IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Journal Européen des Systèmes Automatisés
Vol. 59, No. 10, October, 2026, pp. 2809-2817
Journal homepage: http://iieta.org/journals/jesa

Comparative Evaluation of Data Engineering Methods in Automated Machine Learning Workflows for Multiclass Crop Recommendation

Ana C. Umaquinga-Criollo[1,2], Vladimir Bonilla[3*], Diego H. Peluffo-Ordóñez[4]

[1] Faculty of Engineering, Pontificia Universidad Católica del Ecuador, Quito 170525, Ecuador
[2] Faculty of Applied Sciences Engineering, Universidad Técnica del Norte, Ibarra 100150, Ecuador
[3] Faculty of Digital Engineering and Emerging Technologies, Universidad Internacional del Ecuador, Quito 170411, Ecuador
[4] School of Mathematical and Computational Sciences, Universidad Yachay Tech, Urcuquí 100115, Ecuador

Corresponding Author Email: febonillave@uide.edu.ec

Copyright: ©2026 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/jesa.591005

Received: 20 July 2026
Revised: 4 September 2026
Accepted: 24 September 2026
Available online: 31 October 2026

Keywords:
Automated Machine Learning, data mining, crop variables, classification algorithms

ABSTRACT

Selecting appropriate machine learning algorithms for crop recommendation often requires substantial computational resources and domain expertise. While Automated Machine Learning (AutoML) simplifies model comparison and selection, standard workflows may overlook how data preprocessing affects classifier behavior. This study uses the PyCaret framework to benchmark classification configurations for multiclass crop recommendation using the Harvard Dataverse dataset (1,594 records and seven soil and climatic variables). Eighteen algorithms are evaluated across seven preprocessing scenarios, including Z-score normalization, Principal Component Analysis (PCA), Synthetic Minority Over-sampling Technique (SMOTE), and their combinations. Model performance is assessed using stratified 10-fold cross-validation and an isolated 10% hold-out set, considering accuracy, area under the receiver operating characteristic curve (AUC), F1-score, Cohen's Kappa, and the Matthews Correlation Coefficient (MCC). PCA combined with Quadratic Discriminant Analysis (PCA+QDA) provides the most favorable balance between predictive performance and reported training time, achieving an Accuracy, AUC, F1-score, Cohen's Kappa, and MCC of 1.0000, with a training time of 0.055 s. Across the ten cross-validation folds, Accuracy remains stable at 1.0000 ± 0.0000, while the reserved hold-out set (n = 150) achieves an Accuracy of 1.0000, with a 95% exact binomial confidence interval of 0.9757–1.0000. PCA reduces redundancy among correlated predictors by transforming the original variables into orthogonal components, providing a suitable representation for QDA to estimate class-specific covariance structures. These results indicate stable performance within the evaluated dataset; however, validation on independent agricultural datasets is required to determine whether this performance is maintained across different regions, seasons, and field conditions.

1. INTRODUCTION

The increasing availability of agricultural data driven by advances in sensing technologies, remote monitoring, and digital farming platforms has positioned AI and Machine Learning (ML) as key enablers of data-driven decision-making in modern agriculture. Recent studies demonstrate that ML-based approaches can effectively support critical tasks such as crop yield prediction [1], crop classification and mapping [2], soil physicochemical analysis, and the early detection of pests and diseases via wireless sensor networks, hyperspectral imagery, and deep learning techniques [3-5]. Collectively, these applications highlight the growing role of AI as a decision-support mechanism for improving agricultural productivity and sustainability.

However, full practical adoption remains bottlenecked by the intrinsic complexities of agricultural datasets, which typically exhibit high dimensionality, non-linear interactions, and severe multicollinearity among local climatic and edaphic variables. These characteristics impose strict constraints on conventional ML workflows, severely limiting their generalization capabilities across heterogeneous agroecological environments [6, 7]. This challenge is further compounded by the standard practice of addressing preprocessing, algorithm selection, and hyperparameter tuning in isolation. Rather than integrating these stages, this structural disconnect conceals the intricate feedback loops between initial data transformations and subsequent classifier behavior. In precision agriculture, such interactions assume a critical role, given that data engineering choices directly alter the geometry of the feature space, thereby compromising predictive stability, model robustness, and agronomic interpretability itself.

While data mining techniques and conventional ML models have been applied to crop recommendation problems [7-12], existing implementations predominantly treat data preprocessing, feature selection, and algorithm tuning as decoupled, sequential steps. This static paradigm conceals the

critical feedback loops existing between feature space transformations and downstream classifier behavior. Specifically, raw edaphic and climatic variables often exhibit severe physical multicollinearity and non-linear dependencies. When fed directly into complex classifiers, these structural properties distort decision boundaries, increase parameter search spaces, and lead to unnecessary computational overhead or overfitting. Automated Machine Learning (AutoML) has recently emerged to systematize pipeline optimization and reduce manual intervention [13, 14]. However, an important methodological gap remains in the current literature: AutoML frameworks are often applied primarily as model-search and selection tools operating on predefined input representations. Relatively few studies provide structured benchmarks that systematically examine how distinct data-preconditioning strategies such as orthogonal feature transformation, variance scaling, or synthetic oversampling interact with classifiers characterized by different inductive biases [8, 15, 16]. Consequently, the comparative effect of alternative preprocessing strategies across different classifier families remains insufficiently characterized, particularly within controlled AutoML workflows.

To address this gap, this study establishes a controlled benchmarking framework that evaluates seven structured preprocessing topologies across eighteen supervised classification algorithms within an AutoML environment. By examining the interaction between Principal Component Analysis (PCA) and Quadratic Discriminant Analysis (QDA), this research provides empirical evidence that feature-space transformation can influence classifier behavior and reported training time in multiclass crop recommendation. Guided by the experimental results, this research provides three main contributions: (i) a systematic comparative evaluation of seven data-preprocessing scenarios across eighteen supervised classifiers within a controlled AutoML workflow, (ii) an empirical analysis of the interaction between preprocessing strategies and classifier performance, with particular attention to PCA+QDA, and (iii) a transparent and reproducible experimental framework for evaluating data-engineering strategies in crop recommendation.

2. MATERIALS AND METHODS

2.1 Automated Machine Learning framework

This benchmark employs an AutoML workflow to evaluate data preprocessing strategies and classification algorithms for multiclass crop recommendation. PyCaret (v3.2.0) serves as the computational framework for implementing the experimental pipeline, integrating data preprocessing, feature transformation, model training, comparison, and validation within a unified environment [17-19]. This design enables the comparative assessment of classifiers with different learning mechanisms while maintaining consistent data partitions and validation procedures across all experimental configurations. During model benchmarking, the same experimental protocol is applied to each preprocessing configuration. The experimental environment is initialized with a fixed random seed (session_id = 123), automated preprocessing (preprocess = True), and an internal training proportion of 80% (train_size = 0.8). Eighteen supervised classification algorithms are evaluated using stratified 10-fold cross-validation, which preserves class proportions across folds. Model selection is performed using PyCaret's compare_models() procedure under the same validation scheme for all preprocessing configurations. No additional user-defined hyperparameter search is introduced during this stage; therefore, the classifiers retain the parameter settings generated by their corresponding PyCaret/scikit-learn implementations. The selected pipeline is subsequently assessed on an independent 10% stratified hold-out set, which remains isolated from preprocessing, model comparison, and model selection.

2.2 Computational environment and reproducibility

All experiments are executed in Google Colab Pro using PyCaret 3.2.0. Supporting libraries include SQLAlchemy, Uvicorn, Python-multipart, FastAPI, Pandas, NumPy, Matplotlib, and Seaborn. A fixed session_id of 123 is maintained across all experimental configurations to ensure reproducible data partitions and model comparisons. The same computational environment and experimental settings are used throughout the seven preprocessing configurations.

2.3 Dataset and crop recommendation problem

The empirical data were sourced from the Harvard Dataverse repository [20]. Following data cleaning protocols, which entailed purging missing values, empty records, and duplicate entries, the finalized dataset comprised 1594 observations and eight attributes. The feature space consists of seven continuous predictors describing soil and environmental conditions, whereas the categorical target variable defines a multiclass crop recommendation problem encompassing 15 distinct crop categories. The technical specifications and statistical distributions of these variables are outlined comprehensively in Table 1.

Table 1. Summary of variables in the crop recommendation data

Variable	Type	Description
N	Continuous	Nitrogen content in soil
P	Continuous	Phosphorus content in soil
K	Continuous	Potassium content in soil
Temperature	Continuous	Soil temperature (°C)
Humidity	Continuous	Soil humidity (%)
pH	Continuous	Soil acidity/alkalinity
Rainfall	Continuous	Precipitation (mm)
Label	Categorical	Crop type (15 classes)

2.4 Exploratory data analysis

To systematically map the underlying data topology, the exploratory phase prioritized characterizing variable distributions, identifying anomalies, and uncovering multi-variable dependencies. Methodological verification of data integrity and structural variance relied on a combination of descriptive statistics and diagnostic visualizations, namely histograms, boxplots, and scatterplots. Notably, the emergence of prominent collinear patterns among the edaphic variables provided the empirical justification for integrating dimensionality reduction techniques as explicit experimental treatments within the subsequent modeling pipeline, rather than treating them as static baseline configurations (Figure 1).

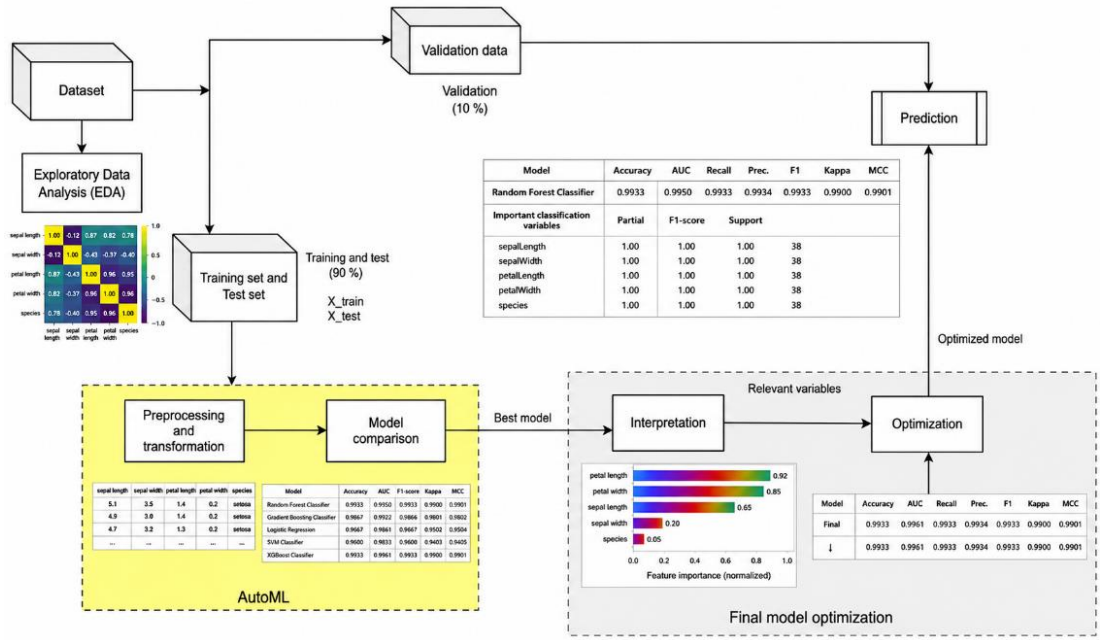

Figure 1. Block diagram representing the methodology followed in this study, including data preparation, data splitting, Automated Machine Learning (AutoML) execution, model evaluation, and prediction

2.5 Experimental design and validation

The experimental protocol is structured to prevent information leakage and preserve unbiased performance estimation. Before any AutoML procedure, a stratified 10% subset is reserved as an independent hold-out set and remains excluded from model development. The remaining data constitute the development set used by PyCaret. Within this set, train_size = 0.8 assigns 80% of the observations to model training, while the remaining 20% forms the internal test partition.

Model comparison is conducted using stratified 10-fold cross-validation under the same validation protocol for all seven preprocessing configurations. The hold-out set is not used for preprocessing, model comparison, or model selection and is accessed only for the final assessment of the selected pipeline.

2.6 Preprocessing scenarios

Seven preprocessing configurations are evaluated to assess the effect of normalization, dimensionality reduction, and oversampling on multiclass crop recommendation (Table 2). Class proportions range from 6.27% to 8.72%, indicating a relatively balanced target distribution. Accordingly, Synthetic Minority Over-sampling Technique (SMOTE) is included as an experimental preprocessing condition rather than as a necessary correction for severe class imbalance, allowing its effect on classifier performance to be compared with the original, normalized, and PCA-transformed data.

The seven preprocessing scenarios are implemented through PyCaret's setup() function under the same experimental protocol. Z-score normalization standardizes numerical predictors to zero mean and unit variance (normalize=True), whereas PCA is implemented using the linear decomposition method (pca=True, pca_method='linear'). SMOTE is activated through fix_imbalance=True and fix_imbalance_method='smote'. For combined scenarios, the corresponding transformations are integrated within the same PyCaret preprocessing pipeline, as specified in Table 2. All configurations use preprocess=True, and no additional preprocessing procedure is applied outside the specified PyCaret pipeline.

Table 2. Preprocessing scenarios evaluated in the Automated Machine Learning (AutoML) pipeline

Scenario	Preprocessing Configuration	PyCaret Settings
S1	Original data (baseline)	Default preprocessing
S2	Z-score normalization	normalize=True
S3	Principal Component Analysis (PCA)	pca=True; pca_method='linear'
S4	Synthetic Minority Over-sampling Technique (SMOTE)	fix_imbalance=True; fix_imbalance_method='smote'
S5	Z-score + PCA + SMOTE	normalize=True; pca=True; pca_method='linear'; fix_imbalance=True; fix_imbalance_method='smote'
S6	Z-score + SMOTE	normalize=True; fix_imbalance=True; fix_imbalance_method='smote'
S7	Z-score + PCA	normalize=True; pca=True; pca_method='linear'

2.7 Algorithmic taxonomy and classification models

To guarantee a comprehensive evaluation of classification mechanisms, the PyCaret v3.2.0 core architecture was leveraged to deploy a diverse matrix of eighteen supervised learning algorithms. This computational suite spans traditional statistical formulations, probabilistic classifiers, instance-based learners, and discriminant analyses, alongside tree-based structures, ensemble architectures, and advanced

gradient boosting frameworks. Rather than relying on a single family of algorithms, this diverse taxonomy enables a systematic benchmarking of models characterized by fundamentally distinct inductive biases and mathematical structures, all evaluated under identical experimental controls within a unified AutoML environment [13, 14, 17]. The operational classification models integrated into this framework are summarized systematically in Table 3.

Table 3. Qualitative description of classification models in Automated Machine Learning (AutoML)

ID	Name	Type of Classification	Reference
1	Logistic Regression	Multiclass, Binary	[15]
2	K Neighbors Classifier	Multiclass, Binary	[21]
3	Naive Bayes	Multiclass, Binary	[22]
4	Decision Tree Classifier	Multiclass, Binary	[23]
5	SVM – Linear Kernel	Binary	[24]
6	SVM – Radial Kernel	Binary	[25]
7	Gaussian Process Classifier	Multiclass, Binary	[26]
8	MLP Classifier	Multiclass, Binary	[27]
9	Ridge Classifier	Multiclass, Binary	[28]
10	Random Forest Classifier	Multiclass, Binary	[29]
11	Quadratic Discriminant Analysis	Multiclass, Binary	[30]
12	Ada Boost Classifier	Multiclass, Binary	[31]
13	Gradient Boosting Classifier	Multiclass, Binary	[32]
14	Linear Discriminant Analysis	Multiclass, Binary	[33]
15	Extra Trees Classifier	Multiclass, Binary	[34]
16	Extreme Gradient Boosting	Multiclass, Binary	[35]
17	Light Gradient Boosting Machine	Multiclass, Binary	[36]
18	Dummy Classifier	Multiclass, Binary	[37]

2.8 Evaluation metrics

To comprehensively quantify classification performance, model performance is evaluated using a set of statistical indicators: Accuracy, area under the receiver operating characteristic (ROC) curve (AUC), Precision, Recall, F1-score, Cohen's Kappa, and the Matthews Correlation Coefficient (MCC). While Accuracy provides an overall measure of correct classification, Cohen's Kappa and MCC complement this assessment by accounting for chance agreement and class distribution, which is particularly relevant in a multiclass classification setting. Final model performance is additionally assessed on the isolated hold-out subset, providing an evaluation on observations not used during model development.

In addition to the cross-validation metrics, statistical uncertainty in the final hold-out evaluation is quantified using a 95% exact binomial confidence interval for classification accuracy. Model stability is also examined by reporting the mean and standard deviation of the performance metrics across the ten stratified cross-validation folds. Finally, the difference between mean cross-validation accuracy and hold-out accuracy is calculated to identify potential performance degradation between the internal and final validation stages.

3. RESULTS

3.1 Automated Machine Learning performance evaluation

The execution of the predictive workflow was managed via the AutoML paradigm within PyCaret v3.2.0, utilizing the core configuration parameters defined in the methodology. The initial data allocation protocols successfully partitioned the 1594 original records, isolating 10% (150 observations) for independent hold-out evaluation to secure an uncompromised final evaluation framework. The remaining 90% of the data matrix (1444 observations) was funneled into the core modeling pipeline, where an internal 80/20 stratified split was enforced to allocate 1154 records for algorithmic training and 290 records for testing and cross-validation assessment.

3.1.1 Preprocessing and experimental execution

To systematically isolate the empirical impact of feature space perturbations on multiclass crop recommendation, the AutoML workflow evaluated eighteen supervised classifiers across seven distinct preprocessing configurations. Built upon PyCaret v3.2.0, the experimental environment applied the common preprocessing procedures defined in the workflow and an 80/20 stratified split within the development set, resulting in 1,154 observations for training and 290 observations for internal testing. The experimental matrix subjected these baseline data to controlled variations: (i) unaltered baseline states, (ii) isolated Z-score normalization, (iii) PCA dimensionality reduction, (iv) synthetic minority oversampling (SMOTE), and three higher-order combinatorial paths: Z-score + PCA + SMOTE, Z-score + SMOTE, and Z-score + PCA. The comparative results across these configurations, summarized in Table 4, show that PCA combined with Quadratic Discriminant Analysis (PCA+QDA) provides the most favorable balance between predictive performance and reported training time.

QDA models class-conditional feature distributions under a multivariate Gaussian assumption and allows each class to have its own covariance matrix, resulting in quadratic decision boundaries. This formulation allows QDA to represent class-specific covariance structures and nonlinear decision boundaries in the transformed feature space. With priors = None, reg_param = 0.0, store_covariance = False, and tol = 0.0001, the PCA+QDA configuration achieved scores of 1.0000 across all evaluated performance metrics: Accuracy, AUC, Recall, Precision, F1-score, Cohen's Kappa, and the MCC. Beyond predictive performance, the PCA+QDA configuration exhibits a low reported training time, completing the training phase in 0.055 s under the specified computational environment. PCA transforms the original feature space into orthogonal components, reducing redundancy among correlated predictors and providing a suitable representation for QDA to estimate class-specific covariance structures. Under the evaluated experimental conditions, this combination achieves high predictive performance while maintaining a low reported training time.

3.1.2 Post-Hoc interpretation via Shapley Additive exPlanations

To examine the contribution of the PCA-derived features to the predictions of the selected PCA+QDA pipeline, a post-hoc interpretability analysis was conducted using Shapley Additive exPlanations (SHAP). Given that the classification framework operates on an orthogonal feature space derived

from PCA, the computed Shapley values evaluate feature importance directly across the principal components (PCs) rather than the raw edaphic attributes. The distribution of SHAP values indicates that the first PCs contribute substantially to the model predictions. These components account for a large proportion of the variance represented in the PCA-transformed feature space. Inspection of the PCA loadings indicates that Nitrogen, Phosphorus, and Potassium contribute substantially to the leading PCs identified as influential in the SHAP analysis. These results suggest that the predictive behavior of the PCA+QDA pipeline is associated with variation captured by the leading PCs; however, interpretation in terms of the original agronomic variables depends on the corresponding PCA loadings.

Table 4. Comparative performance of Automated Machine Learning (AutoML) models across different data preprocessing scenarios

Preprocessing Scenario	Best Model Obtained with AutoML	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
Original	QuadraticDiscriminantAnalysis(priors=None, reg_param=0.0, store_covariance=False, tol=0.0001)	1	1	1	1	1	1	1	0.085
Z-Score	GaussianNB(priors=None, var_smoothing=1e-09)	0.9991	1	0.999	0.999	0.999	0.999	0.999	0.052
Principal Component Analysis (PCA)	QuadraticDiscriminantAnalysis(priors=None, reg_param=0.0, store_covariance=False, tol=0.0001)	1	1	1	1	1	1	1	0.055
Synthetic Minority Over-sampling Technique (SMOTE)	DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None, criterion='gini', max_depth=None, max_features=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, random_state=123, splitter='best')	0.9991	1	0.999	0.999	0.999	0.999	0.999	0.047
Z-Score + PCA + SMOTE	QuadraticDiscriminantAnalysis(priors=None, reg_param=0.0, store_covariance=False, tol=0.0001)	1	1	1	1	1	1	1	0.093
Z-Score + SMOTE	DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None, criterion='gini', max_depth=None, max_features=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, random_state=123, splitter='best')	1	1	1	1	1	1	1	0.268
Z-Score + PCA	QuadraticDiscriminantAnalysis(priors=None, reg_param=0.0, store_covariance=False, tol=0.0001)	0.9991	1	0.999	0.999	0.999	0.999	0.999	0.096

3.1.3 Hold-out evaluation and generalization assessment

To assess predictive performance on previously unseen observations, a hold-out evaluation is conducted using the isolated 10% stratified hold-out set (n = 150). This validation set remains completely isolated from preprocessing, cross-validation, model comparison, and model selection, and is used only for the final evaluation of the selected pipeline detailed in Table 4. When subjected to this unseen data matrix, the selected PCA+QDA pipeline achieves total metric convergence, securing a perfect score of 1.0000 across Accuracy, Precision, Recall, F1-score, Cohen's Kappa, and the MCC. The perfect classification obtained on the hold-out observations indicates stable predictive performance under the evaluated experimental conditions. However, because the hold-out observations originate from the same source dataset, these results support generalization within the evaluated data distribution but do not establish broader generalization to different agricultural environments (See Figures 2 and 3).

3.1.4 Statistical validation and model robustness

The statistical stability of the selected PCA+QDA pipeline is assessed using the stratified 10-fold cross-validation results. Across the ten folds, the model achieves an Accuracy, F1-score, and MCC of 1.0000 ± 0.0000, showing no variation in these metrics across the evaluated partitions. The same behavior is observed for AUC and Cohen's Kappa, both reaching 1.0000 ± 0.0000.

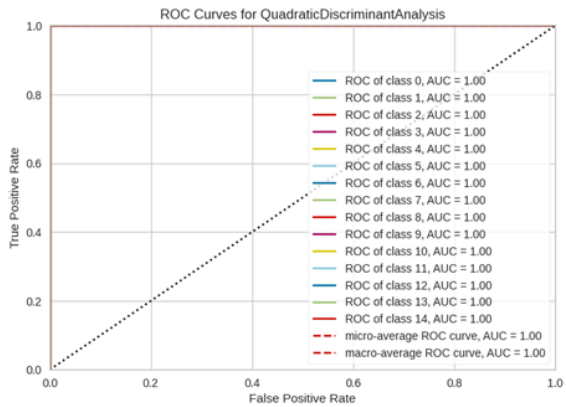

Figure 2. Receiver operating characteristic (ROC) curve of the quadratic discriminant analysis (QDA) model

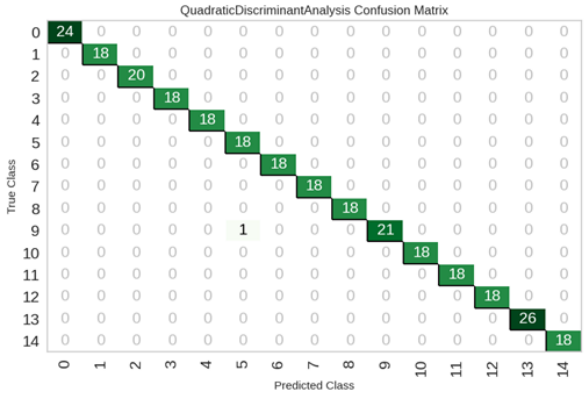

Figure 3. Confusion matrix of the quadratic discriminant analysis (QDA) model

The pipeline is then evaluated on the independent stratified hold-out set (n = 150), where all observations are correctly classified (Accuracy = 1.0000). To account for the uncertainty associated with this finite validation sample, the 95% exact binomial confidence interval for Accuracy is calculated as 0.9757–1.0000. In addition, the observed difference between mean cross-validation Accuracy and hold-out Accuracy is 0.0000, indicating consistent predictive performance between the two validation stages under the evaluated data distribution.

Although these results indicate stable performance under the evaluated experimental conditions, they do not rule out potential overfitting to dataset-specific characteristics. The hold-out partition is drawn from the same source dataset and therefore shares its underlying sampling characteristics. Consequently, the observed stability supports consistent performance on unseen observations drawn from the evaluated data distribution; however, it should not be interpreted as evidence that overfitting is absent. Broader robustness and external generalization require validation on independent datasets collected across different regions, seasons, and field conditions.

3.1.5 Architectural deployment viability

Beyond predictive performance, the PCA+QDA configuration shows a low reported training time under the evaluated experimental conditions, completing model training in 0.055 s in the specified Google Colab Pro environment. The dimensionality reduction provided by PCA, together with the relatively simple structure of QDA, suggests potential for integration into computationally constrained agricultural decision-support systems. However, the present study does not directly evaluate deployment performance; therefore, the reported training time should not be interpreted as evidence of real-time inference or suitability for edge devices. Practical deployment requires additional assessment of inference latency, memory usage, energy consumption, and hardware-specific performance.

4. DISCUSSION

The findings of this study indicate that AutoML provides a systematic framework for comparing preprocessing configurations alongside diverse classification algorithms in crop recommendation tasks. The high predictive performance observed across the seven evaluated scenarios suggests that the soil and environmental variables in the analyzed dataset provide substantial discriminatory information for distinguishing among the crop classes.

The architecture that yielded the best results combined PCA with QDA. PCA transforms the correlated predictors into orthogonal components, reducing redundancy in the representation supplied to QDA. This transformation is particularly relevant to QDA because the classifier estimates a separate covariance matrix for each class and uses these estimates to construct quadratic decision boundaries. A less redundant feature representation can therefore facilitate covariance estimation while preserving class-specific distributional differences. Under the evaluated dataset, this interaction provides a plausible explanation for the strong performance observed with the PCA+QDA pipeline.

The strong performance of QDA suggests that the PCA-transformed feature space provides favorable class separation while retaining class-specific covariance structures that can be modeled through quadratic decision boundaries. By employing quadratic decision boundaries, this algorithm can accommodate class-specific covariance structures and more complex class boundaries than linear discriminant models, a relevant property when working with agricultural data where factors such as pH, nutrients, rainfall, humidity, and temperature may exhibit complex relationships across crop classes.

The exceptionally high classification scores require careful interpretation, particularly because Accuracy, AUC, F1-score, and MCC approach or reach 1.0000 across multiple preprocessing configurations. Methodologically, the independent 10% stratified hold-out set (n = 150) remains isolated from preprocessing, cross-validation, model comparison, and model selection, providing an independent assessment of the selected pipeline on previously unseen observations. From an agronomic perspective, the high performance is also consistent with the discriminative structure of the dataset, where soil and environmental predictors such as Nitrogen, Phosphorus, Potassium, and Rainfall provide substantial differentiation among crop classes under the sampled conditions.

Phosphorus and Potassium show a relatively strong correlation in the original dataset (r = 0.73), indicating redundancy between these predictors. PCA reduces this redundancy by transforming the original variables into orthogonal components. This transformation is relevant to QDA because the classifier estimates a separate covariance structure for each class and defines quadratic decision boundaries. The interaction between the PCA representation and the QDA classification mechanism may contribute to the high performance observed in this configuration. However, the hold-out set is drawn from the same dataset as the training partition. Thus, the results demonstrate strong performance on unseen observations from this dataset, while validation with independent data from other regions, seasons, and field conditions is still required.

Comparison across preprocessing scenarios shows that normalization, PCA, and SMOTE are associated with differences in predictive performance and reported training time. SMOTE-based configurations achieve perfect or near-perfect classification scores but provide no measurable predictive advantage over PCA+QDA. This result is consistent with the relatively balanced class distribution of the original dataset, where class proportions range from 6.27% to 8.72%. Under these conditions, synthetic oversampling does not improve the maximum predictive performance obtained

without class balancing.

The advantage of PCA+QDA is therefore observed in joint consideration of predictive performance and reported training time rather than in classification accuracy alone. PCA+QDA achieves perfect benchmark metrics with a training time of 0.055 s, compared with 0.085 s for QDA on the original data, 0.093 s for Z-score+PCA+SMOTE with QDA, and 0.268 s for Z-score+SMOTE with Decision Tree. Thus, among the configurations reaching perfect classification scores, PCA+QDA provides the lowest training time while maintaining the same predictive performance.

In synthesis, the results support the adoption of AutoML as a transparent, auditable, and reproducible framework for solving classification problems in the agricultural sector. This systematic comparison provides empirical evidence that feature-engineering choices can alter model behavior, even when global performance metrics are similar or near-perfect.

4.1 Technical rationality of the Automated Machine Learning pipeline

The observed performance of the selected configuration can be interpreted in relation to the interaction between the PCA-based feature representation and the statistical characteristics of QDA. The automated framework facilitated a systematic evaluation across the hypothesis space, providing empirical evidence that the integration of PCA with QDA can achieve high predictive performance with a low reported training time under the evaluated experimental conditions. From the perspective of statistical learning theory, tree-based ensemble models like Random Forest or LightGBM rely on recursive space partitioning that increases execution runtime under complex data conditions; conversely, QDA resolves category assignment via deterministic quadratic discriminant functions.

The structural preconditioning provided by PCA may contribute to the favorable behavior observed with QDA in the evaluated dataset. In its raw state, the edaphic matrix suffers from poor conditioning due to the physical multicollinearity of soil nutrients (e.g., the baseline coupling between Phosphorus and Potassium, $r = 0.73$). By projecting the raw feature space onto orthogonal components, PCA reduces linear redundancy among the original predictors and may improve the numerical conditioning of the feature representation used for class-specific covariance estimation. This linear transformation refines the dimensional coordinates, allowing QDA to independently model the class-specific covariance structure through quadratic decision boundaries, in accordance with its statistical assumptions. With reg_param = 0.0 and priors = None, QDA estimates class-specific covariance structures and class probabilities directly from the available training data, without additional covariance regularization or user-defined class priors. Consequently, this configuration shows that the combination of PCA and QDA can achieve high predictive performance with a low reported training time under the evaluated experimental conditions.

4.2 Practical implications, limitations, and future research

From a practical perspective, the PCA+QDA configuration combines high predictive performance with a low reported training time, reaching 0.055 s under the specified experimental environment. This characteristic is relevant for crop recommendation systems that may require periodic retraining as new soil and environmental measurements become available. Moreover, the use of seven soil and climatic predictors provides a compact input structure that may facilitate integration into agricultural decision-support applications. However, the reported training time should not be interpreted as direct evidence of real-time or edge-device performance, which requires specific evaluation of inference latency, memory requirements, energy consumption, and hardware-specific behavior.

Several limitations should be considered when interpreting these findings. Although the final hold-out partition remains isolated during model development, its observations originate from the same source dataset and therefore share its sampling characteristics. Consequently, the reported performance supports generalization within the evaluated data distribution but does not establish external generalization across different agricultural environments. In addition, the dataset comprises 1,594 observations described by seven soil and climatic predictors, which may not capture the variability associated with different geographic regions, growing seasons, management practices, or sensing conditions. The near-perfect classification observed across several preprocessing configurations may also partly reflect the high discriminative structure of this particular dataset.

Future research should prioritize external validation using datasets collected across different geographic regions, growing seasons, soil conditions, and agricultural management practices. The PCA+QDA configuration should also be evaluated under distribution shifts, sensor noise, missing measurements, and naturally imbalanced class distributions. From a deployment perspective, further experiments should quantify inference latency, memory usage, and energy consumption on representative hardware. Incorporating additional agronomic variables and temporally collected observations may further determine whether the observed performance is maintained under more heterogeneous field conditions.

5. CONCLUSIONS

This comparative study evaluates the AutoML paradigm as a systematic framework for comparing predictive performance and training time across multiclass classification pipelines. Within this context, the PyCaret library serves as the operational environment to integrate data standardization Z-Score, feature transformation PCA, and synthetic oversampling SMOTE across 18 classification algorithms under reproducible configurations.

The experimental design evaluates system performance across seven distinct preprocessing scenarios and modeling criteria. The PCA+QDA provides the most favorable balance between predictive performance and reported training time, achieving an Accuracy, AUC, F1-score, Kappa, and MCC of 1.0000, with a training time of 0.055 s. However, these metrics reflect the structural constraints of the current dataset. While these empirical findings indicate strong predictive performance within the evaluated dataset, they do not imply immediate, production-ready deployability.

Future research should evaluate the robustness and external generalization of the PCA+QDA configuration using independent datasets, including conditions involving sensor noise, regional variability, and naturally imbalanced class distributions. Such evaluations are necessary to determine whether the predictive performance observed in this study can

be maintained under heterogeneous field conditions and to assess the potential applicability of the proposed workflow to agricultural decision-support systems.

REFERENCES

[1] Euldji, Vanarase, V., Mane, V., Bhute, H., Tate, A., Dhar, S. (2021). Crop prediction using data mining and machine learning techniques. In 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, pp. 1764-1771. https://doi.org/10.1109/ICIRCA51532.2021.9544724

[2] Sujatha, R., Isakki, P. (2016). A study on crop yield forecasting using classification techniques. In 2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16), Kovilpatti, India, pp. 1-4. https://doi.org/10.1109/icctide.2016.7725357

[3] Senapaty, M.K., Ray, A., Padhy, N. (2024). A decision support system for crop recommendation using machine learning classification algorithms. Agriculture, 14(8): 1256. https://doi.org/10.3390/agriculture14081256

[4] Tripathy, A.K., Adinarayana, J., Vijayalakshmi, K., et al. (2014). Knowledge discovery and leaf spot dynamics of groundnut crop through wireless sensor network and data mining techniques. Computers and Electronics in Agriculture, 107: 104-114. https://doi.org/10.1016/j.compag.2014.05.009

[5] López, J.E.G., Chávez, J.C., Sánchez, A.K.J. (2017). Modelado de una red de sensores y actuadores inalámbrica para aplicaciones en agricultura de precisión. In 2017 IEEE Mexican Humanitarian Technology Conference (MHTC), Puebla, Mexico, pp. 109-116. https://doi.org/10.1109/mhtc.2017.7926210

[6] Song, C., Dong, H. (2021). Application of intelligent recommendation for agricultural information: A systematic literature review. IEEE Access, 9: 153616-153632. https://doi.org/10.1109/access.2021.3127201

[7] Food and Agriculture Organization of the United Nations. (2025). Renewable water availability per person plunges 7 percent in a decade as global scarcity deepens, FAO data shows. https://www.fao.org/newsroom/detail/renewable-water-availability-per-person-plunges-7-percent-in-a-decade-as-global-scarcity-deepens--fao-data-shows/en.

[8] Doshi, Z., Nadkarni, S., Agrawal, R., Shah, N. (2018). AgroConsultant: Intelligent crop recommendation system using machine learning algorithms. In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), pp. 1-6. https://doi.org/10.1109/iccubea.2018.8697349

[9] Kulkarni, N.H., Srinivasan, G.N., Sagar, B.M., Cauvery, N.K. (2018). Improving crop productivity through a crop recommendation system using ensembling technique. In 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), pp. 114-119. https://doi.org/10.1109/csitss.2018.8768790

[10] Kumar, R., Singh, M.P., Kumar, P., Singh, J.P. (2015). Crop selection method to maximize crop yield rate using machine learning technique. In 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), pp. 138-145. https://doi.org/10.1109/icstm.2015.7225403

[11] Martin, T., Fuentes, V., Valtchev, P., Diallo, A.B., Lacroix, R. (2022). Generalized graph pattern discovery in linked data with data properties and a domain ontology. In Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing, pp. 1890-1899. https://doi.org/10.1145/3477314.3507301

[12] Yu, M., Liu, Z., Tang, Y., Jiang, J. (2021). Recognition algorithm of e-commerce click farming based on K-means technology. In 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), pp. 103-106. https://doi.org/10.1109/icsp51882.2021.9408977

[13] Hutter, F., Kotthoff, L., Vanschoren, J. (eds.) (2019). Automated Machine Learning. Springer International Publishing. https://doi.org/10.1007/978-3-030-05318-5

[14] Shen, Z., Zhang, Y., Wei, L., Zhao, H., Yao, Q. (2018). Automated machine learning: From principles to practices. arXiv preprint arXiv:1810.13306. https://doi.org/10.48550/arXiv.1810.13306

[15] Sarker, I.H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3): 160. https://doi.org/10.1007/s42979-021-00592-x

[16] Zhang, Y.A., Sun, Z., Zhang, C., Yin, S., Wang, W., Song, R. (2021). Body weight estimation of yak based on cloud edge computing. EURASIP Journal on Wireless Communications and Networking, 2021(1): 6. https://doi.org/10.1186/s13638-020-01879-y

[17] Ali, M. (2023). PyCaret: An open-source, low-code machine learning library in Python (Version 3.2.0) [Computer software]. PyCaret documentation version 3.0.4. https://pycaret.readthedocs.io.

[18] Whig, P., Gupta, K., Jiwani, N., Jupalle, H., Kouser, S., Alam, N. (2023). A novel method for diabetes classification and prediction with Pycaret. Microsystem Technologies, 29(10): 1479-1487. https://doi.org/10.1007/s00542-023-05473-2

[19] Gal-Tzur, A., Albagli-Kim, S. (2024). Systematic analysis of the literature addressing the use of machine learning techniques in transportation—A methodology and its application. Sustainability, 16(1): 207. https://doi.org/10.3390/su16010207

[20] Singh, R. (2023). Crop recommendation dataset. Harvard Dataverse. https://dataverse.harvard.edu/file.xhtml?fileId=6881146&version=1.0.

[21] Abu Alfeilat, H.A., Hassanat, A.B.A., Lasassmeh, O., et al. (2019). Effects of distance measure choice on K-nearest neighbor classifier performance: A review. Big Data, 7(4): 221-248. https://doi.org/10.1089/big.2018.0175

[22] Berrar, D. (2019). Bayes' theorem and naive Bayes classifier. In Encyclopedia of Bioinformatics and Computational Biology, pp. 403-412. Elsevier. https://doi.org/10.1016/b978-0-12-809633-8.20473-1

[23] Mienye, I.D., Jere, N. (2024). A survey of decision trees: Concepts, algorithms, and applications. IEEE Access, 12: 86716-86727. https://doi.org/10.1109/access.2024.3416838

[24] Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., Lopez, A. (2020). A comprehensive survey on

support vector machine classification: Applications, challenges and trends. Neurocomputing, 408: 189-215. https://doi.org/10.1016/j.neucom.2019.10.118

[25] Shastry, K.A., Sanjay, H.A., Deexith, G. (2017). Quadratic-radial-basis-function-kernel for classifying multi-class agricultural datasets with continuous attributes. Applied Soft Computing, 58: 65-74. https://doi.org/10.1016/j.asoc.2017.04.049

[26] Liu, H., Ong, Y.S., Shen, X., Cai, J. (2020). When Gaussian process meets big data: A review of scalable GPs. IEEE Transactions on Neural Networks and Learning Systems, 31(11): 4405-4423. https://doi.org/10.1109/tnnls.2019.2957109

[27] Lin, R., Zhou, Z., You, S., Rao, R., Kuo, C.C.J. (2024). Geometrical interpretation and design of multilayer perceptrons. IEEE Transactions on Neural Networks and Learning Systems, 35(2): 2545-2559. https://doi.org/10.1109/TNNLS.2022.3190364

[28] Peng, C., Cheng, Q. (2021). Discriminative ridge machine: A classifier for high-dimensional data or imbalanced data. IEEE Transactions on Neural Networks and Learning Systems, 32(6): 2595-2609. https://doi.org/10.1109/tnnls.2020.3006877

[29] Probst, P., Wright, M.N., Boulesteix, A.L. (2019). Hyperparameters and tuning strategies for random forest. WIREs Data Mining and Knowledge Discovery, 9(3): e1301. https://doi.org/10.1002/widm.1301

[30] Tharwat, A. (2016). Linear vs. quadratic discriminant analysis classifier: A tutorial. International Journal of Applied Pattern Recognition, 3(2): 145-180. https://doi.org/10.1504/IJAPR.2016.079050

[31] Mienye, I.D., Sun, Y. (2022). A survey of ensemble learning: Concepts, algorithms, applications, and prospects. IEEE Access, 10: 99129-99149. https://doi.org/10.1109/ACCESS.2022.3207287

[32] Emami, S., Martínez-Muñoz, G. (2025). Condensed-gradient boosting. International Journal of Machine Learning and Cybernetics, 16: 687-701. https://doi.org/10.1007/s13042-024-02279-0

[33] Tharwat, A., Gaber, T., Ibrahim, A., Hassanien, A.E. (2017). Linear discriminant analysis: A detailed tutorial. Artificial Intelligence Communications, 30(2): 169-190. https://doi.org/10.3233/AIC-170729

[34] Prity, F.S., Hasan, M.M., Saif, S.H., et al. (2024). Enhancing agricultural productivity: A machine learning approach to crop recommendations. Human-Centric Intelligent Systems, 4(4): 497-510. https://doi.org/10.1007/s44230-024-00081-3

[35] Chen, T., Guestrin, C. (2016). XGBoost. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785-794. https://doi.org/10.1145/2939672.2939785

[36] Bentéjac, C., Csörgő, A., Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review, 54(3): 1937-1967. https://doi.org/10.1007/s10462-020-09896-5

[37] van de Bijl, E., Klein, J., Pries, J., Bhulai, S., Hoogendoorn, M., van der Mei, R.D. (2025). The Dutch draw: Constructing a universal baseline for binary classification problems. Journal of Applied Probability, 62(2): 475-493. https://doi.org/10.1017/jpr.2024.52