

Statistically grounded comparison of supervised regression algorithms for magnetic ore quality prediction*

Bohdan Kiselov^{1,*†}, Anton Senko^{2,†}, Serhiy Semerikov^{3,†} and Dennis Kuznetsov^{2,†}

¹ Department of Automation, Computer Sciences and Technology, Kryvyi Rih National University, Vitalii Matusevych 11, Kryvyi Rih, 50027, Ukraine

² Department of Computer Systems and Networks, Kryvyi Rih National University, Vitalii Matusevych 11, Kryvyi Rih, 50027, Ukraine

³ Department of Professional and Social-Humanitarian Education, Kryvyi Rih National University, Vitalii Matusevych 11, Kryvyi Rih, 50027, Ukraine

Abstract

This study compares five supervised regression models for predicting the magnetic quality indicator KT10valueMax of individual ore fragments on an industrial conveyor-line sorting system. The fragments are characterised in real time by two sensors: a laser scanner that records a single top-down height map, from which volume, projected area and maximum height are approximated, and a magnetic induction sensor that records a raw induction value together with the used coil area, the centre deviation and the sensor temperature. Seven features derived from these two sources serve as predictors. The models examined are HistGradientBoosting, Random Forest, XGBoost, Support Vector Regression and Ridge regression. Because fragments from the same ore batch share unmeasured conditions, generalisation was estimated with GroupKFold cross-validation at twenty outer folds, so that no scan appeared in both the training and the validation set of a fold. Each model was tuned independently on every fold with the Optuna framework. The coefficient of determination and the root mean squared error were recorded per fold, and the five score vectors were compared with the Friedman test followed by the Nemenyi and the Wilcoxon signed-rank post-hoc tests with Holm correction. XGBoost reached the highest mean coefficient of determination (0.790), with Random Forest close behind (0.777); Ridge regression remained competitive (0.637), which indicates that a large part of the sensor-to-quality mapping is linear. The Friedman test rejected the hypothesis of equal model rank, and the only pair that survived the Holm correction was XGBoost against Ridge. Two folds produced extreme failures that are reported rather than removed, because they expose how boosting and kernel methods behave when a test batch falls outside the training range. XGBoost is recommended for deployment, with Random Forest as a robust alternative.

Keywords

sensor-based ore sorting, magnetic induction sensor, supervised regression, XGBoost, GroupKFold cross-validation, Friedman test, hyperparameter optimisation

1. Introduction

Average ore grades at established deposits keep falling while demand for metals tied to the energy transition keeps rising [1]. To hold metal output steady, operators now process far larger volumes of low-grade rock, and energy, water, reagent use and tailings all grow in step [1,2]. Removing waste

* ADT-ISMDCC 2026: Advanced Digital Technologies: Intelligent Systems, Modeling, and Data-Driven Control, September 22–24, 2026, Odesa, Ukraine

* Corresponding author.

† These authors contributed equally.

✉ kiselov.bogdan@knu.edu.ua (B. Kiselov); antonsenko@knu.edu.ua (A. Senko); semerikov@acnsci.org (S. Semerikov); kuznetsov.dennis.1706@knu.edu.ua (D. Kuznetsov)

ORCID 0009-0007-9338-1031 (B. Kiselov); 0000-0002-4104-8372 (A. Senko); 0000-0003-0789-0272 (S. Semerikov); 0000-0002-2021-5207 (D. Kuznetsov)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

rock before it reaches crushing and grinding is therefore one of the few levers that lowers both cost and environmental load at once [1,2].

Sensor-based ore sorting (SBOS) does exactly this for coarse fractions, typically between 10 mm and 300 mm, on single-piece conveyor lines [1]. Each fragment is measured on the move by one or more sensors; an algorithm labels it product or waste; and a row of compressed-air jets diverts it within milliseconds [1,2]. Diverting gangue ahead of comminution can cut the mass sent to crushing by 20–50%, with matching savings downstream [2].

The sensing technologies in use are varied. X-ray transmission and fluorescence dominate hard-rock applications, followed by near-infrared spectrometry, colour and texture imaging, laser-induced breakdown spectroscopy, electromagnetic induction and prompt gamma neutron activation [1]. None is best for every ore, which is why multi-sensor fusion has drawn attention: complementary measurements combined together discriminate better than any single channel [1,2]. Recent reviews single out fusion with machine learning as the most promising route for ores that defeat single-sensor systems [2]. A threshold rule reads one engineered feature; a regression model reads many at once. It can weigh geometric descriptors – fragment volume, surface area, maximum height, centre deviation – alongside electromagnetic ones – the raw induction value, the active coil area and the sensor temperature – and return a continuous quality estimate per fragment. That fuller view of the measurement space is what makes machine learning (ML) attractive for sorting [1,2].

Which model to use is not a neutral choice. Random Forest (RF), gradient-boosted trees such as XGBoost and HistGradientBoosting, and kernel methods such as Support Vector Regression (SVR) carry different inductive biases, react differently to tuning, and generalise differently on small or structured data [3]. On industrial sensor data these differences open performance gaps that are not always larger than sampling noise, so a test is needed to tell a real gap from a lucky split [4].

Demšar [4] set out the standard answer: analysis of variance is unsuitable for cross-validated scores, and the non-parametric Friedman test followed by Nemenyi or Wilcoxon post-hoc comparisons with Holm correction should be used instead. This protocol is now routine in general ML work [4,5], yet it is rarely applied in ore sorting, where sensing hardware has advanced faster than the statistics used to judge the algorithms behind it.

1.1. Aim and objectives

The study compares five supervised regression models – HistGradientBoosting, Random Forest, XGBoost, SVR and Ridge regression – for estimating the magnetic quality indicator KT10valueMax from combined geometric and electromagnetic features recorded per fragment on an industrial conveyor line. Generalisation is judged with GroupKFold cross-validation ($k = 20$), which keeps each ore batch on one side of every train/validation split and so reports performance close to what deployment would give.

The objectives are:

- to build a reproducible pipeline for training and evaluating regression models on multi-sensor ore-sorting data;
- to compare the models by the coefficient of determination (R^2) and the root mean squared error (RMSE) across folds;
- to apply the Friedman test to the per-fold R^2 ranks;
- to run Nemenyi and Wilcoxon–Holm post-hoc tests to locate the model pairs that differ;
- to identify the algorithm that best balances accuracy, robustness and fitness for real-time sorting.

1.2. Object and subject

The object is an automated conveyor-line sorting system with two sensors: a magnetic induction sensor for the electromagnetic response of each fragment, and a laser scanner for shape from a single top-down view. Seven features follow from them – Volume, Area, MaxHeight, CenterDeviation, UsedCoilArea, SensorTemp and SensorValueRAW – and the target is KT10valueMax, the KT10 reference reading at the fragment centre. The subject is the statistical comparison of supervised regression models on this task, with attention to the validity of the comparison and to what model choice means for sorting in practice.

2. Literature review

2.1. Sensor-based ore sorting

SBOS grew out of early optical and colour sorting into a field built on several physical sensing modalities, each reading a different property of the rock [1,5]. Reviews list the technologies in industrial use as X-ray transmission and dual-energy XRT, X-ray fluorescence, near-infrared spectrometry, laser-induced breakdown spectroscopy, optical colour and texture imaging, prompt gamma neutron activation and electromagnetic induction [1,5]. XRT leads in hard-rock work because it resolves interior density contrasts, whereas magnetic induction suits magnetically susceptible ores such as magnetite and some sulphides [1,5].

Three factors set sorting efficiency: how well the sensor signal tracks ore grade, how reliably the actuator diverts the fragment, and how well the algorithm turns readings into decisions [5]. Faster acquisition and cheaper computing have moved the bottleneck from hardware throughput to the quality of that algorithm, which is what motivates richer data-driven models [5].

A single sensor reads one property, often too little for complex or finely disseminated ores. Fusion architectures address this by processing complementary signals – geometry together with electromagnetic response – jointly [1,2]. Peukert et al. [1] proposed a taxonomy of fusion in mineral processing and showed that fused data widen the range of treatable ore types; Adewuyi et al. [2] reviewed single-sensor limits and put forward a combined machine-learning and computer-vision framework for industry.

2.2. Machine learning for ore quality prediction

Supervised ML for ore-quality prediction has spread over the past decade as sensor datasets and open-source libraries matured [2,5]. The usual pattern compares several algorithms – tree ensembles, kernel machines, linear models – on a domain task, scored by R^2 and RMSE [3]. In mineral processing, ensembles have tended to beat linear baselines on non-linear multivariate tasks; gradient-boosted trees and Random Forests report higher R^2 than linear regression on ore-grade and quality-indicator problems where features interact [2,5]. This matches the wider tabular-data literature, where tree ensembles match or exceed neural networks at modest dataset sizes and low to moderate dimensionality [6,7]. The margin is not guaranteed, though: when the dominant relationship is close to linear – as it may be when the magnetic signal scales nearly with mineral content – a regularised linear model like Ridge can compete at far lower cost and with more transparency [3].

2.3. Regression models compared

Random Forest. RF [7] grows many decision trees, each on a bootstrap sample and a random feature subset per split, and averages them. Averaging cuts variance at little cost in bias, which makes RF a

robust tabular baseline that tolerates noisy or irrelevant inputs [7]. Reported R^2 in mineral-processing tasks runs roughly 0.70–0.95 depending on data quality [2].

Extreme Gradient Boosting. XGBoost [6] builds trees in sequence, each fitting the residual of the current ensemble, with a complexity penalty written into the objective. Sequential correction plus regularisation gives strong tabular accuracy, and the method handles missing values natively [6].

HistGradientBoosting. The histogram-based booster in Scikit-learn bins continuous features before searching for splits, which speeds training while keeping accuracy, and adds L2 regularisation and native missing-value support. It is less studied in mineral processing than XGBoost but competitive on tabular tasks [3].

Support Vector Regression. SVR [10] fits a function inside an ϵ -tube around the data while minimising the weight norm; the radial basis kernel lets it model non-linearity without explicit feature engineering. It is scale-sensitive and depends on careful tuning of C , ϵ and γ , more so than tree methods [3].

Ridge regression. Ridge [11] adds an L2 penalty to least squares, shrinking coefficients to trade a little bias for lower variance. It is the linear baseline: if it matches the non-linear models, the mapping is largely linear; if it lags far behind, the extra complexity is justified.

The five models were chosen on three grounds. First, each has a record in mineral-processing or sensor regression: ensembles lead on ore-grade and quality prediction [2,5], SVR holds up on small-to-medium sensor sets [3], and Ridge is the canonical linear reference [11].

Second, each meets a specific feature of this dataset: Ridge fixes the linear ceiling; RF bagging resists individual anomalous batches; XGBoost regularisation curbs overfitting when an outer fold holds only four or five scans; SVR with an RBF kernel can express the non-additive geometry–magnetism interaction without engineered terms; and HGB represents the default gradient booster in industrial Python stacks.

Third, three plausible alternatives were left out. A multilayer perceptron needs more data than 4,161 measurements across 90 scans of 45 ore pieces can regularise without overfitting; Gaussian process regression scales as $O(n^3)$, which conflicts with retraining on a live line; and k-nearest neighbours interpolates poorly across new batch distributions – the very failure GroupKFold is meant to expose.

2.4. Cross-validation for grouped data

Cross-validation is only valid when the independence it assumes holds. Industrial data rarely meet it: fragments from one batch share conveyor speed, ambient temperature, calibration state and ore source, so they correlate within a group but not across groups. If a batch sits in both the training and validation set of a fold, those correlations inflate the score above what new batches would give – a form of leakage [3].

GroupKFold removes this by forcing every measurement of a group onto one side of each split. The result is a conservative estimate that matches deployment, where the model meets unseen groups [3]. Using $k = 20$ here follows that practice and yields twenty independent scores per model for the comparison.

2.5. Statistical framework

Picking a model by mean score alone, without a significance test, is a known source of false conclusions [3,4]. The mean hides fold-level spread, so a gap of, say, 0.03 R^2 could be real or could be the luck of the split. Demšar [4] formalised this and gave a two-stage test.

Stage one is the Friedman test, a rank-based two-way ANOVA on the per-fold scores; its null is that all models share the same expected rank. If it rejects at $\alpha = 0.05$, stage two locates the differing pairs [4]. This study uses two post-hoc tests: Nemenyi, with one critical difference for all pairs, and Wilcoxon signed-rank per pair with Holm step-down correction [3,4]. The pairing is now standard for rigorous benchmarking [3,4].

2.6. Automated hyperparameter optimisation

A fair comparison needs every model near its best configuration, not at an arbitrary default. Grid search is prohibitive for large spaces such as XGBoost and SVR. Bayesian optimisation instead builds a surrogate of the objective – here the cross-validated R^2 – and picks configurations that balance exploration and exploitation [8]. Optuna [8] implements a define-by-run interface with a Tree-structured Parzen Estimator sampler and early pruning of weak trials, which cuts cost against exhaustive search [8].

2.7. Research gap

The review exposes a gap: individual algorithms have been applied to ore-quality prediction, but rigorous comparisons of several regression families on multi-sensor sorting data – with group-aware cross-validation and non-parametric testing – are scarce. Existing studies often report means without significance tests, ignore batch structure, or tune only some models, biasing the result [2,3]. This study tunes all five models with the same Bayesian procedure, evaluates them under the same GroupKFold scheme, and compares them with the same two-stage test, yielding a statistically grounded recommendation for conveyor-line sorting.

3. Methodology

3.1. Dataset and sensor architecture

The data come from an industrial conveyor-line sorting system with two independent sensors running in parallel. The first is a laser scanner that views each fragment from a single top-down orientation and produces a two-dimensional height map of its upper surface only; the underside and sides are not measured, so this is not a full three-dimensional reconstruction. From the map, two scalars are approximated per fragment: Volume, by numerical integration of the height field over the footprint, and Area, the projected surface area from the laser footprint. Both are approximations rather than direct measurements. A third descriptor, MaxHeight, is the tallest point in the map. Because Volume and Area come from one side only, they systematically underestimate irregular fragments – a point returned to in the Discussion when feature behaviour is interpreted.

The second sensor is a magnetic induction array that scans each fragment over a set of coils. Four values are extracted: SensorValueRAW, the raw induction output; UsedCoilArea, the coil area the fragment covers; CenterDeviation, the offset in millimetres from the planned coil centre; and SensorTemp, the sensor temperature at measurement, included to account for thermal drift in the induction response.

The target, KT10valueMax, is a continuous quality indicator: the KT10 reading at the fragment centre, taken as ground truth. The metadata columns Label, DateTime and SensorId are excluded from the inputs. The seven predictors are therefore Volume, Area, MaxHeight, CenterDeviation, UsedCoilArea, SensorTemp and SensorValueRAW.

The dataset holds 4,161 measurements across 90 single-sided scans of 45 ore pieces drawn from 14 ore batches, each piece scanned from two opposite sides. KT10valueMax spans roughly 0.03 to 624.00,

a wide range that reflects real geological and operational spread between batches. Each batch is measured under nominally constant conditions, but between-batch variation is genuine and is exactly what the model must generalise across. This group structure is what makes a group-aware scheme necessary, as set out in Section 3.4.

3.2. Features and pre-processing

All seven features are numerical and continuous. A pre-processing pipeline is fitted inside each fold, on the training partition only, to keep validation data out of the fitted transforms. It has two steps. The first imputes missing values – caused by triggering failures, communication errors or occlusion – with the training-fold median, which resists outliers from sensor faults. The second standardises each feature to zero mean and unit variance using training-fold parameters. SVR and Ridge both require this [9,10,11]; the tree ensembles are scale-invariant but run through the same pipeline for consistency. Both steps use Scikit-learn [9].

3.3. Candidate models

The five models span a deliberate range, from a regularised linear baseline through a kernel method to ensemble trees, so the conclusions apply across the complexity spectrum.

HistGradientBoosting (HGB) is the histogram-based booster in Scikit-learn [9], built on Friedman’s gradient boosting [12]: an additive ensemble fits trees to the pseudo-residuals under a differentiable loss. Binning features before the split search cuts cost while keeping accuracy [9,12], and L2 regularisation with native missing-value handling helps here.

Random Forest (RF) bags independently grown trees with a random feature subset per split and averages them [7]. A deep single tree has low bias and high variance; averaging many bootstrapped, feature-subsampled trees cuts that variance with little added bias [7]. RF is the reference ensemble against which boosting is judged.

XGBoost (XGB) is a regularised booster with an explicit per-tree complexity penalty and second-order gradient statistics for faster, steadier convergence [6]. Unlike RF it builds trees in sequence, each correcting the last; its tabular efficiency and wide industrial use make it a natural candidate [6].

SVR fits $f(x) = w^T\phi(x) + b$ inside an ϵ -insensitive tube while minimising $\|w\|^2$ in the kernel feature space [10]. Points outside the tube enter the loss through slack variables bounded by C . The RBF kernel maps inputs to a space where non-linear structure becomes linear; three hyperparameters – C , ϵ and γ – are tuned [10].

Ridge regression adds an L2 penalty to least squares, giving the estimator of equation (1):

$$\hat{\beta}_{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \{ \|y - X\beta\|^2 + \lambda \|\beta\|^2 \} \quad (1)$$

where y is the vector of n observed targets; X is the $n \times p$ matrix of standardised features; β is the p -dimensional coefficient vector; and $\lambda \geq 0$ controls the trade-off between fit and shrinkage [11]. The closed form:

$$\hat{\beta}_{\text{ridge}} = (X^T X + \lambda I)^{-1} X^T y \quad (2)$$

is unique regardless of collinearity [11]. Ridge is the linear baseline: its standing against the non-linear models is direct evidence on how linear the sensor-to-quality mapping is.

All five are implemented in Python with Scikit-learn [9] and the XGBoost library [6].

3.4. Cross-validation scheme

Performance is estimated with GroupKFold at $k = 20$ folds [3,9]. On each iteration one fold – made up of scans absent from the other nineteen – is held out for validation while the rest train. The grouping variable is the scan identifier, so all measurements of one scan stay on the same side of every split. Because each piece is scanned from two opposite sides, the paired face of a held-out scan may fall in the training set of another fold, which makes the estimate a slightly optimistic bound relative to piece-level holdout, as noted in Section 5.3. This gives twenty independent (fold, model) scores per algorithm, the matrix used in Section 3.7.

The pipeline of Section 3.2 is re-fitted on each training partition, so imputation medians and standardisation parameters never see validation data. All five models use identical fold partitions, so any difference traces to the algorithm rather than to the split [3].

3.5. Evaluation metrics

Two metrics are computed on each fold's validation set. The primary metric is R^2 , equation (3):

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (3)$$

where y_i is the observed KT10valueMax for fragment i ; $\hat{y}_i$ is the prediction; $\bar{y}$ is the validation-set mean; and n is the fold size. R^2 runs from $-\infty$ to 1, with 1 perfect and 0 equal to predicting the mean [3].

The secondary metric is $RMSE$, equation (4):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4)$$

with symbols as in equation (3). $RMSE$ is in the units of KT10valueMax and reads directly as a mean error [3]. R^2 drives the statistical comparison of Section 3.7; $RMSE$ is reported alongside for absolute error magnitude.

3.6. Hyperparameter optimisation

Each model is tuned with Optuna [8]. For every model and outer fold, an inner loop splits the training partition into a nested 5-fold scheme and Optuna minimises the negated mean inner-fold R^2 . The TPE sampler is used with median pruning of weak trials [8]. The budget is 100 trials per model per outer fold, a balance between thoroughness and cost.

The search spaces are: HGB – learning rate log-uniform on $[10^{-3}, 0.3]$, max leaf nodes from {15, 31, 63, 127, 255}, min samples per leaf from {5, 10, 20, 50}, L2 from $[10^{-3}, 10]$. RF – trees from {100, 200, 400}, max features from {1/3, 1/2, sqrt}, min samples per leaf from {1, 2, 4, 8}. XGBoost – learning rate $[10^{-3}, 0.3]$ log-uniform, max depth from {3, 4, 5, 6, 8, 10}, subsample and column subsample on [0.5, 1.0], L1 and L2 on $[10^{-3}, 10]$ log-uniform. SVR – C on $[10^{-2}, 10^3]$, ϵ on $[10^{-3}, 1.0]$, γ on $[10^{-4}, 10]$, all log-uniform. Ridge – λ on $[10^{-3}, 10^3]$ log-uniform.

This nested loop evaluates each model in the outer fold at the configuration that was best on that fold's training data, not at a default that might favour one algorithm. Any advantage then reflects modelling capability rather than tuning [8].

3.7. Statistical comparison

The per-fold R^2 scores form a $k \times m$ matrix S , with $k = 20$ folds and $m = 5$ models; S_{ij} is model j 's R^2 on fold i . Inference runs in two stages after Demšar [4].

Stage 1 – Friedman test. The null is equal expected rank across folds. Within each fold the models are ranked by R^2 (rank 1 best, ties averaged); model j 's mean rank is R_j . The statistic is equation (5):

$$\chi_F^2 = \frac{12k}{m(m+1)} \sum_{j=1}^m \left(\bar{R}_j - \frac{m+1}{2} \right)^2 \quad (5)$$

where k is the number of folds, m the number of models, and R_j the mean rank of model j [4]. Under the null it is asymptotically chi-squared with $m - 1$ degrees of freedom, rejected at $\alpha = 0.05$ if it exceeds the critical value.

Stage 2 – post-hoc. On rejection, pairs are compared two ways. The Nemenyi test treats all $m(m - 1)/2$ pairs at once: two models differ if their mean-rank gap exceeds the critical difference (CD) of equation (6):

$$CD = q_\alpha \cdot \sqrt{\frac{m(m+1)}{6k}} \quad (6)$$

where q_α is the studentised-range critical value divided by $\sqrt{2}$ at level α for m groups [4]. Nemenyi holds the family-wise error at α and is conservative. The Wilcoxon signed-rank test is also applied per pair on the twenty fold-wise R^2 differences, with raw p-values corrected by Holm's step-down rule [3,4]. Wilcoxon–Holm has more power per pair than Nemenyi while still controlling the family-wise error. Both run in Python with SciPy and scikit-posthocs.

4. Results

4.1. Cross-validated performance

GroupKFold at $k = 20$ produced a 20×5 matrix of per-fold R^2 and a matching RMSE matrix, shown in Table 1 and Table 2. Aggregate statistics – mean, standard deviation, minimum and maximum over the twenty folds – appear in Table 3. All runs used the same fold partition, the same pipeline and independently tuned Optuna hyperparameters per fold.

Table 1

Per-fold R^2 (20 folds $\times$ 5 models). HGB = HistGradientBoosting; RF = Random Forest; XGB = XGBoost. Negative values fall below the constant-mean baseline

Fold	HGB	RF	XGB	SVR	Ridge
1	0.1250	0.3560	0.4093	0.3837	-0.0625
2	0.9570	0.9276	0.9588	0.7055	0.1706
3	-5.7159	0.8498	0.7932	0.4998	-0.3515
4	0.5181	0.5518	0.6152	0.1099	0.4609
5	0.9316	0.9352	0.8612	0.9143	0.8553

Fold	HGB	RF	XGB	SVR	Ridge
6	0.9934	0.9717	0.9940	0.9468	0.9428
7	0.7362	0.9113	0.9267	0.8707	0.7969
8	0.4527	0.5733	0.1595	0.6313	0.7436
9	0.8286	0.8148	0.8653	0.7545	0.6640
10	0.8789	0.8262	0.8903	0.8242	0.7360
11	0.9778	0.9372	0.9341	0.9560	0.9173
12	0.9296	0.8718	0.9126	0.9176	0.7898
13	0.9112	0.8985	0.9195	0.8846	0.8223
14	0.3147	0.5538	0.6260	0.6112	0.6261
15	0.5454	0.1030	0.5594	0.7275	0.6754
16	0.8160	0.8642	0.9064	0.8168	0.7398
17	0.9881	0.9928	0.9931	0.2031	0.9596
18	0.9189	0.8609	0.9016	0.8812	0.7817
19	0.5571	0.7695	0.6600	0.7858	0.6178
20	0.9123	0.9666	0.9135	0.8577	0.8568

Table 2

Per-fold *RMSE* (20 folds $\times$ 5 models), in KT10valueMax units

Fold	HGB	RF	XGB	SVR	Ridge
1	26.14	22.42	21.48	21.94	28.80
2	4.00	5.19	3.91	10.46	17.55
3	42.74	6.39	7.50	11.67	19.18
4	15.77	15.21	14.09	21.43	16.68
5	8.57	8.34	12.21	9.59	12.46
6	17.39	36.09	16.67	49.49	51.33
7	5.83	3.38	3.07	4.08	5.11
8	9.75	8.61	12.08	8.00	6.67
9	6.71	6.97	5.95	8.03	9.39
10	7.58	9.08	7.21	9.13	11.19

Fold	HGB	RF	XGB	SVR	Ridge
11	4.63	7.79	7.98	6.52	8.94
12	4.06	5.49	4.53	4.40	7.02
13	7.13	7.63	6.79	8.13	10.09
14	8.43	6.80	6.22	6.35	6.22
15	7.81	10.97	7.69	6.05	6.60
16	8.41	7.23	6.00	8.39	10.00
17	27.48	21.29	20.96	224.49	50.54
18	4.81	6.30	5.30	5.83	7.90
19	10.84	7.82	9.50	7.54	10.07
20	7.28	4.49	7.23	9.27	9.30

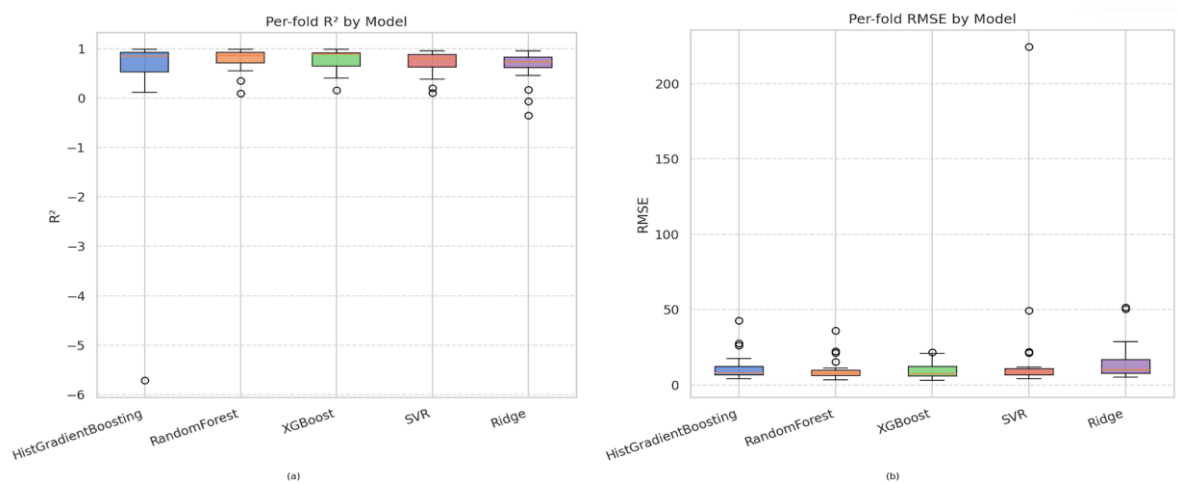


Figure 1: Distribution of per-fold R^2 (a) and RMSE (b) across the 20 GroupKFold folds for each model.

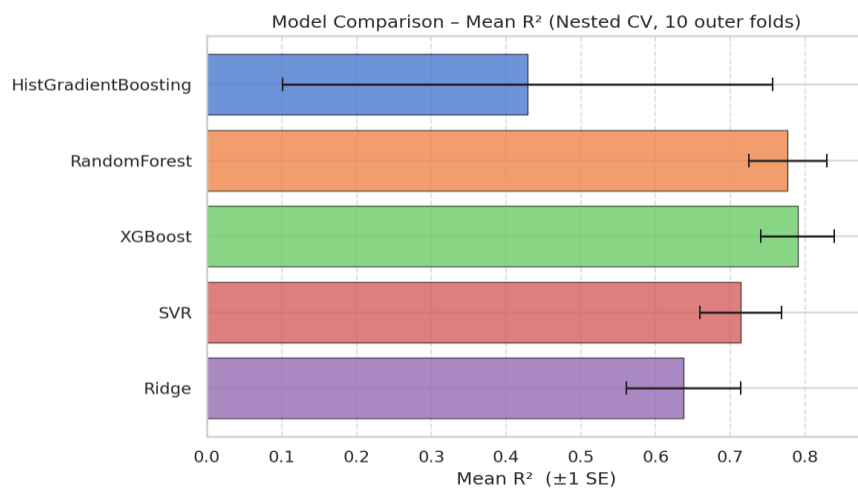


Figure 2: Mean R^2 with ± 1 standard error per model over the 20 folds with nested Optuna tuning.

Table 3

Aggregate performance over 20 folds, ordered by decreasing mean R^2

Model	Mean R^2	Std R^2	Min R^2	Max R^2	Mean RMSE	Std RMSE
XGBoost	0.7900	0.2138	0.1595	0.9940	9.32	5.19
Random Forest	0.7768	0.2273	0.1030	0.9928	10.37	7.68
SVR	0.7141	0.2371	0.1099	0.9560	22.04	47.48
Ridge	0.6371	0.3336	-0.3515	0.9596	15.25	13.08
HistGradientBoosting	0.4288	1.4306	-5.7159	0.9934	11.77	9.63

XGBoost led with mean $R^2 = 0.7900$ ($\sigma = 0.2138$); Random Forest followed at 0.7768 ($\sigma = 0.2273$). Their mean RMSE, 9.32 and 10.37, were the lowest once anomalous folds are set aside. SVR placed third by mean R^2 (0.7141) but carried a mean RMSE of 22.04 and $\sigma = 47.48$, driven by one collapsing fold (Section 4.2). Ridge reached mean $R^2 = 0.6371$, evidence of a strong linear component, but its $\sigma = 0.3336$ and negative minimum (-0.352) show sensitivity to distribution shift across batches. HGB had the lowest mean R^2 (0.4288) and by far the largest σ (1.4306) because of a single extreme failure (Section 4.2).

4.2. Per-fold anomalies

Two folds failed in ways worth recording rather than hiding. In Fold 3, HGB gave $R^2 = -5.716$ and $RMSE = 42.74$, far outside its range elsewhere (R^2 0.125 to 0.993). The training batches of that fold did not represent the held-out batch's feature-quality relation, and HGB, accumulating pseudo-residuals, is especially exposed to such a mismatch; its predictions there beat neither RF nor XGBoost under the same partition. The negative R^2 means it did worse than predicting the training mean.

In Fold 17, SVR gave $R^2 = 0.203$ and $RMSE = 224.49$, the largest single error in the experiment. That fold's test batches reach the upper tail of KT10valueMax. The RBF kernel confines predictions near the convex hull of the training support; once a test fragment lies outside it, the kernel weights collapse and the prediction reverts toward the training mean. This extrapolation failure is what inflates SVR's mean RMSE (22.04) and σ (47.48), so those aggregates do not represent its behaviour on the folds without extreme batches.

4.3. Statistical comparison

4.3.1. Friedman test

Per-fold R^2 was ranked within each fold (rank 1 = best). Mean ranks over the twenty folds were XGBoost 2.00, Random Forest 2.70, HistGradientBoosting 3.05, SVR 3.10, Ridge 4.20, consistent with the R^2 ordering of Table 3. The Friedman statistic of equation (5) was $\chi^2(4) = 20.32$, $p = 0.000432$ – significant at $\alpha = 0.05$ and strong evidence that the models are not equivalent in rank. Post-hoc tests followed.

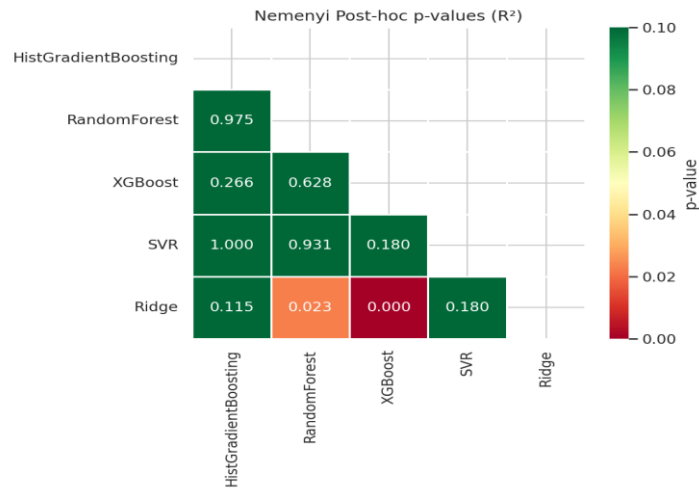
4.3.2. Nemenyi post-hoc test

Nemenyi was applied to all ten pairs (Table 4). Two fell below 0.05: XGBoost vs Ridge ($p = 0.0001$) and Random Forest vs Ridge ($p = 0.0227$). The rest, including XGBoost vs Random Forest ($p = 0.628$) and XGBoost vs SVR ($p = 0.180$), did not. Ridge is thus separable from both ensemble leaders, which are not separable from each other.

Table 4

Nemenyi pairwise p-values (lower triangle). * significant at $\alpha = 0.05$

	HGB	RF	XGB	SVR	Ridge
HGB	–				
RF	0.9751	–			
XGB	0.2659	0.6277	–		
SVR	0.9996	0.9307	0.1796	–	
Ridge	0.1152	0.0227*	0.0001*	0.1796	–

Figure 3: Nemenyi pairwise p-values after rejection of the Friedman null. Cells towards red mark significant differences ($p < 0.05$).

4.3.3. Wilcoxon signed-rank test with Holm correction

Wilcoxon was applied per pair on the twenty fold-wise R^2 differences (Table 5). One pair survived Holm at $\alpha = 0.05$: XGBoost vs Ridge ($p_{\text{raw}} = 0.0027$, $p_{\text{holm}} = 0.0271$). Two more approached it: Random Forest vs Ridge (0.0107, 0.0962) and XGBoost vs SVR (0.0973, 0.584).

Table 5

Wilcoxon signed-rank test with Holm correction, all pairs ($n = 20$). Rejection at $\alpha = 0.05$ in the last column

Pair	p_{raw}	p_{holm}	Reject H_0
HGB – RF	0.3683	1.0000	No
HGB – XGB	0.0825	0.5779	No
HGB – SVR	0.9854	1.0000	No
HGB – Ridge	0.3488	1.0000	No
RF – XGB	0.3683	1.0000	No

Pair	praw	pholm	Reject H_0
RF – SVR	0.2611	1.0000	No
RF – Ridge	0.0107	0.0962	No
XGB – SVR	0.0973	0.5838	No
XGB – Ridge	0.0027	0.0271	Yes
SVR – Ridge	0.0441	0.3524	No

The XGBoost–Ridge rejection is the informative one. At $n = 10$ the smallest achievable Wilcoxon p-value is about 0.002, and after the Holm factor of $m = 10$ for the most extreme pair the corrected value sits near 0.020, just under α . At $n = 20$ the discrete p-value grid is finer, which lets Holm clear the threshold for this pair. Raising the outer fold count from 10 to 20 therefore adds power to Wilcoxon–Holm without loosening the family-wise guarantee [3,4]. Friedman, Nemenyi and Wilcoxon–Holm agree: XGBoost and Random Forest beat Ridge, and XGBoost vs Ridge is the most robustly separable pair.

5. Discussion

5.1. Interpreting the results

XGBoost and Random Forest leading across the twenty folds fits the known strength of tree ensembles on tabular regression with interacting features [6,7]. SensorValueRAW depends on fragment geometry, coil geometry and mineral content non-additively: the response varies with the ratio of volume to UsedCoilArea, with the proximity term CenterDeviation, and with the thermal drift in SensorTemp. Trees capture these interactions through recursive splits without engineered terms, which is why they beat the linear Ridge baseline where several features co-vary across batches [6,7]. XGBoost’s edge over Random Forest, though not significant under Nemenyi ($p = 0.628$), is consistent with regularisation: its explicit L1 and L2 penalties on leaf weights curb overfitting to a single training batch, whereas RF relies on averaging alone [6,7]. In folds with high-KT10 batches (Fold 6 reaches 624.0), XGBoost holds $R^2 = 0.994$ against RF’s 0.972, so the regularisation pays off most at the extremes of the target. Ridge’s mean R^2 of 0.637 confirms a sizeable linear component. It clears $R^2 > 0.60$ in fourteen folds and reaches 0.960 in Fold 17 – the same fold that breaks SVR. Linear extrapolation beyond the training range explains this, where the RBF kernel instead interpolates within the support and falls back to the mean outside it [10]. Ridge’s negative R^2 in Fold 3 (−0.352) shows the linear form failing where non-linear geometry–magnetism interaction dominates and the training data miss the relevant region. HGB’s collapse in Fold 3 ($R^2 = -5.716$) deserves a note. It is competitive in the other nineteen folds and ranks first in Folds 6, 11, 12 and 18. The collapse is an outlier tied to that fold’s batch mix, not a standing weakness. Boosting minimises training residuals in sequence, and when the test set sits in a low-density region the learned mapping can extrapolate badly – more so than RF, which averages independent trees and tolerates a missing region in any one bootstrap [7,12]. RF’s $R^2 = 0.850$ on the same partition confirms the fold is predictable; HGB’s sequential mode simply misassigned residual weight.

5.2. Comparison with literature

Tree ensembles beating kernel and linear methods here matches broader tabular results in industrial and mineral-processing settings [2,5,6,7]. Ore-grade and quality studies on similar sensor data report RF R^2 of 0.70–0.95 for complex multi-sensor inputs, in line with the 0.777 seen here [2,5]. XGBoost’s

edge over RF agrees with benchmarks where its regularisation and second-order approximation help on medium-size tabular data dominated by variance [6].

The comparison follows the two-stage protocol of Demšar [4] as extended by Rainio et al. [3]. A practical point emerges: at $k = 20$, Wilcoxon–Holm finds one significant pair (XGBoost vs Ridge), whereas at $k = 10$ it rejects none, because the achievable p-values are too coarse [3,4]. For structured industrial benchmarking, the choice between 10 and 20 outer folds directly affects post-hoc power, separately from any change in the accuracy of the performance estimate.

5.3. Limitations

Four limitations stand out. First, Volume and Area come from a single top-down scan, so they approximate rather than measure the true geometry; for fragments with significant underside relief they underestimate, which can blunt the geometric features and complicate any interpretation that leans on them.

Second, although 4,161 measurements across 90 scans of 45 ore pieces suffice for the nested scheme, an average test fold holds about 200 measurements from four or five scans. Per-fold R^2 then depends on the target spread within those scans; a fold with one high-KT10 piece (Folds 3 and 17) is more variable than a balanced one. Stratifying scan assignment by mean KT10 could reduce this. A further limitation is that the two faces of a piece are correlated, so scan-level grouping is less strict than piece- or batch-level grouping; piece- or batch-level holdout would give a more conservative estimate and is left to future work.

Third, no feature- or permutation-importance analysis is included, which would show which sensors drive accuracy in good folds and add variance in bad ones, and would test whether the approximate geometric features add anything beyond the magnetic ones. Fourth, the Optuna budget is fixed at 100 trials per model per fold; the SVR space (C , ϵ , γ over four log decades) may need more given its tuning sensitivity.

6. Conclusions

Five supervised regression models – XGBoost, Random Forest, SVR, Ridge and HistGradientBoosting – were compared for predicting the magnetic quality indicator KT10valueMax from seven sensor features on an industrial conveyor-line sorter. GroupKFold at $k = 20$ kept performance estimates free of batch leakage, and Optuna tuning with 100 trials per model per fold kept the comparison fair.

The main findings are:

- XGBoost reached the highest mean R^2 (0.790) and the lowest mean $RMSE$ among the leaders (9.32), with the best mean rank (2.00) and the lead in 12 of 20 folds;
- Random Forest followed closely (mean $R^2 = 0.777$, rank 2.70) as a robust bagging alternative;
- Ridge reached mean $R^2 = 0.637$, confirming a substantial linear component but trailing both ensembles statistically and practically;
- SVR placed third by mean R^2 (0.714) but produced an $RMSE$ outlier in Fold 17 (224.49) from RBF extrapolation failure on high-KT10 batches;
- HistGradientBoosting failed in Fold 3 ($R^2 = -5.716$), which made its mean unreliable, though it was competitive in the other nineteen folds.

The Friedman test rejected equal rank ($\chi^2(4) = 20.32$, $p = 0.000432$). Nemenyi flagged XGBoost vs Ridge ($p = 0.0001$) and Random Forest vs Ridge ($p = 0.0227$); Wilcoxon–Holm, which is weak at $k =$

10, rejected XGBoost vs Ridge at $k = 20$ ($p_{\text{holm}} = 0.0271$), showing that more outer folds raise the power of family-wise correction without breaking its guarantee.

XGBoost is recommended for deployment in this context, with Random Forest as a reliable, interpretable alternative. Both beat Ridge with statistical significance, and their advantage over SVR and HGB is practically meaningful in accuracy and consistency across diverse batches. Future work should decompose feature importance to weigh geometric against electromagnetic inputs, test stacking of XGBoost and RF, and extend the data to further ore types to check whether this ranking holds.

Declaration on Generative AI

During the preparation of this work, the author(s) used Anthropic Claude Opus in order to: Grammar and spelling check, Paraphrase and reword, Improve writing style, Formatting the document. After using these tool(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

- [1] D. Peukert, C. Xu, P. Dowd, A review of sensor-based sorting in mineral processing: the potential benefits of sensor fusion, *Minerals* 12 (2022) 1364. doi:10.3390/min12111364.
- [2] S. O. Adewuyi, A. Anani, K. Luxbacher, S. Ndlovu, From single-sensor constraints to multisensor integration: advancing sustainable complex ore sorting, *Minerals* 15 (2025) 1101. doi:10.3390/min15111101.
- [3] O. Rainio, J. Teuho, R. Klén, Evaluation metrics and statistical tests for machine learning, *Scientific Reports* 14 (2024) 6086. doi:10.1038/s41598-024-56706-x.
- [4] J. Demšar, Statistical comparisons of classifiers over multiple data sets, *Journal of Machine Learning Research* 7 (2006) 1–30.
- [5] X. Luo, K. He, Y. Zhang, P. He, Y. Zhang, A review of intelligent ore sorting technology and equipment development, *International Journal of Minerals, Metallurgy and Materials* 29 (2022) 1647–1655. doi:10.1007/s12613-022-2477-5.
- [6] T. Chen, C. Guestrin, XGBoost: a scalable tree boosting system, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, ACM, San Francisco, CA, 2016, pp. 785–794. doi:10.1145/2939672.2939785.
- [7] L. Breiman, Random forests, *Machine Learning* 45 (2001) 5–32. doi:10.1023/A:1010933404324.
- [8] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: a next-generation hyperparameter optimization framework, in: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '19*, ACM, Anchorage, AK, 2019, pp. 2623–2631. doi:10.1145/3292500.3330701.
- [9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: machine learning in Python, *Journal of Machine Learning Research* 12 (2011) 2825–2830.
- [10] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, 1995.
- [11] A. E. Hoerl, R. W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, *Technometrics* 12 (1970) 55–67. doi:10.1080/00401706.1970.10488634.
- [12] J. H. Friedman, Greedy function approximation: a gradient boosting machine, *Annals of Statistics* 29 (2001) 1189–1232. doi:10.1214/aos/1013203451.