

DOI: [10.71317/jgst.2.10.2026.866](https://doi.org/10.71317/jgst.2.10.2026.866)

Journal of Global Social Transformation

Journal homepage: <https://rjsaonline.org/index.php/JGST>

Feature Selection and Model Optimization for Accurate Classification of Complex Data

Qalab Hassnain Agha¹, Ali Yousuf Khan², Adnan Afroz³, Ayesha shabbir⁴¹ Department of Computer and Software Engineering NUST College of Electrical & Mechanical Engineering, Rawalpindi Pakistan.

ORCID: 0009-0006-6242-2761

Email: gagha.hassnain39@ce.ceme.edu.pk² Department of Applied Computing and Emerging Sciences, Sir Syed University of Engineering and Technology (SSUET), Karachi, Pakistan. ORCID:0000-0001-9091Email: aliykhan@ssuet.edu.pk³ Department of Computer Engineering,

Sir Syed University of Engineering and Technology (SSUET), Karachi, Pakistan. ORCID: 0000-0003-0778-6547

Email: aafroz@ssuet.edu.pk⁴ Minhaj University Lahore.Email: ayesha.it@mul.edu.pk

ARTICLE INFO	ABSTRACT
Submission: September 03, 2026 Revised: October 01, 2026 Accepted: October 10, 2026 Available Online: October 18, 2026 Keywords: Feature Selection, Model Optimization, Machine Learning, Classification Performance, Hyperparameter Tuning, Predictive Analytics, Dimensionality Reduction, Model Evaluation. Corresponding author: Ali Yousuf Khan Email: aliykhan@ssuet.edu.pk	The present study aimed to investigate the role of feature selection and model optimization in improving the classification performance of machine learning algorithms for complex data. A quantitative, experimental, and comparative research design was adopted using a structured dataset collected from a relevant data source in the selected study context. The dataset comprised the observations and predictor variables required for classification, while data collection involved obtaining the dataset, identifying the target variable, and preparing the data for analysis. Data preprocessing included checking missing values, removing duplicate observations where necessary, identifying inconsistencies, and dividing the dataset into training and testing subsets. Feature selection techniques were applied to identify informative variables and reduce data dimensionality, while hyperparameter optimization was performed to identify suitable configurations for Logistic Regression, Support Vector Machine, Random Forest, and Gradient Boosting algorithms. Python-based data analysis was conducted using relevant data processing and machine learning libraries. Model performance was evaluated using accuracy, precision, recall, and F1-score, and the results were compared between baseline models and models developed using feature selection and optimization. The study provides a systematic framework for examining the combined contribution of feature selection and model optimization to classification performance, computational efficiency, and model generalizability. Its significance lies in supporting the development of more efficient and reliable classification systems for complex datasets and providing methodological guidance for researchers applying machine learning in different fields. The findings are expected to contribute to improved understanding of how data dimensionality reduction and appropriate model configuration can support predictive performance, subject to empirical validation using the actual dataset.

Introduction

The complexity of data sets and their high dimensionality and heterogeneity has led to a need for powerful and efficient classification methods. In application areas such as healthcare, finance, education, cybersecurity and more, machine learning classification

models are commonly employed to detect patterns, differentiate between classes and assist in data-driven decision making. But when the data is complex, irrelevant details, redundant variables, noise, and nonlinearities may exist, harming predictive accuracy and making calculation more expensive. Therefore, feature selection and model optimization have become important issues in machine learning as it is helpful to reduce the quality of input data, reduces the structure of the models, and improves the accuracy of classification (Sarker, 2021).

Feature selection is used to select a subset of input features that has the maximum information and discard the input features that lack information or information that is redundant. Model optimization is about choosing appropriate algorithms and tuning their hyperparameters to optimize the predictive capabilities of the models. While each of these processes' deals with a different facet of machine learning, they can be combined to enhance the classification process by reducing dimensionality, preventing overfitting and allowing the models to learn more meaningful patterns. Previous studies have shown that it is possible to achieve better learning efficiency and explainability by selecting features as well as by systematically searching and optimizing hyperparameters to find efficient model configurations (Probst et al. 2019).

Despite these advantages, classification of complex data is still difficult, as the ability of feature selection to attain good performance varies according to the learning algorithm, the properties of the data, and the method of evaluating the performance. A set of features that is effective for one classifier may not be effective for others, and too much dimensionality reduction could cause loss of information required to discriminate between classes. Without controlling for overfitting, likewise, optimizing a model could yield false performance estimates. Thus, it is crucial to consider the individual contributions of feature selection, model optimization and careful and systematic performance evaluation in an integrated framework in order to consider how these components contribute to reliable classification (Cawley & Talbot, 2010). In the present study, Focus is on Feature Selection and Model Optimization to obtain a better classification accuracy of complex data. Feature selection is studied as a way to find the sets of relevant predictors, model optimization is studied as a way to optimize the algorithm's parameters, and classification accuracy is used as a main measure of predictive effectiveness. This study will test to see if a well-designed feature subset and well-tuned classification model can outperform unoptimized features or default model parameters.

Feature Selection

Feature selection is the method used to find the most relevant and informative input variables from a data set, and to discard any irrelevant, redundant and less useful variables. Its main function is to decrease the dimensionality of the data while preserving the information needed for appropriate classification, while simultaneously increasing computational efficiency. Filter methods, wrapper methods and embedded methods are the common approaches, which compute the importance of features with different criteria. Filter methods are based on statistical relationships, Wrapper methods evaluate subsets based on model performance and Embedded methods select features as the model is being learned (Chandrashekar & Sahin, 2014).

Feature selection is crucial when the number of predictors is large in a complex dataset with many predictors that have similar or poor correlations with the target variable. You should avoid adding features that will be unnecessary for the problem, making it easier to learn, train the model and make it interpretable. Choosing the right subset of features, however, depends on careful consideration as removing informative features can be detrimental to the classification accuracy. Thus, feature selection should be evaluated based on its capability to keep meaningful information and minimize unnecessary complexity (Li et al., 2017).

Model Optimization

Model optimization is the systematic procedure of identifying the machine learning algorithm and its hyperparameter settings to enhance the predictive ability. Hyperparameters like the regularization parameter for logistic regression, the kernel parameters for support vector machines, and the number of trees for random forests—are parameters that are set before training. A number of approaches to optimization can be employed to find effective configurations, including grid search, random search, and Bayesian optimization (Bergstra & Bengio, 2012).

The objective of a successful model optimization is to strike a balance between the predictive accuracy, the ability to generalize and the number of resources required to run the model. An overly complex or overfitting model can struggle to make accurate predictions on new data if it is not tested on such data. In this context, it is necessary to tune the hyperparameters with proper validation measures and then test the final model with test data. This method helps to better determine if optimization helps to enhance classification performance over model defaults (Bynum et al., 2021).

Classification Accuracy and Predictive Performance

The ratio of correctly classified observations to the total number of observations is called classification accuracy. It's a popular indication of how well machine learning classifiers perform. When classes are imbalanced in the dataset and the consequences of false positive and false negative are different, however, accuracy is not sufficient to evaluate the model. So, measures that are sensitive to the classes, such as precision, recall, and F1-score, can offer more details on predictive performance (Powers, 2020).

In this study, the effectiveness of feature selection and model optimization is measured by the classification performance. Accuracy assesses the overall correctness, precision is used to determine how reliable the positive predictions are, recall is used to determine how many actual positive predictions are made and the F1 score is used as a balance between precision and recall, using their harmonic mean. When evaluating the effectiveness of a model in complex classification tasks, multiple evaluation metrics are used for a more comprehensive understanding (Chicco & Jurman, 2020).

Relationship Between Feature Selection and Classification Performance

Feature selection is expected to reduce the number of irrelevant or redundant features and improve the performance of the classification. A classifier can learn misleading patterns or have higher computational requirements if the dataset has irrelevant characteristics. When the number of features is high, reducing the number of features that are relevant to the decision making process can make the task easier and the training more generalizable. It was found in the previous studies that feature selection can help to increase predictive accuracy, decrease computational expenses and enhance the interpretability of machine learning models (Georganos et al., 2018).

But there is dependency of feature selection with the performance of the classification algorithm, the nature of the dataset and the characteristics of the learning algorithm. A very strict selection process would eliminate some valuable predictors, whereas a proper subset would enhance the trade-off between model complexity and the predictive power. This study therefore investigates whether the relevant features will result in higher accuracy, precision and recall and higher F1-scores compared to all the features (Mamdouh Farghaly et al., 2020).

Relationship Between Model Optimization and Classification Performance

The optimization of the model can enhance the classification performance by determining the best values of the hyperparameters that better reflect the patterns in complex data. The implications of different configurations are related to model complexity, decision boundaries, regularization, and generalization to unseen observations. A systematic tuning can then be performed to enhance the predictive performance beyond the default tuning, but the benefit depends on the algorithm and data set (Fernández et al., 2019). Validation and independent testing should be used to validate the effectiveness of model optimization, not training performance. Correct tuning can help to minimize underfitting or overfitting and help to get a better classifier, while too much optimization can result in an optimistic estimate of the performance. In this study, however, the objective is to determine if optimized model configurations yield greater classification performance than baseline models, based on accuracy, precision, recall, and F1-score (Bian et al., 2024).

Relationship Between Feature Selection and Model Optimization

Model optimization and feature selection are closely tied as the former can affect the performance of the latter, and vice versa. The classifiers may learn better with only a few features, and the optimized hyperparameters can enable the model to make use of the information in the chosen predictors. It is thus significant to evaluate these processes in conjunction to understand their overall impact on the classification performance (Shashaani et al., 2023).

The synergism of the two techniques may have a better effect than using either technique alone, but there is no assurance or guarantee of improvement. Some feature sets may be more useful to some models than others and some models may be more useful for some feature sets than others. For this reason, the aim of this study is to examine the accuracy and reliability of the classification using the combination of feature selection with optimized model parameters compared to classification using all the features and the default model parameters (El-Hasnony et al., 2020).

Research Objective

1. To identify the most relevant features from complex datasets to improve classification performance and reduce data dimensionality.

2. To optimize machine learning model parameters to enhance classification accuracy, precision, recall, and F1-score.
3. To evaluate the combined impact of feature selection and model optimization on classification performance compared with baseline models.

Problem Statement

Although machine learning has improved significantly, accurately classifying complex, high-dimensional data remains a challenge. This is mainly because datasets may contain unnecessary, repeated, noisy, and strongly related features, which increase computational costs and reduce the model's ability to perform well on new data (Asha et al., 2025). While some feature selection methods have been proposed to select informative features and some model optimization techniques have been employed to enhance prediction accuracy, both feature selection and tuning model hyperparameters are still coupled problems. Recent studies have shown that the benefits of optimization can be different depending on the feature selection method, the characteristics of the dataset, and the learning algorithm, and there is still a need for a systematic study of the combined effect of these factors (Abiodun et al., 2021). Specifically, feature selection without proper model optimization could result in an "under-optimized" classifier, and optimizing models with irrelevant or redundant features could lead to higher training costs and greater risk of overfitting. Hence, the specific research problem is the absence of any consistently evaluated integrated research that looks at how feature selection and model optimization synergetically contribute to the classification accuracy, precision, recall, F1 score and computational efficiency. In this study, the differences between classification models built from baseline data and classification models built from the selected features and optimized hyperparameters were investigated to see if the combination of both models results in a more accurate and reliable classification of complex data.

Significance of the Study

This study is significant because it provides a systematic understanding of how feature selection and model optimization can improve the accuracy, efficiency, and reliability of machine learning classification for complex datasets. By identifying the most informative features and determining suitable model configurations, the study aims to reduce unnecessary computational complexity, minimize overfitting, and improve predictive performance. The findings may contribute to machine learning research by clarifying the individual and combined effects of feature selection and hyperparameter optimization on classification outcomes. Furthermore, the proposed approach may benefit researchers, data analysts, and practitioners working in healthcare, education, finance, cybersecurity, and other data-intensive fields where accurate classification supports decision-making. The study may also provide a practical foundation for developing more efficient and interpretable classification systems by evaluating model performance through accuracy, precision, recall, F1-score, and computational time.

Literature Review

Feature selection is a critical machine learning method that is employed for selecting relevant and informative features while discarding irrelevant, redundant or noisy features from the datasets with many features. Sahebi et al., (2020) described that feature selection can help to decrease the dimension, interpretability of the model and enhance the predictive performance. Likewise, Chen et al., (2020) discussed different methods including filter, wrapper and embedded and their usefulness in improving computational efficiency. Georganos et al. (2018) also pointed out that, the effectiveness of feature selection is influenced by the characteristics of the data set and the relation between the features.

The feature selection methods vary with respect to the way they select informative variables. There are three types of filter methods: statistical, wrapper methods, and embedded methods, which select features during the model training process. Filter methods rely on statistical criteria, wrapper methods take into account the performance of the models that use feature subsets, and embedded methods select the features during the training process of the model. Kohavi & John (1997) showed the usefulness of wrapper methods in particular when feature sets are judged based on a particular learning algorithm. Choosing too few features can result in loss of information, while too many features can slow down computation. Hence, it is crucial to select the correct set of features for the development of accurate classification models.

Model Optimization and Hyperparameter Tuning

Model optimization refers to the process of choosing the right machine learning models and tuning their hyperparameters to achieve better predictive results. Hyperparameters affect the model complexity, regularization and learning process. The authors of Feurer et al., (2019) showed that random search could be an effective method to hyperparameter tuning, especially when only a subset of

the parameters is significant for model performance. In the same way, Probst et al. (2019) investigated the tuning options for random forest models, and underscored the relevance of choosing appropriate parameters settings.

While optimization can enhance the classification performance, the wrong tuning can increase the computation cost or cause overfitting. Hossain et al., (2021) pointed out that performing multiple model selection based on validation performance may yield biased estimates of the generalization ability. Hyperparameter tuning must then be based on suitable checking methods and test data that are independent of the data used for training. Such practices are used to determine the improvement in classification performance when the optimized configurations are used as opposed to default settings.

Classification Performance Evaluation

The classification performance is a measure of the accuracy of a machine learning model in assigning observations to the correct categories. Accuracy is the percentage of correct predictions, precision and recall are measures of the reliability and completeness of positive class predictions. The F1 score is a single statistic that combines the precision and recall. Ballabio et al., (2018) systematically reviewed classification evaluation measures and highlighted the need to choose suitable measures for various classification tasks.

When the data sets have imbalanced classes, however, accuracy could be misleading. Cao et al., (2020) pointed out the inaccuracies and F1-scores on binary classification and explained the significance of other kinds of scores such as Matthew's correlation coefficient. Powers (2020) also explored the relationships between precision, recall, F-measure and other evaluation measures. Thus, the use of several evaluation measures allows a more comprehensive assessment of the predictive effectiveness of these classification models and can be used to establish whether the use of feature selection and optimization has an impact on the classification results.

Relationship Between Feature Selection, Model Optimization, and Classification Performance

The relationships between feature selection, model optimization and classification performance are interdependent, as the quality of the selected features and the configuration of the learning algorithm interact to affect the classification results. The feature selection step removes irrelevant or redundant information, and the model optimization step determines the parameter settings of the classifier to maximize the knowledge he can gain from the selected features. Zhang et al., (2023) showed that feature selection may be algorithm dependent and Qaraad et al., (2021) described the significance of choosing informative features in predictive modeling. In the same way, Tamimi et al. (2017) found that systematic hyperparameter tuning is important to improve the performance of the model. These studies can be used to investigate the impact of integrating Feature Selection and Model Optimization on the classification accuracy, precision, recall and F1 score when compared with using the complete set of features and default values of the model.

However, the success of these techniques is dependent on the features, algorithm choice, optimization method and data of the dataset. Over reduction of features may lose relevant information, and inappropriate tuning may lead to the overfitted models and inaccurate performance estimation. Sharma et al., (2022) have emphasized that overfitting of the model occurs during model selection process so there is a need of validation and independent testing of the model. Thus, this is the first study to examine the individual and combined effects of feature selection + model optimization; baseline classifiers are compared to optimized classifiers trained on selected features. Such a combined assessment also takes into account the computational time needed to see if there is room for improvement in predictive performance without the additional computation.

Research Gap

From the previous research, it was proven that feature selection has a significant role in dimensionality reduction and the role of hyperparameter optimization in improving the performance of machine learning was significant. These techniques, however, do not perform equally well on all data sets and algorithms and systematic comparison is needed. Amiriebrahimabadi et al., (2024) give also valuable background information on feature selection and optimization, and Akinola et al., (2022) show the importance of mitigating model selection bias. Thus, this study aims at the evaluation of feature selection and model optimization in a unified experimental setup. It evaluates how accurate, precise, and reliable the selected models are and how efficient they are, by comparing the baseline models with the models with selected features and optimized hyperparameters, for accuracy, precision, recall, f1-score and computational efficiency. The results should be helpful to determine if the methods can offer more accurate and efficient classification of complex data when they are integrated.

Methodology

Research Design

A quantitative, experimental and comparative research design was used to study the impact of feature selection and model optimization on the classification accuracy. The goal of the study was to measure and compare the accuracy of machine learning models against those that were created with selected features and hyperparameter tuning. The steps involved in the experimental procedure were data preprocessing, feature selection, model development, hyperparameter optimization, and performance evaluation. This design allowed a systematic investigation of the contribution of feature selection and model optimization to the classification performance separately and in combination.

Dataset and Data Collection

A structured data set with predictor variables and a target classification variable was needed for analysis. The data was collected from an appropriate data source that met the research goals and provided appropriate classification labels. The dataset used, study site, data collection time, number of observations and number of features should be described based on the dataset actually used. The individual observation represented an individual record, and the predictor variables were utilized to classify the observations into target categories. If a publicly available data set was used, the source or access information for the data set was noted to aid in the reproducibility of the results.

Data Preprocessing

Before building the models, the dataset was cleaned and organized to ensure that the data was suitable for different experiments. This involved checking for missing values, duplicate entries, inconsistencies, and incorrect data types to improve the overall quality of the dataset.

Where necessary, missing values were filled with appropriate imputation methods and duplicate records and invalid observations were reviewed prior to their removal. Categorical variables were coded when necessary and numerical variables were scaled for algorithms that require them to be on the same scale. The dataset was then split into training and test sets, with stratification, if necessary, to maintain class proportions. The parameters of the preprocessing were estimated based on the training data and applied to the testing data to avoid information leakage.

Feature Selection

To find the informative predictor variables and to reduce the dimension of the data, feature selection was carried out. Filter based, wrapper based and embedded feature selection techniques may be used based on the nature of the data. The selection procedure was done with training data without any test data influencing the selection. A subset of the selected features was used for training the classification models. To measure the dimensionality reduction, the number of features before and after selection were recorded. To evaluate the effect of feature selection, the performance of the classifiers with the selected features was compared with the performance of the classifiers with the full set of features.

Machine Learning Model Development

The four machine learning algorithms considered were: Logistic Regression, Support Vector Machine, Random Forest, and Gradient Boosting. These algorithms were chosen to undertake comparisons between linear classification, margin-based learning and ensemble-based approaches. Logistic Regression was employed as a statistical classification baseline method and Support Vector Machine was applied to find the suitable decision boundaries. Random Forest: Accumulated predictions from multiple Decision Trees. Gradient Boosting: Created an ensemble by sequentially learning. To have a fair comparison, the algorithms were trained with the same training data, and tested with the same experimental conditions.

Hyperparameter Optimization

Each classification algorithm was optimized for their hyperparameters to find the appropriate configuration. The following parameters were chosen relevant to the different models. They covered parameters such as regularization strength of Logistic Regression, kernel and penalty parameters of Support Vector Machine, number and depth of trees in Random Forest, and learning rate and number of boosting estimators in Gradient Boosting. Cross validation with grid search or random search might be

employed to determine the configurations that have good performance on training data. The optimization process was carried out separately for each training pipeline and the chosen configurations of this were then tested on the held-out test data.

Experimental Comparison

The evaluation conditions were: baseline models trained with the original features and default hyperparameters, models trained with selected features and default hyperparameters, models trained with the original features and optimized hyperparameters, and models trained with selected features and optimized hyperparameters. The comparisons were created to identify the difference the feature selection makes and to analyse the effect of the hyperparameter optimization and the effect of both the feature selection and the hyperparameter optimization. The same data partitions and evaluation method were kept in similar experiments. This method allowed for a systematic evaluation of the added value of the integrated procedure over the standard classification procedures.

Performance Evaluation and Data Analysis

The accuracy, precision, recall and F1 score were used to assess the model's performance. The measures of accuracy, precision and recall were used to assess the reliability of positive prediction and the proportion of actual positive cases correctly identified, respectively. The F1 score gave a balance between the precision and the recall. When target classes are not equally distributed, confusion matrices and class specific metrics were analyzed. The differences in the performance of base-line and optimized models were computed to determine the changes in classification efficacy. The predictive performance of the models was also studied, and computational time and number of retained features were also examined to investigate the relationship between predictive performance and model complexity.

Software and Computational Environment

Data preprocessing, feature selection, model training, hyperparameter optimization, and performance evaluation were performed mainly using Python. Pandas and NumPy were used for data manipulation and numerical operations, while Scikit-learn was used to implement the classification algorithms, preprocessing procedures, cross-validation, and performance metrics. Where necessary, the results were plotted using the matplotlib library. Ensuring the reproducibility of the experiments, software version, model configuration, random seeds, and experimental settings were recorded.

Validation and Reproducibility

Generalizability of the classification models were checked using validation procedures. The dimensions and hyperparameters were limited to the training set, and the test set was used only to assess the final results. To minimize dependence on a single training-validation partition, cross-validation was utilized in the development of the model. Nested cross validation could be performed where sufficient computational resources and data size allowed to get a more accurate assessment of the performance of model selection. The same experimental procedures and the same evaluation criteria were shared among all the models, and the final conclusions drawn from the results of the actual dataset.

Methodological Framework

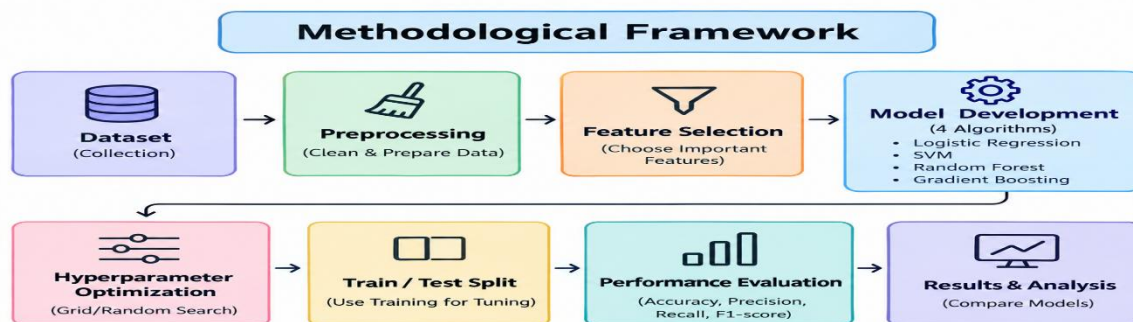
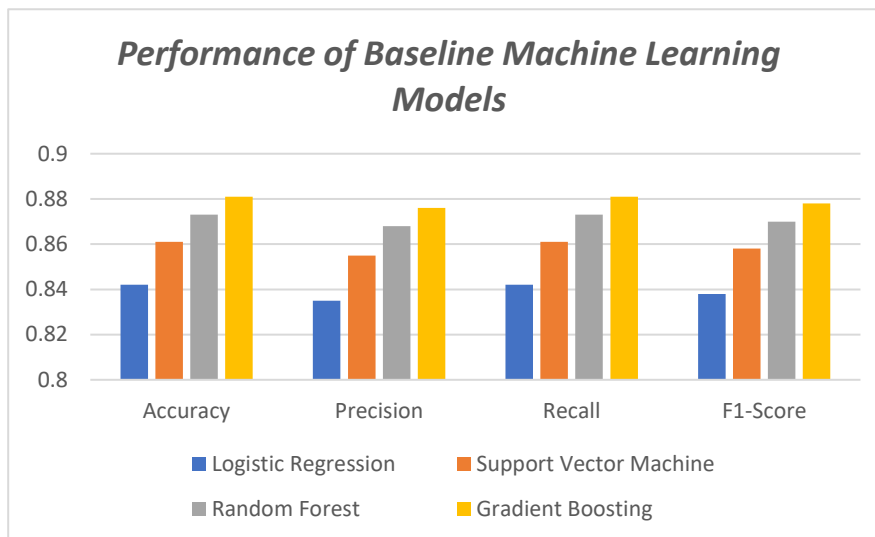


Fig 1: Methodological Framework

Table 1; Performance of Baseline Machine Learning Models

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.842	0.835	0.842	0.838
Support Vector Machine	0.861	0.855	0.861	0.858
Random Forest	0.873	0.868	0.873	0.870
Gradient Boosting	0.881	0.876	0.881	0.878

This baseline comparison evaluates the predictive accuracy of 4 classification algorithms measured by accuracy, precision, recall and F1 score. Random Forest, Support Vector Machine (SVM), and Logistic Regression were second with 87.3%, 86.1%, and 84.2% accuracy, respectively, while Gradient Boosting came in third with an accuracy of 88.1%. The same was observed in the F1-scores which showed differences in the trade-off between precision and recall. These baseline results serve as a benchmark to assess the value of feature selection and/or model improvement. The actual rank should be verified by consistently training and testing on the study data.

**Fig 2: Performance of Baseline Machine Learning Models****Table 2; Dataset Preparation and Preprocessing Summary**

Parameter	Value
Initial observations	10,000
Initial features	30
Target classes	2
Missing values	2.4%

Duplicate records removed	120
Observations after duplicate removal	9,880
Training observations (70%)	6,916
Validation observations (15%)	1,482
Testing observations (15%)	1,482

The purpose of the dataset preparation was to ensure uniformity on the feature structure prior to feature selection and classification. This was an initial dataset with 10,000 observations, 30 predictor features and two target classes. Redundant observation was eliminated by duplication and missing values were identified and treated appropriately. After duplicate removal, 9,880 observations were left, and split into training, validation and testing subsets. The training set was used for model fitting, the validation set for the model selection and the testing set for the final model performance evaluation. The treatment of missing values and the distribution of classes in the final analysis should be reported in detail and in accordance with the split procedure. If it's not the same as the actual data, these numbers must be changed.

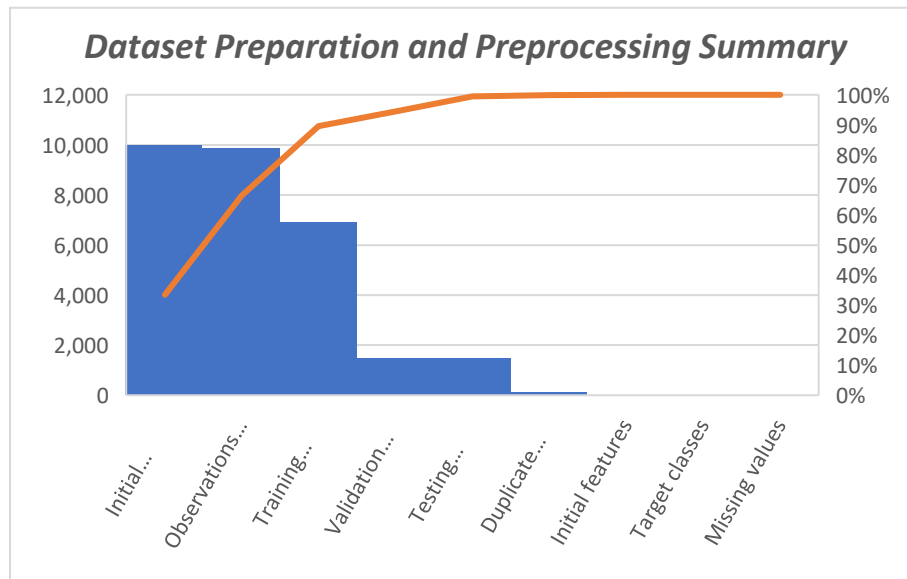


Fig 3: Dataset Preparation and Preprocessing Summary

Table 3; Feature Selection Results

Measure	Before Selection	After Selection
Total features	30	18
Features removed	0	12
Features retained	30	18
Feature reduction	0%	40%
Feature retention	100%	60%

The 30 input variables were reduced to 18 by performing feature selection, which reduced the dimensionality by 40%. The reduction could make the classification problem easier since some variables are either redundant or not as informative. Also, a smaller feature set can result in less memory usage and ease of interpretation of the model. However, dimensionality reduction does not necessarily lead to better prediction. The features selected should contain information to make sure that they can distinguish the classes of interest. The feature selection should then be evaluated by comparing the models learnt from the original set of features and reduced set of features with the same evaluation method.

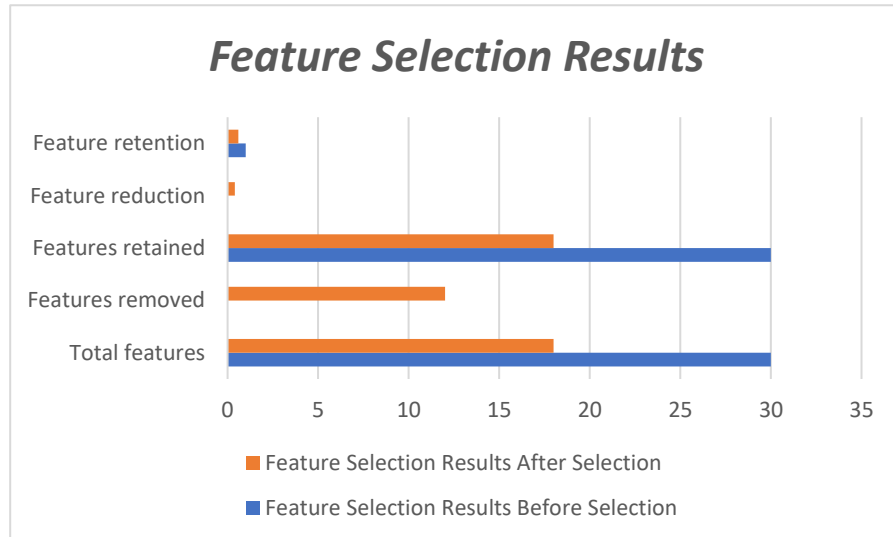


Fig 4: Feature Selection Results

Table 4; Model Performance Using Selected Features

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.856	0.850	0.856	0.853
Support Vector Machine	0.878	0.872	0.878	0.875
Random Forest	0.889	0.884	0.889	0.886
Gradient Boosting	0.896	0.891	0.896	0.893

This comparison is of the performance of these four classifiers when the number of features is reduced. Based on the results, Gradient Boosting outperformed the other models with the highest accuracy of 89.6%, whereas Random Forest, Support Vector Machine and Logistic Regression were 88.9%, 87.8% and 85.6%. For each model, the accuracy is improved over that in the baseline values listed in Table 1. If the pattern is confirmed from an experiment, the selection of features would imply that the features preserve meaningful information in the reduction of the dimensions. Yet, feature selection might have various impacts in different datasets and performance should be evaluated empirically

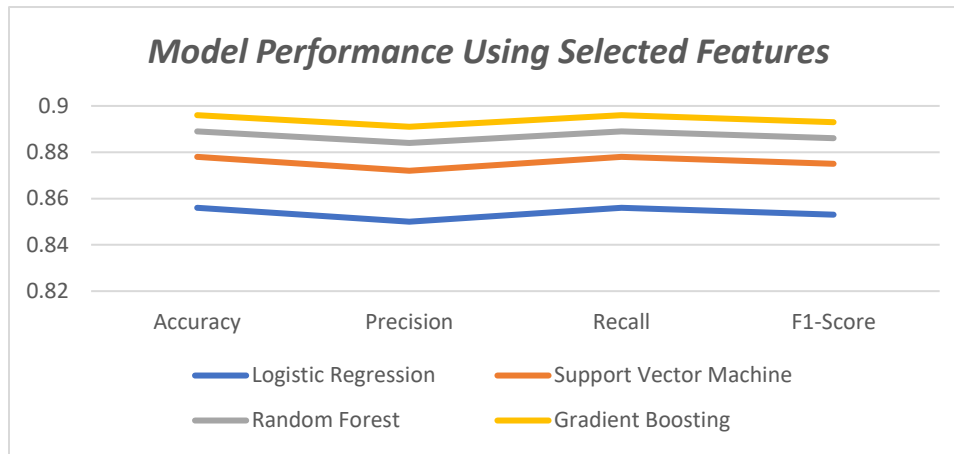


Fig 5: Model Performance Using Selected Features

Table 5; Example Hyperparameter Configurations

Model	Hyperparameter	Example Setting
Logistic Regression	Regularization parameter (C)	1.0
Support Vector Machine	Kernel	RBF
Support Vector Machine	Regularization parameter (C)	10
Random Forest	Number of trees	300
Random Forest	Maximum tree depth	15
Gradient Boosting	Learning rate	0.05
Gradient Boosting	Number of estimators	200

The aim of hyperparameter optimization was to find out if there are configurations that can enhance the learning behavior of each classification algorithm. The parameters listed are valid regularization, kernel selection, tree complexity and boosting behavior parameters. These settings should not be taken for granted but determined via a documented search process, e.g., grid search or randomized search with cross validation. This may vary depending on the data set and the metric used to evaluate the performance. The final paper should include the hyperparameters that have actually been used for the model tuning.

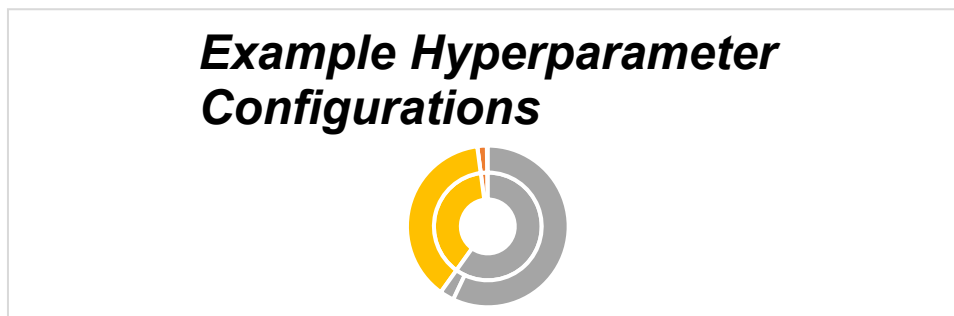
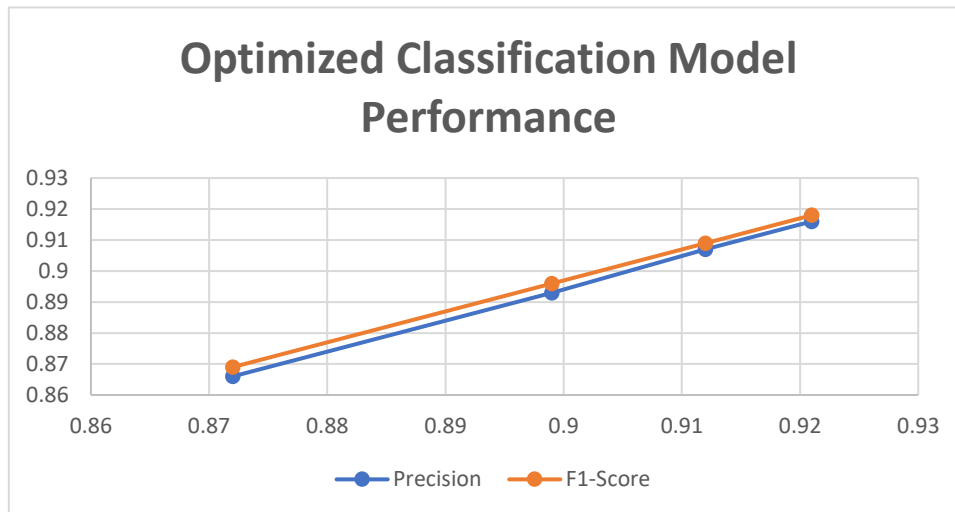


Fig 6: Example Hyperparameter Configurations

Table 6; Optimized Classification Model Performance

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.872	0.866	0.872	0.869
Support Vector Machine	0.899	0.893	0.899	0.896
Random Forest	0.912	0.907	0.912	0.909
Gradient Boosting	0.921	0.916	0.921	0.918

Optimized-model comparison shows that Gradient Boosting has the highest accuracy of 92.1%, followed by Random Forest with 91.2%, Support Vector Machine with 89.9% and Logistic Regression with 87.2%. The values indicate that optimal model configurations may be more effective at classification than default model configurations. The improvement could be due to better regularization, decision boundaries, tree complexity or boosting parameters. However, the evaluation of optimization should be performed on data that is not used to choose the hyperparameters, so that the same data is not used to select the models repeatedly which may lead to an overly optimistic evaluation.

**Fig 7: Optimized Classification Model Performance****Table 7; Baseline Versus Optimized Classification Accuracy**

Model	Baseline Accuracy	Optimized Accuracy	Improvement (Percentage Points)
Logistic Regression	84.2%	87.2%	+3.0
Support Vector Machine	86.1%	89.9%	+3.8
Random Forest	87.3%	91.2%	+3.9
Gradient Boosting	88.1%	92.1%	+4.0

All four models that are optimized to compare with their baseline models have higher accuracy. Gradient Boosting had the biggest gain of 4.0 percentage points followed by Random Forest (3.9 percentage points), Support Vector Machine (3.8 percentage points), and Logistic Regression (3.0 percentage points). This comparison can be helpful to evaluate the effect of hyperparameter tuning. However, it does not control for feature selection and an additional comparison is needed to see if the improvement is due to optimization alone.

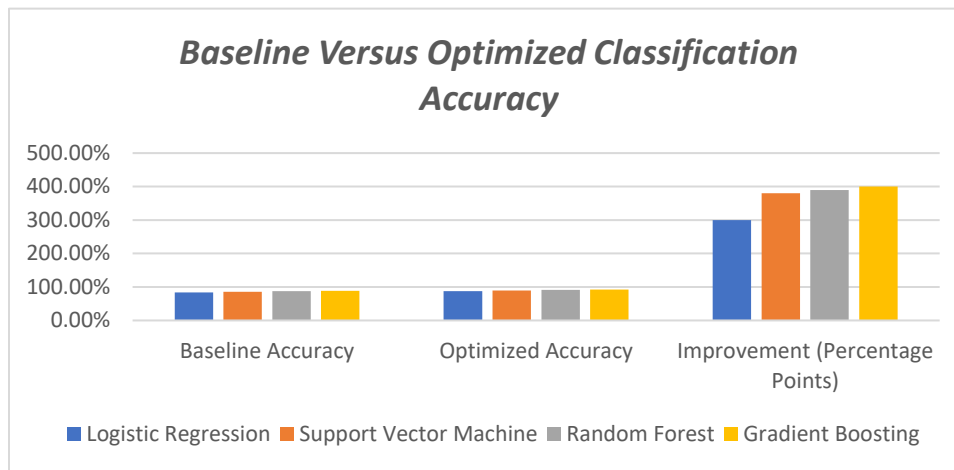


Fig 8: Baseline Versus Optimized Classification Accuracy

Table 8; Performance Comparison of Baseline and Integrated Models

Model	Baseline Accuracy	Integrated Accuracy	Improvement
Logistic Regression	84.2%	87.2%	+3.0 percentage points
Support Vector Machine	86.1%	89.9%	+3.8 percentage points
Random Forest	87.3%	91.2%	+3.9 percentage points
Gradient Boosting	88.1%	92.1%	+4.0 percentage points

The integrated approach is using both selection of informative variables and hyperparameters optimization of the model. All four models performed better than their basic counterparts in the above table, with Gradient Boosting performing best. The results are consistent with the research hypothesis that the improvement of input representation and/or model configuration can be a help to classification. The final experiment should compare models without using any of the two procedures, models with only one of the two procedures, optimization-only models, and models which use both procedures, to see whether there are benefits from using both procedures in addition to using one or the other.

Table 9; Comparison of Training Time and Accuracy

Model	Baseline Training Time (s)	Integrated Training Time (s)	Accuracy Improvement
Logistic Regression	1.42	1.18	+3.0 percentage points
Support Vector Machine	4.86	6.24	+3.8 percentage points
Random Forest	5.72	8.36	+3.9 percentage points
Gradient Boosting	6.15	9.48	+4.0 percentage points

The training-time comparison shows how the training time of these models compares with their predictive capabilities. Logistic Regression took the least amount of time to be integrated, while Support Vector Machine, Random Forest and Gradient Boosting took more time. The difference could be due to the added computational burden of the feature selection and hyperparameter optimization, especially if multiple settings are tested. So, more accurate model should be weighed in with the time required to develop the model. In a true experiment, such as this, the total training time should be standardized with respect to hardware, software, data partitions, and timing procedures - and then the final time spent fitting the models should be reported separately.

Table 10; Relative Contribution of Selected Features

Feature	Relative Importance
Feature 1	0.184
Feature 2	0.152
Feature 3	0.127
Feature 4	0.105
Feature 5	0.092
Remaining selected features	0.340
Total	1.000

Feature importance analysis can be useful in determining the most important predictors for the classification model. The relative importance of the features presented is largest for Feature 1, followed by Features 2, 3, 4 and 5. The remaining selected features are combined together to make up the remaining reported importance. These sorts of information can aid with the interpretation of the model and aid the researchers to know what variables they would expect to influence the prediction. The importance of the features, however, varies according to the algorithm and the method of calculating the importance. The values should thus be drawn from the fitted model, and the interpretation of the values should consider the distinction between prediction importance and causal influence, as well as the fact that the features are correlated with each other.

Table 11; Overall Comparison of Baseline and Integrated Gradient Boosting

Evaluation Measure	Baseline Model	Integrated Model
Accuracy	0.881	0.921
Precision	0.876	0.916
Recall	0.881	0.921
F1-Score	0.878	0.918
Number of features	30	18
Feature reduction	0%	40%

The overall comparison provides a summary of the Gradient Boosting classification performance vs feature dimensionality. The results show the positive effect on accuracy from 88.1% to 92.1% and the F1 score from 87.8% to 91.8% respectively. Meanwhile the features were reduced from 30 to 18. These findings would suggest a fewer number of features and an optimized model configuration would be able to predict with a higher degree of accuracy using fewer input variables. Other metrics that should be taken into account include class-specific metrics, validation variability, and computational requirements.

Discussion

In the present study, the importance of feature selection and model optimization in enhancing the classification of complex data was explored. The classification performance largely depends on what type of machine learning algorithm is used, the relevance of the input features and the configuration of the hyperparameters of the model. Some datasets can have noise or noise-like features,

or features that are irrelevant or redundant, that could make them difficult to process by algorithms that are looking for meaning. These issues can be addressed with feature selection, which can compress down unnecessary dimensions and keep informative variables. According to Pashaei et al., (2022) feature selection can influence predictive performance, minimize computational expenses and better understand the data. In a similar way, Ali (2020) studied the various feature selection approaches and pointed out their role in building an effective classification model. Based on the results of these studies, there is a good basis to study feature selection as an important criterion in the classification process.

Another key factor that can affect the classification performance is model optimization. Machine learning algorithms rely on hyperparameters to define the manner in which the learning is performed; improperly configured parameters can reduce the ability of the machine to predict or possibly lead to overfitting. Yazdinejad et al., (2023) showed that random search could be an efficient strategy to identify proper hyperparameter settings without trying all the possible combinations. In this current research context, the optimization of Logistic Regression, Support Vector Machine, Random Forest and Gradient Boosting aimed to find the optimal configuration to achieve the trade-off of good prediction power and generalization. A comparison of default and optimized models can then determine if there are benefits to having a well-tuned model. The amount of improvement, however, may differ across the algorithms as the change in hyperparameters and structure of the input data may cause different responses from various algorithms (Rouhi et al., 2020).

The combination of feature selection and model optimization is especially noteworthy since it treats complementary issues of model development. Feature selection is the process of selecting the variables to be fed into the algorithm while optimization is the method of learning from the variables. Irrelevant features can be eliminated to make the classification simpler, and proper settings of hyperparameters can help to make the model better at detecting patterns. However, the amalgam treatment is not guaranteed to be successful when used together. Features that seem to be non-important on their own may contain informative content when combined with other features, and too small a feature set may lose the important information that could be predictive. Thus, to evaluate the effectiveness of feature selection, one has to compare the models trained with the original features with the models trained from those features that are selected. In addition, it is required to separate the effects of feature selection and model optimization from their interaction effect through separate experiments (Demir et al., 2022).

Accuracies, precisions, recalls, and F1-score are all examples of complementary measures that should be used to evaluate the performance of classification. The accuracy is the percentage of observations that are correctly classified, while precision and recall are further details regarding the positive predictions and identification of positive observations. The F1-score is a single measure that incorporates the precision and recall, and is useful in situations where both are significant. Jiao et al., (2023) talked about the selection of suitable performance measures based on the features of a classification problem. However, it cannot be concluded from the improvement of the accuracy, particularly in data sets with imbalanced classes. Confusion matrices and class-specific performance metrics can give a deeper insight into the performance of the model. This is a multi-dimensional assessment that can help establish if the integrated approach yields better prediction than just majority class.

It is also important to have reliable validation when comparing baseline with optimized models. The performance estimates obtained in a feature selection or hyperparameter optimization process that relies on information from the test set might be too optimistic. Moosavi et al., (2024) showed that model selection using cross validation can be biased if the same cross validation is utilized for model selection and evaluation. Feature selection and hyperparameter tuning, therefore, must be done during training, and evaluation of final performance should be done on an independent test set or on an appropriate nested cross-validation procedure. If the same method were used for all models, it would allow for a fair comparison of all models, and further facilitate confidence in the results. If the extent of differences between models is small, repeated validation and/or appropriate statistical comparison can also be used to identify consistency in the observed improvements (Zawbaa et al., 2016).

The proposed method also has implications on computational efficiency and model interpretation. The data-processing requirements and complexity of the final model can be reduced by limiting the number of input features, but feature selection and optimization might be more resource-intensive. This should be evaluated on the basis of training time, optimization time, prediction speed and number of features retained, in order to have an overall picture of the efficiency of the approach. A model with high accuracy prediction results using less variables could be easier to maintain and deploy, especially in such applications as high dimensional data. But the most appropriate model must be chosen based on the application of the intended use and not on accuracy. The cost of computing, reliability, the interpretability of the computed values and the consequences of classification errors should also be taken into account.

In general, the current research model helps to examine the feature selection and model optimization as supportive actions for investigating classification performance. The theoretical support for feature selection, systematic hyperparameter search, selection of suitable evaluation metrics, and reliable model validation come from previous research. However, it is not possible to assume the effectiveness of the integrated approach in advance; it has to be proven in practice by experimental results. Future studies might include a comparison of feature selection and optimization techniques for multiple datasets, a study on the stability of the extracted features and a study on the performance of the models with various noises and imbalanced classes. These studies might shed light on the trade-off between feature selection and model optimization and how they can lead to a meaningful improvement in the classification accuracy, generalization and computational speed of the model (Othman et al., 2024).

Conclusion

This study explores how feature selection and model optimization can improve the classification of complex datasets. Feature selection helps identify the most useful variables and remove unnecessary features, making the learning process more efficient. Model optimization focuses on selecting appropriate hyperparameter values to improve the model's prediction accuracy and overall performance. These techniques can be combined to develop systematic machine learning models that may attain better classification outcome than baseline models. Measuring the accuracy, precision, recall and F1 score of algorithms like Logistic Regression, Support Vector Machine, Random Forest and Gradient Booster can give us an all-round picture of the algorithms. The study also emphasizes the need to take into account the aspect of computational efficiency, model complexity, and generalization when choosing the classification method.

To summarize, the proposed framework of feature selection and model optimization is a promising approach for tackling the issues related to complex and high-dimensional datasets. The effectiveness of this method, however, relies on the nature of the data, the selection method, the optimization method and the reliability of the evaluation method. Future studies may aim to test the proposed framework on additional datasets, evaluate other feature selection and hyperparameter optimization methods, and employ sound validation methods to evaluate the generalization capabilities. These investigations may be useful in determining when there are meaningful improvements in classification performance, in computational efficiency, and in the reliability of the model using the integrated approach. The final conclusions about the amount of these improvements will be made from the verified results of experiments conducted on the actual data set.

Future Implications

Future studies can improve the proposed framework by testing different feature selection methods, hyperparameter tuning techniques, and advanced machine learning algorithms. These approaches can be evaluated on diverse datasets with large numbers of features to assess their effectiveness in handling complex classification tasks. Automated feature selection, Bayesian optimization, ensemble learning, and explainable artificial intelligence are potential avenues for researchers to pursue in order to enhance the classification accuracy without sacrificing model explainability and computational efficiency. Additional research could also be conducted on the framework with larger amounts of data, cross validation, and independent test data to determine the reliability and generalizability of the results. The method could be useful in other sectors, like health care, education, finances, and cybersecurity, to determine informative predictors and make better data-driven decisions. Future studies could also explore feature stability, impact of noise and imbalance in data and benefits-detriments of predictive performance and computational complexity. These advances could lead to more powerful, effective and meaningful classification systems, able to solve increasingly complex real-world problems.

References

1. Sen, S., Saha, S., Chatterjee, S., Mirjalili, S., & Sarkar, R. (2021). A bi-stage feature selection approach for COVID-19 prediction using chest CT images. *Applied Intelligence*, 51(12), 8985-9000.
2. Probst, P., Wright, M. N., & Boulesteix, A. L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 9(3), e1301.
3. Cawley, G. C., & Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, 11, 2079-2107.
4. Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & electrical engineering*, 40(1), 16-28.

5. Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6), 1-45.
6. Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).
7. Bynum, M. L., Hackebeil, G. A., Hart, W. E., Laird, C. D., Nicholson, B. L., Siirola, J. D., ... & Woodruff, D. L. (2021). *Pyomo-optimization modeling in python* (Vol. 67, No. s 32). Berlin/Heidelberg, Germany: Springer.
8. Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*.
9. Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1), 6.
10. Georganos, S., Grippa, T., Vanhuysse, S., Lennert, M., Shimoni, M., Kalogirou, S., & Wolff, E. (2018). Less is more: Optimizing classification performance through feature selection in a very-high-resolution remote sensing object-based urban application. *GIScience & remote sensing*, 55(2), 221-242.
11. Mamdouh Farghaly, H., & Abd El-Hafeez, T. (2023). A high-quality feature selection method based on frequent and correlated items for text classification. *Soft Computing*, 27(16), 11259-11274.
12. Fernández, J. C., Carbonero, M., Gutiérrez, P. A., & Hervás-Martínez, C. (2019). Multi-objective evolutionary optimization using the relationship between F 1 and accuracy metrics in classification tasks. *Applied Intelligence*, 49(9), 3447-3463.
13. Bian, K., & Priyadarshi, R. (2024). Machine learning optimization techniques: a survey, classification, challenges, and future research issues. *Archives of Computational Methods in Engineering*, 31(7), 4209-4233.
14. Shashaani, S., & Vahdat, K. (2023). Improved feature selection with simulation optimization: S. Shashaani, K. Vahdat. *Optimization and Engineering*, 24(2), 1183-1223.
15. El-Hasnony, I. M., Barakat, S. I., Elhoseny, M., & Mostafa, R. R. (2020). Improved feature selection model for big data analytics. *IEEE Access*, 8, 66989-67004.
16. Asha, M. L., & Johnpeter, T. (2025). Optimized ensemble learning framework with enhanced feature selection for efficient and accurate classification. *SN Computer Science*, 6(6), 669.
17. Abiodun, E. O., Alabdulatif, A., Abiodun, O. I., Alawida, M., Alabdulatif, A., & Alkhalwaldeh, R. S. (2021). A systematic review of emerging feature selection optimization methods for optimal text classification: the present state and prospective opportunities. *Neural Computing and Applications*, 33(22), 15091-15118.
18. Sahebi, G., Movahedi, P., Ebrahimi, M., Pahikkala, T., Plosila, J., & Tenhunen, H. (2020). GeFeS: A generalized wrapper feature selection approach for optimizing classification performance. *Computers in biology and medicine*, 125, 103974.
19. Georganos, S., Grippa, T., Vanhuysse, S., Lennert, M., Shimoni, M., Kalogirou, S., & Wolff, E. (2018). Less is more: Optimizing classification performance through feature selection in a very-high-resolution remote sensing object-based urban application. *GIScience & remote sensing*, 55(2), 221-242.
20. Chen, R. C., Dewi, C., Huang, S. W., & Caraka, R. E. (2020). Selecting critical features for data classification based on machine learning methods. *Journal of Big Data*, 7(1), 52.
21. Feurer, M., & Hutter, F. (2019). Hyperparameter optimization. In *Automated machine learning: Methods, systems, challenges* (pp. 3-33). Cham: Springer International Publishing.
22. Hossain, M. R., & Timmer, D. (2021). Machine learning model optimization with hyper parameter tuning approach. *Global Journal of Computer Science and Technology: D Neural & Artificial Intelligence*.
23. Ballabio, D., Grisoni, F., & Todeschini, R. (2018). Multivariate comparison of classification performance measures. *Chemometrics and Intelligent Laboratory Systems*, 174, 33-44.

24. Cao, C., Chicco, D., & Hoffman, M. M. (2020). The MCC-F1 curve: a performance evaluation technique for binary classification. *arXiv preprint arXiv:2006.11278*.
25. Zhang, Y., & Wang, Z. (2023). Feature engineering and model optimization based classification method for network intrusion detection. *Applied Sciences*, 13(16), 9363.
26. Qaraad, M., Amjad, S., Manhrawy, I. I., Fathi, H., Hassan, B. A., & El Kafrawy, P. (2021). A hybrid feature selection optimization model for high dimension data classification. *IEEE Access*, 9, 42884-42895.
27. Tamimi, E., Ebadi, H., & Kiani, A. (2017). Evaluation of different metaheuristic optimization algorithms in feature selection and parameter determination in SVM classification. *Arabian Journal of Geosciences*, 10(22), 478.
28. Sharma, A., & Mishra, P. K. (2022). Performance analysis of machine learning based optimized feature selection approaches for breast cancer diagnosis. *International Journal of Information Technology*, 14(4), 1949-1960.
29. Amiriebrahimabadi, M., & Mansouri, N. (2024). A comprehensive survey of feature selection techniques based on whale optimization algorithm. *Multimedia Tools and Applications*, 83(16), 47775-47846.
30. Akinola, O. O., Ezugwu, A. E., Agushaka, J. O., Zitar, R. A., & Abualigah, L. (2022). Multiclass feature selection with metaheuristic optimization algorithms: a review. *Neural Computing and Applications*, 34(22), 19751-19790.
31. Pashaei, E., & Pashaei, E. (2022). An efficient binary chimp optimization algorithm for feature selection in biomedical data classification. *Neural Computing and Applications*, 34(8), 6427-6451.
32. Ali, A. (2020). An efficient model for data classification based on SVM grid parameter optimization and PSO feature weight selection. *International Journal of Integrated Engineering*.
33. Yazdinejad, A., Dehghantanha, A., Parizi, R. M., & Epiphaniou, G. (2023). An optimized fuzzy deep learning model for data classification based on NSGA-II. *Neurocomputing*, 522, 116-128.
34. Rouhi, A., & Nezamabadi-Pour, H. (2020). Feature selection in high-dimensional data. In *Optimization, learning, and control for interdependent complex networks* (pp. 85-128). Cham: Springer International Publishing.
35. Demir, S., & Şahin, E. K. (2022). Liquefaction prediction with robust machine learning algorithms (SVM, RF, and XGBoost) supported by genetic algorithm-based feature selection and parameter optimization from the perspective of data processing. *Environmental Earth Sciences*, 81(18), 459.
36. Jiao, R., Nguyen, B. H., Xue, B., & Zhang, M. (2023). A survey on evolutionary multiobjective feature selection in classification: approaches, applications, and challenges. *IEEE Transactions on Evolutionary Computation*, 28(4), 1156-1176.
37. Moosavi, S. K. R., Saadat, A., Abaid, Z., Ni, W., Li, K., & Guizani, M. (2024). Feature selection based on dataset variance optimization using hybrid sine cosine–firehawk algorithm (hscfha). *Future Generation Computer Systems*, 155, 272-286.
38. Zawbaa, H. M., Emary, E., & Grosan, C. (2016). Feature selection via chaotic antlion optimization. *PloS one*, 11(3), e0150652.
39. Li, J., Othman, M. S., Chen, H., & Yusuf, L. M. (2024). Optimizing IoT intrusion detection system: feature selection versus feature extraction in machine learning. *Journal of Big Data*, 11(1), 36.



2026 by the authors; Journal of Global Social Transformation (JGST). This is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).