STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 0–24.
Published online in International Academic Press (www.IAPress.org)

A Comparative Review of Machine Learning Models for Parkinson's Disease Classification

Khaled Nasri [1], Youssef Boulkhiout [1,*], Samir Balbal [1,2], Abdelouahab Moussaoui [1]

[1] *Department of Computer Science, Faculty of Sciences, Setif 1 University – Ferhat ABBAS, Setif, Algeria*
[2] *Artificial Intelligence Laboratory, Faculty of Sciences, Setif 1 University – Ferhat ABBAS, Setif, Algeria*

Abstract Parkinson's disease (PD) is a chronic, progressive neurological condition that disrupts motor and non-motor abilities, underscoring the urgent need for early and accurate detection to enable prompt therapeutic measures. Although many studies report near-perfect accuracy on the widely used Oxford Parkinson's voice dataset, these figures usually come from a single small hold-out split, without uncertainty quantification, statistical testing or interpretability, which makes them hard to compare and to translate into clinical practice. This study therefore proposes a rigorous and reproducible benchmarking protocol for logistic regression, decision trees, k-nearest neighbors, support vector machines, multilayer perceptrons, Random Forests, XGBoost, LightGBM, CatBoost, as well as voting and stacking ensembles. The protocol combines a subject-disjoint 75/25 hold-out split, 5-fold subject-grouped cross-validation with SMOTE re-fitted inside each fold, training-only preprocessing and augmentation, subject-level bootstrap confidence intervals, McNemar tests with Holm correction (including a subject-clustered variant), clinical error analysis, computational cost and SHAP-based explainability. The stacking ensemble achieved the best and most stable cross-validated performance (accuracy $93.15 \pm 3.78\%$, AUC $98.00 \pm 0.76\%$, sensitivity $96.36 \pm 3.40\%$) and, with Random Forest, CatBoost, XGBoost and soft voting, reached 95.92% test accuracy (95% CI: 91.67–100%), 97.30% sensitivity and 91.67% specificity, i.e., one missed PD recording out of 37. None of the 55 pairwise differences remained significant after Holm correction, showing that the top models form a statistically indistinguishable tier on this small cohort, in which fundamental frequency (MDVP:Fo), pitch period entropy (PPE), shimmer (APQ5) and detrended fluctuation analysis (DFA) were the most influential vocal biomarkers. This study examines the strengths, weaknesses, and diagnostic value of modern machine learning methods while considering key challenges such as class imbalance, generalization, data leakage, and interpretability. It provides useful guidance for selecting appropriate ML tools for Parkinson's disease screening and helps pave the way for future progress in multimodal systems and explainable artificial intelligence for healthcare applications.

Keywords Artificial intelligence, Parkinson's disease, Machine learning, Multilayer perceptron, Ensemble learning, Explainable AI

AMS 2010 subject classifications 68T05, 62H30, 62P10

DOI: 10.19139/soic-2310-5070-4001

1. Introduction

Parkinson's disease is a long-term, advancing neurological illness that damages both movement-related and non-movement-related functions, producing symptoms such as shaking at rest, muscle stiffness, slowed movements, loss of balance, memory problems, and noticeable changes in speech. Ranking as the second most prevalent neurodegenerative disease worldwide after Alzheimer's, PD places an enormous

*Correspondence to: Youssef Boulkhiout (Email: youssef.boulkhiout@ univ-setif.dz). Department of Computer Science, Faculty of Sciences, University of Setif 1 - Ferhat Abbas, Setif, Algeria (19000).

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 202x International Academic Press

burden on healthcare systems, families, and economies. Detecting the condition at an early stage is vital because current treatments work best when started before extensive brain cell loss occurs. Unfortunately, traditional diagnosis depends largely on a neurologist's clinical judgment and patient-reported history, which can vary between examiners and sometimes leads to delayed or incorrect conclusions.

In the last few years, artificial intelligence—especially machine learning (ML)—has become a highly effective aid for identifying PD more objectively and earlier than conventional methods. ML algorithms can automatically discover hidden patterns in many types of patient data, including voice samples, walking characteristics, handwriting dynamics, brain scans, and blood or cerebrospinal fluid markers, often detecting subtle signs that clinicians might overlook. Researchers have tested a broad spectrum of approaches, ranging from long-established techniques such as support vector machines (SVM), random forests, k-nearest neighbors, decision trees, logistic regression, and artificial neural networks (multilayer perceptrons), to modern gradient-boosting systems (XGBoost, LightGBM, CatBoost) and deep-learning architectures. When applied to standard benchmark collections like the UCI/Oxford Parkinson's voice dataset and other clinical databases, these methods have repeatedly achieved impressive classification results. Comparable ML pipelines have recently been applied to other clinical prediction problems, such as heart disease prediction [1] and transcriptomic Alzheimer's disease prediction [2], as well as to speech-based recognition tasks with hybrid deep architectures [3], confirming the growing role of data-driven decision support in health and speech analytics.

Motivation. Despite this progress, the literature on voice-based PD detection suffers from a reproducibility problem rather than an accuracy problem. The Oxford dataset contains 195 recordings from only 31 subjects, and many published pipelines split it at the recording level, oversample or select features before partitioning, tune hyperparameters on the test data, or rely on a single test set of about 39 samples. One or two test recordings then change accuracy by 2.5–5 points, so the reported 98–100% accuracies can be optimistic and are not directly comparable. In screening, moreover, accuracy is less informative than sensitivity (missed PD cases) and specificity (unnecessary referrals), and clinicians require explanations grounded in known dysphonia markers. A transparent, statistically grounded comparison of standard and ensemble learners is therefore needed before any model can be recommended for clinical use.

Nevertheless, important obstacles remain, among them unequal numbers of healthy and affected cases in datasets, choosing the most informative features, making model decisions understandable to doctors, and ensuring reliable performance across diverse patient groups. The present paper offers a detailed side-by-side evaluation of the latest ML algorithms used for PD detection, analyzing their underlying principles, advantages, weaknesses, performance measures, and published outcomes. The main contributions of this work are:

1. A leakage-controlled protocol with a subject-disjoint train/test split and subject-grouped cross-validation, in which imputation, standardization, SMOTE (re-fitted inside each fold) and noise augmentation are fitted on training data only, with fixed model configurations and no feature selection or tuning on held-out data.
2. A robustness-oriented evaluation of eleven classical, boosting, neural and ensemble classifiers reporting cross-validation mean $\pm$ standard deviation, test-set 95% subject-level bootstrap confidence intervals and the train/CV/test generalization gap, instead of a single accuracy value.
3. A statistical comparison of all 55 model pairs using exact McNemar tests with Holm–Bonferroni correction and a subject-clustered variant accounting for repeated recordings of the same speaker.
4. A clinically oriented error analysis based on sensitivity, specificity, per-class recall/F1 and confusion matrices, with explicit discussion of false negatives and decision thresholds.
5. An interpretability and efficiency analysis identifying the dominant vocal biomarkers, quantifying SHAP explanation quality and measuring training and inference cost for low-resource deployment.

By bringing together current evidence and highlighting persisting difficulties and promising research directions—particularly multimodal data fusion and explainable artificial intelligence—this work delivers a clear, up-to-date reference for scientists and medical professionals interested in applying ML to improve

Parkinson's disease diagnosis and monitoring. The remainder of the paper is organized as follows: Section 2 analyzes related work and identifies the research gap; Section 3 describes the classifiers; Section 4 details the dataset and the experimental protocol; Section 5 presents and discusses the results; and Section 6 concludes the paper.

2. Related Work

Advancements in AI for Parkinson's disease identification through computational models have evolved rapidly, with a strong emphasis on unobtrusive indicators such as vocal patterns, script kinematics, and ambulatory metrics. Research efforts routinely utilize the Oxford PD screening collection [4] to mitigate issues like uneven category distributions and faint indicator identification via methods including minority class expansion, attribute filtering, and combined predictive frameworks [5, 6]. The following paragraphs summarize key advancements from 2010 to 2025, ordered by reported classification accuracy, focusing on binary/multiclass PD detection and excluding non-comparable studies. Each study highlights unique contributions to diagnostic accuracy and reliability.

In 2014, Hariharan et al. proposed a hybrid intelligent system combining Gaussian mixture model-based clustering, PCA/LDA for dimensionality reduction, and sequential feature selection with classifiers like LS-SVM, PNN, and GRNN [7]. Using the UCI Parkinson's voice dataset, this approach achieved a remarkable 100% accuracy, emphasizing dysphonia features for accurate PD detection. Similarly, in 2025, Abioye et al. developed a stacked ensemble model using CatBoost and random forest as base classifiers with Logistic Regression as the meta-learner [8]. Applied to the Kaggle PD voice dataset and validated on the PADS smartwatch dataset, it achieved 100% accuracy, a ROC AUC of 1.00, and a Brier score of 0.0071. SHAP analysis identified spread1, spread2, and pitch period entropy as key biomarkers, with Random Over Sampler addressing data imbalance and 20-fold stratified cross-validation ensuring robustness, making it ideal for tele-medicine applications.

Saha and Nath (2025) introduced a lightweight CNN-based ensemble approach integrating multilayer perceptron (MLP) and a hybrid ParkRLP model as base learners, with a modified eight-layer PD-CNN as the meta-classifier [9]. Using the UCI Parkinson's voice dataset with SMOTE for class balancing, their model achieved 99.47% accuracy and a 99.50% F1-score, outperforming standalone ML and CNN methods. Similarly, Celik and Başaran (2023) proposed a hybrid SkipConNet + random forest framework, extracting deep speech features with a customized CNN and classifying them with random forest [10]. Tested on PD-Dataset and PDO-Dataset (speech), it achieved 99.11% and 98.30% accuracy, respectively, improving 3–17.19% over conventional Random Forest models, showcasing the power of integrating DL feature extraction with ensemble classifiers.

In a 2023 study, Srinivasan and colleagues applied a multiclass machine learning framework based on feed-forward neural networks (FNN) to the UCI Parkinson's speech dataset, incorporating SMOTE for class balancing, feature selection techniques, and extensive hyperparameter optimization [11]. Their optimized FNN model recorded an accuracy of 99.11%, recall of 98.78%, and F1-score of 99.23%. The same authors also reported that a Kernel SVM configuration surpassed 95% accuracy, confirming the value of systematic tuning and preprocessing. Another 2023 investigation by Alshammri et al. systematically evaluated several classical and neural approaches including MLP, SVM, random forest, decision tree, and KNN on the identical voice dataset [12]. The best-performing MLP configuration yielded 98.31% accuracy, perfect precision (100%), 98% recall, and 99% F1-score, whereas SVM attained 95% accuracy with an F1-score of 97%, reinforcing the diagnostic potential of acoustic features when combined with SMOTE oversampling, careful feature reduction, and GridSearchCV-based optimization.

Avci and Dogantekin (2016) introduced an expert system named GA-WK-ELM that integrates a genetic algorithm for automatic optimization of wavelet-kernel extreme learning machine parameters, obtaining 96.81% classification accuracy on the standard UCI Parkinson's voice repository [13]. Roobini et al. (2022) carried out a head-to-head comparison of XGBoost versus logistic regression using vocal characteristics

from the same UCI repository; XGBoost delivered 96% accuracy and a Matthews Correlation Coefficient of 0.89, substantially outperforming the 79% accuracy of logistic regression, with gains attributed to Pearson correlation-guided feature selection [14]. Lamba et al. (2021) designed a hybrid pipeline that combined multiple feature selection strategies (mutual information, extra-trees, and genetic algorithms) with conventional classifiers such as naïve bayes, KNN, and Random Forest; the strongest combination—genetic algorithm paired with random forest—reached 95.58% accuracy after SMOTE-based imbalance correction [15].

Nishat et al. (2021) performed an extensive benchmarking of gradient boosting variants (LightGBM, XGBoost, Gradient Boosting, and AdaBoost) on the UCI voice dataset, identifying LightGBM as the top performer with 93.39% accuracy after GridSearchCV tuning, closely followed by XGBoost and standard Gradient Boosting (both above 90%), while AdaBoost lagged at 87.22% [16]. One of the earliest comparative analyses, conducted by Das (2010), showed that neural network architectures could achieve 92.9% accuracy—superior to regression, decision trees, and DM neural methods—thereby contributing to lower clinical misdiagnosis rates [17]. Govindu and Palwe (2023) investigated machine learning applicability in telehealth settings using MDVP-derived acoustic features, where Random Forest produced 91.83% accuracy and 0.95 sensitivity, indicating feasibility for remote PD screening [18]. Finally, Shivangi et al. (2019) developed a dual-module system comprising a CNN-based VGFR Spectrogram Detector for gait analysis and an ANN-based Voice Impairment Classifier; the system attained respective accuracies of 88.1% on gait data (PhysioNet) and 89.15% on voice data (UCI), illustrating the promise of multimodal diagnostic frameworks [19].

Additional studies provide context but focus less on direct detection. Islam et al. (2024) reviewed ML/DL for PD detection across handwriting, voice, and spiral datasets, emphasizing biomarkers and decision-support tools [20]. Sood and Khandnor (2019) compared KNN, Random Forest, Decision Tree, and CNN on the UCI spiral dataset, stressing algorithm selection [21]. Kumar et al. (2020) evaluated 13 ML models on the UCI voice dataset, highlighting random forest, boosted logistic regression, and MLP for mobile health applications [22]. Ramani and Sivagami (2011) laid foundational work using data mining on the UCI dataset [23], while Grover et al. (2018) focused on severity prediction (UPDRS) using deep neural networks [24]. Rani et al. (2024) achieved 98.9% accuracy for sEMG-based motion prediction, relevant for related neuro-muscular applications but not PD-specific [25].

These studies reflect a trend toward ensemble methods, DL hybrids, and robust data preprocessing, achieving near-perfect accuracies (up to 100%). Voice biomarkers dominate due to their non-invasive nature, with future research likely to explore multimodal integration and real-world validation for scalable, reliable PD diagnosis.

Critical analysis and research gap. A closer reading shows that the reported performance is conditioned by the evaluation protocol rather than by the classifier alone. Four recurrent factors explain accuracies of 98–100% on the Oxford data. (i) *Subject overlap*: each of the 31 speakers contributes about six phonations, so a random recording-level split places recordings of the same person in both subsets and lets models recognize speaker identity instead of disease-related dysphonia; subject-wise validation yields markedly lower but more realistic estimates on PD speech data [26]. (ii) *Pre-split preprocessing*: applying SMOTE, feature selection or normalization to the full dataset before partitioning leaks information about the test samples into training [27]. (iii) *Optimistic model selection*: tuning hyperparameters or selecting features on the data used for reporting inflates results, a bias that strictly separated validation avoids [28]. (iv) *Small, single test sets*: with 39 test recordings one error corresponds to 2.56 points, so differences between studies usually lie within sampling noise, yet confidence intervals and significance tests [29] are rarely reported. Finally, only a minority of works analyze which acoustic features drive the predictions, although explainable AI is increasingly a prerequisite for clinical adoption [30].

Consequently, the gap addressed here is not yet another classifier that marginally increases accuracy, but the absence of a unified, leakage-aware and statistically grounded benchmark jointly reporting stability (cross-validation variance), uncertainty (confidence intervals), significance (corrected pairwise tests), clinical error profiles, interpretability and computational cost for the same set of classical, boosting,

neural and ensemble models. Rather than claiming a new state of the art, this study provides such a reference protocol, which allows the near-perfect results of the literature to be interpreted in the proper context.

3. Methods

In this study, a range of machine learning algorithms were developed and systematically compared to assess their performance on the target dataset. The selected models include gradient boosting techniques, conventional machine learning classifiers, neural network architectures, and ensemble strategies. This diverse selection allows for a comprehensive evaluation of different learning paradigms and predictive capabilities. The following subsections provide a concise description of each algorithm.

3.1. Logistic Regression

Logistic regression is among the most commonly used and well-known statistical techniques for classification, especially in binary and multi-class scenarios. Logistic regression models the probability of belonging to a particular class using the logistic sigmoid function applied to a linear combination of the predictors. Model parameters are usually estimated using maximum likelihood estimation, which is a procedure for identifying the set of parameters that best describe the given data. Because of its simplicity, computational efficiency, and interpretability, LR continues to serve as a reliable benchmark in many predictive modeling applications. It is especially suitable when the association between the predictors and the outcome can be reasonably represented as linear in the log-odds domain [31].

3.2. K-Nearest Neighbours

KNN is a lazy and instance-based algorithm, which means that the classification of a new observation will be determined by the majority class in its k nearest neighbors within the feature space [33]. The similarity between two samples can be calculated based on distance metrics like Euclidean or Manhattan distance; and in contrast to other learning methods, there is no need for a training process with KNN. However, the performance of KNN strongly depends on appropriate feature scaling and the careful selection of the parameter k. In addition, prediction can become computationally expensive for large datasets, since the algorithm must compare each new instance with all training samples [34].

3.3. Multilayer Perceptron

MLP is a kind of feed-forward artificial neural networks which consists of an input layer, hidden layers and an output layer. The neurons in these layers use nonlinear activation functions on the weighted sum of their inputs, which makes the network able to learn high-level features [35]. Training is typically performed using the backpropagation algorithm, in which gradients are propagated through the network to iteratively update connection weights and minimize prediction error. Owing to their flexibility and strong capability to approximate nonlinear relationships, MLPs have been widely used in classification tasks across many domains. However, their performance often depends on the availability of sufficient training data, as well as careful tuning of network architecture and hyperparameters [36].

3.4. Support Vector Machine

The support vector machine is an efficient supervised learning algorithm which attempts to find the hyperplane with maximum margin between classes. If the data is not linearly separable, then kernel functions like polynomial and radial basis function are used to map the data points to a higher dimension where the effective separation is feasible. This makes the SVM a suitable candidate for complex and high dimensional classification tasks. Even though the training process may be computationally intensive on large scale datasets, SVMs have proven to perform well in terms of generalization ability. These

characteristics have made them especially valuable in medical diagnosis and other sensitive classification applications [37].

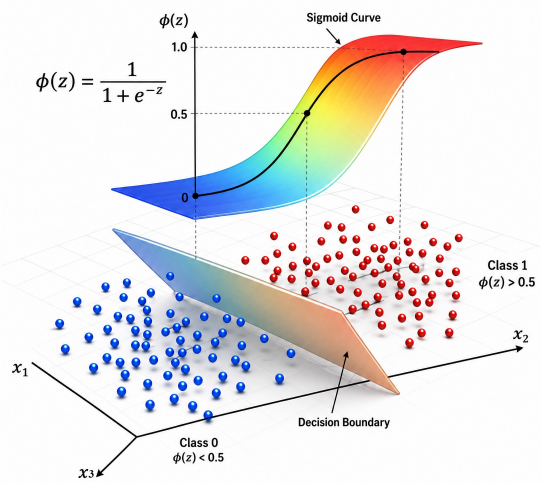


Figure 1. Visualization of the logistic regression classifier.

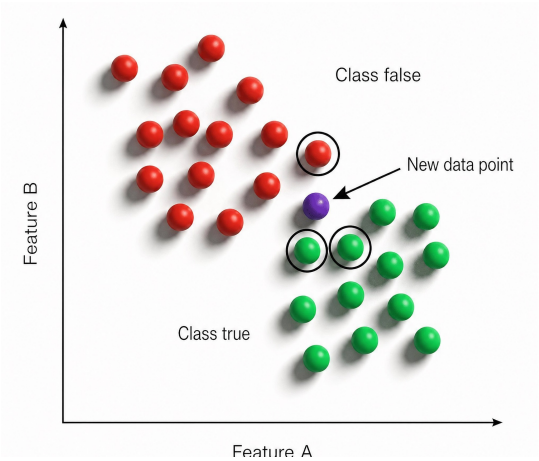


Figure 2. Visualization of the k-nearest neighbors classifier.

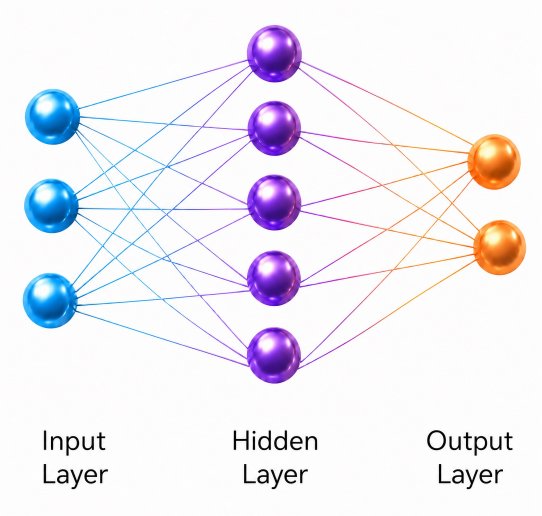


Figure 3. Visualization of the multilayer perceptron classifier.

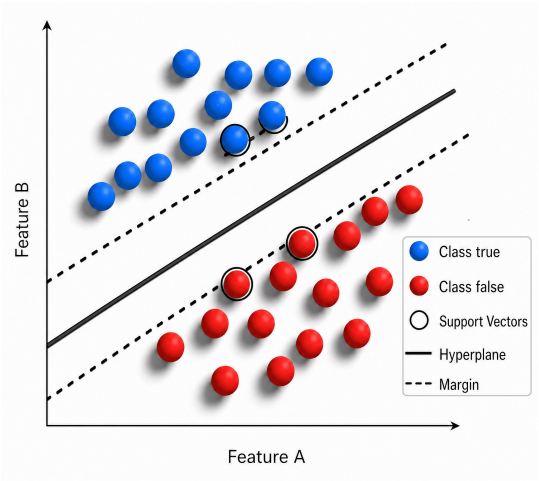


Figure 4. Visualization of the support vector machine classifier.

3.5. Decision Tree

Decision tree is a type of supervised machine learning algorithm that divides the data set into subgroups by using feature values recursively, leading to a tree-like structure where the internal nodes denote decision rules, while the leaf nodes indicate the outcome. The simple and intuitive nature of decision tree allows it to deal with both categorical and continuous features while also incorporating nonlinearity in the data. Thanks to its straightforward nature, the decision tree algorithm is known for being highly interpretable and easy to implement. Nonetheless, the weakness of a decision tree lies in the tendency of overfitting,

especially when the tree grows deeper and fits the training data too well. To overcome this problem and boost performance, decision trees are commonly used as building blocks for ensemble algorithms such as Random Forest and boosting [38].

3.6. Ensemble Learning

Ensemble learning is based on the fusion of predictions from several classifiers to generate an even more powerful classifier. This process is usually done via voting schemes, which can either be hard voting by majority rules or soft voting by average prediction probabilities [39]. By integrating models with complementary strengths, including boosting algorithms, conventional machine learning classifiers, and neural networks, ensemble methods are able to capture diverse patterns within the data. This collaborative approach often reduces both variance and bias, leading to improved robustness and predictive accuracy compared with individual models used in isolation [40].

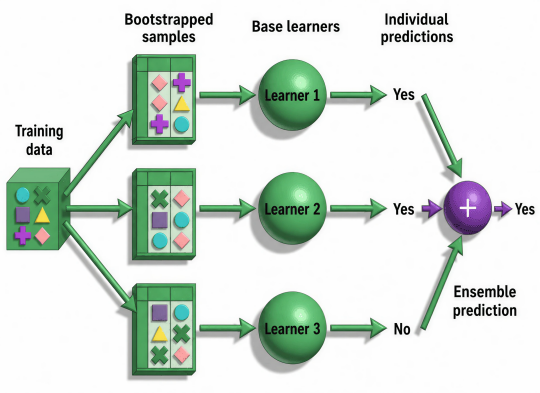


Figure 5. Visualization of the general ensemble learning framework.

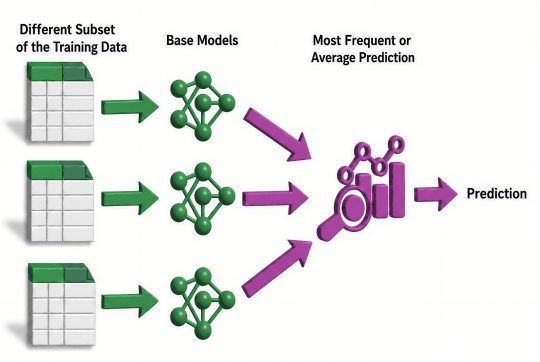


Figure 6. Visualization of the bagging-based ensemble learning approach.

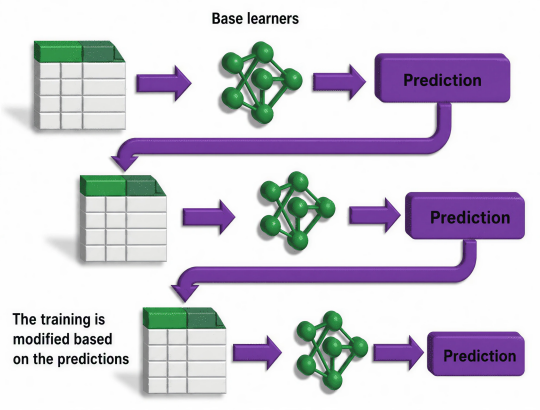


Figure 7. Visualization of the boosting-based ensemble learning approach.

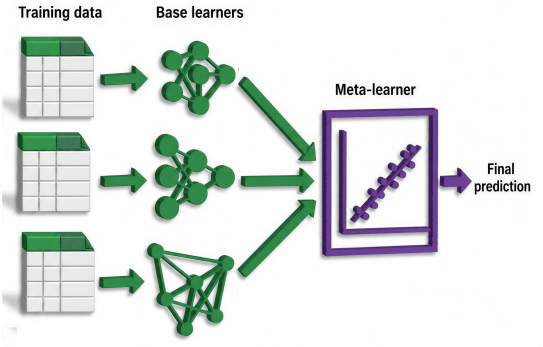


Figure 8. Visualization of the stacking-based ensemble learning approach.

3.7. Random Forest

Random forest is a machine learning technique that works using ensembles of decision trees. Individual trees are trained on bootstrap samples from the data set, whereas a random selection of features is used in

splitting criteria. Randomness from both sources makes trees less correlated with each other, which leads to lower model variance and avoids overfitting. Predictions are produced by aggregation of the output of individual trees by voting in case of classification and averaging in case of regression. Owing to its robustness, stability, and strong predictive capability, RF is particularly well suited for the analysis of complex and noisy healthcare datasets [41].

3.8. Extreme Gradient Boosting

XGBoost represents a highly sophisticated approach to gradient boosting based on the construction of a sequence of decision trees to optimize a smooth loss function. The method employs such regularization strategies as learning rate reduction and feature subsampling to prevent overfitting and increase generalization ability. In addition to being highly efficient, it also allows for parallelization and distributed computing. Due to high accuracy, scalability, and consistency, it has become one of the most popular machine learning algorithms in recent years [42].

3.9. Light Gradient Boosting Machine

LightGBM is a fast machine learning algorithm for gradient boosting that ensures computational speed and high levels of prediction accuracy. The algorithm uses histogram-based decision trees and leaf-wise splitting to grow the branch that provides maximum information gain. This adaptive structure enables faster training and lower memory usage compared with many conventional boosting methods. In addition, its support for parallel processing and GPU acceleration makes it particularly effective for large-scale classification tasks. Owing to its speed, scalability, and competitive performance, LightGBM has become a popular choice for modern machine learning applications [43].

3.10. Categorical Gradient Boosting

CatBoost is a sophisticated form of gradient boosting that is specially made to deal with datasets that have categorical features. CatBoost builds an ensemble of decision trees, adding one at a time, each one training on the errors made by the others. The algorithm employs ordered boosting to limit overfitting and uses symmetric tree structures to improve computational efficiency. Among its main strengths are native handling of categorical features, reduced preprocessing requirements, faster training, and stable predictive performance. These characteristics make CatBoost a particularly effective choice for structured biomedical and healthcare datasets [44].

4. Application

Figure 9 illustrates the full end-to-end workflow designed for Parkinson's disease detection using the Oxford Parkinson's disease detection dataset from the UCI repository. Recordings are first assigned to a training subset (146 recordings, 75%) and a test subset (49 recordings, 25%) at the *subject level*, so that all phonations of a speaker belong to exactly one subset; the test subset is never used for fitting, resampling, feature ranking or model configuration. All data-dependent operations are learned on the training subset only: mean imputation and z-score standardization of the 22 vocal features, SMOTE oversampling of the minority (healthy) class and a noise-injection augmentation step. Stability is assessed with 5-fold subject-grouped stratified cross-validation on the training subset, in which SMOTE is re-fitted inside each training fold and every validation fold contains only original recordings of unseen speakers.

A diverse collection of state-of-the-art classifiers is then employed and trained on the rebalanced training data, encompassing traditional algorithms (Support Vector Machines, Random Forests, k-Nearest Neighbors, Decision Trees, Logistic Regression, and fully connected Neural Networks) alongside advanced gradient boosting systems (CatBoost, LightGBM, and XGBoost). In addition, a robust soft-voting ensemble and a stacking ensemble are constructed on top of the individual learners to achieve higher

stability and generalization. Each model is characterized by its 5-fold cross-validation performance on the training subset (mean ± standard deviation), which measures stability, and by its performance on the unseen test subset with 95% bootstrap confidence intervals, which measures generalization. Evaluation relies on accuracy, AUC-ROC, precision, sensitivity, specificity, F1-score and Cohen's kappa, complemented by significance testing, error analysis, computational cost and explainability metrics.

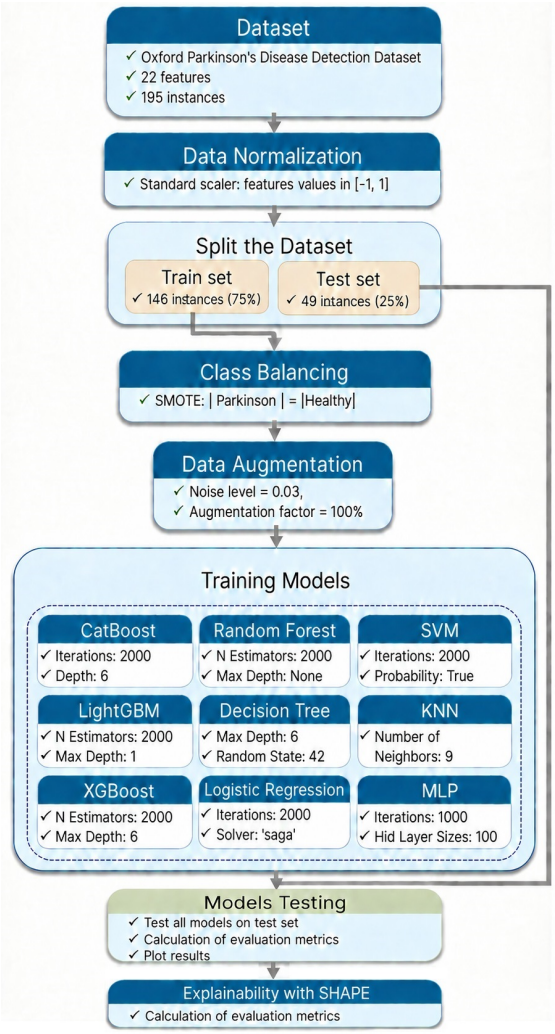


Figure 9. The pipeline used to test models: subject-level train/test split, training-only preprocessing (imputation, scaling, SMOTE, augmentation), 5-fold subject-grouped cross-validation, hold-out testing with subject-level bootstrap confidence intervals, McNemar–Holm tests and SHAP-based explainability analysis.

4.1. Dataset

This Parkinson's disease dataset for detection was created by Max Little from the University of Oxford in collaboration with the National Center for Voice and Speech in Denver, Colorado. The dataset includes 195 recordings of sustained phonation from 31 subjects, out of which 23 were patients with Parkinson's disease while the rest 8 were healthy. The data is supplied as a clean, comma-separated ASCII file containing no missing entries; each record represents a single voice sample and consists of 22 continuous acoustic

attributes—mainly different forms of jitter and shimmer, fundamental frequency statistics, harmonic-to-noise ratios, and several nonlinear complexity measures—along with a binary class label named "status" (0 = healthy, 1 = PD), resulting in 147 PD-positive and 48 control instances.

This collection has become a standard reference dataset in biomedical machine learning for early, non-invasive Parkinson's disease screening based solely on vocal signals. Its relatively small scale, clear clinical significance, and moderate class imbalance (approximately 3:1 ratio) make it particularly well-suited for benchmarking classification techniques, feature selection strategies, and remote-monitoring solutions. Hosted on the UCI Machine Learning Repository and cited in hundreds of studies, it continues to be one of the most popular resources for voice-based PD detection research, regularly supporting published accuracies exceeding 90%.

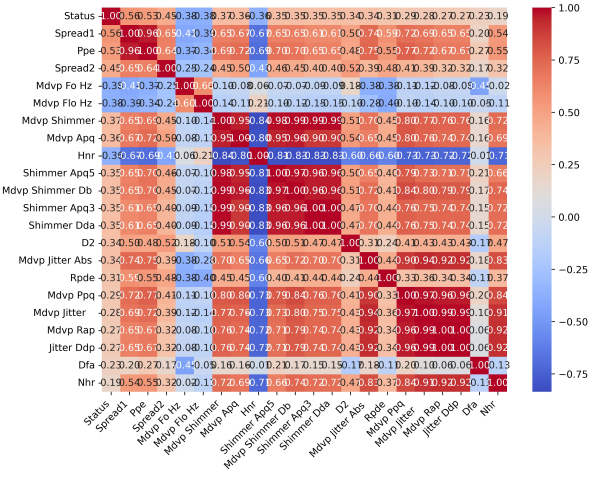


Figure 10. Correlation Heatmap of 10 Features.

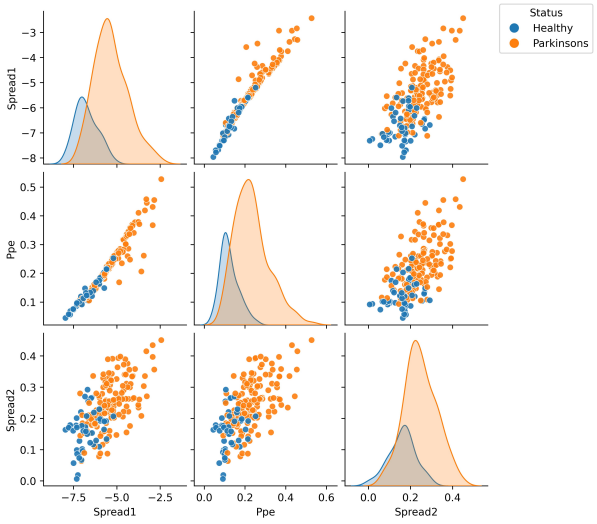


Figure 11. Dataset Feature Relationships.

4.2. Data Normalization

To guarantee numerical stability and optimal gradient flow during training, every acoustic feature was first cast to the float32 precision type. Although the dataset contains no missing entries, a mean imputer fitted on the training subset was kept in the pipeline for robustness. Subsequently, standardization (z-score normalization) was performed using scikit-learn's standard scaler, which subtracts the feature mean and divides by its standard deviation computed from the training partition only. This transformation yields a distribution with zero mean and unit variance for each of the 22 predictors, significantly accelerating convergence of gradient-based learners and preventing dominance by features with larger original scales. The mathematical expression for the standardized value z of a raw feature x is:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where μ and σ represent the mean and standard deviation of the respective feature estimated exclusively on the training set. The same training-subset statistics are applied to the test recordings. In the internal cross-validation, the standardized training subset is partitioned into folds, so the validation folds contribute to μ and σ; this unsupervised step uses no class labels, has no effect on tree-based learners and never involves the test subset.

4.3. Feature Set

No feature selection or dimensionality reduction was performed: all 22 acoustic attributes of Table 1 were used by every model, and the `name` (recording identifier) and subject identifier columns were excluded from the predictors. This removes a frequent source of optimistic bias and guarantees that all classifiers are compared on the same input space. The feature-importance ranking of Section 4.10 is computed on the training subset only and serves interpretation, not input selection.

Table 1. Overview of Features in the Oxford Parkinson's Disease Detection Dataset

Feature	Description
`name`	Subject identifier and recording index (ASCII format)
`MDVP:Fo (Hz)`	Average fundamental frequency of vocal oscillation
`MDVP:Fhi (Hz)`	Highest observed fundamental frequency
`MDVP:Flo (Hz)`	Lowest observed fundamental frequency
`MDVP:Jitter(%)`, `MDVP:Jitter(Abs)`, `MDVP:RAP`, `MDVP:PPQ`, `Jitter:DDP`	Multiple indices quantifying cycle-to-cycle frequency instability (jitter)
`MDVP:Shimmer`, `MDVP:Shimmer(dB)`, `Shimmer:APQ3`, `Shimmer:APQ5`, `MDVP:APQ`, `Shimmer:DDA`	Multiple indices quantifying cycle-to-cycle amplitude instability (shimmer)
`NHR`, `HNR`	Noise-to-harmonics ratio and harmonics-to-noise ratio
`status`	Binary health label (1 = PD, 0 = healthy)
`RPDE`, `D2`	Nonlinear measures of dynamical complexity
`DFA`	Detrended fluctuation analysis — fractal scaling exponent
`spread1`, `spread2`, `PPE`	Three nonlinear indicators of fundamental frequency perturbation

4.4. Dataset Splitting

The Oxford Parkinson's disease detection data set, available from the UCI Machine Learning Repository, was made available as a pre-formatted CSV data set consisting of 195 sustained vowel phonations from 31 subjects (23 having been diagnosed with Parkinson's disease and 8 being healthy control patients). Upon initial investigation of the data set showing that there were no missing values, each phonation was described using 22 continuous vocal acoustic measurements. Because each subject contributed several phonations, a random recording-level split would place recordings of the same speaker in both subsets and allow models to exploit speaker identity. The data set was therefore partitioned at the subject level: every subject identifier was assigned as a whole either to the training subset (146 recordings) or to the test subset (49 recordings), i.e., a 75/25 ratio, while preserving the original class distribution of about 75% PD and 25% healthy recordings in both subsets (Table 2). The larger test subset also refines the granularity of the test metrics, since one error now corresponds to 2.04 percentage points.

Table 2. Class distribution of the training and test subsets before and after SMOTE.

Subset	Total	Healthy (%)	PD (%)	PD:Healthy
Test	49	12 (24.49)	37 (75.51)	3.08
Train (before SMOTE)	146	36 (24.66)	110 (75.34)	3.06
Train (after SMOTE)	220	110 (50.00)	110 (50.00)	1.00

4.5. Class Balancing

Due to the pronounced class imbalance in the Oxford Parkinson's Disease Detection Dataset (roughly 75% PD cases and 25% healthy controls), the synthetic minority over-sampling technique (SMOTE) [45] was employed solely on the training subset. SMOTE creates additional synthetic examples for the underrepresented healthy class by interpolating between each healthy instance and its $k = 3$ nearest neighbors within the 22-dimensional standardized feature space, resulting in a fully balanced training set (from 36 to 110 healthy recordings, i.e., 220 training samples; Table 2) while leaving the original test partition untouched. In cross-validation, SMOTE is re-fitted inside each training fold, so synthetic samples are never generated from, nor evaluated on, validation recordings, and class-balanced loss weights were used by the learners that support them (Table 4). This approach successfully mitigates bias toward the dominant class, substantially boosts recall for healthy individuals, and produces more equitable and clinically reliable diagnostic outcomes.

4.6. Data Augmentation

Given the modest size of the dataset and the resulting risk of overfitting, we expanded the training partition through a targeted noise-injection strategy. After SMOTE, each class of the training set was enlarged to twice the size of the largest class (augmentation factor of 100%) by drawing recordings of that class with replacement and adding zero-mean Gaussian noise to the drawn copies only, $\tilde{\mathbf{z}} = \mathbf{z} + \boldsymbol{\varepsilon}$ with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2_{\text{aug}}\mathbf{I})$ and $\sigma_{\text{aug}} = 0.03$ in standardized units, i.e., 3% of the standard deviation of each feature, giving 440 training samples (220 per class). Augmentation is applied only to the training subset used for the final models; it is not used inside the cross-validation folds (which apply SMOTE only) and never touches validation or test recordings. Regarding physiological validity, repeated phonations of the same speaker naturally differ in jitter, shimmer and noise measures because of breath control, recording conditions and the frame-based estimation of these quantities; a perturbation of 3% of one standard deviation is far smaller than this within-speaker variability, so it acts as an input-level regularizer that discourages decision boundaries relying on exact feature values, rather than as a generator of new clinical cases. We nevertheless acknowledge that isotropic Gaussian noise ignores the correlations among jitter and shimmer variants; physiologically constrained augmentation on the raw audio signal is left for future work.

4.7. Performance Evaluation Metrics

The capability of each classifier was measured both by 5-fold cross-validation on the training subset and on the untouched test set using a wide array of well-established evaluation criteria: overall accuracy, precision, sensitivity (recall), specificity, area under the receiver operating characteristic curve (AUC-ROC), F1-score, and Cohen's Kappa statistic. Because accuracy can mask clinically important errors on imbalanced data, particular emphasis is placed on sensitivity and specificity, the number of false negatives, the balanced accuracy and the negative predictive value. To facilitate interpretation, several visualization aids were also generated, such as ROC curves, confusion matrices, cross-validation distributions and confidence-interval plots.

- **Confusion Matrix**: Displays the complete distribution of predicted versus actual classes in a binary classification scenario, structured as follows:

Table 3. Standard layout of a confusion matrix for binary diagnostic tasks.

		Predicted Class	
		Healthy	**Parkinson's**
True Class	**Healthy**	TN	FP
	Parkinson's	FN	TP

- **Accuracy (ACC)**: Indicates the fraction of instances correctly classified out of the total.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

- **F1-Score**: The harmonic average of precision and recall, especially valuable when classes are unevenly distributed.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

with

$$\text{Precision} = \frac{TP}{TP + FP}, \ \text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

- **Sensitivity, Specificity, Balanced Accuracy and NPV**: Sensitivity (recall of the PD class) is the proportion of PD recordings correctly detected, and its complement is the false-negative rate, i.e., the proportion of patients who would be missed by a screening tool. Specificity is the proportion of healthy recordings correctly identified; its complement determines unnecessary referrals.

$$\text{Sens} = \frac{TP}{TP + FN}, \quad \text{Spec} = \frac{TN}{TN + FP}, \quad \text{BalAcc} = \frac{\text{Sens} + \text{Spec}}{2}, \quad \text{NPV} = \frac{TN}{TN + FN} \tag{5}$$

- **AUC-ROC**: Measures the model's ability to distinguish between classes across all possible decision thresholds. The ROC curve is constructed by plotting the True Positive Rate (Recall) against the False Positive Rate:

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

The AUC value represents the area beneath this curve (calculated using the trapezoidal rule), where values closer to 1 reflect stronger discriminative performance.

- **Cohen's Kappa (κ)**: Assesses inter-rater agreement adjusted for chance, especially valuable when classes are unevenly distributed. Values range from -1 to $+1$:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{7}$$

with $p_o = \frac{TP+TN}{N}$, $p_e = \frac{(TP+FP)(TP+FN)+(FN+TN)(FP+TN)}{N^2}$ where N is the total number of observations.

4.8. Statistical Analysis

Three complementary procedures were used to quantify uncertainty and significance.

- **Stability.** For every model, 5-fold stratified *subject-grouped* cross-validation (StratifiedGroupKFold of scikit-learn, shuffling, seed 42) was run on the training subset, so that all recordings of a subject fall in the same fold and each validation fold contains only unseen speakers; SMOTE is re-fitted on each training fold. Each metric is reported as mean ± standard deviation over the folds, and the difference $\Delta = \text{Test} - \text{CV}$ indicates the optimism of the single hold-out estimate.
- **Confidence intervals.** 95% confidence intervals of the test metrics were obtained with a class-stratified, subject-level (cluster) percentile bootstrap [46]: in each of $B = 2000$ replicates, whole subjects are resampled with replacement within each class, so that the correlation between recordings of the same speaker is respected. The same replicates are used for all models (paired bootstrap), and Wilson score intervals are also reported for specificity, which is estimated from only 12 healthy recordings.

- **Pairwise significance.** All $\binom{11}{2} = 55$ model pairs were compared on the same test recordings with the McNemar test [47, 29], based on the discordant counts n_{01} and n_{10}; the exact two-sided binomial version was used, since $n_{01} + n_{10} < 25$ for all pairs. Because several recordings belong to the same speaker, the clustered McNemar statistic of Durkalski et al. [48], with subjects as clusters, was additionally computed, and the family-wise error rate was controlled with the Holm–Bonferroni procedure [49] at $\alpha = 0.05$.

4.9. Model Configuration and Regularization

Given the very small number of training subjects, no automated hyperparameter search was carried out: searching a large grid on 146 recordings from about two dozen speakers would itself overfit the validation folds and would require a nested cross-validation that the sample size cannot support [28].

Instead, every model was used with a single, fixed configuration chosen *a priori* from commonly recommended settings for small tabular data sets, identical in cross-validation and in the final training, and never adjusted after inspecting test results. These configurations and the regularization mechanisms of each model are summarized in Table 4; parameters not listed keep the library defaults. Overfitting is further controlled by class weighting, training-only preprocessing and input-noise augmentation. The soft-voting ensemble averages the predicted probabilities of the nine base learners, whereas the stacking ensemble trains a logistic-regression meta-learner on out-of-fold (internal 5-fold) base-learner probabilities, which prevents it from seeing predictions made on its own training data. All random components use the seed 42.

Table 4. Fixed model configurations and regularization mechanisms (parameters not listed use library defaults).

Model	Configuration (regularization / overfitting control)
LR	solver=saga, max_iter=2000, CW; default ℓ_2, $C = 1$
KNN	$k = 9$ (Euclidean, uniform); large k smooths the decision boundary
SVM	RBF ($C = 1$, γ =scale), CW, probability=True, max_iter=10000; soft-margin C, kernel width γ
DT	max_depth=6, CW; depth limit (pre-pruning)
MLP	1 hidden layer, 100 ReLU, Adam, max_iter=1000, $\alpha = 10^{-4}$; ℓ_2 weight decay
RF	max_depth=None, CW; bagging + random feature subsampling
XGBoost	max_depth=6, scale_pos_weight=1; shrinkage, ℓ_2 leaf penalty, depth limit
LightGBM	lr=0.1, min_child_samples=10, CW; shrinkage + minimum leaf size
CatBoost	depth=6, lr=0.1, auto_class_weights=Balanced; ordered boosting, ℓ_2 leaf regularization, symmetric trees
Voting	soft voting over the nine base learners; probability averaging
Stacking	nine base learners + logistic-regression meta-learner, cv=5, passthrough=False; out-of-fold meta-training

4.10. Explainability Analysis

Interpretability was examined at two levels. First, a global ranking of the 22 acoustic features was obtained from the native importance scores of two gradient-boosting models fitted on the training subset before resampling: LightGBM (split-based importance) and CatBoost (prediction-value change), both with 2000 trees and class-balanced weights; each importance vector was min–max normalized to $[0, 1]$ and the two were averaged into a consensus score.

Second, local SHAP attributions [30] were computed for every classifier on the 49 test recordings, using the exact TreeSHAP explainer for the tree-based models (CatBoost, XGBoost, LightGBM, RF, DT) and the model-agnostic KernelSHAP explainer (50 background samples from the training set, 100 coalition samples) for LR, KNN, SVM, MLP, Voting and Stacking. The quality of these explanations was quantified on a random subsample of 30 test recordings, using the standardized training mean (the zero vector) as masking baseline and the top 20% (four) features as the "top-k" set: *fidelity*, the correlation between the SHAP mass of randomly masked feature subsets and the resulting drop in predicted PD probability (25 perturbations per instance, higher is better); *infidelity* [50], the expected squared difference between the

attribution-weighted Gaussian perturbation ($\sigma = 0.1$) and the actual change in model output (lower is better); *sparsity*, the mean Gini coefficient of |SHAP| (higher values indicate attributions concentrated on fewer features); *stability*, the mean normalized ℓ_2 distance between the SHAP vector of an instance and those of five noisy copies ($\sigma = 0.1$, 10 instances, lower is better); *comprehensiveness* and *sufficiency* [51], the change in predicted PD probability when the top-k features are masked or exclusively kept (higher comprehensiveness and sufficiency closer to zero are better); and *feature agreement* and *rank correlation* [52], the overlap of the top-10 features and the Spearman correlation between the global rankings produced by SHAP and by LIME (200 perturbation samples). The total computation time, including the LIME explanations, was also recorded.

4.11. Training Models

All classifiers in this work were fitted using the imbalance-corrected and augmented training partition. The selected algorithms were deliberately chosen based on their proven effectiveness with structured tabular datasets and their ability to model intricate relationships among the 22 acoustic features. Following the individual training phase, two advanced ensemble techniques such as stacking and soft-voting were employed to combine the probability outputs of the base learners.

The pipeline was implemented in Python with scikit-learn, imbalanced-learn, XGBoost, LightGBM, CatBoost and SHAP, and run on a laptop with an Intel Core i7-8665U CPU at 1.90 GHz and 32 GB of RAM. Training time is the wall-clock time needed to fit a model on the 440 training samples (including the evaluation of training metrics), and inference time the time needed to predict the labels and probabilities of the 49 test recordings. Threshold-dependent metrics use each classifier's default decision rule, i.e., a predicted PD probability of 0.5 (the sign of the decision function for the SVM).

5. Results and Discussion

In this section, the performance of multiple machine learning models trained on the UCI Parkinson's disease dataset is presented and analyzed. Following the protocol of Section 4, the results cover test performance (Section 5.1), cross-validation stability (Section 5.2), confidence intervals (Section 5.3), statistical significance (Section 5.4), clinical error analysis, generalization gap, computational cost (Section 5.7) and explainability. Every algorithm was assessed using the same standardized performance measures such as accuracy (ACC), area under the ROC curve (AUC), precision, recall, specificity, F1-score, and Cohen's Kappa.

5.1. Test Performance

Every algorithm was trained on the SMOTE-balanced and augmented training subset and assessed on the subject-disjoint test subset (49 recordings: 37 PD and 12 healthy) using accuracy (ACC), AUC, precision, recall, F1-score and Cohen's Kappa, whose numerical outcomes are summarized in Table 5. The visualization of models performance using ROC curves and confusion matrices are summarized in Figure 12 and Figure 13.

Under the revised subject-disjoint protocol, no single classifier dominates: CatBoost, XGBoost, Random Forest, soft Voting and Stacking share the best results (95.92% accuracy, 97.30% precision, recall and F1-score, $\kappa = 88.96\%$) and produce identical confusion matrices (36 true positives, one false negative, one false positive, 11 true negatives), so the tree-based learners and the two ensembles built on them take the same decision on every test recording and differ only in how they rank them, Random Forest reaching the highest AUC (99.55%). This group is therefore a robust top tier rather than a single winner. LightGBM follows closely, with the same recall and the second highest AUC (98.42%) but one additional false positive, which lowers its precision, accuracy and kappa.

Table 5. Test Performance Metrics of Different Models (%).

Model	Accuracy	AUC	Precision	Recall	F1-score	Cohen's Kappa
CatBoost	**95.92**	97.52	**97.30**	**97.30**	**97.30**	**88.96**
XGBoost	**95.92**	97.97	**97.30**	**97.30**	**97.30**	**88.96**
LightGBM	93.88	98.42	94.74	**97.30**	96.00	82.97
RF	**95.92**	**99.55**	**97.30**	**97.30**	**97.30**	**88.96**
LR	87.76	92.12	91.89	91.89	91.89	66.89
DT	91.84	91.78	97.14	91.89	94.44	79.10
SVM	89.80	94.14	90.00	**97.30**	93.51	69.86
KNN	77.55	82.55	88.24	81.08	84.51	44.03
MLP	91.84	97.30	97.14	91.89	94.44	79.10
Voting	**95.92**	96.62	**97.30**	**97.30**	**97.30**	**88.96**
Stacking	**95.92**	97.07	**97.30**	**97.30**	**97.30**	**88.96**

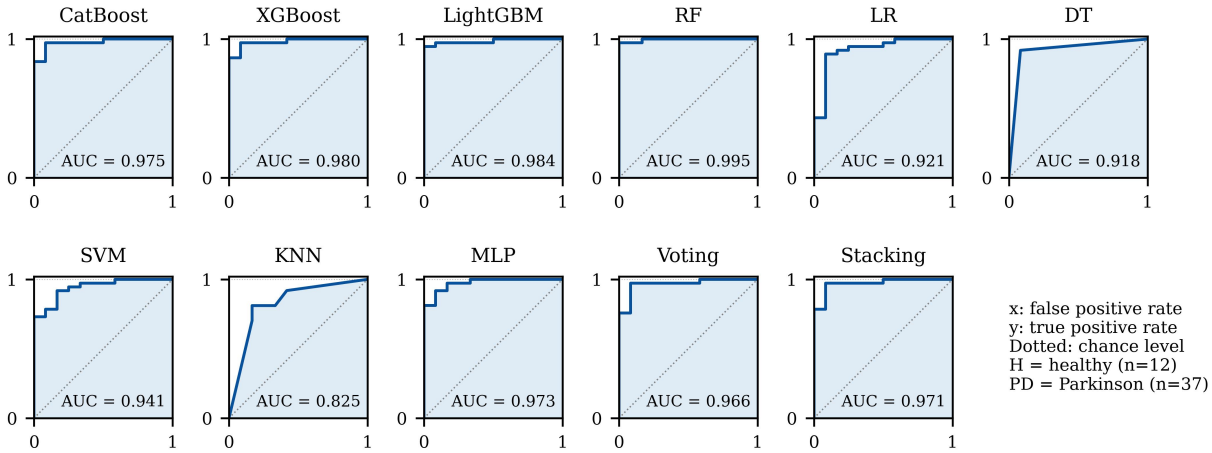


Figure 12. Receiver operating characteristic curves of all evaluated machine learning models on the test subset.

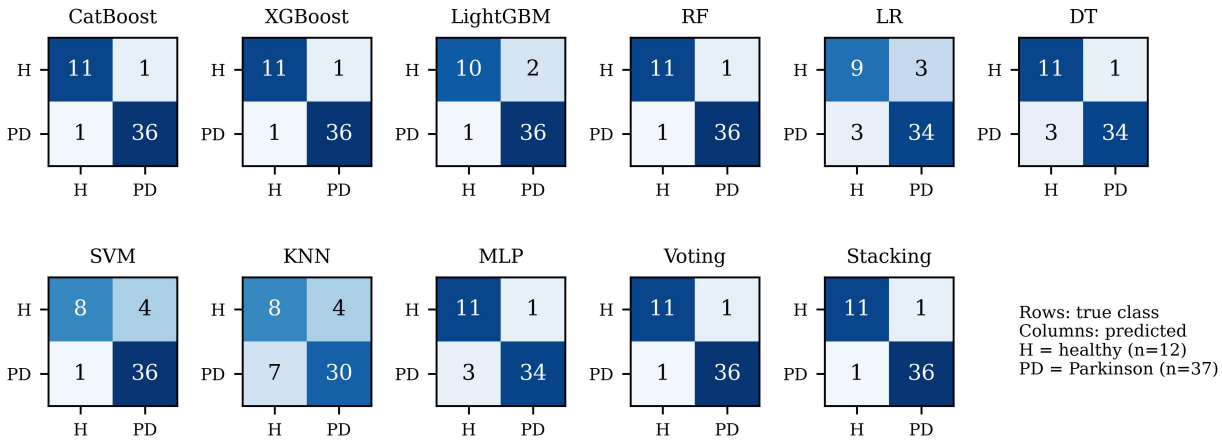


Figure 13. Confusion matrices corresponding to all evaluated classification models on the test subset.

The MLP, best in the original single 80/20 split, drops to 91.84% accuracy and 91.89% recall (three missed PD recordings), illustrating how sensitive the ranking of flexible, high-variance learners is to the composition of a small test set. The Decision Tree reaches the same thresholded results as the MLP but with a much weaker AUC (91.78%), reflecting the coarse probability estimates of a single depth-limited tree, whereas Logistic Regression is limited by its linear decision function ($\kappa = 66.89\%$). SVM favors sensitivity (97.30%) at the cost of specificity, misclassifying four of the twelve healthy recordings, and KNN is the weakest model overall (77.55% accuracy, seven missed PD recordings): as an instance-based method relying on distances in the raw feature space, it is the most affected by speaker-to-speaker variability when test speakers are unseen during training. The complete values are given in Table 5.

The plots of Figures 14–16 confirm that the ensemble and gradient-boosting approaches form the leading group, with Random Forest providing the best discrimination. Since several of these differences correspond to only one to three test recordings, they are examined below in terms of cross-validation stability (Section 5.2), confidence intervals (Section 5.3) and statistical significance (Section 5.4).

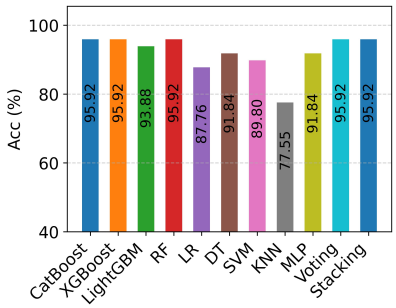


Figure 14. Comparative classification accuracy (%) achieved by the evaluated machine learning models.

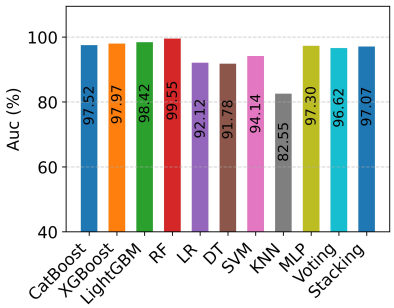


Figure 15. Comparison of area under the ROC curve values obtained by the evaluated models.

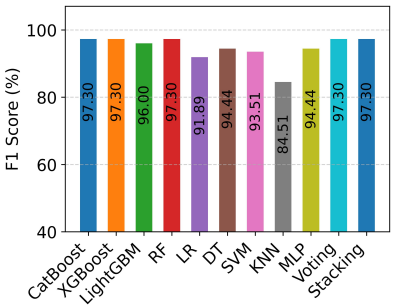


Figure 16. Comparative F1-score results of the evaluated classifiers.

5.2. Cross-Validation Stability

Table 6 and Figure 17 report the 5-fold subject-grouped cross-validation results on the training subset.

Table 6. 5-fold cross-validation performance on the training subset (mean ± standard deviation, %), sorted by mean accuracy.

Model	**Acc.**	**AUC**	**Prec.**	**Sens.**	**F1**	κ
CatBoost	90.39±3.42	96.25±0.83	94.35±5.78	93.64±6.80	93.60±2.33	73.33±11.35
XGBoost	86.99±4.56	94.76±1.93	90.99±6.55	92.73±5.45	91.52±2.67	62.59±17.43
LightGBM	89.72±3.78	96.83±1.10	93.23±5.32	93.64±3.64	93.26±2.23	71.06±13.51
RF	91.13±4.03	94.71±2.57	93.85±3.39	94.55±5.30	94.07±2.86	76.18±9.52
LR	74.64±3.59	84.09±5.32	88.33±5.94	77.27±4.98	82.12±2.28	37.84±14.42
DT	85.01±7.76	81.30±6.39	90.61±3.36	89.09±8.43	89.74±5.70	61.60±17.18
SVM	83.63±8.04	92.22±7.90	93.31±4.95	84.55±10.98	88.25±6.11	61.23±16.79
KNN	87.68±6.38	96.69±4.09	**98.18±3.64**	85.45±8.81	91.04±4.96	71.64±13.12
MLP	89.10±6.85	97.56±2.02	95.13±3.18	90.00±7.27	92.40±4.96	73.15±15.69
Voting	91.79±3.48	96.31±2.00	95.47±2.66	93.64±4.64	94.46±2.48	78.48±8.22
Stacking	**93.15±3.78**	**98.00±0.76**	94.74±3.26	**96.36±3.40**	**95.49±2.49**	**81.11±10.38**

The stacking ensemble is the best and one of the most stable models, with the highest mean accuracy ($93.15 \pm 3.78\%$), AUC ($98.00 \pm 0.76\%$) and sensitivity ($96.36 \pm 3.40\%$), followed by soft voting, Random Forest and CatBoost. Boosting models display the lowest AUC variability (±0.83–1.10%), whereas SVM, DT, KNN and MLP show fold-to-fold accuracy standard deviations of 6.4–8.0 points, roughly twice

those of the ensembles. The MLP reaches a high mean AUC (97.56%) but only $89.10 \pm 6.85\%$ accuracy, showing how a single small split can overstate the merit of a high-variance learner, while logistic regression is clearly the weakest model under cross-validation (74.64% accuracy, $\kappa = 37.84\%$), confirming that the class boundary in this feature space is markedly non-linear.

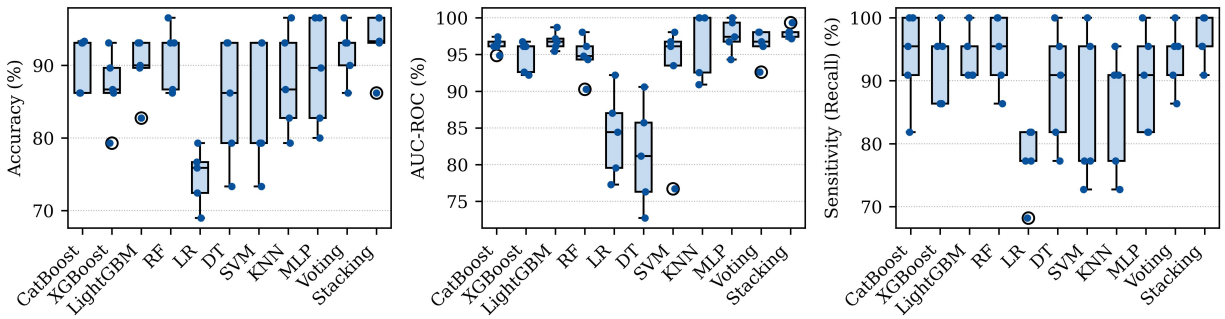


Figure 17. Distribution of 5-fold cross-validation accuracy (left) and AUC-ROC (right) for all models; points denote individual folds.

5.3. Confidence Intervals of Test Performance

Because the test subset contains only 49 recordings, the point estimates of Table 5 are complemented by 95% subject-level bootstrap confidence intervals, reported in Table 7 together with the specificity of each model.

Table 7. Test performance of the evaluated models (%, $n = 49$: 37 PD, 12 healthy) with 95% subject-level (cluster) bootstrap confidence intervals in brackets ($B = 2000$). †Specificity intervals are Wilson score intervals.

Model	Acc.	AUC	Sens. (Recall)	Spec.†	F1	κ
CatBoost	**95.92** [91.67, 100]	97.52 [94.59, 100]	**97.30** [91.67, 100]	**91.67** [64.6, 98.5]	**97.30**	**88.96**
XGBoost	**95.92** [91.67, 100]	97.97 [95.60, 100]	**97.30** [91.67, 100]	**91.67** [64.6, 98.5]	**97.30**	**88.96**
LightGBM	93.88 [89.58, 96.08]	98.42 [95.37, 100]	**97.30** [91.67, 100]	83.33 [55.2, 95.3]	96.00	82.97
RF	**95.92** [91.67, 100]	**99.55** [98.61, 100]	**97.30** [91.67, 100]	**91.67** [64.6, 98.5]	**97.30**	**88.96**
LR	87.76 [82.35, 93.75]	92.12 [86.49, 98.15]	91.89 [86.11, 97.30]	75.00 [46.8, 91.1]	91.89	66.89
DT	91.84 [85.71, 97.96]	91.78 [86.11, 98.65]	91.89 [86.11, 97.44]	**91.67** [64.6, 98.5]	94.44	79.10
SVM	89.80 [83.33, 96.00]	94.14 [87.39, 99.10]	**97.30** [91.67, 100]	66.67 [39.1, 86.2]	93.51	69.86
KNN	77.55 [59.18, 92.00]	82.55 [66.67, 96.40]	81.08 [58.33, 97.44]	66.67 [39.1, 86.2]	84.51	44.03
MLP	91.84 [83.33, 98.04]	97.30 [93.52, 100]	91.89 [80.56, 100]	**91.67** [64.6, 98.5]	94.44	79.10
Voting	**95.92** [91.67, 100]	96.62 [93.06, 100]	**97.30** [91.67, 100]	**91.67** [64.6, 98.5]	**97.30**	**88.96**
Stacking	**95.92** [91.67, 100]	97.07 [93.24, 100]	**97.30** [91.67, 100]	**91.67** [64.6, 98.5]	**97.30**	**88.96**

Two observations temper these numbers. First, the confidence intervals are wide: the lower accuracy bound of the top tier (91.67%) overlaps the point estimates of LightGBM, MLP and DT, and the specificity intervals derived from only 12 healthy recordings span more than 30 points. Second, test accuracy exceeds the cross-validated mean for ten of the eleven models (Δ from +2.8 points for Stacking to +13.1 points for LR), and every tree ensemble and the MLP reach 100% training accuracy. Because both the test subset and the validation folds are subject-disjoint from the data used for fitting, this gap is not caused by speaker overlap; it reflects the fact that the test subset contains only a handful of speakers (two of them healthy), so its composition strongly influences the estimate. The cross-validated values of Table 6 should therefore be regarded as the more conservative estimate of expected performance. Among the top-tier models, the stacking ensemble exhibits the smallest gap between test and CV accuracy (+2.77

points), which, with its low CV variance, supports it as the most robust choice rather than merely the most accurate one.

5.4. Statistical Significance of Model Differences

Table 8 summarizes the exact McNemar tests for the most discordant model pairs. Before correction, KNN differed from the five top-tier models ($n_{01} = 9$, $n_{10} = 0$, $p = 0.0039$), as well as from MLP ($p = 0.016$), LightGBM ($p = 0.021$), SVM ($p = 0.031$) and DT ($p = 0.039$).

After Holm correction for the 55 comparisons, however, *no* pairwise difference is statistically significant (smallest adjusted $p = 0.215$), the subject-clustered test yields even larger p-values (all Holm-adjusted $p = 1.0$), and the top-tier models have zero discordant predictions with respect to each other. On a 49-recording test set the eleven classifiers therefore cannot be strictly ranked: ensembles and boosting models form a leading group whose superiority over single learners is consistent across metrics and folds, but is not statistically demonstrable on the hold-out data. This is precisely why single-split accuracy differences of a few points, as frequently reported in the literature, should not be interpreted as evidence of methodological superiority.

Table 8. Exact McNemar tests with Holm correction (55 pairs) for the most discordant pairs. Top tier = Stacking, Voting, RF, CatBoost and XGBoost (identical test predictions). n_{01}: only the first model correct; n_{10}: only the second model correct.

Pair	n_{01}	n_{10}	p_{exact}	p_{Holm}	$p_{\text{clust.}}$	$p_{\text{clust.,Holm}}$
Top tier vs. KNN	9	0	0.004	0.215	0.083	1.000
MLP vs. KNN	7	0	0.016	0.781	0.071	1.000
LightGBM vs. KNN	9	1	0.021	1.000	0.088	1.000
SVM vs. KNN	6	0	0.031	1.000	0.157	1.000
DT vs. KNN	8	1	0.039	1.000	0.178	1.000
Top tier vs. LR	4	0	0.125	1.000	0.102	1.000
LR vs. KNN	6	1	0.125	1.000	0.166	1.000
Top tier (within)	0	0	1.000	1.000	1.000	1.000

5.5. Clinical Error Analysis and Decision Threshold

In voice-based PD screening, a false negative means that a patient is reassured and may not be referred to a neurologist, delaying treatment during the window in which it is most effective, whereas a false positive only leads to an additional examination; false negatives are therefore the more costly error. Table 9 details the error profile of each model. The top-tier models miss one PD recording out of 37 (false-negative rate 2.70%) and raise one false alarm out of 12 healthy recordings, whereas MLP and DT miss three (negative predictive value 78.57%, i.e., roughly one in five "healthy" predictions would actually be a patient). SVM illustrates the opposite trade-off (one missed patient but four false alarms), and KNN combines seven false negatives with four false positives (NPV 53.33%), which makes it unsuitable for screening in this configuration. The per-class F1-scores confirm that the minority healthy class remains the hardest to recognize even after SMOTE.

All threshold-dependent metrics above use the conventional cut-off of 0.5. The ROC curves (Figure 12) and the high AUC values of the top models (97–99.6%) indicate that lowering the threshold would recover the remaining false negative at the price of a few additional false positives, which is generally acceptable for a first-line screening tool whose positive outputs are confirmed by a neurologist, whereas a higher threshold could be used in a confirmatory setting where specificity is prioritized. Because the test set contains only 12 healthy recordings, any threshold optimized on it would be unstable; operating points should therefore be selected on cross-validated (out-of-fold) predictions for a target sensitivity (e.g., $\geq 95\%$) and then validated prospectively. Whether the missed PD case belongs to a subject with mild

Table 9. Test-set error analysis at the default threshold of 0.5 (37 PD, 12 healthy). BalAcc: balanced accuracy; PPV/NPV: positive/negative predictive value; $F1_H$: F1-score of the healthy class; MCC: Matthews correlation coefficient.

Model	TP	FN	FP	TN	BalAcc	PPV / NPV	$F1_H$	MCC
Top tier*	36	**1**	**1**	11	**94.48**	97.30 / 91.67	**91.67**	**0.890**
LightGBM	36	**1**	2	10	90.32	94.74 / 90.91	86.96	0.831
LR	34	3	3	9	83.45	91.89 / 75.00	75.00	0.669
DT	34	3	**1**	11	91.78	97.14 / 78.57	84.62	0.795
SVM	36	**1**	4	8	81.98	90.00 / 88.89	76.19	0.710
KNN	30	7	4	8	73.87	88.24 / 53.33	59.26	0.446
MLP	34	3	**1**	11	91.78	97.14 / 78.57	84.62	0.795

*Stacking, Voting, RF, CatBoost and XGBoost.

dysphonia should be examined together with clinical severity information, which is not available in the public dataset.

5.6. Ablation of the Ensemble Layer and Overfitting

Comparing each ensemble with its constituent learners provides a component-level ablation of the combination step. Under cross-validation, stacking improves mean accuracy by 2.02 points and kappa by 4.93 points over the best single base learner (RF) while reducing the AUC standard deviation from 2.57% to 0.76%, whereas soft voting yields a smaller gain (+0.66 accuracy points). On the test set both ensembles reproduce exactly the predictions of RF, CatBoost and XGBoost, so their benefit lies in stability rather than in hold-out accuracy. The comparison of training, cross-validation and test accuracy (Table 10) shows that all tree ensembles and the MLP memorize the SMOTE-balanced training set (100% training accuracy) and lose 7–13 points under cross-validation, whereas the more strongly regularized LR and SVM have smaller capacity but also lower accuracy. Regularization and fold-wise preprocessing are therefore essential with 146 training recordings, and the reported cross-validation variance is a necessary complement to any point estimate.

Table 10. Generalization gap: accuracy (%) on the training subset, in 5-fold subject-grouped cross-validation and on the test subset; Δ: Test − CV accuracy (points).

Model	Train	CV	Test	Δ
CatBoost	100.00	90.39	95.92	+5.53
XGBoost	100.00	86.99	95.92	+8.93
LightGBM	100.00	89.72	93.88	+4.15
RF	100.00	91.13	95.92	+4.79
LR	84.93	74.64	87.76	+13.11
DT	99.32	85.01	91.84	+6.83
SVM	90.41	83.63	89.80	+6.16
KNN	95.89	87.68	77.55	−10.13
MLP	100.00	89.10	91.84	+2.73
Voting	100.00	91.79	95.92	+4.13
Stacking	100.00	93.15	95.92	+2.77

Table 11. Computational cost of the evaluated models.

Model	Time (s)		
	Train	Test	XAI
CatBoost	11.035	0.015	8.91
XGBoost	0.629	0.024	2.30
LightGBM	0.336	0.017	3.83
RF	5.604	0.339	218.84
LR	0.094	**0.011**	1.54
DT	**0.023**	**0.011**	**0.41**
SVM	0.051	0.013	8.56
KNN	0.033	0.022	7.57
MLP	1.518	**0.011**	1.68
Voting	20.665	0.521	348.47
Stacking	106.225	0.393	349.31

5.7. Computational Cost

Table 11 reports the wall-clock cost of each model: the training time (fitting on the 440 training samples, including the evaluation of the training metrics), the inference time on the 49 test recordings and the time required by the explainability analysis. Training costs differ by more than three orders of magnitude. The

single learners DT, KNN, SVM and LR are trained almost instantly (0.023–0.094 s) and the histogram-based LightGBM and XGBoost stay below one second despite using 2000 trees, whereas the MLP, Random Forest and CatBoost need 1.5–11 s, the latter because of its ordered boosting scheme. The ensembles are the most costly: soft Voting (20.7 s) must train all nine base learners and Stacking (106.2 s) additionally re-trains them five times to produce out-of-fold predictions for its meta-learner, i.e., about 170 times slower than XGBoost although both produce identical test predictions. Inference, in contrast, is fast everywhere: nine models classify the 49 recordings in less than 0.025 s (below 0.5 ms per recording), and only Random Forest, Stacking and Voting, which evaluate thousands of fully grown trees, reach 7–11 ms per recording, still negligible for screening. The cost of explanation follows the prediction cost, from 0.4–4 s for DT, LR, MLP, XGBoost and LightGBM and 7.6–8.9 s for KNN, SVM and CatBoost, to 218.8 s for Random Forest and about 350 s for Voting and Stacking, whose KernelSHAP estimates propagate through all nine base learners.

For low-resource clinical settings, such as primary-care centers or smartphone-based screening without GPU or cloud access, XGBoost and CatBoost offer the best accuracy–cost trade-off among the top-tier models: they reach the best test performance with training times of seconds, sub-millisecond inference, compact tree models and explanations computed in less than 10 s. Random Forest provides the best AUC and the most faithful explanations of the top tier but is more costly to explain, whereas Stacking remains preferable when training can be performed centrally and robustness across data partitions is the priority, since only its inference cost (≈ 8 ms per recording) is incurred at the point of care.

5.8. Explainability and Vocal Biomarkers

The consensus importance ranking of the 22 acoustic features identifies as most influential predictors the average fundamental frequency MDVP:Fo (0.714), pitch period entropy PPE (0.640), the five-point amplitude perturbation quotient Shimmer:APQ5 (0.589), detrended fluctuation analysis DFA (0.490), local shimmer MDVP:Shimmer (0.482), the maximum fundamental frequency MDVP:Fhi (0.459), correlation dimension D2 (0.427) and spread2 (0.421). Three groups of biomarkers therefore dominate: (i) pitch level and pitch-control irregularity (Fo, Fhi, PPE, spread1/spread2), reflecting the reduced laryngeal control and monopitch speech typical of hypokinetic dysarthria; (ii) amplitude instability (shimmer measures), associated with incomplete and irregular vocal-fold closure; and (iii) nonlinear dynamical measures (DFA, D2, RPDE), which capture turbulent, aperiodic noise in the voice signal. These findings agree with the original dysphonia analysis of Little et al. [6] and with the SHAP analysis of Abioye et al. [8].

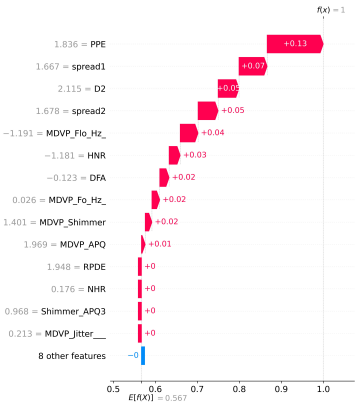


Figure 18. SHAP waterfall explanation for an MLP prediction, showing the contribution of the most influential features.

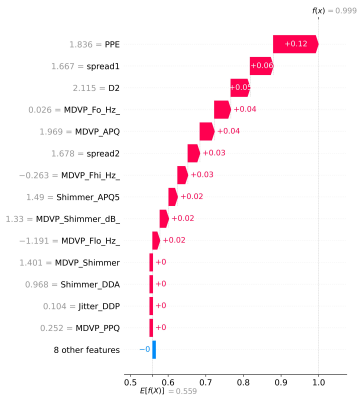


Figure 19. SHAP waterfall explanation for a Voting ensemble prediction, showing feature-level contributions to the model output.

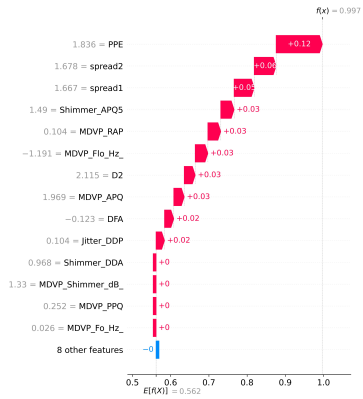


Figure 20. SHAP waterfall explanation for a Stacking ensemble prediction, illustrating the contribution of individual features to the prediction.

The jitter family (MDVP:Jitter(%), RAP, PPQ, Jitter:DDP, Jitter(Abs)) receives low individual importance (0.02–0.22), which does not mean that frequency perturbation is irrelevant: the five variants are highly inter-correlated (Jitter:DDP is, by definition, three times MDVP:RAP), so tree-based models select only one of them at each split and the importance is diluted across the family. The same collinearity explains the discrepancy between the two boosting models and motivates reporting a consensus ranking rather than that of a single model. Finally, the prominence of MDVP:Fo must be interpreted with caution, because fundamental frequency also depends strongly on sex and individual anatomy; in a small cohort it may partly act as a speaker- or sex-related cue, which is why the subject-disjoint evaluation adopted here, and future validation on sex-balanced cohorts, are essential.

Table 12. Quality metrics of the SHAP explanations (30 test recordings). ↑: higher is better; ↓: lower is better; → 0: closer to zero is better. FA: top-10 SHAP feature agreement; ρ: Spearman correlation between SHAP and LIME rankings. Computation times are given in Table 11.

Model	**Fidelity↑**	**Infidelity↓**	**Sparsity↑**	**Stability↓**	**Compr.↑**	**Suff.→ 0**	**FA↑**	ρ↑
CatBoost	0.511	0.086	0.555	**0.304**	−0.251	−0.134	0.8	0.801
XGBoost	0.370	0.129	0.570	0.336	−0.203	−0.186	**0.9**	0.843
LightGBM	0.360	0.396	0.573	0.383	−0.236	−0.117	0.7	0.506
RF	0.666	**0.002**	0.508	0.318	−0.068	−0.111	0.8	0.735
LR	**0.812**	0.003	0.716	0.404	**−0.048**	−0.028	0.8	0.764
DT	0.761	0.097	**0.813**	0.434	−0.333	**0.000**	**0.9**	**0.867**
SVM	0.487	**0.002**	0.704	0.459	−0.130	−0.114	0.6	0.593
KNN	0.430	**0.002**	0.701	0.578	−0.311	−0.081	0.6	0.395
MLP	0.488	0.004	0.718	0.514	−0.209	0.006	0.7	0.657
Voting	0.533	0.003	0.696	0.532	−0.176	−0.142	0.6	0.565
Stacking	0.406	0.004	0.700	0.628	−0.223	−0.209	0.7	0.798

Table 12 quantifies the quality of the SHAP explanations. Intrinsically simple models provide the most faithful explanations (LR 0.812, DT 0.761, followed by RF 0.666), and DT also yields the sparsest attributions. Among the top-tier models, RF offers the best compromise between accuracy and explanation quality, with the lowest infidelity and the least negative comprehensiveness; the larger infidelity of the boosting models is partly a scale effect, since TreeSHAP attributes their raw log-odds output whereas infidelity compares attributions with changes in predicted probability. The exact TreeSHAP explanations are also the most stable (CatBoost, RF and XGBoost show the smallest change of attributions under input noise), whereas the sampling-based KernelSHAP explanations of Stacking, KNN and Voting are the least stable, and independent explanation methods agree best for DT and XGBoost and worst for KNN and LightGBM. Comprehensiveness is negative for all models, i.e., replacing the four most important features by the training mean does not decrease the predicted PD probability, and sufficiency is close to zero, meaning that the top four features alone preserve the prediction. Two properties of this dataset explain these values: masking to the mean of a class-balanced training set moves a recording towards a region that several models already score as PD, and the redundancy among jitter and shimmer variants allows correlated features to compensate for masked ones. Removal-based metrics should therefore be interpreted relative to the baseline rather than as evidence that the selected features are irrelevant. Overall, the analysis reveals an accuracy–interpretability trade-off: RF, and to a lesser extent XGBoost and CatBoost, combine top-tier accuracy with stable and comparatively consistent explanations, and are therefore the most suitable candidates when clinicians require both.

5.9. Comparison with Previous Studies

Several earlier works on the same dataset report higher accuracies than ours, e.g., 100% [7, 8], 99.47% [9] or 98.31% [12], but these values are not directly comparable with the present results. As analyzed in Section 2, they were typically obtained from a single recording-level split in which recordings of the same speaker appear in training and test data, with a small test set and without confidence intervals,

whereas the present evaluation is subject-disjoint; on such a test set the difference between 95.92% and 100% corresponds to two recordings and lies within our 95% confidence interval. Moreover, the gap we observe between hold-out and cross-validated accuracy (up to 13 points) shows how strongly a single favorable split can inflate reported performance. The contribution of the present benchmark is therefore not a higher accuracy, but a more reliable characterization of what can be expected from each family of models: ensemble and boosting methods are consistently the most accurate and stable, their advantage over single learners is not statistically significant on 49 test recordings, a false-negative rate of about 3% is achievable at the default threshold, and the decisions rely on physiologically meaningful pitch, shimmer and nonlinear dysphonia markers.

5.10. Limitations

Several limitations should be considered. (i) Although the train/test split and the cross-validation are subject-disjoint, the test subset contains few speakers and only one fixed partition was evaluated; repeated subject-level splits or leave-one-subject-out validation would further reduce the dependence on partition composition, and the standardization used during cross-validation was estimated on the whole training subset. (ii) The cohort includes only 31 speakers (8 healthy) recorded under controlled conditions, so external validation on larger and more diverse cohorts (languages, microphones, disease stages) is required. (iii) Only sustained vowel phonations and precomputed MDVP-based features are used. (iv) Model configurations were fixed rather than tuned, the default decision threshold was not optimized, and a full component ablation (without SMOTE or augmentation) was not performed. (v) The explanation metrics are affected by feature collinearity, and the analysis does not include clinical severity information.

5.11. Summary

In summary, the stacking ensemble emerged as the most robust classifier, combining the best cross-validated accuracy, AUC and sensitivity with low variance, while RF, CatBoost, XGBoost and soft voting reached the same hold-out performance with one false negative among 37 patients. Ensemble methods provide strong, reliable alternatives suitable for practical deployment, with XGBoost and CatBoost offering the best accuracy–cost trade-off for low-resource settings. While simpler models such as logistic regression and decision trees lag behind, they may still serve specialized roles where interpretability or low computational cost is required. However, for maximizing predictive performance, the ensemble approaches clearly stand out, although their superiority over single learners could not be confirmed statistically on the present test set.

6. Conclusion

The present investigation evaluated eleven classical, boosting, neural and ensemble classifiers for voice-based Parkinson's disease detection on the Oxford dataset under a leakage-controlled and statistically grounded protocol. The stacking ensemble achieved the best and most stable cross-validated performance ($93.15 \pm 3.78\%$ accuracy, $98.00 \pm 0.76\%$ AUC and $96.36 \pm 3.40\%$ sensitivity), and five models (Stacking, Voting, RF, CatBoost and XGBoost) reached 95.92% test accuracy (95% CI: 91.67–100%), 97.30% sensitivity and 91.67% specificity, missing a single PD recording out of 37. Random Forest obtained the highest test AUC (99.55%). Beyond these figures, the study shows that (i) single-split accuracies on this dataset are optimistic, as test accuracy exceeded the cross-validated estimate by up to 13 points; (ii) after Holm correction none of the 55 pairwise differences is statistically significant, so the leading models should be regarded as a statistically equivalent tier rather than strictly ranked; (iii) false negatives, the most critical errors in screening, vary from one (top tier, SVM, LightGBM) to seven (KNN) among 37 patients, highlighting the value of sensitivity-oriented model selection and threshold tuning; (iv) fundamental frequency, pitch period entropy, shimmer and nonlinear measures such as DFA and D2 are the dominant vocal biomarkers; and (v) XGBoost and CatBoost combine top-tier performance with sub-millisecond

inference, making them attractive for low-resource deployment, whereas logistic regression is limited by the non-linear structure of the data and single decision trees and KNN are the least reliable.

Taken together, the results highlight the substantial promise of machine learning, especially ensemble strategies, for delivering precise, dependable, and clinically meaningful voice-based PD screening. Future investigations should concentrate on subject-independent (leave-one-subject-out) validation, sensitivity-constrained threshold selection, physiologically informed augmentation on raw audio, full component ablations, and external validation on larger and more diverse patient cohorts to ensure robust generalization in real-world diagnostic settings.

REFERENCES

1. H. Elbehiery, M. A. Ebrahim, M. Eassa, A. Abdelhafeez, A. Omar, and H. Mahmoud, "Improving heart disease prediction accuracy through machine learning algorithms," *Statistics, Optimization & Information Computing*, vol. 14, no. 2, pp. 736–755, 2025. doi: 10.19139/soic-2310-5070-2319.
2. O. K. M. El-Shahat, B. S. Tawfik, and M. N. Refaie, "Optimal gene selection and machine learning framework for Alzheimer's disease prediction using transcriptomic data," *Statistics, Optimization & Information Computing*, vol. 14, no. 5, pp. 2276–2296, 2025. doi: 10.19139/soic-2310-5070-2723.
3. T. AbuAin, "A hybrid approach of long short term memory and transformer models for speech emotion recognition," *Statistics, Optimization & Information Computing*, vol. 14, no. 1, pp. 340–351, 2025. doi: 10.19139/soic-2310-5070-2521.
4. M. Little, "Parkinsons," *UCI Machine Learning Repository*, 2007. doi: 10.24432/C59C74.
5. M. Little, P. McSharry, S. Roberts, D. Costello, and I. Moroz, "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection," *Nature Precedings*, pp. 1–1, 2007. doi: 10.1038/npre.2007.326.1
6. M. A. Little, P. E. McSharry, E. J. Hunter, J. Spielman, and L. O. Ramig, "Suitability of Dysphonia Measurements for Telemonitoring of Parkinson's Disease," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 4, pp. 1015–1022, 2009. doi: 10.1109/TBME.2008.2005954
7. M. Hariharan, K. Polat, and R. Sindhu, "A new hybrid intelligent system for accurate detection of Parkinson's disease," *Computer Methods and Programs in Biomedicine*, vol. 113, no. 3, pp. 904–913, 2014. doi: 10.1016/j.cmpb.2014.01.004
8. O. A. Abioye, A. Evwiekpaefe, P. Odion, and O. J. Awujoola, "Improving Parkinson's diagnosis: A stacked ensemble learning with vocal biomarkers and data balancing," *IT Journal Research and Development*, vol. 10, no. 1, pp. 67–78, 2025. doi: 10.25299/itjrd.2025.22310
9. D. K. Saha and T. D. Nath, "A lightweight CNN-based ensemble approach for early detecting Parkinson's disease with enhanced features," *International Journal of Speech Technology*, 2025. doi: 10.1007/s10772-025-10182-4
10. G. Celik and E. Başaran, "Proposing a new approach based on convolutional neural networks and random forest for the diagnosis of Parkinson's disease from speech signals," *Applied Acoustics*, vol. 211, 109476, 2023. doi: 10.1016/j.apacoust.2023.109476
11. S. Srinivasan, P. Ramadass, S. K. Mathivanan, K. P. Selvam, B. D. Shivahare, and M. A. Shah, "Detection of Parkinson disease using multiclass machine learning approach," *Scientific Reports*, vol. 13, 12345, 2023. doi: 10.1038/s41598-024-64004-9
12. R. Alshammri, G. Alharbi, E. Alharbi, and I. Almubark, "Machine learning approaches to identify Parkinson's disease using voice signal features," *Frontiers in Artificial Intelligence*, vol. 6, 1084001, 2023. doi: 10.3389/frai.2023.1084001
13. D. Avci and A. Dogantekin, "An expert diagnosis system for Parkinson disease based on genetic algorithm-wavelet kernel-extreme learning machine," *Parkinson's Disease*, vol. 2016, Article ID 5264743, 2016. doi: 10.1155/2016/5264743
14. M. S. Roobini, Y. R. K. Reddy, U. S. Girish Royal, A. K. Singh, and K. Babu, "Parkinson's disease detection using machine learning," in *Proc. IEEE Conf. Publication*, 2022. doi: 10.1109/IC3IOT53935.2022.9768002
15. R. Lamba, T. Gulati, H. F. Alharbi, and A. Jain, "A hybrid system for Parkinson's disease diagnosis using machine learning techniques," *International Journal of Speech Technology*, vol. 24, no. 3, pp. 565–574, 2021. doi: 10.1007/s10772-021-09837-9
16. M. M. Nishat, T. Hasan, S. M. Nasrullah, F. Faisal, M. A. R. Asif, and M. A. Hoque, "Detection of Parkinson's Disease by Employing Boosting Algorithms," in *Proc. Joint 10th Int. Conf. Informatics, Electronics & Vision (ICIEV) and 5th Int. Conf. Imaging, Vision & Pattern Recognition (ICIVPR)*, IEEE, pp. 1–6, 2021. doi: 10.1109/ICIEVicIVPR52578.2021.9564108
17. R. Das, "A comparison of multiple classification methods for diagnosis of Parkinson disease," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1568–1572, 2010. doi: 10.1016/j.eswa.2009.06.040
18. A. Govindu and S. Palwe, "Early detection of Parkinson's disease using machine learning," *Procedia Computer Science*, vol. 218, pp. 249–261, 2023. doi: 10.1016/j.procs.2023.01.007
19. Shivangi, A. Johri, and A. Tripathi, "Parkinson disease detection using deep neural networks," in *Proc. IEEE Conf. Publication*, 2019. doi: 10.1109/IC3.2019.8844941
20. M. A. Islam, M. Z. H. Majumder, M. A. Hussein, K. M. Hossain, and M. S. Miah, "A review of machine learning and deep learning algorithms for Parkinson's disease detection using handwriting and voice datasets," *Heliyon*, vol. 10, no. 9, e25469, 2024. doi: 10.1016/j.heliyon.2024.e25469
21. T. Sood and P. Khandnor, "Classification of Parkinson's disease using various machine learning techniques," in *Proc. Advances in Computing and Data Sciences (ICACDS 2019), Communications in Computer and Information Science*, Springer, Singapore, vol. 1045, pp. 296–311, 2019. doi: 10.1007/978-981-13-9939-8_27

22. T. Kumar, P. Sharma, and N. Prakash, "Comparison of machine learning models for Parkinson's disease prediction," in *Proc. 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, IEEE, pp. 480–485, 2020. doi: 10.1109/UEMCON51285.2020.9298033
23. R. G. Ramani and G. Sivagami, "Parkinson disease classification using data mining algorithms," *International Journal of Computer Applications*, vol. 32, no. 9, pp. 17–22, 2011. https://www.ijcaonline.org/archives/volume32/number9/3932-5571/
24. S. Grover, S. Bhartia, Akshama, A. Yadav, and K. R. Seeja, "Predicting Severity of Parkinson's Disease Using Deep Learning," *Procedia Computer Science*, vol. 132, pp. 1788–1794, 2018. doi: 10.1016/j.procs.2018.05.154
25. G. J. Rani, M. F. Hashmi, and G. Muhammad, "Variational mode decomposition and empirical wavelet transform-based feature extraction and ensemble classifier for lower limb movement prediction with surface electromyography signal," *IEEE Access*, vol. 12, pp. 45561–45573, 2024. doi: 10.1016/j.apacoust.2023.109476
26. B. E. Sakar, M. E. Isenkul, C. O. Sakar, A. Sertbas, F. Gurgen, S. Delil, H. Apaydin, and O. Kursun, "Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 4, pp. 828–834, 2013. doi: 10.1109/JBHI.2013.2245674.
27. S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, "Leakage in data mining: Formulation, detection, and avoidance," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 4, article 15, 2012. doi: 10.1145/2382577.2382579.
28. S. Varma and R. Simon, "Bias in error estimation when using cross-validation for model selection," *BMC Bioinformatics*, vol. 7, article 91, 2006. doi: 10.1186/1471-2105-7-91.
29. T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998. doi: 10.1162/089976698300017197.
30. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 4765–4774, 2017.
31. D. R. Cox, *The regression analysis of binary sequences*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), vol. 20, no. 2, pp. 215–232, 1958. doi: 10.1111/j.2517-6161.1958.tb00292.x.
32. D. W. Hosmer Jr., S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, John Wiley & Sons, 2013. doi: 10.1080/00401706.1992.10485291.
33. T. Cover and P. Hart, *Nearest neighbor pattern classification*, IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21–27, 1967. doi:. doi: 10.1109/TIT.1967.1053964.
34. N. S. Altman, *An introduction to kernel and nearest-neighbor nonparametric regression*, The American Statistician, vol. 46, no. 3, pp. 175–185, 1992. doi: 10.1080/00031305.1992.10475879.
35. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning representations by back-propagating errors*, Nature, vol. 323, no. 6088, pp. 533–536, 1986. doi: 10.1038/323533a0.
36. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 2002. doi: 10.1109/5.726791.
37. C. Cortes and V. Vapnik, *Support-vector networks*, Machine Learning, vol. 20, no. 3, pp. 273–297, 1995. doi: 10.1007/BF00994018.
38. J. R. Quinlan, *Induction of decision trees*, Machine Learning, vol. 1, no. 1, pp. 81–106, 1986. doi: 10.1007/BF00116251.
39. T. G. Dietterich, *Ensemble methods in machine learning*, International Workshop on Multiple Classifier Systems, pp. 1–15, 2000, Springer. doi: 10.1007/3-540-45014-9_1.
40. R. Polikar, *Ensemble based systems in decision making*, IEEE Circuits and Systems Magazine, vol. 6, no. 3, pp. 21–45, 2006. doi: 10.1109/MCAS.2006.1688199.
41. L. Breiman, *Random forests*, Machine Learning, vol. 45, no. 1, pp. 5–32, 2001. doi: 10.1023/A:1010933404324.
42. T. Chen and C. Guestrin, *XGBoost: A scalable tree boosting system*, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794, 2016. doi: 10.1145/2939672.2939785. https://github.com/dmlc/xgboost
43. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, *LightGBM: A highly efficient gradient boosting decision tree*, Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017. https://github.com/Microsoft/LightGBM.
44. L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, *CatBoost: unbiased boosting with categorical features*, Advances in Neural Information Processing Systems (NeurIPS), vol. 31, 2018. https://github.com/catboost/catboost
45. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002. doi: 10.1613/jair.953.
46. B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, Chapman & Hall/CRC, New York, 1993.
47. Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947. doi: 10.1007/BF02295996.
48. V. L. Durkalski, Y. Y. Palesch, S. R. Lipsitz, and P. F. Rust, "Analysis of clustered matched-pair data," *Statistics in Medicine*, vol. 22, no. 15, pp. 2417–2428, 2003. doi: 10.1002/sim.1438.
49. S. Holm, "A simple sequentially rejective multiple test procedure," *Scandinavian Journal of Statistics*, vol. 6, no. 2, pp. 65–70, 1979.
50. C.-K. Yeh, C.-Y. Hsieh, A. S. Suggala, D. I. Inouye, and P. Ravikumar, "On the (in)fidelity and sensitivity of explanations," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.
51. J. DeYoung, S. Jain, N. F. Rajani, E. Lehman, C. Xiong, R. Socher, and B. C. Wallace, "ERASER: A benchmark to evaluate rationalized NLP models," in *Proc. 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4443–4458, 2020. doi: 10.18653/v1/2020.acl-main.408.
52. S. Krishna, T. Han, A. Gu, J. Pombra, S. Jabbari, S. Wu, and H. Lakkaraju, "The disagreement problem in explainable machine learning: A practitioner's perspective," *arXiv preprint* arXiv:2202.01602, 2022.