Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.2, April 2026, pp. 1125~1135
e-ISSN: 2548-6861

A Comparative Study of Hyperparameter Optimization for CatBoost: Random Search, Optuna, and Successive Halving

Claudian Tikulimbong Tangdilomban [1]*, Kusman Sadik [2]*, Aji Hamim Wigena [3]*
* Statistics and Data Science, Department of Statistics, IPB University, Bogor, Indonesia
claudian_tangdilomban@apps.ipb.ac.id [1], kusmans@apps.ipb.ac.id [2], aji_hw@apps.ipb.ac.id [3]

Article Info

Article history:

Received 2025-11-22
Revised 2026-02-16
Accepted 2026-02-27

Keyword:

CatBoost,
Hyperparameter optimization,
Successive Halving,
Optuna,
Random Search.

ABSTRACT

This study aims to evaluate the effectiveness of three hyperparameter optimization approaches Random Search, Successive Halving, and Optuna in the CatBoost algorithm for modeling individual income using the 2024 SAKERNAS data. Model performance was assessed using RMSE, MAE, and R-squared, complemented by a significance test based on 10,000 bootstrap resamples to ensure that performance differences were not driven by random variation. The results indicate that Optuna yields the most accurate predictive performance, followed by Successive Halving and Random Search. The RMSE values, which range from several hundred thousand to approximately one million rupiah, are consistent with the characteristics of the income variable, which is measured in rupiah and exhibits a heavy-tailed distribution. The feature importance analysis reveals a generally consistent ranking structure across methods, although moderate variation is observed for several features. These findings confirm that Optuna is the most effective tuning strategy, while Successive Halving serves as an efficient alternative for large-scale datasets. Overall, this study highlights the critical role of optimization strategies in improving predictive performance without compromising interpretability stability, making it particularly relevant for analytical applications in micro-level socio-economic data.

I. INTRODUCTION

Model performance in machine learning is highly dependent on hyperparameter configuration. Hyperparameters govern learning behavior, model capacity, and convergence speed; therefore, their selection must be carried out carefully [1]. The increasing complexity of modern methods particularly deep learning and gradient boosting further necessitates the use of automated, efficient, and adaptive hyperparameter optimization frameworks capable of handling complex data structures [2]. Hyperparameter optimization has become increasingly critical given that the search space is high-dimensional and non-linear, making manual tuning infeasible in terms of both time and computational resources. CatBoost is one of the most competitive algorithms for modeling large-scale and heterogeneous data. Surveys on CatBoost indicate that it is a strong candidate for Big Data scenarios due to its ability to handle categorical features, structured information, and its ordered boosting mechanism that mitigates target leakage[3], [4]. CatBoost is built upon gradient boosting via functional gradient descent, where a strong model is constructed by greedily aggregating base learners iteratively. This theoretical foundation allows CatBoost to achieve greater stability than conventional boosting techniques, particularly when applied to mixed-type data.

Although CatBoost is known for its high performance, the quality of the model remains highly dependent on the hyperparameter search process. Various approaches have been proposed to optimize CatBoost configurations, ranging from Random Search simple yet effective in exploring the hyperparameter space randomly to Successive Halving, which progressively eliminates weaker candidates, and Optuna, which provides adaptive Bayesian optimization. Hartono (2025) finds that Random Search can yield competitive performance in CatBoost Regression and that the model is sensitive to the train–test split ratio [5]. Meanwhile, study [6] demonstrates that Optuna can significantly improve

CatBoost performance compared to default configurations, including raising medical prediction accuracy from 65% to 72%. Other studies even report that Optuna-optimized ensemble learning models can achieve accuracies above 90% across various application domains. These findings suggest that hyperparameter tuning is not merely a supplementary procedure, but a key component determining the quality of machine learning models.

However, studies directly comparing the effectiveness of hyperparameter optimization methods specifically Random Search, Successive Halving, and Optuna on CatBoost remain limited. This gap is particularly pronounced in socio-economic datasets, which tend to be skewed and heavy-tailed. More specifically, there is no consensus regarding which optimization method is most suitable for microeconomic data that are not only skewed but also characterized by complex missing-value patterns and heteroskedasticity. Economic and social variables such as household income and expenditure typically exhibit skewed and heavy-tailed distributions [7], which can disrupt the stability of both statistical and machine learning models. Previous research indicates that models such as Vector Autoregression (VAR) may exhibit instability when confronted with distributions far from normality [8]. In the context of individual income, [9] highlights that the distributional characteristics of the target variable are often not adequately analyzed alongside covariate effects.

This study empirically examines how different hyperparameter optimization strategies perform when applied to the structural characteristics of Indonesian labor-force microdata. The SAKERNAS dataset is characterized by heavy-tailed and zero-inflated income distributions, dominance of categorical variables, and substantial socio-economic heterogeneity. In contrast to prior studies that rely primarily on benchmark or relatively balanced datasets, this analysis provides methodological insights into income modeling in a developing-country context, with direct relevance to micro-level socio-economic analysis and public policy. Therefore, this study comparatively evaluates three hyperparameter optimization methods Random Search, Optuna, and Successive Halving applied to CatBoost using SAKERNAS data, which features a skewed target distribution. The objective is to identify the most effective and efficient optimization approach. The evaluation encompasses not only predictive accuracy but also computational efficiency and interpretability stability, all of which are important for data-driven decision-making. To enhance model stability, this study additionally applies a log transformation to the target variable.

Accordingly, this research aims to provide a comprehensive assessment of how different hyperparameter optimization methods affect CatBoost performance and to offer practical recommendations for modeling socio-economic data with complex distributional characteristics. The findings are expected to guide researchers and practitioners in selecting appropriate optimization strategies for modeling Indonesian labor-force data, while also contributing to the machine learning literature on developing-country data with distinct characteristics.

This study provides an original contribution through its specific focus on Indonesian microeconomic data (SAKERNAS), which possesses unique and methodologically challenging characteristics: heavy-tailed income distribution, zero-inflation, and a predominance of categorical variables. Unlike previous studies that generally use relatively balanced benchmark datasets, this analysis tests the robustness of various hyperparameter optimization methods under real-world socioeconomic conditions in a developing country. The methodological implications are significant because these findings can serve as a guideline for income modeling in contexts of high inequality and structural heterogeneity, which are relevant not only to machine learning researchers but also to economists and public policy makers.

II. METHODOLOGY

Data preprocessing was conducted through winsorization of the upper percentiles of household income and a logarithmic transformation to stabilize variance. Winsorization was applied at the 99th percentile to cap extreme upper-tail values while preserving rank order. Logarithmic transformation (log1p) was then applied to positive income values to stabilize variance and reduce heteroskedasticity. The transformed variable was used for modeling, with predictions back-transformed (expm1) for evaluation in original rupiah scale. The dataset was subsequently partitioned into training and testing subsets using an 80:20 hold-out scheme. To ensure a fair comparison, all optimization methods—Random Search, Successive Halving, and Optuna—were executed using an identical hyperparameter search space. Model evaluation was performed using a 5-fold cross-validation scheme for each method, including Successive Halving, which internally applies cross-validation for every parameter candidate.

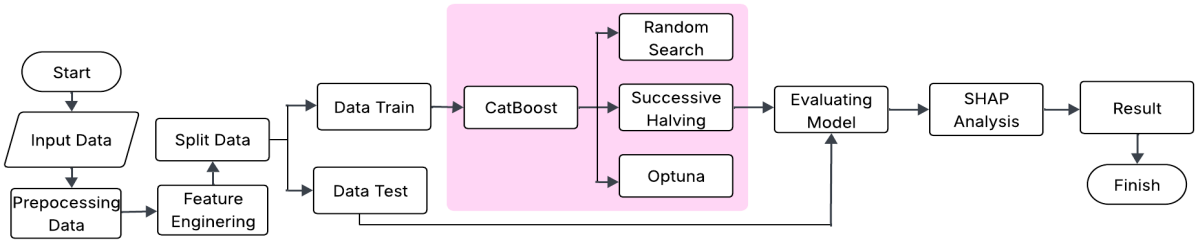

Figure 1. Analysis Procedures

A. Data

This study employs data from the 2024 National Labor Force Survey (SAKERNAS), published by Badan Pusat Statistik (BPS). The dataset analyzed is a subset for West Java Province, consisting of 49,074 observations and 20 variables. The structure of the dataset is dominated by categorical variables capturing demographic information, employment status, job characteristics, and industry sectors. After data extraction and filtering, the dataset contained no missing values across the selected variables. A detailed description of the variables is provided in Table 1.

TABLE I
RESEARCH VARIABLES

Variabel	Description
R16	Net monthly income/wages/salary received from the main job during the past month
K4	Gender
K10	Age
KODE_KAB	District code
R12A	Main reason for temporarily not working during the past week
R14B	Assisted by non-permanent workers and paid
R14D	Business registered in the licensing system
R15B_KBJI2	Indonesian Standard Classification of Occupations (KBJI 2014)
R18A_BLN	Month started working in the current job or business activity
R18A_THN	Year started working in the current job or business activity
R19A_BLT	Rounded number of hours worked in the past week excluding break time
R19A_JML	Total hours worked in the past week excluding break time
R19B	Usual weekly working hours excluding break time
R20B8	Use of internet for internet banking in the main job during the past month
R24C	Net initial monthly wage/salary (cash and in-kind) when starting the main job
R25B	Work accident insurance from the institution/company/business
R25D	Old-age security benefits provided by the institution/company/business
R25I	Wage compliance with the provincial minimum wage (UMP) provided by the institution/company/business

In this study, the dataset is utilized as tabular data to evaluate the performance of three hyperparameter optimization methods for the CatBoost algorithm, namely Random Search, Optuna, and Successive Halving. The SAKERNAS data contain numerous categorical features, which is one of the reasons for selecting CatBoost, as the algorithm efficiently handles categorical variables without requiring additional encoding procedures. The complex yet standardized structure of the dataset enables objective model evaluation within a relevant prediction scenario.

B. CatBoost

CatBoost is a Gradient Boosting Decision Tree (GBDT) algorithm developed by Yandex, designed to handle categorical features without requiring preprocessing steps such as one-hot encoding or label encoding [10]. The strength of CatBoost lies in two core mechanisms—Target Statistics (TS) and Ordered Boosting—which effectively reduce overfitting and target leakage in tabular data [11].

Each category is represented by the mean value of the target within that category. Let the training dataset be defined as:

$$D = \{(X_i, Y_i)\}_{i=1}^{n},$$

and let $x_{(i,k)}$ denote the categorical value of the k-th feature for observation i. This categorical value is transformed into:

$$x_{i,k} = \frac{\sum_{j=1}^{n}(x_{j,k} = x_{i,k}) \cdot y_j}{\sum_{j=1}^{n}(x_{j,k} = x_{i,k})}$$

However, this approach is prone to target leakage because the label Y_i may influence the representation of its own category. CatBoost addresses this issue through Ordered Target Statistics, in which categorical statistics are computed based on a random permutation such that each observation uses only information from the samples that appear before it in the permutation.

For a permutation $\sigma = (\sigma_1, \ldots, \sigma_n)$, the Ordered TS is defined as:

$$x_{(\sigma_p,k)} = \frac{\sum_{j=1}^{p-1}(x_{\sigma_j,k} = x_{\sigma_p,k}) \cdot y_{\sigma_j} + aP}{\sum_{j=1}^{p-1}(x_{\sigma_j,k} = x_{\sigma_p,k}) + a}$$

where P denotes the prior representing the global mean of the target, and $a > 0$ is a smoothing parameter [4]. This formulation ensures that the label of an observation is never used to encode its own categorical value, thereby preventing information leakage. CatBoost further employs multiple random permutations at each boosting iteration to reduce variance and improve the stability of the estimates [13].

CatBoost was deliberately selected as the sole predictive model to isolate the effect of hyperparameter optimization strategies under a fixed algorithmic structure. Including multiple algorithms (e.g., XGBoost, LightGBM, or econometric regression models) would introduce additional sources of variation—such as differences in feature encoding schemes, learning mechanisms, and inductive biases—that could obscure the comparative evaluation of optimization methods. Accordingly, this study emphasizes internal methodological validity rather than cross-algorithm generalization.

C. Random Search

Random Search is one of the fundamental approaches to hyperparameter optimization, operating by randomly selecting parameter configurations from the defined search space. This method is effective because it does not rely on a structured grid and is capable of exploring the parameter

space more broadly through a trial-and-error mechanism [14]. Additional advantages include its simplicity of implementation, ease of parallelization, and flexibility in dynamically increasing or reducing the number of trials on the fly [15].

The Random Search process begins by identifying the tunable parameters of a model and specifying the range of values to be explored. Subsequently, combinations of parameter values are sampled at random, and the model is trained using the selected combinations. The model's performance is then evaluated using predetermined evaluation metrics. This evaluation serves as the basis for continuing the exploration by iteratively testing new combinations until the best-performing configuration is identified [16], [17].

In the final stage, the parameter configuration with the highest performance is selected as the final model. The complete workflow of the Random Search procedure is illustrated in Figure 2 [18].

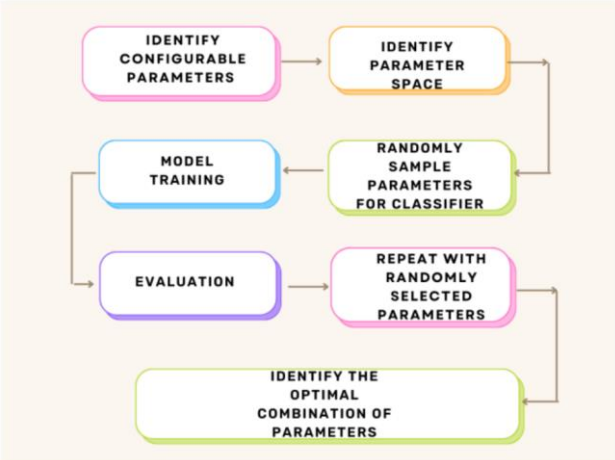

Figure 2. Step-by-step process of random search execution

D. Optuna

Optuna is a next-generation hyperparameter optimization framework based on Bayesian optimization, designed to provide flexibility, efficiency, and ease of use in model parameter search [2]. The framework adopts a define-by-run approach, which allows the hyperparameter search space to be constructed dynamically as the objective function is executed. This design makes Optuna more adaptive and capable of handling complex parameter-space structures compared to the define-and-run approach used in frameworks such as Hyperopt.

In practice, Optuna requires the specification of an objective function, a hyperparameter search space, and a validation scheme. The optimization process proceeds through a series of trials, where each trial evaluates one independent combination of hyperparameters. The model's performance is assessed using predefined metrics, and Optuna automatically selects the best configuration based on the objective value. This approach has been shown to accelerate the optimization process while reducing the need for manual parameter tuning [19].

During optimization, Optuna defines the entire search process as a *study*, while each objective function evaluation is treated as a *trial*. To improve search efficiency, Optuna leverages the Tree-structured Parzen Estimator (TPE), a Bayesian optimization algorithm that updates its probabilistic distributions based on previous trial outcomes. Through this mechanism, Optuna has demonstrated superior performance compared to many other black-box optimization frameworks, offering advantages in terms of efficiency, flexibility, and ease of integration across diverse machine learning models [2].

E. Successive Halving

Successive Halving is classified as an early-stopping-based approach to hyperparameter optimization. In the initial iteration, each candidate configuration is allocated a small amount of computational resources (e.g., a limited number of iterations or a restricted data subset). Candidates with the poorest performance are gradually eliminated. After each evaluation, only the best-performing models are retained and retrained with a larger resource allocation in the subsequent stage. This process continues iteratively until a single best hyperparameter configuration remains. The method has become increasingly popular due to its competitive performance and its integration into widely used Python machine learning libraries, particularly scikit-learn, through the implementations *HalvingGridSearchCV* and *HalvingRandomSearchCV* [20].

Historically, the Successive Halving algorithm was first introduced in the context of identifying the best arm in stochastic bandit problems under a fixed budget setting by Karnin et al. (2013) [21]. Subsequent studies have demonstrated that the algorithm is also effective in non-stochastic environments. The basic principle of Successive Halving is straightforward: given a fixed computational budget, resources are evenly allocated across all arms; their performance is evaluated; the bottom half of the arms is eliminated; and the process is repeated until only the best-performing arm remains [22]. The iterative elimination steps and reallocation of computational resources in Successive Halving can be visualized in Figure 3 [23].

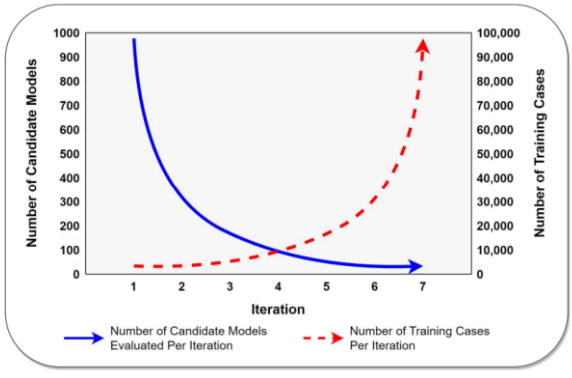

Figure 3. A graphical representation of the successive halving algorithm

In this study, Successive Halving was implemented using the same discrete hyperparameter space as Random Search and adopted a candidate-elimination strategy rather than progressive budget allocation. At each halving stage, all candidate configurations were fully trained using their respective hyperparameter settings, while poorly performing candidates were discarded based on validation performance. The computational efficiency of the method arises from progressively reducing the number of candidate configurations evaluated in later stages, rather than from partial model training.

The number of boosting iterations (n_estimators) was treated as a tunable hyperparameter with discrete values {300, 500, 800}, consistent with Random Search. An elimination factor of 2 was applied, such that approximately half of the candidate configurations were removed at each halving stage. This design ensures a fair comparison across optimization methods while adhering to the standard implementation of *HalvingRandomSearchCV*.

F. Hyperparameter Optimization

To ensure a fair comparison across optimization methods, all approaches were applied to the same set of tuned hyperparameters with aligned conceptual ranges.

TABLE II
HYPERPARAMETER SEARCH SPACE

Hyperparameter	Random Search	Successive Halving	Optuna
depth	{4,6,8}	{4,6,8}	{4,6,8}
learning_rate	{0.02, 0.03, 0.05}	{0.02, 0.03, 0.05}	[0.025, 0.05]
n_estimators	{300, 500, 800}	{300, 500, 800}	[400, 800]
l2_leaf_reg	{5, 10, 20}	{5, 10, 20}	{1, 3, 5, 10, 20}
subsample	{0.6, 0.8}	{0.6, 0.8}	[0.7, 1.0]

In Optuna, continuous and integer-valued hyperparameters were sampled from predefined bounded domains, with the learning rate optimized on a log-scale over the interval [0.025, 0.05], the number of estimators treated as an integer variable within [400, 800], and the subsample ratio sampled continuously from [0.7, 1.0]. Random Search and Successive Halving were implemented over a discrete hyperparameter space to reflect standard practical usage, whereas Optuna employs continuous probabilistic sampling through a TPE-based Bayesian optimization framework. Despite these differences, the tuned parameters and their ranges were aligned to ensure that performance differences arise from optimization strategies rather than search space design.

III. RESULT AND DISCUSSION

A. Data Exploration

The individual income variable in the 2024 SAKERNAS dataset is analyzed to understand the fundamental characteristics of the target to be predicted. Examining the distribution is essential for identifying initial patterns, detecting structural imbalances, and determining the need for data transformation prior to the modeling stage.

Figure 4 presents the overall income distribution, including respondents with zero income. Many observations are concentrated at very low-income levels, with a large group of respondents falling within the range of zero to a few hundred thousand rupiah. This pattern produces a highly right-skewed distribution, characterized by a long right tail driven by a small number of respondents with very high income. Such characteristics are common in Indonesian labor-force data, which are dominated by informal workers and unpaid family workers.

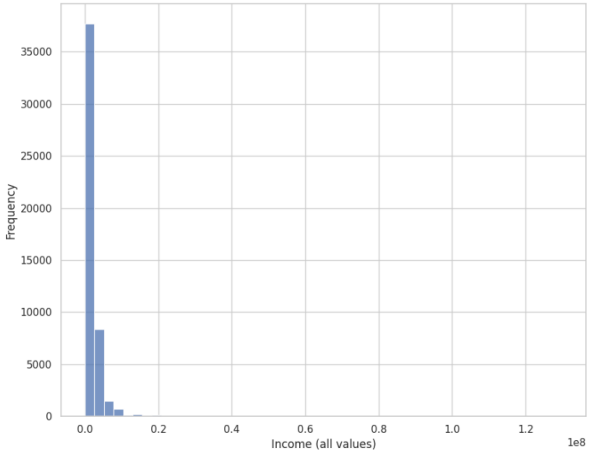

Figure 4. Distribution of SAKERNAS 2024 Income Data

To obtain a clearer picture of the income composition, Figure 5 presents the proportion of respondents with zero income and positive income. The diagram shows that 43.4% of respondents reported no income during the survey period, while 56.6% had positive income. The relatively large share of zero-income respondents highlights the need for special consideration in the analysis, as the presence of this group substantially affects both the shape of the distribution and the variability structure of the income data.

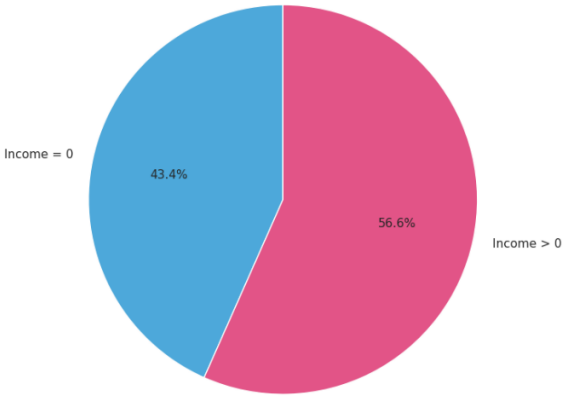

Figure 5. Distribution of Income Levels

In the original scale, the extremely wide range of income values results in a disproportionate distribution, where observations with high income dominate the overall structure and obscure patterns among respondents in the low- to middle-income range. To obtain a more informative representation, the positive income values were subsequently transformed into a logarithmic scale.

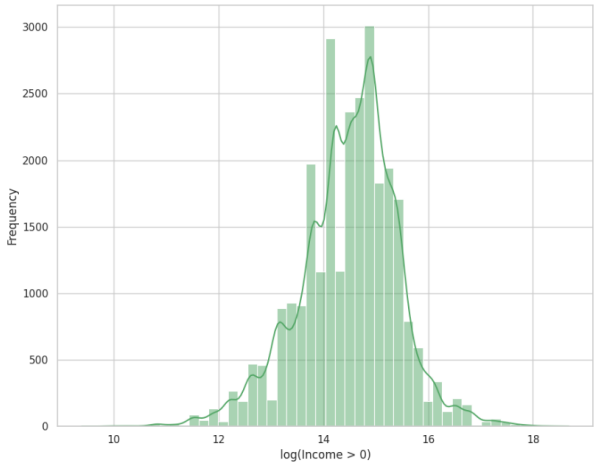

Figure 6. Distribution of log-Transformed Income Data

Figure 6 shows that applying a logarithmic transformation to positive income values produces a more stable and structured distribution while reducing the heteroskedasticity present in the original scale. In highly skewed data such as SAKERNAS income, differences between respondents may span several orders of magnitude, resulting in non-homogeneous residual variance. By mapping income values to a logarithmic scale, the range of variability becomes more compressed, and the overall data spread appears more proportional. The log transformation helps stabilize variance, particularly in the upper-income range, which previously dominated the scale of the distribution.

This condition facilitates clearer identification of relationships between predictors and the target variable, as variation patterns within the middle-income range—the majority of the population—are no longer overshadowed by a small set of extreme values. The stabilizing effect therefore provides strong methodological justification for employing a log transformation in exploratory analysis of income distributions, especially in microdata such as SAKERNAS, which exhibit severe inequality and wide value ranges.

Winsorizing the log-transformed income is implemented as an additional step to minimize the influence of outliers that persist even after transformation. Although boosting algorithms such as CatBoost are relatively robust to extreme values, observations lying far beyond the typical population range may still affect the stability of evaluation and distributional interpretation. Winsorization constrains the influence of these values without altering the fundamental structure of the data, resulting in a more controlled distribution that is no longer dominated by a few extreme observations and is therefore more representative for modeling. This approach strengthens the quality of the descriptive analysis and supports the reliability of subsequent modeling procedures.

B. Model Performance Evaluation

Model performance was evaluated using three primary metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R-squared). These metrics were selected because they provide a comprehensive assessment of predictive accuracy and model stability within the context of socio-economic data. RMSE is sensitive to large errors, MAE is more robust to outliers, and R-squared indicates the proportion of variation in individual income that can be explained by the model. Table 2 presents the performance comparison of the CatBoost model using the three hyperparameter optimization methods—Random Search, Optuna, and Successive Halving. In general, lower RMSE and MAE values combined with higher R-squared values indicate better predictive performance.

TABLE III
MODEL PERFORMANCE EVALUATION METRICS

Method	RMSE	MAE	R^2	Time (second)
Random Search	1,326,146	563,568.39	0.6997	1394
Successive Halving	1,346,790	578,837.90	0.6903	1245
Optuna	1,295,337	549,964.35	0.7135	7057

Table 3 presents the evaluation metrics obtained from the three hyperparameter optimization methods applied to the CatBoost model. Among the three methods evaluated, Optuna demonstrates the strongest performance, achieving an RMSE of 1,295,337, an MAE of 549,964, and an R-squared of 0.7136. These results indicate that the Bayesian Optimization strategy employed by Optuna is capable of identifying hyperparameter configurations that more effectively capture income variability in the SAKERNAS dataset. Random Search also shows performance gains, with an RMSE of 1,326,146 and an MAE of 563,568, although the improvement is less substantial than that achieved by Optuna. In contrast, Successive Halving produces the highest RMSE (1,346,790) and an MAE of 578,837, suggesting that this method is less effective at reducing large prediction errors, although it remains relatively competitive in minimizing average absolute error.

RMSE values in the range of one to two million rupiah should not be interpreted as indicative of poor model performance. Rather, they reflect the characteristics of the income variable in SAKERNAS, which exhibits an extremely wide range and a heavy-tailed distribution. In microdata such as SAKERNAS, individual income may range from zero to more than one hundred million rupiah per month. Therefore, predictive deviations on the order of one million rupiah remain reasonable and meaningful within an economic context, and the interpretation of error metrics must consider

both the rupiah scale and the structural inequality inherent in income distributions.

Differences in performance across optimization methods can be explained by the underlying mechanisms of their hyperparameter search strategies. These findings align with the principle of Successive Halving, which eliminates candidate configurations at early stages. In complex and noisy datasets such as SAKERNAS, hyperparameter configurations requiring more iterations to converge may be prematurely discarded, resulting in a suboptimal final model. While efficient for large-scale datasets, this strategy carries the risk of eliminating configurations that achieve optimal performance only after extended training. In contrast, Random Search explores the hyperparameter space randomly without leveraging information from previous evaluations, leading to more moderate improvements. Optuna, through its Bayesian Optimization approach, exploits information from past trials to guide the search toward promising regions of the parameter space, thereby producing the most stable and accurate model.

Overall, these findings demonstrate that the choice of hyperparameter optimization method has substantive implications for CatBoost performance on large-scale socio-economic data. Optuna emerges as the most effective method in balancing accuracy and efficiency, while Successive Halving offers a fast alternative with distinct error characteristics. These insights are highly relevant for empirical research and data-driven policymaking, particularly in contexts that require accurate prediction while maintaining model interpretability and robustness.

C. Segment-wise Performance Across Income

Aggregate evaluation metrics provide an overall assessment of model performance but may conceal differences in predictive accuracy across income groups. Therefore, a segment-wise evaluation is conducted by dividing individuals into income quartiles. Log-transformed income is first used to enable relative comparison across income segments by mitigating scale effects arising from the heavy-tailed income distribution.

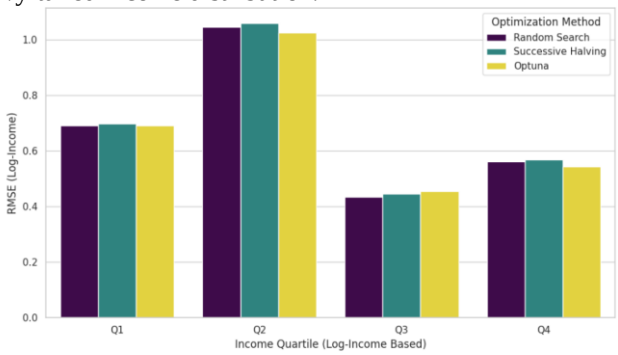

Figure 7. Segment-wise RMSE on Log-Income Scale

Figure 7 presents segment-wise RMSE on the log-income scale. Prediction errors are relatively similar in the lowest-income group (Q1), with RMSE values ranging from 0.691 to 0.697 across optimization methods. In contrast, substantially higher errors are observed in the second quartile (Q2), where RMSE increases to 1.026–1.060, indicating greater heterogeneity in income dynamics around the middle of the distribution. The lowest relative errors occur in the third quartile (Q3), while RMSE increases again in the highest-income group (Q4), reaching 0.543–0.568. Across most quartiles, Optuna achieves the lowest or near-lowest RMSE, particularly in Q2 and Q4.

While log-scale RMSE facilitates fair comparison across income segments, it does not convey the economic magnitude of prediction errors. For this reason, segment-wise RMSE is also evaluated on the original income scale (rupiah) using real-income–based quartiles.

TABLE IV
SEGMENT-WISE RMSE ON THE REAL-INCOME SCALE (RUPIAH)

Method	Q1	Q2	Q3	Q4
Random Search	75,405	774,024	694,860	2,608,728
Successive Halving	77,154	784,539	717,180	2,646,025
Optuna	82,155	772,049	710,552	2,531,451

Table 4 reports prediction errors in rupiah across income groups. In the lowest-income quartile (Q1), RMSE values range from approximately 75–82 thousand rupiah, whereas errors increase substantially in higher-income segments. RMSE values in the middle-income groups (Q2–Q3) reach approximately 690–785 thousand rupiah, and exceed 2.5 million rupiah in the highest-income quartile (Q4), reflecting the heavy-tailed nature of the income distribution. Across all income groups, Optuna consistently yields lower RMSE values, reducing prediction error in Q4 by approximately 100–130 thousand rupiah compared to the other optimization methods.

From a policy-oriented perspective, this segment-wise evaluation demonstrates that model performance is not disproportionately driven by specific income groups, supporting the robustness of the proposed approach for socio-economic analysis.

D. Significance Test

The evaluation of performance differences across tuning methods was conducted using a bootstrap resampling significance test to ensure that variations in model performance were not merely the result of sample fluctuations but reflected systematic differences. This test employed 10,000 bootstrap replications, allowing the distribution of RMSE differences between models to be estimated with greater precision and without reliance on distributional assumptions. The null hypothesis assumes that there is no difference in predictive performance between two hyperparameter optimization methods, while the alternative hypothesis assumes that one method yields lower prediction error. Statistical significance was assessed at the 5% level. In addition to empirical p-values, 95% confidence intervals of

RMSE differences were computed directly from the bootstrap distribution. A performance difference was considered statistically significant when the confidence interval excluded zero. In this study, the p-value criterion was adopted as the primary basis for determining statistical significance, which is sufficient for nonparametric bootstrap comparison. The results of the test are presented in Table 3.

TABLE V
BOOTSTRAP SIGNIFICANCE TEST

Comparison	p-value	Effect Mean	CI 95%	Std Dev
Random Search vs Halving Successive	0.99995	−20,553.74	[-28,294 ; -12,233]	4,091.13
Random Search vs Optuna	0.00005	30,822.77	[22,558 ; 39,379]	4,279.33
Halving Successive vs Optuna	0.00005	51,499.89	[39,950 ; 63,264]	6,056.63

The bootstrap results presented in Table 5 show a consistent pattern in distinguishing the performance of the three tuning methods. For the comparison between Random Search and Successive Halving, the p-value of 0.99995 is far above the significance threshold of $\alpha = 0.05$. This indicates that the RMSE difference between the two methods is not statistically significant. The effect size of $-20{,}553$ is also relatively small, suggesting insufficient evidence to conclude the presence of a stable performance difference. The effect size is defined as the mean RMSE difference between models across 10,000 bootstrap replications. Thus, the two methods yield comparable predictive performance in the context of SAKERNAS income data.

In contrast, the comparison between Random Search and Optuna yields a markedly different result. The p-value of $0.00005 < \alpha = 0.05$ indicates that the performance improvement of Optuna is statistically significant. The effect size of 30,823 confirms that this improvement is not attributable to random fluctuations but represents a consistent pattern across bootstrap replications. Although the magnitude of the effect is moderate, the stability of the improvement provides strong evidence that Optuna produces more efficient hyperparameter configurations for capturing the complex variability of income.

The superiority of Optuna becomes even more evident when compared with Successive Halving. For the Halving vs. Optuna comparison, the p-value of 0.00005 again indicates strong statistical significance. The effect size of 51,499 is the largest among all model pairs, suggesting that Optuna not only outperforms the alternatives statistically but also delivers the most substantial improvement in predictive performance. This value reflects a consistent reduction in RMSE that is practically meaningful within the context of microeconomic data characterized by high variance. In monetary terms, this implies that the model tuned with Optuna achieves, on average, approximately 51,500-rupiah lower error under bootstrap sampling. Although modest in nominal size, this difference is statistically significant and consistent across bootstrap samples.

Overall, p-values greater than 0.05 indicate that performance differences between methods cannot be distinguished from random variation, whereas p-values below 0.05—when accompanied by a positive and moderately sized effect—indicate that the improvement in performance has a genuine statistical basis. Based on both indicators, Optuna emerges as the most consistent and superior tuning approach compared with Random Search and Successive Halving.

E. Feature Importance

Feature-importance stability was quantitatively evaluated using rank-based correlation measures, including Spearman's rank correlation and Kendall's Tau, to assess the consistency of feature rankings across different hyperparameter optimization methods. The stability analysis of feature importance was conducted to assess whether hyperparameter optimization alters the interpretability of the CatBoost model in predicting individual income. In socio-economic contexts, models are required not only to achieve high predictive accuracy but also to maintain interpretive consistency, ensuring that substantive insights and policy implications derived from the model can be reliably supported.

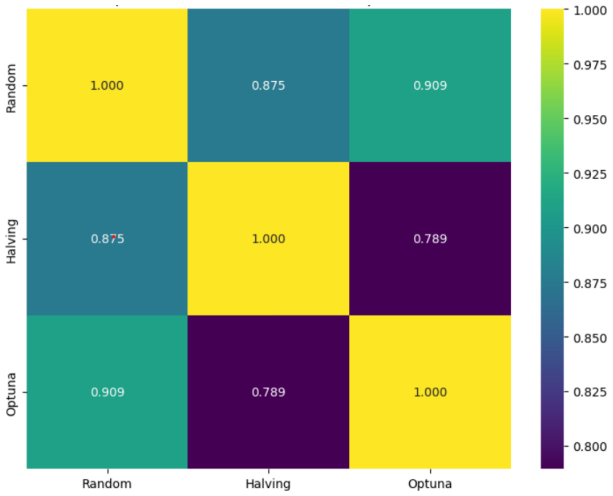

Figure 8. Normalized Feature Importance Across Tuning Methods

Spearman correlation was employed to assess the agreement of feature-importance rankings across models. Figure 8 demonstrates a very high level of consistency among the three tuning methods. The Random Search vs. Optuna pair records the highest correlation ($\rho = 0.909$), followed by Random Search vs. Successive Halving ($\rho = 0.875$), and Successive Halving vs. Optuna ($\rho = 0.789$). The Spearman correlations indicate strong rank-order consistency, with p-values< 0.001 confirming their statistical significance. These findings suggest that despite utilizing different optimization

strategies—random exploration, iterative elimination, or Bayesian optimization—the global structure of feature rankings remains stable.

TABLE VI
SPEARMAN TEST

Comparing	Spearman r	p-value
Random Search vs Halving Successive	0.875	0.0000
Random Search vs Optuna	0.909	0.0000
Halving Successive vs Optuna	0.789	0.0001

Table 6 reinforces these visual findings through a significance test on the Spearman correlation coefficients. All pairs of methods exhibit p-values < 0.001, indicating that the agreement in feature-ranking patterns is not due to statistical coincidence. Thus, it can be concluded that changes in hyperparameter configurations—whether driven by random exploration, iterative elimination, or Bayesian optimization—do not produce any meaningful shifts in the model's interpretive structure. This stability is particularly important in socio-economic analysis, as it ensures that the interpretation of each variable's contribution to individual income remains consistent even when the model undergoes technical adjustments aimed at improving accuracy. In other words, hyperparameter optimization enhances predictive performance without compromising the coherence of the model's interpretability.

The stability analysis of feature importance was conducted to determine whether the hyperparameter optimization process leads to differences in interpretive structure or whether the model retains a consistent pattern of feature contributions. To this end, two types of visualizations were used: the Rank Stability Plot and the Kendall's Tau Correlation Heatmap.

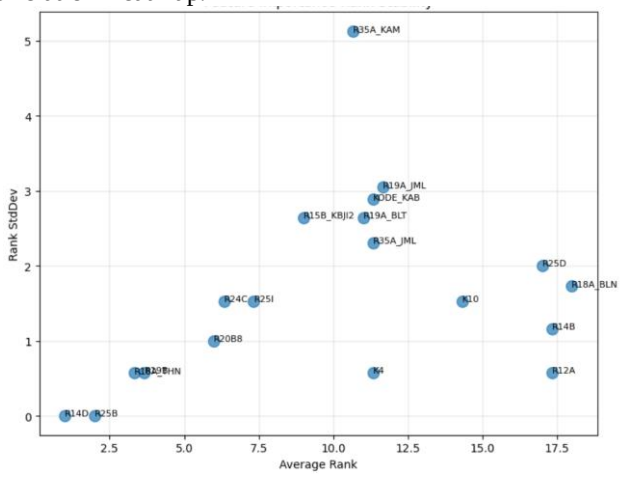

Figure 9. Rank Stability Plot

A deeper examination using the Rank Stability Plot reveals varying degrees of consistency across features. Structural variables such as R14D (business registered in the licensing system), R25B (work accident insurance), and R25I (wage compliance with the minimum provincial wage) exhibit rank deviations close to zero, indicating that these features consistently receive high importance rankings across models, although this analysis does not imply causal relationships. In contrast, variables such as R19A_JML (weekly working hours) and demographic attributes (K4, K10) display larger ranking fluctuations, suggesting that they are more sensitive to hyperparameter configurations.

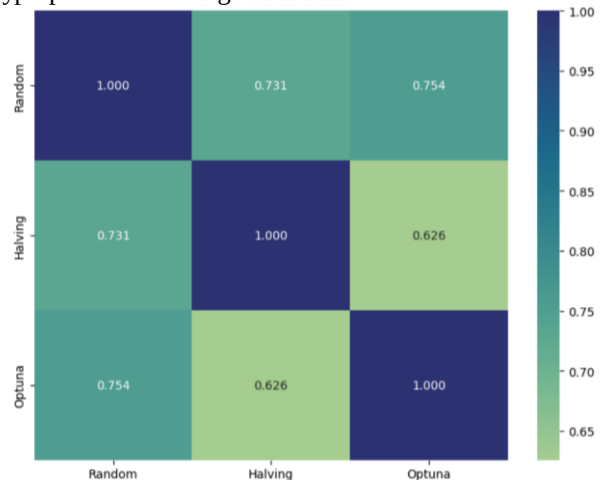

Figure 10. Kendall Tau Correlation Heatmap

The consistency of this pattern is further supported by the Kendall's Tau Correlation (Figure 10), which ranges from 0.626 to 0.754, indicating moderate to strong agreement in ordinal alignment. These results are consistent with the Spearman correlation findings, whereby the most influential features consistently appear at the top of the rankings, and the less informative features remain at the bottom, regardless of the tuning method applied.

The observed interpretability stability has important implications for socio-economic policy research. Structural factors such as business legality, employment protection, and wage compliance consistently emerge as strong determinants of income, whereas contextual variables such as working hours and demographic characteristics exhibit a more flexible role. These findings validate that hyperparameter optimization in CatBoost enhances predictive accuracy without compromising the coherence of the model's interpretive structure, thereby ensuring the reliability of evidence-based policy analysis.

Accordingly, it can be concluded that all three tuning methods yield models with equivalent interpretability, where differences in accuracy are not accompanied by substantive differences in feature-importance structure.

F. Discussion, Limitations and Policy Implication

This study relies on cross-sectional SAKERNAS data from a single year (2024). As a result, the findings reflect model behavior under a specific economic condition and labor-market structure. Temporal generalization across different economic cycles or labor-market shocks is not directly

evaluated in this study. Although the proposed modeling framework demonstrates strong predictive performance and interpretability stability, the results should not be interpreted as causal. The model captures statistical associations rather than structural economic mechanisms. Consequently, direct policy prescriptions based solely on model outputs should be made with caution. Future studies may extend this evaluation to multi-year SAKERNAS data or alternative socio-economic datasets to assess temporal robustness and policy sensitivity under different macroeconomic conditions. From a policy perspective, the observed sensitivity to hyperparameter optimization underscores the importance of methodological transparency in data-driven decision making. Policymakers should be aware that model performance—and consequently, predictions affecting resource allocation—can vary significantly based on technical choices often invisible to non-experts. We recommend establishing validation protocols that test model robustness across multiple optimization strategies before deployment in high-stakes socioeconomic applications.

IV. CONCLUSION

This study demonstrates that the choice of hyperparameter optimization method significantly influences the performance of the CatBoost model in predicting individual income using the 2024 SAKERNAS dataset. Among the three approaches evaluated, Optuna delivers the best performance based on RMSE, MAE, and R-squared, further supported by a 10,000-replication bootstrap significance test confirming that the improvement is not attributable to statistical randomness. Successive Halving shows competitive performance and efficient use of computational resources, whereas Random Search delivers results that fall between those of Successive Halving and Optuna within the same evaluation framework.

The RMSE values—ranging from several hundred thousand to around one million rupiah—reflect the heavy-tailed and highly variable nature of the income distribution, indicating that error interpretation must account for this distributional structure. The feature-importance analysis reveals consistent interpretive patterns across tuning methods, characterized by high Spearman correlations and stable feature rankings. Thus, hyperparameter optimization improves model accuracy without altering the substantive structure of feature contributions.

Overall, this study highlights Optuna as the most effective optimization method for obtaining optimal CatBoost hyperparameters in large-scale socio-economic data. Successive Halving serves as an efficient alternative when computational resources are limited. These findings provide valuable guidance for researchers and practitioners in selecting appropriate tuning strategies for microdata modeling and support evidence-based decision-making in socio-economic research. Future research may extend this framework to other tree-based or econometric models to assess the generalizability of hyperparameter optimization strategies across algorithms.

ACKNOWLEDGMENT

The author extends his sincere appreciation to the supervisors of the Statistics and Data Science Study Program for their invaluable guidance, direction, and insightful feedback throughout the research process. Their support and expertise have served as a crucial foundation in the completion of this scientific work. Gratitude is also expressed to the School of Data Science, Mathematics, and Informatics, IPB University, for providing the facilities, opportunities, and an academically supportive environment that greatly contributed to the development of this research. The author further conveys his appreciation to all individuals who dedicated their time to offer constructive criticism and suggestions, which have significantly enhanced the quality of this study.

REFERENCES

[1] B. Bischl *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 2021, doi: 10.1002/widm.1484.

[2] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp. 2623–2631, 2019, doi: 10.1145/3292500.3330701.

[3] J. T. Hancock and T. M. Khoshgoftaar, "CatBoost for big data: an interdisciplinary review," *J. Big Data*, vol. 7, no. 94, 2020, doi: 10.1186/s40537-020-00369-8.

[4] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: Unbiased boosting with categorical features," *arXiv*, 2019.

[5] F. W. Hartono, Muljono, and A. Z. Fanani, "Improving the Accuracy of House Price Prediction using Catboost Regression with Random Search Hyperparameter Tuning: A Comparative Analysis," *Adv. Sustain. Sci. Eng. Technol.*, vol. 6, no. 3, 2024, doi: 10.26877/asset.v6i3.602.

[6] S. Jeganathan, A. R. Lakshminarayanan, S. Parthasarathy, A. A. A. Khan, and K. J. Sathick, "OptCatB: Optuna Hyperparameter Optimization Model to Forecast the Educational Proficiency of Immigrant Students based on Cat Boost Regression," *J. Internet Serv. Inf. Secur.*, vol. 14, no. 2, pp. 111–132, 2024, doi: 10.58346/JISIS.2024.I2.008.

[7] L. B. Klebanov, Y. V. Kuvaeva-Gudoshnikova, and S. T. Rachev, "Heavy-Tailed Probability Distributions: Some Examples of Their Appearance," *Mathematics*, vol. 11, no. 14, pp. 1–7, 2023, doi: 10.3390/math11143094.

[8] S. Karlsson, S. Mazur, and H. Nguyen, "Vector autoregression models with skewness and heavy tails," *J. Econ. Dyn. Control*, vol. 146, 2023, doi: 10.1016/j.jedc.2022.104580.

[9] K. Ouédraogo and D. Barro, "An Approach of Estimating the Value at Risk of Heavy-tailed Distribution using Copulas," *Eur. J. Pure Appl. Math.*, vol. 15, no. 4, pp. 2074–2085, 2022, doi: 10.29020/nybg.ejpam.v15i4.4280.

[10] A. V. Dorogush, V. Ershov, and A. Gulin, "CatBoost: gradient boosting with categorical features support," *arXiv*, pp. 1–7, 2018, [Online]. Available: http://arxiv.org/abs/1810.11363

[11] X. Wei, C. Rao, X. Xiao, L. Chen, and M. Goh, "Risk assessment of cardiovascular disease based on SOLSSA-CatBoost model," *Expert Syst. Appl.*, vol. 219, no. January, p. 119648, 2023, doi: 10.1016/j.eswa.2023.119648.

[12] D. Micci-Barreca, "A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems," *SIGKDD Explor.*, vol. 3, no. 1, pp. 27–32, 2001, doi: 10.1145/507533.507538.

[13] G. Huang *et al.*, "Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions," *J. Hydrol.*, vol. 574, no. April, pp. 1029–1041, 2019, doi:

10.1016/j.jhydrol.2019.04.085.

[14] A. R. M. Rom, N. Jamil, and S. Ibrahim, "Multi objective hyperparameter tuning via random search on deep learning models," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 22, no. 4, pp. 956–968, 2024, doi: 10.12928/TELKOMNIKA.v22i4.25847.

[15] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.

[16] L. Villalobos-Arias, C. Quesada-López, J. Guevara-Coto, A. Martínez, and M. Jenkins, "Evaluating hyper-parameter tuning using random search in support vector machines for software effort estimation," *PROMISE 2020 - Proc. 16th ACM Int. Conf. Predict. Model. Data Anal. Softw. Eng. Co-located with ESEC/FSE 2020*, 2020, doi: 10.1145/3416508.3417121.

[17] R. Aschauer, "Predictive Modeling of Next Product to Buy in the Banking Sector Using Boosting Techniques," no. June, 2010.

[18] M. Ali, M. S. Azam, and T. Shahzad, "Random Search-Based Parameter Optimization on Binary Classifiers for Software Defect Prediction," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 10, no. 2, pp. 476–488, 2024, doi: 10.26555/jiteki.v10i2.28973.

[19] H. Yang, Z. Tian, X. Li, and H. Liu, "An Antenna Optimization Method Based on Optuna-ANN," *2023 IEEE 11th Asia-Pacific Conf. Antennas Propagation, APCAP 2023 - Proc.*, vol. volume1, pp. 1–3, 2023, doi: 10.1109/APCAP59480.2023.10469687.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, and B. Thirion, "Scikit-learn: Machine Learning in Python Fabian," *J. ofMachine Learn. Res.*, vol. 12, pp. 2826–2830, 2011, doi: 10.4018/978-1-5225-9902-9.ch008.

[21] Z. Karnin, T. Koren, and O. Somekh, "Almost optimal exploration in multi-armed bandits," *30th Int. Conf. Mach. Learn. ICML 2013*, vol. 28, no. PART 3, pp. 2275–2283, 2013.

[22] K. Jamieson and A. Talwalkar, "Non-stochastic best arm identification and hyperparameter optimization," *Proc. 19th Int. Conf. Artif. Intell. Stat. AISTATS 2016*, 2015.

[23] D. S. Soper, "Hyperparameter Optimization Using Successive Halving with Greedy Cross Validation," *Algorithms*, vol. 16, no. 1, 2023, doi: 10.3390/a16010017.