
Research article

Adaptive meta-learning for lithofacies classification from composite well logs using a gradient boosted stacking ensemble

Paolo Dell'Aversana*

Independent Researcher, Italy

* **Correspondence:** Email: paolodellaversa2024@virgilio.it; Tel: +39 3400736394.

Abstract: Accurate lithofacies classification from composite well logs is essential for subsurface characterization and reservoir modeling. In this work, we present a machine learning stacking ensemble empowered by a gradient boosting meta-learner to improve predictive performance and robustness in well log interpretation. The framework integrates multiple heterogeneous base learners, including random forest, gradient boosting, support vector machine, and K-nearest neighbors, whose probabilistic outputs are combined by a gradient boosting meta-learner trained on out-of-fold predictions. Feature standardization and base-learner training were performed within each training fold to avoid information leakage. This architecture allows the meta-learner to capture nonlinear interactions among base models and correct systematic errors that static ensemble methods, such as majority voting, cannot address. We evaluated the approach on a real dataset of composite well logs comprising seven log types and approximately 3000 samples. Comparative experiments demonstrate that the proposed stacking ensemble consistently outperforms individual classifiers and majority voting, achieving higher accuracy, balanced F1-scores, and improved generalization. The results highlight the effectiveness of adaptive meta-learning for lithofacies prediction, providing a robust and reproducible methodology for complex, noisy, and heterogeneous well log datasets.

Keywords: composite well logs; machine meta-learning; automatic classification; litho-fluid facies; adaptive learning; gradient boosting

1. Introduction

Over the past two decades, in parallel with developments in other fields, the adoption of machine

learning (ML) in geosciences has increased significantly [1–4]. Representative applications include rock and seismic facies recognition [5,6], automated seismic interpretation [7], and, more recently, the interpretation and classification of well log data, particularly for lithofacies prediction [8–13].

Early research in this domain primarily focused on the application and optimization of single-classifier approaches to overcome the nonlinearity of petrophysical responses. For instance, Bestagini et al. [8] demonstrated the significant potential of gradient boosting decision trees, showing that boosting-based iterative learning is particularly effective at achieving high accuracy even when training datasets are relatively small or constrained by limited well coverage.

Furthering this comparative line of inquiry, Dell'Aversana [9] provided a systematic evaluation of diverse standalone machine learning (ML) algorithms. The author investigated the effectiveness of six supervised machine learning algorithms for lithofacies classification, establishing a five-step workflow ranging from data analysis to performance evaluation via cross-validation. The study introduces a “systemic approach” using parallel, integrated algorithms, demonstrating that this method provides robust, consistent results suitable for real-time well-site geological operations.

Tewari and Dwivedi [10] presented a comparative study highlighting the effectiveness of heterogeneous ensemble methods. Particularly, stacking frameworks outperform individual machine learning models in identifying geological lithofacies. Utilizing a field dataset, the research demonstrated that combining diverse base learners (SVM, random forests, and neural networks) with a meta-learner significantly improves predictive accuracy and mitigates individual model biases. The authors implemented two types of heterogeneous ensemble approaches—stacking and voting—and compared their performance against several widely used machine learning classifiers, including support vector machines (SVM), multilayer perceptrons (MLP), gradient boosting (GB), and random forests (RF). These models were trained and evaluated using well log data from Kansas oil fields, with careful preprocessing steps such as normalization, noise filtering, feature selection, and parameter tuning through grid search and cross-validation. The results showed that heterogeneous ensemble methods outperform individual classifiers in lithofacies classification tasks. In particular, the stacking ensemble, which combines the predictions of multiple base learners through a meta-classifier, achieves the highest overall accuracy and demonstrates strong robustness even under challenging conditions such as class imbalance. The study reports that stacking improves predictive performance compared to traditional models and provides more generalized and reliable results across different lithofacies.

Data complexity and heterogeneity, together with the intrinsic dependence on context, motivate the use of ensemble learning strategies capable of integrating complementary predictive behavior from multiple base learners [14,15]. Ensemble learning has therefore become a central paradigm in well log analysis, as it reduces variance, mitigates individual model biases, and improves robustness in the presence of noisy or overlapping petrophysical responses. Simple ensemble approaches, such as hard majority voting, are commonly employed as reference baselines for lithofacies classification [14]. However, voting-based schemes implicitly assume equal reliability among base learners and do not account for systematic error patterns or differences in predictive confidence.

More advanced ensemble architectures overcome these limitations by introducing a meta-learning layer that combines the probabilistic outputs of heterogeneous base learners. Meta-learning, or “learning to learn” [16,17], refers to methods where a model learns how to optimize, adapt, or combine other models rather than directly mapping inputs to outputs. In ensemble learning, for example, a meta-learner (such as gradient boosting) is trained on the predictions of base learners to correct systematic errors, capture nonlinear interactions, and adaptively weight model outputs. Meta-learning is

particularly useful for complex, heterogeneous, or noisy datasets, as it improves generalization, reduces overfitting, and enables models to leverage complementary strengths of multiple learners. In essence, meta-learning allows a system to learn how to learn, enhancing predictive performance beyond individual models.

In particular, stacking [18,19] is an ensemble meta-learning technique where multiple models are combined in a hierarchical way rather than by simple averaging or voting. First, several independent models are trained on the same dataset. These models form the base layer (layer 0), and each one produces its own prediction. They work in parallel and do not interact with each other. Although sometimes called weak learners, they are not necessarily weak, just diverse. Next, a meta-model (layer 1) is trained using the predictions of the base models as its input features. Instead of learning directly from the original data, this final model learns how to best combine the base models' outputs to produce the final prediction. In the case of well logs classification, in stacking ensembles, each sample is represented in a transformed feature space defined by base-model class probabilities, optionally augmented by the original standardized well log measurements. This representation enables the meta-learner to capture nonlinear interactions among base learners and to correct correlated errors that cannot be addressed through static aggregation rules. In other words, stacking ensembles combine multiple heterogeneous classifiers through a meta-learning stage that learns how to optimally integrate their predictions. Previous studies have demonstrated that stacking approaches often outperform individual classifiers and simple aggregation rules such as majority voting, especially in complex classification problems involving noisy and heterogeneous geoscientific data [10,15–18]. These approaches allow the meta-learner to capture systematic error patterns produced by the base learners and to construct context-dependent decision rules.

Within this framework, gradient boosting represents a particularly effective choice for meta-learning [20]. Gradient boosting iteratively fits weak learners to the residual errors of the current model, learning how base learners fail rather than simply averaging their predictions. When used as a meta-learner, gradient boosting operates on out-of-fold probabilistic predictions of the base learners together with the original features, enabling the construction of context-aware and condition-dependent decision rules. This property is especially relevant for composite well logs [21], where lithological discrimination often requires the joint interpretation of physical measurements and model uncertainty patterns.

Building upon these considerations, this paper addresses the problem of automatic lithofacies classification from composite well logs using a machine learning stacking ensemble empowered by a gradient boosting meta-learner. We compare our integrated approach with a baseline majority voting ensemble, as well as with other machine learning models and with deep learning techniques. This design allows a direct and quantitative assessment of the added value of meta-learning with respect to static ensemble aggregation and other ML approaches.

The next section describes our methodology in detail. The construction of the base learners and the hard majority voting ensemble is presented as the main reference benchmark. Then, the formulation and role of the gradient boosting meta-learner are described, emphasizing its corrective and context-aware behavior within the stacking framework. Finally, in the applicative part of the paper, the complete stacking ensemble architecture, including cross-validated training and evaluation metrics, is applied to real composite well log data.

2. Methodology

The proposed methodology employs a set of classical machine-learning classifiers integrated through two ensemble strategies: (1) a hard majority-voting scheme used as a baseline reference model and (2) a stacking ensemble in which the predictions of the base learners are combined by a gradient boosting meta-learner. The objective of this architecture is to exploit the complementary predictive behavior of heterogeneous classifiers and to allow the meta-learner to model systematic errors produced by the individual models.

2.1. Majority voting ensemble

As a baseline ensemble method, a hard majority voting classifier was constructed. Hard majority voting is a widely used ensemble aggregation strategy in machine learning, where the final class prediction corresponds to the most frequent prediction among the base learners [15,22]. In this scheme, each base learner produces an independent class prediction for a given input sample. The final predicted class is determined by selecting the most frequent prediction among the base learners. Formally, the predicted class for sample x_i is defined as

$$\hat{y}_i = \text{mode}\{\hat{y}_{i1}, \hat{y}_{i2}, \dots, \hat{y}_{iN}\} \quad (1)$$

where $\hat{y}_{ij} = B_j(x_i)$ represents the class predicted by the j -th base learner B_j , N denotes the number of classifiers in the ensemble, and “mode” denotes the “statistical mode”, i.e., the class label predicted most frequently among the base learners.

Four classical machine-learning classifiers were employed as base learners (see Appendix 1 for details on hyperparameter settings):

- Random forest (RF): an ensemble of decision trees trained using bootstrap aggregation and random feature selection [22,23].
- Gradient boosting (GB): a boosting algorithm that sequentially builds weak learners by minimizing the residual prediction error [24,25].
- Support vector machine (SVM): a kernel-based classifier using a radial basis function (RBF) kernel and probabilistic output calibration [26].
- k-nearest neighbors (KNN): an instance-based classifier using $k = 6$ nearest neighbors [27].

Although the final majority voting decision is based on discrete class labels, each base learner can also produce probabilistic estimates of class membership. For sample, x_i , the probability predicted by classifier B_j for class c can be expressed as

$$\hat{p}_{ij}^{(c)} = P(y = c \mid x_i, B_j) \quad (2)$$

where $\hat{p}_{ij}^{(c)}$ represents the estimated probability that sample x_i belongs to class c .

The majority voting ensemble serves as a baseline model for comparison with the more advanced stacking ensemble introduced in the following section.

We remark that gradient boosting was adopted both as a base learner and as a meta-learner within the stacking framework to exploit its strong capability to model complex, nonlinear relationships while maintaining flexibility across different learning levels. At the base layer, gradient

boosting models were employed to capture high-order interactions and heterogeneous patterns directly from the input feature space, benefiting from their robustness to noise and their ability to handle mixed data distributions. At the meta-learning stage, gradient boosting was used to learn an optimal, nonlinear combination of the base learners' predictions. In this role, the model does not re-learn the original classification problem but instead identifies complementary behaviors, systematic biases, and context-dependent strengths among the base learners, thereby enhancing the overall predictive performance of the ensemble. The effectiveness of this choice is further supported by appropriate regularization of the meta-learner and by the use of out-of-fold predictions during training, which mitigate overfitting and ensure that the meta-model generalizes beyond the training data. This dual use of gradient boosting allows the stacking architecture to benefit from both powerful local decision rules at the base level and a flexible yet controlled aggregation strategy at the ensemble level.

Although the selection of base learners is constrained, it can be broadened to include a wide range of machine learning techniques, such as decision trees, naïve Bayes classifiers, and different neural network architectures [28–38]. Early structural approaches include the formalization of decision tree induction algorithms by Quinlan [28], which provided a transparent framework for rule-based learning. Comprehensive theoretical and practical frameworks for general pattern classification systems were later compiled by Duda et al. [29]. Within probabilistic modeling, the theoretical boundaries and remarkable optimality of the simple Bayesian classifier under zero-one loss functions were rigorously investigated by Domingos and Pazzani [30], while Zhang [31] further expanded on the theoretical conditions under which the Naive Bayes approach maintains optimal performance. To improve predictive accuracy by combining weak baselines, ensemble methods emerged prominently through the adaptive online learning and boosting frameworks developed by Freund and Schapire [32], which built directly upon the theoretical foundations of weak learnability established earlier by Schapire [33]. Concurrently, a major paradigm shift in statistical modeling occurred through Vapnik's [34] comprehensive statistical learning theory, which subsequently enabled Boser et al. [35] to formulate training algorithms for optimal margin classifiers and Support Vector Machines. These methodologies find deep roots in the classical behavior of sequential predictors analyzed by Cover [36] and the comprehensive probabilistic principles of pattern recognition formalized by Devroye et al. [37]. Finally, addressing the degradation and optimization challenges of very deep networks, He et al. introduced deep residual learning (ResNets), enabling robust feature representation and high-accuracy image recognition [38].

2.2. Gradient boosting as a powerful meta-learner

Gradient boosting, when used as a meta-learner in a stacking ensemble, plays a very specific and rather elegant role. It is not just “another classifier on top”, but a mechanism that *learns how the base models fail* and how their errors interact. At the meta-learning level, each sample is no longer represented only by physical or geological features. Instead, it is described by a richer vector that includes:

- the original standardized features, and
- the class probabilities predicted by each base learner.

This creates a new feature space where each dimension encodes *model beliefs* rather than raw measurements. The gradient boosting meta-learner operates entirely in this space. It is an additive, stage-wise optimization method. The meta-model $M(x)$ is constructed as a sum of base-learners:

$$M(x) = \sum_{t=1}^T \gamma_t h_t(x) \quad (3)$$

where $h_t(x)$ are weak learners, and γ_t are their weights.

The key idea is that each new learner is trained to correct the residual errors of the ensemble built so far. Instead of computing gradients with respect to parameters, gradient boosting computes gradients with respect to the model function itself. At iteration t , the residuals are defined as:

$$r_i^{(t)} = -\frac{\partial \mathcal{L}(y_i, M(x_i))}{\partial M(x_i)} \Big|_{M=M^{(t-1)}} \quad (4)$$

where $\mathcal{L}$ is the classification loss function (typically log-loss for probabilistic classification). In the stacking context, this means the meta-learner does not simply average or vote over base models. Instead, it learns rules such as:

- “Trust the random forest for class A unless the SVM strongly disagrees.”
- “When KNN and SVM both assign high probability to class B, override the random forest.”
- “Correct systematic confusion between two lithologies when all base models show similar uncertainty patterns.”

These interactions are inherently nonlinear, and gradient boosting is particularly effective at modeling them. An important detail in our implementation is that the meta-learner sees not only base-model predictions but also the original features. As a result, the gradient boosting meta-learner can learn hybrid rules of the form:

$$\text{If } (\log \text{ value} > \theta) \wedge (P_{\text{RF}}(\text{class } c) < 0.4) \Rightarrow \text{class } d \quad (5)$$

This is crucial for composite logs, where geological context often disambiguates cases in which statistical classifiers alone are uncertain.

Another key aspect is the cross-validated training of the stacking layer. The meta-learner is trained on out-of-fold predictions of the base learners, preventing information leakage and ensuring that the meta-model learns genuine generalization patterns rather than memorizing training labels. Conceptually, we can think of gradient boosting as a *cognitive referee*. Majority voting counts hands raised. Gradient boosting listens to *how confidently* each *expert* speaks, remembers past mistakes, and gradually learns when to overrule consensus. This is why, in heterogeneous and noisy domains such as composite well logs, a gradient boosting meta-learner often outperforms majority voting: it is not democratic; it is corrective, adaptive, and context aware. In summary, stacking with gradient boosting turns an ensemble into a “learning system about learning systems”.

2.3. Stacking ensemble with a gradient boosting meta-learner

To improve predictive performance beyond simple voting, a stacking ensemble was implemented. In stacking, the outputs of base learners are combined as inputs to the meta-learner, which produces the final prediction. Formally, for a sample x_i , the meta-learner input vector is:

$$z_i = \text{concat}(x_i, \hat{p}_{i1}, \hat{p}_{i2}, \dots, \hat{p}_{iN}) \quad (6)$$

where $\hat{p}_{ij}$ are the class probabilities predicted by base learner B_j . As explained earlier, the gradient boosting classifier is used as the meta-learner to model nonlinear interactions between base learner outputs and the original features. The final prediction is:

$$\hat{y}_i^{\text{stacking}} = M(z_i) \quad (7)$$

where M represents the gradient boosting meta-learner. The stacking classifier is trained using 5-fold cross-validation to reduce overfitting and improve generalization.

To ensure maximum robustness and prevent data leakage, the stacking ensemble was implemented using a pipeline-based architecture. Each base learner (RF, GB, SVM, and k-NN) was wrapped in a dedicated pipeline that includes an independent StandardScaler. This configuration ensures that during the 5-fold cross-validation process, the feature scaling (mean and variance) is computed exclusively on the training folds and subsequently applied to the validation folds. Following Equation 6, the meta-learner was trained using the “passthrough method”, which concatenates these out-of-fold probabilistic predictions with the locally standardized original features.

2.4. Evaluation metrics

The performance of each method is evaluated on the test set using standard metrics. The first is accuracy:

- $$\text{Accuracy} = \frac{1}{M} \sum_{i=1}^M 1\{\hat{y}_i = y_i\}$$

where:

M is the total number of samples in the test set.

y_i is the true class label of the i -th sample.

$\hat{y}_i$ is the predicted class label produced by the classifier for the i -th sample.

$1\{\hat{y}_i = y_i\}$ is the indicator function, which equals:

- 1 if the prediction is correct ($\hat{y}_i = y_i$),
- 0 otherwise.

Therefore, the summation counts the number of correct predictions, and dividing by M yields the fraction of correctly classified samples, i.e., the overall accuracy of the model on the test set.

Additional metrics include:

- Weighted F1-score: this accounts for class imbalance by weighting F1-scores by class support.
- Macro F1-score: this computes the F1-score independently for each class and averages them.
- Balanced accuracy: this is the average of the recall obtained on each class.

3. Application to well log analysis and classification

3.1. Data

To validate the stacking ensemble empowered by gradient boosting for lithofacies classification for classifying composite well log data, we analyzed a dataset comprising almost 3000 samples for seven logs.

The dataset consists of high-quality petrophysical measurements acquired from a vertical exploration well located in a prolific offshore hydrocarbon basin. The data corresponds to a complex sequence of Middle Jurassic to Upper Triassic age, characterized by a structural setting involving faulted closures within a major regional anticline.

The target interval represents a heterogeneous reservoir sequence where the depositional environment transitioned from fluvial channels to marginal marine settings with significant fluvial influence. From a lithological perspective, the section comprises:

- Reservoir units: predominantly quartzose sandstones, ranging from fine to coarse-grained, with variable calcareous and silicic cementation.
- Non-reservoir units: interbedded carbonaceous claystones, shaly siltstones, and occasional limestone stringers.
- Fluid content: the interval includes both brine-saturated zones and hydrocarbon-bearing sandstones, providing a diverse range of petrophysical responses for the classification task.

For additional geological details, we included a schematic lithofacies definition framework in Table A2, in Appendix 2.

Measurements include sonic, deep resistivity (Rdep), density (DEN), neutron porosity (NEU), photoelectric absorption (PEF), gamma ray (GR), and spontaneous potential (SP) logs. As an illustrative example, Figure 1 presents the resistivity log, where hydrocarbon-bearing intervals are clearly identifiable, at depths between 1100 and 1150 m by their elevated resistivity values. Figure 2 shows the log correlation matrix. It summarizes the pairwise statistical correlations between different well-log measurements over the same depth interval, quantifying how strongly logs vary together (positively, negatively, or weakly). It is useful because it provides rapid insight into petrophysical relationships, helps identify redundant or complementary logs, supports feature selection for machine-learning and inversion workflows, assists in quality control and anomaly detection, and evaluates whether the available logs offer sufficient independent information for effective lithofacies discrimination. Figure 3 shows three examples of composite log cross plots. These are scatter plots that display the relationship between two well log measurements recorded within the same depth intervals, often after normalization or scaling. Each point represents a data sample, and additional information (such as depth or lithofacies) can be encoded using color or symbols. They are useful because they can reveal petrophysical relationships between logs, highlight lithology and fluid-related trends, show overlapping or separation of lithofacies in feature space, help assess log complementarity and redundancy, and provide intuitive guidance for manual interpretation. In the figure, “normalized” means that each well-log variable has been standardized with respect to its own statistical distribution

using z-score normalization, $x_{\text{norm}} = \frac{x - \mu}{\sigma}$, where μ and σ are the mean and standard deviation of the log computed over the analyzed depth interval. This transformation centers each log to zero-mean and scales it to unit variance, removing differences in physical units and dynamic ranges. As a result, the cross-plots emphasize intrinsic relationships and facies overlap in the feature space, rather than being dominated by scale effects. Based on core sample analysis, the lithofacies classification scheme for this well consists of various classes. These include prevalent shale, sandstone with interbedded shale or siltstone, and sandstones partially gas-saturated at varying levels. Appendix 3 includes the exact sample count and percentage for each of the six lithofacies classes, along with their distribution in the training and testing sets based on the stratified split used to evaluate the model’s performance. Additional log statistics are included in the same appendix.

A key challenge of this automatic classification task lies in the significant overlap of these lithofacies within the feature space, which complicates discrimination both for human interpreters and for machine-learning algorithms.

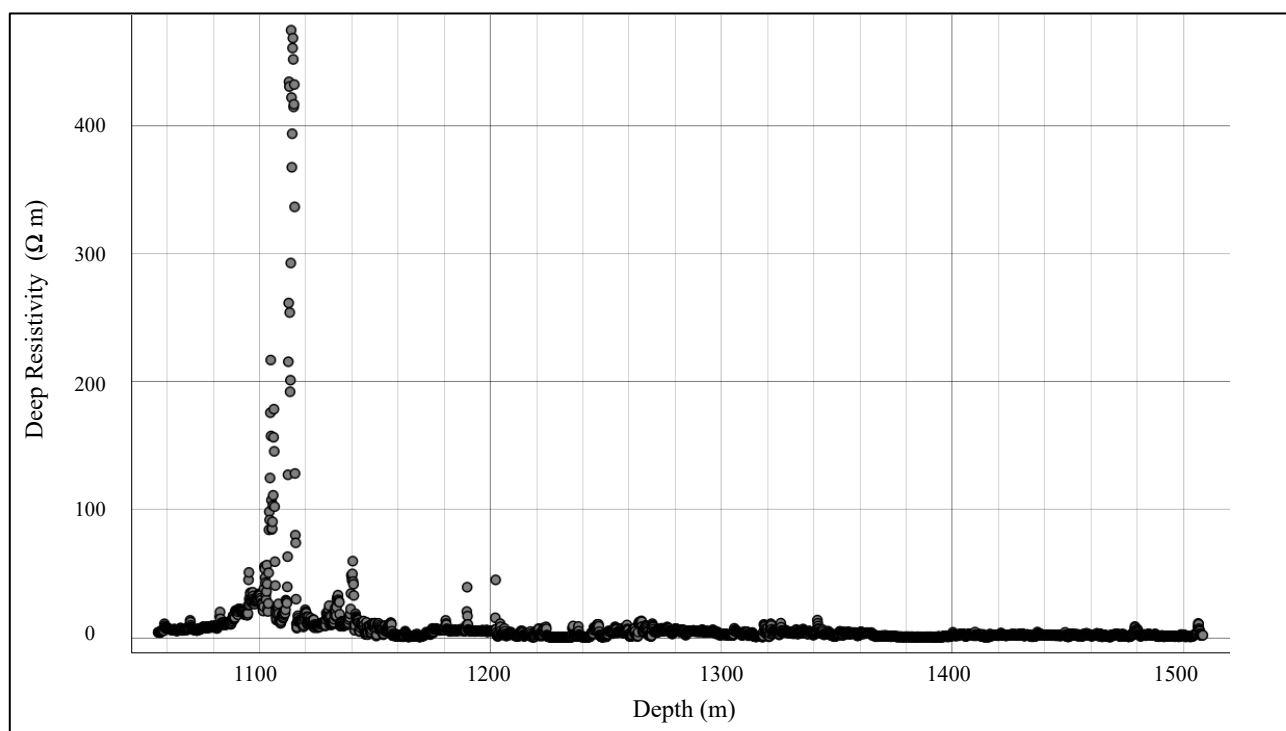


Figure 1. Resistivity log in a selected depth interval, including the reservoir zone (at a depth of about 1100 m).

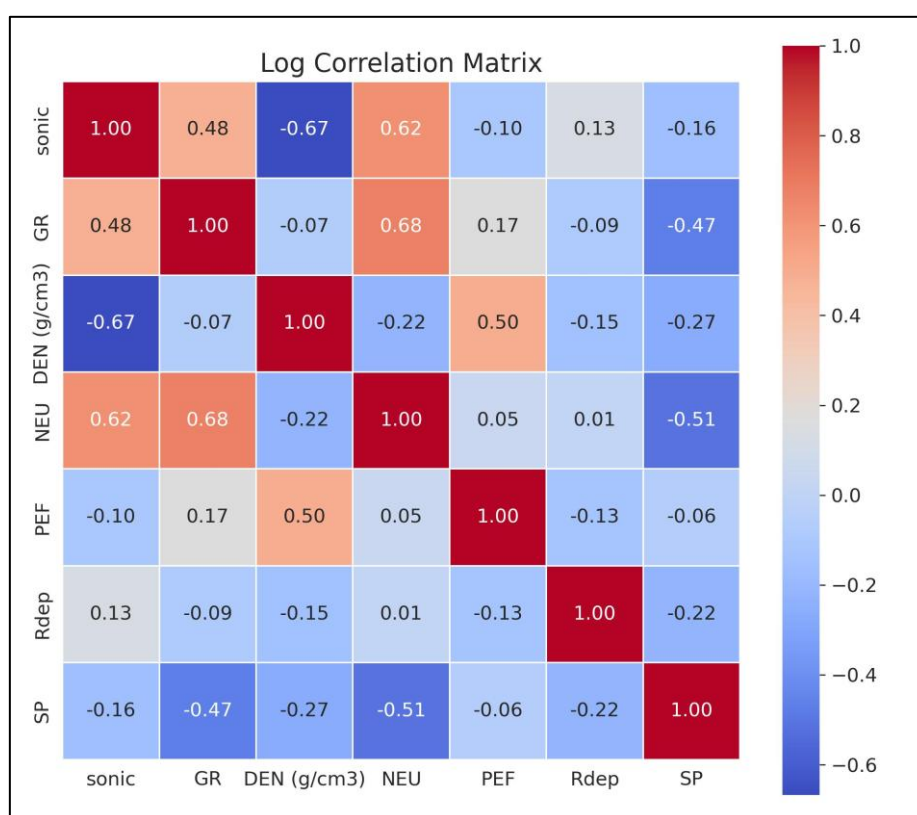


Figure 2. Log correlation matrix. The color scale represents the correlation value.

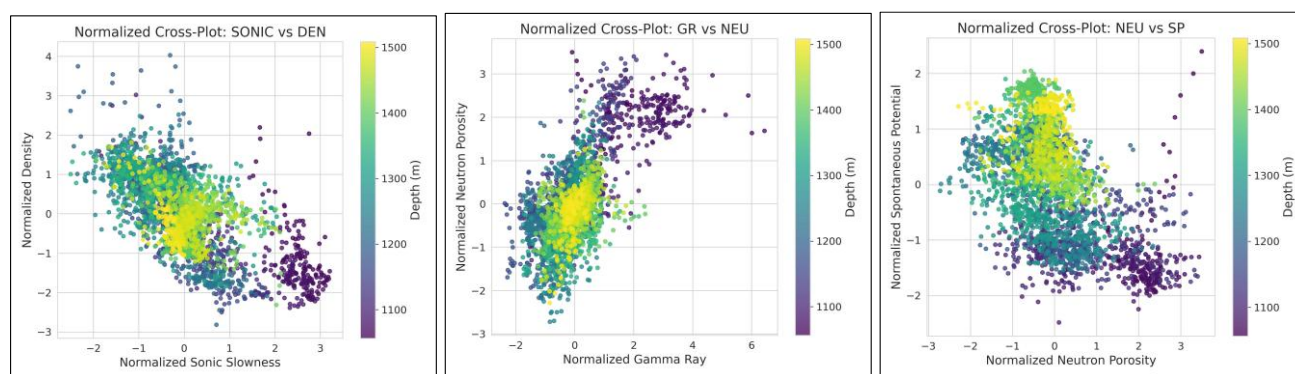


Figure 3. Examples of log cross-plots. Left panel: normalized sonic vs. density; central panel: normalized gamma ray vs. neutron; right panel: normalized neutron vs. spontaneous potentials. All variables are z-score normalized (see text). Each point represents a depth sample.

3.2. Workflow and models' performance

3.2.1. Dataset statistics and preprocessing workflow

The dataset comprises 2969 depth samples, with a uniform depth sampling interval of about 0.15 m, typical of high-resolution wireline acquisition. To ensure model robustness, the data underwent the following structured preprocessing workflow:

Outlier removal: values outside physically meaningful ranges (e.g., negative density or extreme resistivity spikes) were identified and clipped.

Missing value imputation: minor gaps in the sonic or density logs were handled via linear interpolation or k-nearest neighbors (k-NN) imputation, where geological consistency allowed.

Normalization: all features were standardized using a Z-score transformation to ensure that variables with different scales (e.g., resistivity vs. neutron porosity) contribute equally to the loss function.

3.2.2. Class distribution and data splitting

Statistical analysis of the data reveals a notable class imbalance: the primary lithological components (sandstone/siltstone and prevalent sandstone, totaling 62.8%) significantly outweigh the more discrete units such as the sand reservoir (12.0%) and prevalent shale (10.7%). This distribution highlights the heterogeneous nature of the reservoir and justifies the implementation of an ensemble-based stacking meta-learner to maintain high classification accuracy across all classes. For model evaluation, the dataset was partitioned into training and test sets. Multiple tests have been executed using variable ratios between training and test data (ranging from a 50/50 ratio to an 80/20 ratio) to assess the model's stability and sensitivity under different data availability scenarios. In fact, given the significant class imbalance and the stratigraphic continuity of well logs, we wanted to ensure that the performance of the gradient boosting meta-learner was not an artifact of a specific split.

To prevent “data leakage” and account for the vertical continuity of geological formations, the split was performed by selecting continuous depth intervals rather than random point-wise sampling, ensuring that the test set remains entirely “unseen” in a stratigraphic context.

To evaluate the predictive performance of the proposed machine learning stacking ensemble

empowered by a gradient boosting meta-learner, a 5-fold cross-validation strategy was employed on the composite well log dataset to optimize the hyperparameters and train the stacking ensemble's meta-learner (as well as the other learners). Finally, the predictive performance was evaluated on the held-out test set to ensure an unbiased assessment.

This evaluation protocol represents an integrated, hierarchical protocol: the stratified, continuous depth blocks define the outer train/test partition (configured at a conservative 50/50 ratio for the baseline comparisons), whereas the 5-fold cross-validation operates strictly as an inner loop within the training set to optimize parameters and generate out-of-fold predictions without data leakage.

3.2.3. Schematically speaking

- Stratified continuous block splitting: To prevent spatial data leakage caused by the vertical continuity of geological formations, the dataset was partitioned using continuous depth intervals rather than random point-wise sampling. This block selection was stratified to maintain proportional representation of the six imbalanced lithofacies classes across both the training and testing subsets.

- Train/test partitioning: While multiple train/test ratios (from 50/50 to 80/20) were analyzed to assess model stability, the primary benchmark comparisons, McNemar tests, and detailed classification reports utilize the conservative 50/50 split.

- Inner 5-fold cross-validation: Within the training partition, a 5-fold cross-validation scheme was implemented. This inner loop was used for hyperparameter tuning and to generate the out-of-fold probabilistic predictions required to train the gradient boosting meta-learner without information leakage.

- Final evaluation: Once all models were optimized within the internal cross-validation framework, final performance metrics were verified using the completely unseen, held-out test partition.

In our benchmark, we compared the advanced stacking ensemble with a baseline hard majority voting ensemble, with a convolutional neural network (CNN), a residual network (ResNet), a random forest (RF) model, and a logistic regression (LR) model, which serve as reference benchmarks. We remark that random forest and logistic regression serve exclusively as standalone baseline models, and gradient boosting was selected as the sole meta-learner for this specific framework. Because well log data contains highly nonlinear, context-dependent multi-log interactions, linear meta-learners like logistic regression often fail to capture complex inter-model error patterns. Conversely, using an ensemble like random forest as a meta-learner can result in high variance and correlated errors when processing out-of-fold base-model probability distributions. Consequently, stacking experiments utilizing LR or RF as alternative meta-learners were not explicitly conducted for this study.

The random forest implementation in our benchmark utilizes an ensemble of 200 decision trees.

Logistic regression is employed as a fundamental linear baseline to evaluate the necessity of nonlinear models for this specific classification task.

The following are brief descriptions of the two deep learning architectures used for this benchmark.

The CNN architecture is designed as a simple sequential 1D feature extractor. The input layer receives the reshaped well-log data as a 1D sequence. The convolutional layer uses a Conv1D layer with 32 filters and a kernel size of 3 to capture local patterns and correlations between adjacent features. It uses the ReLU activation function. A flatten layer transforms the 1D feature maps into a single vector. A dense (fully connected) layer with Softmax activation classifies the data into the target categories based on the extracted spatial features.

The ResNet architecture implements a “skip connection” (residual learning) to improve gradient flow. The architecture starts with an initial Conv1D layer (32 filters) to extract basic features. A second Conv1D layer processes the data, and its output is added directly to the output of the first layer. This shortcut allows the model to learn identity mappings, preventing performance degradation. Instead of flattening, it uses the “Global-Average-Pooling-1D” function to reduce the dimensionality by taking the average of each feature map, which helps reduce overfitting. A final dense layer with Softmax activation performs the final classification.

Table 1. Final performance metrics evaluated on the independent held-out test set (50/50 split) across all classification models. Standard deviation (SD) values represent validation variability captured during the internal 5-fold cross-validation training phase.

Method	Accuracy ($\pm$ SD)	F1-weighted	F1-macro	Balanced accuracy
Gradient boosting stacking	0.917 $\pm$ 0.014	0.917	0.921	0.917
Majority voting	0.893 $\pm$ 0.016	0.892	0.905	0.902
Standard CNN	0.811 $\pm$ 0.020	0.809	0.829	0.826
ResNet	0.717 $\pm$ 0.023	0.706	0.699	0.696
Standalone random forest	0.900 $\pm$ 0.015	0.899	0.913	0.908
Standalone logistic regression	0.785 $\pm$ 0.021	0.782	0.806	0.814

Table 1 summarizes the final performance metrics evaluated on the independent held-out test set for all approaches. The corresponding standard deviations (SD) reflect the model stability captured during the internal 5-fold cross-validation training phase.

Final performance was strictly assessed on a held-out test set that remained independent during the meta-training phase to ensure an unbiased evaluation.

In Table 1, accuracy serves as a general performance indicator; however, it can be misleading in the presence of class imbalance. To address this, the weighted F1-score was employed, representing the harmonic mean of precision and recall weighted by class prevalence. Furthermore, the macro F1-score was calculated to treat all classes equally, providing a transparent assessment of performance on minority classes. Finally, balanced accuracy (the arithmetic mean of class-specific recall) was utilized to verify the model’s ability to generalize across the entire dataset without being biased toward majority classes.

These results clearly show that traditional ensembles significantly outperform the other machine learning models in this specific domain. The stacking ensemble achieves the highest scores across all metrics (e.g., 0.917 balanced accuracy), suggesting it is the most reliable for practical applications.

While logistic regression is limited by its linear assumptions and random forest struggles with correlated errors among base learners, the gradient boosting approach effectively captures nonlinear interactions and iteratively corrects residual misclassifications.

The lower performance of ResNet (0.717) compared to that of the CNN (0.811) is an interesting finding. It suggests that for our specific dataset, a simpler 1D-CNN architecture captures features better than a more complex residual network, likely due to the size or nature of the input features.

To assess the significance of the performance improvements, pairwise McNemar tests were performed between the gradient boosting stacking ensemble and all benchmark models. The stacking architecture demonstrates a statistically significant improvement over the majority voting baseline, standard CNN, ResNet, and logistic regression under the strict Bonferroni-adjusted threshold. However, it is important to acknowledge that the comparison with the standalone random forest model

does not achieve statistical significance after applying the Bonferroni correction, although it remains significant at the uncorrected level.

In Table 2, we report the contingency matrix only for gradient boosting stacking vs. majority voting, as an example. While analogous contingency matrices were computed for all evaluated methods, they have been omitted here for the sake of conciseness and are included in the dedicated Appendix 4 at the end of the paper. All pairwise McNemar tests and their exact p-values are included in the dedicated Appendix 4 at the end of the paper, evaluated under a Bonferroni-adjusted significance threshold.

Table 2. Contingency table (gradient boosting stacking vs. majority voting).

2 × 2 Contingency table: gradient boosting stacking vs. majority voting			
	Majority voting: Correct	Majority voting: Incorrect	Row totals
Gradient boosting stacking: Correct	1294 (both correct)	69 (only stacking correct)	1363
Gradient boosting stacking: Incorrect	36 (only majority voting correct)	86 (both incorrect)	122
Column totals	1330	155	1485

For completeness of the comparison between stacking ensemble and majority voting methods, we report in Figures 4 and 5 the corresponding confusion matrix. The gradient boosting stacking model outperforms the majority voting baseline across nearly all categories. While the baseline achieves a perfect 100% on prevalent shale (see values on the main diagonal), it struggles with complex facies like sandstone/siltstone (82.0%) and sandstone/shale (87.1%). In contrast, the stacking ensemble improves these to 89.0% and 89.3%, respectively, significantly reducing misclassifications between siltstone and sandstone. Overall, the stacking approach provides a more robust and balanced classification, proving more reliable for geological interpretation than simple majority logic.

Table 3 presents the detailed class-wise classification report of the gradient boosting stacking model evaluated on a 50/50 train/test split. It lists the precision, recall, and F1-score percentages for each of the six lithofacies, cross-referenced with their absolute sample counts (support).

Table 3. Class-wise evaluation report (50% test split) for the gradient boosting stacking method.

Method	Precision	Recall	F1-score	Support (train/test)
Prevalent sandstone with low gas saturation	0.931174089	0.877862595	0.903732809	410/410
Prevalent shale	0.980392157	0.980392157	0.980392157	159/159
Prevalent siltstone	0.919739696	0.965831435	0.942222222	104/104
Sand reservoir with high gas saturation	0.911504425	0.895652174	0.903508772	178/178
Sandstone/shale	0.912408759	0.892857143	0.902527076	111/112
Sandstone/siltstone	0.894117647	0.889929742	0.892018779	522/522

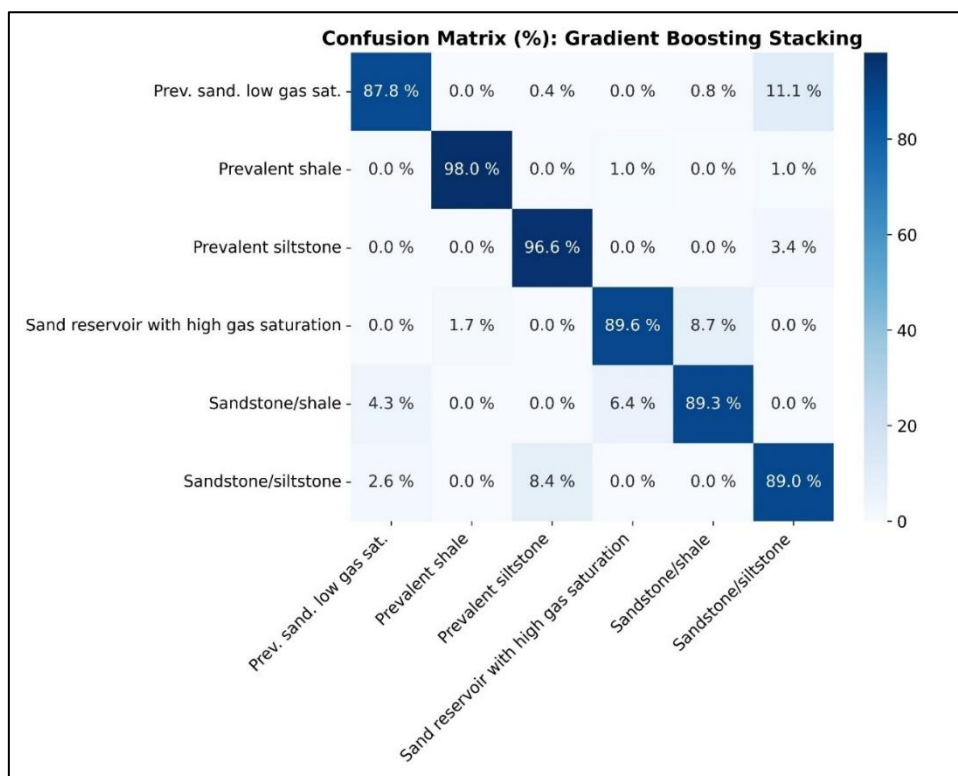


Figure 4. Confusion matrix for gradient boosting stacking.

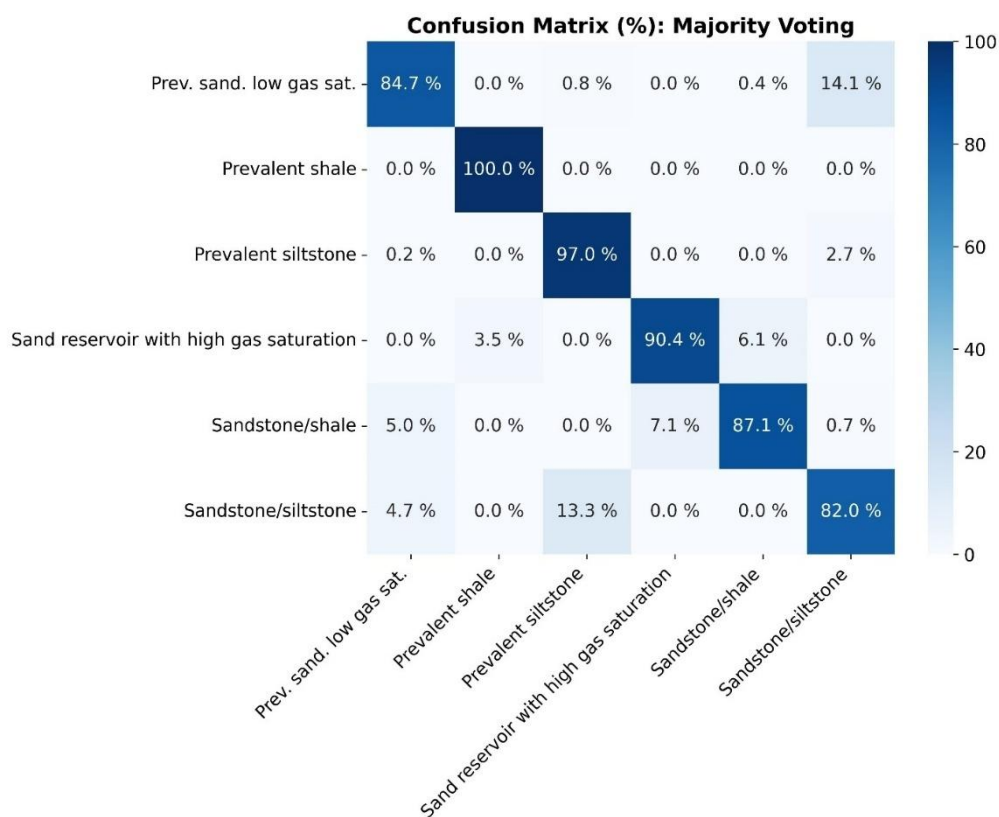


Figure 5. Confusion matrix for majority voting.

3.3. Classification results

Classification results obtained by the gradient boosting method are shown in Figures 6 and 7, plotted on resistivity and density logs, respectively. The results are mutually consistent and physically meaningful and are supported by additional composite well logs in the dataset.

The lithofacies labels used for validating (and for training) the machine learning models were calibrated using a combination of conventional core analysis and rotary sidewall cores (MSCT). These physical samples provided the ground truth for mineralogical composition, grain size, and pore-fluid distribution. The core-to-log depth matching was performed by correlating high-resolution core gamma ray measurements with the wireline logs. Adjustments were made to account for the intrinsic depth shifts between the driller's depth and the logger's depth, ensuring that each labeled interval accurately reflects the petrophysical response recorded by the tools.

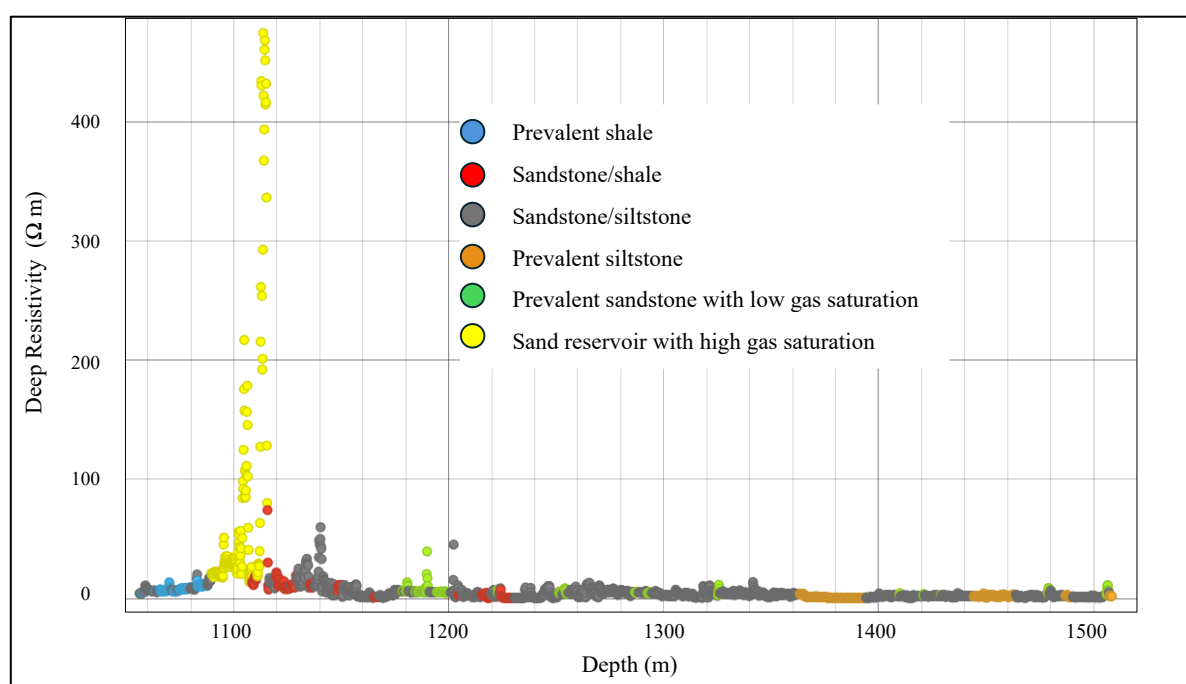


Figure 6. Example of lithofacies classification projected on the resistivity log.

Resistivity is primarily sensitive to fluid type and saturation, while density reflects lithology, porosity, and matrix–fluid interactions. The observed density values exceeding 2.7–2.8 g/cm³ in gas-bearing intervals are not indicative of unreliable data but rather reflect a complex matrix mineralogy characterized by sideritic-calcareous cementation, pyritic nodules, and glauconite—as validated by core laboratory data (grain densities up to 2.93 g/cm³)—which locally increases the bulk density and partially offsets the density-lowering effect of the gas. When analyzed jointly, the composite logs delineate coherent facies trends and realistic transitions between shale-dominated intervals, mixed lithologies, and hydrocarbon-bearing sandstones. However, litho-fluid classification from well logs is intrinsically challenging due to property overlap between facies, gradual geological transitions, variable porosity, and the non-unique response of individual logs. As shown in this study, our classification approach effectively handles these complexities by learning nonlinear multi-log

relationships and exploiting collective patterns emerging from composite log responses. Its architecture accommodates overlapping classes and transitional facies, leading to robust and geologically consistent classifications with high internal coherence and excellent performance metrics.

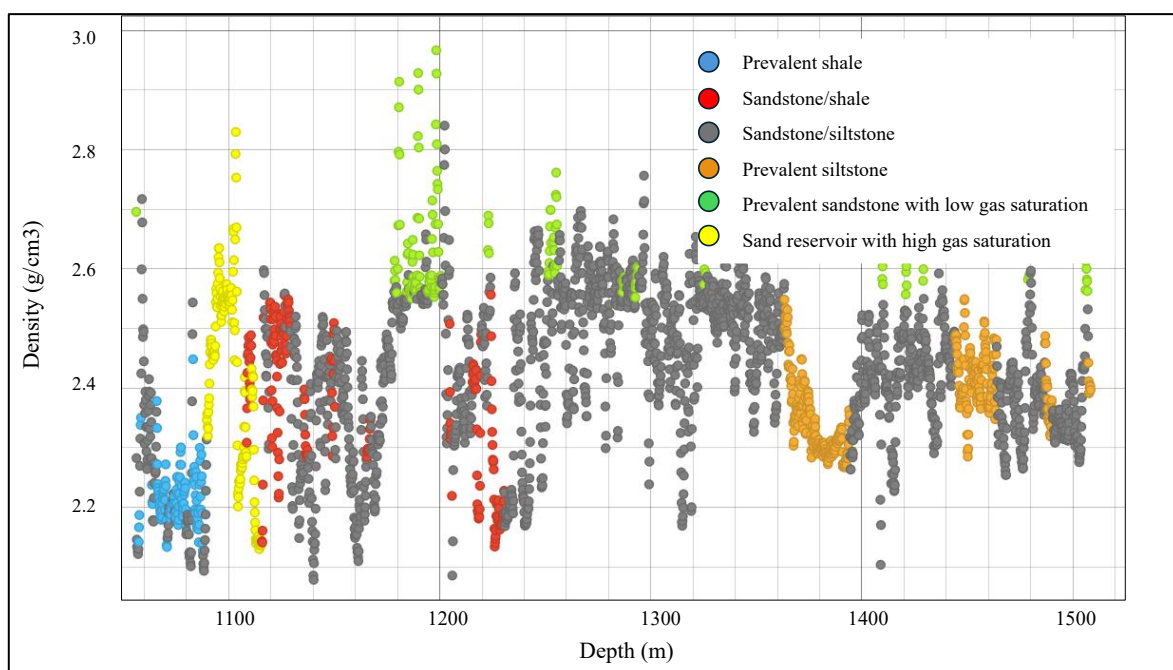


Figure 7. Example of lithofacies classification projected on the density log.

4. Discussion

The results evaluated on the independent held-out test set demonstrate that the stacking ensemble empowered by a gradient boosting meta-learner outperforms the baseline hard majority voting ensemble across all evaluated metrics, including accuracy, weighted and macro F1-scores, and balanced accuracy. This performance gain can be attributed to several key advantages inherent to the stacking approach. First, by combining heterogeneous base learners, the stacking ensemble exploits complementary predictive strengths, allowing the meta-learner to resolve conflicts among base models. The gradient boosting meta-learner further enhances this capability by learning context-dependent error correction rules, effectively identifying when certain classifiers are more reliable for specific lithofacies or log response patterns. This is particularly important for classes with overlapping or subtle log signatures, where simple voting schemes often fail to capture class-specific nuances. The improved macro F1-score and balanced accuracy highlight the method's ability to handle class imbalance and minority classes, providing a more equitable classification performance across all lithofacies. Second, the framework demonstrates robustness to noise and heterogeneity in the input logs. By incorporating probabilistic outputs of base learners together with original log features, the gradient boosting meta-learner can model nonlinear interactions between features and predictions, effectively mitigating systematic misclassifications observed in the majority voting ensemble.

However, several limitations are worth noting.

First, it is worth noting that the primary objective of this study is to demonstrate the overarching

benefits and performance enhancements of a meta-learning framework over traditional standalone baselines, rather than to provide an exhaustive comparative analysis of various alternative meta-learners at Layer-1. While it cannot be completely ruled out that alternative configurations (such as using random forest or logistic regression as meta-learners) might yield localized variations in performance, such an extensive benchmark goes beyond the defined scope of this paper. To avoid any ambiguity, this study explicitly establishes its boundaries at the current benchmarking level, reserving a comprehensive evaluation of diverse meta-learning algorithms for a subsequent dedicated publication.

Second, the complexity of the stacking ensemble increases both training and inference time compared to simpler ensemble methods (about 20%). The requirement to train multiple base learners and a meta-learner may pose challenges when scaling to extremely large datasets or real-time applications. Additionally, the interpretability of the final model is lower than that of individual classifiers, as decision rules are embedded in the meta-learner and are not explicitly transparent. While gradient boosting allows partial inspection of feature importance and decision paths, full interpretability remains limited, particularly for end-users without experience in ensemble methods.

An additional limitation of this study concerns the dataset, which is derived from a single-well sequence. While the high-resolution internal validation demonstrates the robustness of the stacking architecture within this specific domain, the geological features captured may not fully represent regional or basin-wide variations. Therefore, caution should be exercised when extrapolating these findings to other fields without localized recalibration.

Despite these limitations, the proposed stacking framework provides a robust and reproducible approach for lithofacies classification, offering substantial improvements over traditional methods. Future applications could explore several directions:

- Integration of additional heterogeneous base learners, such as convolutional neural networks or recurrent architectures, to capture spatial or sequential dependencies in well-log data.
- Application to multi-well or field-scale datasets, enabling cross-well generalization and more comprehensive subsurface characterization.
- Time-lapse well log analysis, where stacking ensembles could help identify lithological changes over time during production or injection processes.
- Hybrid interpretability approaches, combining meta-learner decision paths with explainable AI techniques, to provide geoscientists with actionable insights alongside high predictive performance.

5. Conclusions

In this work, we proposed a machine-learning stacking ensemble empowered by a gradient boosting meta-learner for automatic lithofacies classification from composite well logs. The framework integrates heterogeneous base learners—random forest, gradient boosting, support vector machine, and K-nearest neighbors—and exploits their probabilistic outputs within a meta-learning architecture specifically designed to correct systematic errors and capture nonlinear interactions among models.

Cross-validated experiments on real composite well log data demonstrate that the stacking ensemble consistently outperforms a baseline hard majority voting approach, as well as other classical learners, including deep learning models, across all evaluation metrics, like accuracy, weighted and macro F1-scores, and balanced accuracy. The observed improvements are particularly relevant for minority and transitional lithofacies, highlighting the ability of the gradient boosting meta-learner to

adaptively weight base classifiers according to geological context and model uncertainty.

Beyond quantitative performance gains, the proposed methodology provides a robust and reproducible framework for handling noisy, heterogeneous, and overlapping petrophysical responses. By combining physical log measurements with model-level uncertainty information, the stacking ensemble provides a robust and structured framework for lithofacies identification. While it remains a primarily data-driven approach, its hierarchical architecture allows for a more nuanced integration of diverse petrophysical logs compared to single-model baselines, leading to higher reliability in complex geological settings.

In conclusion, while the proposed meta-learning model offers a powerful and stable tool for high-resolution subsurface classification within our study area, its reliance on a single-well dataset underscores the need for broader geographical validation. Future research will focus on testing the framework across multi-well datasets and diverse geological basins to confirm its generalizability and transferability on a larger scale.

Use of Generative-AI tools declaration

The author declares they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This study received no specific grant or financial support from any public or private funding institution.

Data availability

The raw data, specific well identifiers, and precise geographic coordinates are confidential and cannot be made publicly accessible.

Conflict of interest

The authors declare no conflict of interest.

References

1. Dramsch JS (2020) Chapter One—70 years of machine learning in geoscience in review. *Adv Geophys* 61: 1–55. <https://doi.org/10.1016/bs.agph.2020.08.002>
2. Binetti MS, Massarelli C, Uricchio VF (2024) Machine learning in geosciences: A review of complex environmental monitoring applications. *Mach Learn Knowl Extr* 6: 1263–1280. <https://doi.org/10.3390/make6020059>
3. Lary DJ, Alavi AH, Gandomi AH, et al. (2016) Machine learning in geosciences and remote sensing. *Geoscience Frontiers* 7: 3–10. <https://doi.org/10.1016/j.gsf.2015.07.003>
4. Aminzadeh F, de Groot P (2006) *Neural networks and other soft computing techniques with applications in the oil industry*, Houten, The Netherlands: EAGE Publications.

5. Zhang L, Zhan C (2017) Machine learning in rock facies classification: An application of XGBoost. *Proceedings of the International Geophysical Conference*, 1371–1374. <https://doi.org/10.1190/IGC2017-351>
6. Zhao T, Jayaram V, Roy A, et al. (2015) A comparison of classification techniques for seismic facies recognition. *Interpretation* 3: SAE29–SAE58. <https://doi.org/10.1190/int-2015-0044.1>
7. Barnes AE, Laughlin KJ (2002) Investigation of methods for unsupervised classification of seismic data. *Proceedings of the SEG Technical Program Expanded Abstracts 2002*, Salt Lake City, USA, 2221–2224. <https://doi.org/10.1190/1.1817152>
8. Bestagini P, Lipari V, Tubaro S (2017) A machine learning approach to facies classification using well logs. *Proceedings of the SEG Technical Program Expanded Abstracts 2017*, Houston, USA, 2137–2142. <https://doi.org/10.1190/segam2017-17729805.1>
9. Dell'Aversana P (2019) Comparison of different machine learning algorithms for lithofacies classification from well logs. *Bollettino di Geofisica Teorica ed Applicata* 60: 69–80. <https://doi.org/10.4430/BGTA0256>
10. Tewari S, Dwivedi UD (2020) A comparative study of heterogeneous ensemble methods for the identification of geological lithofacies. *J Petrol Explor Prod Technol* 10: 1849–1868. <https://doi.org/10.1007/s13202-020-00839-y>
11. Bhattacharya S, Carr TR, Pal M (2016) Comparison of supervised and unsupervised approaches for mudstone lithofacies classification: Case studies from the Bakken and Mahantango–Marcellus shale, USA. *J Nat Gas Sci Eng* 33: 1119–1133. <https://doi.org/10.1016/j.jngse.2016.04.055>
12. Singh H, Seol Y, Myshakin EM (2020) Automated well-log processing and lithology classification by identifying optimal features through unsupervised and supervised machine-learning algorithms. *SPE J* 25: 2778–2800. <https://doi.org/10.2118/202477-PA>
13. Zhang J, He Y, Zhang Y, et al. (2022) Well-logging-based lithology classification using machine learning methods for high-quality reservoir identification: A case study of Baikouquan Formation in the Mahu area of Junggar Basin, NW China. *Energies* 15: 3675. <https://doi.org/10.3390/en15103675>
14. Dogan A, Birant D (2019) A weighted majority voting ensemble approach for classification, *Proceedings of the 4th International Conference on Computer Science and Engineering (UBMK)*, Samsun, Turkey, 1–6. <https://doi.org/10.1109/UBMK.2019.8907028>
15. Zhou ZH (2012) *Ensemble methods: Foundations and algorithms*, New York, Boca Raton: Chapman and Hall/CRC. <https://doi.org/10.1201/b12207>
16. Tian Y, Zhao X, Huang W (2022) Meta-learning approaches for learning-to-learn in deep learning: A survey. *Neurocomputing* 494: 203–223. <https://doi.org/10.1016/j.neucom.2022.04.078>
17. He K, Pu N, Lao M, et al. (2023) Few-shot and meta-learning methods for image understanding: A survey. *Int J Multimed Info Retr* 12: 14. <https://doi.org/10.1007/s13735-023-00279-4>
18. Wolpert DH (1992) Stacked generalization. *Neural Networks* 5: 241–259. [https://doi.org/10.1016/S0893-6080\(05\)80023-1](https://doi.org/10.1016/S0893-6080(05)80023-1)
19. Jia R, Lv Y, Wang G, et al. (2021) A stacking methodology of machine learning for 3D geological modeling with geological-geophysical datasets, Laochang Sn camp, Gejiu (China). *Comput Geosci* 151: 104754. <https://doi.org/10.1016/j.cageo.2021.104754>
20. Natekin A, Knoll A (2013) Gradient boosting machines, a tutorial. *Front Neurobot* 7: 21. <https://doi.org/10.3389/fnbot.2013.00021>

21. Zhang J, Liu G, Wei Z, et al. (2025) Machine learning-based prediction of well logs guided by rock physics and its interpretation. *Sensors* 25: 836. <https://doi.org/10.3390/s25030836>
22. Breiman L (2001) Random Forests. *Mach Learn* 45: 5–32. <https://doi.org/10.1023/A:1010933404324>
23. Ho TK (1995) Random decision forests, *Proceedings of the Third International Conference on Document Analysis and Recognition*, Montreal, QC, Canada, 278–282. <https://doi.org/10.1109/ICDAR.1995.598994>
24. Friedman JH (1999) Stochastic gradient boosting. Technical Report, Stanford University. Available at: <https://statweb.stanford.edu/~jhf/ftp/stobst.pdf>.
25. Friedman JH (2001) Greedy function approximation: A gradient boosting machine. *Ann Statist* 29: 1189–1232. <https://doi.org/10.1214/aos/1013203451>
26. Cortes C, Vapnik V (1995) Support-vector networks. *Mach Learn* 20: 273–297. <https://doi.org/10.1007/BF00994018>
27. Cover T, Hart P (1967) Nearest neighbor pattern classification, *IEEE Transactions on Information Theory*, 13: 21–27. <https://doi.org/10.1109/TIT.1967.1053964>
28. Quinlan JR (1986) Induction of decision trees. *Mach Learn* 1: 81–106. <https://doi.org/10.1007/BF00116251>
29. Duda RO, Hart PE, Stork DG (2001) *Pattern classification*, New York: Wiley.
30. Domingos P, Pazzani M (1997) On the optimality of the simple Bayesian classifier under zero-one loss. *Mach Learn* 29: 103–130. <https://doi.org/10.1023/A:1007413511361>
31. Zhang H (2004) The optimality of Naive Bayes, *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS)*, 562–567.
32. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. *J Comput Syst Sci* 55: 119–139. <https://doi.org/10.1006/jcss.1997.1504>
33. Schapire RE (1990) The strength of weak learnability. *Mach Learn* 5: 197–227. <https://doi.org/10.1007/BF00116037>
34. Vapnik VN (1998) *Statistical learning theory*, New York: Wiley.
35. Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers, *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, 144–152. <https://doi.org/10.1145/130385.130401>
36. Cover T (1965) Behavior of sequential predictors of binary sequences, *Proceedings of the 4th Prague Conference on Information Theory*, 263–272.
37. Devroye L, Györfi L, Lugosi G (1996) *A probabilistic theory of pattern recognition*, New York: Springer. <https://doi.org/10.1007/978-1-4612-0711-5>
38. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778. <https://doi.org/10.1109/CVPR.2016.90>

Supplementary

Appendix 1

Table A1. Hyperparameter table.

Model	Hyperparameter	Value	Motivation
Random forest (RF)	n_estimators	200	Sufficient to stabilize variance without excessive cost
	max_depth	None	Allow full tree growth to capture nonlinear log–facies relationships
	min_samples_split	2	Default, no strong regularization imposed
	min_samples_leaf	1	Preserve minority facies
	bootstrap	True	Improve robustness and variance reduction
	class_weight	balanced	Mitigate lithofacies imbalance
Gradient boosting (base learner)	n_estimators	200	Trade-off between bias and variance
	learning_rate	0.05	Conservative learning to reduce overfitting
	max_depth	3	Weak learners, consistent with boosting philosophy
	subsample	0.8	Improve generalization and noise robustness
	loss	log_loss	Probabilistic multi-class classification
Support vector machine (SVM)	kernel	RBF	Capture nonlinear log relationships
	C	10	Moderate regularization
	gamma	scale	Automatic scaling based on feature variance
	probability	True	Required for stacking
	class_weight	balanced	Handle class imbalance
K-nearest neighbors (KNN)	n_neighbors	6	Empirically stable compromise
	weights	distance	Reduce the influence of distant samples
	metric	minkowski (p=2)	Euclidean distance in normalized space
Gradient boosting (meta-learner)	n_estimators	200	Increased capacity for error correction
	learning_rate	0.05	Strong regularization at meta-level
	max_depth	3	Prevent overfitting on meta-features
	subsample	0.8	Improve generalization
	input features	log features + base probabilities	Context-aware meta-learning (Eq. 6)
	training strategy	5-fold out-of-fold predictions	Prevent information leakage

Hyperparameter table for each base-learner and for the gradient boosting meta-learner. The hyperparameters for both base learners and the meta-learner were determined through an iterative empirical process, prioritizing model stability and the prevention of overfitting.

Appendix 2

Table A2. Geological and petrophysical/reservoir significance of the lithofacies classes.

Class name	Geological definition and core evidence	Representative well log responses	Petrophysical and reservoir significance
Prevalent shale. Typical log responses: highest stable gamma ray baseline tracking consistently above 220 API (up to 363 API). High apparent neutron porosity (NEU > 0.45 v/v) due to shale bounds; low deep resistivity.	Dark grey to black, hard, homogeneous, non-calcareous, and micromicaceous claystones/shales. Cores show laminated fractures with localized calcite infills.	High gamma ray baseline; high neutron porosity; low deep resistivity.	Non-reservoir/seal: micro-porosity only; horizontal and vertical permeability approach zero. Acts as a regional vertical seal.
Sandstone/shale. Typical log responses: highly serrated or erratic gamma ray trace fluctuating between 130 and 190 API. Separation between deep resistivity and shallow-reading tools due to mud filtrate invasion.	Interbedded fine-grained sandstones and medium-dark grey claystones. Cores display wavy to flaser bedding, localized bioturbation, and carbonaceous inclusions.	Highly serrated or erratic gamma ray trace; separation between deep resistivity and shallow-reading tools due to mud filtrate invasion.	Marginal/poor reservoir: highly variable petrophysical properties. Porosity is suppressed; permeability ranges from poor to moderate.
Sandstone/siltstone. Typical log responses: intermediate gamma ray values tracking between 95 and 140 API; stable, moderate sonic travel time; moderate-to-high bulk density (DEN around 2.45–2.55 g/cm ³)	Brownish-grey, hard, blocky siltstone grading to very fine quartzose sandstones. Cores show micaceous fragments, minor silica/calcite cementation, and occasional trace fossils.	Intermediate gamma ray values; stable, moderate sonic travel time; moderate bulk density.	Non-reservoir to poor reservoir: low visible matrix porosity. Permeability is severely restricted by the silt matrix, typically remaining below the net sand cutoff.
Prevalent siltstone. Typical log responses: moderate-to-high gamma ray baseline ranging between 140 and 180 API; locally high bulk density (DEN > 2.55 g/cm ³); low-to-moderate deep resistivity.	Homogeneous, hard, splinty, very argillaceous siltstone. Cores include abundant micromica, bioturbated laminations, and localized carbonaceous/coaly material.	Moderate-to-high gamma ray baseline; locally high bulk density; low-to-moderate deep resistivity.	Non-reservoir: micro-porous matrix with negligible effective porosity. Net sand flag is inactive. Serves as internal barrier/baffle to fluid flow.
Prevalent sandstone with low gas saturation. Typical log responses: deflected low gamma ray curve tracking between 75 and 95 API; moderate deep resistivity; high neutron porosity (NEU around 0.28–0.35 v/v) and bulk density (DEN around 2.40–2.50 g/cm ³) locally affected by heavy minerals.	Fine-to-medium grained, subangular-to-subrounded, moderately sorted quartzose sandstone. Cores reveal water invasion and weak, mottled, or poor hydrocarbon shows.	Deflected (low) gamma ray curve; moderate deep resistivity; high sonic travel time and high bulk density locally affected by heavy minerals.	Water-bearing reservoir: active net sand. High water saturation. Found below the calibrated free water level (FWL).
Sand reservoir with high gas saturation. Typical log responses: low gamma ray baseline tracking below 70 API. Prominent	Fine-to-coarse grained, well-sorted, friable quartzose sandstone with (only locally high) silica/calcite cement.	Sharp gamma ray cutback; very high, spiked deep resistivity; crossover	Pay-zone reservoir: premium reservoir unit. Matrix properties are excellent; horizontal

crossover/separation between density (RHOB) and neutron (NPHI) curves driven by high gas saturation. High reservoir quality with high core porosities ranging from 25% to over 30%.	Cores show strong hydrocarbon odor, extensive natural oil staining, and fluorescence.	or separation between density and neutron porosity curves, with localized high bulk density values driven by specific heavy matrix minerals.	permeabilities are high. Low water saturation, confirming high hydrocarbon volume.
---	---	--	--

Table A2 provides a schematic summary of the fundamental geological, petrophysical, and reservoir characteristics of the lithofacies identified in this study. This synthesis serves as a formal complement to the descriptions provided in the visual data presented in Figures 2, 3, 6, and 7 and in Table A4 in Appendix 3.

Appendix 3

The table below outlines the exact sample count and percentage for each of the six lithofacies classes. Multiple testing configurations were executed to validate the robust performance of the stacking workflow across varying partitioning constraints, specifically evaluated using dynamic training/testing ratios ranging from a 50/50 split up to an 80/20 split. The table below showcases the class distribution cross-referenced against the limits of these configurations.

Table A3. Class-wise sample distribution, training/testing ratio profiles, and well log curve statistics

Class name	Total sample count	Total percentage (%)	50/50 ratio partition (train/test)	80/20 ratio partition (train/test)
Prevalent shale	318	10.7%	159/159	254/64
Sandstone/shale	223	7.5%	111/112	178/45
Sandstone/siltstone	1045	35.2%	522/522	836/209
Prevalent siltstone	208	7.0%	104/104	166/42
Prevalent sandstone with low gas saturation	819	27.6%	410/410	655/164
Sand reservoir with high gas saturation	356	12.0%	178/178	285/71
Total dataset	2969	100.0%	1484/1485	2374/595

Well log curve statistics

The following table summarizes the baseline statistics for each of the primary wireline logging curves used as input features in the predictive workflow. All values are reported after quality control, outlier removal, and missing-value handling.

Table A4. Basic statistical parameters of the well log features.

Log curve	Feature description	Minimum	Maximum	Mean	Std. deviation	Missing value (%)
Sonic	Ac. trav. time ($\mu\text{s}/\text{ft}$)	70.08	125.32	98.45	14.12	0.00%
GR	Gamma ray (API)	68.64	363.16	162.30	52.65	0.00%
DEN	Bulk density (g/cm^3)	2.03	2.97	2.42	0.18	0.00%
NEU	Neutron porosity (v/v)	0.19	0.56	0.38	0.09	0.00%
PEF	Photoelectric fact. (b/e)	1.54	13.82	2.85	1.14	0.00%
Rdep	Deep resistivity (Ωm)	0.63	474.19	28.40	72.50	0.00%
SP	Spontaneous pot. (mV)	43.35	75.23	54.15	5.24	0.00%

Data quality and pre-processing notes:

- Missing values: Due to rigorous standard operating and data validation procedures prior to loading the database into the machine learning environment, intervals containing incomplete runs or major tool washouts were resolved, leading to an active missing-value proportion of 0.00% across the final continuous evaluated dataset.
- Log feature behavior: The wide dynamic range of GR (68.64 to 363.16 API) and Rdep (spiking up to 474.19 Ωm) confirms the high contrast between the organic-rich clay baselines and the high-resistivity fluid intervals. The maximum value of DEN (2.97 g/cm^3) accurately captures the heavy mineral components discussed in the text, reflecting the physical reality of the matrix.

Appendix 4

The tables below are complementary to Table 2 in the paper and represent the contingency table for each of the alternative methods used in the benchmark with the gradient boosting method.

To control the family-wise error rate across our five simultaneous model comparisons ($m = 5$, including majority voting), we applied the Bonferroni correction. With a base significance level of $\alpha = 0.01$, the adjusted significance threshold is $0.01/5 = 0.0020$.

The exact McNemar χ^2 test statistics (computed with Edwards' continuity correction) and their exact p -values are detailed below:

- GB vs. standard CNN: $\chi^2 = 105.47$, $p = 9.61 \times 10^{-25}$ (significant);
- GB vs. ResNet: $\chi^2 = 246.83$, $p = 1.28 \times 10^{-55}$ (significant);
- GB vs. logistic regression: $\chi^2 = 129.35$, $p = 5.69 \times 10^{-30}$ (significant);
- GB vs. majority voting: $\chi^2 = 9.75$, $p = 1.79 \times 10^{-3}$ (significant);
- GB vs. random forest: $\chi^2 = 7.27$, $p = 7.02 \times 10^{-3}$ (not significant under the strict Bonferroni threshold, but significant at the uncorrected level $\alpha = 0.01$).

2 × 2 contingency table: gradient boosting stacking vs. standard CNN

	Standard CNN: Correct	Standard CNN: Incorrect	Row totals
Gradient boosting stacking: Correct	1153 (both correct)	210 (only stacking correct)	1363
Gradient boosting stacking: Incorrect	45 (only standard CNN correct)	77 (both incorrect)	122
Column totals	1198	287	1485

2 × 2 contingency table: gradient boosting stacking vs. ResNet

	ResNet: Correct	ResNet: Incorrect	Row totals
Gradient boosting stacking: Correct	1004 (both correct)	359 (only stacking correct)	1363
Gradient boosting stacking: Incorrect	43 (only ResNet correct)	79 (both incorrect)	122
Column totals	1047	438	1485

2 × 2 contingency table: gradient boosting stacking vs. random forest

	Random forest: Correct	Random forest: Incorrect	Row totals
Gradient boosting stacking: Correct	1303 (both correct)	60 (only stacking correct)	1363
Gradient boosting stacking: Incorrect	33 (only random forest correct)	89 (both incorrect)	122
Column totals	1336	149	1485

2 × 2 contingency table: gradient boosting stacking vs. logistic regression

	Logistic regression: Correct	Logistic regression: Incorrect	Row totals
Gradient boosting stacking: Correct	1116 (both correct)	247 (only stacking correct)	1363
Gradient boosting stacking: Incorrect	50 (only logistic regression correct)	72 (both incorrect)	122
Column totals	1166	319	1485



AIMS Press

© 2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)