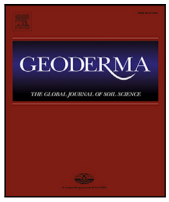
Geoderma 437 (2023) 116604



Contents lists available at ScienceDirect

Geoderma

journal homepage: www.elsevier.com/locate/geoderma





Model averaging of machine learning algorithms for digital soil mapping: A minimum variance framework

Patrick Bogaert [a,*], Ruhollah Taghizadeh-Mehrjardi [b], Nikou Hamzehpour [c]

[a] Earth and Life Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium
[b] Department of Geosciences, University of Tübingen, Tübingen, Germany
[c] Department of Soil Science and Engineering, Faculty of Agriculture, University of Maragheh, Maragheh, Iran

ARTICLE INFO

Handling Editor: Budiman Minasny

Keywords:
Best linear predictor
BLUP
Data fusion
Soil mapping
Urmia Lake

ABSTRACT

In the digital soil mapping framework, machine learning (ML) algorithms are currently the most popular methods for the spatial prediction of soil properties. The fast developments of easy-to-use software implementations for a large panel of ML algorithms have encouraged comparison studies between algorithms, with the goal of ranking their performances and identifying the best ones among them. However, as no firm conclusions can be drawn about the best ML algorithm to be used in general, this suggests that combining a set of them could be a better approach. Numerous methods have been proposed to do so, most of them relying on a linear weighting of the individual algorithms. However, there are almost as many methods for linearly weighting ML algorithms as there are ML algorithms, thus leaving the problem unsolved. Moreover, these weighting methods are mostly used out-of-the-box, without paying a proper attention to the associated hypotheses. In this paper, we propose to address this issue by setting the problem in a more formal framework. Starting from classical hypotheses, it is shown how the benefit of averaging various ML algorithms can be estimated from their joint performances. Relying afterwards on the most commonly used linear weighting schemes, it is reminded that, as long as the performance metrics are based on mean square errors, the best averaging method is by essence the best linear (unbiased) predictor. Using a more general Bayesian framework, it is also shown that accounting for conditional biases when weighting ML algorithms is a key issue for obtaining improved predictions, and explicit formulas are proposed for that goal. Finally, these theoretical results are illustrated and discussed using a soil data set collected over an arid and semi-arid region in Iran where clay content, calcium carbonate equivalent, soil organic carbon and electrical conductivity were measured in topsoil samples.

1. Introduction

In the realm of digital soil mapping (DSM), nonparametric and nonlinear machine learning (ML) algorithms are currently the most popular methods for the spatial predictions of various soil properties, either used alone or hybridized with more traditional geostatistical methods. Based on an extensive literature review, Chen et al. (2022) confirmed that ML methods have become by far the most commonly used predictive models in broad-scale DSM since 2011. In their review paper, Wadoux et al. (2020) pointed out that the wide adoption of ML for soil properties mapping was made possible by the increase in data availability, the ease of accessing environmental spatial data, and the development of many "easy-to-follow" software implementations, where the focus for model selection is most exclusively revolving around accurate predictions, instead of targeting a better understanding of mechanistic soil processes.

As emphasized by Zhang et al. (2022), this availability of a large panel of ML algorithms has encouraged model comparison studies, that have shown that different learners could generate digital soil maps that are drastically different. As a result, it has been recommended that model comparison should be carried out as part of best practice in DSM (Heung et al., 2016). This has led to many papers where several ML methods are used in parallel, with the goal of ranking their respective performances and identifying the best one amongst them, typically based on usual performance metrics. For predicting soil carbon fractions and total carbon in Florida, Keskin et al. (2019) used classification and regression tree (CART), bagged regression tree (BRT), gradient boosting regression tree (GBRT), random forest (RF), support vector machine (SVM), partial least square regression (PLSR), regression kriging (RK), ordinary kriging (OK) and compared their performances. Mishra et al. (2020) compared gradient boosting machine

* Corresponding author.
E-mail address: patrick.bogaert@uclouvain.be (P. Bogaert).

https://doi.org/10.1016/j.geoderma.2023.116604
Received 29 March 2023; Received in revised form 1 July 2023; Accepted 5 July 2023
Available online 18 July 2023
0016-7061/© 2023 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

(GBM), multinarrative adaptive regression spline, RF, SVM and RK to predict the spatial variation of surface soil organic carbon (SOC) stocks across the northern circumpolar permafrost region. Tajik et al. (2020) compared PLSR, generalized linear model (GLM), recursive partitioning and regression trees (RPART), SVM, RF and k-nearest neighbors (kNN) for predicting SOC at various soil depths in Iran, where the respective performances of these methods depend on the depth. Rostaminia et al. (2021) used RF and Cubist (CB) along with their hybridized (kriging) version for predicting of SOC stocks in an arid rangeland, Iran, while Zeraatpisheh et al. (2022) used RF, CB, SVM, and PLSR for mapping SOC using environmental covariates.

In a soil salinity framework, Vermeulen and Van Niekerk (2017) used CART, SVM, RF and kNN for predicting salt-affected soils within the Vaalharts and Breede River irrigation schemes of South Africa. Wu et al. (2018) used support vector regression (SVR) and RF algorithms to estimate soil salinity in Iraq. In three distinct papers, Wang et al. (2019, 2020, 2021) used various combinations of Least Absolute Shrinkage and Selection Operator (LASSO), Multiple Adaptive Regression Splines (MARS), CART, RF, Stochastic Gradient Treeboost (SGT), PLSR, artificial neural network (ANN), convolutional neural network (CNN) and SVM for predicting of soil salinity in various areas in China. Zarei et al. (2021) compared GBM, Extreme Gradient Boosting (XGB) and RF to predict the spatial variation of surface soil salinity in Iran, while (Ge et al., 2022) compared the results obtained from bagging, CART, RF and GBRT when updating a soil salinity map in the Ebinur Lake basin, China. In a soil conservation framework in northern Taiwan, Nguyen et al. (2021) used various bagging, boosting and stacking techniques that involved RF, CB, GBM, SVM and ANN.

Clearly, such studies offer the benefit of comparing a wide panel of ML techniques but, beyond the strict framework of the data sets that were used, no firm conclusions can be drawn from them about the best method to be recommended. Indeed, no single ML method outperforms the other ones in general, and the best method can differ in the same study, e.g. when predicting distinct soil properties or when predicting a same soil property but at distinct depths (Tajik et al., 2020). Moreover, as no inference is conducted in general about the uncertainty associated with the estimated performance metrics, selecting the best method based on these ranked metrics is arguable when several ML exhibit quite comparable results. These difficulties motivated the more recent use of averaging techniques, where results from several ML methods are combined, thus avoiding the need to select a single method and with the hope of improving prediction quality. With this goal in mind, Nketia et al. (2022) used a weighted average of RF and GBM, where the weights are based on their respective squared-prediction errors, i.e., a Bates–Granger averaging (BGA). Malone et al. (2014) compared equal weights averaging (EQA), BGA, Granger–Ramanathan averaging (GRA) and Bayesian model averaging (BMA) with finite mixture model and recommend the use of GRA as a good balance between performance and implementation easiness. O'Rourke et al. (2016) compared BGA and GRA for a large set of soil properties and reached the same conclusion, although the benefit of averaging largely varied between properties. For soil texture mapping, Román Dobarco et al. (2017) and Caubet et al. (2019) compared BGA and GRA too and concluded that GRA yields better performance and improved accuracy, but to various extent depending on the predicted fraction. Chen et al. (2020) compared GRA, BMA, CB, residual-based CB and a bias-corrected Variance Weighted (BC-VW) approach; they concluded that all averaging approaches performed better than individual SOC maps and recommended BC-VW as the approach for model averaging. Taghizadeh-Mehrjardi et al. (2021) used the results from twelve ML models to construct a so-called super learner (SL), which is a convex linear weighting of these models, that improved the accuracy of the predicted soil maps in comparison with the individual models. Zhang et al. (2022) used the same SL approach for averaging the results from nine ML models for predicting soil pH at three depths intervals; they concluded that SL does not necessarily outperformed the best ML method. Finally, Taghizadeh-Mehrjardi et al. (2022) used the results from nine ML models with a large panel of averaging techniques, i.e., EQA, BGA, GRA, BMA and two information criterion averaging (ICA) based on Akaike's information criterion (AIC) and Bayes information criterion (BIC); they concluded that averaging consistently performed better than the best single ML technique, regardless of model averaging techniques and soil properties, where GRA and BMA were the best performing methods for all soil properties.

As seen from this literature review, the consensus is that averaging results of multiple ML methods is beneficial in general, but the extent of this benefit is largely varying and the underlying reasons for that are not investigated at all. Moreover, as there are potentially as many averaging techniques as there are individual ML methods, the problem of selecting the best single ML method is thus replaced with the problem of selecting the best averaging technique, with no theoretical guidelines that are offered for that goal. In this paper, we propose to bridge this gap by setting the problem in a more formal context. Starting from classical hypotheses, it is shown how the benefit of averaging various ML methods can be estimated from their joint performances. Relying afterwards on the linear weighting paradigm which is the most commonly used, it is reminded that, as long as the performance metrics are based on mean square errors, the best linear averaging technique is by essence the best linear (unbiased) predictor, as classically derived from statistical theory. Finally, using a more general Bayesian framework, it is shown that accounting for conditional biases when merging ML results is a key issue for obtaining improved predictions. Corresponding simple analytical results are also proposed for that goal under realistic hypotheses. These results are finally illustrated and discussed when merging the results from five ML algorithms that aim at predicting several soil properties in an arid and semi-arid region of Iran.

2. Material and methods

2.1. Study area and sampling

For illustrating the theoretical results presented in this paper, we used a data set collected in an arid and semi-arid area in the northwest part of Iran. All measurements are located close to the western shoreline of the Urmia Lake, which is the greatest salty lake in the Middle East and the sixth largest saltwater lake on Earth (Fig. 1). The study site covers an area of about 41 km^2. The geographical coordinates of the area range from 45°06′00" to 45°23′00" E and 37° 19′00" to 37°56′00" N. Its western boundary is adjacent to the Urmia Plain and extends towards Urmia Lake for almost 6 km on the east. The elevation ranges from 1270 to 1300 meters a.s.l., while the average annual precipitation and mean annual temperature are 367 mm and 13.4 °C, respectively. According to the Iranian Ministry of Energy (2014), the potential evaporation rate of the region ranges from 900 to 1170 mm. Geologically, the study area is composed of playas and alluvial deposits (Geological Survey of Iran, 1995).

Recently, the lake has been highly endangered and reduced by about 88%. Salt accumulation in surrounding soils is a major issue. A monitoring of the soil properties was thus conducted in July 2018. A stratified random sampling method was used to select the sampling locations. First, various playa surfaces were identified and 128 surficial soil samples (at a depth of 0–5 cm) were collected from these areas. The soil samples were passed through a 2 mm sieve and various soil properties were measured. This includes clay content (in %), calcium carbonate equivalent (CCE, in %), SOC (in %) and electrical conductivity (EC, in dS/m). Clay content was measured using a hydrometer method (Gee and Or, 2002). CCE was quantified using a back titration of the remaining HCl (Page et al., 1982). SOC was determined by using wet oxidation technique (Nelson and Sommers, 1996). EC was measured in a saturated paste extract using a Jenwey conductivity meter (model 4510) following the method proposed by Rhoades (1996).

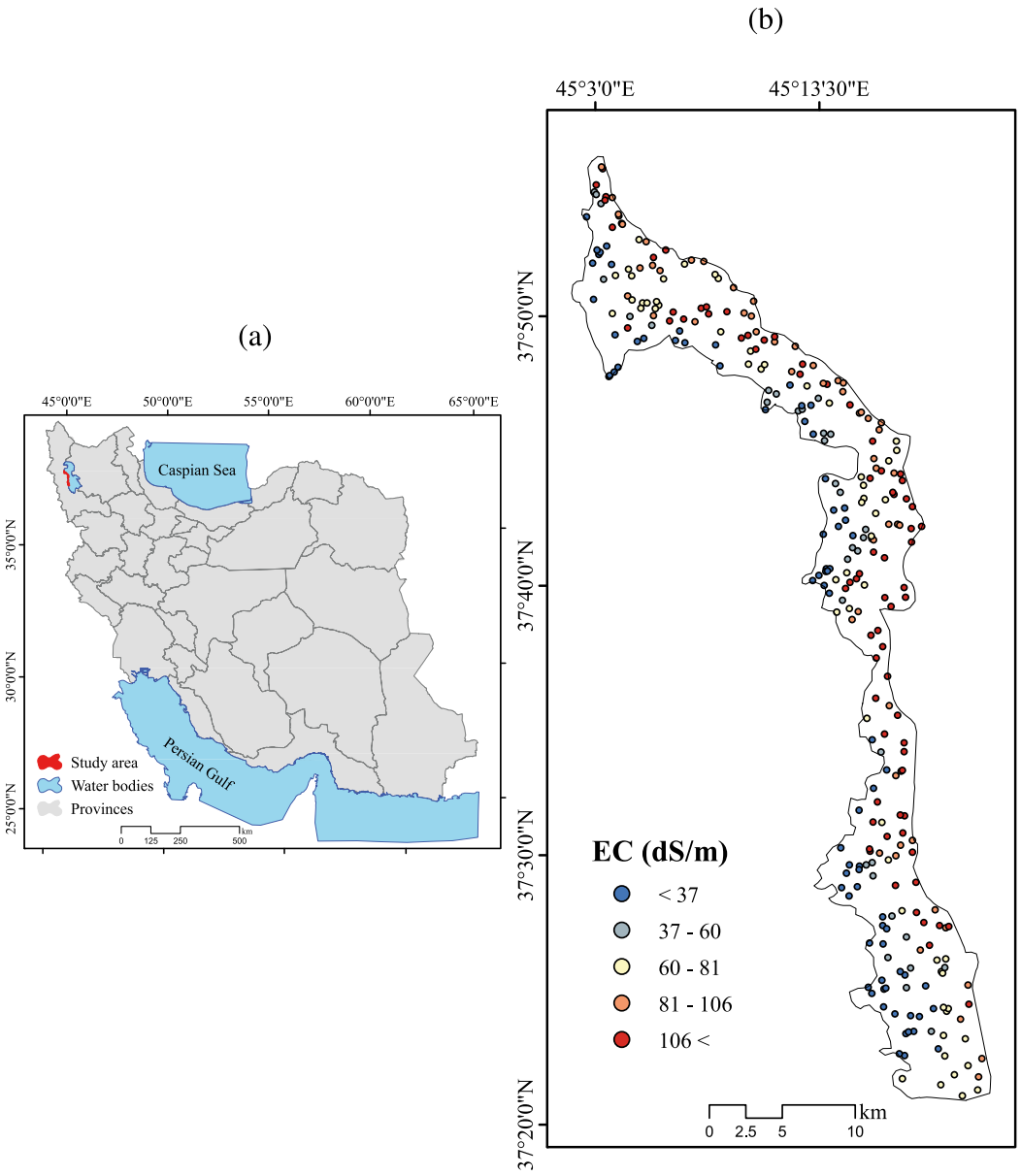


Fig. 1. Location of the study area in Iran (a) and location of the soil samples inside the study area (b), with values for electrical conductivity (EC) that are presented for the sake of illustration.

2.2. Machine learning algorithms

For predicting soil properties over the study area, various ML algorithms were considered and compared. In order to account for the primary soil-forming factors of the region, a wide range of environmental covariates was first selected. This included remotely sensed images from Landsat −8 (Roy et al., 2014) and Sentinel-2 (Drusch et al., 2012) that were acquired in July 2018, along with a DEM (O'Loughlin et al., 2016). Euclidean distances (Behrens et al., 2018) and Oblique distances (Møller et al., 2020) were also used as covariates. The DEM was used to compute various terrain attributes such as slope and aspect at a 30 m spatial resolution in the SAGA system (Passy and Théry, 2018). Subsequently, the covariates were upscaled using simple smoothing (mean) filters with various neighborhood sizes ranging from 30 to 900 meters for capturing the spatial distribution of soil properties. Altogether, a total of 1,480 environmental covariates were defined but highly correlated covariates (with correlations greater than 90%) were discarded. Only 55 covariates were finally selected at various spatial scales for training the ML models (see Table 1).

Five distinct ML algorithms were considered in this study, namely k-nearest neighbors (kNN), random forest (RF), extreme gradient boosting (XGB), cubist (CB) and support vector regression (SVR). A brief description of these algorithms is provided in Appendix C. The georeferenced sample points were thus overlaid with the selected environmental covariates, and the ML models were trained and tested to predict soil properties in the study area. Extended technical details about covariates selection, predictions and ML algorithms are given in Taghizadeh-Mehrjardi et al. (2021) and are not repeated here for the sake of brevity. Table C.1 provides the values of the optimized hyperparameters for the various machine learners and soil properties.

For avoiding overfitting issues and allowing internal validation, a leave-one-out cross-validation (LOOCV) was considered for tuning the ML algorithms and for comparing at the same time the benefits of the various methods that aim at fusing the separate predictions from these algorithms. Considering alternately each of the $n = 128$ samples for a given soil property, the five ML algorithm were trained $n - 1$ times using the other $n - 1$ values of the samples and the average of these $n - 1$ predictions were used as the predicted value for the

Table 1
Environmental covariates and spatial scales (in meters) that were considered for predicting soil properties. The 55 crosses indicate those that were considered after removing the highly correlated ones.

Environmental covariates	30	60	90	150	300	450	700	900
Elevation	x				x			
Aspect		x				x		x
Average gradient above flow path	x			x			x	
Slope Length and Steepness factor	x				x			x
Valley depth	x		x			x		
Curvature	x	x		x		x	x	
Channel networks base level	x					x		
Topographic wetness index	x			x				
Multi-resolution Valley Bottom Flatness Index	x							x
Multi-resolution Ridge-top Flatness Index			x			x		
Blue band of Landsat-8 [0.482 μm]		x					x	
Red band of Landsat-8 [0.654 μm]	x							
Near-infrared band of Landsat-8 [0.864 μm]	x					x		
Shortwave IR-1 band of Landsat-8 [1.608 μm]	x							
Blue band of Sentinel-2 [0.490 μm]	x							
Vegetation Red Edge of Sentinel-2 [0.705 μm]					x			
Vegetation Red Edge of Sentinel-2 [0.783 μm]		x						
Near-infrared band of Sentinel-2 [0.842 μm]							x	
Vegetation Red Edge of Sentinel-2 [0.865 μm]				x				
Shortwave IR-2 band of Sentinel-2 [2.190 μm]	x							
Transformed vegetation index(Landsat-8 based)	x							
Salinity Index (Landsat-8 based)							x	
Brightness Index (Landsat-8 based)			x					
Clay Index (Landsat-8 based)	x							
Transformed vegetation index(Sentinel-2 based)						x		
Salinity index (Sentinel-2 based)								x
Brightness Index (Sentinel-2 based)				x				
Carbonate index (Sentinel-2 based)	x			x				x
Depth-to-the-groundwater	x							
Euclidean distances (128 maps)	x	x					x	
Oblique distances (18 maps)	x				x			

actual value that was not accounted for during this training process. The whole procedure is done separately for each of the soil properties to be predicted. The validation stage thus consists in comparing these averaged predictions to the actual values for the various soil properties, as well as comparing the fused predictions to the actual values. A more classical procedure would be to build first a training and a validation data sets using a random split of the $n = 128$ samples and to apply a k-fold cross-validation for tuning first the algorithms using the training data set, then to check afterwards for the prediction performances using the validation data set. However, at the light of our data set and the goal of our study, this procedure proves to be inconvenient. A typical 75%–25% random split of the data set would lead to a sample size of 32 for the validation data set, which impairs our ability to draw robust conclusions as the results may largely depend on the random selection for this sample. This could be avoided by using a nested cross-validation procedure where the whole data set is repeatedly and randomly split into a training and validation data set, as used by Zhang et al. (2022). However, as we aim at comparing the performances of various methods for fusing the separate predictions of these ML algorithms as well, this would require in turn a separation between a training and a validation data sets for this purpose. In this context, the proposed LOOCV is easier to implement, even if LOOCV is computationally much more demanding as the whole tuning process of the ML algorithms needs to be done n times instead of k times when using k-fold cross-validation, where a typical value is $k = 10$. It is mentioned in the literature that LOOCV can be subject to overfitting. However, tests that were conducted (not shown here) confirmed that results for LLOCV and 10-fold cross-validation where consistent, with no obvious overfitting issues.

The corresponding results for the prediction performances of the soil properties using the five ML algorithms are presented in Table 2. Overall, it can be seen that EC is the best predicted property, while CCE is the worst one. With respect to the determination coefficient R^2 and Lin's concordance correlation coefficient R_L, it can be seen too that the performances of the ML algorithms depend on the soil property, with best performances obtained using RF for clay content, CCE and EC while SOC is better predicted using XGB. It is worth noting here that R^2 is measuring the part of the variance that is explained by each predictor, while R_L is measuring the agreement between measurements and predicted values with respect to the 1:1 line, that corresponds to the expected result for a perfect predictor. See Lin (1989) for more details.

Table 2
Main performance metrics of the various predictors using LOOCV, including mean error (ME), root-mean-square error (RMSE), determination coefficient (R^2) and Lin's concordance correlation coefficient (R_L). Parts (a), (b), (c) and (d) refer to clay content, CCE, SOC and EC, respectively.

Method	ME	RMSE	R^2	R_L	ME	RMSE	R^2	R_L
		(a)				(b)		
kNN	−1.717	8.493	0.546	0.668	0.373	4.771	0.331	0.427
RF	−0.092	7.659	0.615	0.726	−0.071	4.572	0.382	0.437
XGB	0.046	7.719	0.609	0.748	0.219	4.633	0.367	0.510
CB	−0.010	8.321	0.546	0.700	0.347	5.231	0.195	0.360
SVR	0.315	8.180	0.562	0.694	0.274	4.629	0.369	0.430
		(c)				(d)		
kNN	−0.086	0.445	0.507	0.627	2.268	19.371	0.739	0.851
RF	−0.054	0.417	0.559	0.650	2.206	18.487	0.762	0.858
XGB	−0.007	0.365	0.657	0.787	1.478	18.625	0.757	0.862
CB	0.016	0.404	0.580	0.725	3.434	18.972	0.754	0.853
SVR	0.029	0.373	0.642	0.743	3.781	18.947	0.757	0.848

Legend : kNN = k-nearest neighbor, RF = random forest, XGB = extreme gradient boosting, CB = cubist, SVR = support vector regression.

As an illustration, Figs. 2 and 3 are visually presenting the results of the five predictors for SOC and EC, respectively. For SOC, it can be seen that low values tend to be overestimated while it is the opposite for high values, and this is particularly clear for kNN and RF. This effect is less obvious for EC, that seems less impacted by this conditional bias in general.

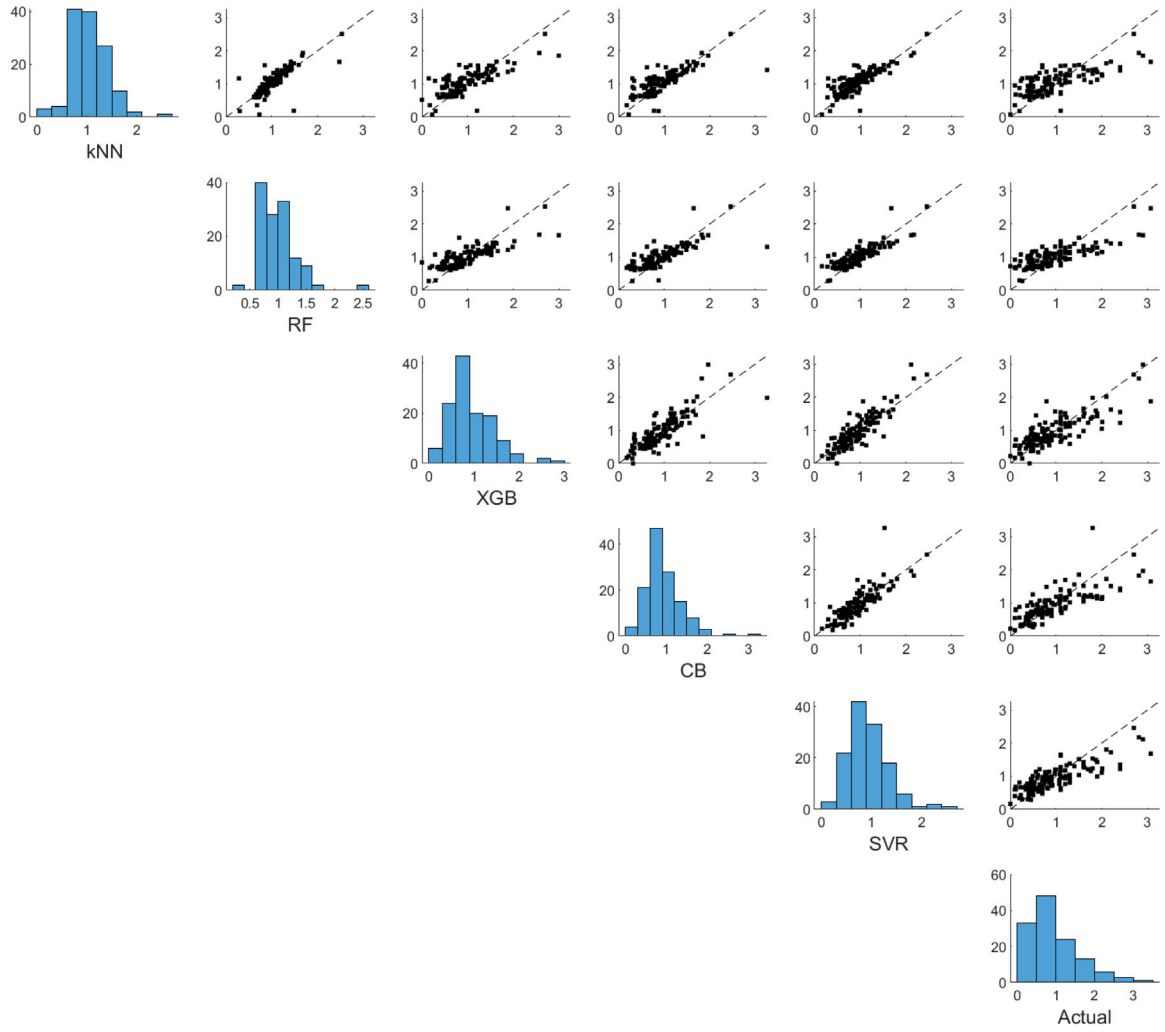


Fig. 2. Histogram of measured SOC values ($n = 128$), along with histograms of the corresponding ML predicted values and their scatter plots. Last column is relating actual SOC values (on the x-axis) with the various predictions (on the y-axis). Dashed line is the 1:1 line.

2.3. The general framework

In order to ease the discussion and to make it as general as possible, we are considering here that our aim is to predict at best a random variable Y (i.e. either clay content, CCE, SOC or EC values in the framework of our data set) by using a set of various predictors $X_1, \ldots, X_n$ (i.e., the set of n individual ML algorithms), where the prediction errors $\varepsilon_i = Y - X_i$ are likely not independent from each other. For the sake of notational simplicity, let us define $\mathbf{X} = (X_1, \ldots, X_n)'$ and $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)'$ as the vector of predictors and prediction errors, respectively. We aim here at obtaining a single predictor Y^p of Y that combines at best the various predictors in $\mathbf{X}$. From standard prediction theory (see, e.g., Papoulis and Pillai, 2002), it is known that the best predictor (in the mean square sense) of Y based on $\mathbf{X}$ is the conditional expectation $\mu_{Y|\mathbf{x}}$. Stated in other words, setting $y^p = \mu_{Y|\mathbf{x}}$ yields the lowest possible value for $Var[Y - Y^p]$. More generally, we could also be interested by the full conditional distribution $f(y|\mathbf{x})$ from which this conditional expectation $\mu_{Y|\mathbf{x}}$ and the corresponding conditional variance $\sigma^2_{Y|\mathbf{x}}$ can be computed afterwards.

Starting from a joint multivariate normal distribution hypothesis, we will show hereafter how $\mu_{Y|\mathbf{x}}$ and $\sigma^2_{Y|\mathbf{x}}$ are easily obtained as long as there is no conditional bias. Under these assumptions, the corresponding results allow assessing the benefit of merging several predictors, by providing a lower bound for $\sigma^2_{Y|\mathbf{x}}$. As $\mu_{Y|\mathbf{x}}$ is also a linear weighting of $\mathbf{x}$ under these assumptions, we will briefly remind that most of the commonly used merging techniques are based on a similar linear weighting paradigm and we will parallel them with the derivation of the best linear (unbiased) predictor. Finally, we will set the problem in a more general Bayesian framework that permits addressing the conditional bias issue. We will finally set the conditions that lead to simple formulas for computing $\mu_{Y|\mathbf{x}}$ and $\sigma^2_{Y|\mathbf{x}}$ using this Bayesian framework.

2.4. Multivariate normal approach

For the subsequent developments, we consider that prediction residuals $\boldsymbol{\varepsilon}$ can be considered as independent from Y, so that we can write

$$\boldsymbol{\varepsilon} \perp Y \quad \Longleftrightarrow \quad f(\boldsymbol{\varepsilon}|y) = f(\boldsymbol{\varepsilon}) \quad \forall y \tag{1}$$

This automatically entails that

$$E[\varepsilon_i|y] = \mu_{\varepsilon_i} \quad \forall i \quad ; \quad Cov[\varepsilon_i, \varepsilon_j|y] = \sigma_{\varepsilon,ij} \quad \forall i,j \tag{2}$$

or, in matrix notation,

$$\boldsymbol{\mu}_{\boldsymbol{\varepsilon}|y} = \boldsymbol{\mu}_{\boldsymbol{\varepsilon}} \quad ; \quad \Sigma_{\boldsymbol{\varepsilon}|y} = \Sigma_{\boldsymbol{\varepsilon}} \tag{3}$$

where $\boldsymbol{\mu}_{\boldsymbol{\varepsilon}} = \{\mu_{\varepsilon_i}\}$ and $\Sigma_{\boldsymbol{\varepsilon}} = \{\sigma_{\varepsilon,ij}\}$ are the expectation vector and covariance matrix of $\boldsymbol{\varepsilon}$, respectively. As $X_i = Y - \varepsilon_i$, the left part of

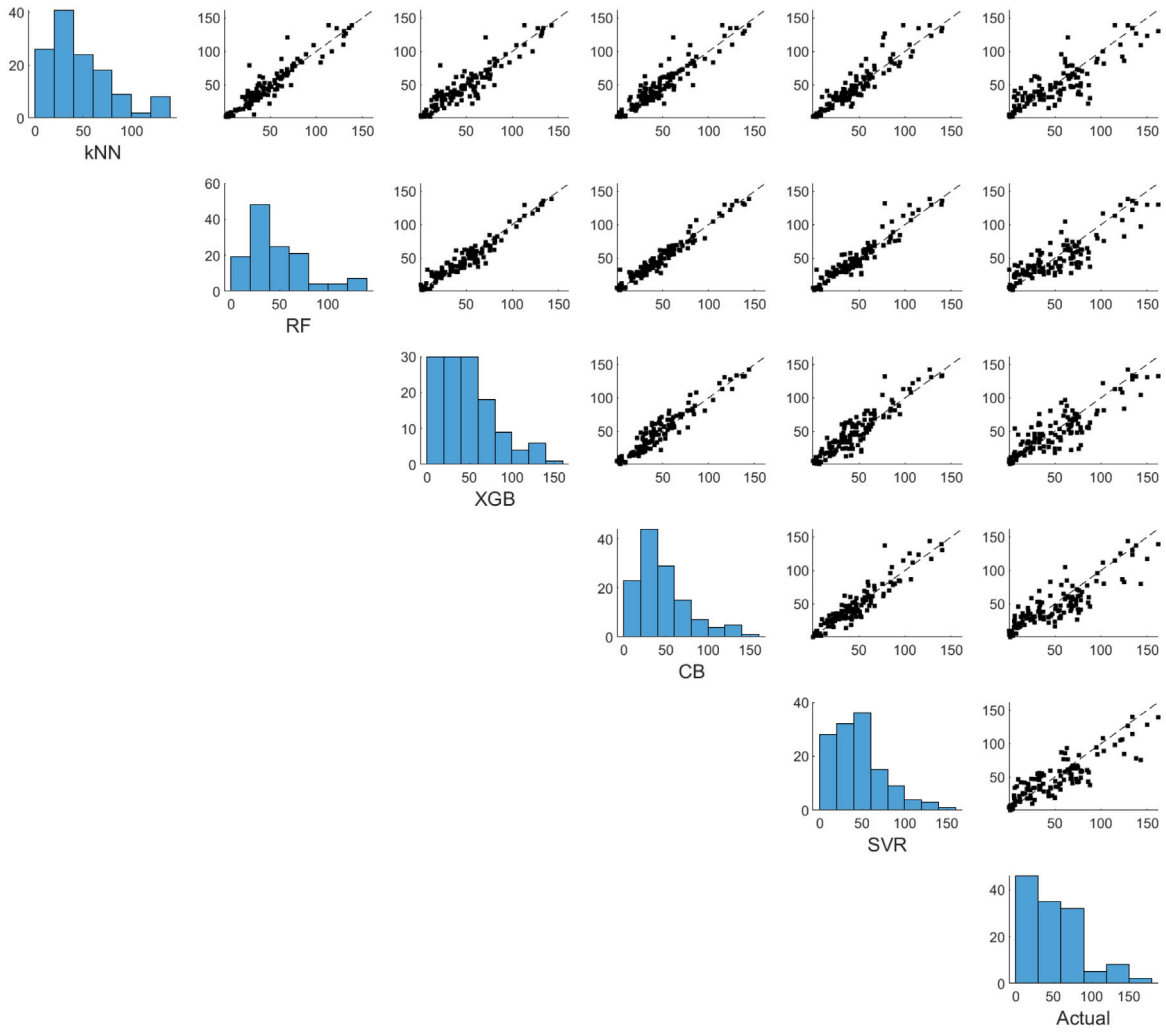


Fig. 3. Histogram of measured EC values (n = 128), along with histograms of the corresponding ML predicted values and their scatter plots. Last column is relating actual EC values (on the x-axis) with the various predictions (on the y-axis). Dashed line is the 1:1 line.

Eq. (2) is equivalent to state that $E[X_i|y] = y - \mu_{\varepsilon_i}$, i.e. each X_i predictor of Y is conditionally unbiased up to an additive constant μ_{ε_i}. Stated in other words, for any given value y, each X_i predictor yields on the average the correct y value up to this additive constant μ_{ε_i}, that does not depend on y and where $\mu_{\varepsilon_i} \neq 0$ in general. The right part of Eq. (2) states additionally that the variances and covariances of the prediction residuals are the same whatever the value for y.

Let us now assume that Y and $\boldsymbol{\varepsilon}$ are univariate and multivariate normal distributions, respectively, that are completely characterized by their corresponding expectations $\mu_Y, \boldsymbol{\mu}_\varepsilon$ and variances/covariances $\sigma_Y^2, \Sigma_\varepsilon$. From the independence assumption in Eq. (1), we thus have

$$\begin{pmatrix} Y \\ \boldsymbol{\varepsilon} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_Y \\ \boldsymbol{\mu}_\varepsilon \end{pmatrix}, \begin{pmatrix} \sigma_Y^2 & \mathbf{0}' \\ \mathbf{0} & \Sigma_\varepsilon \end{pmatrix} \right) \tag{4}$$

where $\mathbf{0}$ is the null vector. As a result, with $X_i = Y - \varepsilon_i \ \forall i$, we can write

$$\begin{pmatrix} Y \\ \mathbf{X} \end{pmatrix} = \begin{pmatrix} 1 & \mathbf{0}' \\ \mathbf{1} & -\mathbf{I} \end{pmatrix} \begin{pmatrix} Y \\ \boldsymbol{\varepsilon} \end{pmatrix} \tag{5}$$

where $\mathbf{1}$ is the unit vector and $\mathbf{I}$ is the identity matrix. Clearly, as the joint distribution of Y and $\boldsymbol{\varepsilon}$ is multivariate normal and because $\mathbf{X}$ is expressed as a linear combination of Y and $\boldsymbol{\varepsilon}$, the joint distribution of Y and $\mathbf{X}$ is thus multivariate normal too, with

$$\begin{pmatrix} Y \\ \mathbf{X} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_Y \\ \mu_Y \mathbf{1} - \boldsymbol{\mu}_\varepsilon \end{pmatrix}, \begin{pmatrix} \sigma_Y^2 & \sigma_Y^2 \mathbf{1}' \\ \sigma_Y^2 \mathbf{1} & \sigma_Y^2 \mathbf{1}\mathbf{1}' + \Sigma_\varepsilon \end{pmatrix} \right) \tag{6}$$

2.4.1. Conditional distributions and best predictor

From the well-known properties of the multivariate normal distribution (see e.g. Tong, 2012), the conditional distribution $f(y|\mathbf{x})$ is then univariate normal, with

$$Y|\mathbf{x} \sim N(\mu_{Y|\mathbf{x}}, \sigma_{Y|\mathbf{x}}^2) \tag{7}$$

where the conditional $\mu_{Y|\mathbf{x}}$ expectation and conditional variance $\sigma_{Y|\mathbf{x}}^2$ are given by

$$\mu_{Y|\mathbf{x}} = \mu_Y + \sigma_Y^2 \mathbf{1}'(\sigma_Y^2 \mathbf{1}\mathbf{1}' + \Sigma_\varepsilon)^{-1}(\mathbf{x} - \mu_Y \mathbf{1} + \boldsymbol{\mu}_\varepsilon) \tag{8}$$

$$\sigma_{Y|\mathbf{x}}^2 = \sigma_Y^2 - \sigma_Y^4 \mathbf{1}'(\sigma_Y^2 \mathbf{1}\mathbf{1}' + \Sigma_\varepsilon)^{-1}\mathbf{1} \tag{9}$$

Similarly, as we will need this result later on in a Bayesian framework, the joint conditional distribution $f(\mathbf{x}|y)$ is multivariate normal too, with

$$\mathbf{X}|y \sim N(\mu_{\mathbf{X}|y}, \Sigma_{\mathbf{X}|y}) \tag{10}$$

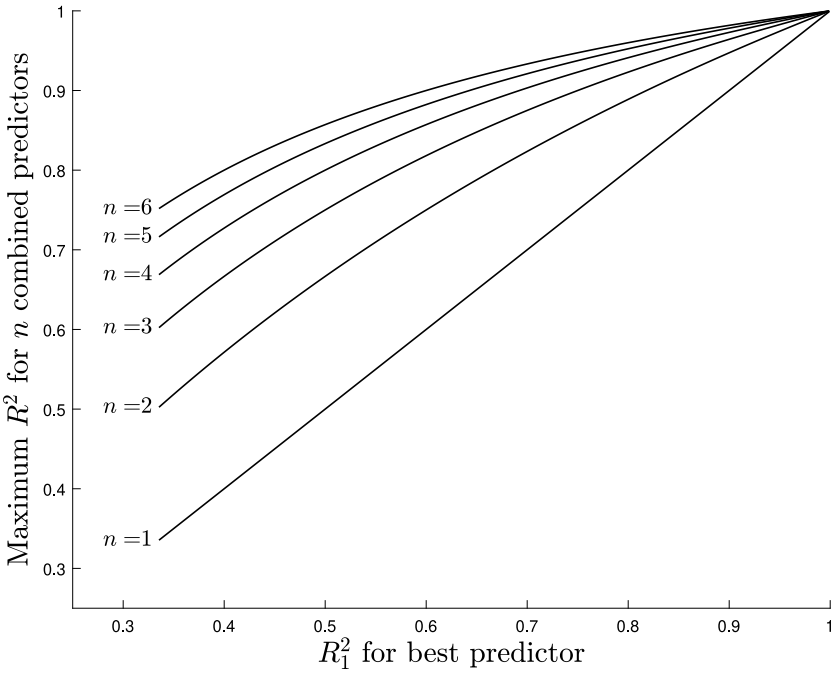


Fig. 4. Determination coefficient R^2 when combining n predictors as a function of the determination coefficient R_1^2 for the best predictor and the number n of predictors. For a given R_1^2, each curve gives the maximum R^2 that can be reached when assuming independence and identical (minimum) variance for the prediction errors.

where one obtains after simplifications, as expected from Eqs. (1) and (2), the results

$$\mu_{\mathbf{X}|y} = y\mathbf{1} - \boldsymbol{\mu}_{\boldsymbol{\epsilon}} \quad ; \quad \Sigma_{\mathbf{X}|y} = \Sigma_{\boldsymbol{\epsilon}} \tag{11}$$

As a summary, if one assumes $Y \perp \boldsymbol{\epsilon}$ and multivariate normality for Y and $\boldsymbol{\epsilon}$, the conditional distribution $f(y|\mathbf{x})$ that accounts for all predictors together is thus easily obtained, and its expectation $\mu_{Y|\mathbf{x}}$ as given by Eq. (8) is the best predictor (in the mean square sense) of Y given the various X_i's.

Using Eq. (8) requires of course inference about the expectations and variances/covariances. As the various predictors are trained using a set of K observed values for Y, we have at hand for each observed values y its corresponding predictions $x_1, \ldots, x_n$ and prediction errors $y - x_1, \ldots, y - x_n$, so that

$$\begin{gathered} \widehat{\mu}_Y = \frac{1}{K}\sum_{k=1}^{K} y_k \quad ; \quad \widehat{\sigma}_Y^2 = \frac{1}{K-1}\sum_{k=1}^{K}(y_k - \widehat{\mu}_Y)^2 \\ \widehat{\mu}_{\epsilon_i} = \frac{1}{K}\sum_{k=1}^{K}(y_k - x_{ik}) \quad ; \quad \widehat{\sigma}_{\epsilon,ij} = \frac{1}{K-1}\sum_{k=1}^{K}(y_k - x_{ik} - \widehat{\mu}_{\epsilon_i})(y_k - x_{jk} - \widehat{\mu}_{\epsilon_j}) \end{gathered} \tag{12}$$

where y_k is one of these observed values and where $x_{1k}, \ldots, x_{nk}$ are the corresponding predicted values from the n predictors.

2.4.2. Variance lower bound when combining predictors

An important question is to assess the benefit of combining n predictors instead of just selecting the best predictor among the n available ones. For doing this, let us assume without loss of generality that predictors are ranked in increasing order of variances $\sigma_{\epsilon_i}^2$ so that, when used separately, X_1 is the best predictor and X_n is the worst one (in the mean square sense). After simplification of Eq. (9), we have for a single X_i predictor the result

$$\sigma_{Y|x_i}^2 = \sigma_Y^2\left(1 - \frac{\sigma_Y^2}{\sigma_Y^2 + \sigma_{\epsilon_i}^2}\right) \qquad \forall i = 1, \ldots, n \tag{13}$$

so that $\sigma_{Y|x_1}^2 \leq \sigma_{Y|x_i}^2 \ \forall i$. On the other side, when using all predictors together, the minimum possible value for $\sigma_{Y|\mathbf{x}}^2$ is reached when all prediction errors are independent and where $\sigma_{\epsilon_i}^2 \to \sigma_{\epsilon_1}^2 \ \forall i$, so that Eq. (9) now becomes

$$\sigma_{Y|\mathbf{x}}^2 = \sigma_Y^2 - \sigma_Y^4 \mathbf{1}'(\sigma_Y^2 \mathbf{1}\mathbf{1}' + \sigma_{\epsilon_1}^2 \mathbf{I})^{-1}\mathbf{1} \tag{14}$$

A comparison of $\sigma_{Y|x_1}^2$ with $\sigma_{Y|\mathbf{x}}^2$ as given by Eqs. (13) and (14) thus provides the requested information about the maximum possible reduction of variance that one could expect by combining n predictors instead of selecting the best one among them. For the sake of illustration, Fig. 4 shows the relationship between the corresponding determination coefficients R_1^2 and R^2 when using the single best predictor X_1 or when using n predictors, respectively, with

$$R_1^2 = 1 - \frac{\sigma_{Y|x_1}^2}{\sigma_Y^2} \quad ; \quad R^2 = 1 - \frac{\sigma_{Y|\mathbf{x}}^2}{\sigma_Y^2} \tag{15}$$

with $R_1^2 \leq R^2$. It can be seen that increasing the number n of predictors does increase R^2, but the benefit of an additional predictor becomes less and less important as n increases. Clearly, the relative benefit (as measured by R^2/R_1^2) of using jointly several predictors is limited when the best predictor has a R_1^2 value close to 1, while it increases as R_1^2 tends to 0. From a theoretical viewpoint, this emphasizes the advantage of combining several predictors and it provides at the same time a clear upper bound for R^2 when combining n of them in a multivariate normal framework. It is worth emphasizing that much lower R^2 values could be obtained for correlated prediction errors or for predictors having higher variances $\sigma_{\epsilon_i}^2$ than $\sigma_{\epsilon_1}^2$ (with in particular $\sigma_{Y|\mathbf{x}}^2 = \sigma_{Y|x_1}^2$ and $R^2 = R_1^2$ when all predictors are perfectly correlated). It however provides useful reference values about what can be expected at best from a combination of n predictors.

2.5. Linear weighting, BLP and BLUP

Under the multivariate normal assumption, it is worth emphasizing that the best predictor (in the mean square sense) as given by Eq. (8) is a linear weighting of the individual predictors. Indeed, Eq. (8) can be rewritten as

$$y^p - \mu_Y = \mathbf{w}'(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}) \tag{16}$$

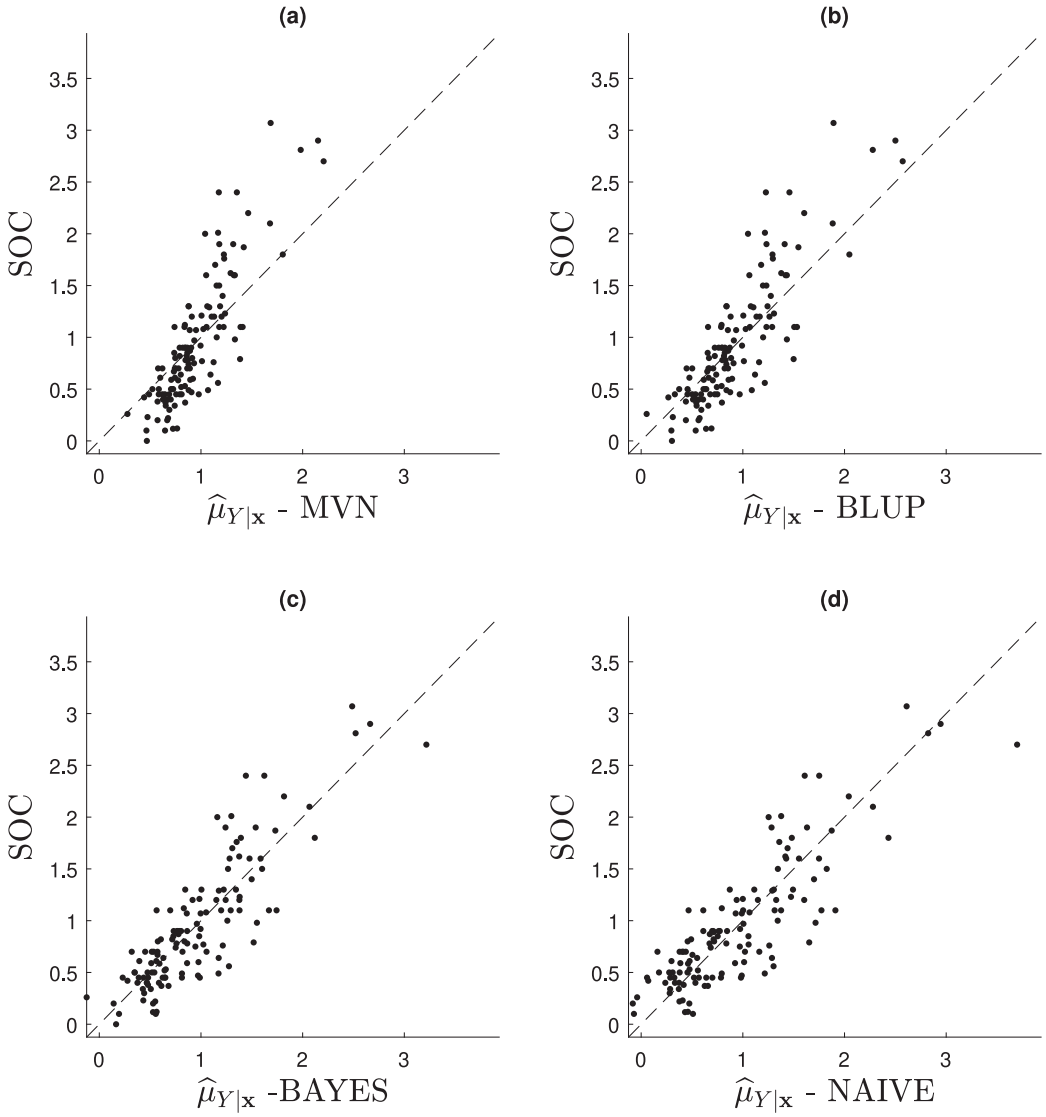


Fig. 5. Scatter plots of measured SOC values versus the averaged predicted SOC values using (a) multivariate normal approach, (b) best linear unbiased predictor, (c) Bayes approach and (d) Naive Bayes approach. Dashed line is the 1:1 line.

with $y^p = \mu_{Y|\mathbf{x}}$, where $\mathbf{w}' = (w_1, \ldots, w_n)$ is a vector of weights, with $\mathbf{w}' = \sigma_Y^2 \mathbf{1}'(\sigma_Y^2 \mathbf{1}\mathbf{1}' + \Sigma_\epsilon)^{-1}$, and where $\mathbf{x} - \mu_\mathbf{X}$ is the zero-mean centered vector of predictors, with $\mu_\mathbf{X} = \mu_Y \mathbf{1} - \boldsymbol{\mu}_\epsilon$ from eq. (6)1sd . This brings to the fore the connection with a linear predictor given by

$$y^p = w_1 x_1 + \cdots + w_n x_n = \mathbf{w}'\mathbf{x} \tag{17}$$

and with the various linear weighting schemes that are frequently used in the DSM literature, where weights are typically constrained to sum up to one (or even constrained to be non negative) and are obtained from various heuristics. The most commonly found are equal weighting averaging (EQA), Bates–Granger averaging (BGA), Granger–Ramanathan averaging (GRA), Information criterion averaging (ICA) and Bayesian Model Averaging (BMA). EQA is the simplest one, with weights $w_i = 1/n$. BGA (Bates and Granger, 1969) gives weights $w_i \propto 1/\sigma^2_{\epsilon,i}$, thus just accounting for the variances of the prediction errors. GRA (Granger and Ramanathan, 1984) assigns weights that are the (ordinary) least square parameter estimates of $\mathbf{w}$ when Eq. (17) is interpreted as a multiple linear regression model, i.e., when $\mu_{Y|\mathbf{x}} = \mathbf{w}'\mathbf{x}$. ICA (Buckland et al., 1997, 2002) assigns weights $w_i \propto \exp(-I_i/2)$ where I_i is involving the log-likelihood for the ith predictor and a penalty for the number of parameters, as given either from the Akaike's information criterion (AIC) or Bayes information criterion (BIC). BMA (Hoeting et al., 1999) proposes various ways of assigning weights but corresponds to GRA in the context of Eq. (17). A more extensive description of these averaging methods is given by Diks and Vrugt (2010).

Whatever the numerous possible ways of heuristically assigning weights $\mathbf{w}$ in Eq. (17), there is however a single possible choice for $\mathbf{w}$ that minimizes $Var[Y - Y^p]$, and this corresponds to the best linear (unbiased) predictor (BLP and BLUP) of Y. For the sake of brevity and completeness, the corresponding mathematical developments are given in Appendix A and only the corresponding final results are presented here, i.e. (i) using Eq. (16) with no constraints imposed on the weights $\mathbf{w}$, it results from Eq. (A.7) that the BLP is precisely the conditional expectation given by Eq. (8) and this predictor is unbiased by definition, while (ii) using Eq. (17) with $\mathbf{w}$ summing to one corresponds to the BLUP, with weights given by Eq. (A.11), and where this constraint on $\mathbf{w}$ implicitly assumes without necessity that $\mu_{X_i} = \mu_Y \ \forall i$, i.e. that

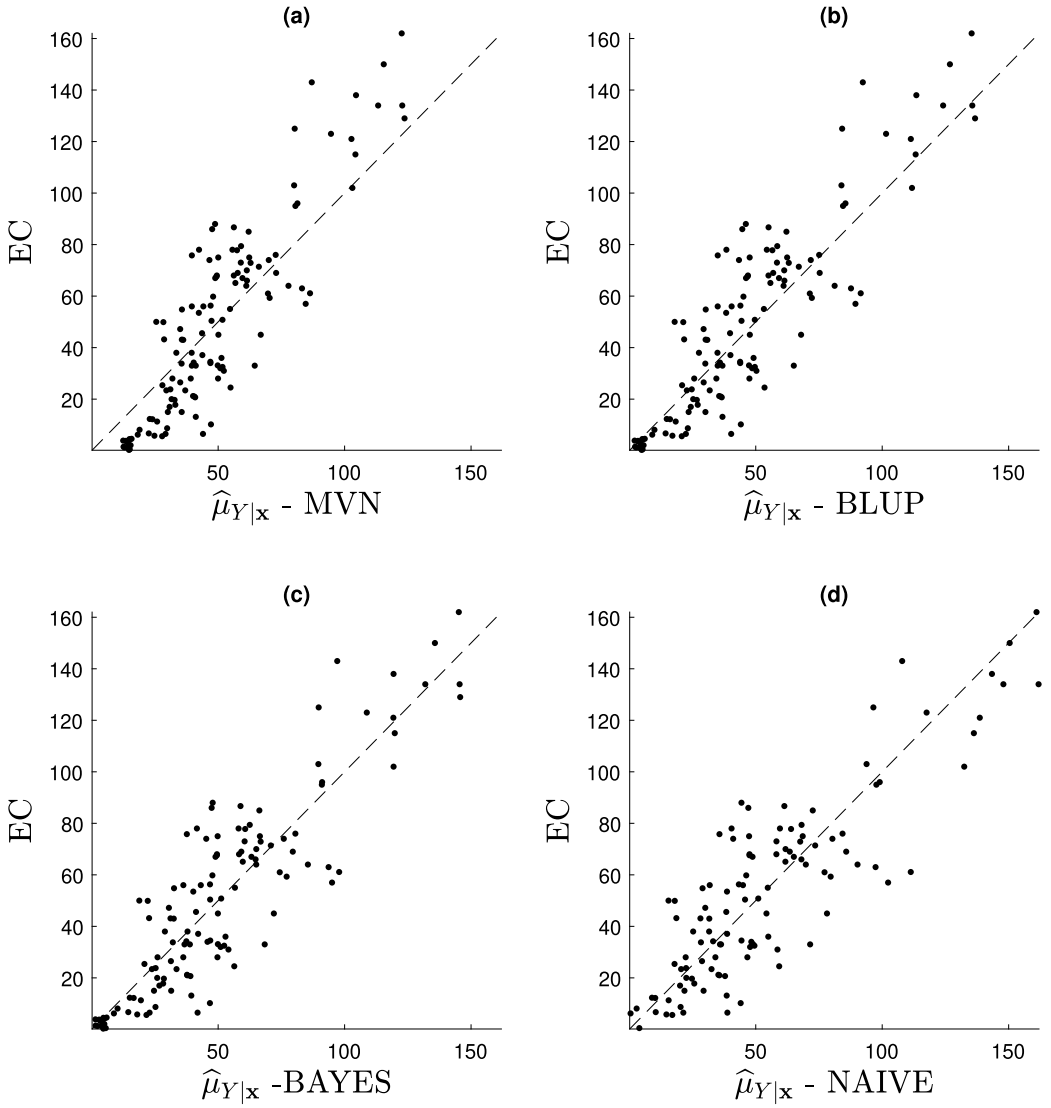


Fig. 6. Scatter plots of measured EC values versus the averaged predicted EC values using (a) multivariate normal approach, (b) best linear unbiased predictor, (c) Bayes approach and (d) Naive Bayes approach. Dashed line is the 1:1 line.

$\mu_{\epsilon_i} = 0$. Indeed, biased predictors are naturally accounted for in Eq. (6), so there is no point here to impose this constraint.

As a summary, as long as one is using a linear averaging of $\mathbf{X}$ as a predictor Y^p of Y that aims at minimizing the prediction variance $Var[Y - Y^p]$, there is no objective reason to depart from Eq. (8), which corresponds to the best linear weighting in the mean square sense. This is correct as long as Eq. (1) holds true, i.e. when the X_i's predictors of Y are possibly biased but when no conditional bias occurs so that $E[X_i|y] = y - \mu_{\epsilon_i}$ $\forall i$ holds true. In the DSM literature, this condition is not routinely assessed by authors, and most of the time only summary performance metrics are presented for the results of the various ML methods that are used, thus preventing the identification of such conditional biases. However, it is expected that they naturally occur. E.g., in a soil salinity framework, Ge et al. (2022) reported graphical results that compare predicted and measured EC using four ML methods, where these predicted values are conditionally biased to various extents. Similarly, for our data set, it is clear from Fig. 2 that conditional biases might occur too. In such circumstances, linearly weighting the predictors without accounting for these conditional biases can lead to disappointing results. We propose hereafter an alternate way of tackling this issue by relying on a Bayesian approach, that does not rely on specific assumptions. Starting from the general results, we propose afterwards specific assumptions that greatly reduce the computational burden and that address this problem.

2.6. Bayes approach

Without any specific assumption, we can write from Bayes theorem that

$$f(y|\mathbf{x}) = \frac{f(\mathbf{x}|y)f(y)}{f(\mathbf{x})} \propto f(\mathbf{x}|y)f(y) \tag{18}$$

where $f(\mathbf{x})$ is a constant value that acts a normalization factor. By comparison with the multivariate normal approach, this formulation has the advantage of allowing arbitrary (and possibly distinct) ways of expressing $f(\mathbf{x}|y)$ and $f(y)$. E.g., the a priori distribution $f(y)$ does not need to be normal, while a joint multivariate normal distribution can still be used for $f(\mathbf{x}|y)$ using Eqs. (10) and (11) and their corresponding parameters estimates in Eq. (12). More complex modeling approaches could also be used for $f(\mathbf{x}|y)$ when normality cannot realistically be assumed, e.g. using copulas (see, e.g., Harry, 2014).

Table 3
Main performance metrics for the best ML algorithm (either RF or XGB; see also Table 2) and for the various averaging approaches. Parts (a), (b), (c) and (d) refer to clay content, CCE, SOC and EC, respectively.

Method	ME	RMSE	R^2	R_L	ME	RMSE	R^2	R_L
		(a)				(b)		
RF	−0.092	7.659	0.615	0.726	−0.071	4.572	0.382	0.437
MVN	−0.000	8.286	0.550	0.636	0.000	4.884	0.295	0.333
BLUP	−0.418	7.497	0.633	0.745	0.090	4.452	0.415	0.488
BAYES	−0.000	7.226	0.658	0.793	−0.000	3.504	0.637	0.778
NAIVE	−0.000	8.236	0.555	0.798	−0.000	4.004	0.526	0.771
		(c)				(d)		
XGB	−0.007	0.365	0.657	0.787	1.478	18.625	0.757	0.862
MVN	−0.000	0.385	0.617	0.701	0.000	18.746	0.752	0.832
BLUP	0.019	0.344	0.695	0.793	2.665	17.455	0.790	0.873
BAYES	−0.000	0.316	0.742	0.852	0.000	17.095	0.794	0.885
NAIVE	−0.000	0.343	0.696	0.854	0.000	18.314	0.763	0.888

Legend : RF = random forest, SVR = support vector regression, MVN = multivariate normal approach, BLUP = best linear unbiased predictor, BAYES = Bayes approach, NAIVE = naive Bayes approach.

Table 4
Conditional correlation matrix of $\mathbf{X}|y$ as estimated from residuals around the regression lines $\mu_{X_i|y}$. Parts (a), (b), (c) and (d) refer to clay content, CCE, SOC and EC, respectively. All estimated correlations greater than 0.174 (in absolute value) are statistically significant ($n = 128$, two-sided test, $\alpha = 0.05$).

	kNN	RF	XGB	CB	SVR	kNN	RF	XGB	CB	SVR
			(a)					(b)		
kNN	1	0.726	0.604	0.599	0.838	1	0.710	0.328	0.315	0.637
RF		1	0.857	0.845	0.816		1	0.449	0.268	0.594
XGB			1	0.819	0.723			1	0.183	0.443
CB				1	0.698				1	0.266
SVR					1					1
			(c)					(d)		
kNN	1	0.576	0.441	0.482	0.600	1	0.761	0.692	0.741	0.727
RF		1	0.417	0.384	0.532		1	0.848	0.895	0.787
XGB			1	0.525	0.613			1	0.814	0.683
CB				1	0.554				1	0.736
SVR					1					1

Legend : kNN = k-nearest neighbor, RF = random forest, XGB = extreme gradient boosting, CB = cubist, SVR = support vector regression.

A noteworthy simplification of Eq. (18) is obtained if one assumes additionally that prediction errors are mutually independent conditionally to y, so that one can write

$$f(\mathbf{x}|y) = \prod_{i=1}^{n} f(x_i|y). \tag{19}$$

Plugging Eq. (19) into Eq. (18) and using again Bayes theorem for each $f(x_i|y)$ so that $f(x_i|y) = f(y|x_i)f(x_i)/f(y)$ then gives the result

$$f(y|\mathbf{x}) \propto f(y) \prod_{i=1}^{n} \frac{f(y|x_i)}{f(y)} \tag{20}$$

where again all $f(x_i)$'s are constant values. This corresponds to the continuous version of the so-called naive Bayes classifier (Sammut and Webb, 2017). The main advantage of Eq. (20) is to allow using arbitrary (and again possibly distinct) way of expressing the various conditional distributions $f(y|x_i)$ for each predictor without requiring any inference about their joint distribution. It can be shown that naive Bayes fusion is precisely the most rational choice when no information is at hand for the joint distribution (Bogaert and Fasbender, 2007). However, for correlated prediction errors, it is also expected that better results are obtained by properly accounting for the joint distribution of these errors.

The Bayes approach is particularly useful when Eq. (1) does not hold true, either because there is a conditional bias so that $E[\varepsilon_i|y]$ depends on y, or because the variability of the prediction residuals depends on y, to the opposite of what is stated in Eq. (2). We will only address here the conditional bias issue, by considering that $\mu_{X_i|y} \neq y - \mu_{\varepsilon i}$ in general, where each regression is a function that needs to be inferred. Without loss of generality, we will consider here a linear regression model approach, with

$$\mu_{X_i|y} = \mathbf{u}_i'(y)\boldsymbol{\beta}_i \quad \forall i \tag{21}$$

where $\mathbf{u}_i'(y) = (u_{i1}(y), \ldots, u_{ip}(y))$ is a vector of functions that depend on y only and $\boldsymbol{\beta}_i$ is vector of p unknown parameters, with $\widehat{\boldsymbol{\beta}}_i$ estimates obtained from ordinary least squares. Assuming in parallel that $\Sigma_{\mathbf{X}|y} = \Sigma_\varepsilon$ (i.e. second-order moments are still invariant with respect to y), the corresponding conditional mean and covariance matrix estimates are easily obtained from a set of K observed values y and corresponding predicted values $x_1, \ldots, x_n$, with

$$\begin{aligned} \widehat{\mu}_{X_i|y_k} &= \mathbf{u}_i'(y_k)\widehat{\boldsymbol{\beta}}_i \\ \widehat{\sigma}_{\varepsilon,ij} &= \frac{1}{K-1}\sum_{k=1}^{K}(x_{ik} - \widehat{\mu}_{X_i|y_k})(x_{jk} - \widehat{\mu}_{X_j|y_k}) \end{aligned} \tag{22}$$

Assuming that Y and $\mathbf{X}|y$ are still normally distributed with $Y \sim N(\mu_Y, \sigma_Y^2)$ and $\mathbf{X}|y \sim N(\boldsymbol{\mu}_{\mathbf{X}|y}, \Sigma_\varepsilon)$, Eq. (18) is then easily computed as a function of y, with

$$\begin{aligned} f(y) &\propto \exp\left(-\frac{1}{2}(y-\mu_Y)^2/\sigma_Y^2\right) \\ f(\mathbf{x}|y) &\propto \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}|y})'\Sigma_\varepsilon^{-1}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}|y})\right) \end{aligned} \tag{23}$$

It is worth emphasizing that the posterior distribution $f(y|\mathbf{x})$ is not normally distributed in general, as it depends on the expression for $\boldsymbol{\mu}_{\mathbf{X}|y}$ that can be arbitrarily chosen. However, when the specific choice of a simple linear regression is made, with

$$\mu_{X_i|y} = \beta_{0,i} + \beta_{1,i}y \quad \forall i = 1, \ldots, n \tag{24}$$

then $f(y|\mathbf{x})$ is indeed normally distributed, with expectation and variance that are given by

$$\mu_{Y|\mathbf{x}} = \frac{\mu_Y + \sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}(\mathbf{x}-\boldsymbol{\beta}_0)}{1+\sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1} \quad ; \quad \sigma_{Y|\mathbf{x}}^2 = \frac{\sigma_Y^2}{1+\sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1} \tag{25}$$

where $\boldsymbol{\beta}_0 = (\beta_{0,1}, \ldots, \beta_{0,n})'$ and $\boldsymbol{\beta}_1 = (\beta_{1,1}, \ldots, \beta_{1,n})'$ are the vector of intercepts and slopes in Eq. (24), respectively (see proof in Appendix B). It can also be shown that a similar conclusion holds true when quadratic terms $\beta_{2,i}y^2$ are included in Eq. (24), but the corresponding proof and resulting formulas are omitted here.

Clearly, Eq. (25) considerably facilitates the task by avoiding using Eq. (18) for numerically computing the whole $f(y|\mathbf{x})$ distribution and subsequently its expectation and variance, as Eq. (25) provides explicit formulas under these assumptions. Similarly to Eq. (8), it is worth emphasizing too that $\mu_{Y|\mathbf{x}}$ in Eq. (25) is a linear weighting scheme of the $\mathbf{x}$ values under the aforementioned assumptions. It includes Eq. (8) as a particular case when there is no conditional bias, i.e. when $\boldsymbol{\beta}_1 = \mathbf{1}$ and $\boldsymbol{\beta}_0 = \mu_\varepsilon$. It is also possible to assess the impact that correlations between prediction errors have on the final results by setting $\Sigma_\varepsilon = diag\{\sigma_{\varepsilon_1}^2, \ldots, \sigma_{\varepsilon_n}^2\}$ in Eq. (25), this corresponding to the naive Bayes fusion that assumes independence of the prediction errors for the various predictors.

3. Results and discussion

For illustrating the previous theoretical results, the data set described in Sections 2.1 and 2.2 is used hereafter. The corresponding numerical results for the various averaging techniques are exhaustively presented in several tables. However, for the sake of brevity, the figures are only presenting the most representative and illustrative results for a selection of the soil properties.

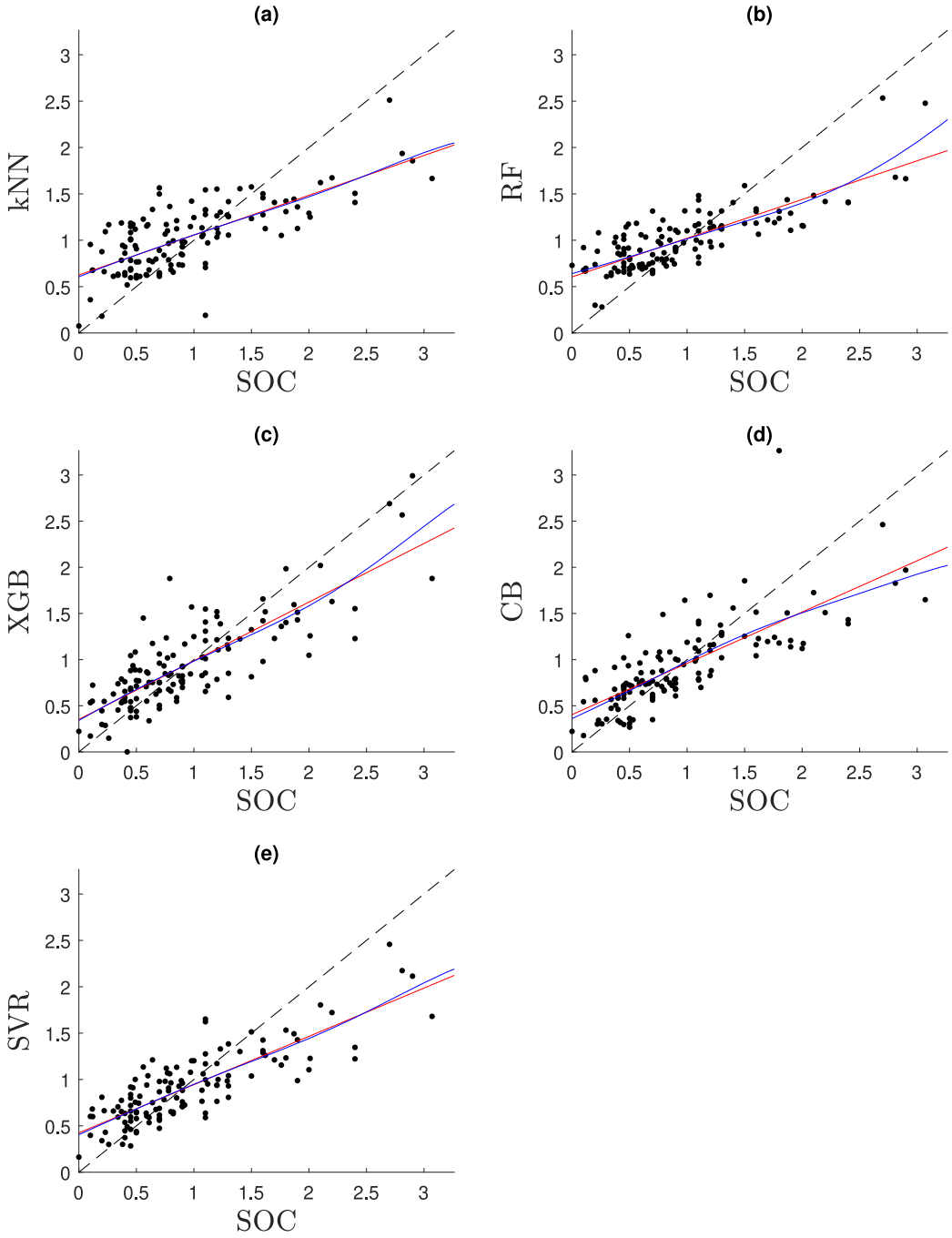


Fig. 7. Scatter plots of the x_i predicted SOC values using the various ML algorithms versus y measured SOC values. Dashed Line is the 1:1 line. Red line is the estimated regression line $\widehat{\mu}_{X_i|y} = \widehat{\beta}_{0,i} + \widehat{\beta}_{1,i} y$, while blue line is the nonparametric loess regression estimate.

The multivariate normal (MVN) approach of Section 2.4 is making use of Eq. (8), where the expectations μ_Y, μ_ϵ and variance/covariances σ_Y^2, Σ_ϵ are estimated from Eq. (12). From the previously given theoretical results, MVN is the best linear predictor and is thus expected to outperform other linear averaging methods when no conditional bias occurs. The corresponding $\widehat{\mu}_{Y|\mathbf{x}}$ estimated values are presented in Figs. 5a and 6a for SOC and EC only, while associated performance metrics are given in Table 3 for all soil properties. It can be seen that MVN does not improve the R^2 metrics and may even lead to lower performances when compared to the single use of the best ML algorithm (either RF or XGB). Table 3 emphasizes that MVN is unbiased by design, with a mean error (ME) equal to 0, but it remains conditionally biased as seen from Figs. 5a and Fig. 6a. Using BLUP that enforces weights summing to one might seem to improve R^2 and R_L, but it does not necessarily solve the conditional bias issue, where the offset from the 1:1 line is clear from Figs. 5b and 6b. These results do not come as a surprise. When MVN is properly accounting for the fact that predictors are possibly biased, BLUP is assuming without reason that this is not the case. Moreover, the underlying hypotheses stated in Eq. (2) are clearly not fulfilled here. As an illustration for SOC, Fig. 7 shows that the hypothesis $E[X_i|y] = y - \mu_{\epsilon_i}$ does not hold true, so that the various predictors are conditionally biased, showing to various extents an overestimation (an underestimation) of SOC when SOC values are low (are high). By assuming that $E[X_i|y]$ can reasonably be modeled

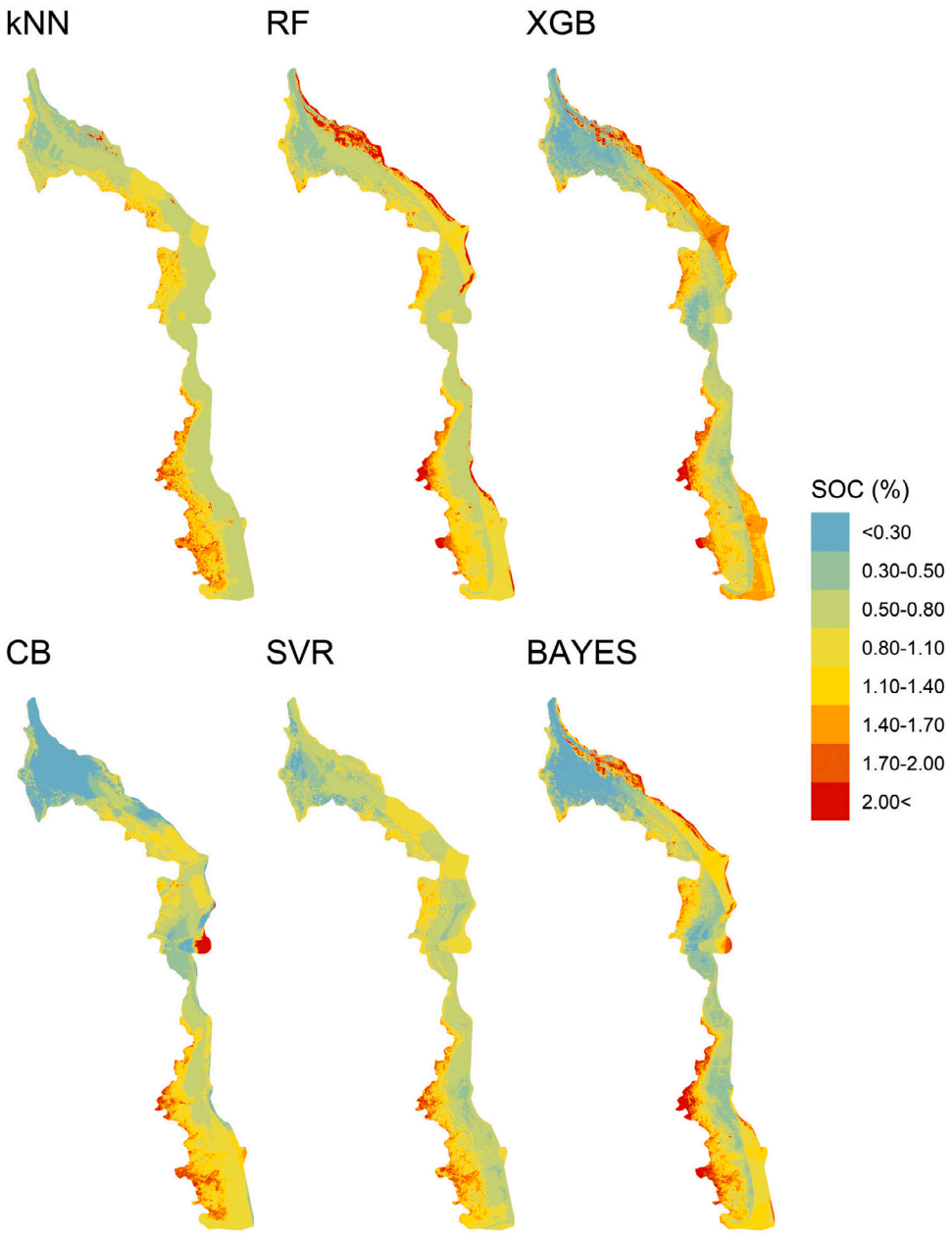


Fig. 8. Maps of the predicted SOC values using the various ML algorithms (kNN, RF, XGB, CB and SVR) that have been averaged afterwards using the Bayesian approach (BAYES).

using Eq. (24), the parameters $\boldsymbol{\beta}_0$ and $\boldsymbol{\beta}_1$ are easily estimated from ordinary least squares. The corresponding estimated regression lines $\widehat{\mu}_{X_i|y} = \widehat{\beta}_{0,i} + \widehat{\beta}_{1,i} y$ are presented in Fig. 7. For the sake of comparison, the nonparametric loess regression estimates (i.e., a locally weighted regression that fits a smooth curve through the points in the scatter plot) of the conditional expectations $\widehat{\mu}_{X_i|y}$ are also displayed in Fig. 7. It can be seen that these loess estimates compare well to the assumption of a simple regression line, even if some discrepancies can be observed at the extremities of the scatter plots. The same general findings hold true for the other soil properties.

For the Bayes approach (BAYES) of Section 2.6, it is assumed here that Y and $\mathbf{X}|y$ are normally distributed with $Y \sim N(\mu_Y, \sigma_Y^2)$ and $\mathbf{X}|y \sim N(\boldsymbol{\mu}_{\mathbf{X}|y}, \Sigma_\epsilon)$, so that Eq. (25) can be used, where the estimate for Σ_ϵ is given by Eq. (22). The corresponding $\widehat{\mu}_{Y|\mathbf{x}}$ estimated values are presented in Fig. 5c and 6c for SOC and EC only, while associated performance metrics are given in Table 3 for all soil properties. As for MVN, it is clear that ME is equal to 0 by design. From Fig. 5c and Fig. 6c, one can see that $\widehat{\mu}_{Y|\mathbf{x}}$ are now aligned along the 1:1 line, while Table 3 shows that BAYES outperforms BLUP. Clearly, BAYES yields conditionally unbiased prediction at almost no additional computing and inference costs. In agreement with Section 2.4.2, it can also be seen that the benefit of averaging the five ML predictions decreases as the performances for these separate predictions increase. The lowest increase for R^2 is observed for EC, while the highest increase is observed for CCE.

From an examination of the estimated correlations between predictors residuals

$$\widehat{\rho}_{\epsilon,ij} = \frac{\widehat{\sigma}_{\epsilon,ij}}{\sqrt{\widehat{\sigma}_{\epsilon,ii}\widehat{\sigma}_{\epsilon,jj}}} \tag{26}$$

with variances and covariances estimated from Eq. (22), one can see in Table 4 that they largely depend on the soil properties. As these

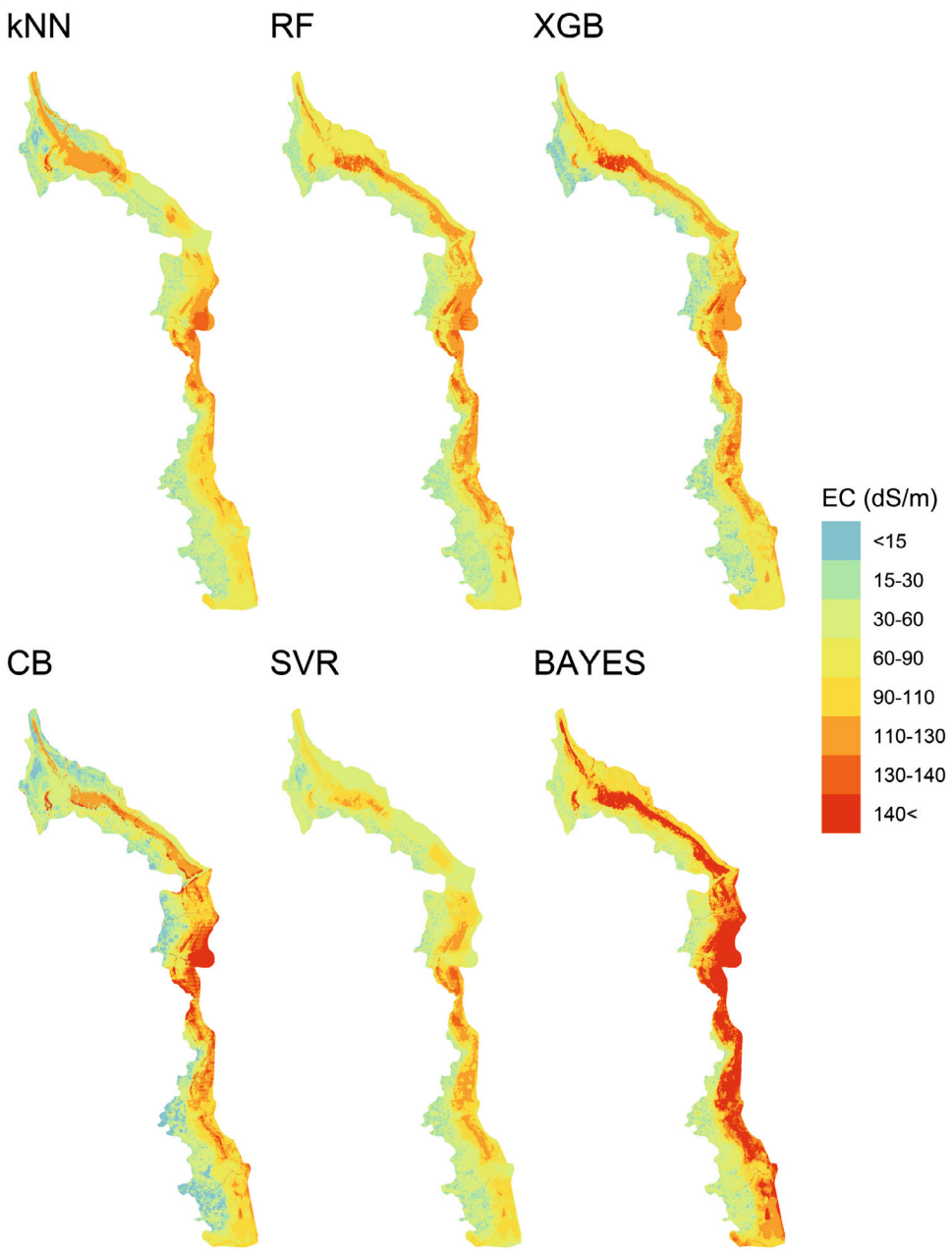


Fig. 9. Maps of the predicted EC values using the various ML algorithms (kNN, RF, XGB, CB and SVR) that have been averaged afterwards using the Bayesian approach (BAYES).

correlations are all statistically significant, a naive Bayes approach (NAIVE) would be theoretically less efficient by wrongly assuming that $\Sigma_\epsilon = diag\{\sigma^2_{\epsilon_1}, \ldots, \sigma^2_{\epsilon_n}\}$. The corresponding $\hat{\mu}_{Y|\mathbf{x}}$ estimated values for NAIVE and BAYES are presented in Fig. 5 and 6 for SOC and EC only, while the performance metrics are given in Table 3 for all soil properties. With respect to R^2, these metrics confirm that NAIVE results are inferior to those for BAYES, as expected, though the difference are not dramatic here. Prior to any analysis, no general conclusions can be drawn about the fact that NAIVE is or is not a reasonable simplification of BAYES in general. However, when it is the case, it opens the possibility of using directly Eq. (20) without assuming normal distributions for $f(y)$ and for the various $f(y|x_i)$'s. This can turn out to be an advantage when normality assumption does not hold reasonably well, even after transformation. However, for our specific data set, this does not offer any clear advantage.

As the values of the environmental covariates are at hand at any arbitrary location over the study area, the various ML algorithms can be used to predict and map the values of clay content, CCE, SOC and EC over a regular grid. Figs. 8 and 9 present the predicted maps using the five ML algorithms for SOC and EC only. Although these ML algorithms yield correlated predictions in general (as seen from Figs. 2 and 3), this might nevertheless leads to distinct spatial patterns on the maps. Selecting a single map among those based on the fact that it corresponds to the ML algorithm with the best global performance (where RF is the best one for clay content, CCE and EC, while XGB is the best one for SOC) is quite arguable, especially when global performances are close to each other. In these situations, averaging the predictions allows the user to combine at best these maps, with patterns that are borrowed from the various maps (as seen from the averaged maps using the Bayes approach on Figs. 8 and 9), and with improved prediction performances by comparison with the selection of a single ML algorithm (as seen from Table 3). Clearly, these differences in spatial patterns for the five ML algorithms are higher for SOC than for EC, as the predictions performances for SOC are lower than for

EC. It is worth noting too that the range of the predicted EC values using the BAYES approach is now extended by comparison with the five ML predictions, with a higher proportion of high EC values. This is the consequence of the correction for the conditional bias, along with the fact that high EC values (EC > 130 dS/m) are more represented in the ML predicted maps than in the predicted values from the training sample.

4. Conclusions

A wide panel of ML algorithms are now implemented in various computer packages and are widely accessible to the soil science community. Beyond the acknowledged performances of ML algorithms with respect to prediction accuracy in general, this accessibility can largely explain their popularity for digital soil mapping. As a logical consequence, this leads to a large amount of papers that aim at comparing the respective benefits of various ML algorithms that are applied on the same data set. However, no firm conclusions can be drawn about the best algorithm to be used in general. Accordingly, the idea of combining them seems quite natural and proved to be beneficial too. Most commonly, linearly weighting schemes are proposed. However, when used out-of-the-box and without proper care, they can also lead to puzzling or even disappointing results. In this paper, it was first shown that the maximum reachable benefit from a linear weighting can be assessed and only requires inference about the expectations and covariance matrix of the prediction errors. It was also reminded that the conditional expectation of this multivariate normal distribution is precisely the best linear predictor in the mean square sense. Accordingly, as long as the goal is to minimize the prediction variance, the reasons for selecting other weighting schemes are unclear. The fact that ML predictions are expected to be conditionally biased too is rarely assessed. We have shown that a general Bayesian framework is able to handle these issues and its benefits were illustrated using several soil properties that were predicted from various ML algorithms. We believe that this paper can be helpful for the DSM community in general, by emphasizing the importance of a proper formulation of the problem if one aims at maximizing the benefit of averaging predictions.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Appendix A. Best linear (unbiased) predictor

Let us consider first a predictor as given by Eq. (16), with

$$Y^p - \mu_Y = \mathbf{w}'(\mathbf{X} - \mu_{\mathbf{X}}) \quad \Longleftrightarrow \quad Y^p = \mu_Y + \mathbf{w}'(\mathbf{X} - \mu_{\mathbf{X}}) \tag{A.1}$$

where the corresponding expected prediction error and prediction variance are given by

$$E[Y - Y^p] = E[Y] - E[\mu_Y] - \mathbf{w}' \overbrace{E[(\mathbf{X} - \mu_{\mathbf{X}})]}^{=0} = \mu_Y - \mu_Y = 0 \tag{A.2}$$

$$Var[Y - Y^p] = Var[Y - \mathbf{w}'\mathbf{X}] = \sigma_Y^2 - 2\mathbf{w}'\boldsymbol{\sigma}_{Y,\mathbf{X}} + \mathbf{w}'\Sigma_{\mathbf{X}}\mathbf{w} \tag{A.3}$$

where $\boldsymbol{\sigma}_{Y,\mathbf{X}}$ is the covariance vector between Y and $\mathbf{X}$. From Eq. (A.2), it is clear that Y^p is unbiased whatever the choice for $\mathbf{w}$. Minimizing $Var[Y - Y^p]$ with respect to $\mathbf{w}$ is obtained by setting the corresponding derivatives equal to 0, i.e.

$$\frac{\partial}{\partial \mathbf{w}} Var[Y - Y^p] = -2\boldsymbol{\sigma}_{Y,\mathbf{X}} + 2\Sigma_{\mathbf{X}}\mathbf{w} = \mathbf{0} \quad \Longleftrightarrow \quad \mathbf{w} = \Sigma_{\mathbf{X}}^{-1}\boldsymbol{\sigma}_{Y,\mathbf{X}} \tag{A.4}$$

and this solution is unique as $Var[Y - Y^p]$ is convex everywhere with respect to $\mathbf{w}$, with

$$\frac{\partial^2}{\partial \mathbf{w}' \partial \mathbf{w}} Var[Y - Y^p] = 2\Sigma_{\mathbf{X}} \tag{A.5}$$

where $\det(2\Sigma_{\mathbf{X}}) > 0$ as $\Sigma_{\mathbf{X}}$ is a positive definite matrix. As from Eq. (6) we have $\boldsymbol{\sigma}_{Y,\mathbf{X}} = \sigma_Y^2\mathbf{1}$ and $\Sigma_{\mathbf{X}} = \sigma_Y^2\mathbf{1}\mathbf{1}' + \Sigma_\varepsilon$, substituting this in Eq. (A.4) yields

$$\mathbf{w}' = \sigma_Y^2\mathbf{1}'(\sigma_Y^2\mathbf{1}\mathbf{1}' + \Sigma_\varepsilon)^{-1} \tag{A.6}$$

As $\mu_{\mathbf{X}} = \mu_Y\mathbf{1} - \boldsymbol{\mu}_\varepsilon$ from Eq. (6) too, substituting this result along with Eq. (A.6) in Eq. (A.1) yields

$$Y^p = \mu_Y + \sigma_Y^2\mathbf{1}'(\sigma_Y^2\mathbf{1}\mathbf{1}' + \Sigma_\varepsilon)^{-1}(\mathbf{X} - \mu_Y\mathbf{1} + \boldsymbol{\mu}_\varepsilon) \tag{A.7}$$

where the right-hand sides of Eqs. (8) and (A.7) are the same, thus showing that $y^p = \mu_{Y|\mathbf{x}}$. Stated in other words, the best linear predictor based on Eq. (A.1) corresponds to the conditional expectation of the multivariate normal distribution. It is worth noting that the weights $\mathbf{w}$ obtained this way do not sum up to one in general, i.e. $\mathbf{1}'\mathbf{w} \neq 1$.

Though there is no obvious necessity to do so here, a commonly used formulation is based on Eq. (17) where $Y^p = \mathbf{w}'\mathbf{X}$, with the constraint that $\mathbf{1}'\mathbf{w} = 1$. This constraint comes from the hypothesis that $\mu_{X_i} = \mu_Y \ \forall i$, or in matrix notation $\mu_{\mathbf{X}} = \mu_Y\mathbf{1}$, so that fulfilling the unbiasedness condition

$$E[Y - Y^p] = E[Y] - \mathbf{w}'E[\mathbf{X}] = \mu_Y - \mu_Y\mathbf{w}'\mathbf{1} = \mu_Y(1 - \mathbf{w}'\mathbf{1}) = 0 \tag{A.8}$$

requires that $\mathbf{w}'\mathbf{1} = 1 \Longleftrightarrow \mathbf{1}'\mathbf{w} = 1$, a classical condition associated with the definition of the BLUP. Minimizing $Var[Y - Y^p]$ with respect to $\mathbf{w}$ under this constraint is easily done using the Lagrangian formalism, where one needs now to minimize the objective function $O(\mathbf{w}, \eta)$ given by

$$O(\mathbf{w}, \eta) = Var[Y - Y^p] + 2\eta(\mathbf{1}'\mathbf{w} - 1) \tag{A.9}$$

where the prediction variance $Var[Y - Y^p]$ is still given by Eq. (A.3) and where η is the Lagrange parameter associated with the constraint. Setting all derivatives with respect to $\mathbf{w}$ and η equal to 0 yields

$$\begin{cases} \dfrac{\partial}{\partial \mathbf{w}} O(\mathbf{w}, \eta) = -2\boldsymbol{\sigma}_{Y,\mathbf{X}} + 2\Sigma_{\mathbf{X}}\mathbf{w} + 2\eta\mathbf{1} = \mathbf{0} \\ \dfrac{\partial}{\partial \eta} O(\mathbf{w}, \eta) = 2(\mathbf{1}'\mathbf{w} - 1) = 0 \end{cases} \tag{A.10}$$

or equivalently, after simplification and in matrix notation,

$$\begin{pmatrix} \Sigma_{\mathbf{X}} & \mathbf{1} \\ \mathbf{1}' & 0 \end{pmatrix} \begin{pmatrix} \mathbf{w} \\ \eta \end{pmatrix} = \begin{pmatrix} \boldsymbol{\sigma}_{Y,\mathbf{X}} \\ 1 \end{pmatrix} \quad \Longleftrightarrow \quad \begin{pmatrix} \mathbf{w} \\ \eta \end{pmatrix} = \begin{pmatrix} \Sigma_{\mathbf{X}} & \mathbf{1} \\ \mathbf{1}' & 0 \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{\sigma}_{Y,\mathbf{X}} \\ 1 \end{pmatrix} \tag{A.11}$$

where the matrix inverse in eq (A.11) can numerically be computed and for which an explicit expression is also at hand using standard properties for the inverse of block matrices (see, e.g., Eves, 1980).

Appendix B. Results for the Bayes approach

Proving Eq. (25) is easily done by identification. Let us remember that the probability distribution function $f(z)$ of a normally distributed random variable $Z \sim N(\mu, \sigma^2)$ is given by

$$f(z) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}(z - \mu)^2/\sigma^2\right) \tag{B.1}$$

Taking the logarithm and developing the square yields

$$\ln f(z) = \left(-\frac{1}{2\sigma^2}\right)z^2 + \left(\frac{\mu}{\sigma^2}\right)z + \left(-\frac{\mu^2}{2\sigma^2} - \sigma\sqrt{2\pi}\right) \tag{B.2}$$

which is a second-degree polynomial with respect to z, so that μ and σ^2 can be identified from the coefficients for the z^2 and z terms of such a polynomial. Using now Eq. (18) yields

$$\ln f(y|\mathbf{x}) = \ln f(\mathbf{x}|y) + \ln f(y) + cste \tag{B.3}$$

where from Eq. (23) we have

$$\begin{cases} \ln f(y) = \left(-\frac{1}{2\sigma_Y^2}\right)y^2 + \left(\frac{\mu_Y}{\sigma_Y^2}\right)y + cste \\ \ln f(\mathbf{x}|y) = -\frac{1}{2}\left(\mathbf{x}'\Sigma_\varepsilon^{-1}\mathbf{x} - 2\mathbf{x}'\Sigma_\varepsilon^{-1}\boldsymbol{\mu}_{\mathbf{X}|y} + \boldsymbol{\mu}'_{\mathbf{X}|y}\Sigma_\varepsilon^{-1}\boldsymbol{\mu}_{\mathbf{X}|y}\right) + cste \end{cases} \tag{B.4}$$

Using Eq. (24), the conditional expectation vector $\boldsymbol{\mu}_{\mathbf{X}|y}$ is given by

$$\boldsymbol{\mu}_{\mathbf{X}|y} = \boldsymbol{\beta}_0 + \boldsymbol{\beta}_1 y \tag{B.5}$$

where $\boldsymbol{\beta}_0 = (\beta_{0,1}, \ldots, \beta_{0,n})'$ and $\boldsymbol{\beta}_1 = (\beta_{1,1}, \ldots, \beta_{1,n})'$. We thus have

$$\begin{aligned} \mathbf{x}'\Sigma_\varepsilon^{-1}\boldsymbol{\mu}_{\mathbf{X}|y} &= (\mathbf{x}'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1)y + \mathbf{x}'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_0 \\ \boldsymbol{\mu}'_{\mathbf{X}|y}\Sigma_\varepsilon^{-1}\boldsymbol{\mu}_{\mathbf{X}|y} &= (\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1)y^2 + (2\boldsymbol{\beta}_0'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1)y + \boldsymbol{\beta}_0'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_0 \end{aligned} \tag{B.6}$$

Substituting Eq. (B.6) into Eq. (B.4) and collecting terms for y and y^2 in Eq. (B.3) yields

$$\ln f(y|\mathbf{x}) = -\frac{1}{2}\left(\frac{1}{\sigma_Y^2} + \boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1\right)y^2 + \left(\frac{\mu_Y}{\sigma_Y^2} + \mathbf{x}'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1 - \boldsymbol{\beta}_0'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1\right)y + cste \tag{B.7}$$

that corresponds to the expression of a normal distribution with respect to y. By identification from the coefficients for the y^2 and y terms, we obtain

$$\begin{cases} \frac{1}{\sigma_{Y|\mathbf{x}}^2} = \frac{1}{\sigma_Y^2} + \boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1 = \frac{1 + \sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1}{\sigma_Y^2} \\ \frac{\mu_{Y|\mathbf{x}}}{\sigma_{Y|\mathbf{x}}^2} = \frac{\mu_Y}{\sigma_Y^2} + \mathbf{x}'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1 - \boldsymbol{\beta}_0'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1 = \frac{\mu_Y + \sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}(\mathbf{x} - \boldsymbol{\beta}_0)}{\sigma_Y^2} \end{cases} \tag{B.8}$$

Solving for $\sigma_{Y|\mathbf{x}}^2$ and $\mu_{Y|\mathbf{x}}$ thus finally gives

$$\begin{cases} \sigma_{Y|\mathbf{x}}^2 = \frac{\sigma_Y^2}{1 + \sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1} \\ \mu_{Y|\mathbf{x}} = \frac{\mu_Y + \sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}(\mathbf{x} - \boldsymbol{\beta}_0)}{1 + \sigma_Y^2\boldsymbol{\beta}_1'\Sigma_\varepsilon^{-1}\boldsymbol{\beta}_1} \end{cases} \tag{B.9}$$

Appendix C. Brief description of machine learners

The k-nearest neighbor (kNN) algorithm was originally proposed by Nemes et al. (2006). It is one of the simplest ML algorithms commonly used for regression and classification problems. It predicts the value of the target variable by considering the mean of the k-nearest neighboring samples in the feature space (Aalen, 1992) using an appropriate distance metric (Kuhn and Johnson, 2013). The hyperparameter to be optimized is the value of k.

The random forest (RF) algorithm is one of the most popular ML algorithms. It is an ensemble technique that leverages predictions from multiple randomized regression trees, where the results for the individual trees are aggregated to generate the final prediction (Breiman, 2001). The hyperparameters to be optimized are the number of trees in the ensemble, the minimum node size that affects trees depth and complexity, and the number of input features (mtry) to be randomly selected when splitting at each node.

The extreme gradient boosting (XGB) algorithm is an ensemble tree technique related to the gradient boosting framework (Chen and Guestrin, 2016). It builds an ensemble of strong learners from weak learners and iteratively builds multiple decision trees based on the prediction errors of the previous tree model. This iterative process enables the ensemble to learn from its mistakes for improving prediction performances. The hyperparameters to be optimized are the number of boosting rounds (nrounds), the learning rate (η), the maximum depth of each tree (max.depth), and a parameter λ that controls the regularization to prevent overfitting.

Table C.1
Optimized values of the hyperparameters for the various machine learners and soil properties.

Learner	Hyperparameter	Clay	CCE	SOC	EC
kNN	k	6	35	30	3
RF	num.trees	303	764	905	137
	min.node.size	15	11	5	7
	mtry	23	21	25	16
XGB	nrounds	211	351	344	478
	max.depth	3	2	3	2
	η	0.17	0.10	0.17	0.38
	λ	0.11	0.21	0.14	0.22
CB	committees	71	57	76	30
	rules	43	12	32	12
	extrapolation	7	10	3	8
	neighbors	8	2	4	5
SVR	kernel	radial	radial	radial	radial
	c	6	6	3	3
	σ	0.71	0.32	0.66	0.15

Legend : kNN = k-nearest neighbor, RF = random forest, XGB = extreme gradient boosting, CB = cubist, SVR = support vector regression, CCE = calcium carbonate equivalent, SOC = soil organic carbon, EC = electrical conductivity.

The Cubist (CB) algorithm is an extension of Quinlan's M5 model tree (Quinlan, 1992). It is a rule-based model that is able to capture both linear and nonlinear relationships between the target and explanatory variables. The algorithm involves creating several rule-based models known as committees on which the final prediction model is based (John et al., 2021). The hyperparameters to be optimized are the number of committees, the number of rules to be included, the number of neighbors considered for splitting and an extrapolation parameter that controls the extent to which predictions can fall outside the range of values in the training data.

The support vector regression (SVR) algorithm was initially introduced by Vapnik (1963) to handle regression problems. It uses nonlinear transformation techniques such as kernel functions to map the original input space into a higher-dimensional feature space and it performs a linear regression estimation within this space (Noble, 2006). The hyperparameters to be optimized are the choice of the kernel function and its associated parameters, namely c (a penalty parameter) and σ (the kernel width).

For all machine learners used in this paper, the modeling was done using the mlr3 package (Lang et al., 2019) in R version (R. Core Team, 2023) with the assistance of RStudio (RStudio Team, 2023). Table C.1 specifies the optimized hyperparameters that were used for each machine learner and for each soil property.

References

Aalen, O.O., 1992. Modelling Heterogeneity in Survival Analysis by the Compound Poisson Distribution. Ann. Appl. Probab. 2 (4), 951–972.

Bates, J., Granger, C., 1969. The combination of forecasts. J. Oper. Res. Soc. 20, 451–468.

Behrens, T., Schmidt, K., Viscarra Rossel, R.A., Gries, P., Scholten, T., MacMillan, R.A., 2018. Spatial modelling with Euclidean distance fields and machine learning. Eur. J. Soil Sci. 69 (5), 757–770.

Bogaert, P., Fasbender, D., 2007. Bayesian data fusion in a spatial prediction context: A general formulation. Stoch. Environ. Res. Risk Assess. 21, 695–709.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Buckland, S., Burnham, K., Augustin, N., 1997. Model selection: An integral part of inference. Biometrics 53, 603–618.

Buckland, S., Burnham, K., Augustin, N., 2002. Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach, second Ed. Springer, New York.

Caubet, M., Román Dobarco, M., Arrouays, D., Minasny, B., Saby, N.P., 2019. Merging country, continental and global predictions of soil texture: Lessons from ensemble modelling in France. Geoderma 337, 99–110.

Chen, S., Arrouays, D., Leatitia Mulder, V., Poggio, L., Minasny, B., Roudier, P., Libohova, Z., Lagacherie, P., Shi, Z., Hannam, J., Meersmans, J., de Forges, A.C.R., Walter, C., 2022. Digital mapping of GlobalSoilMap soil properties at a broad scale: A review. Geoderma 409, 115567.

Chen, T., Guestrin, C., 2016. XGBoost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 785–794.

Chen, S., Mulder, V.L., Heuvelink, G.B., Poggio, L., Caubet, M., Román Dobarco, M., Walter, C., Arrouays, D., 2020. Model averaging for mapping topsoil organic carbon in France. Geoderma 366, 114237.

Diks, C.G., Vrugt, J.A., 2010. Comparison of point forecast accuracy of model averaging methods in hydrologic applications. Stoch. Environ. Res. Risk Assess. 24, 809–820.

Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P., Meygret, A., Spoto, F., Sy, O., Marchese, F., Bargellini, P., 2012. Sentinel-2: ESA's optical high-resolution mission for GMES operational services. Remote Sens. Environ. 120, 25–36.

Eves, H., 1980. Elementary Matrix Theory, Reprint Ed. Dover, New York.

Ge, X., Ding, J., Teng, D., Wang, J., Huo, T., Jin, X., Wang, J., He, B., Han, L., 2022. Updated soil salinity with fine spatial resolution and high accuracy: The synergy of sentinel-2 MSI, environmental covariates and hybrid machine learning approaches. Catena 212, 106054.

Gee, G.W., Or, D., 2002. Particle size analysis. In: Methods of Soil Analysis, Part 4: Physical Methods. Soils Science Society of America, pp. 255–293.

Geological Survey of Iran, 1995. Geological Quadrangle Map. Geology Organization of Iran, Tehran.

Granger, C.W., Ramanathan, R., 1984. Improved methods of combining forecasts. J. Forecast. 3 (2), 197–204.

Harry, J., 2014. Dependence Modeling with Copulas. In: CRC Monographs on Statistics and Applied Probability, Chapman and Hall, New York.

Heung, B., Ho, H.C., Zhang, J., Knudby, A., Bulmer, C.E., Schmidt, M.G., 2016. An overview and comparison of machine-learning techniques for classification purposes in digital soil mapping. Geoderma 265, 62–77.

Hoeting, J.A., Madigan, D., Raftery, A.E., Volinsky, C.T., 1999. Bayesian model averaging: A tutorial. Statist. Sci. 14 (4), 382–401.

Iranian Ministry of Energy, 2014. The National Water Master Plan Study in the Aras, Sefidrood, Between Sefidrood and Haraz, Atrac and Urmia. Agricultural Water Use Study in Urmia, Lake Basin, Tehran.

John, K., Kebonye, N., Agyeman, P., Ahado, S., 2021. Comparison of Cubist models for soil organic carbon prediction via portable XRF measured data. Environ. Monit. Assess. 193, 1–15.

Keskin, H., Grunwald, S., Harris, W.G., 2019. Digital mapping of soil carbon fractions with machine learning. Geoderma 339, 40–58.

Kuhn, M., Johnson, K., 2013. Applied Predictive Modeling. Springer, New York.

Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., Au, Q., Casalicchio, G., Kotthoff, L., Bischl, B., 2019. mlr3: A modern object-oriented machine learning framework in R. J. Open Source Softw..

Lin, L.I.-K., 1989. A concordance correlation coefficient to evaluate reproducibility. Biometrics 45 (1), 255–268.

Malone, B.P., Minasny, B., Odgers, N.P., McBratney, A.B., 2014. Using model averaging to combine soil property rasters from legacy soil maps and from point data. Geoderma 232–234, 34–44.

Mishra, U., Gautam, S., Riley, W.J., Hoffman, F.M., 2020. Ensemble machine learning approach improves predicted spatial variation of surface soil organic carbon stocks in data-limited Northern Circumpolar Region. Front. Big Data 3.

Møller, A.B., Beucher, A.M., Pouladi, N., Greve, M.H., 2020. Oblique geographic coordinates as covariates for digital soil mapping. SOIL 6 (2), 269–289.

Nelson, D.W., Sommers, L.E., 1996. Total carbon, organic carbon, and organic matter. In: Methods of Soil Analysis, Part 3: Chemical Methods. John Wiley & Sons, Ltd, pp. 961–1010.

Nemes, A., Rawls, W., Pachepsky, Y., Van Genuchten, M., 2006. Sensitivity analysis of the nonparametric nearest neighbor technique to estimate soil water retention. Vadose Zone J. 5, 1222–1235.

Nguyen, K.A., Chen, W., Lin, B.-S., Seeboonruang, U., 2021. Comparison of ensemble machine learning methods for soil Erosion pin measurements. ISPRS Int. J. Geo-Inf. 10 (1).

Nketia, K., Asabere, S., Ramcharan, A., Herbold, S., Erasmi, S., Sauer, D., 2022. Spatio-temporal mapping of soil water storage in a semi-arid landscape of northern Ghana – a multi-tasked ensemble machine-learning approach. Geoderma 410, 115691.

Noble, W., 2006. What is a support vector machine? Nature Biotechnol. 24 (12), 1565–1567.

O'Loughlin, F., Paiva, R., Durand, M., Alsdorf, D., Bates, P., 2016. A multi-sensor approach towards a global vegetation corrected SRTM DEM product. Remote Sens. Environ. 182, 49–59.

O'Rourke, S., Stockmann, U., Holden, N., McBratney, A., Minasny, B., 2016. An assessment of model averaging to improve predictive power of portable vis-NIR and XRF for the determination of agronomic soil properties. Geoderma 279, 31–44.

Page, A.L., Miller, R.H., Keeney, D.R., 1982. Total carbon, organic carbon, and organic matter. In: Methods of Soil Analysis, Part 2: Chemical and Microbiological Properties. American Society of Agronomy.

Papoulis, A., Pillai, S.U., 2002. Probability, Random Variables and Stochastic Processes, fourth Ed. McGraw-Hill Europe, London.

Passy, P., Théry, S., 2018. The use of SAGA GIS modules in QGIS. In: QGIS and Generic Tools. John Wiley & Sons, Ltd, pp. 107–149.

Quinlan, J.R., 1992. Learning with continuous classes. In: 5th Australian Joint Conference on Artificial Intelligence, Vol. 92. pp. 343–348.

R. Core Team, 2023. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, URL https://www.R-project.org/.

Rhoades, J.D., 1996. Salinity: Electrical conductivity and total dissolved solids. In: Methods of Soil Analysis. John Wiley & Sons Ltd, pp. 417–435.

Román Dobarco, M., Arrouays, D., Lagacherie, P., Ciampalini, R., Saby, N.P., 2017. Prediction of topsoil texture for region centre (France) applying model ensemble methods. Geoderma 298, 67–77.

Rostaminia, M., Rahmani, A., Mousavi, S.R., Taghizadeh-Mehrjardi, R., Maghsodi, Z., 2021. Spatial prediction of soil organic carbon stocks in an arid rangeland using machine learning algorithms. Environ. Monit. Assess. 193.

Roy, D., Wulder, M., Loveland, T., C.E., W., Allen, R., Anderson, M., Helder, D., Irons, J., Johnson, D., Kennedy, R., Scambos, T., Schaaf, C., Schott, J., Sheng, Y., Vermote, E., Belward, A., Bindschadler, R., Cohen, W., Gao, F., Hipple, J., Hostert, P., Huntington, J., Justice, C., Kilic, A., Kovalskyy, V., Lee, Z., Lymburner, L., Masek, J., McCorkel, J., Shuai, Y., Trezza, R., Vogelmann, J., Wynne, R., Zhu, Z., 2014. Landsat-8: Science and product vision for terrestrial global change research. Remote Sens. Environ. 145, 154–172.

RStudio Team, 2023. RStudio: Integrated Development Environment for R. RStudio, Inc., Boston, MA, URL http://www.rstudio.com/.

Sammut, C., Webb, G.I. (Eds.), 2017. Encyclopedia of Machine Learning and Data Mining. In: Springer Reference, Springer, New York.

Taghizadeh-Mehrjardi, R., Hamzehpour, N., Hassanzadeh, M., Heung, B., Ghebleh Goydaragh, M., Schmidt, K., Scholten, T., 2021. Enhancing the accuracy of machine learning models using the super learner technique in digital soil mapping. Geoderma 399, 115108.

Taghizadeh-Mehrjardi, R., Khademi, H., Khayamim, F., Zeraatpisheh, M., Heung, B., Scholten, T., 2022. A comparison of model averaging techniques to predict the spatial distribution of soil properties. Remote Sens. 14 (3).

Tajik, S., Ayoubi, S., Zeraatpisheh, M., 2020. Digital mapping of soil organic carbon using ensemble learning model in mollisols of Hyrcanian forests, northern Iran. Geoderma Reg. 20, e00256.

Tong, Y., 2012. The Multivariate Normal Distribution. In: Springer Series in Statistics, Springer, New York.

Vapnik, V.N., 1963. Pattern recognition using generalized portrait method. Autom. Remote Control 24, 774–780.

Vermeulen, D., Van Niekerk, A., 2017. Machine learning performance for predicting soil salinity using different combinations of geomorphometric covariates. Geoderma 299, 1–12.

Wadoux, A.M.-C., Minasny, B., McBratney, A.B., 2020. Machine learning for digital soil mapping: Applications, challenges and suggested solutions. Earth-Sci. Rev. 210, 103359.

Wang, J., Peng, J., Li, H., Yin, C., Liu, W., Wang, T., Zhang, H., 2021. Soil salinity mapping using machine learning algorithms with the Sentinel-2 MSI in arid areas, China. Remote Sens. 13 (2).

Wang, N., Xue, J., Peng, J., Biswas, A., He, Y., Shi, Z., 2020. Integrating remote sensing and landscape characteristics to estimate soil salinity using machine learning methods: A case study from southern Xinjiang, China. Remote Sens. 12 (24).

Wang, F., Yang, S., Yang, W., Yang, X., Jianli, D., 2019. Comparison of machine learning algorithms for soil salinity predictions in three dryland oases located in Xinjiang Uyghur autonomous region (XJUAR) of China. Eur. J. Rem. Sens. 52 (1), 256–276.

Wu, W., Zucca, C., Muhaimeed, A.S., Al-Shafie, W.M., Fadhil Al-Quraishi, A.M., Nangia, V., Zhu, M., Liu, G., 2018. Soil salinity prediction and mapping by machine learning regression in central Mesopotamia, Iraq. Land Degrad. Dev. 29 (11), 4005–4014.

Zarei, A., Hasanlou, M., Mahdianpari, M., 2021. A Comparison of Machine Learning Models for Soil Salinity Estimation Using Multi-Spectral Earth Observation Data. ISPRS Ann. Photogramm. Rem. Sens. Spatial Inform. Sci. 53, 257–263.

Zeraatpisheh, M., Garosi, Y., Reza Owliaie, H., Ayoubi, S., Taghizadeh-Mehrjardi, R., Scholten, T., Xu, M., 2022. Improving the spatial prediction of soil organic carbon using environmental covariates selection: A comparison of a group of environmental covariates. Catena 208, 105723.

Zhang, J., Schmidt, M.G., Heung, B., Bulmer, C., Knudby, A., 2022. Using an ensemble learning approach in digital soil mapping of soil pH for the Thompson-Okanagan region of British Columbia. Can. J. Soil Sci..