

# Hyperparameter Optimization in Machine Learning Classification Models: A Case Study in the Retail Industry

Kıymet Tabak<sup>1</sup> and Selçuk Alp<sup>2</sup>

<sup>1</sup>Department of Industrial Engineering, Yildiz Technical University, Istanbul, Turkey

<sup>2</sup>Department of Statistics, Yildiz Technical University, Istanbul, Turkey

(Received 24 September 2025; Revised 18 January 2025; Accepted 02 February 2026; Published online 23 March 2026)

**Abstract:** It is now important to forecast customer purchase patterns and classify stores appropriately considering the effects of the global pandemic and changing consumer preferences in the retail sector. Machine learning (ML) algorithms can be used to predict customer buying patterns and classify stores. However, conventional ML algorithms are very sensitive to their parameters. Hyperparameter optimization (HPO) is used for tuning the parameters of the model with grid search and random search algorithms and finding the best parameters. The aim of this study is to analyze which HPO approach performs best in determining parameters in ML-based classification algorithms. In this study, for the store classification in the retail sector, gradient boosting classifier, XGBClassifier, and random forest (RF) methods are selected. Model parameters are determined by using two different HPO methods for each model, and performance metrics are compared. In these three classification models, monthly sales amount, population in the store region, GDP per capita, and warehouse size of the stores are used between January 2017 and December 2021. In the analysis that evaluated the pre-pandemic and pandemic periods together in the study, RF is the best prediction algorithm with an accuracy score of 98%. The RF algorithm has an accuracy rate of 99% for both periods separately. Our analyses evaluated pre-pandemic, pandemic, and both periods together. The HPO that gives the best results is the random search tuning. This study focuses on evaluating the comparative performance of grid search and random search HPO techniques across multiple ML models for retail store classification.

**Keywords:** hyperparameter optimization; machine learning classification; retail industry

## I. INTRODUCTION

Machine learning (ML) is a broad word including many algorithms that execute intelligent predictions derived from a dataset [1]. In ML, different learning methods were used depending on the requested result. They are basically divided into three general groups: supervised, unsupervised, and reinforcement learning. Classes are predetermined in supervised algorithms. In this study, supervised learning algorithms were used because the store classes are known in advance. Store classification is not only a technical task but also a business-critical function that informs inventory decisions, staffing, and localized marketing strategies. An accurate classification helps businesses allocate resources efficiently, adjust regional product assortments, and plan promotional activities more effectively. Hence, building robust store classification models using ML is directly relevant to operational efficiency and strategic agility in retail. This classification output can be directly integrated into store segmentation strategies, tailored marketing campaigns, staffing optimization, and demand forecasting. It serves as a foundation for decision-support systems in retail management. In supervised learning from ML algorithms, the inputs of the algorithm are arranged according to the desired result/output [2]. Supervised learning is prevalent in classification problems as the objective is typically to enable the computer to learn a classification system that we have devised [3]. Commonly used supervised classification algorithms are linear classifiers, logistic regression

(LR), Naïve Bayes (NB) classifier, support vector machine (SVM), K-NN classifier, decision tree (DT), random forest (RF), artificial neural networks (ANN), Bayesian networks, etc. [2]. Many factors affect the performance of an ML model, such as the structure, size of the data, the number of classes, algorithms, performance verification methods, sampling methods, and feature selection methods. Hyperparameters also influence the efficiency of classification algorithms [4]. Hyperparameters are parameters that change according to the problem and dataset. The choice of hyperparameters usually varies according to the designer's intuition, experience from previous problems, current trends, etc. However, recently, different techniques have been put forward to choose the most suitable hyperparameter group in the most optimal way for the solution of the problem. These techniques are called hyperparameter optimization (HPO), which is also used in deep learning [5].

In this study, extreme gradient boosting (XGBoost), gradient boosting, and RF algorithms are used for the store classification in the retail sector. Random search and grid search HPO are performed to tune the parameters of these algorithms. The primary objective of this study is to comparatively evaluate the effectiveness of two widely used HPO methods—grid search and random search—in the context of a retail store classification problem. While various ML models such as RF, XGBoost, and gradient boosting are employed in the analysis, the main contribution of the study lies not in introducing a novel classification method but in demonstrating how different HPO techniques influence model performance across distinct time periods (pre-pandemic, pandemic, and both combined). In this sense, the retail classification task

Corresponding author: Selçuk Alp (e-mail: [alp@yildiz.edu.tr](mailto:alp@yildiz.edu.tr)).

**Table I.** Literature research using machine learning algorithms

| Study                          | Issue                               | Algorithms   | Accuracy (%) |
|--------------------------------|-------------------------------------|--|--------------|
| Ahmad and Aljoumaa (2019)      | Customer churn prediction           | NB, DT, ANN  | 93.30        |
| Li and Marikannan (2019)       | Customer churn prediction           | NB, DT, ANN  | 86.71        |
| Win and Bo (2020)              | Customer lifetime value             | RF, AdaBoost   | 84.27        |
| Lalwani <i>et al.</i> (2020)   | Customer churn prediction           | AdaBoost, XGBoost, LR, NB, SVM, RF, DT                       | 84.00        |
| Bentejac <i>et al.</i> (2021)  | A comparative analysis              | XGBoost, LightGBM, CatBoost, RF                              | 80.00        |
| Belete and Huchaiah (2022)     | Prediction of HIV/AIDS test results | GradientBoost, SVM, Extra Tree, K-NN, DT, AdaBoost, RF, LR   | 87.60        |
| Elgeldawi <i>et al.</i> (2021) | Arabic sentiment classification     | Bayesian optimization, PSO, GA                               | 95.62        |
| Valarmathi and Sheela (2021)   | Coronary artery disease prediction  | RF, XGBoost  | 97.20        |
| Kocoglu and Ozcan (2022)       | Customer churn prediction           | Extreme Learning Machine, NB, K-Nearest Neighbor (K-NN), SVM | 93.10        |
| Kaur <i>et al.</i> (2020)      | Prediction of Parkinson's disease   | Deep learning  | 91.69        |
| Kilinc and Rohrhirsch (2023)   | Cross-buyer predict model           | GradientBoost, RF, SVM, ANN                                  | 90.00        |
| Coskun and Çetin (2022)        | Detecting attack types              | AdaBoost, CatBoost, GradientBoost, LightGBM                  | 99.00        |
| Pfob <i>et al.</i> (2022)      | Classify breast lesion              | LR, XGBoost, MARS, SVM                                       | 81.20        |

Source: Compiled by the author(s).

serves as a real-world case study to assess HPO efficacy under practical data constraints. The data between January of 2017 and March of 2020 represented the pre-pandemic period, while data for March of 2020 and December of 2021 represented the pandemic period. This temporal segmentation enables the analysis of how major external disruptions—such as the COVID-19 pandemic—can alter customer behavior, feature distributions, and ultimately model performance. These periods differ not only in terms of temporal boundaries but also in customer behavior, purchase patterns, and store operations. By analyzing each period separately, the study aims to observe whether the structural changes in the market alter the predictive capabilities of ML models and the comparative effectiveness of hyperparameter tuning techniques. Evaluating model accuracy and optimization techniques across these distinct periods helps assess the robustness and adaptability of ML classifiers under real-world market shifts. Tabak Kızgın and Alp [31] give an empirical study that demonstrates how such seasonal impacts are reflected at the store level. The authors claim that RF-based hybrid models achieve good classification accuracy and that store clusters stay mostly consistent. In the second part of the study, literature studies on the subject, research methods in the third part, and application in the fourth part are given. Finally, the fifth part contains conclusions and discussion.

The rest of the paper is organized as follows. Section II reviews related studies in ML classification and HPO. Section III describes the research methodology, including the dataset, preprocessing steps, and applied algorithms. Section IV presents experimental results and comparative performance analyses. Section V discusses the findings, and Section VI concludes the paper with implications and directions for future research.

## II. LITERATURE REVIEW

There are many studies on classification problems using ML and deep learning algorithms. Some of the recent studies on these issues are listed below.

Lalwani *et al.* [6] used ML techniques to predict customer churn in the telecom industry and observed that XGBClassifier gave the best accuracy score. Ahmad *et al.* [7] reported that the XGBoost classifier was the model that gave the best Area Under the Curve (AUC) score in the model they created for predicting customer churn in the telecom sector data. In another study, Li and Marikannan [8] built three predictive models that used the “grid search” algorithm with NB, DT, and ANN for telecommunication customer churn predictive models and reported that DT combined with “grid search” has the highest accuracy score. Win and Bo [9] presented a prediction model with the RF algorithm to classify customers in the retail industry. In the study, they stated that they increased the accuracy rate by using random search, one of the hyperparameter search algorithms, to improve the model. The accuracy rate, which was 81.46% with the default hyperparameters, increased to 84.27% with the hyperparameter setting. Kilinc and Rohrhirsch [10] tried to improve classification performance by using the random search HPO method for four ML models (gradient boosting, RF, SVM, and ANN) using bank customer savings data. Belete and Huchaiah [11] used the grid search hyperparameter tuning algorithm while estimating the information they obtained from the HIV/AIDS dataset with classification algorithms. Bentejac *et al.* [12] compared the results obtained using ML models such as XGBoost, LightGBM, CatBoost, RF, and gradient boosting with and without hyperparameter tuning. Elgeldawi *et al.* [13] in their study on sentiment analysis optimized the parameters of six different ML algorithms (LR, ridge classifier, SVM classifier, DT, RF, and NB classifiers) using five different hyperparameter methods (grid search, random search, Bayesian optimization, particle swarm optimization (PSO), and genetic algorithm). Pfob *et al.* [14] aimed to optimize five ML algorithms (LR with Elastic Net penalty, XGBoost tree, multivariate adaptive regression spline (MARS), and SVM) in the detection of breast lesions by using grid search algorithm. Valarmathi and Sheela [15] tried HPO methods grid search, random search, and genetic programming to optimize the performance of ML models RF and XGBoost for coronary artery disease prediction. Kaur *et al.*

[16] analyzed to improve the performance of the deep learning model by optimizing with grid search algorithms to predict the early onset of Parkinson's disease.

Despite various studies applying ML to customer or sales prediction, relatively few have explored store classification as a strategic decision-support tool, especially in the context of dynamic conditions like the COVID-19 pandemic.

### III. RESEARCH METHODS

#### A. ML ALGORITHMS

Three ML algorithms (RF, gradient boosting, and XGBoost) were used in this study. HPO has been applied in the solution of these MLs. These MLs are described in the following section. RF is classified as an ensemble learning technique that produces several classifiers and consolidates their outcomes for prediction [17]. It is important to note that RF is used for two different purposes in practice. In some RF applications, the focus is on the creation of a well-accurate classification or regression rule intended to be used as a predictor on future data [18]. One ML technique that falls under the ensemble method category is the gradient boosting algorithm. AdaBoost was initially introduced by Freund and Schapire [19] for classification issues [20]. Each tree learns from the previous tree, and by improving it, it creates new trees that are more successful. To classify a sample, a weighted vote is taken from all trees in the community [10]. XGBoost is an effective and scalable application of gradient boosting, which is the tool used in competitions between ML methods due to its features such as easy parallel processing and high estimation accuracy [21]. The most important factor behind the success of XGBoost is its scalability in all scenarios [22]. The classification task in this study involves a multi-class setting with six distinct store classes, each defined by a combination of brand types and store formats. Unlike binary classification problems, multi-class classification introduces increased complexity in terms of class imbalance, decision boundary overlap, and performance metric interpretation. Therefore, algorithms capable of handling multi-class problems efficiently—such as RF, gradient boosting, and XGBoost—were selected. To accurately evaluate model performance, macro-averaged metrics such as accuracy, precision, recall, and F1-score were computed, ensuring equal weight to each class regardless of frequency. This approach addresses potential bias introduced by dominant classes in the dataset. Additionally, class distribution among the six store classes is moderately imbalanced. Therefore, macro-averaged performance metrics were preferred to ensure that all classes, including less frequent ones, are equally represented in the model evaluation.

#### B. HYPERPARAMETER OPTIMIZATION

There are two types of parameters in ML models: model parameters and hyperparameters. Model parameters are parameters that are determined at the beginning of the learning process and can be updated during the learning process (e.g., the weights of neurons in neural networks). Hyperparameters are parameters that are set before the ML model starts the learning process and are not updated during the learning process [23]. The number of decision trees ( $n\_estimators$ ), the decision tree's maximum depth ( $max\_depth$ ), the split criteria ( $criterion$ ), the minimum number of samples for internal node split ( $min\_samples\_split$ ), the minimum number of samples for leaf nodes ( $min\_samples\_leaf$ ), the maximum number

of features ( $max\_features$ ), and whether bootstrapping is enabled ( $bootstrap$ ) are among the RF classifier's tunable hyperparameters [24]. Hyperparameters that can be adjusted for XGBoost can be the learning rate ( $learning\_rate$ ), the minimum loss reduction ( $gamma$ ), the maximum depth of the tree ( $max\_depth$ ), the fraction of features to be evaluated at each split ( $colsample\_bylevel$ ), and the subsampling rate ( $subsample$ ). Hyperparameters that can be adjusted for gradient boosting can be learning rate,  $max\_depth$ ,  $subsample$ ,  $max\_features$ , and  $min\_samples\_split$  [12]. Parameters and hyperparameters of ML algorithms used in this study are given in Table II.

For a fair comparison, identical hyperparameter ranges are defined for both grid search and random search. The number of evaluated configurations is kept equal across different methods for each algorithm. Random search samples parameter combinations are uniformly applied from the same predefined ranges used in grid search. This ensured that observed performance differences reflect search strategy efficiency rather than unequal computational budgets. In this study, widely used and high-performance grid search and random search algorithms were used. Grid search is one of the common HPO techniques. The main disadvantage of grid search is its inefficiency for high-dimensional hyperparameter configuration space, as the number of evaluations increases exponentially as the number of hyperparameters increases. Random search is another HPO technique. Random search, in contrast to grid search, selects a predetermined number of parameter possibilities from the given distribution [23]. The grid search optimization algorithm considers increasing amounts of training data in the HPO stages. Initially, it uses a small subset of the training data for hyperparameter tuning by applying sequential optimization. Later, hyperparameter settings merge parameters that yield better results, allowing the most appropriate solution to be reached quickly [16]. Hyperparameter tuning is usually done manually, and a grid is progressively refined over the hyperparameter space [25]. Random search is similar to grid search in that it searches within a manually specified configuration area, but it does so by randomly selecting a set of sample points from it. This makes random search more efficient than grid search because it can explore a wider range of possible hyperparameter values in it, whereas grid search can spend too much time exploring trivial dimensions and thus insufficiently cover more important dimensions. The time complexity of the random search is linear with respect to the number of points sampled from the configuration space [26]

**Table II.** Hyperparameters of ML algorithms used in this study

| ML algorithms      | Hyperparameter       |
|--------------------|----------------------|
| RF classifier      | $n\_estimators$      |
|                    | $max\_depth$         |
|                    | $min\_simple\_split$ |
|                    | $min\_samples\_leaf$ |
|                    | $criterion$          |
|                    | $max\_features$      |
|                    | $max\_depth$         |
| XGBoost classifier | $min\_child\_weight$ |
|                    | $max\_leaf\_nodes$   |
|                    | $gamma$              |
|                    | $N\_estimators$      |
|                    | $max\_depth$         |
| Gradient boosting  | $min\_simple\_split$ |

## C. DATA COLLECTION

The dataset used in this study is obtained from the internal sales records of a leading retail company operating in Turkey. The data are collected and anonymized by the company for internal analytical purposes and subsequently provided to the authors for academic research under confidentiality agreements. Due to commercial sensitivity, the dataset is not publicly available; however, aggregated and anonymized variables are used to ensure privacy and compliance with data protection regulations. In this study, sales data from 284 stores belonging to one of Turkey's leading retail sector leaders was used. The features in the dataset used are explained in Table III.

The store name, year, and month variables are not used in the analysis. The input parameters of the models were created with store-based product type, number of sales, and sales amount variables by adding the warehouse size, region-based per capita income, and population number.

## D. DATA PREPROCESSING

The following data processing steps were applied to prepare the data for ML algorithms.

*Step 1. Grouping Product Types:* Twenty-five different product types sold in stores were combined into six main groups and shown in Table IV. Main groups feature, which is a categorical variable, has been converted to binary variables.

*Step 2. Creating a classification output by combining store indicator and store type:* Indicator refers to the brands sold in the store. Three different brand types can be sold in stores. I brand, T brand, and M brand are available. A minimum of one and a maximum of three brands can be sold in stores. According to the brands sold in the available stores, the indicators are I, T, M, IM, and ITM. Store type is defined as four different types: standard, executive, outlet, and high

**Table III.** Features of data

| Feature name   | Feature type |
|----------------|--------------|
| Store name     | String       |
| Year           | Integer      |
| Month          | Integer      |
| Product type   | Categorical  |
| Sales amount   | Float        |
| Sales quantity | Integer      |
| Indicator      | Categorical  |
| Store type     | Categorical  |

**Table IV.** Grouping product types

| Main groups | Product types  |
|-------------|--|
| Accessories | Leather accessories, jewelry, textile accessories, cosmetics, other accessories, beachwear staff |
| Bottom wear | Skirts, jeans, pants, shorts   |
| Outwear     | Coat, topcoat, other outwear   |
| One piece   | Jacket, dresses, jumpsuit  |
| Top wear    | Blouse, basics, sweatshirt, knitwear   |
| Shoes       | Shoes, boots, slipper, sandals   |

**Table V.** Grouping stores combining two inputs

| Indicator | Store type | Final class |
|-----------|------------|-------------|
| I         | S          | 1           |
| IM        | E          | 2           |
| ITM       | E          | 3           |
| M         | H          | 4           |
| O         | O          | 5           |
| T         | H          | 6           |

level. Not all brands are available in every store type. Therefore, six different store classes were created from the existing combination of indicator and type. This composite variable was used as the classification target. The model output parameter is given in Table V.

*Step 3. Adding the population data to the model on a province basis:* The female population aged between 18 and 50 years, which is the target audience of the company, was included in the data on a province basis. This data is taken from TUIK (Turkish Statistical Institute).

*Step 4. Addition of store warehouse square meters:* Warehouse information was added to the model to examine the effect of store warehouse information on customer purchasing behavior.

*Step 5. Addition of per capita on a province basis:* Gross national product per capita on the basis of provinces using the gross national product information per capita in the provinces where the stores are purchased from TUIK has been added to the model.

In the last case, the inputs in the data model were target audience income, per capita gross national product, and warehouse information of stores, sales numbers, and amounts in five different top product categories. First, the dataset is divided into two in order to analyze the consumer's pre-pandemic and pandemic periods. Data between January 2017 and March 2020 represent the pre-pandemic period, while data for March 2020 and December 2021 represent the pandemic period. Afterward, pre-pandemic and pandemic periods (between January 2017 and December 2021) were also analyzed as a period.

## E. HYPERPARAMETER TUNING WITH ML ALGORITHMS

The following are some steps that must be taken in order to succeed in the study:

*Step 1. Normalization:* There are variations in the units of the generated dataset. Consequently, normalization of absolute maxima is carried out.

*Step 2. Splitting the data:* The normalized dataset is randomly separated as 70% training data and 30% test data. This randomization is expected to improve model performance. The output column is split for both test data and training data.

*Step 3. Base model creation:* A base model was created for each classification algorithm (gradient boosting classifier, XGBoost classifier, and RF) with the processed data.

*Step 4. Tuning with hyperparameters:* Grid search and random search tuning methods, which are among the HPO methods,

were applied for all three classification algorithms. As a result of the tuning process, six different results were obtained.

*Step 5. Evaluation of performance metrics:* Performance metrics of six different models created with two hyperparameter tuning methods for three classification algorithms were compared.

## IV. RESULTS

The segmentation of the data by time period aimed to explore whether the underlying changes in consumer behavior during the pandemic influenced the predictive power of ML models and the relative effectiveness of tuning strategies. Table VI shows the results of pre-pandemic, pandemic, and both periods. The XGBoost classifier gave the highest result with 93% for both grid search and random search tuning methods, while during the pandemic period, the combination of RFs with random trees and random search method gave the highest result with 94%. For the pre-pandemic period, the RF classifier achieved the highest result (0.9767) using random search tuning methods, while during the pandemic period, the combination of RFs with random trees and random search method gave the highest result with 0.9788. For both periods, the RF classifier also achieved the highest accuracy (0.9767) using the random search tuning method. Random search, which is one of the hyperparameter tuning methods, seems to give more successful results in ML algorithms than grid search. Although our findings indicate that HPO methods affect model accuracy, the study by Tabak Kızılgın and Alp [31] reveals that model selection and model architectures (e.g., the strong individual performance of RF and the 90% accuracy of the RF + SVM hybrid) are also equally decisive. This situation indicates that best practice requires not only hyperparameter search but also the simultaneous consideration of an appropriate model/hybrid strategy. The gradient boosting algorithm gave the worst accuracy rates for the pre-pandemic and pandemic periods. While it gave an accuracy score of 0.7588 for the pandemic period, it gave an accuracy rate of

0.6805 for the pandemic period. However, this score increased to 0.7685 when the data of both periods entered the model together.

Assuming that the pre-pandemic and pandemic periods data have a different structure, separate analyses are carried out for the two periods. Then, by combining the data of the two periods, the number of data was increased, the model was established, and the model performances were observed. The increase in the number of data can be stated as the reason for the increase in the success of the models. RF, which gave the best result of the pandemic period, gave an accuracy rate of 0.9891 when data from both periods were combined. Performance metrics are compared across models to evaluate relative effectiveness. It can be concluded that the random search tuning is a successful way to solve the problem.

## V. DISCUSSION

The findings of this study highlight the effectiveness of RF classifiers in multi-class retail store classification tasks. Across all three-time segments, RF consistently yielded the highest accuracy scores, particularly when paired with random search HPO. This suggests that RF not only performs well with limited or imbalanced data but also benefits from flexible tuning strategies. Conversely, gradient boosting demonstrated the weakest performance in most scenarios, which can be attributed to its sensitivity to parameter configurations and susceptibility to overfitting in volatile or smaller datasets such as those during the pandemic. The relatively stable performance of XGBoost further reinforces the importance of model selection under context-specific constraints. The impact of the pandemic on data structure and consumer behavior underscores the need for adaptable ML solutions. The study confirms that HPO plays a crucial role in maximizing model performance under such evolving conditions. In particular, random search appears to outperform grid search in terms of flexibility and computational efficiency. While Kilinc and Rohrhirsch [10] found the accuracy rate to be 90% in their study on bank customers with a two-stage ML approach, the accuracy rate was calculated as 98% in

**Table VI.** Results

| Period       | Classification model | Hyperparameter model | Accuracy      | Recall        | F-score       | Precision     |
|--------------|----------------------|----------------------|---------------|---------------|---------------|---------------|
| Pre-pandemic | Gradient boosting    | Grid search          | 0.7558        | 0.6505        | 0.7459        | 0.6617        |
|              | Gradient boosting    | Random search        | 0.8139        | 0.7787        | 0.8135        | 0.8201        |
|              | XGBClassifier        | Grid search          | 0.8023        | 0.7128        | 0.7508        | 0.7889        |
|              | XGBClassifier        | Random search        | 0.8667        | 0.8556        | 0.8667        | 0.8457        |
|              | Random forest        | Grid search          | 0.8223        | 0.8179        | 0.8022        | 0.7989        |
|              | <b>Random forest</b> | <b>Random search</b> | <b>0.9767</b> | <b>0.9761</b> | <b>0.9760</b> | <b>0.9721</b> |
| Pandemic     | Gradient boosting    | Grid search          | 0.6805        | 0.6612        | 0.6860        | 0.6860        |
|              | Gradient boosting    | Random search        | 0.8139        | 0.8139        | 0.7946        | 0.8183        |
|              | XGBClassifier        | Grid search          | 0.7909        | 0.7909        | 0.7720        | 0.7913        |
|              | XGBClassifier        | Random search        | 0.8403        | 0.8509        | 0.8400        | 0.8602        |
|              | Random forest        | Grid search          | 0.7790        | 0.7790        | 0.7608        | 0.7816        |
|              | <b>Random forest</b> | <b>Random search</b> | <b>0.9766</b> | <b>0.9745</b> | <b>0.9757</b> | <b>0.9718</b> |
| Both period  | Gradient boosting    | Grid search          | 0.7685        | 0.7559        | 0.7558        | 0.7502        |
|              | Gradient boosting    | Random search        | 0.7907        | 0.7907        | 0.7736        | 0.7837        |
|              | XGBClassifier        | Grid search          | 0.8023        | 0.7128        | 0.7938        | 0.8158        |
|              | XGBClassifier        | Random search        | 0.8581        | 0.8681        | 0.8601        | 0.8611        |
|              | Random forest        | Grid search          | 0.9883        | 0.9000        | 0.9311        | 0.9773        |
|              | <b>Random forest</b> | <b>Random search</b> | <b>0.9891</b> | <b>0.9893</b> | <b>0.9895</b> | <b>0.9881</b> |

this study. Rao *et al.* [27] investigated the hyperparameter-tuned studies for cough detection in the COVID-19 period for a deep learning model and reported an AUROC of 82.23%. Hamdi *et al.* [28] reported an accuracy score of 83% by testing COVID-19 infections from chest X-rays with HPO in multi-classification algorithms. Adedigba *et al.* [29] used the optimal hyperparameter selection method of deep learning models for COVID-19 chest X-ray classification and reported a validation accuracy score of 96.83%. Kalliola *et al.* [30] reported an Root Mean Square Error (RMSE) value of 2.5 % in their study on neural network HPO for the prediction of property prices in Helsinki.

## VI. CONCLUSION

This study aimed to compare the effectiveness of two widely used HPO techniques (grid search and random search) on the classification performance of three ML algorithms (RF, gradient boosting, and XGBoost). Using real-world data from the retail sector, including sales volume, demographic features, and macroeconomic indicators, the models were trained to classify stores into six distinct categories. The analyses were conducted over three different temporal periods: the pre-pandemic period (January 2017–March 2020), the pandemic period (March 2020–December 2021), and a combined dataset including both. The results indicated that RF, when optimized with random search, consistently outperformed the other models across all scenarios, achieving the highest accuracy rate of 98.9% in the combined period. XGBoost also yielded strong results, whereas gradient boosting showed comparatively lower performance. The consistent superiority of random search over grid search in this context highlights the importance of flexible and exploratory optimization techniques, especially in cases where parameter search spaces were large and complex. Furthermore, the analysis confirmed that hyperparameter tuning significantly influences classification performance, particularly in dynamic conditions such as those introduced by the COVID-19 pandemic. Overall, this study provided a practical framework for applying ML and HPO in multi-class classification problems within the retail industry. Future research could investigate whether predictive performance can be further improved by incorporating additional features into the model. Potential variables include customer-related economic, financial, or demographic data, as well as environmental factors such as store infrastructure, surrounding parking facilities, or in-store traffic. Moreover, more advanced optimization methods—such as Bayesian optimization or evolutionary algorithms—could be explored to enhance both model accuracy and computational efficiency.

## CONFLICT OF INTEREST STATEMENT

The author(s) declare that they have no conflicts of interest to report regarding the present study.

## REFERENCES

- [1] J. A. Nichols, H. W. H. Chan, and M. A. A. Baker, "Machine learning: Applications of artificial intelligence to imaging and diagnosis," *Biophys. Rev.*, vol. 11, no. 1, pp. 111–118, 2018.
- [2] F. Y. Osisanwo *et al.*, "Supervised machine learning algorithms: Classification and comparison," *Int. J. Comput. Trends Technol.*, vol. 48, no. 3, pp. 128–138, 2017.
- [3] J. J. Jui *et al.*, "Optimal energy management strategies for hybrid electric vehicles: A recent survey of machine learning approaches," *J. Eng. Res.*, vol. 12, no. 3, pp. 454–467, 2024.
- [4] F. Kocoglu and T. Ozcan, "A grid search optimized extreme learning machine approach for customer churn prediction," *J. Eng. Res.*, vol. 11, no. 3, pp. 103–112, 2023.
- [5] S. Hanifi, A. Cammarono, and H. Zare-Behtash, "Advanced hyperparameter optimization of deep learning models for wind power prediction," *Renew. Energy*, vol. 221, no. 15, pp. 1–12, 2023.
- [6] P. Lalwani *et al.*, "Customer churn prediction system: A machine learning approach," *Computing*, vol. 104, no. 2, pp. 271–294, 2022.
- [7] A. K. Ahmad, A. Jafar, and K. Aljoumaa, "Customer churn prediction in telecom using machine learning in big data platform," *J. Big Data*, vol. 6, no. 1, pp. 1–24, 2019.
- [8] K. G. Li and B. P. Marikannan, "Hyperparameters tuning and model comparison for telecommunication customer churn predictive models," In *3rd Global Conference on Computing & Media Technology* (pp. 475–483), July 2020.
- [9] T. T. Win and K. S. Bo, "Predicting customer class using customer lifetime value with Random Forest Algorithm," In *2020 International Conference on Advanced Information Technologies* (pp. 236–241), Nov. 2020.
- [10] M. S. Kilinc and R. Rohrhirsch, "Predicting customers' cross-buying decisions: A two-stage machine learning approach," *J. Bus. Anal.*, vol. 6, no. 3, pp. 180–187, 2023.
- [11] D. M. Belete and M. D. Huchaiah, "Grid search inhyperparameter optimization of machine learning models for prediction of HIV/AIDS test results," *Int. J. Comput. Appl.*, vol. 44, no. 9, pp. 875–886, 2022.
- [12] C. A. Bentejac, A. Csörgö, and G. Martinez-Munoz, "A comparative analysis of gradient boosting algorithms," *Artif. Intell. Rev.*, vol. 54, no. 1, pp. 1937–1967, 2021.
- [13] E. Elgeldawi *et al.*, "Hyperparameter Tuning for Machine Learning Algorithms used for arabic sentiment analysis," *Informatics*, vol. 8, no. 4, pp. 1–21, 2021.
- [14] A. Pfob, S. Lu, and C. Sidey-Gibbons, "Machine learning in medicine: A practical introduction to techniques for data pre-processing, hyperparameter tuning, and model comparison," *BMC Med. Res. Methodol.*, vol. 22, no. 1, p. 15, 2022.
- [15] R. Valarmathi and T. Sheela, "Heart disease prediction using hyper parameter optimization (HPO) tuning," *Biomed. Signal Process. Control*, vol. 70, no. 1, p. 10, 2021.
- [16] S. Kaur, H. Aggarwal, and R. Rani, "Hyper-parameter optimization of deep learning model for prediction of Parkinson's disease," *Mach. Vis. Appl.*, vol. 31, no. 1, pp. 1–15, 2020.
- [17] M. M. Ramadhan *et al.*, "Parameter tuning in random forest based on grid search method for gender classification based on voice frequency," in *2017 International Conference on Computer Science and Engineering* (pp. 625–629), 2017.
- [18] P. Probst, M. N. Wright, and A. L. Boulesteix, "Hyperparameters and tuning strategies for random forest," *WIREs Data Min. Knowl. Discov.*, vol. 9, no. 3, pp. 1–15, 2019.
- [19] Y. Freund and R. E. Schapire, "Experiments with a New Boosting Algorithm. International Conference on Machine Learning," *Journal Title* (pp. 148–156), July 1996.
- [20] O. González-Recio, J. A. Jiménez-Montero, and R. Alenda, "The gradient boosting algorithm and random boosting for genome-assisted evaluation in large data sets," *J. Dairy Sci.*, vol. 96, no. 1, pp. 614–624, 2013.
- [21] S. Demir and E. K. Şahin, "An investigation of feature selection methods for soil liquefaction prediction based on tree-based ensemble algorithms using AdaBoost, gradient boosting, and XGBoost," *Neural Comput. Appl.*, vol. 35, no. 4, pp. 3173–3190, 2023.

- [22] S. Khan *et al.*, “XGBoost-enhanced ensemble model using discriminative hybrid features for the prediction of sumoylation sites,” *BioData Min.*, vol. 18, no. 12, pp. 1–18, 2025.
- [23] I. Yang and A. Shami, “On hyperparameter optimization of machine learning algorithms: Theory and practice,” *Neurocomputing*, vol. 415, no. 1, pp. 295–306, 2020.
- [24] N. Zhu *et al.*, “Optimization of the random forest hyperparameters for power industrial control systems intrusion detection using an improved Grid Search Algorithm,” *Appl. Sci.*, vol. 12, no. 20, pp. 1–16, 2022.
- [25] H. N. Fakhouri *et al.*, “Novel hybrid success history intelligent optimizer with gaussian transformation: Application in CNN hyperparameter tuning,” *Cluster Comput.*, vol. 27, no. 3, pp. 3717–3739, 2024.
- [26] N. Subaşı, “Comprehensive analysis of grid and randomized search on dataset performance,” *Eur. J. Eng. Appl. Sci.*, vol. 7, no. 2, pp. 77–83, 2024.
- [27] S. Rao *et al.*, “Deep Learning with hyper-parameter tuning for COVID-19 Cough Detection,” in *2021 12th International Conference on Information, Intelligence, Systems & Applications* (pp. 1–5), 2021.
- [28] I. Hamdi, M. Ridzuan, and M. Yaqub, “Hyperparameter optimization for COVID-19 chest X-Ray classification,” *arXiv preprint arXiv:2201.10885*, 2022.
- [29] A. P. Adedigba *et al.*, “Optimal hyperparameter selection of deep learning models for COVID-19 chest X-ray classification,” *Intell. Based Med.*, vol. 5, no. 1, pp. 1–12, 2021.
- [30] J. Kalliola, J. Kapočiūtė-Dzikienė, and R. Damaševičius, “Neural network hyperparameter optimization for prediction of real estate prices in Helsinki,” *PeerJ Comput. Sci.*, vol. 7, p. e444, 2021.
- [31] K. Tabak Kızgın and S. Alp, “Studying the impact of changing consumer behavior during crisis periods through store classification,” *Int. J. Comput. Intell. Syst.*, vol. 17, no. 288, pp. 1–16, 2024.