AN AWESOME STUDY IN ANIMAL SCIENCE

A PREPRINT

C. P. James Chen
School of Animal Sciences
Virginia Tech
Blacksburg, VA 24061
niche@vt.edu

May 5, 2024

ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Keywords Object detection · Model selection · Model generalization

1 Introduction

1.1 Modeling

Modeling is an essential tool for hypothesis formulation and decision-making. It functions as a structured investigatory framework that allows researchers to explore system understanding through the summary and analysis of empirical data. Carefully constructed and evaluated models offer the potential to extend this understanding by enabling the extrapolation of results to novel trials and conditions. Although only one focus of the science of modeling, the predictive role is often explicitly or implicitly the ultimate goal of models derived within the precision agriculture context. Through this lens, modeling provides opportunity to standardize and formalize research advancement, through developing

quantitative constructs that accumulate prior knowledge derived by the broader the scientific community. Evaluating model performance becomes particularly critical when considering this role within the knowledge generation enterprise, necessitating a rigorous and standardized approach that allows for both reproducibility and comparability. As more and more model-based exercises are developed using slightly different methods, or slightly different datasets, it becomes increasingly challenging to evaluate, characterize, compare, and balance information generated by the resulting modeling tools, particularly when results are conflicting. Specifically, reporting model performance through poorly-defined metrics or incomplete procedures can create opportunity for confusion, misinterpretation, and miscommunication, while can ultimately result in distrust in model-based tools and impede scientific progress.

Here, we review two types of challenges that can be encountered during the model evaluation process: challenges in data structure and challenges in evaluation approach. Data structure challenges include those inherent to the types of data used in a modeling exercise. For continuous data types, challenges include measurement variance, extreme observations, and underlying variation structures like blocks. For categorical data, challenges largely center around the balance or lack thereof between categories. Challenges in the evaluation approach are driven by decision-making about which data are used for model derivation and which are used for evaluation. We will review these challenges in this study.

1.2 Model Evaluation

Model evaluation in the context of predictive analytics seeks to explore how well a model can generalize to new prediction contexts not seen during model training. Although commonly referred to as "model validation" in the literature, this term implies a false degree of confidence given that the word “validation” means to prove something true. There is no single test, or recognized suite of tests, to prove a model valid. Instead, the term "evaluation," which involves assessing the value, nature, character, or quality of something, is more fitting. It is essential to evaluate a model performance on unseen data to ensure the approach is applicable to new experiments. To this end, cross-validation (CV) is widely recognized as a standard method for model evaluation.

The most common CV method is K-fold CV, which partitions the dataset into K equally sized folds. In each iteration, one fold is reserved as the test set (i.e., new data, noted as $\mathcal{D}_{\text{test}}$), while the remaining folds are used as the training set (noted as $\mathcal{D}_{\text{train}}$) to construct the model. Once the model completes training, it is evaluated on the $\mathcal{D}_{\text{test}}$ to obtain a estimate of the model performance $\hat{g}$. The process will iterate K times until each fold has been used as the $\mathcal{D}_{\text{test}}$ once. And the average performance over all K folds is deemed as the expected generalization performance of the model $\mathbb{E}[\hat{g}]$ on new data.

However, there is always a evaluation bias between the estimated performance $\mathbb{E}[\hat{g}]$ and the true generalization performance $\mathcal{G}$, which can only be approximated by evaluating the same model on an infinite number of unseen data. If RMSE is used as the performance metric, which is lower the better for the model performance, a positive evaluation bias $\mathbb{E}[\hat{g}] - \mathcal{G}$ suggests that the model evaluation procedure concludes a pessimistic estimation of the model performance,

since the true performance is expected to be lower than the estimated performance. Another aspect of model evaluation error is the variance of each estimated performance $\hat{g}$ across the K folds. For example, there are five estimates in a 5-fold cross-validation. The variance among these five estimates is defined as the evaluation variance. A high evaluation variance suggests that the performance is sensitive to the choice of the test set $\mathcal{D}_{\text{test}}$, which may be caused by a small sample size or an over-complex model.

There is a trade-off relationship between the evaluation bias and variance from a squared evaluation bias. When performing K-fold CV with a fixed sample size and model complexity, the choice of K is the pivotal element shaping the model evaluation. When the K is set to a larger value; each training set $\mathcal{D}_{\text{train}}$ is larger in size, resulting in a model trained on a more representative subset of the population of interest, leading to lower bias. However, a large K comes with a trade-off: the corresponding test subset $\mathcal{D}_{\text{test}}$ is compressed in size, making the tested model more sensitive to the specific data points, and thus inflating the validation variance. Conversely, a smaller K, along with a minor training set $\mathcal{D}_{\text{train}}$, reduces their representativeness and increases bias. Nevertheless, a larger size of the test set $\mathcal{D}_{\text{test}}$ leads to more consistent estimations across the folds and, consequently, reduces the validation variance.

Leave-one-out cross-validation (LOOCV) is a variant of K-fold CV where K equals the sample size of the complete dataset $\mathcal{D}$. It provides an unbiased estimation of model performance because the training set $\mathcal{D}_{\text{train}}$ closely resembles the unseen population of interest, given its size of $N - 1$, where N is the sample size. However, as the trade-off discussion suggested, this method can lead to high validation variance due to the model is evaluated on one sample at a time. The true unbiased nature of LOOCV is fully realized only when all K folds are utilized. Performing an incomplete LOOCV can introduce significant bias because of the inherent high validation variance, which often occurs when training each model iteration is prohibitively time-consuming or computationally demanding. In specific contexts, such as genomic prediction, strategies like the one described by Cheng et al. leverage the matrix inverse lemma, which allows for computational savings by avoiding the inversion of large matrices in each fold. This technique significantly reduces the dependency of computational resources on the sample size (Cheng et al., 2017). Van Dixhoorn et al. exemplify the use of LOOCV with a small dataset, aiming to predict cow resilience with limited data resources (van Dixhoorn et al., 2018). Nevertheless, for large datasets, LOOCV is generally not recommended due to computational inefficiency. Further details of bias-variance trade-off have been extensively explored in the statistical literature (Hastie et al., 2009; Cawley and Talbot, 2010).

1.3 Model Selection

Model selection becomes necessary when models are not entirely determined by the data alone. For example, in a regularized linear regression model such as a ridge regression (Hoerl and Kennard, 1970) or the least absolute shrinkage and selection operator (LASSO) (Tibshirani, 1996), it is essential to define a regularization parameter, λ, before fitting the model to the data. A larger λ value yields a more regularized model, which tends to reduce smaller coefficients to negligible values or zero. This approach helps in preventing overfitting noise in the training data. The definition of loss functions for the regularized models were described in 22 and 23 of the Appendix.

These pre-defined parameters, which influence model fitting and remain constant during the training process, are known as hyperparameters. Beyond regularized models, hyperparameters are crucial in other predictive models, enhancing flexibility and robustness. For example, in the Support Vector Regression (SVR) (Drucker et al., 1996), the regressors X are projected onto a linear subspace to approximate the target variable Y. By choosing a suitable kernel function, which transforms the regressors into a non-linear space, as a hyperparameter, SVR can more effectively capture non-linear relationships, thus significantly improving model performance. Another hyperparameter example is the number of latent variables in the Partial Least Square (PLS) Regression (Abdi, 2003), which condenses the original regressors into a more manageable set of latent variables, reducing multicollinearity issues. Fewer latent variables might lose significant information from the original regressors, while too many can lead to overfitting. Similarly, in Random Forest (Breiman, 2001), hyperparameters such as tree depth and the number of trees dictate model complexity. The same applies to the number of hidden layers and the size of filters in convolutional neural networks (LeCun, 1989). All these examples highlight the fact that selecting the most suitable hyperparameters, which is known as hyperparameter tuning, is crucial for optimizing model performance. Feature selection is another crucial aspect of model selection. This process involves fitting the model to a selected subset of the original features, particularly essential in high-dimensional data scenarios where the number of features exceeds the number of observations, leading to poor model generalization. For instance, Ghaffari et al., 2019 sought to predict health traits in 38 multiparous Holstein cows using a metabolite profiling strategy. Out of 170 metabolites, only 12 were identified as effective discriminators between healthy and over-conditioned cows and were thus selected for the predictive model. Therefore, optimizing feature subsets is a vital model selection strategy that significantly affects model performance. Including the model selection process within the cross-validation is essential to avoid common pitfalls. The risk of inflated model performance arises when model selection is guided by results on the test dataset. Even if the chosen model is subjected to k-fold cross-validation afterward, its selection bias toward the test set can lead to overestimating its efficacy. This issue has been highlighted in statistical literature (such as Hastie et al., 2009). A practical solution is to divide the dataset into training, validation, and test sets. The validation set is then used for model selection, ensuring the test set remains completely unused during the training phase, thereby providing a more accurate measure of model performance. For instance, the study by Rovere et al. exemplifies best practices in hyperparameter tuning and feature selection by employing an independent cross-validation step prior to assessing model performance. This approach enabled the precise selection of relevant spectral bands from the mid-infrared spectrum and the optimal number of latent dimensions in PLS with Bayesian regression for predicting the fatty acid profile in milk (Rovere et al., 2021). Similarly, Becker et al. demonstrated a robust evaluation by using nested cross-validation loops; the inner loop conducted a grid search for the best hyperparameters in logistic regression, while the outer loop was designed to evaluate the performance of the resulting optimized model (Becker et al., 2021). Both examples underscore the importance of separating model selection from performance evaluation to ensure the validity and reliability of the results.

1.4 Cross Validation Design with Block Effects

Blocking is an essential approach in experimental design to control for variations that can confound the variable of interest. For instance, Lahart et al., (2019) investigated the dry matter intake of grazing cows using mid-infrared (MIR) spectroscopy technology across multiple herds under varying experimental conditions. Given the significant variation between herds, which may contribute to individual differences in both dry matter intake and MIR spectra, it is crucial to consider the herd as a blocking factor before evaluating the predictability of dry matter intake using MIR spectra. This consideration should also extend to model validation. In the cited study, variations in dry matter intake, the primary focus of the prediction model, were observed to exceed one standard deviation among some herds. In cross-validation, if samples from the same herd are assigned to different folds, with one fold used as the test set, the model is likely to achieve high accuracy. This accuracy may largely result from explaining the inter-herd variation rather than individual variations in dry matter intake, leading to an overestimation of model performance. To avoid this pitfall, block cross-validation, where each block (i.e., herd in this example) is used as a fold, is recommended for unbiased model validation. Literature reviews have indicated that block cross-validation effectively evaluates model performance on external or unseen datasets (Bresolin and Dórea, 2020). In the same study by Lahart et al., three cross-validation strategies were compared: random cross-validation (Random CV), which randomly assigns samples to folds; within-herd validation, training and testing the model within each herd; and across-herd validation (Block CV), where each herd is used as a fold and tested in turn. The results showed that performance estimates in block CV were noticeably lower than the other two strategies, supporting the hypothesis that ignoring block effects inflates model performance. Other studies considering block effects, including diet (Grelet et al., 2020), herd (Rovere et al., 2021), and farm location (Adriaens et al., 2020; Mota et al., 2022), have shown similar results in cross-validation, demonstrating block CV's effectiveness in evaluating model performance on external datasets.

1.5 Model Performance Metrics

Model performance metrics serve as quantitative indicators for evaluating model performance. They are critical for benchmarking various modeling approaches and for evaluating hypotheses underpinning these different approaches. Choosing appropriate metrics to support hypothesis testing is crucial, as in-ideal selection may lead to overly optimistic conclusions. Due to the different goals of regression and classification tasks, it is critical to ensure that these different model types are evaluated using different metrics. As such, metrics for regression and classification are discussed individually.

1.5.1 Metrics in Regression Tasks

Regression models aim to predict continuous variables and are commonly employed in diverse applications, such as estimating body condition scores (Spoliansky et al., 2016; Yukun et al., 2019), body weight (Song et al., 2018; Xavier et al., 2022), milk composition (Mäntysaari et al., 2019; Frizzarin et al., 2021; Rovere et al., 2021; Mota et al., 2022), efficiency of feed resource usage (Appuhamy et al., 2016; de Souza et al., 2018; Grelet et al., 2020), and

early-lactation behavior (van Dixhoorn et al., 2018). The metrics in regression tasks evaluate the agreement between the predicted value $\hat{y}$ and the true values y. The agreement can be generally quantified in two ways: error-based metrics and linearity-based metrics. Error-based metrics focus on the deviation of each pair of predicted and true values, while linearity-based metrics consider overall linear relationships between the predictions and the truths. The root mean square error (RMSE) and the mean absolute error (MAE) are two common error-based metrics:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{1}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2}$$

where y_i and $\hat{y}_i$ are the true and predicted values, respectively, and n is the sample size. Both metrics preserve the scale of the original data, making them easy to interpret in real-world units. Additionally, compared to MAE, RMSE penalizes large errors more due to the squared term, making it more sensitive to outliers. In the cow production, monitoring animal body weight is a common practice to aid in the management of dairy cows. Studies by Song et al. and Xavier et al. have utilized RMSE to assess the effectiveness of three-dimensional cameras in estimating dairy cow body weight, yielding RMSE values of 41.2 kg and 12.1 kg, respectively (Song et al., 2018; Xavier et al., 2022). These figures provide a straightforward value for farmers to gauge whether the prediction error is tolerable, considering their specific operational costs and management thresholds. In essence, RMSE translates complex model accuracy into practical insights for productive agricultural units. When evaluating the same model across different traits, which may have different scales, a common practice is to express error metrics in a scale-free manner. This can be achieved by expressing RMSE as a percent of the mean observed value, such as root mean squared percentage error (RMSPE), or as a Root Mean Standard Deviation Ratio (RSR) that normalizes the RMSE by the standard deviation of the observed values:

$$\text{RMSPE} = \frac{\text{RMSE}}{\bar{y}} \tag{3}$$

$$\text{RSR} = \frac{\text{RMSE}}{\sigma_y} \tag{4}$$

where $\bar{y}$ and σ_y are the mean and standard deviation of the observed values, respectively. When expressed as a percent, RMSPE typically ranges from 0 and above, with values closer to 0 indicating perfect prediction. Much like expressing RMSE as a percent, RSR is valuable to interpret RMSE in terms of the context of the variance in the observations. Values below 1 suggest that the model yields predictions less variable than the standard deviation, while values above 1 suggest that the prediction is imprecise.

On the other hand, Pearson's correlation coefficients (r) and the coefficient of determination (R^2) are two common linearity-based metrics:

$$\begin{aligned} r &= \frac{\text{cov}(y, \hat{y})}{\sigma_y \sigma_{\hat{y}}} \\ &= \frac{\sum_{i=1}^{n}(y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2 \sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})^2}} \end{aligned} \tag{5}$$

$$\begin{aligned} R^2 &= 1 - \frac{SS_{\text{residual}}}{SS_{\text{total}}} \\ &= 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \end{aligned} \tag{6}$$

where SS_{residual} is the residual sum of squares and SS_{total} is the total sum of squares. Each y_i and $\hat{y}_i$ are the ith elements of the actual response vector y and the predicted response vector $\hat{y}$, respectively. $\bar{y}$ and $\bar{\hat{y}}$ are their respective means. Both r^2 and R^2 are scale invariant, meaning their values are unaffected by the scale of the observed data because they are normalized by the variation in the denominator.

The correlation coefficient r measures the strength of the linear relationship between two continuous variables, y and $\hat{y}$, and ranges from -1 to 1. A value of 0 indicates no prediction accuracy in the evaluated model. One special characteristic of correlation r is that it is unaffected by the scale of the predictions or biases; it focuses on the relative changes in the predicted values compared to the true values. Thus, even if the prediction biases are scaled up or down, the correlation r between $\hat{y}$ and y remains the same. This property is particularly useful when the focus is more on ranking predictions rather than their absolute values. For example, this metric has been used to evaluate models that identify high-performing production individuals, demonstrating the ability to predict nutrient digestibility in dairy cows (de Souza et al., 2018) and to select models based on their ability to rank traits such as feed intake and milk composition in dairy cows (Dórea et al., 2018; Rovere et al., 2021).

The coefficient of determination R^2 quantifies model performance from the proportion of variance in the dependent variable that is predictable from the independent variables. It ranges from negative infinity to 1, where 1 indicates that the model explains all the variance in the dependent variable, and 0 indicates that the model performs no better than predicting all samples as the mean of the observed values. R^2 is useful in comparing multiple regression models, as demonstrated in studies that regress body weight of dairy cows on a set of morphological traits (Xavier et al., 2022), examine the relationship between milk spectral profiles and nitrogen utilization efficiency (Grelet et al., 2020), and evaluate the predictive performance of milk fatty acid composition (Mäntysaari et al., 2019).

It worth noting that many literatures have misinterpreted the relationship between r and R^2. The coefficient of determination R^2 is not always equivalent to the square of the correlation coefficient r^2. The equivalence only holds when the same dataset is used for both model fitting and evaluation in a least squares regression model. The model assumes a zero covariance between the fitted residual and the predicted values $\hat{y}$, and it also assumes that the residuals (i.e., prediction biases) are centered on zero. In practice when predictions are made on new data, those assumptions

are often violated, leading to discrepancies between r^2 and R^2. A details derivation of the equivalence is provided in Equation 24 25 in the Appendix.

In addition to r^2 and R^2, another linearity-based metric is Lin's concordance correlation coefficient (CCC) (Lin, 1989):

$$\begin{aligned} \rho_c &= \frac{2\rho\sigma_y\sigma_{\hat{y}}}{\sigma_y^2 + \sigma_{\hat{y}}^2 + (\bar{y} - \bar{\hat{y}})^2} \\ &= \frac{2\text{cov}(y, \hat{y})}{\sigma_y^2 + \sigma_{\hat{y}}^2 + (\bar{y} - \bar{\hat{y}})^2} \end{aligned} \tag{7}$$

where ρ is the Pearson correlation coefficient r. The CCC is a comprehensive metric because it considers both the correlation and the scale bias between the predicted and true values. It fills the gap left by r^2 where the scale bias is ignored. Geometrically, CCC measures how well the predicted values $\hat{y}$ fall on the 45-degree line in a scatter plot of the predicted (x-axis) and true values (y-axis). It is advantageous over R^2 because it consistently ranges from -1 to 1, making it easier to interpret and compare across different studies. The CCC is crucial when precise predictions are required for both the scale and the rank of the trait of interest, such as in studies predicting cotton crop yields based on soil and terrain profiles (Jones et al., 2022).

1.5.2 Metrics in Classification Tasks

Classification models aim to predict categorical outcomes such as 'healthy' or 'sick,' 'susceptible' or 'resistant,' and 'high yield' or 'low yield.' To evaluate classification performance, one must first establish a confidence threshold to dichotomize the prediction probabilities. For instance, if a prediction probability exceeds the threshold, the sample is predicted as a positive sample. It is worth mentioning that this threshold is adjustable to fine-tune model performance for particular uses.

Accuracy is the most straightforward metric for evaluating classification models:

$$\begin{aligned} \text{Accuracy} &= \frac{\text{Total Correct Predictions}}{\text{Total Predictions}} \\ &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \end{aligned} \tag{8}$$

where TP, TN, FP, and FN represent the number of true positives, true negatives, false positives, and false negatives, respectively. It summarizes an overall model performance by calculating the proportion of correctly classified samples among all samples. Nonetheless, accuracy can be misleading when the classes are imbalanced. For example, if a study predicting the presence of a specific event, of which the prevalence was only 10%. In this case, a model that predicts all samples as negative would achieve an accuracy of 90%, which is misleadingly high. To address this issue, precision and recall are introduced:

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{Total Predicted Positives}} \\ &= \frac{\text{TP}}{\text{TP} + \text{FP}} \end{aligned} \tag{9}$$

$$\begin{aligned} \text{Recall} &= \frac{\text{TP}}{\text{Total Actual Positives}} \\ &= \frac{\text{TP}}{\text{TP} + \text{FN}} \end{aligned} \tag{10}$$

Precision and recall refine the assessment of a classification model by offering insights that accuracy alone may overlook. Precision calculates the fraction of true positives among all positive predictions, essentially measuring the trustworthiness of positive predictions made by the model 9. High precision is crucial in scenarios where false positives incur significant costs, and false negatives are more tolerable. For instance, in contexts where clinical treatments and culling are expensive, such as detecting bovine tuberculosis (Denholm et al., 2020) or mastitis (Kandeel et al., 2019) using non-invasive methods, a high-precision model is crucial to minimize unnecessary costs and interventions from false positives. On the other hand, recall, also known as sensitivity, quantifies the ratio of true positives to all actual positives, assessing the model's ability to identify positive cases 10. High recall is essential where missing a positive case has serious consequences, or where false positives are easily rectifiable. For instance, detecting lameness or abnormal gait is crucial, as these can indicate underlying pathologies (O'Leary et al., 2020) and impact welfare-related transport decisions (Stojkov et al., 2018). An automated detection system (Alsaaod et al., 2019; Kang et al., 2020; O'Leary et al., 2020) with high recall can mitigate economic losses by flagging at-risk cows. The benefit here lies in the feasibility of re-examining false positives, thus preventing more severe outcomes from undetected cases.

However, it is worth emphasizing that precision and recall focus predominantly on positive samples. Inappropriately assigning a predominant background event as the positive class can lead to skewed interpretations. Hence, the Receiver Operating Characteristic (ROC) curve provides an another crucial tool for assessing a model's performance in a label-agnostic manner, meaning it is not biased by the class distribution as precision and recall are. An ROC curve plots one minus specificity against sensitivity. The equations for specificity and sensitivity are as follows:

$$\begin{aligned} \text{Specificity} &= \frac{\text{TN}}{\text{Total Actual Negatives}} \\ &= \frac{\text{TN}}{\text{FP} + \text{TN}} \end{aligned} \tag{11}$$

$$\begin{aligned} \text{Sensitivity} &= \text{Recall} \\ &= \frac{\text{TP}}{\text{Total Actual Positives}} \\ &= \frac{\text{TP}}{\text{TP} + \text{FN}} \end{aligned} \tag{12}$$

A model's effectiveness, as depicted on the ROC curve, is gauged by how closely a point on the curve approaches the top-left corner. A steep ascent from the left side of the curve signifies the model's ability to correctly identify most true positives while incurring a low rate of false positives. A random guess, with a 50% chance of correct prediction, corresponds to a diagonal line on the ROC curve. In dairy science, the ROC curve has been extensively utilized, for example, in predicting mastitis from milk composition (Jensen et al., 2016) and diagnosing pregnancy using spectroscopy technology (Delhez et al., 2020). In this hypothetical example, the ROC curve also demonstrates robustness and label-invariance with a consistent AUC of 0.875, regardless of whether the original or inverted labels are used.

In addtion to the metrics, the Matthews Correlation Coefficient (MCC) provides a symmetrical measure of the quality of binary classifications. The MCC considers both positive and negative samples in the dataset, providing a balanced measure of a model's performance (Chicco and Jurman, 2020). It is defined as:

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}} \tag{13}$$

The equation 13 symmetrically incorporates all four components of TP, TN, FP, and FN). This symmetry makes MCC invariant to class distribution changes. The coefficient ranges from -1 to 1, where 1 indicates perfect classification, 0 indicates no better performance than random guessing, and -1 signifies total disagreement between prediction and observation. In a case study that used feeding and daily activity behaviors to diagnose Bovine Respiratory Disease in dairy calves, MCC proved particularly insightful (Bowen et al., 2021). The models in this study exhibited strong performance on negative samples (i.e., healthy calves), which were more prevalent, resulting in high specificity. However, sensitivity was relatively low at 0.54. In this context, MCC, with a value of 0.36, provided a more nuanced and representative measure of model performance, especially given the skew towards negative samples.

1.5.3 Study Objectives

This simulation study aims to highlight how biased or over-optimistic estimations of model performance usually come from inappropriately conducting CV, a technique crucial for characterizing expected model performance on "new" data. We demonstrate how common pitfalls, including using the exact data for both training and model assessment, excluding the model selection process from CV, and neglecting experimental block effects, contribute to challenges in model evaluation. Further, we scrutinize common metrics used in evaluating prediction models, including those used for regression and classification tasks. Because no single metric provides a comprehensive perspective of model

performance, we seek, through this work, to highlight the importance of understanding the underlying theory of each metric to avoid misleading conclusions.

There are five simulation studies being conducted to address these challenges. The first simulation study will focus on the bias-variance trade-off in CV, demonstrating how the choice of K in K-fold CV affects the evaluation bias and variance. The second simulation study will investigate the impact of mistakenly using the same data for model selection and evaluation, highlighting the inflated model performance. The third simulation study will explore the effect of excluding block effects in CV, demonstrating how ignoring block effects can lead to over-optimistic model performance. The fourth simulation study will present four hypothetical predictions made in the same regression tasks, leading to different interpretations with different metrics. The fifth simulation study will demonstrate the impact of imbalanced data on classification model evaluation, showing how the choice of metrics can lead to misleading conclusions. Overall, this series of simulation studies aims to guide researchers in accurately and consistently reporting model performance, thereby supporting integrity and scientific rigor in prediction modeling research.

Appendix

Cross Validation

Model cross validation aims to evaluate how well a given model generalizes to an independent dataset that it has not seen during the training process. The most common method is K-fold cross-validation (**K-fold CV**). To implement the K-fold CV, the available dataset, denoted as $\mathcal{D}$, is partitioned into K equally sized folds. We can express the dataset as below:

$$\begin{aligned}\mathcal{D} &= \{(X, Y)\} \\ &= \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_K, Y_K)\}\end{aligned} \tag{14}$$

where $X \in \mathbb{R}^{n \times p}$ represents the input features, and $Y \in \mathbb{R}^{n \times 1}$ symbolizes the ground truth labels for a single target variable. The value of n corresponds to the total number of samples, while p represents the number of features. In each iteration of the K-fold CV, a single fold is reserved as the test set, $\mathcal{D}_{\text{test}}$ (or $\mathcal{D}_k$), to act as unseen data, while the remaining folds make up the training set $\mathcal{D}_{\text{train}}$ (or $\mathcal{D}_{\text{-k}}$):

$$\begin{aligned}\mathcal{D}_{\text{train}} &= \mathcal{D}_{\text{-k}} \\ &= \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_{k-1}, Y_{k-1}), (X_{k+1}, Y_{k+1}), \ldots, (X_K, Y_K)\} \\ \mathcal{D}_{\text{test}} &= \mathcal{D}_k \\ &= \{(X_k, Y_k)\}\end{aligned} \tag{15}$$

After splitting the dataset into $\mathcal{D}_{\text{-k}}$ and $\mathcal{D}_k$, the examined model f is trained on the training set $\mathcal{D}_{\text{-k}}$ and denoted as $f_{\mathcal{D}_{\text{-k}}}$. The hold-out test set $\mathcal{D}_k$ is then used to evaluate the model performance $\hat{g}\left(f_{\mathcal{D}_{\text{-k}}}\right)$, which is defined by comparing the predicted labels $\hat{Y}_k = f_{\mathcal{D}_{\text{-k}}}(X_k)$ with the true labels Y_k using a performance metric $\mathcal{L}$ (e.g., RMSE or R^2):

$$\begin{aligned} \hat{g}(f_{\mathcal{D}_{\text{-k}}}) &= \mathcal{L}(Y_k, \hat{Y}_k) \\ &= \mathcal{L}(Y_k, f_{\mathcal{D}_{\text{-k}}}(X_k)) \end{aligned} \tag{16}$$

To estimate the generalization performance of a model $\mathbb{E}[\hat{g}(f_{\mathcal{D}})]$, the K-fold CV procedure is repeated K times until each fold has been used as the test set $\mathcal{D}_k$ once. The entire dataset $\mathcal{D}$ is leveraged to calculate the average prediction performance over all K folds. The model's generalization performance can be expressed as:

$$\begin{aligned} \mathbb{E}[\hat{g}(f_{\mathcal{D}})] &= \mathbb{E}[\hat{g}(f_{\mathcal{D}_{\text{-k}}})] \\ &= \frac{1}{K}\sum_{k=1}^{K} \hat{g}(f_{\mathcal{D}_{\text{-k}}}) \end{aligned} \tag{17}$$

It is noted that $\mathbb{E}[\hat{g}(f_{\mathcal{D}})]$ is equivalent to $\mathbb{E}[\hat{g}(f_{\mathcal{D}_{\text{-k}}})]$ in K-fold CV. It is because the $\mathbb{E}[\hat{g}(f_{\mathcal{D}})]$ is estimated by averaging all $\hat{g}(f_{\mathcal{D}_{\text{-k}}})$ over K folds, which is also the definition of $\mathbb{E}[\hat{g}(f_{\mathcal{D}_{\text{-k}}})]$.

Cross Validation Bias and Variance

The true generalization performance of the model $G(f_{\mathcal{D}})$ can only be approximated by averaging the performance metrics over infinite unseen datasets. However, in practice, the dataset $\mathcal{D}$ is finite and therefore, there is always a bias when using a finite dataset to estimate $G(f_{\mathcal{D}})$. The bias is known as validation bias:

$$\text{Bias} = \mathbb{E}[\hat{g}\left(f_{\mathcal{D}}\right)] - G(f_D) \tag{18}$$

For example, if RMSE is used as the performance metric, a positive validation bias suggests that the model validation procedure concludes a pessimistic estimation of the model performance, since the true performance is expected to be lower than the estimated performance. Another aspect of model validation is the variance of the estimated performance. For example, in a 5-fold cross-validation, there are five estimates of the model performance. The variance among these five estimates is known as validation variance. A high validation variance suggests that the performance is sensitive to the choice of the test set $\mathcal{D}_k$, which may be caused by a small sample size or an over-complex model. The validation variance can be defined as:

$$
\begin{aligned}
\text{Variance} &= \mathbb{E}[(\hat{g}(f_{\mathcal{D}_{\text{-k}}}) - \mathbb{E}[\hat{g}(f_{\mathcal{D}})])^2] \\
&= \mathbb{E}[\hat{g}^2(f_{\mathcal{D}_{\text{-k}}}) - 2\hat{g}(f_{\mathcal{D}_{\text{-k}}})\mathbb{E}[\hat{g}(f_{\mathcal{D}})] + \mathbb{E}^2[\hat{g}(f_{\mathcal{D}})]] \\
&= \mathbb{E}[\hat{g}^2(f_{\mathcal{D}_{\text{-k}}})] - 2\mathbb{E}[\hat{g}(f_{\mathcal{D}_{\text{-k}}})]\mathbb{E}[\hat{g}(f_{\mathcal{D}})] + \mathbb{E}^2[\hat{g}(f_{\mathcal{D}})] \\
&= \mathbb{E}[\hat{g}^2(f_{\mathcal{D}_{\text{-k}}})] - \mathbb{E}^2[\hat{g}(f_{\mathcal{D}})]
\end{aligned}
\tag{19}
$$

Combining the Equations 18 and 19, the mean squared error (MSE) of the model validation can be decomposed as:

$$
\begin{aligned}
\text{MSE} &= \mathbb{E}[(\hat{g}(f_{\mathcal{D}_{\text{-k}}}) - G(f_{\mathcal{D}}))^2] \\
&= \mathbb{E}[\hat{g}^2(f_{\mathcal{D}_{\text{-k}}})] - 2\mathbb{E}[g(f_{D_{\text{-k}}})]G(f_{\mathcal{D}}) + G^2(f_{\mathcal{D}}) + \\
&\quad \mathbb{E}^2[\hat{g}(f_{\mathcal{D}_{\text{-k}}})] - \mathbb{E}^2[\hat{g}(f_{D_{\text{-k}}})] \\
&= (\mathbb{E}^2[\hat{g}(f_{\mathcal{D}_{\text{-k}}})] - 2\mathbb{E}[\hat{g}(f_{D_{\text{-k}}})]G(f_{\mathcal{D}}) + G^2(f_{\mathcal{D}})) + \\
&\quad (\mathbb{E}[\hat{g}^2(f_{\mathcal{D}_{\text{-k}}})] - \mathbb{E}^2[\hat{g}(f_{\mathcal{D}_{\text{-k}}})]) \\
&= (\mathbb{E}[\hat{g}(f_{\mathcal{D}_{\text{-k}}})] - G(f_{\mathcal{D}}))2 + (\mathbb{E}[g2(f_{\mathcal{D}_{\text{-k}}})] - E^2[\hat{g}(f_{\mathcal{D}_{\text{-k}}})]) \\
&= (\mathbb{E}[\hat{g}(f_{\mathcal{D}})] - G(f_{\mathcal{D}}))^2 + (\mathbb{E}[\hat{g}^2(f_{\mathcal{D}_{\text{-k}}})] - \mathbb{E}^2[\hat{g}(f_{\mathcal{D}})]) \\
&= \text{Bias}^2 + \text{Variance}
\end{aligned}
\tag{20}
$$

Hyperparameter

Here are the loss functions for ordinary least squares (OLS), ridge regression, and LASSO regression, respectively:

$$
\mathcal{L}_{\text{OLS}}(\beta) = \sum_{i=1}^{n}(y_i - x_i\beta)^2 \tag{21}
$$

$$
\mathcal{L}_{\text{ridge}}(\beta) = \sum_{i=1}^{n}(y_i - x_i\beta)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 \tag{22}
$$

$$
\mathcal{L}_{\text{LASSO}}(\beta) = \sum_{i=1}^{n}(y_i - x_i\beta)^2 + \lambda\sum_{j=1}^{p}|\beta_j| \tag{23}
$$

Where x_i and y_i represent the ith row of the design matrix X and the response vector Y, respectively. The term n denotes the sample size, and β is the coefficient vector. All three models aim to find the optimal β that minimizes their respective loss function, $\mathcal{L}$. In the regularized models (i.e., ridge and LASSO regression), the vector length of β is penalized in the loss function.

Sqaured Correlation Coefficient r^2 and Determination Coefficient R^2

The squared Pearson correlation coefficient, r^2, is not necessarily equivalent to the coefficient of determination, R^2. This equivalence holds true specifically in the context of least squares regression when the same model and data are used for both fitting and evaluation. However, this may not be the case when the model is assessed using new data. To demonstrate the equivalence between r^2 and R^2 under these specific conditions, we begin by assuming that the covariance between the predicted values $\hat{Y}$ and the residuals ϵ is zero:

$$\begin{aligned} \text{cov}(Y, \hat{Y}) &= \text{cov}(\hat{Y} + \epsilon, \hat{Y}) \\ &= \text{cov}(\hat{Y}, \hat{Y}) + \text{cov}(\hat{Y}, \epsilon) \\ &= \text{var}(\hat{Y}) + \text{cov}(\hat{Y}, \epsilon) \\ &= \text{var}(\hat{Y}) \end{aligned} \tag{24}$$

With the assumption that $\bar{\hat{Y}} = \bar{Y}$, which typically holds when $\epsilon \sim N(0, \sigma^2)$, the squared correlation coefficient r^2 is expressed as follows:

$$\begin{aligned} r^2 &= \frac{\text{cov}^2(Y, \hat{Y})}{\text{var}(Y)\text{var}(\hat{Y})} \\ &= \frac{\text{var}(\hat{Y})^2}{\text{var}(Y)\text{var}(\hat{Y})} \\ &= \frac{\text{var}(\hat{Y})}{\text{var}(Y)} \\ &= \frac{\sum_{i=1}^{n} (\hat{Y}_i - \bar{\hat{Y}})^2}{\sum_{i=1}^{n} (Y_i - \bar{Y})^2} \\ &= \frac{\sum_{i=1}^{n} (\hat{Y}_i - \bar{Y})^2}{\sum_{i=1}^{n} (Y_i - \bar{Y})^2} \\ &= \frac{SS_{\text{residual}}}{SS_{\text{total}}} \\ &= R^2 \end{aligned} \tag{25}$$

where SS_{residual} is the residual sum of squares and SS_{total} is the total sum of squares. Each Y_i and $\hat{Y}_i$ are the ith elements of the actual response vector Y and the predicted response vector $\hat{Y}$, while $\bar{Y}$ and $\bar{\hat{Y}}$ are their respective means. This proof highlights that under certain assumptions, r^2 and R^2 can indeed be equivalent, but such conditions are specific

to least squares regression where errors are normally distributed and predictions are unbiased estimates of the actual values.

2 Conclusion

Your conclusion here

Acknowledgments

This was was supported in part by.....

References

[1] George Kour and Raid Saabne. Real-time segmentation of on-line handwritten arabic script. In *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, pages 417–422. IEEE, 2014.

[2] George Kour and Raid Saabne. Fast classification of handwritten on-line arabic characters. In *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, pages 312–318. IEEE, 2014.

[3] Guy Hadash, Einat Kermany, Boaz Carmeli, Ofer Lavi, George Kour, and Alon Jacovi. Estimate and replace: A novel approach to integrating deep neural networks with existing applications. *arXiv preprint arXiv:1804.09028*, 2018.

3 Headings: first level

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. See Section 3.

3.1 Headings: second level

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

$$\xi_{ij}(t) = P(x_t = i, x_{t+1} = j|y, v, w; \theta) = \frac{\alpha_i(t) a_{ij}^{w_t} \beta_j(t+1) b_j^{v_{t+1}}(y_{t+1})}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i(t) a_{ij}^{w_t} \beta_j(t+1) b_j^{v_{t+1}}(y_{t+1})} \tag{26}$$

3.1.1 Headings: third level

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Paragraph Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

4 Examples of citations, figures, tables, references

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum

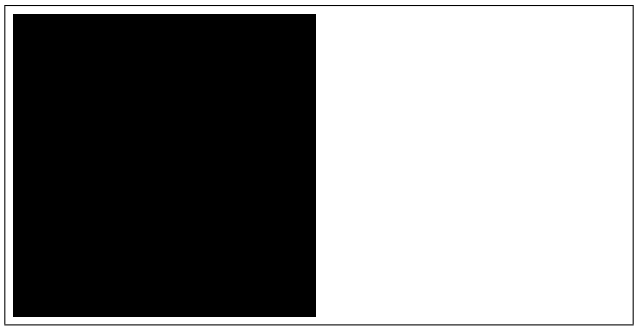

Figure 1: Sample figure caption.

fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui. [1, 2] and see [3].

The documentation for `natbib` may be found at

`http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf`

Of note is the command `\citet`, which produces citations appropriate for use in inline text. For example,

```
\citet{hasselmo} investigated\dots
```

produces

Hasselmo, et al. (1995) investigated…

`https://www.ctan.org/pkg/booktabs`

4.1 Figures

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi. See Figure 1. Here is how you add footnotes. [1] Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

[1] Sample of the first footnote.

Table 1: Sample table title

	Part	
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

4.2 Tables

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo. See awesome Table 1.

4.3 Lists

- Lorem ipsum dolor sit amet
- consectetur adipiscing elit.
- Aliquam dignissim blandit est, in dictum tortor gravida eget. In ac rutrum magna.