Zahra: Bulletin Big data, Data Science and Artificial Intelligence
Vol. 2, No. 1, [Juni 2023], pp. [21–30]
E-ISSN 2830-7984

Research Article Open Access (CC–BY-SA)

Differential Evolution–Based Weighted Ensemble of Extra Trees, XGBoost, LightGBM, and CatBoost for Relative Soil Fertility Classification

Mazlina Abdul Majid [1], Septiana Vratiwi [2], Agus Perdana Windarto [3*], Sankar Kumar Roy [4]

[1] *Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdullah, Pahang, Malaysia.*
[2] *Faculty of Intelligent Computing, Universiti Malaysia Perlis (UniMAP), Malaysia*
[3] *Department of Informatics, Master's Program, STIKOM Tunas Bangsa, Pematangsiantar, Indonesia*
[4] *Department of Applied Mathematics, Vidyasagar University, Midanpore, India*

[1] *mazlina@umpsa.edu.my;* [2] *septiana@studentmail.unimap.edu.my;* [3] *agus.perdana@amiktunasbangsa.ac.id;* [4] *sankroy2006@gmail.com*
** Corresponding author*

Abstract

This study evaluates differential-evolution weighting of Extra Trees, XGBoost, LightGBM and CatBoost for three-class relative soil fertility prediction. A 20,000-row sample from a 135,143-row local dataset is divided into 15,747 training and 4,253 test observations using coordinate blocks. The original Fertility_Level is constant; relative classes are defined by training-set Fertility thresholds of 431 and 485. Three-fold group cross-validation generates out-of-fold probabilities from four fixed learners. Differential evolution searches four bounded logits, transformed into positive normalized weights, to minimize negative pooled macro-F1 plus a small penalty for deviation from equal weights. Recorded test macro-F1 is 86.98% for equal weighting and 88.41% for differential-evolution weighting. However, Extra Trees alone reaches 88.82% and also has higher accuracy, 92.19% versus 91.72%. Weight optimization therefore improves the equal-weight ensemble but does not outperform the recorded single-model comparator. The learned weights and optimizer convergence record are not embedded. One split, previously inspected historical test data and relative labels limit the findings to a retrospective comparison, without establishing universal ensemble superiority or agronomic validity.

Keywords: relative soil fertility; DE weighted ensemble; spatial cross-validation; model selection; macro-F1

1. Introduction

The growing demand for spatial soil information has encouraged the use of machine learning to support fertility assessment and soil management. Recent reviews describe a broad range of learning approaches for this purpose [1], [2], while soil science-informed modeling emphasizes that a predictive relationship must remain consistent with the meaning and limitations of the target [3]. Soil-fertility-index prediction and real-time fertility analysis illustrate different application settings [4], [5]. These contributions motivate investigating relative fertility classification, but they do not make a class formed from a numerical score equivalent to an independently validated agronomic diagnosis. The present study therefore distinguishes the supplied fertility measurements from the three relative labels constructed for the experiment.

The choice of predictors and evaluation design is central to interpreting a reported improvement. Studies of UAV hyperspectral nutrient monitoring, spatial-neighbor information and parent-material covariates examine different sources of predictive information [6], [7], [8]. Greedy selection, recursive variable selection and broader feature-selection research show why the input representation should be stated explicitly [9], [10], [11]. General feature-engineering and scaling studies likewise treat preprocessing as part of the learning procedure [12], [13].

These findings motivate documenting the available predictors and every transformation rather than attributing a higher score to an unspecified feature intervention. In the current experiment, Latitude, Longitude, NDVI, Soil_Moisture and Elevation are the only five original predictors; Fertility and Fertility_Level are excluded from the model input.

Spatial dependence and model selection require separate attention. Soil-profile leakage, digital-soil-mapping partition experiments and extrapolation studies show that performance depends on the validation unit and prediction setting [14], [15], [16]. Spatial validation and clustered-sample assessment address related concerns [17], [18], [19]. Hyperparameter-optimization and automated-learning research further requires an explicit search domain, objective and budget [20], [21], [22]. The present comparison uses one coordinate-block holdout and training-only group cross-validation. Its purpose is to assess the recorded model configurations under that partition, with macro-F1 as the selection endpoint. This scope does not establish transfer to a new region, calibration quality or a fertilizer recommendation.

The present study examines whether differential evolution can improve a four-model probability ensemble relative to equal weights under one coordinate-block holdout. Extra Trees, XGBoost, LightGBM and CatBoost generate three-fold out-of-fold probabilities on the training data. Differential evolution selects normalized weights using a regularized pooled macro-F1 objective. The comparison retains the recorded Extra Trees single-model benchmark, allowing the study to report whether combining models also surpasses that individual alternative.

2. Method

2.1. Related Work

Recent soil-fertility reviews synthesize models for assessing fertility and supporting management decisions [1], [2]. Soil science-informed machine learning places the scientific target and domain knowledge within that modeling process [3]. The urban Loess Plateau study predicts a soil fertility index using tree-based models [4], while a real-time analysis study considers fertility analysis and crop-related outputs [5]. These studies concern different targets, measurements and application settings. Their performance estimates cannot validate the relative classes used here, which are defined by training-set quantiles of a supplied Fertility score.

Environmental representation has been examined through UAV hyperspectral nutrient monitoring [6], explicit soil spatial-neighbor information [7], and the addition of parent-material information in Bogor [8]. These interventions supply different information to a model. The current five-variable input does not contain hyperspectral bands, neighboring measured soil responses or a parent-material map. Consequently, their reported benefits provide context for representation design rather than evidence that the current predictors recover the same soil processes.

Feature-selection research in digital soil mapping includes modified greedy selection for parsimonious models [9] and recursive variable selection for multiple soil properties [10]. A recent high-dimensional study evaluates interactions between metaheuristic selection, model architecture and prediction target [23]. General surveys distinguish selection of an existing subset from construction of new predictors [11], [12]. This distinction matters when a method title mentions feature selection: an evaluated search procedure does not by itself establish that its selected model uses fewer inputs or that the selected subset is stable.

Ensemble-based micronutrient mapping provides an application of combined learners to regional soil information [24]. A comparative soil-pH study evaluates tree models together with residual kriging [25], and Bayesian–machine-learning synergy for soil organic matter combines spatial modeling with a machine-learning predictor [26]. Multi-feature WMTR-CatBoost modeling addresses soil organic carbon and nitrogen [27]. These are useful examples of changing the learner or integrating complementary information. Their continuous targets, covariate sets and spatial treatments differ from the present three-class experiment; they do not imply that a particular ensemble or boosting implementation must win here.

Hyperparameter optimization treats the search algorithm, domain, objective and evaluation budget as elements of one procedure [20]. Automated-learning reviews place these choices alongside preprocessing and learner selection [21], and empirical tree-tuning research demonstrates the importance of stating the effective tree configuration [22]. A bounded search should therefore be distinguished from a fully documented winning run. The number of configured trials alone does not recover the trial history, selected parameters or variability of repeated optimization.

A recent review describes differential evolution through its search operators, control parameters and algorithm variants [28]. Optimization of a population of parameter vectors is conceptually separate from averaging predictions of several learners. In the present experiment, differential evolution searches four softmax logits whose normalized values weight fixed base-model probabilities. It does not tune the base learners, select a feature subset or implement a new adaptive differential-evolution variant. The relevant comparison is therefore the recorded weighting procedure against equal weights and the supplied Extra Trees single-model benchmark.

Leakage research describes how preprocessing, model selection and repeated inspection can compromise the interpretation of an assessment [29], [30], [31]. REFORMS provides recommendations for reporting machine-learning-based science [32]. These contributions motivate recording the exact development boundary and retaining the data, search settings and final model information. In the current study, the relative-label thresholds are obtained from the complete outer training set before inner cross-validation; they are not re-estimated separately in each inner fold. Thus, the reported CV objective is conditional on that outer-training target definition.

Method-comparison research explains why benchmarks and comparators affect the conclusions drawn from an experiment [33], [34]. Cross-validation theory distinguishes a development estimate from the behavior of a final fitted procedure [35]. For this study, the relevant contrast is between the explicitly recorded configurations on the same held-out rows. A single split and one search seed do not establish a general ranking of algorithms, and differences between differently budgeted procedures do not isolate an optimizer effect.

The soil-profile leakage study specifically considers dependent observations within profiles [14], whereas partition-randomness research examines the effect of resampling choices in digital soil mapping [15]. Spatial+ and clustered-sample assessment address geographic dependence under their respective validation designs [17], [18]. Block-design research further shows that the chosen block structure affects what a spatial test means [19]. Here, coordinate-based group identifiers are kept disjoint across the outer split and across inner folds. No distance buffer is implemented, so disjoint group codes are not interpreted as a guarantee of independence at every spatial scale.

Multiclass metric guidance distinguishes overall accuracy from class-balanced and support-weighted summaries [36], [37]. Learning-curve research considers changes in predictive behavior with fitting-set size [38]. These studies motivate reporting macro-F1, balanced accuracy, per-class recall and confusion counts, together with the recorded internal learning curves. The curves reuse development observations after configuration selection, and therefore serve as diagnostics rather than independent confirmation of the final result.

Uncertainty validation in digital soil mapping concerns the reliability of probabilistic predictions [39]; classifier-calibration research similarly distinguishes probability reliability from hard-label correctness [40]. The review of explainable spectral soil-fertility modeling examines another dimension of model interpretation [41]. Deployment studies document additional constraints beyond an accuracy table [42]. These contributions define questions requiring separate evidence. No calibration experiment, explanation analysis, operational timing benchmark or field-level decision evaluation is inferred from the stored confusion matrices.

2.2. Dataset

The experiment uses the local Sakri_Soil_Fertility_Data.csv file, which contains 135,143 rows and seven columns. A deterministic pandas sample of 20,000 rows is drawn with random_state=42 and its row index is reset. The five numerical predictors are Latitude, Longitude, NDVI, Soil_Moisture and Elevation. Fertility is used only to form the response, and the supplied Fertility_Level column is constant at High for every source row. Therefore, the study does not use the originally proposed four-category target. Table 1 records the actual dataset and partition.

Table 1. Verified local data and experimental partition

Attribute	Observed specification
Source file	135,143 rows × 7 columns; no missing values in the seven required fields
Experiment sample	20,000 rows; pandas sample random_state=42
Inputs / excluded columns	Five numerical inputs; Fertility and Fertility_Level excluded from predictors
Relative target	Class 0: $F < 431$; class 1: $431 \leq F < 485$; class 2: $F \geq 485$
Training partition	15,747 rows; 78 encoded blocks; class counts 5,184 / 5,235 / 5,328
Test partition	4,253 rows; 22 encoded blocks; class counts 2,402 / 786 / 1,065

The original dataset label is uninformative for classification because every row is High. The notebook instead calculates the one-third and two-thirds quantiles of Fertility using only the outer training rows, obtaining 431 and 485. The same thresholds label the held-out rows, with values equal to a threshold placed in the higher class. The

terms class 0, class 1 and class 2 are used throughout to emphasize their relative meaning. The source does not establish an agronomic unit, field-validated fertility threshold or 30 m observation resolution. The notebook names https://data.mendeley.com/datasets/h73wn2nw56/1 as a candidate source while explicitly stating that identity with the local Sakri file is unverified. This analysis therefore identifies the local file and its actual row counts rather than attributing the experiment to that public record. The source-file SHA-256 is b212a3791007499648ef9c1faebe7018d15253560c0b993a8ac46eb669a54ed7.

Table 2. Implemented development-to-test flow and its limits

Stage	Implemented handling
Sample and group	Sample 20,000 rows; form coordinate-grid codes using 0.01-degree latitude/longitude bins
Outer partition	GroupShuffleSplit: n_splits=1, test_size=0.22, random_state=42; test_size is a group fraction
Define labels	Training Fertility quantiles only; fixed thresholds reused for inner CV and outer test
Inner validation	GroupKFold with three folds; validation sizes 5,253, 5,247 and 5,247
Final comparison	Refit selected configurations on the outer training rows and retain the stored test outputs

The outer train and test group identifiers are disjoint, as are the training and validation groups in each inner fold. The 0.22 split setting refers to groups, so it yields 4,253 rows rather than exactly 22% of 20,000 rows. The grouping has no explicit buffer or distance-based exclusion zone. Group codes, relative labels and a single outer split define the tested task; they do not establish an independently sampled new geographic region.

2.3. Research Framework

Figure 1 summarizes the common data boundary and the recorded comparison arms. Every arm uses the same five original covariates, training-defined target and outer test membership. The selected objective is based on the training group folds; the outer test metrics are retained as the final stored comparison.

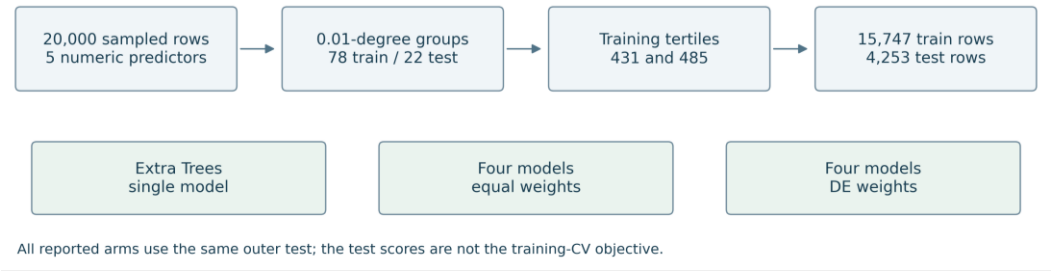

Figure 1. Shared coordinate-block data boundary and comparison arms

Three reported arms share the same split: ExtraTrees_Single, Equal_Weighted_Ensemble with four weights of 0.25, and DE_Weighted_Ensemble. The four base-model settings are fixed before weight selection.

Table 3. Recorded comparison and controlled factors

Component	Study-specific implementation
Fixed comparator	Three reported arms share the same split: ExtraTrees_Single, Equal_Weighted_Ensemble with four weights of 0.25, and DE_Weighted_Ensemble. The four base-model settings are fixed before weight selection.
Selection arm	Three-fold GroupKFold generates one out-of-fold probability vector per training row and base learner. Differential evolution searches four logits in [−5,5], then softmax-normalizes them; the objective uses pooled OOF macro-F1 and an equal-weight penalty.
Shared controls	Same sampled rows, coordinate groups, relative-label thresholds and outer test cases
Reported endpoints	Accuracy, macro-F1, weighted F1, balanced accuracy and class-wise confusion counts
Interpretation boundary	One recorded partition and seed; no independent agronomic validation or repeated-search uncertainty

2.4. Proposed Method

Figure 2 details the implemented representation, selection and refitting stages. The equations in the Proposed Method make the target, grouping, prediction rule and search objective explicit. They describe the supplied code and do not introduce a new experiment.

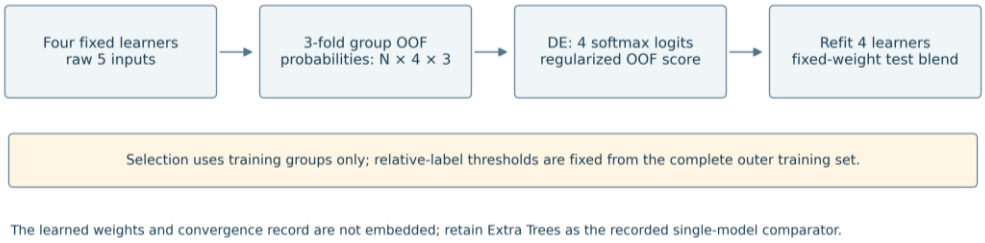

Figure 2. Implemented method and selection boundary

Let $\mathcal{T}$ and $\mathcal{E}$ denote the fixed outer training and test indices. For numerical Fertility score F_i, the empirical training quantiles in (1) define the task. The indicator $\mathbf{1}[P]$ equals one when statement P is true. Equality with a threshold enters the higher class, matching searchsorted with side set to right. These labels express position relative to the training distribution, not validated agronomic thresholds.

$$q_1 = Q_{1/3}(F_{\mathcal{T}}) = 431, \quad q_2 = Q_{2/3}(F_{\mathcal{T}}) = 485, \quad y_i = \mathbf{1}[F_i \geq q_1] + \mathbf{1}[F_i \geq q_2]. \tag{1}$$

For latitude a_i and longitude b_i, (2) gives the coordinate-grid code used for grouping. The minima are computed on the 20,000-row sample, before splitting; the grid width is 0.01 degrees. GroupShuffleSplit assigns complete codes to either outer partition, and GroupKFold uses the training codes for the three inner folds. No geographic distance buffer is applied.

$$u_i = \left\lfloor \frac{a_i - \min_j a_j}{0.01} \right\rfloor, \quad v_i = \left\lfloor \frac{b_i - \min_j b_j}{0.01} \right\rfloor, \quad g_i = 1000u_i + v_i. \tag{2}$$

Table 4. Fixed base-learner configurations before weight selection

Learner	Explicit settings
Extra Trees	220 trees; max_features=0.75; min_samples_leaf=1; n_jobs=2; seed 42
XGBoost	240 rounds; depth 7; learning_rate=0.07; subsample=0.90; colsample_bytree=0.85; hist; multi:softprob; mlogloss; n_jobs=2; seed 42
LightGBM	260 rounds; learning_rate=0.055; num_leaves=48; min_child_samples=17; verbosity=−1; n_jobs=2; seed 42
CatBoost	240 iterations; depth 7; learning_rate=0.095; MultiClass; thread_count=2; verbose=False; seed 42

The four base learners use the same raw five-variable input, class ordering and GroupKFold partitions. Let $\mathcal{I}_f \subset \mathcal{T}$ be validation fold f, $m \in \{1,2,3,4\}$ index Extra Trees, XGBoost, LightGBM and CatBoost, and $k \in \{0,1,2\}$ index classes. Let $\mathcal{T}_{-f}$ contain the rows in the other two folds. Equation (3) defines an out-of-fold (OOF) probability: the learner predicting row i was fitted on $\mathcal{T}_{-f}$. Each training row receives one vector from each learner.

$$P^{\mathrm{OOF}}_{imk} = \hat{p}^{(-f)}_{mk}(x_i), \qquad i \in \mathcal{I}_f, \quad m = 1,\ldots,4, \quad k = 0,1,2. \tag{3}$$

The search variables are four logits $z_m \in [-5,5]$. Softmax normalization in (4) uses $c = \max_m z_m$ for numerical stability. It gives strictly positive weights whose sum is one, so the procedure does not perform exact zero-weight model elimination. Equal weighting corresponds to $w_m = 1/4$.

$$w_m(z) = \frac{\exp(z_m - c)}{\sum_{r=1}^{4} \exp(z_r - c)}, \qquad c = \max_r z_r, \qquad \sum_{m=1}^{4} w_m(z) = 1. \tag{4}$$

For any probability tensor P, (5) combines the four model scores and predicts the class with the largest weighted probability. The same class order is retained for OOF scoring and final test prediction. No calibration model is fitted.

$$\overline{p}_{ik}(z) = \sum_{m=1}^{4} w_m(z) P_{imk}, \qquad \hat{y}_i(z) = \arg\max_{k \in \{0,1,2\}} \overline{p}_{ik}(z). \tag{5}$$

Table 5. Search domain and evaluation controls

Component	Configured domain or control
Base learners	Extra Trees, XGBoost, LightGBM and CatBoost; fixed configurations in Table 4
Search variables	Four logits, each bounded by −5 and 5; weights obtained by softmax

Component	Configured domain or control
Objective	Negative pooled training-OOF macro-F1 + 0.0002 × sum of squared deviations from 0.25
OOF construction	Three GroupKFold splits; probabilities aligned to classes 0, 1 and 2
DE controls	maxiter=80; popsize=10; tol=0.0005; seed=42; polish=True; workers=1
Final evaluation	Refit all four learners on the outer training rows; apply fixed weights to test probabilities

For pooled OOF predictions, TP_k, FP_k and FN_k are counts over all outer-training rows. Equation (6) gives the primary score. This is a single pooled macro-F1, not the arithmetic mean of three fold-specific macro-F1 values.

$$\mathrm{MacroF1}(y, \hat{y}) = \frac{1}{3}\sum_{k=0}^{2} \frac{2\mathrm{TP}_k}{2\mathrm{TP}_k + \mathrm{FP}_k + \mathrm{FN}_k}. \tag{6}$$

Differential evolution minimizes (7), using the OOF tensor in (3). The second term, with coefficient 0.0002, penalizes departures from equal weights and discourages concentration on one learner. It is part of the implemented selection criterion and is retained here even though it is small.

$$\mathcal{J}(z) = -\mathrm{MacroF1}\left(y_{\mathcal{T}}, \hat{y}_{\mathcal{T}}(z; P^{\mathrm{OOF}})\right) + 0.0002 \sum_{m=1}^{4} \left(w_m(z) - \frac{1}{4}\right)^2. \tag{7}$$

Writing $\mathcal{Z}_{\mathrm{eval}}$ for the candidate logits evaluated by the optimizer, (8) defines the returned best evaluated solution. The call sets maxiter=80, popsize=10, tol=0.0005, seed=42, polish=True and workers=1. The four free dimensions imply a nominal population of 40 under the default population initialization. These are configured controls, not an archived convergence guarantee.

$$z^* \in \underset{z \in \mathcal{Z}_{\mathrm{eval}} \subset [-5,5]^4}{\arg\min} \mathcal{J}(z), \qquad w^* = w(z^*). \tag{8}$$

After weight selection, each base learner is refitted on all outer-training rows. The resulting test probabilities replace P in (5), with w^* fixed. Accuracy, weighted F1, balanced accuracy and confusion counts are reported on the same 4,253 test rows for ExtraTrees_Single, equal weighting and DE weighting. The exact learned weights, OOF scores and optimizer convergence record are not embedded, so a numerical value is not invented for w^*.

The notebook writes optimized_weights.csv, oof_individual_comparison.csv and best_params.json, but their numeric contents are not embedded or attached. The learned four weights, pooled OOF scores and convergence flag are therefore not reconstructed from the test results. The supplied single-model holdout comparator is Extra Trees; results for the other three learners individually are not available.

The notebook warns that the historical test had already been inspected. The stored scores are therefore treated as retrospective evidence, rather than an untouched confirmatory evaluation. Bootstrap code or export statements do not supply an uncertainty interval when no corresponding numerical output is embedded.

3. Results and Discussion

3.1. Overall Classification Performance

The following results are taken from the embedded notebook outputs. Aggregate and class-level metrics are cross-checked against the archived confusion-matrix counts. No estimator is retrained to replace or improve the reported outcome. All models in Table 6 are evaluated on the same 4,253 outer-test observations.

Table 6. Recorded outer-test classification performance

Model	Accuracy (%)	Macro-F1 (%)	Weighted F1 (%)	Balanced acc. (%)
ExtraTrees_Single	92.19	88.82	92.20	88.94
Equal_Weighted_Ensemble	90.52	86.98	90.61	87.37
DE_Weighted_Ensemble	91.72	88.41	91.78	88.74

Differential-evolution weighting increases macro-F1 by 1.42 percentage points relative to equal weighting, and accuracy by 1.20 points. It correctly classifies 3,901 test observations versus 3,850 for equal weighting. ExtraTrees_Single nevertheless classifies 3,921 correctly, with macro-F1 higher than the DE ensemble by 0.41 points and accuracy higher by 0.47 points. Extra Trees also has higher weighted F1 and balanced accuracy. The strongest recorded overall comparator is therefore the single Extra Trees model.

3.2. Class-Level Errors

Table 7 reports class recall and F1 on the 0–1 scale. Test supports are 2,402, 786 and 1,065 for relative classes 0, 1 and 2. Class 1 is the smallest test class and remains a major source of confusion. These labels describe training-defined Fertility intervals, not field-validated low, medium or high fertility.

Table 7. Per-class test metrics reconstructed from recorded counts

Model	Class	Support	Precision	Recall	F1
ExtraTrees_Single	0	2402	0.9786	0.9713	0.9749
ExtraTrees_Single	1	786	0.7903	0.7863	0.7883
ExtraTrees_Single	2	1065	0.8924	0.9108	0.9015
Equal_Weighted_Ensemble	0	2402	0.9690	0.9513	0.9601
Equal_Weighted_Ensemble	1	786	0.7338	0.7646	0.7489
Equal_Weighted_Ensemble	2	1065	0.8959	0.9052	0.9005
DE_Weighted_Ensemble	0	2402	0.9772	0.9621	0.9696
DE_Weighted_Ensemble	1	786	0.7686	0.7901	0.7792
DE_Weighted_Ensemble	2	1065	0.8972	0.9099	0.9035

Relative to equal weighting, the DE ensemble increases correct predictions from 2,285 to 2,311 in class 0, from 601 to 621 in class 1, and from 964 to 969 in class 2. Relative to ExtraTrees_Single, it makes 22 fewer correct class-0 predictions and one fewer correct class-2 prediction, while gaining three correct class-1 predictions. The marginal class-1 gain is insufficient to offset the other changes in the macro-F1 comparison.

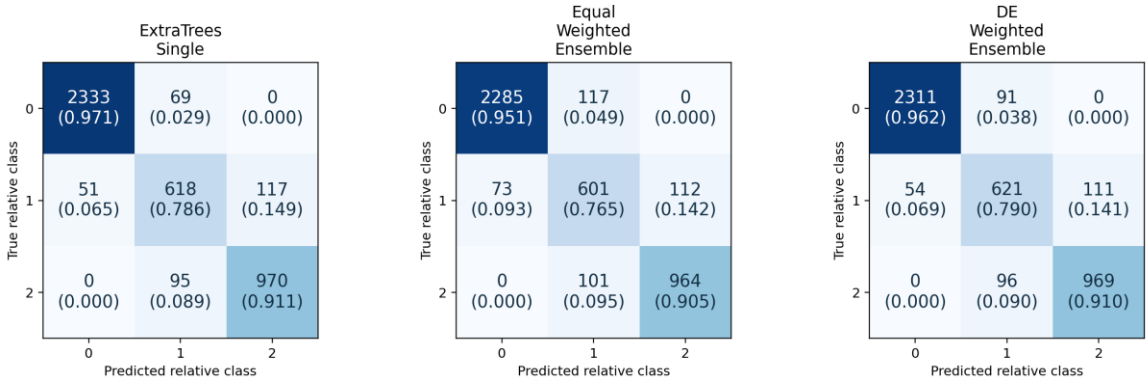

Figure 3. Test confusion matrices; entries show counts and row-normalized proportions

3.3. Internal Learning Curves

Figure 4 reproduces the recorded internal training and validation curves. Fitting subsets contain 4,181, 7,765 and 11,947 observations; the internal validation partition contains 3,800 rows. The split is made inside the outer training data using GroupShuffleSplit, with test_size=0.24 and random_state=50. Smaller fitting subsets are stratified row samples from the internal fitting pool. Selected settings remain fixed during curve refits.

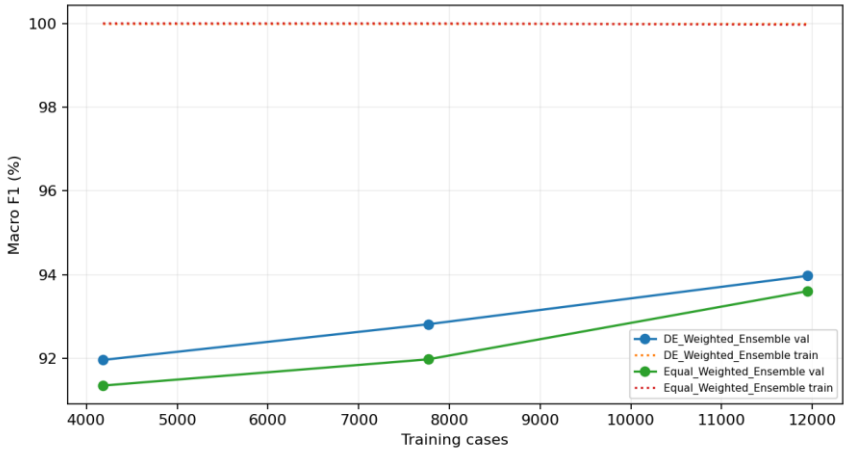

Figure 4. Archived training and internal-validation macro-F1 learning curves

The recorded DE validation curve lies above the equal-weight curve at all three fitting sizes, and both validation curves rise with more fitting observations. Training macro-F1 stays approximately 100%. The fixed DE weights were selected from OOF predictions using the complete outer-training set, which includes the rows later assigned to this curve-validation partition. The curve is therefore a reused-development diagnostic rather than an independent evaluation of weight selection.

3.4. Discussion and Limitations

The result supports the narrower conclusion that learned weights improve this equal-weight ensemble. It does not support the claim that the weighted ensemble surpasses the reported single-model benchmark. The remaining individual learners are not separately evaluated on the held-out set in the supplied outputs, so a ranking against every constituent model cannot be reported. The equal-weight penalty and pooled OOF objective are components of the implemented method; neither probability calibration nor statistical independence among model errors is demonstrated.

The target is a discretization of a supplied numerical Fertility score, and its agronomic validity is not established. The provenance match to the public dataset link remains unverified. The evaluation covers one seed and one coordinate-block split without an explicit spatial buffer; no repeated-seed confidence interval, calibrated probability assessment, feature-importance stability or hardware-specific deployment benchmark is reported. Learning curves reuse development observations after configuration selection. Prior historical-test inspection further limits confirmation. These limits require a bounded interpretation of the observed configuration differences.

4. Conclusion

Differential-evolution weighting improves the recorded four-model ensemble from 86.98% to 88.41% macro-F1 and from 90.52% to 91.72% accuracy. Extra Trees alone remains stronger overall, with 88.82% macro-F1 and 92.19% accuracy. The study therefore reports a positive weighting contrast against equal weights and a negative contrast against the available single-model benchmark. Missing learned weights and convergence outputs, one spatial split and prior historical-test inspection limit reproducibility and confirmation. Complete optimizer records, individual-model comparisons, repeated group partitions and independent agronomic validation are needed before recommending the ensemble for operational use.

References

[1] M. Sujatha and C. D. Jaidhar, “Machine learning-based approaches to enhance the soil fertility—A review,” *Expert Systems with Applications*, vol. 240, Art. no. 122557, 2024, doi: 10.1016/j.eswa.2023.122557.

[2] S. Muthulakshmi, M. R. Backiyavathy, M. Gopalakrishnan, R. Kalpana, C. Thangamani, and S. I. Pavithra, “Insights of Machine Learning Approach for Soil Fertility Assessment and Management Strategy,” *Communications in Soil Science and Plant Analysis*, vol. 56, no. 3, pp. 436–457, 2025, doi: 10.1080/00103624.2024.2416920.

[3] B. Minasny et al., “Soil Science-Informed Machine Learning,” *Geoderma*, vol. 452, Art. no. 117094, 2024, doi: 10.1016/j.geoderma.2024.117094.

[4] X. Shen et al., “Assessment and Prediction of Soil Fertility in Urban Areas of the Loess Plateau Based on Machine Learning Methods,” *CLEAN – Soil, Air, Water*, vol. 53, no. 9, Art. no. e70039, 2025, doi: 10.1002/clen.70039.

[5] K. Gunasekaran, K. A., and P. Sreevardhan, “Real-time soil fertility analysis, crop prediction, and insights using machine learning and deep learning algorithms,” *Frontiers in Soil Science*, vol. 5, Art. no. 1652058, 2025, doi: 10.3389/fsoil.2025.1652058.

[6] K. Liu et al., “Monitoring soil nutrients using machine learning based on UAV hyperspectral remote sensing,” *International Journal of Remote Sensing*, vol. 45, no. 14, pp. 4897–4921, 2024, doi: 10.1080/01431161.2024.2371618.

[7] Z. Chen, Z. Wang, X. Wang, Z. Shi, and S. Chen, “Including soil spatial neighbor information for digital soil mapping,” *Geoderma*, vol. 451, Art. no. 117072, 2024, doi: 10.1016/j.geoderma.2024.117072.

[8] D. Cahyana, Y. Sulaeman, B. Barus, Darmawan, and B. Mulyanto, “Improving digital soil mapping in Bogor, Indonesia using parent material information,” *Geoderma Regional*, vol. 33, Art. no. e00627, 2023, doi: 10.1016/j.geodrs.2023.e00627.

[9] X. Zhang et al., “Improving model parsimony and accuracy by modified greedy feature selection in digital soil mapping,” *Geoderma*, vol. 432, Art. no. 116383, 2023, doi: 10.1016/j.geoderma.2023.116383.

[10] J. Huang et al., “Mapping Soil Properties in the Haihun River Sub-Watershed, Yangtze River Basin, China, by Integrating Machine Learning and Variable Selection,” *Sensors*, vol. 24, no. 12, Art. no. 3784, 2024, doi: 10.3390/s24123784.

[11] M. R. Islam, A. A. Lima, S. C. Das, M. F. Mridha, A. R. Prodeep, and Y. Watanobe, "A Comprehensive Survey on the Process, Methods, Evaluation, and Challenges of Feature Selection," *IEEE Access*, vol. 10, pp. 99595–99632, 2022, doi: 10.1109/access.2022.3205618.
[12] P. Koukaras and C. Tjortjis, "Data Preprocessing and Feature Engineering for Data Mining: Techniques, Tools, and Best Practices," *AI*, vol. 6, no. 10, Art. no. 257, 2025, doi: 10.3390/ai6100257.
[13] L. B. V. de Amorim, G. D. C. Cavalcanti, and R. M. O. Cruz, "The choice of scaling technique matters for classification performance," *Applied Soft Computing*, vol. 133, Art. no. 109924, 2023, doi: 10.1016/j.asoc.2022.109924.
[14] K. John, D. D. Saurette, and B. Heung, "The problematic case of data leakage: A case for leave-profile-out cross-validation in 3-dimensional digital soil mapping," *Geoderma*, vol. 455, Art. no. 117223, 2025, doi: 10.1016/j.geoderma.2025.117223.
[15] D. Radočaj, M. Jurišić, I. Plaščak, and L. Galić, "Randomness in Data Partitioning and Its Impact on Digital Soil Mapping Accuracy: A Comparison of Cross-Validation and Split-Sample Approaches," *Agronomy*, vol. 15, no. 11, Art. no. 2495, 2025, doi: 10.3390/agronomy15112495.
[16] F. Hateffard, L. Steinbuch, and G. B. M. Heuvelink, "Evaluating the extrapolation potential of random forest digital soil mapping," *Geoderma*, vol. 441, Art. no. 116740, 2024, doi: 10.1016/j.geoderma.2023.116740.
[17] Y. Wang, M. Khodadadzadeh, and R. Zurita-Milla, "Spatial+: A new cross-validation method to evaluate geospatial machine learning models," *International Journal of Applied Earth Observation and Geoinformation*, vol. 121, Art. no. 103364, 2023, doi: 10.1016/j.jag.2023.103364.
[18] S. de Bruin, D. J. Brus, G. B. M. Heuvelink, T. van Ebbenhorst Tengbergen, and A. M. J.-C. Wadoux, "Dealing with clustered samples for assessing map accuracy by cross-validation," *Ecological Informatics*, vol. 69, Art. no. 101665, 2022, doi: 10.1016/j.ecoinf.2022.101665.
[19] A. Stock, "Choosing blocks for spatial cross-validation: lessons from a marine remote sensing case study," *Frontiers in Remote Sensing*, vol. 6, Art. no. 1531097, 2025, doi: 10.3389/frsen.2025.1531097.
[20] B. Bischl et al., "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *WIREs Data Mining and Knowledge Discovery*, vol. 13, no. 2, Art. no. e1484, 2023, doi: 10.1002/widm.1484.
[21] M. Baratchi et al., "Automated machine learning: past, present and future," *Artificial Intelligence Review*, vol. 57, no. 5, Art. no. 122, 2024, doi: 10.1007/s10462-024-10726-1.
[22] R. Gomes Mantovani et al., "Better trees: an empirical study on hyperparameter tuning of classification decision tree induction algorithms," *Data Mining and Knowledge Discovery*, vol. 38, no. 3, pp. 1364–1416, 2024, doi: 10.1007/s10618-024-01002-5.
[23] Y. Wei, H. Hu, R. Li, X. Li, and F. Wang, "Digital Mapping of Soil and Water Indicators in Arid Regions Driven by High-Dimensional Environmental Covariates: A Comprehensive Evaluation of Metaheuristic Feature Selection and Hybrid Deep Learning Frameworks," *Remote Sensing*, vol. 18, no. 17, Art. no. 2859, 2026, doi: 10.3390/rs18172859.
[24] S. Dasgupta et al., "Developing regional soil micronutrient management strategies through ensemble learning based digital soil mapping," *Geoderma*, vol. 433, Art. no. 116457, 2023, doi: 10.1016/j.geoderma.2023.116457.
[25] Y. Tian et al., "Digital Mapping of Soil pH Using Tree-Based Models Coupled with Residual Kriging," *Land*, vol. 15, no. 3, Art. no. 365, 2026, doi: 10.3390/land15030365.
[26] Z. Zhuo et al., "Enhancing Digital Mapping of Soil Organic Matter in Cropland With Bayesian-Machine Learning Synergy," *Soil Use and Management*, vol. 42, no. 2, Art. no. e70225, 2026, doi: 10.1111/sum.70225.
[27] J. Zhang, N. Hou, H. Yang, X. Han, and M. Huo, "Prediction of soil organic carbon and nitrogen: a WMTR-CatBoost model based on multi-feature fusion," *European Journal of Remote Sensing*, vol. 59, no. 1, Art. no. 2711851, 2026, doi: 10.1080/22797254.2026.2711851.
[28] M. F. Ahmad, N. A. M. Isa, W. H. Lim, and K. M. Ang, "Differential evolution: A recent review based on state-of-the-art works," *Alexandria Engineering Journal*, vol. 61, no. 5, pp. 3831–3872, 2022, doi: 10.1016/j.aej.2021.09.013.
[29] S. Kapoor and A. Narayanan, "Leakage and the reproducibility crisis in machine-learning-based science," *Patterns*, vol. 4, no. 9, Art. no. 100804, 2023, doi: 10.1016/j.patter.2023.100804.
[30] M. A. Lones, "Avoiding common machine learning pitfalls," *Patterns*, vol. 5, no. 10, Art. no. 101046, 2024, doi: 10.1016/j.patter.2024.101046.
[31] A. Apicella, F. Isgrò, and R. Prevete, "Don't push the button! Exploring data leakage risks in machine learning and transfer learning," *Artificial Intelligence Review*, vol. 58, no. 11, Art. no. 339, 2025, doi: 10.1007/s10462-025-11326-3.
[32] S. Kapoor et al., "REFORMS: Consensus-based Recommendations for Machine-learning-based Science," *Science Advances*, vol. 10, no. 18, Art. no. eadk3452, 2024, doi: 10.1126/sciadv.adk3452.
[33] S. Friedrich and T. Friede, "On the role of benchmarking data sets and simulations in method comparison studies," *Biometrical Journal*, vol. 66, no. 1, Art. no. 2200212, 2024, doi: 10.1002/bimj.202200212.

[34] J. L. J. Pereira, K. Smith-Miles, M. A. Muñoz, and A. C. Lorena, "Optimal selection of benchmarking datasets for unbiased machine learning algorithm evaluation," *Data Mining and Knowledge Discovery*, vol. 38, no. 2, pp. 461–500, 2024, doi: 10.1007/s10618-023-00957-1.

[35] S. Bates, T. Hastie, and R. Tibshirani, "Cross-Validation: What Does It Estimate and How Well Does It Do It?," *Journal of the American Statistical Association*, vol. 119, no. 546, pp. 1434–1445, 2024, doi: 10.1080/01621459.2023.2197686.

[36] S. Farhadpour, T. A. Warner, and A. E. Maxwell, "Selecting and Interpreting Multiclass Loss and Accuracy Assessment Metrics for Classifications with Class Imbalance: Guidance and Best Practices," *Remote Sensing*, vol. 16, no. 3, Art. no. 533, 2024, doi: 10.3390/rs16030533.

[37] J. Opitz, "A Closer Look at Classification Evaluation Metrics and a Critical Reflection of Common Evaluation Practice," *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 820–836, 2024, doi: 10.1162/tacl_a_00675.

[38] T. Viering and M. Loog, "The Shape of Learning Curves: A Review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 6, pp. 7799–7819, 2023, doi: 10.1109/tpami.2022.3220744.

[39] J. Schmidinger and G. B. M. Heuvelink, "Validation of uncertainty predictions in digital soil mapping," *Geoderma*, vol. 437, Art. no. 116585, 2023, doi: 10.1016/j.geoderma.2023.116585.

[40] T. Silva Filho, H. Song, M. Perello-Nieto, R. Santos-Rodriguez, M. Kull, and P. Flach, "Classifier calibration: a survey on how to assess and improve predicted class probabilities," *Machine Learning*, vol. 112, no. 9, pp. 3211–3260, 2023, doi: 10.1007/s10994-023-06336-7.

[41] J. V. Ribeiro, S. Barbon Junior, F. R. dos Santos, and F. L. Melquiades, "Explainability in vis-NIR and XRF-based modeling for soil fertility analysis: A comprehensive review," *TrAC Trends in Analytical Chemistry*, vol. 199, Art. no. 118776, 2026, doi: 10.1016/j.trac.2026.118776.

[42] A. Paleyes, R.-G. Urma, and N. D. Lawrence, "Challenges in Deploying Machine Learning: A Survey of Case Studies," *ACM Computing Surveys*, vol. 55, no. 6, pp. 1–29, 2023, doi: 10.1145/3533378.