Türk Mühendislik Araştırma ve Eğitimi Dergisi
**Turkish Journal of Engineering Research and Education**
4(2), 140-160, (2025)

**RESEARCH ARTICLES**

# Predicting Honeybee Colony Health Using Weather and Apiary Data with Machine Learning

## *Makine Öğrenmesi ile Hava Durumu ve Arıcılık Verilerini Kullanarak Bal Arısı Koloni Sağlığının Tahmini*

**Ümit Yılmaz**
**Balıkesir Üniversitesi, Bigadiç Meslek Yüksekokulu, Yönetim ve Organizasyon Bölümü, Balıkesir, Türkiye.**

**Received / Geliş: 05.10.2025**
**Accepted / Kabul: 24.10.2025**
***Corresponding Author:** **Ümit Yılmaz** umityilmaz@balikesir.edu.tr

**ABSTRACT:** Honey bee colonies are essential for global food security but continue to suffer heavy losses from interacting biological and environmental stressors. Predicting colony health is therefore a priority for sustainable apicultural management. This study examines the feasibility of forecasting honey bee colony health using weather and seasonal variables together with field assessments from the Healthy Colony Checklist. A dataset of 1,277 inspections from apiaries in North Carolina and Utah, integrated with meteorological records from nearby stations, was analyzed. Engineered features included vapor pressure deficit, temperature–humidity interactions, wind energy estimates, and seasonal encodings. The prediction was structured as a binary classification task (healthy vs. unhealthy). Several machine learning models were tested, emphasizing tree-based ensembles such as Random Forest, Extra Trees, Light Gradient Boosting Machine, Categorical Boosting, Gradient Boosting, Histogram-based Gradient Boosting, and Extreme Gradient Boosting. Ensemble strategies, including optimized soft voting and stacking, were also applied. Results showed accuracies of 75–76% with ROC AUC values near 0.80. Precision exceeded 0.70, while recall remained modest (~0.55). Seasonality was the dominant predictor, with weather indicators providing complementary value. Findings confirm the usefulness of agrometeorological data for decision-support in apiculture but also highlight the limits of weather-only models. Incorporating hive-level biological and management factors with advanced learning methods is recommended.

**Keywords:** Honey bee health, healthy colony checklist, machine learning, ensemble methods

**ÖZ:** Bal arısı kolonileri, küresel gıda güvenliği için hayati öneme sahiptir; ancak biyolojik ve çevresel stres faktörlerinin etkileşimi nedeniyle yüksek kayıplar yaşamaya devam etmektedir. Bu nedenle koloni sağlığının tahmin edilmesi, sürdürülebilir arıcılık yönetimi açısından öncelikli bir konudur. Bu çalışma, hava durumu ve mevsimsel değişkenlerle birlikte Sağlıklı Koloni Kontrol Listesi saha değerlendirmelerinden yararlanarak bal arısı koloni sağlığının öngörülebilirliğini incelemektedir. Kuzey Carolina ve Utah'taki arılıklardan 1.277 denetim ve yakın meteoroloji istasyonu kayıtları kullanılmıştır. Türetilen özellikler arasında buhar basıncı açığı, sıcaklık–nem etkileşimleri, rüzgâr enerjisi tahminleri ve mevsimsel kodlamalar bulunmaktadır. Tahmin görevi, ikili sınıflandırma (sağlıklı vs. sağlıksız) olarak kurgulanmıştır. Birçok makine öğrenmesi modeli test edilmiş, özellikle Rastgele Orman, Aşırı Rastgele Ağaçlar, Hafif Gradyan Artırma Makinesi, Kategorik Artırma, Gradyan Artırma, Histogram Tabanlı Gradyan Artırma ve Aşırı Gradyan Artırma gibi ağaç tabanlı topluluk yöntemleri üzerinde durulmuştur. Optimize edilmiş yumuşak oylama ve yığınlama gibi topluluk stratejileri de uygulanmıştır. Sonuçlar, doğruluk oranlarının %75–76 aralığında, ROC AUC değerlerinin ise 0,80'e yakın olduğunu göstermiştir. Hassasiyet %0,70'in üzerinde gerçekleşirken, duyarlılık düşük kalmıştır (~0,55). Mevsimsellik en baskın belirleyici olurken, yağış ve nem gibi hava durumu göstergeleri ek katkı sağlamıştır. Bulgular, tarımsal meteorolojik verilerin karar destek sistemlerinde yararlı olduğunu, ancak biyolojik ve yönetimsel değişkenlerin gelişmiş yöntemlerle bütünleştirilmesinin gerektiğini ortaya koymaktadır.

**Anahtar Kelimeler:** Bal arısı sağlığı, sağlıklı koloni kontrol listesi, makine öğrenmesi, topluluk yöntemleri

Ümit Yılmaz, Orcid: 0000-0003-4268-8598

## 1. INTRODUCTION

Honey bees (Apis mellifera) play a vital role in agriculture by pollinating a wide variety of fruit, nut, and vegetable crops, contributing an estimated USD 12–50 billion annually to U.S. agricultural production [1]. Despite their importance, beekeepers in recent years have experienced unsustainably high colony losses, often exceeding 40% annually [2]. These losses pose a threat not only to honey production but also to pollination services essential for global food security [3]. Multiple interacting stressors contribute to this decline, including pests, pathogens, pesticides, nutritional deficits, and extreme weather conditions [4]. Weather and climate variability are particularly influential, as harsh winters, prolonged droughts, or excessive rainfall can disrupt foraging, thermoregulation, and overall colony resilience [5-7]. Seasonal dynamics also inherently shape colony health: colonies typically weaken in late winter or early spring, build to peak strength in summer, and face significant challenges during overwintering [8].

Routine hive inspections remain a fundamental tool for monitoring colony status. To standardize such assessments, the Healthy Colony Checklist (HCC) was introduced by Cazier, et al. [9]. The HCC evaluates six critical criteria during an inspection: presence of all brood stages, sufficient adult bee population, a productive queen, adequate food and pollen stores, absence of excessive stressors, and sufficient comb space. Each criterion is scored as present (1) or absent (0). If all six are satisfied, the colony is classified as "Healthy"; otherwise, it is "Unhealthy." This binary health outcome provides a biologically interpretable indicator directly linked to colony viability. However, the HCC represents only a point-in-time observation, whereas colony health is dynamic and highly dependent on external environmental conditions.

Recent advances in precision apiculture increasingly emphasize the integration of hive inspection data with environmental and sensor-based measurements to achieve more accurate predictions of colony health. Carlini, et al. [10] demonstrated the diagnostic potential of gut microbial profiling, showing that specific bacterial taxa such as Commensalibacter and Snodgrassella were strongly associated with winter survival, with machine learning models achieving The Receiver Operating Characteristic – Area Under the Curve (ROC-AUC) > 0.90. This indicates that microbiome analysis can serve as a powerful biomarker for colony resilience. Hammami and Abdulaziz [11] advanced this direction by developing BeeBetter, a multimodal Artificial Intelligence (AI) framework leveraging Internet of Things (IoT) sensors and deep learning (YOLOv8, VGGNet, Short-Time Fourier Transform-Convolution Recurrent Neural Network), which reached F1-scores of 96% in image-based detection tasks, confirming the promise of smart hive technologies.

Similarly, Liang [12] highlighted the value of multimodal fusion by proposing an Attention-based Multimodal Neural Network that combined visual and audio data, outperforming single-modality Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) models by as much as 33%, and achieving over 92% accuracy across multiple health states. Large-scale collaborative initiatives have also reinforced the need for harmonized methodologies; the B-GOOD project [13] introduced a standardized Health Status Index based on integrated sensor, field, and laboratory data, demonstrating how consistent multi-source datasets can enable predictive models at scale.

Complementary research has focused on specialized modalities. Torky, et al. [14] achieved 95% accuracy using a MobileNet+Adam framework to detect hive abnormalities such as Varroa destructor and missing queens, outperforming deeper architectures like ResNet50 and GoogleNet. Zhu, et al. [15] explored acoustic and environmental predictors of overwinter survival, with bagging-based anomaly detection achieving ROC-AUC of 0.73, underscoring the predictive value of hive power audio features. Finally, Rafael Braga, et al. [16] combined in-hive sensors, weather variables, and inspection records in supervised learning models, with Random Forest (RF) reaching up to 98% accuracy, thereby validating the effectiveness of hybrid approaches.

Taken together, these contributions illustrate rapid progress in applying AI, IoT, and multimodal data to honey bee health monitoring. However, most

existing approaches are either resource-intensive (e.g., sensor-rich systems) or domain-specific (e.g., microbiome-only studies), pointing to the need for practical, generalizable models that can integrate readily available inspection and environmental data.

Building on this prior work, the present study focuses on predictive modeling of colony health using an open-source dataset that integrates standardized HCC inspection results with nearest-station meteorological observations. Unlike many previous studies, no in-hive sensors or hive scale data were used; instead, we investigated the predictive capacity of weather and calendar variables alone. Our objectives were: (i) to construct and engineer biologically relevant features from inspection and weather records, (ii) to train and evaluate several machine learning classifiers for predicting colony health status (healthy vs. unhealthy), and (iii) to interpret the models to identify which environmental and seasonal factors are most strongly associated with honey bee health. By pursuing these aims, we seek to demonstrate the feasibility of predicting colony health from accessible data.

## 2. MATERIALS AND METHODS

### *2.1 Materials*

The dataset used in this study originates from the publicly available repository "Predicting Honeybee Health: The Healthy Colony Checklist, Hive Scale and Weather Data" [17, 18]. This dataset is designed to help monitor and predict honeybee health. It includes information from hives in North Carolina and Utah, including records of colony health as measured by the standardized HCC protocol. Furthermore, it integrates colony inspections with meteorological observations from the nearest weather stations, covering the period between February 29, 2016, and August 29, 2019, with a total of 1,277 daily observations.

The dataset comprises six complementary files that collectively capture both apiary-level characteristics and environmental conditions relevant to honeybee health. The Apiary Information file provides the basic details of each apiary, including its unique identifier, name, city, and state. The Hive Information file contains hive-specific records, where each hive is linked to an apiary through unique identifiers. The HCC Inspections file documents colony inspections following the standardized HCC, recording parameters such as brood, bees, queen status, food, stressors, and space availability, alongside the binary health outcome. The Weather Stations file supplies metadata about the nearest weather stations connected to each apiary. The Hourly Weather file includes hourly meteorological observations such as temperature, humidity, dew point, wind characteristics, atmospheric pressure, precipitation, and daylight-related data. Finally, the Weather Observations file provides aggregated daily climate variables that are aligned with inspection dates, thereby integrating meteorological conditions with colony health data.

The variables employed for honeybee health prediction are summarized in Table 1. Weather data from the nearest station to each apiary were extracted, aggregated into daily averages, and matched with the corresponding HCC inspection dates by aligning each inspection to the nearest day's meteorological record. These variables include average daily temperature (converted from Fahrenheit to Celsius for consistency), relative humidity, dew point, wind speed, wind gust, atmospheric pressure, and total precipitation. Additionally, temporal and locational descriptors such as inspection date (further transformed into derived features like month, season, and day of year), apiary, city, and station name were included for contextual reference. The variable InspectionID was retained solely as a reference index and excluded from predictive modeling.

The engineered features were designed to capture biologically relevant aspects of colony health by extending raw meteorological variables. Moisture-related indicators included dew point temperature (computed using the Magnus–Tetens approximation), vapor pressure deficit (VPD), and relative humidity deficit, all of which describe atmospheric saturation levels and potential desiccation stress within hives.

To represent combined environmental stressors, we incorporated temperature–humidity (T·RH) and temperature–precipitation (T·P) interactions. These highlight conditions where heat and humidity exacerbate pathogen risks or where warm, rainy periods alter nectar flow and foraging dynamics.

**Table 1:** Variables used for honeybee health prediction.

| Variable | Description |
|---|---|
| InspectionID | Unique identifier of each inspection. Serves only as a reference index and was not included in predictive modeling. |
| InsptDate | Date of the inspection. Transformed into derived temporal features such as month, season, and day of year. |
| Apiary | Name of apiary. |
| City | City of the apiary; also, the city of the related weather station. |
| Station | Name of the weather station from which weather data was collected. |
| Temperature_nearest (T) | Average daily air temperature (°F, converted to °C). Fundamental driver of bee activity and colony metabolism. |
| Humidity_nearest (RH) | Relative humidity (%). Influences hive microclimate and susceptibility to diseases. |
| Dew_Point_nearest (DP) | Dew point temperature (°F). Reflects atmospheric saturation and condensation potential. |
| Wind_Speed_nearest (W) | Mean daily wind speed (mph). Directly affects bees' foraging and flight efficiency. |
| Wind_Gust_nearest (WG) | Maximum daily wind gust (mph). Extreme gusts may disrupt bee flight. |
| Pressure_nearest (PR) | Atmospheric pressure (inHg). Indicator of weather changes influencing bee behavior. |
| Precip_nearest (P) | Daily precipitation (inches). Rainfall constrains foraging activity and food collection. |

Wind effects were approximated by the cube of wind speed, reflecting the physical scaling of wind energy and its disproportionate disruption of bee flight. Precipitation skewness was addressed through a logarithmic transformation, dampening extreme rainfall influence while retaining biological signal.

Finally, seasonal timing was represented using month, season, day of year, and cyclic encodings, while Growing Degree Days (GDD) above a 10°C baseline quantified biologically effective thermal resources, capped to avoid overstating heat beyond 30°C. Together, these engineered variables provided ecologically interpretable representations of atmospheric conditions, seasonal cycles, and stress factors, strengthening the model's ability to capture meaningful drivers of honeybee health.

Table 2 summarizes these engineered features along with their definitions, mathematical formulations, and explanatory roles.

The input variables include both raw meteorological and apiary-related measurements (e.g., temperature, humidity, wind, precipitation, date, location) and engineered features derived to capture seasonal, atmospheric, and biological patterns relevant to honeybee health.

**Table 2:** Engineered features for honeybee health prediction.

| Feature | Definition / Formulation | Explanatory Role |
|---|---|---|
| Month | Months (1–12) | Indicates time of year |
| Season | Winter/Spring/Summer/Fall | Captures broad seasonal phase of colony cycle. |
| Day of Year (DOY) | Integer day of year (1–366) | Captures temporal position of inspections within the year. |
| Cyclic Encoding of Seasonality | $DOY_{sin} = \sin\left(\frac{2\pi(DOY-1)}{365}\right)$<br>$DOY_{cos} = \cos\left(\frac{2\pi(DOY-1)}{365}\right)$ | Encodes annual periodicity and seasonal effects on bee activity. |
| Dew Point Temperature | Magnus–Tetens:<br>$\alpha = \frac{aT}{b+T} + \ln\left(\frac{RH}{100}\right)$;<br>$T_d = \frac{b\alpha}{a-\alpha}$<br>$a = 17.27, b = 237.7$ | Indicates atmospheric moisture conditions influencing hive environment. |
| Vapor Pressure Deficit (VPD) | $e_s = 0.6108 \cdot \exp\left(\frac{17.27T}{T+237.3}\right)$<br>$e_a = e_s \cdot \frac{RH}{100}$<br>$VPD = e_s - e_a$ | Reflects atmospheric dryness; high VPD values indicate arid stress. |
| Relative Humidity Deficit | $RH_{deficit} = 100 - RH$ | Shows deviation from saturation, indicating dehydration stress. |
| Temperature–Humidity Interaction | $T \cdot RH$ | Represents combined effect of heat and humidity, linked to colony stress and disease risk. |
| Temperature–Precipitation Interaction | $T \cdot P$ | Captures joint effect of rainfall and high temperatures on nectar flow and foraging. |
| Wind Energy Approximation | $Wind_{pow3} = W^3$ | Approximates wind energy; higher values hinder flight activities. |
| Logarithmic Transformation of Precipitation | $Precip_{log1p} = \ln(1+P)$ | Reduces skewness in rainfall data, avoiding dominance of extreme values. |
| Growing Degree Days (Base 10°C) | $GDD_{base10} = \max(0, \min(T, T_{cap}) - T_{base})$<br>$T_{base} = 10°C$<br>$T_{cap} = 30°C$ | Reflects biologically effective heat accumulation supporting colony development. |

Note: Units are as follows: temperature and dew point in °C (after conversion), wind in mph (Wind³ in mph³), precipitation in inches (log1p transformed), VPD in kPa.

The target variable was defined using the HCC protocol, which evaluates six key biological conditions: presence of all brood stages (eggs, larvae, pupae) ($\boldsymbol{Q_1}$), adequate number and age structure of worker bees ($\boldsymbol{Q_2}$), presence of a young and productive queen ($\boldsymbol{Q_3}$), sufficient food and pollen storage ($\boldsymbol{Q_4}$), absence of observable stress factors ($\boldsymbol{Q_5}$), availability of adequate comb space ($\boldsymbol{Q_6}$). Each condition is scored as 1 (present) or 0 (absent). The Healthy Colony Checklist Score (HCCS) is calculated as Eq. (1):

$$\mathrm{HCCS} = \sum_{i=1}^{6} \frac{Q_i}{6}, Q_i \in \{0,1\} \qquad (1)$$

If HCCS=1.0, the colony is labeled as Healthy = Yes; otherwise, it is labeled as Healthy = No. However, in the dataset used for this study, the Healthy labels (Yes/No) were already pre-computed and included in the HCC Inspections file; therefore, no additional calculation was required during preprocessing. In the raw data, approximately 36% of inspections were labeled as healthy ("Yes"), while 64% were labeled as unhealthy ("No"), indicating a class imbalance toward unhealthy outcomes.

### *2.2 Methods*

The prediction task was formulated as a binary classification problem, in which the objective was to

determine the health status of each hive inspection. The positive class was defined as Healthy = Yes, indicating that the colony satisfied all criteria specified in the HCC. A range of supervised learning algorithms was employed, with a particular emphasis on ensemble tree-based methods due to their ability to accommodate heterogeneous data types and to capture complex feature interactions. Specifically, the models implemented included RF, Extra Trees (ET), Gradient Boosting (GB) (both the standard scikit-learn implementation and the Histogram-based Gradient Boosting (HistGB) variant), as well as gradient boosting frameworks such as Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), and Categorical Boosting (CatBoost). These methods were selected in view of their demonstrated effectiveness in structured data classification and their capacity to model nonlinear dependencies. For comparative purposes, a logistic regression (LR)–based stacking ensemble was also evaluated, although the primary focus remained on tree-based ensembles. All models were trained using a stratified 80/20 train–test split to preserve class balance within the test partition. Within the training split, a 10-fold stratified cross-validation was applied to generate Out-of-Fold (OOF) predictions, optimize classification thresholds, and enable OOF-based weight tuning for soft voting ensembles. For stacking, meta-learners included LR with a logit link (LRCV+logit) and extended Stacking++ variants with LR and XGBoost. Final performance was reported on the independent test set using both the default 0.50 threshold and the optimized thresholds.

#### *2.2.1 RF*

RF is an ensemble method that builds multiple decision trees using bootstrap samples and random feature subsets, reducing variance and improving accuracy compared to a single tree [19, 20]. Predictions are aggregated through majority voting for classification or averaging for regression, with Out-of-Bag data providing unbiased internal validation [21]. RF is effective in modeling complex nonlinear relationships, though traditional equal-weight voting can be improved by weighted schemes to enhance stability and accuracy [22, 23].

#### *2.2.2 ET*

ET classifier, also known as Extremely Randomized Trees, is a bagging-based ensemble method that builds multiple decision trees using random subsets of features and cut-points to reduce overfitting and variance [24, 25]. Unlike RF, ET grows trees on the full dataset with fully randomized splits, improving generalization while minimizing bias. For classification, predictions are combined through majority voting, and for regression, outputs are averaged, making ET efficient and effective for high-dimensional datasets [26, 27].

#### *2.2.3 CatBoost*

CatBoost is a gradient boosting algorithm designed to handle categorical features using oblivious decision trees as base learners, effectively reducing overfitting and improving generalization [28, 29]. Unlike traditional methods, it processes categorical data during training through ordered boosting and robust encoding, preventing data leakage and noise effects [30, 31]. By sequentially building decision trees to correct previous errors, CatBoost achieves high accuracy with minimal parameter tuning, making it efficient for both small-scale and high-dimensional datasets [31, 32].

#### *2.2.4 GB*

GB is an ensemble method that builds decision trees sequentially, with each new tree correcting the errors of its predecessors [33, 34]. It minimizes a loss function using gradient descent, iteratively adding weak learners to improve overall accuracy [35, 36]. To prevent overfitting, stopping criteria such as iteration limits or performance thresholds are applied [37].

#### *2.2.5 XGBoost*

XGBoost is an optimized version of Gradient Boosted Decision Trees (GBDT) that integrates regularization to prevent overfitting and improve accuracy [38, 39]. As a tree-based ensemble method, it combines multiple weak learners using additive training strategies and supports efficient computation through parallel processing [40]. XGBoost is widely recognized for its speed, scalability, and ability to handle missing values and outliers, making it effective for both regression and classification tasks [41, 42].

#### 2.2.6 *LightGBM*

LightGBM is an efficient gradient boosting framework developed by Microsoft for regression, classification, and ranking tasks. It improves over traditional GBDT by using Gradient-based One-Sided Sampling and Exclusive Feature Bundling, which reduce training time and memory usage while preserving accuracy [43, 44]. LightGBM applies a leaf-wise growth strategy and histogram-based splitting, enabling fast training, scalability, and effective handling of large-scale, sparse, and imbalanced data [45, 46].

#### 2.2.7 *HistGB*

HistGB accelerates model training and inference by binning continuous feature values into histograms, thereby reducing both memory usage and computational cost [47, 48]. Inspired by LightGBM, it efficiently handles missing values by directing them to the appropriate child node during training and prediction [49]. By combining the accuracy of traditional GB with improved speed and efficiency, HistGB is particularly effective for large datasets [50, 51].

#### 2.2.8 *Soft Voting (OOF-optimized)*

The ensemble classifier is an intricate machine learning technique that combines multiple classifiers and leverages their complementary strengths to achieve superior predictive performance. Among different integration strategies, voting is one of the fundamental mechanisms, where predictions from base classifiers are aggregated to determine the final output. A voting classifier can be of three types: hard, soft, and weighted. In soft voting, the predicted probabilities of each class are averaged to generate the final output, whereas in hard voting, the majority vote determines the class, and in weighted voting, classifier contributions vary depending on their performance [52]. In its optimized form, Soft Voting employs OOF validation to tune model weights instead of assigning them equally, thereby maximizing ensemble performance. This OOF-optimized weighting improves both stability and accuracy compared to the traditional equal-weight scheme, especially in heterogeneous model ensembles. In this study, several OOF-optimized soft voting ensembles were tested, including combinations of RF and ET (RF+ET), as well as extended variants integrating LightGBM (RF+ET+LightGBM), XGBoost (RF+ET+XGBoost), and CatBoost (RF+ET+CatBoost).

#### 2.2.9 *Stacking / Stacking++*

The term stacking, short for stacked generalization, refers to an ensemble learning method that combines the predictions of multiple base models through a higher-level learning process. In this approach, base learners (level 0 models) are trained independently, and their predictions are used as input features for a meta-model (level 1 model), which learns the optimal way to aggregate them. The architecture may consist of two or more base learners, with the meta-model integrating their predictions to generate the final output. This process includes the initial training data, base models, primary predictions, the secondary meta-model, and the ultimate predictions, enabling the method to leverage information across models effectively [53]. Stacking has been shown to achieve high accuracy by utilizing estimation results from multiple models, making it especially powerful in handling complex datasets [54]. Unlike simple voting, stacking enhances predictive power by discovering non-linear relationships among base model outputs. The extended version, Stacking++, incorporates OOF predictions to prevent information leakage and reduce overfitting, thereby ensuring more robust generalization performance. In this study, stacking ensembles were implemented using multiple meta-learners. Specifically, a logistic regression with a logit link was employed as the primary meta-learner, while alternative variants with LR and XGBoost were also tested in extended Stacking++ configurations. This allowed for a comparative assessment of different meta-models for combining base learner outputs.

### 2.3 *Feature Engineering and Selection*

To ensure robustness and avoid redundancy in the predictive models, a correlation-based feature selection procedure was applied. First, the pairwise Pearson correlation coefficients ($|\boldsymbol{r}|$) among all candidate features were computed, as illustrated in Figure 1. Table 3 summarizes the most highly correlated pairs, where strong dependencies were observed, particularly between raw meteorological variables and their engineered counterparts.

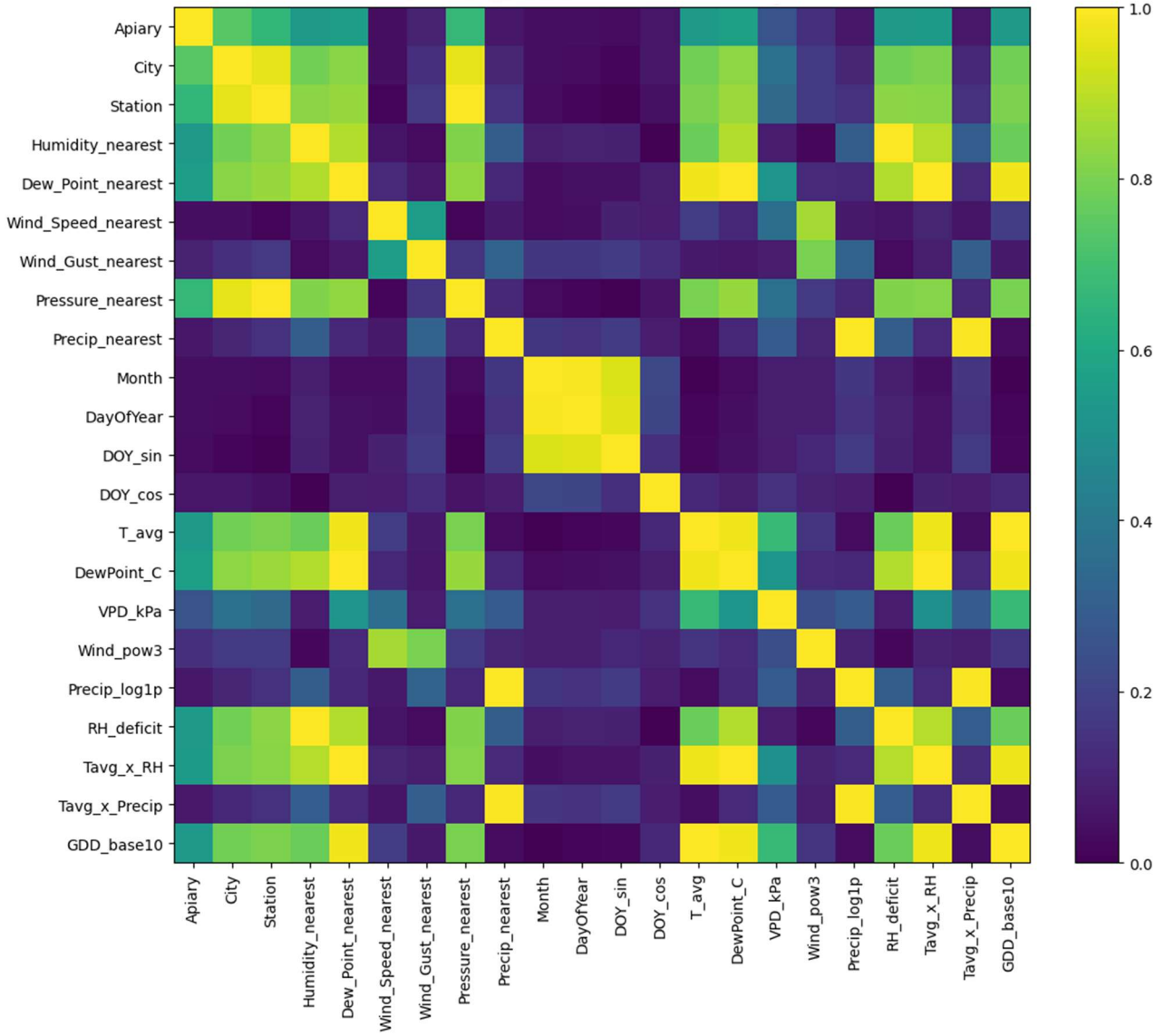


**Figure 1:** Features correlation heatmap ($|r|$).

Based on these results, to mitigate multicollinearity, features with $|\boldsymbol{r}| \geq \mathbf{0.98}$ were iteratively removed, prioritizing the retention of variables with higher biological interpretability and lower redundancy. Although several of the excluded variables, such as GDD_base10 and Dew_Point_nearest, possess biological relevance to colony thermoregulation and seasonal development, they exhibited near-perfect correlations with average temperature and related features. To prevent numerical instability and redundancy, one representative variable per correlation cluster was retained. This ensured that the models preserved ecological interpretability while maintaining statistical independence among predictors.

This iterative dropping procedure resulted in the exclusion of the following features: Humidity_nearest, Precip_nearest, GDD_base10, Dew_Point_nearest, Pressure_nearest, Tavg_x_RH, Tavg_x_Precip, Month, T_avg.

After the removal, the revised correlation structure is shown in Figure 2 and demonstrated a substantial reduction in extreme correlations. As presented in Table 4, no feature pairs exceeded the $|\boldsymbol{r}| \geq \mathbf{0.98}$ threshold.

**Table 3:** Highly correlated feature pairs (Top 30, before iterative drop).

| Feature 1 | Feature 2 | $\|r\|$ | Feature 1 | Feature 2 | $\|r\|$ |
|---|---|---|---|---|---|
| Humidity_nearest | RH_deficit | 1.000000 | T_avg | Tavg_x_RH | 0.974714 |
| Precip_nearest | Precip_log1p | 0.999997 | City | Station | 0.963460 |
| T_avg | GDD_base10 | 0.999996 | City | Pressure_nearest | 0.963291 |
| Dew_Point_nearest | DewPoint_C | 0.999651 | DayOfYear | DOY_sin | 0.956708 |
| Station | Pressure_nearest | 0.998155 | Month | DOY_sin | 0.944971 |
| Dew_Point_nearest | Tavg_x_RH | 0.997885 | RH_deficit | Tavg_x_RH | 0.889920 |
| DewPoint_C | Tavg_x_RH | 0.997066 | Humidity_nearest | Tavg_x_RH | 0.889920 |
| Precip_log1p | Tavg_x_Precip | 0.995842 | Humidity_nearest | Dew_Point_nearest | 0.886496 |
| Precip_nearest | Tavg_x_Precip | 0.995670 | Dew_Point_nearest | RH_deficit | 0.886496 |
| Month | DayOfYear | 0.991417 | Humidity_nearest | DewPoint_C | 0.885958 |
| DewPoint_C | GDD_base10 | 0.980050 | DewPoint_C | RH_deficit | 0.885958 |
| T_avg | DewPoint_C | 0.980033 | Wind_Speed_nearest | Wind_pow3 | 0.866032 |
| Dew_Point_nearest | GDD_base10 | 0.979308 | Station | DewPoint_C | 0.851434 |
| Dew_Point_nearest | T_avg | 0.979298 | Pressure_nearest | DewPoint_C | 0.845968 |
| Tavg_x_RH | GDD_base10 | 0.974725 | Station | Dew_Point_nearest | 0.843969 |

**Table 4:** Correlated feature pairs remaining after feature reduction (Top 30).

| Feature 1 | Feature 2 | $\|r\|$ | Feature 1 | Feature 2 | $\|r\|$ |
|---|---|---|---|---|---|
| City | Station | 0.963460 | Apiary | RH_deficit | 0.539266 |
| DayOfYear | DOY_sin | 0.956708 | DewPoint_C | VPD_kPa | 0.525907 |
| DayOfYear | Quarter | 0.928020 | City | VPD_kPa | 0.369605 |
| DewPoint_C | RH_deficit | 0.885958 | Wind_Speed_nearest | VPD_kPa | 0.362132 |
| DOY_sin | Quarter | 0.871980 | Station | VPD_kPa | 0.345987 |
| Wind_Speed_nearest | Wind_pow3 | 0.866032 | Wind_Gust_nearest | Precip_log1p | 0.318897 |
| Station | DewPoint_C | 0.851434 | Precip_log1p | RH_deficit | 0.295659 |
| City | DewPoint_C | 0.828518 | VPD_kPa | Precip_log1p | 0.292110 |
| Station | RH_deficit | 0.828016 | Apiary | VPD_kPa | 0.260199 |
| Wind_Gust_nearest | Wind_pow3 | 0.799265 | DOY_cos | Quarter | 0.242218 |
| City | RH_deficit | 0.790294 | VPD_kPa | Wind_pow3 | 0.239562 |
| Apiary | City | 0.740964 | DayOfYear | DOY_cos | 0.210855 |
| Apiary | Station | 0.663408 | Wind_Gust_nearest | DOY_sin | 0.173912 |
| Apiary | DewPoint_C | 0.566495 | DOY_sin | Precip_log1p | 0.173877 |
| Wind_Speed_nearest | Wind_Gust_nearest | 0.556427 | Station | Wind_pow3 | 0.168898 |

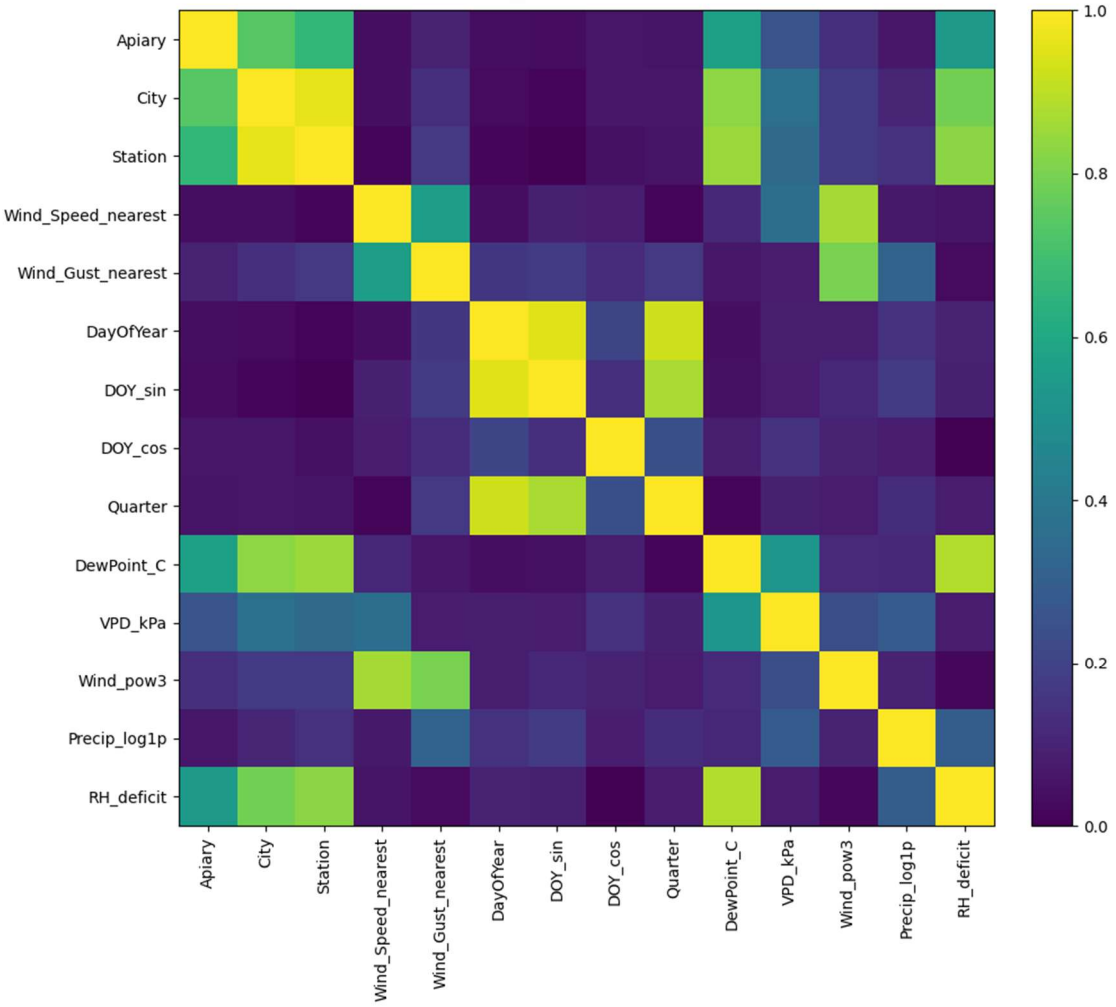


**Figure 2:** Features correlation heatmap after iterative drop ($|r|$).

### *2.4 Evaluation Metrics*

Model performance was assessed using accuracy, precision, recall, F1-score, and ROC-AUC. The evaluation relied on confusion matrices, from which the following indices were derived: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP denotes instances correctly identified as positive, while TN refers to instances correctly identified as negative. FP corresponds to cases incorrectly classified as positive, and FN represents cases incorrectly classified as negative [55]. The definitions of the performance evaluation metrics are provided as follows:

- **Accuracy:** Accuracy represents the proportion of correctly classified instances relative to the total number of predictions. It is computed using Eq. (2), which expresses the ratio of all true predictions made by the model to the overall number of predictions [56].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

- **Precision:** Precision measures the proportion of correctly identified positive instances among all instances predicted as positive. As expressed in Eq. (3), it is calculated as the ratio of TPs to the sum of TPs and FPs [57].

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

- **Recall:** Recall reflects the model's capacity to correctly identify relevant positive cases and is also referred to as the true positive rate. As expressed in Eq. (4), it is defined as

the ratio of correctly classified positive instances to the total number of actual positive instances [58].

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

- **F1-Score:** The F1-Score is defined as the harmonic mean of Precision and Recall, providing a balanced measure that accounts for both metrics. This relationship is mathematically expressed in Eq. (5) [59].

$$F1 - Score = \frac{2 * Recall * Precision}{Recall + Precision} \quad (5)$$

- **ROC-AUC:** ROC-AUC evaluates the model's ability to distinguish between the positive and negative classes across different threshold values. The ROC curve is constructed by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR). TPR is mathematically equivalent to Recall (Eq. 4), while FPR is defined as the proportion of FPs among all actual negatives, as shown in Eq. (6). The ROC-AUC is then obtained by calculating the area under this curve, as shown in Eq. (7), where values closer to 1 indicate superior discriminative performance [60].

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \quad (6)$$

$$\mathrm{ROC} - \mathrm{AUC} = \int_0^1 \mathrm{TPR}(\mathrm{FPR})\mathrm{d}(\mathrm{FPR}) \quad (7)$$

## 3. RESULTS

Given the class imbalance in the dataset, where unhealthy colonies were more prevalent, relying solely on accuracy could lead to misleading conclusions. Although stratified sampling and threshold optimization were applied during training, the underlying class imbalance (36% healthy vs. 64% unhealthy) may still influence the models' generalization ability. This imbalance can bias classifiers toward the majority "unhealthy" class, potentially limiting sensitivity to minority healthy cases. Therefore, the reported accuracy values should be interpreted alongside precision and recall to better reflect the model's robustness across both classes.

To provide a more comprehensive evaluation, precision and recall values for the "Healthy" class were reported in order to capture error tendencies in both directions. In practical terms, a FN—misclassifying an unhealthy colony as healthy—can be particularly costly for colony management. Therefore, while an ideal model might prioritize recall of unhealthy cases, the present study initially adopted overall accuracy as the primary benchmark. To refine performance estimation, post-training threshold optimization was performed for each model. Instead of applying the default probability cutoff of 0.50, the threshold that maximized accuracy during cross-validation was selected. This procedure frequently produced thresholds greater than 0.50, reflecting a conservative approach in which colonies were predicted as "Healthy" only with higher confidence. As a result, precision increased at the expense of some recall. Beyond individual classifiers, ensemble strategies such as soft voting and stacking (including its optimized variant) were also explored, enabling comparison between simple model averaging and meta-learning approaches. Final performance metrics were therefore computed on the independent test set using both the default threshold and the optimized thresholds for direct comparison.

Table 5 reports the baseline classification results of individual models at the default probability threshold of 0.50. The performance metrics indicate that all models achieved broadly similar accuracies, clustering around 74–75%. Among them, RF and XGBoost performed marginally better, with accuracy values slightly above 75% and ROC-AUC close to 0.80. The confusion matrices in Figure 3 further illustrate these baseline results, showing that while models performed consistently in distinguishing between classes, FNs remained relatively frequent, reflecting a conservative tendency to classify healthy colonies as potentially unhealthy.

**Table 5:** Classification results of individual models at default threshold (0.50).

| Model | Accuracy | Precision | Recall | F1-Score | ROC-AUC |
|---|---|---|---|---|---|
| RF | 0.7526 | 0.6628 | 0.6514 | 0.6571 | 0.7954 |
| XGBoost | 0.7505 | 0.6687 | 0.6229 | 0.6450 | 0.7958 |
| ET | 0.7484 | 0.6849 | 0.5714 | 0.6231 | 0.7891 |
| LightGBM | 0.7484 | 0.6570 | 0.6457 | 0.6513 | 0.7954 |
| CatBoost | 0.7464 | 0.6587 | 0.6286 | 0.6433 | 0.7890 |
| HistGB | 0.7380 | 0.6601 | 0.5771 | 0.6159 | 0.7863 |
| GB | 0.7380 | 0.6644 | 0.5657 | 0.6111 | 0.7764 |

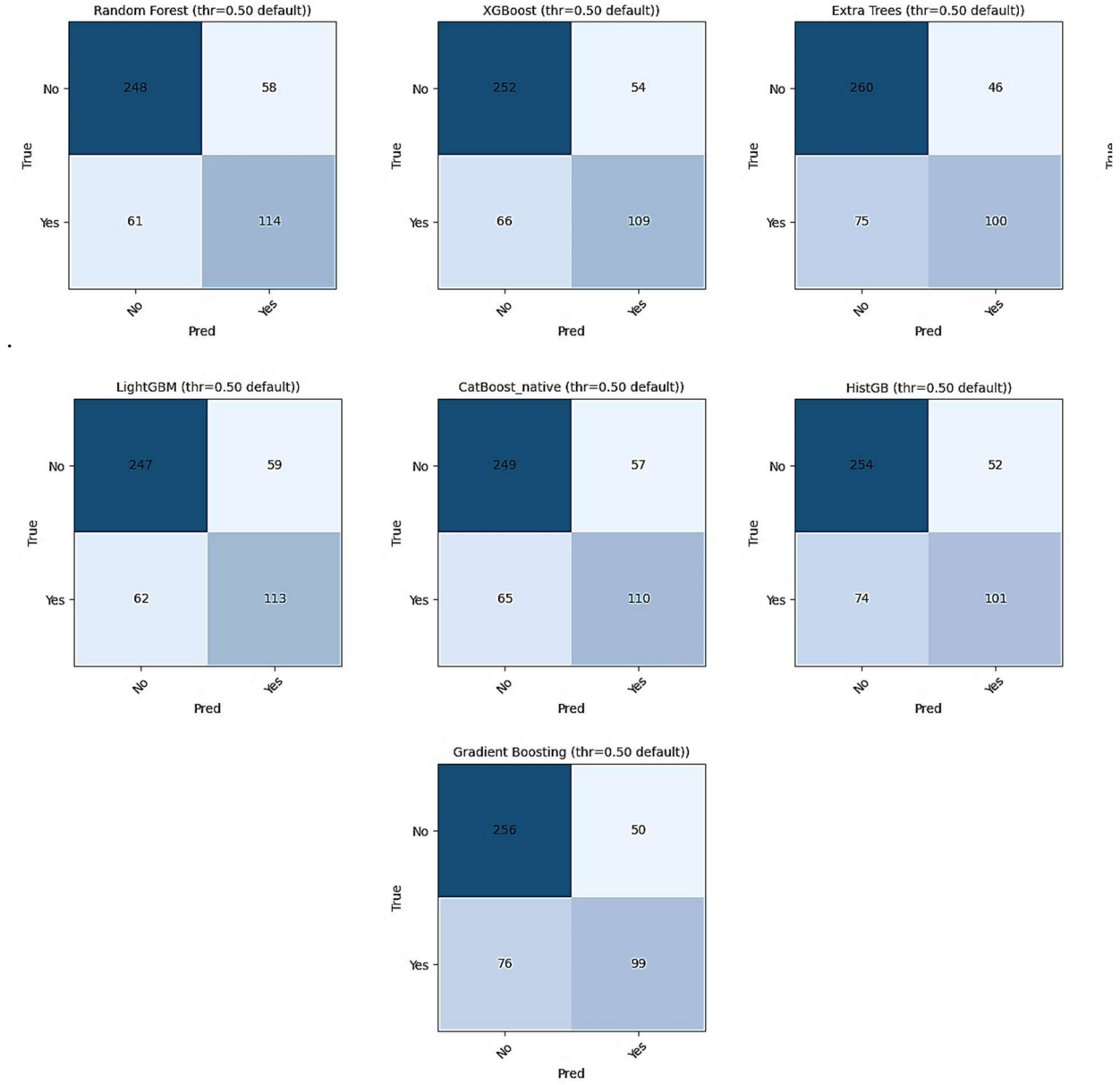


**Figure 3:** Confusion matrices of default-threshold (0.50).

Table 6 presents the results of individual models when evaluated at accuracy-optimized thresholds. Compared to the default setting, threshold adjustment provided modest gains, particularly for ET and LightGBM, both reaching 75.88% accuracy. This demonstrates that fine-tuning the decision threshold can slightly enhance predictive performance, although the overall improvements

remained limited. The corresponding confusion matrices in Figure 4 confirm that optimized thresholds improved the balance between TPs and TNs, reducing FPs without substantially increasing FNs

**Table 6:** Classification results of individual models with optimized thresholds.

| Model | Accuracy | Precision | Recall | F1-Score | ROC-AUC |
|---|---|---|---|---|---|
| ET (thr=0.58) | 0.7588 | 0.7287 | 0.5371 | 0.6184 | 0.7891 |
| LightGBM (thr=0.59) | 0.7588 | 0.7153 | 0.5600 | 0.6282 | 0.7954 |
| RF (thr=0.57) | 0.7547 | 0.6993 | 0.5714 | 0.6289 | 0.7954 |
| XGBoost (thr=0.54) | 0.7526 | 0.6892 | 0.5829 | 0.6316 | 0.7958 |
| GB (thr=0.53) | 0.7484 | 0.6929 | 0.5543 | 0.6159 | 0.7764 |
| CatBoost (thr=0.61) | 0.7464 | 0.7054 | 0.5200 | 0.5987 | 0.7890 |
| HistGB (thr=0.52) | 0.7422 | 0.6735 | 0.5657 | 0.6149 | 0.7863 |

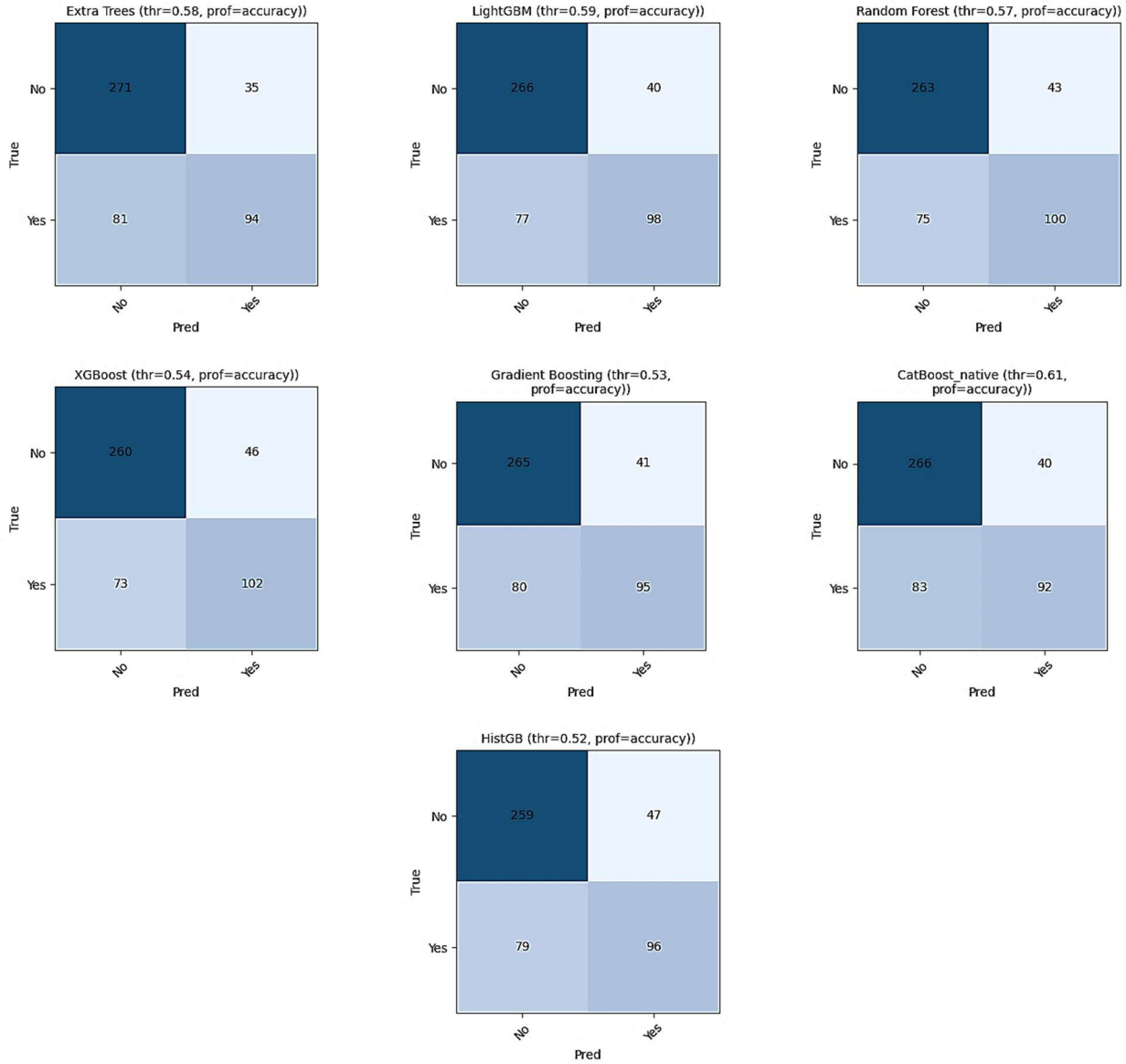


**Figure 4:** Confusion matrices of per-model (accuracy-optimum).

**Table 7:** Classification results of soft voting ensembles with optimized weights and thresholds.

| Model | Accuracy | Precision | Recall | F1-Score | ROC-AUC |
|---|---|---|---|---|---|
| Soft Voting OPT (RF+ET \| thr=0.59) | 0.7588 | 0.7218 | 0.5486 | 0.6234 | 0.7924 |
| Soft Voting OPT (RF+ET+LightGBM \| thr=0.60) | 0.7588 | 0.7185 | 0.5543 | 0.6258 | 0.7966 |
| Soft Voting OPT (RF+ET+XGBoost \| thr=0.58) | 0.7526 | 0.7000 | 0.5600 | 0.6222 | 0.7956 |
| Soft Voting OPT (RF+ET+CatBoost \| thr=0.62) | 0.7484 | 0.7411 | 0.4743 | 0.5784 | 0.7912 |

The unweighted soft-voting ensemble constructed from the two best-performing models, ET and RF, achieved an accuracy of 75.88%, essentially matching the strongest individual classifier. Incorporating a third model, such as LightGBM, did not yield further improvement, indicating that the top models were capturing highly similar patterns and that their errors were substantially correlated. Table 7 summarizes these results, while the confusion matrices in Figure 5 show that soft voting ensembles largely reproduced the same misclassification patterns as their base learners, underscoring the limited diversity among top-performing classifiers.

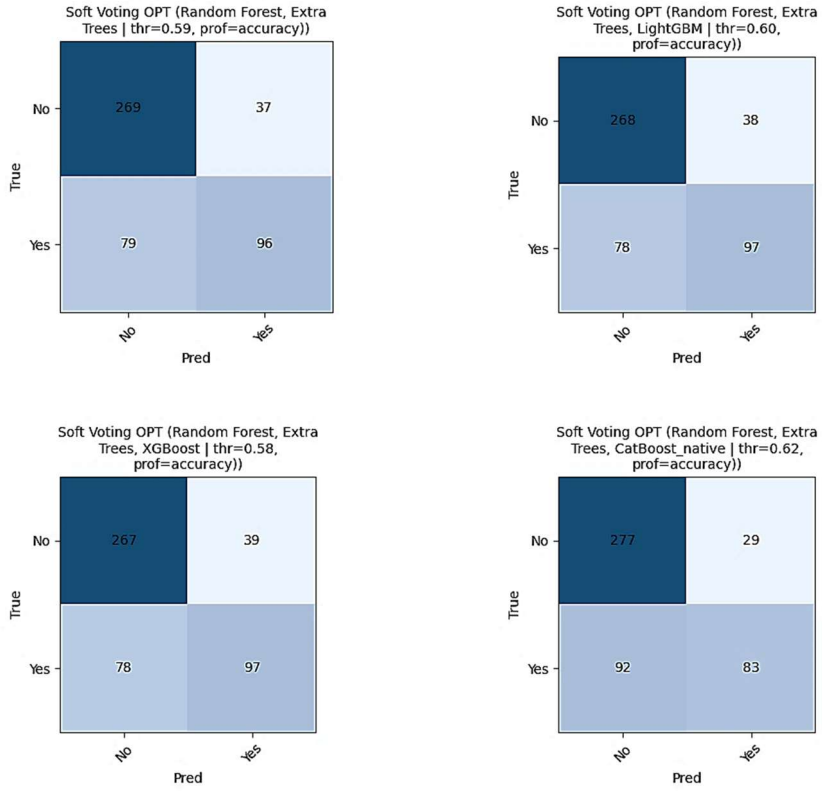


**Figure 5:** Confusion matrices of soft voting — optimized (OOF).

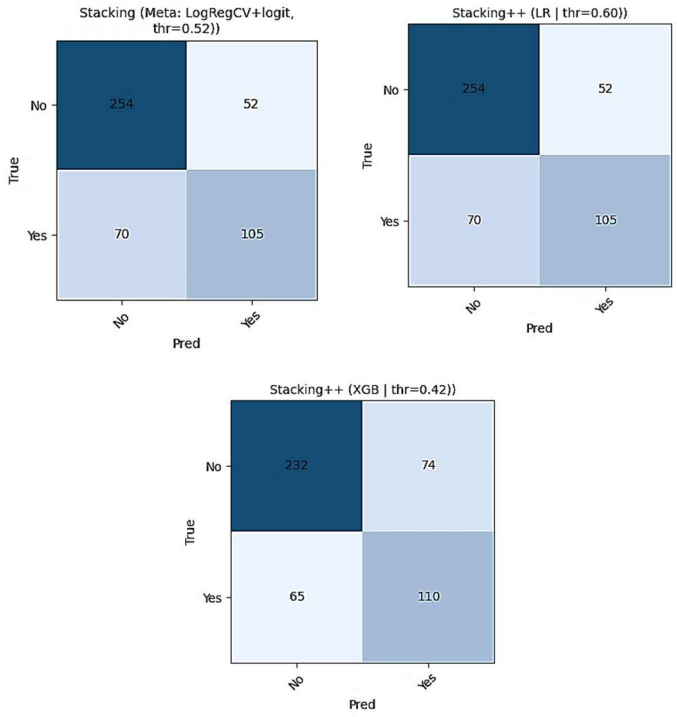


**Figure 6:** Confusion matrices of stacking / stacking++.

The stacking approach with logistic regression as a meta-learner plateaued at approximately 74.64%, despite its theoretical ability to assign differentiated weights to individual classifiers. Stacking++ variants similarly produced no substantial performance improvements. These results are detailed in Table 8, while the confusion matrices in Figure 6 illustrate that misclassification patterns persisted across stacking-based ensembles, reflecting that more complex integration strategies offered only marginal gains over the strongest single models.

**Table 8:** Classification results of stacking and stacking++ meta-models.

| Model | Accuracy | Precision | Recall | F1-Score | ROC-AUC |
|---|---|---|---|---|---|
| Stacking (LRCV+logit \| thr=0.52) | 0.7464 | 0.6688 | 0.6000 | 0.6325 | 0.7961 |
| Stacking++ (LR \| thr=0.60) | 0.7464 | 0.6688 | 0.6000 | 0.6325 | 0.7915 |
| Stacking++ (XGBoost \| thr=0.42) | 0.7110 | 0.5978 | 0.6286 | 0.6128 | 0.7835 |

When considered together, the aggregated results in Table 9 confirm that all competitive models converged within a narrow performance band, with accuracies clustered around 75–76%. This consolidated view highlights that, despite variations in thresholds and ensemble strategies, the achievable upper bound remained stable across methods.

**Table 9:** Aggregated classification results of all models.

| Model | Accuracy | Precision | Recall | F1-Score | ROC-AUC |
|---|---|---|---|---|---|
| ET (thr=0.58) | 0.7588 | 0.7287 | 0.5371 | 0.6184 | 0.7891 |
| Soft Voting OPT (RF+ET \| thr=0.59) | 0.7588 | 0.7218 | 0.5486 | 0.6234 | 0.7924 |
| Soft Voting OPT (RF+ET+LightGBM \| thr=0.60) | 0.7588 | 0.7185 | 0.5543 | 0.6258 | 0.7966 |
| LightGBM (thr=0.59) | 0.7588 | 0.7153 | 0.5600 | 0.6282 | 0.7954 |
| RF (thr=0.57) | 0.7547 | 0.6993 | 0.5714 | 0.6289 | 0.7954 |
| RF (thr=0.50 default) | 0.7526 | 0.6628 | 0.6514 | 0.6571 | 0.7954 |
| Soft Voting OPT (RF+ET+XGBoost \| thr=0.58) | 0.7526 | 0.7000 | 0.5600 | 0.6222 | 0.7956 |
| XGBoost (thr=0.54) | 0.7526 | 0.6892 | 0.5829 | 0.6316 | 0.7958 |
| XGBoost (thr=0.50 default) | 0.7505 | 0.6687 | 0.6229 | 0.6450 | 0.7958 |
| Soft Voting OPT (RF+ET+CatBoost \| thr=0.62) | 0.7484 | 0.7411 | 0.4743 | 0.5784 | 0.7912 |
| LightGBM (thr=0.50 default) | 0.7484 | 0.6570 | 0.6457 | 0.6513 | 0.7954 |
| ET (thr=0.50 default) | 0.7484 | 0.6849 | 0.5714 | 0.6231 | 0.7891 |
| GB (thr=0.53) | 0.7484 | 0.6929 | 0.5543 | 0.6159 | 0.7764 |
| CatBoost (thr=0.50 default) | 0.7464 | 0.6587 | 0.6286 | 0.6433 | 0.7890 |
| Stacking (Meta: LRCV+logit, thr=0.52) | 0.7464 | 0.6688 | 0.6000 | 0.6325 | 0.7961 |
| Stacking++ (LR \| thr=0.60) | 0.7464 | 0.6688 | 0.6000 | 0.6325 | 0.7915 |
| CatBoost (thr=0.61) | 0.7464 | 0.7054 | 0.5200 | 0.5987 | 0.7890 |
| HistGB (thr=0.52) | 0.7422 | 0.6735 | 0.5657 | 0.6149 | 0.7863 |
| HistGB (thr=0.50 default) | 0.7380 | 0.6601 | 0.5771 | 0.6159 | 0.7863 |
| GB (thr=0.50 default) | 0.7380 | 0.6644 | 0.5657 | 0.6111 | 0.7764 |
| Stacking++ (XGBoost \| thr=0.42) | 0.7110 | 0.5978 | 0.6286 | 0.6128 | 0.7835 |

The ROC curves presented in Figure 7 confirm that all models, including individual classifiers and ensemble variants, demonstrated comparable discrimination ability, with AUC values clustering around 0.78–0.80. The similarity of ROC profiles across models reinforces the observation from Table 9 that the current feature set imposes a practical ceiling on predictive performance. Achieving higher accuracy would likely require the integration of additional predictive features, such as direct hive condition indicators or management-related variables not captured by weather data, or the application of more advanced modeling strategies. Nonetheless, tree-based ensembles remain among the most effective approaches for tabular prediction tasks of this nature.

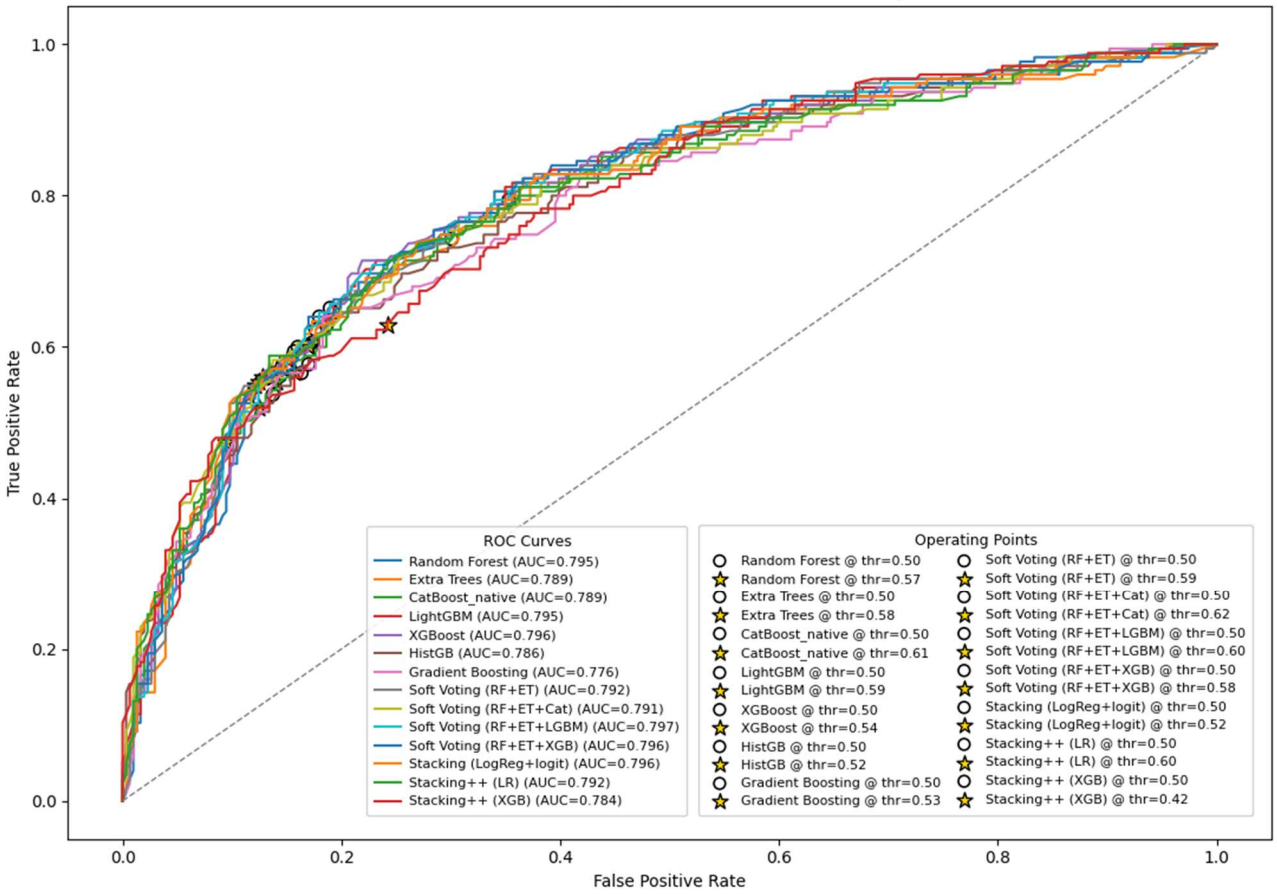


**Figure 7:** ROC curves.

## 4. DISCUSSION

The findings of this study demonstrate that honeybee colony health can be predicted with moderate accuracy using weather-related and seasonal variables. Tree-based ensemble methods, particularly RF, ET, and LightGBM, consistently achieved accuracies up to 75.88%, with precision values exceeding 0.70 but recall remaining modest (approximately 0.55). This performance pattern indicates a natural ceiling imposed by the current feature set, where the predictive information contained in weather and calendar-derived inputs has been largely saturated. Further improvements in accuracy are therefore likely dependent on expanding the feature space rather than relying solely on methodological refinements.

One important observation is the trade-off between precision and recall when adjusting classification thresholds. Higher thresholds than the conventional 0.50 cutoff improved precision in identifying healthy colonies but reduced recall, meaning that more genuinely healthy colonies were misclassified as unhealthy. While this conservative bias can reduce false alarms, it also highlights the importance of aligning modeling goals with practical priorities, such as ensuring that unhealthy colonies are not overlooked in operational beekeeping contexts.

Advanced ensemble strategies, such as soft voting and stacking, did not produce substantial gains beyond the strongest single models. This finding suggests that the effectiveness of ensemble methods was constrained by limited feature diversity, a pattern consistent with other domains where models converge on similar decision boundaries. Consequently, performance enhancements will likely require more diverse and informative predictors rather than additional ensemble complexity.

Seasonality emerged as the most dominant factor shaping colony health outcomes, consistent with the biological rhythm of honeybee colonies. Calendar-derived features captured the cyclical

nature of hive activity, while meteorological factors such as temperature, precipitation, and vapor pressure deficit provided secondary but meaningful contributions. The prominence of these temporal and weather signals validates the use of agrometeorological data as valuable inputs for apicultural decision-support systems, although it also underscores the importance of contextualizing predictions within seasonal practices.

Future research directions should emphasize both methodological and applied advances. In particular, resampling and data augmentation techniques such as SMOTE and ADASYN could be explored to mitigate class imbalance and improve recall, particularly for minority healthy colonies. On the methodological side, advanced deep learning architectures, particularly CNNs and RNNs, as well as time-series models, may help capture nonlinear dependencies and temporal dynamics that tree-based ensembles may not fully exploit. On the applied side, integrating direct hive-level indicators (e.g., brood patterns, hive weight, acoustic monitoring) and beekeeper management records (e.g., queen replacement, feeding practices) is expected to substantially enhance predictive performance. Collaboration with beekeepers to validate model predictions under real-world conditions will be essential to ensure the development of reliable and practical tools for colony health management.

## 5. CONCLUSIONS

This study demonstrated that the health status of honeybee colonies, as assessed through the HCC, can be predicted with approximately 76% accuracy using weather variables and calendar-derived information. Seasonality was identified as the most influential factor, while meteorological indicators such as ambient moisture and fair-weather conditions provided additional predictive capacity. Among the tested methods, tree-based ensembles (ET, LightGBM, RF) achieved the highest performance, although further ensemble combinations did not yield improvements, indicating a performance plateau given the available features.

The models achieved relatively high precision (>0.70) in classifying healthy colonies, thereby minimizing false alarms, but recall remained modest (approximately 0.55), suggesting a conservative bias. These findings confirm that temporal and climatic conditions leave a measurable imprint on colony health and support the use of agrometeorological data in apicultural decision-support systems.

Nonetheless, the reliance on weather and seasonal data highlights clear limitations, as colony-level factors (e.g., brood status, hive strength, disease incidence) and management practices exert considerable influence yet were not included in this study. Incorporating such metrics, together with environmental context (land use, forage availability), and applying advanced approaches such as deep neural networks and time-series models, are expected to enhance predictive capacity.

From an applied perspective, validation through beekeeper collaboration represents a critical next step. Practical decision-support tools based on simple inputs (e.g., date, location, recent weather) could assist in early detection of at-risk colonies and guide interventions such as supplemental feeding, pest management, or queen replacement. Ensuring robustness across diverse apiaries and climates will be essential for broad applicability.

In summary, the present work establishes a valuable baseline, showing that tree-based ensembles are robust within the limits of the current feature set and demonstrate the feasibility of predicting colony health from accessible data. At the same time, the study outlines clear directions for methodological and applied advancements, guiding the development of more comprehensive tools that support sustainable apicultural decision-making.

**Author Contribution:** The author conceived the study, collected and organized the data, conducted the analysis, and interpreted the findings. The author wrote the manuscript, discussed the results within the context of the study, and finalized the paper.

**Conflicts of Interest:** The author declares that there is no conflict of interest.

## 6. REFERENCES


[1] J. Marcelino *et al.*, "The movement of western honey bees (Apis mellifera L.) among U.S. states and territories: history, benefits, risks, and mitigation strategies," *Front. Ecol. Evol.*, 10, 850600, 2022.

[2] E. J. García-Vicente *et al.*, "Main causes of producing honey bee colony losses in southwestern Spain: a novel machine learning-based approach," *Apidologie,* 55(5), 67, 2024.

[3] J. Tang *et al.*, "Survey results of honey bee colony losses in winter in China (2009–2021)," *Insects,* 14(6), 554, 2023.

[4] B. Branchiccela *et al.*, "Impact of nutritional stress on the honeybee colony health," *Sci. Rep.*, 9(1), 10156, 2019.

[5] Z. Şengül, B. Yücel, G. Saner, and Ç. Takma, "Investigating the impact of climate parameters on honey yield under migratory beekeeping conditions through decision tree analysis: the case of İzmir province," *ANADOLU Ege Tarımsal Araştırma Enstitüsü Dergisi,* 33(2), 268-280, 2023.

[6] M. Güneşdoğdu and A. Şekeroğlu, "Honey bee (Apis mellifera L.) nutrients and nutritional physiology: A review," in *Current Studies on Agriculture, Forest and Aquatic Products*, M. N. İzgi ed. Türkiye: Iksad Publishing House, 2024, 3-46.

[7] K. A. Overturf *et al.*, "Winter weather predicts honey bee colony loss at the national scale," *Ecol. Indic.*, 145, 109709, 2022.

[8] Z. N. Ulgezen, C. van Dooremalen, and F. van Langevelde, "Why does resource availability matter for honeybee colonies in spring?," *Insectes Soc.*, 72, 405-411, 2025.

[9] J. A. Cazier, R. Rogers, E. E. Hassler, and J. T. Wilkes, "The healthy colony checklist (HCC) Part I: A framework for aggregating hive inspection data," *Bee Culture,* 29-32, 2018.

[10] D. B. Carlini *et al.*, "Quantitative microbiome profiling of honey bee (Apis mellifera) guts is predictive of winter colony loss in northern Virginia (USA)," *Sci. Rep.*, 14(1), 11021, 2024.

[11] H. Hammami and N. Abdulaziz, "BeeBetter: A multi-modal beehive system for honeybee health monitoring and hazard detection," in Proc. *7th Int. Conf. Signal Process. Inf. Secur. (ICSPIS)*, Nov. 12–14, 2024, 1-5.

[12] A. Liang, "Developing an AI-based integrated system for bee health evaluation," *IEEE Access*, 158703-158713, 2024.

[13] C. van Dooremalen *et al.*, "Bridging the gap between field experiments and machine learning: The EC H2020 B-GOOD project as a case study towards automated predictive health monitoring of honey bee colonies," *Insects,* 15(1), 76, 2024.

[14] M. Torky, A. A. Nasr, and A. E. Hassanien, "Recognizing beehives' health abnormalities based on MobileNet deep learning model," *Int. J. Comput. Intell. Syst.,* 16(1), 135, 2023.

[15] Y. Zhu *et al.*, "Early prediction of honeybee hive winter survivability using multi-modal sensor data," in *Proc. IEEE Int. Workshop Metrol. Agric. Forestry (MetroAgriFor)*, Nov. 6–8, 2023, 657-662.

[16] A. R. Braga *et al.,* "A method for mining combined data from in-hive sensors, weather and apiary inspections to forecast the health status of honey bee colonies," *Comput. Electron. Agric.,* 169, 105161, 2020.

[17] E. Lower, S. P. Kollaparthi, R. Rogers, E. Hassler, and J. Cazier, "Predicting honeybee health: the healthy colony checklist, hive scale and weather data," *Data Anal. Good,* 2, 1-25, 2024.

[18] E. Lower, S. Kollaparthi, R. Rogers, E. Hassler, and J. Cazier, *Predicting Honeybee Health: The Healthy Colony Checklist, Hive Scale and Weather Data*. Mendeley Data, 2025.

[19] T. Luo, J. Qu, and S. Cheng, "Technological opportunity discovery based on VERGM and random forest model," *Expert Syst. Appl.,* 293, 128712, 2025.

[20] J. Luo, L. Wang, W. Gao, and H. Jiang, "Prediction of ventilation air methane explosion in regenerative thermal oxidation based on hyperparameter-optimized random forest algorithm," *J. Loss Prev. Process Ind.,* 98, 105757, 2025.

[21] K. F. Chin *et al.*, "Predicting unmeasured asymmetry time spectra in μSR experiments using random forest," *J. Magn. Magn. Mater.,* 629, 173320, 2025.

[22] M. Badrakh, N. Tserendash, E. Choindonjamts, and G. Albert, "Potential of random forest machine learning algorithm for geological mapping using PALSAR and Sentinel-2A remote sensing data: A case study of Tsagaan-

uul area, southern Mongolia," *J. Asian Earth Sci.: X,* 14, 100204, 2025.

[23] Z. Guo, T. Huang, Z. Wu, T. Lin, and H. Huang, "A study on dynamic cleaning of charging pile electric energy metering data based on improved random forest algorithm," *Measurement,* 256, 118114, 2025.

[24] M. Matboli *et al.*, "Machine learning-based stratification of prediabetes and type 2 diabetes progression," *Diabetol. Metab. Syndr.,* 17(1), 227, 2025.

[25] M. R. C. Acosta, S. Ahmed, C. E. Garcia, and I. Koo, "Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks," *IEEE Access,* 8, 19921-19933, 2020.

[26] S. Zhang, "Drug usage classification based on personality and demographic features using a combination of sampling and machine learning algorithms," *Comput. Methods Biomech. Biomed. Eng.,* 1-22, 2025.

[27] M. Seyyedattar, S. Zendehboudi, and S. Butt, "Relative permeability modeling using extra trees, ANFIS, and hybrid LSSVM–CSA methods," *Nat. Resour. Res.,* 31(1), 571-600, 2022.

[28] W. Kong, P. Hou, X. Liang, F. Gao, and Q. Liu, "An interpretable rockburst prediction model based on SSA-CatBoost," *Tunn. Undergr. Space Technol.,* 164, 106820, 2025.

[29] M. Rahimi *et al.*, "Meticulous estimation of maize actual evapotranspiration: A comprehensive explainable CatBoost algorithm reinforced with Jackknife uncertainty paradigm," *Comput. Electron. Agric.,* 237, 110599, 2025.

[30] Y. Hu *et al.*, "Predictive optimization of educational buildings' environmental performance under future climate scenarios using Catboost and SHAP," *Sol. Energy,* 300, 113746, 2025.

[31] Z. Fan, J. Gou, and S. Weng, "Complementary CatBoost based on residual error for student performance prediction," *Pattern Recognit.,* 161, 111265, 2025.

[32] M. H. Sulaiman, Z. Mustaffa, A. S. Samsudin, A. I. Mohamed, and M. M. Saari, "Electric vehicle battery state of charge estimation using metaheuristic-optimized CatBoost algorithms," *Franklin Open,* 11, 100293, 2025.

[33] X. Zhang, H. Wang, G. Yu, and W. Zhang, "Machine learning-driven prediction of hospital admissions using gradient boosting and GPT-2," *Digit. Health,* 11, 20552076251331319, 2025.

[34] M. K. Hossen and M. S. Uddin, "From data to insights: Using gradient boosting classifier to optimize student engagement in online classes with explainable AI," *Educ. Inf. Technol.,* 30(13), 18089-18130, 2025.

[35] E. Ismail, W. Gad, and M. Hashem, "HEC-ASD: A hybrid ensemble-based classification model for predicting autism spectrum disorder disease genes," *BMC Bioinformatics,* 23(1), 554, 2022.

[36] L. W. Rizkallah, "Enhancing the performance of gradient boosting trees on regression problems," *J. Big Data,* 12(1), 35, 2025.

[37] S. Rahman, M. Irfan, M. Raza, K. Moyeezullah Ghori, S. Yaqoob, and M. Awais, "Performance analysis of boosting classifiers in recognizing activities of daily living," *Int. J. Environ. Res. Public Health,* 17(3), 1082, 2020.

[38] A. Fatty, A.-J. Li, and Z.-G. Qian, "An interpretable evolutionary extreme gradient boosting algorithm for rock slope stability assessment," *Multimed. Tools Appl.,* 83(16), 46851–46874, 2024.

[39] I. B. Mustapha *et al.*, "Comparative analysis of gradient-boosting ensembles for estimation of compressive strength of quaternary blend concrete," *Int. J. Concr. Struct. Mater.,* 18(1), 20, 2024.

[40] H. Emami, S. Emami, and V. Rezaverdinejad, "A backtracking search-based extreme gradient boosting algorithm for soil moisture prediction using meteorological variables," *Earth Sci. Inf.,* 18(2), 181, 2025.

[41] M. Achite, H. Nasiri, O. M. Katipoğlu, M. Abdallah, R. Moazenzadeh, and B. Mohammadi, "A coupled extreme gradient boosting-MPA approach for estimating daily reference evapotranspiration," *Theor. Appl. Climatol.,* 156(2), 113, 2025.

[42] T. Kavzoglu and A. Teke, "Advanced hyperparameter optimization for improved spatial prediction of shallow landslides using extreme gradient boosting (XGBoost)," *Bull. Eng. Geol. Environ.,* 81(5), 201, 2022.

[43] Q. Guo, H. Wang, and Y. Tian, "Automated algorithm selection for black-box optimization

using light gradient boosting machine," *Swarm Evol. Comput.*, 98, 102071, 2025.

[44] T. O. Omotehinwa, D. O. Oyewola, and E. G. Moung, "Optimizing the light gradient-boosting machine algorithm for an efficient early detection of coronary heart disease," *Inf. Health*, 1(2), 70–81, 2024.

[45] S. Zhang, Z. Wang, and X. Su, "A study on the interpretability of network attack prediction models based on light gradient boosting machine (LGBM) and SHapley additive explanations (SHAP)," *Comput. Mater. Continua*, 83(3), 5781–5809, 2025.

[46] S. Radhika, A. Prasanth, and K. K. D. Sowndarya, "A reliable speech emotion recognition framework for multi-regional languages using optimized light gradient boosting machine classifier," *Biomed. Signal Process. Control*, 105, 107636, 2025.

[47] M. Elattar, A. Younes, I. Gad, and I. Elkabani, "Explainable AI model for PDFMal detection based on gradient boosting model," *Neural Comput. Appl.*, 36(34), 21607–21622, 2024.

[48] M. Tamim Kashifi and I. Ahmad, "Efficient histogram-based gradient boosting approach for accident severity prediction with multisource data," *Transp. Res. Rec.*, 2676(6), 236–258, 2022.

[49] M. Saied, S. Guirguis, and M. Madbouly, "A comparative study of using boosting-based machine learning algorithms for IoT network intrusion detection," *Int. J. Comput. Intell. Syst.*, 16(1), 177, 2023.

[50] S. Seth, G. Singh, and K. Kaur Chahal, "A novel time-efficient learning-based approach for smart intrusion detection system," *J. Big Data*, 8(1), 111, 2021.

[51] A. Habib, B. Alibrahim, M. Z. Alnunu, H. Moussa, and M. Habib, "Comprehensive assessment on estimating the thermodynamic and mechanical properties of multicomponent Fe–Cr-based alloys using machine learning techniques," *Discover Mater.*, 5(1), 76, 2025.

[52] I. Chhillar and A. Singh, "An improved soft voting-based machine learning technique to detect breast cancer utilizing effective feature selection and SMOTE-ENN class balancing," *Discover Artif. Intell.*, 5(1), 4, 2025.

[53] R. Dey and R. Mathur, "Ensemble learning method using stacking with base learner, a comparison," in *Proc. Int. Conf. Data Analytics Insights (ICDAI)*, Singapore: Springer, 2023, 159–169.

[54] B. A. Ture, A. Akbulut, A. H. Zaim, and C. Catal, "Stacking-based ensemble learning for remaining useful life estimation," *Soft Comput.*, 28(2), 1337–1349, Jan. 2024.

[55] H. Ye, H. Qin, Y. Tang, N. Ungvijanpunya, and Y. Gou, "Mapping an intelligent algorithm for predicting female adolescents' cervical vertebrae maturation stage with high recall and accuracy," *Prog. Orthod.*, 25(1), 20, 2024.

[56] P. Singh *et al.*, "An ensemble-driven machine learning framework for enhanced water quality classification," *Discover Sustain.*, 6(1), 552, 2025.

[57] D. K. Dake, E. Nwiah, G. S. Klogo, and W. X. Ativi, "Instructor-assisted question classification system using machine learning algorithms with N-gram and weighting schemes," *Discover Artif. Intell.*, 3(1), 29, 2023.

[58] A. Cisneros Eufracio, R. S. Perez Alvarado, J. A. Rosales Huamani, U. R. Villanueva, J. L. Castillo Sequera, and J. M. Gomez Pulido, "Rock block fall prediction prototype by structural control applied to slopes using Quantum Machine Learning (QML)," *J. Supercomput.*, 81(2), 422, 2025.

[59] M. Abdelsattar, A. AbdelMoety, and A. Emad-Eldeen, "Advanced machine learning techniques for predicting power generation and fault detection in solar photovoltaic systems," *Neural Comput. Appl.*, 37(15), 8825–8844, 2025.

[60] H. GhorbanTanhaei, P. Boozary, S. Sheykhan, M. Rabiee, F. Rahmani, and I. Hosseini, "Predictive analytics in customer behavior: Anticipating trends and preferences," *Results Control Optim.*, 17, 100462, 2024.