, MW, and RFI), including an extra one to select the most important features that predict DFI, ADG, BFT, and MW simultaneously. Then, for each SNPs subset, a RF and a Support Vector for Regression (SVR) (explained in detail in the "Learner" section below) were fitted to all training data using the optimal hyperparameters. This scheme and the same data split were used to compare prediction performance in the same conditions for the different strategies and the different number of features selected (i.e., most informative SNPs).

Two general strategies were implemented to predict RFI. The first one consists in the indirect prediction of RFI which requires predicting their components in a previous step (strategies i and ii) and the second one consists in predict RFI directly (strategies iii and iv) (see Figure 3). Here is a description of all four strategies:

I Multiple single-output: DFI, ADG, BFT, and MW were predicted individually using information from the genotype. Then, the prediction of RFI was calculated as the difference between the predicted DFI and the value that results from applying the multiple regression model obtained from the corresponding training set to the predicted values of ADG, BFT, and MW.

II Multi-output: DFI, ADG, BFT, and MW were predicted simultaneously using information from the genotype. Then, the prediction of RFI was calculated as the difference between the predicted DFI and the value that results from applying the multiple regression model obtained from the corresponding training set to the predicted values of ADG, BFT, and MW.

III Stacking: This strategy consists of two consecutive prediction procedures. In the first one (base models), DFI, ADG, BFT, and MW are predicted individually from the genotype. Then, these individual predictions and the genotype are used as predictors to predict RFI (meta-model).

IV Single-output: The prediction of RFI is made using only the SNPs as features. This strategy was used as the benchmark.

2.5 | Predictive performance metrics

Given the actual values of n test samples $y = (y_1, \ldots y_n)$ and their predictions $\hat{y} = (\hat{y}_1, \ldots \hat{y}_n)$, the predictive performance of the models was evaluated using several

RECORDS
Phenotypes: DFI, ADG, MW, BFT ⟶ RFI = DFI − β_1ADG − β_2MW − β_3BFT
Genotypes: SNPs

	STRATEGY		FEATURES	OUTPUT	SELECTION CRITERION
Indirect prediction	i)	Multiple single-output	SNPs SNPs SNPs SNPs	$\widehat{DFI}$ $\widehat{ADG}$ $\widehat{MW}$ $\widehat{BFT}$	RFI = $\widehat{DFI}$ − $\beta_1\widehat{ADG}$ − $\beta_2\widehat{MW}$ − $\beta_3\widehat{BFT}$
	ii)	Multi-output	SNPs	$\widehat{DFI}, \widehat{ADG}, \widehat{MW}, \widehat{BFT}$	RFI = $\widehat{DFI}$ − $\beta_1\widehat{ADG}$ − $\beta_2\widehat{MW}$ − $\beta_3\widehat{BFT}$
Direct prediction	iii)	Stacking	Base models: SNPs SNPs SNPs SNPs	$\widehat{DFI}$ $\widehat{ADG}$ $\widehat{MW}$ $\widehat{BFT}$	
			Meta-model: SNPs+$\widehat{DFI}$ + $\widehat{ADG}$ + $\widehat{MW}$ + $\widehat{BFT}$	$\widehat{RFI}$	$\widehat{RFI}$
	iv)	Single-output	SNPs	$\widehat{RFI}$	$\widehat{RFI}$

FIGURE 3 Descriptive graph illustrating the four strategies applied in this study. DFI: daily feed intake, ADG: average daily gain, MW: metabolic weight, BFT: back fat thickness, RFI: residual feed intake and SNP: single-nucleotide polymorphism. The selection criterion refers to the variable used to rank the selection candidates.

regression and ranking metrics. As regression metrics, we used the mean absolute error (MAE) defined as the difference between the observed and the predicted value:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

and the relative absolute error (RAE):

$$RAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{\sum_{i=1}^{n} |y_i - \overline{y}_{\text{train}}|},$$

where $\overline{y}_{\text{train}}$ is the sample mean of the actual value y of the training samples.

As ranking functions, we used the Spearman correlation between the ranking of the observed and the predicted value

$$\text{Spearman correlation} = 1 - \frac{6 \sum_{i=1}^{n} \left(\text{pos}(y_i) - \text{pos}(\hat{y}_i)\right)^2}{n(n^2 - 1)}$$

and two-loss functions based on the same approach: zero–one loss and rank distance loss. These measurements consider only the top X% in a ranking of the total samples. For each predicted value that is part of the top X%, the loss is not incremented if the real value is also in that top X%. Nevertheless, if the predicted value does not satisfy that criterion, we would sum 1 to the loss function (zero–one loss) or an amount equal to the difference between the real and the predicted position in the ranking of all values (rank distance loss):

$$\text{zero} - \text{one loss} = \frac{1}{n\text{TOP}} \times \sum_{i=1}^{n\text{TOP}} \begin{cases} 1 & \leftarrow & \text{pos}(\hat{y}_i) > n\text{TOP} \\ 0 & \leftarrow & \text{otherwise} \end{cases},$$

$$\text{rank distance loss} = \frac{1}{n\text{TOP}} \times \sum_{i=1}^{n\text{TOP}} \begin{cases} \dfrac{\text{pos}(y_i) - \text{pos}(\hat{y}_i)}{n-1} & \leftarrow & \text{pos}(\hat{y}_i) > n\text{TOP} \\ 0 & \leftarrow & \text{otherwise} \end{cases},$$

being $n\text{TOP}$ the number of samples in the top X% of the samples and pos the position in the ranking. The closer to zero, these losses are the better performance. For this study, we set X to 10 assuming that the best 10% of the selection candidates were selected.

2.6 | Feature selection stability

Feature selection stability was estimated to find the smallest and the most stable subset of SNPs that leads to the best performance, removing either noisy or irrelevant features. The concept of stability of feature selection was defined for the first time by Kalousis et al. (2005). An algorithm is stable if a small change in data does not lead to a large change in the subset of SNPs selected. Different measures to estimate the stability of feature selectors have been discussed in the literature (Khaire & Dhanalakshmi, 2019). Nogueira and Brown (2016) defined the five properties that every stability measure should meet: (i) to allow variation in the number of features selected; (ii) to be a decreasing function with regard to the sample variance; (iii) to be upper/lower bounded by constants independent of the number of features selected; (iv) to achieve its maximum only when all selected feature sets across training sets are identical and; (v) to be corrected for a chance. According to these properties, Nogueira et al. (2018) proposed a new stability estimator defined as:

$$\hat{\Phi}(S) = 1 - \frac{\frac{1}{p} \sum_{i=1}^{p} s_f^2}{E\left[\frac{1}{p} \sum_{i=1}^{p} s_f^2 \mid H_0\right]} = 1 - \frac{\frac{1}{p} \sum_{i=1}^{p} s_f^2}{\frac{\overline{d}}{p}\left(1 - \frac{\overline{d}}{p}\right)},$$

where p is the total number of SNPs (i.e 45,610), $\overline{d}$ is the average number of features selected over the M feature sets (in this case $M = 10$, one feature set per each outer fold) and $s_f^2 = \frac{M}{M-1} \hat{p}_f \left(1 - \hat{p}_f\right)$ is the unbiased sample variance of the fth SNP, being $\hat{p}_f$ the mean of the frequency of the feature f.

Using the above stability measurement, the stability was estimated for each SNPs subset which ranges from −1 to 1, with 1 being the value for the most stable system.

2.7 | Learners

Support vectors for regression and RF were taken as learners to predict RFI in all strategies, except for the multi-output strategy for which only RF was adopted because SVR, in contrast to RF, does not have a natural way to extend to multi-output regression.

A support vector for regression is a type of support vector machine that is used for predicting continuous variables. The objective of SVR is to find the flattest hyperplane that includes the maximum number of points corresponding to observed data within a pre-defined threshold which is the maximum error (epsilon, ε). The algorithm uses a kernel function (linear, quadratic, radial basis function, etc.) whose purpose is to express the similarity between two vectors mapping lower dimensional data into

higher dimensional data. A kernel function returns the inner product between two vectors in some transformed or feature space (which can be of infinite dimension). In general, a kernel can be expressed as $k(x, z) = \varphi(x)'\varphi(z)$, where x and z are two vectors in the original space and $\varphi(x)$ and $\varphi(z)$ are the vectors in the transformed feature space. In the case of the linear kernel, the kernel function would be the original function, that is, the inner product between the two vectors. In this study, we applied a linear kernel.

Given a set of training data $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)$ where $\boldsymbol{x}_i \in R^p, y_i \in R$, $\boldsymbol{x}_i$ is p-dimensional input vector (i.e., genotypic codes of p SNPs) and y_i is the phenotypic value. It can be assumed that $f(\boldsymbol{x})$ is a linear function of form $f(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{x} + b$, where $\boldsymbol{w}$ is the vector of regression coefficients and b is the bias. In SVR, the objective is to find a function $f(\boldsymbol{x})$ that minimizes the following regularized loss function:

$$\min\left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} L(e_i)\right),$$

where $\|w\|$ is the l2-norm of the coefficient vector $\mathbf{w}$ which represents the flatness of the function, $e_i = y_i - f(x_i)$ is the error associated with each training data point, C is a positive regularization parameter that controls the trade-off between the complexity of the model and the training error, and L denotes the loss function:

$$L_\varepsilon(e) = \begin{cases} 0 \text{ if } |e| < \varepsilon \\ |e| - \varepsilon \text{ otherwise} \end{cases}.$$

A deep explanation of this method can be found in (Smola & Schölkopf, 2004). For the analysis, we used a linear kernel and ε equal to 0.1. The cost hyperparameter C was optimized through a grid-search procedure over the set of values {0.001, 0.01, 0.1, 1}.

Random forest is a bagging technique of a multitude of independent decision trees at training time and outputting the mean prediction of the individual trees. Decision trees start with the root of the tree and follow splits based on variable outcomes until a leaf node is reached and the result is given. In the case of multi-output regression, the leaves store a vector instead of storing a single value. Each component of the vector is the prediction of each target variable. The criterion used to measure the quality of the split was the mean squared error (MSE):

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

where n is the number of data points, f_i is the value returned by the model and y_i is the actual value for data point (x_i, y_i). Different values were optimized through a grid search: the minimum number of samples required to split an internal node {2, 5, 10} and the minimum number of samples required to be at the leaf node {2, 4}. The number of trees was set up to 500 and the maximum depth of the tree to 16. Random forest was also used for feature selection, with the same grid-search but changing the number of trees to 100 and the maximum depth of the tree to 2 and 4. Feature importance is calculated as the decrease in the node of the mean absolute error weighted by the probability of reaching that node. The node probability can be calculated by the number of samples that reach the node, divided by the total number of samples. The higher the value the more important the feature is. A further explanation of this algorithm can be found in (Breiman, 2001).

The analysis was performed by using the package scikit-learn (Fabian Pedregosa Gaël Varoquaux, 2011). The SVR and RF algorithms were implemented with the *SVR* and *RandomForestRegressor* functions, respectively.

3 | RESULTS

3.1 | Stability of feature selection

The stability of feature selection for each SNP subset is presented in Figure 4. The results indicate that the stability of the model increases as the number of SNPs selected increases, ranging from 0.17 for 400 SNPs to 0.45 for 3000 SNPs. However, as we will describe in detail in the following sections, the quality of the prediction increases to reach a maximum with subsets between 1000 and 1200 SNPs remaining constant beyond this maximum. Therefore, for the sake of simplicity, only results for subsets of up to 1400 SNPs will be shown. To compare the prediction performance of the two strategies and the benchmark, we focus on the results obtained using the 1000 SNP subset (stability = 0.13).

3.2 | Prediction performance

Results correspond to the predictive performance of RFI achieved following the four proposed prediction strategies: multiple single-output, multi-output, stacking, and the benchmark (single-output strategy). Notice that all the strategies were conducted with nested 10-fold CV using different SNP subsets each time. Thus, for each SNP subset, there were 10 prediction performances (one per each fold).
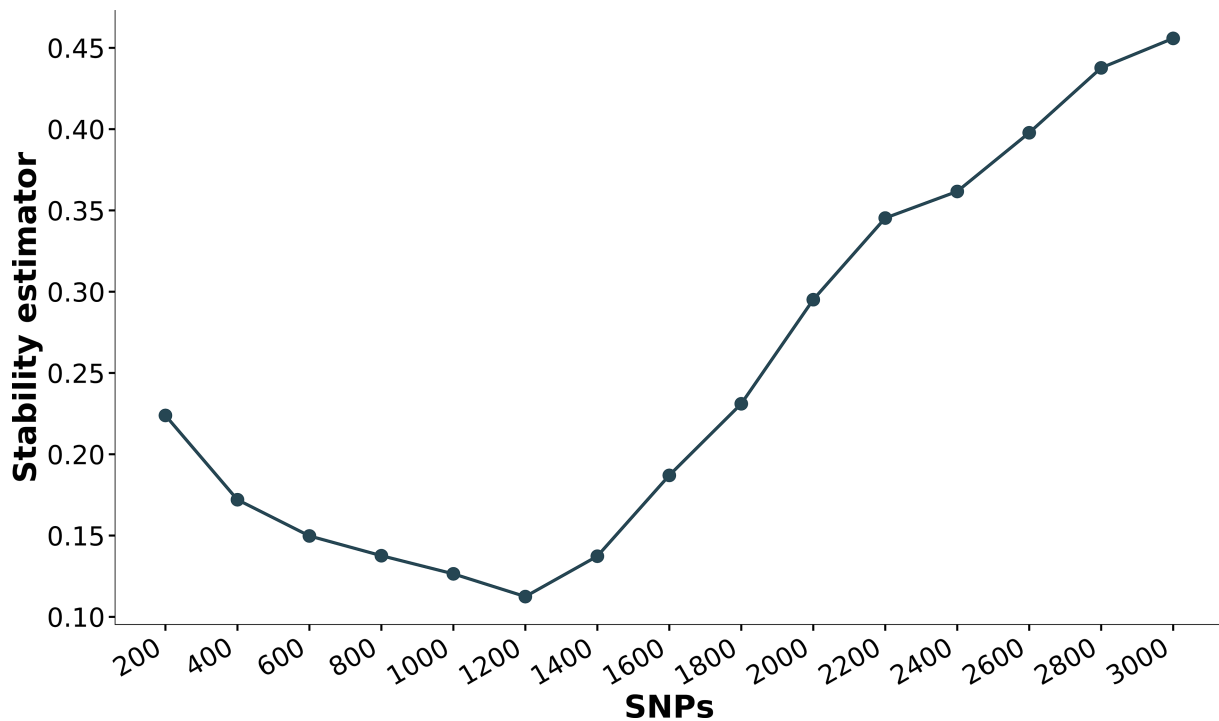
FIGURE 4 Stability of feature selection for each single-nucleotide polymorphism (SNP) subset.

3.3 | Prediction performance of the traits involved in the definition of RFI

The prediction performance of the traits involved in the definition of RFI (i.e., DFI, ADG, BFT, and MW) was evaluated. All these traits were predicted individually (single-output method) and simultaneously (multi-output method). Random forest was implemented for both methods while SVR was only applied to the single-output method. Figures 5 and 6 show the boxplot of the ranking metrics (i.e., Spearman correlation, zero–one loss and rank distance loss) obtained from observed and predicted values of DFI, ADG, BFT, and MW with RF for single and multi-output methods, respectively. Results for SVR can be seen in Figure S1. Unlike expected, the performance of the individual predictions of ADG, DFI, BFT, and MW was not improved when simultaneous predictions of these variables were carried out. For the subset of 1000 SNPs, the highest prediction performance was obtained by implementing the single-output method for BFT. In this case, the mean (SD) was 0.40 (0.02) for Spearman correlation, 0.71 (0.06) for zero–one loss, and 0.24 (0.04) for rank distance loss. For the same trait, the values achieved with the multi-output method were 0.28 (0.04), 0.79 (0.06), and 0.31 (0.05) for Spearman correlation, zero–one loss, and rank distance, respectively. With the multi-output method, the highest predictive performance was obtained for DFI (mean Spearman correlation [SD] = 0.31 [0.04], mean zero–one loss [SD] = 0.78 [0.04], and mean rank

distance loss [SD] = 0.28 [0.03]). For both methods, ADG showed the poorest prediction.

3.4 | Prediction performance of RFI

The ranking metrics obtained when RF for regression was used as a learner to predict RFI following multiple single-output, multi-output, stacking, and single-output strategies are shown in Figure 7. In all cases, the single-output strategy had the best prediction performance. Therefore, the individual and simultaneous prediction of the variables involved in the definition of RFI prior to the calculation of this parameter (indirect strategies) does not present any advantage over the classical prediction method (single-output strategy), nor does the inclusion of the individual prediction of the RFI components as predictor variables of RFI together with the genotype (stacking strategy). The results obtained for regression metrics were in line with those obtained for the ranking metrics and they are shown in Figure S2. For the subset of 1000 SNPs and the benchmark, the mean (SD) of all the metrics was: 0.23 (0.04) for Spearman correlation, 0.83 (0.04) for zero–one loss, and 0.33 (0.03) for rank distance loss. Focusing on the indirect strategies, the multi-output strategy did not offer any advantage over the multiple single-output strategy as expected based on results on RFI components. For the subset of 1000 SNPs, the mean (SD) of the ranking metrics used to assess the prediction performance of RFI
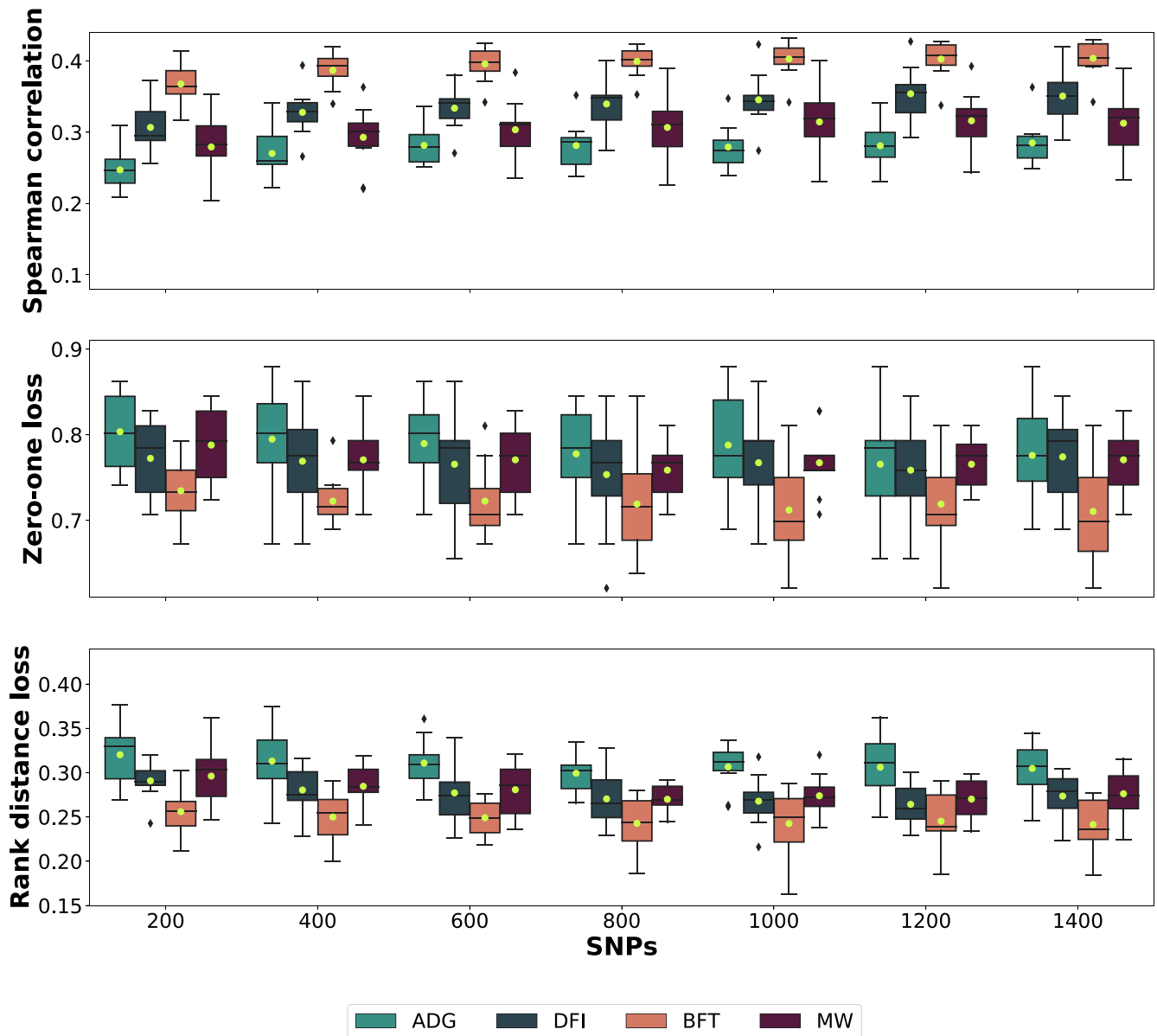
FIGURE 5 Boxplot of the three ranking metrics (Spearman correlation, zero–one loss, and rank distance loss) between observed and predicted values of daily feed intake (DFI), average daily gain (ADG), backfat thickness (BFT), and metabolic weight (MW) with the single-output method using random forest and different subsets sizes of single-nucleotide polymorphisms (SNPs) as predictor variables. The green circle represents the mean.

with the multi-output strategy and the multiple single-output strategy were respectively: 0.20 (0.04) and 0.19 (0.04) for Spearman correlation, 0.86 (0.05) and 0.84 (0.04) for zero–one loss, and 0.37 (0.03) and 0.36 (0.03) for rank distance loss.

The prediction performance obtained when SVR was used as a learner is shown in Figure 8 for the ranking metrics. The results for the regression metrics are in Figure S3 and are consistent with the results obtained from ranking metrics. For this learner, the means of the different metrics obtained with the single-output and the stacking strategies were very close. For the subset of 1000 SNPs, the mean (SD) of the ranking metrics used to assess the

prediction performance of the stacking method and the benchmark were, respectively: 0.20 (0.06) and 0.19 (0.05) for Spearman correlation, 0.85 (0.05) and 0.85 (0.04) for zero–one loss, and 0.34 (0.04) and 0.35 (0.05) for rank distance loss. In contrast, the prediction of RFI though the indirect strategy demonstrated the poorest performance. For the subset of 1000 SNPs, the mean (SD) of the ranking metrics were: 0.13 (0.04) for the Spearman correlation, 0.84 (0.03) for zero–one loss and 0.38 (0.03) for rank distance loss.

Regarding the learner used (SVR and RF), differences in prediction performance were very small. Thus, for example, the mean (SD) of the stacking strategy was better
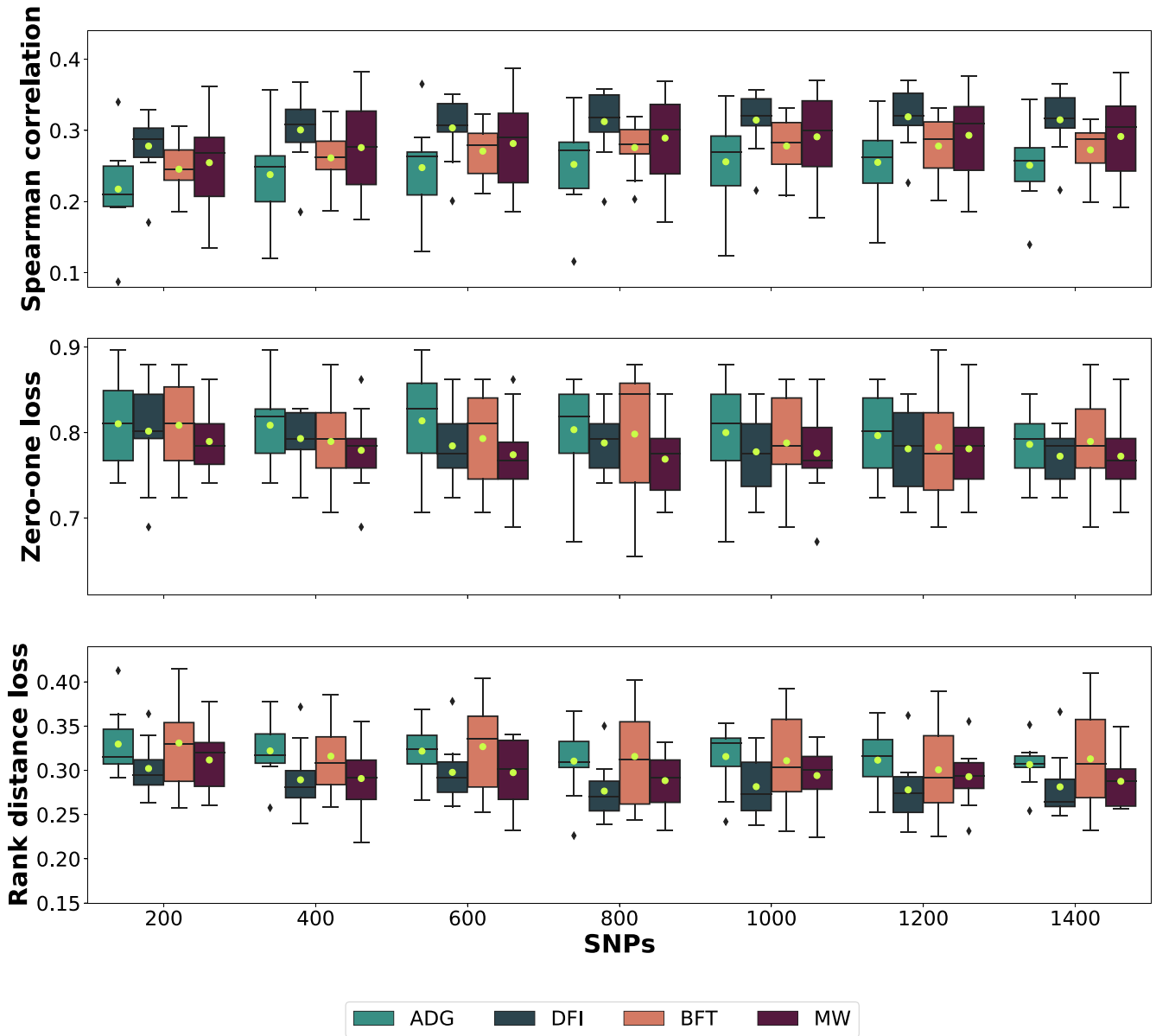
FIGURE 6 Boxplot of the three ranking metrics (Spearman correlation, zero–one loss, and rank distance loss) between observed and predicted values of daily feed intake (DFI), average daily gain (ADG), backfat thickness (BFT), and metabolic weight (MW) with the multi-output method using random forest and different subsets sizes of single-nucleotide polymorphisms (SNPs) as predictor variables. The green circle represents the mean.

with SVR (mean Spearman correlation (SD) = 0.20 (0.06)) than with the RF (mean Spearman correlation [SD] = 0.17 [0.04]) when a subset of 1000 SNPs was used as predictor variables. In contrast, the prediction performance of the single-output strategy was better with RF (mean Spearman correlation [SD] = 0.23 [0.04]) than with SVR (mean Spearman correlation [SD] = 0.19 [0.05]) likewise the prediction performance of the multiple single-output strategy (mean Spearman correlation [SD] = 0.19 [0.04] and mean Spearman correlation [SD] = 0.13 [0.04], with RF and SVR respectively).

4 | DISCUSSION

Several applications for multi-output regression have been reported over the years because of the multiple advantages it offers (Blockeel et al., 2000; Burnham et al., 1999; Struyf & Džeroski, 2006). However, to the best of our knowledge, no study has explored the benefits of multi-output regression methods to predict RFI from the genotype until now. In the literature, some studies on the prediction of RFI using ML algorithms have been presented (Piles et al., 2021; Tusell et al., 2020; Yao et al., 2016). Tusell
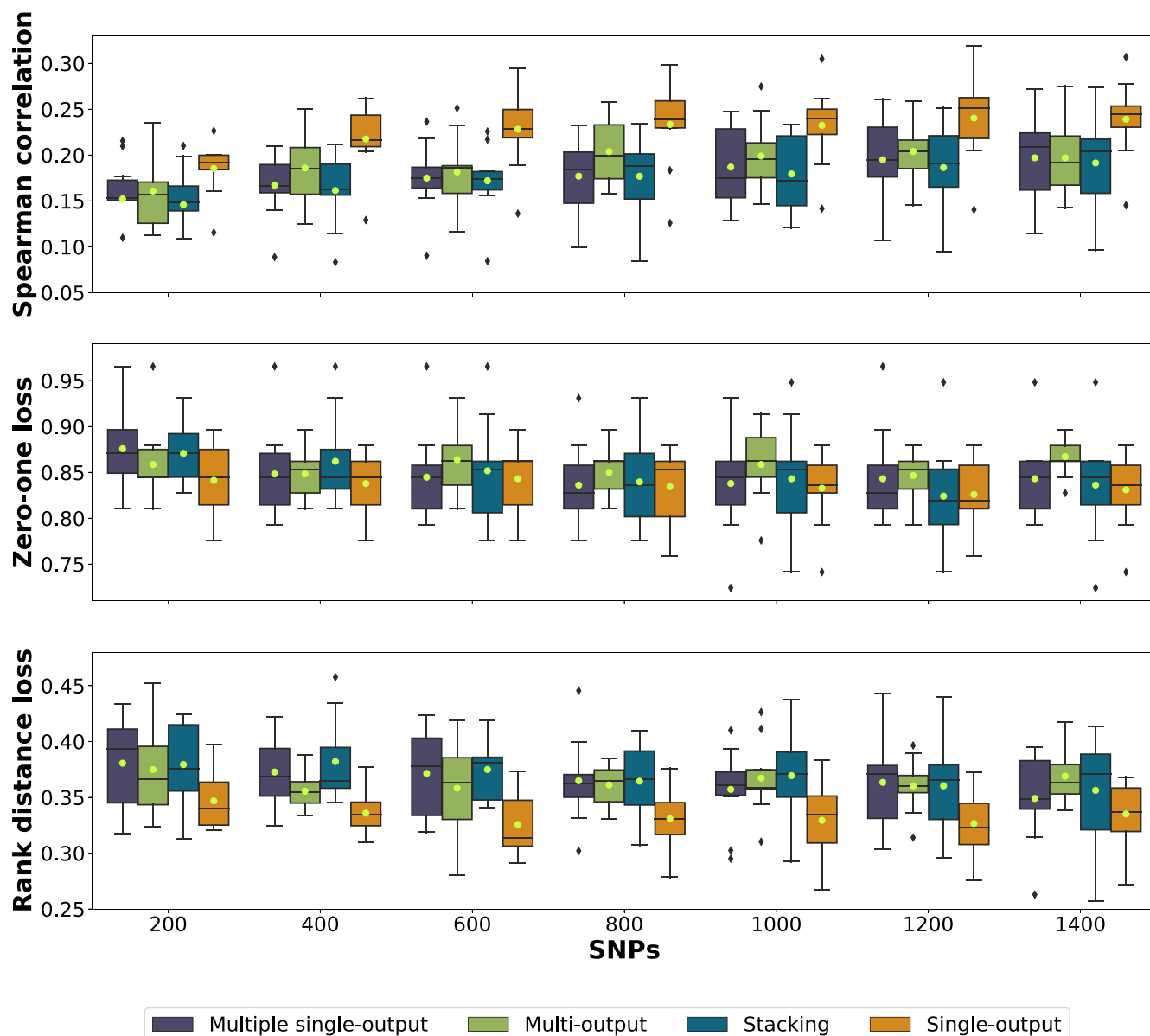
FIGURE 7 Boxplot of prediction performance assessed with ranking metrics using random forest for the indirect (multiple single-output and multi-output strategies) and direct strategies (stacking and single-output strategies) with different subsets sizes of single-nucleotide polymorphisms (SNPs) as predictor variables. The green circle represents the mean.

et al. (2020) and Piles et al. (2021) predicted this trait in a single-output model from the genotype using different sources of information on a population of pigs. In these studies, the highest prediction performance of RFI, in terms of Spearman correlation, was 0.34 with SVR and 50 SNPs (Tusell et al., 2020). To improve this prediction performance, the benefits of multi-output and stacking methods were explored in the present research using the same population of pigs.

Multi-output models ensure better predictive performance when the targets are correlated (Breiman & Friedman, 1997; Similä & Tikka, 2007). This is a consequence of the fact that they can contribute knowledge about each other whereas, single-output models cannot

exploit this information naturally. This was demonstrated by Jia and Jannink (2012) with simulated data. These authors showed that when two traits were genetically uncorrelated and only one of them was highly heritable, the multiple-output model was inferior to the single-output model, while it was superior when the traits were genetically correlated and only one of them was highly heritable. In the case of the traits involved in the definition of RFI (i.e., DFI, ADG, BFT, and MW) it is well-known that they are moderately heritable and are correlated between them (Mora et al., 2022). Therefore, it was expected that the information on the traits involved in the definition of RFI considered jointly could improve the prediction performance of every single trait. Different studies
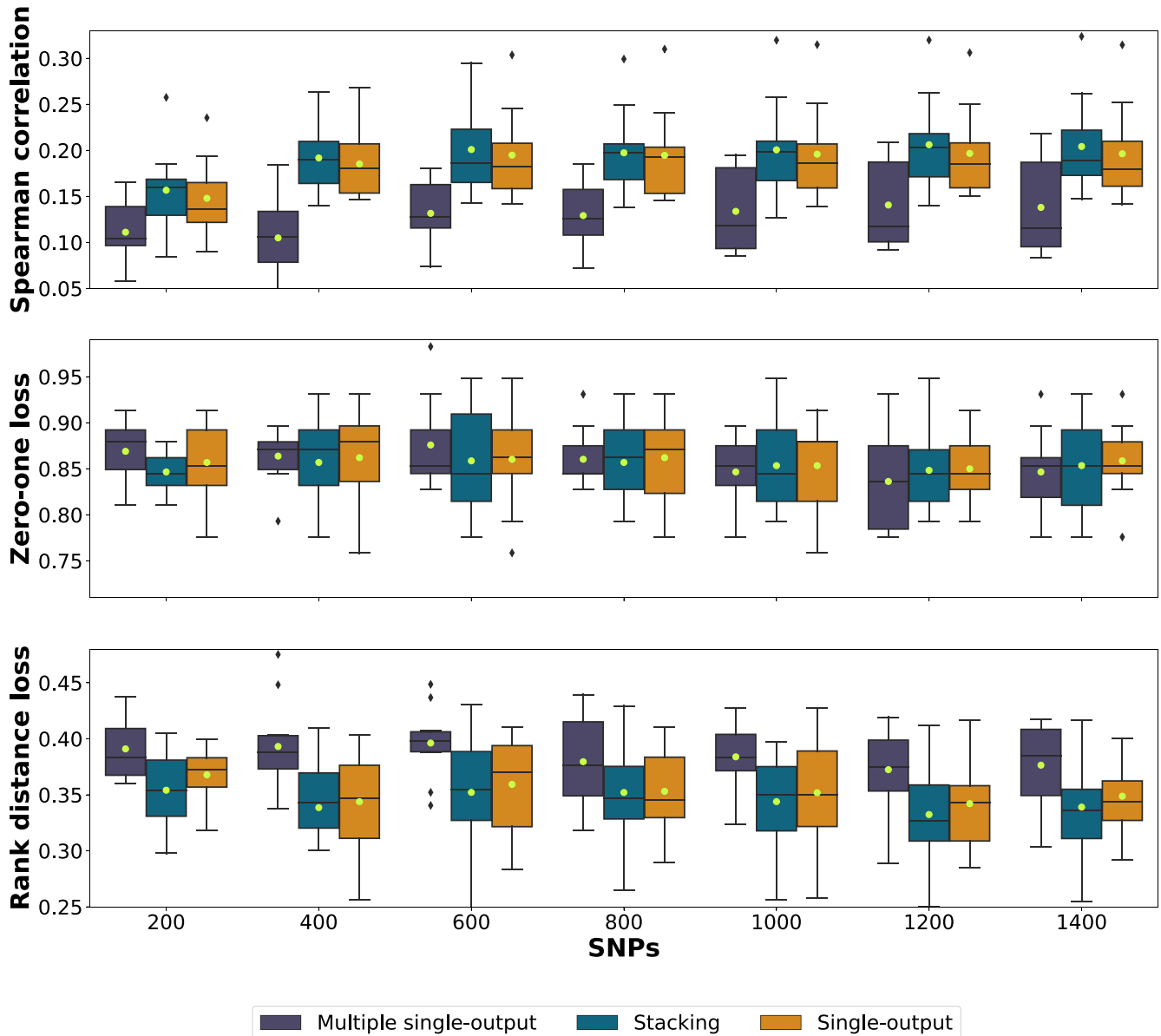
FIGURE 8 Boxplot of prediction performance assessed with ranking metrics using support vector for regression for the indirect (multiple single-output strategy) and direct strategies (stacking and single-output strategies) with different subsets sizes of single-nucleotide polymorphisms (SNPs) as predictor variables. The green circle represents the mean.

in different areas of research have applied multi-output methods to improve individual predictions. For example, in Han et al. (2012), the aim was to predict the multiple gas tank level simultaneously. In this case, the accuracy of the single-output model was rather poor perhaps because the single-output model can hardly reflect the dynamics of the multiple tanks, especially when a certain tank is offline. In another study, the advantages of multi-output method were used to predict different biophysical parameters from remote sensing images simultaneously, leading to an improvement with respect to the single-output regression approach (Tuia et al., 2011). However, not in all cases multi-output methods outperformed single-output methods, but even in those cases, a multi-output method may be advantageous if the training period is shorter. This

is the case of the study of Kocev et al. (2009), in which there is no statistical difference between the multi and single-output methods in terms of model performance when they tried to predict multiple scores of the condition of the vegetation at a given site. In our study, neither multi-output nor stacking approaches offered an advantage, in terms of prediction performance, over the single-output strategy to predict RFI. As expected, the simultaneous prediction of the traits involved in the definition of RFI did not improve their individual predictions. This could be because the individual predictions of DFI, ADG, BFT, and MW from SNP information were not accurate enough (see Figure 5) to contribute relevant information. Thus, the prediction performance of computing RFI using the individuals or simultaneous predictions of its components

(indirect strategies) was very similar when RF was used as a learner. Likewise, the individual predictions along with the genotype as predictors (stacking strategy) did not help to improve the prediction performance of RFI. If there are certain variables that show accurate predictions based on the genotype individually, it is possible that the prediction of another variable, which does not have strong individual predictions, could be enhanced if it is closely correlated with the well-predicted variables. In such cases, the well-predicted variables would provide valuable information while the poorly predicted ones contribute mostly to noise. We believe that this applies to both multi-output and staking analysis. To demonstrate this, we have performed the prediction by implementing the staking strategy using the actual values of the variables DFI, ADG, BFT, and MW. This analysis would correspond to the extreme case of staking as we have implemented it, but in which the values of the RFI components would have been perfectly predicted. In Figure S4 are the results of this demonstration compared with the single-output and stacking strategies. The results show that the quality of the RFI prediction is almost perfect when the RFI components have been perfectly predicted (stacking actual values).

Although the individual predictions of the RFI components may not be highly accurate it does not necessarily imply that they cannot provide valuable information to enhance the accuracy when used in multi-output or stacking strategies. In our case, we had no prior information on what the minimum thresholds of individual prediction quality and correlation between traits would be for these to provide valuable information in a multi-output model to improve individual predictions. Hence, it was necessary to carry out the test. The individual prediction of ADG was previously reported by Tusell et al. (2020) reaching a value of 0.30 for SC with 1000 SNPs using SVR as a learner. In our study, with the same leaner and the same population, this prediction performance was 0.26 with 1000 SNPs, but it was not statistically different from the former. Srivastava et al. (2021) also predicted BFT, among other carcass traits, using SNPs as predictors in a cattle population. The prediction performance of this trait reached its maximum with SVR and it was 0.34 in terms of correlation, lower than the value we obtained with our individual prediction (0.40) but not statistically different either. Therefore, no good prediction performances have been reported for these individual traits using ML algorithms and the genotype as the predictor.

In the stacking method, we take the prediction of the traits and the genotype to predict RFI. The improvement of the stacking method over the single-output has been described in several studies. In the agricultural field, Sapkota et al. (2020) proposed a Bayesian multi-output regressor stacking to improve the genomic selection of grain composition traits, and this method achieved an improvement over the single-output model while the multi-output did

not bring any improvement. Regarding the prediction of RFI, Martin et al. (2021) predicted this trait in Holstein cows with staking methods using different sources of information such as blood metabolite data and sensor-derived behaviour. In this case, in the meta-model stage, two learning algorithms were combined: RF and gradient boosting, while in the present study, we combine the individual predictions of the traits involved in the definition of RFI in the meta-model phase. Similarly to our study, the RFI prediction models provided poor accuracy and the staking method did not confer any advantage, being this prediction more sensitive to the type of data used as predictors. This might be because, RFI is hard to predict due to its definition. The predictive model could capture any variation in RFI corresponding to an additional energy sink not considered in the RFI calculation. By contrast, in our study, we included the same energy sinks in the predictive model that we used to compute RFI, and even so, an improvement of this prediction was not achieved. Another alternative of stacking could have been to use the simultaneous predictions of DFI, ADG, BFT, and MW in the meta-model instead of the individual predictions. This strategy was not carried out considering the results we obtained when comparing simultaneous and individual predictions of individual traits.

An alternative to multi-output regressor methods is to transform the multi-output model into multiple single-output models. In this area, the concept of regressor chains was introduced by Spyromitros-Xioufis et al. (2016). This method consists of selecting a random chain of the set of target variables, then each model makes a prediction in the order specified by the chain using all the features provided to the model plus the predictions of models in the upstream positions of the chain. However, the main problem with this method is that is very sensitive to the selected positions in the chain, increasing the number of possible configurations of the chain as the number of targets to predict increases. Thus, in our study, we decided not to test this method to predict RFI because the order to predict the variables is not clearly known beforehand.

Apart from the most used metrics to evaluate the prediction performance, we proposed two novel loss functions (zero–one loss and rank distance loss) to evaluate the prediction performance of RFI. In the breeding programs of prolific species, the objective is to select a percentage of the best animals. These loss functions are suitable in these cases since they evaluate the quality of the classification based on belonging to the group of the best 10% candidates. To the best of our knowledge, this is the first time that a prediction is evaluated with this type of loss function in the context of a breeding program.

For all the strategies, machine learning algorithms were tested with different subsets of data that contained an

increasing number (from 200 to 3000 by 200) of the most informative SNPs selected with RF. The idea was to find the smallest SNPs subset that is steady and carries a good prediction performance. This would allow the use of low-density SNP panels, a cost-effective practice for breeding programs since many animals could be genotyped. Our results showed that the quality of the prediction increases to reach a maximum with subsets of between 1000 and 1200 SNPs and remains constant beyond this maximum. However, the stability of feature selection for the subset of 1000 SNPs was only 0.13 out of 1. This could indicate that this set of predictors is not steady enough, that is, this set is quite sensitive to changes in the training set. Hence, producing a low-density SNP chip including those SNPs would not be advisable. In this study, RF for regression was carried out for feature selection but other methods could offer better stability. Piles et al. (2021) explore the influence on the stability of the selected subset of SNPs of various combinations of feature selection methods and learners for predicting RFI from the genotype. They demonstrated that different feature selection algorithms performed similarly well for prediction, but they showed wide differences in terms of stability. In our study, the goal was to compare the four strategies under the same conditions, and thus we chose the SNPs subset with the best prediction performance regardless of its model stability. For the same reason, only RF for regression and SVR were used as learners of our predictive models. Many ML algorithms can predict more than one output simultaneously, but RF for regression is a good choice for their predictive power, the computational time, and because it is easy to understand (Kocev et al., 2009). Moreover, RF can be considered a scalable algorithm, hence, it is promising when it deals with dimensional data since it works with subset of data. Compared with SVR, no significant differences were observed between these two learners for the different prediction strategies.

5 | CONCLUSIONS

Considering that improving FE is critical in breeding programs, this study evaluates the potential advantages of multi-output and stacking models to improve the RFI prediction compared with single-output models. Unlike expected, the simultaneous prediction of the traits involved in the definition of RFI did not improve its prediction quality with respect to their individual predictions. In addition, these individual predictions were not accurate enough to provide relevant information to improve the RFI prediction jointly with the genotype. This indicates that complex models do not necessarily outperform simple approaches and that each case should be assessed before making decisions on the method. In our case, for the algorithms and feature selection methods tested, the single-output RF for regression using a subset of 1000 SNPs seems to be the best choice to predict RFI in this population of growing/finishing pigs.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Mónica Mora https://orcid.org/0000-0002-8308-0100
Pablo González https://orcid.org/0000-0002-9250-0920
José Ramón Quevedo https://orcid.org/0000-0001-7211-4312
Elena Montañés https://orcid.org/0000-0003-0609-8945
Llibertat Tusell https://orcid.org/0000-0003-2419-7330
Rob Bergsma https://orcid.org/0000-0002-8254-5535
Miriam Piles https://orcid.org/0000-0001-8265-9930

REFERENCES

Blockeel, H., De Raedt, L., & Ramon, J. (2000). Top-down induction of clustering trees. *Proc. 15th Intl. Conf. On Machine Learning*. https://doi.org/10.48550/arXiv.cs/0011032

Borchani, H., Varando, G., Bielza, C., & Larranaga, P. (2015). *A survey on multi-output regression*. Data Mining and Knowledge Discovery. https://doi.org/10.1002/widm.1157

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

Breiman, L., & Friedman, J. H. (1997). Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society. Series B (Methodological)*, *59*, 3–54. https://doi.org/10.1111/1467-9868.00054

Burnham, A. J., MacGregor, J. F., & Viveros, R. (1999). Latent variable multivariate regression modeling. *Chemometrics and Intelligent Laboratory Systems*, *48*, 167–180. https://doi.org/10.1016/S0169-7439(99)00018-0

Fabian Pedregosa Gaël Varoquaux, A. G. A. V. M., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830. https://doi.org/10.5555/1953048.2078195

Han, Z., Liu, Y., Zhao, J., & Wang, W. (2012). Real time prediction for converter gas tank levels based on multi-output least square support vector regressor. *Control Engineering Practice*, *20*, 1400–1409. https://doi.org/10.1016/j.conengprac.2012.08.006

Jia, Y., & Jannink, J. L. (2012). Multiple-trait genomic selection methods increase genetic value prediction accuracy. *Genetics*, *192*, 1513–1522. https://doi.org/10.1534/genetics.112.144246

Kalousis, A., Prados, J., & Hilario, M. (2005). Stability of feature selection algorithms: A study on high-dimensional spaces. *Knowledge and Information Systems*, *12*, 95–116. https://doi.org/10.1007/s10115-006-0040-8

Khaire, U., & Dhanalakshmi, R. (2019). Stability of feature selection algorithm: A review. *Journal of King Saud University-Computer and Information Sciences*, *34*, 1060–1073. https://doi.org/10.1016/j.jksuci.2019.06.012

Kocev, D., Džeroski, S., White, M. D., Newell, G. R., & Griffioen, P. (2009). Using single- and multi-target regression trees and ensembles to model a compound index of vegetation condition. *Ecological Modelling*, *220*, 1159–1168. https://doi.org/10.1016/j.ecolmodel.2009.01.037

Koch, R. M., Swiger, L. A., Chambers, D., & Gregory, K. E. (1963). Efficiency of feed use in beef cattle. *Journal of Animal Science*, *22*, 486–494. https://doi.org/10.2527/jas1963.222486x

Liang, M., Chang, T., An, B., Duan, X., Du, L., Wang, X., & Gao, H. (2021). A stacking ensemble learning framework for genomic prediction. *Frontiers in Genetics*, *12*, 600040. https://doi.org/10.3389/fgene.2021.600040

Long, N., Gianola, D., Rosa, G. J. M., & Weigel, K. A. (2011). Application of support vector regression to genome-assisted prediction of quantitative traits. *Theoretical and Applied Genetics*, *123*, 1065–1074. https://doi.org/10.1007/s00122-011-1648-y

Martin, M. J., Dórea, J. R. R., Borchers, M. R., Wallace, R. L., Bertics, S. J., DeNise, S. K., & White, H. M. (2021). Comparison of methods to predict feed intake and residual feed intake using behavioral and metabolite data in addition to classical performance variables. *Journal of Dairy Science*, *104*, 8765–8782. https://doi.org/10.3168/jds.2020-20051

Mora, M., David, I., Gilbert, H., Rosa, G. J. M., Sánchez, J. P., & Piles, M. (2022). Analysis of the causal structure of traits involved in sow lactation feed efficiency. *Genetics, Selection, Evolution*, *54*, 53. https://doi.org/10.1186/s12711-022-00744-4

Nogueira, S., & Brown, G. (2016). Measuring the stability of feature selection. In *Machine learning and knowledge discovery in databases. ECML PKDD 2016 Lecture Notes in Computer Science* (Vol. *9852*). Springer. https://doi.org/10.1007/978-3-319-46227-1_28

Nogueira, S., Sechidis, K., & Brown, G. (2018). On the stability of feature selection algorithms. *Journal of Machine Learning Research*, *18*, 1–54. https://doi.org/10.5555/3122009.3242031

Piles, M., Bergsma, R., Gianola, D., Gilbert, H., & Tusell, L. (2021). Feature selection stability and accuracy of prediction models for genomic prediction of residual feed intake in pigs using machine learning. *Frontiers in Genetics*, *12*, 611506. https://doi.org/10.3389/fgene.2021.611506

R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Sapkota, S., Boatwright, J., Jordan, K., Boyles, R., & Kresovich, S. (2020). Multi-trait regressor stacking increased genomic prediction accuracy of sorghum grain composition. *Agronomy*, *10*, 1221. https://doi.org/10.3390/agronomy10091221

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*.

Similä, T., & Tikka, J. (2007). Input selection and shrinkage in multiresponse linear regression. *Computational Statistics & Data Analysis*, *52*, 406–422. https://doi.org/10.1016/j.csda.2007.01.025

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, *14*, 199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88

Spyromitros-Xioufis, E., Groves, W., Tsoumakas, G., & Vlahavas, I. (2012). Multi-label classification methods for multi-target regression. https://doi.org/10.1007/s10994-016-5613-5

Spyromitros-Xioufis, E., Tsoumakas, G., Groves, W., & Vlahavas, I. (2016). Multi-target regression via input space expansion: Treating targets as inputs. *Machine Learning*, *104*, 55–98. https://doi.org/10.1007/s10994-016-5546-z

Srivastava, S., Lopez, B. I., Kumar, H., Jang, M., Chai, H. H., Park, W., & Lim, D. (2021). Prediction of Hanwoo cattle phenotypes from genotypes using machine learning methods. *Animals (Basel)*, *11*, 2066. https://doi.org/10.3390/ani11072066

Struyf, J., & Džeroski, S. (2006). Constraint based induction of multi-objective regression trees. In *Lecture notes in computer science* (Vol. *3933*). Springer. https://doi.org/10.1007/11733492_13

Tuia, D., Verrelst, J., Alonso, L., Perez-Cruz, F., & Camps-Valls, G. (2011). Multioutput support vector regression for remote sensing biophysical parameter estimation. *IEEE Geoscience and Remote Sensing Letters*, *8*, 804–808. https://doi.org/10.1109/LGRS.2011.2109934

Tusell, L., Bergsma, R., Gilbert, H., Gianola, D., & Piles, M. (2020). Machine learning prediction of crossbred pig feed efficiency and growth rate from single nucleotide polymorphisms. *Frontiers in Genetics*, *11*, 567818. https://doi.org/10.3389/fgene.2020.567818

Wang, X., Shi, S., Wang, G., Luo, W., Wei, X., Qiu, A., & Ding, X. (2022). Using machine learning to improve the accuracy of genomic prediction of reproduction traits in pigs. *Journal of Animal Science and Biotechnology*, *13*, 60. https://doi.org/10.1186/s40104-022-00708-0

Yao, C., Zhu, X., & Weigel, K. A. (2016). Semi-supervised learning for genomic prediction of novel traits with small reference populations: An application to residual feed intake in dairy cattle. *Genetics, Selection, Evolution*, *48*, 84. https://doi.org/10.1186/s12711-016-0262-5

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Mora, M., González, P., Quevedo, J. R., Montañés, E., Tusell, L., Bergsma, R., & Piles, M. (2023). Impact of multi-output and stacking methods on feed efficiency prediction from genotype using machine learning algorithms. *Journal of Animal Breeding and Genetics*, *00*, 1–15. https://doi.org/10.1111/jbg.12815