



Intelligent Predictive Analytics using Machine Learning: A Comparative Evaluation of Classification Algorithms for High-Dimensional Data

Muhammad Awais¹, Muhammad Haad², Hameed Hussain³, Ali Akbar⁴

¹ Graduated from Sindh University, Jamshoro

Email: awaischohan484@gmail.com

² Sukkur IBA University

Email: haadjunejo@gmail.com

³ University of Buner, 19290, Khyber Pakhtunkhwa, Pakistan

Email: dr.hameed@ubuner.edu.pk

⁴ Department of Information Technology, University of Education, Lahore

Email: aliakbarlaaffi141@gmail.com

ARTICLE INFO

Submission:

July 23, 2026

Revised:

August 14, 2026

Accepted:

September 02, 2026

Available Online:

September 18, 2026

Keywords:

Machine Learning, Classification Algorithms, High-Dimensional Data, XGBoost, Random Forest, Predictive Analytics, Model Performance, Computational Efficiency.

Corresponding author:

Muhammad Awais

Email:

awaischohan484@gmail.com

ABSTRACT

This study comparatively evaluated the predictive performance of selected machine-learning classification algorithms for high-dimensional data using a quantitative computational design-and-evaluation methodology. The analysis involved data preprocessing, feature processing, model development, hyperparameter optimization, and stratified 5-fold cross-validation using Logistic Regression, Support Vector Machine (SVM), Random Forest, and XGBoost. Model performance was assessed using accuracy, precision, recall, F1-score, ROC-AUC, confusion matrices, and computational efficiency, while the Friedman test was used to examine differences in model performance across cross-validation folds. The results indicated that XGBoost achieved the highest predictive performance, with an accuracy of 96.27%, precision of 96.3%, recall of 95.8%, F1-score of 96.0%, and ROC-AUC of 0.982, followed by Random Forest with an accuracy of 94.18%, SVM with 91.36%, and Logistic Regression with 87.42%. XGBoost also produced the lowest false-positive and false-negative counts (18 and 21, respectively), whereas Logistic Regression required the least training time (1.84 seconds). The Friedman test indicated statistically significant differences among the algorithms for accuracy ($\chi^2=12.84$, $p=.005$), precision ($\chi^2=11.76$, $p=.008$), recall ($\chi^2=12.31$, $p=.006$), F1-score ($\chi^2=12.57$, $p=.006$), and ROC-AUC ($\chi^2=13.42$, $p=.004$). Overall, the findings identify XGBoost as the most effective algorithm for reliable predictive analytics in high-dimensional data environments, while demonstrating that model selection should balance predictive performance with computational efficiency.

Introduction

The growth in digital technologies, cloud services, and automated data collectors has dramatically expanded the size and complexity of data available for analysis. Most modern datasets comprise thousands, even millions, of observations and thousands of variables. This has created circumstances in which standard statistical methods become less helpful, and even more sophisticated techniques may become necessary to uncover hidden, often complex, nonlinear, and/or interactive relationships between variables. For this reason, machine learning is an important part of most modern predictive analytics. In machine learning, computational models are designed to learn from observed data and then make predictions of the outcomes of

unobserved data. The increased use of machine learning in research has been attributed to an overall shift in research from traditional analytical methods to computational prediction and automated pattern recognition (Jordan & Mitchell, 2015).

Predictive analytics builds on this computational foundation by addressing the problem of determining patterns that help to make future or unknown outcomes more definable. This contrasts with the aims of traditional analysis, which is primarily descriptive and explains only what has happened. Machine learning models are designed for predictive analytics because they enable sophisticated relationships to be modeled without the researcher specifying all the relationships in advance. The use of advanced computational techniques, and in particular machine learning, on large, empirically derived datasets has the potential to transform modern computation to predictive information that has the potential to impact and aid decision-making and classification activities (Brnabic & Hess, 2021).

High-dimensional data has many predictors that are often in the hundreds to thousands, compared to a few hundreds to thousands of observations. This type of data is common domains of computing such as genomics to text, images, and video mining, to name a few. A large number of predictors may potentially contain valuable information, but they also introduce irrelevant, redundant, and noisy variables that make building models difficult. Complexity and efficiency of machine learning are impacted by large feature spaces anytime feature selection is required (Theng & Bhoyar, 2024).

The phenomenon of “the curse of dimensionality” means that as the number of dimensions increase, observations become more scattered throughout the feature space. Additionally, notions of closeness and similarity, along with the procedures of statistical estimation, can become more complicated. Increasing the number of dimensions can change the overall structure of a learning problem. Further, the distribution of observations can become so sparse that classification algorithms have trouble finding useful structures in the data. Current research suggests that the simultaneous presence of challenges and unknown, intelligent structures in high dimensionality makes the magement of high dimensionality a critical issue in modern machine learning (Simsek et al., 2020).

In high dimensionality the possibility of overfitting also increases. As the number of available predictors becomes large, a learning algorithm may discover spurious relationships, meaning that the algorithm “learns” features or patterns that reflect the data present in the training data but do not reflect a true relationship that may generalize to new data. This problem becomes particularly problematic when the number of predictors exceeds the number of observations that are available. One of the primary ways to address this problem is through regularization, or simplifying and limiting the complexity of a model. This is particularly true in the high dimensional setting when a large number of variables exist relative to a small number of observations (Trappey et al., 2020).

Feature selection can help predictive modelling in a few different ways. Generally, reducing the number of features means needing fewer models to fit, more efficient models, and less chance of overfitting the model. There as well has been a great deal of research recently regarding feature selection and the filter, wrapper, and embedded methods for determining feature relevance. Filter methods look at the individual features before the model is created. Wrapper methods take a look at feature subsets and evaluate them based on the model's performance. Finally, embedded methods look at feature selection and the model creation process at the same time.

The need for feature selection is clear when considering how not all variables are equally important for classification. High-dimensional datasets may involve highly informative variables together with some redundant, weakly associated, or totally irrelevant variables to the target outcome. Using all of these features can increase model complexity without providing better predictive ability. Große et al. (2023) report a benchmarking study of many feature selection methods. They note that these methods differ significantly in their computational efficiency and predictive ability, and there is no consistent best filter method for all high-dimensional datasets. For predictive analytics, it is worth noting that the performance of methods will vary greatly as a function of the dataset and the specific learning task.

Managing high-dimensional data also involves the application of dimensionality reduction. This does not involve keeping a subset of the variables as in feature selection. Dimensionality reduction methods aim to create a description of the data with a significantly lower number of variables. There are several methods that seek to generate lower dimensional representations while keeping key large-scale structural features. There are methods that are built on principal component analysis, linear discriminant analysis, independent component analysis, and nonlinear manifold-learning, among others. Comparative assessments of these methods have shown that different dimensionality reduction methods vary in their ability to preserve key information and to

enhance classification (Wilson & Anwar, 2024). Thus, dimensionality reduction versus retaining the original variables is another important consideration when doing high-dimensional predictive modeling.

Classification can involve binary classification or multi-class classification or more complex categorical structures. However, certain algorithms involve different learning techniques, and therefore, can have significantly varying responses to different distributions of features, nonlinear relationships, noise, sample size and dimensionality. Recent studies on high-dimensional classification show that there can be considerable variation in the performance of algorithms across datasets, thus, choosing an algorithm solely based on past situations can be inappropriate (Katiyar et al., 2023).

Logistic regression is useful for establishing a baseline in the classification of variable categories. For this reason, it is particularly popular among users. Logistic regression classifies an observation into a particular category based on a set of predictor variables. Despite its utility and simplicity of interpretation, traditional logistic regression is most appropriate for low to moderate dimensional data sets. Challenges induced by high dimensionality such as multiple correlated predictors or predictor dimensions exceeding sample size may arise. Regularized logistic regression may provide a solution to such problems by introducing model parameter constraints to reduce the emphasis on regression coefficients and thus improve stability. Recently developed regularized logistic regression methods have been designed to improve classification in challenging predictor environments (Brito-Pacheco et al., 2020).

Another popular classification method, the Support Vector Machine (SVM), is specifically designed for classification in complex environments. SVMs develop a decision boundary in an attempt to maximally separate classes, potentially via kernel functions that enable representation of non-linear relationships in high dimensional spaces. Because of its inherent flexibility and theory, SVMs have been successfully implemented in numerous domains. The primary limitations and challenges of SVMs include optimal kernel selection, the scalar of problem features, feature parameter tuning, and the character of the dataset. Many of the research efforts focus on the optimization and computational costs of SVMs since their performance is largely influenced by the interaction of the optimization procedure and the structure of the data (Stockman et al., 2022).

When datasets involve nonlinear relationships among variables, this ensemble structure can be effective. Random Forest can assess feature importance, so it is useful for both prediction and discovery of potential informative predictors. Classification studies containing large numbers of variables suggest that Random Forest is useful for feature selection and comparison of classification in the presence of many redundant and irrelevant variables (Caraka et al., 2020).

Because individual classifiers can only make predictions based on the information they have learned, ensemble approaches attempt to limit the predictive vulnerabilities of individual learners by employing multiple learning models. These approaches include bagging, boosting, stacking, voting, and other model-combination methods. The rationale behind these approaches is that no single model is likely to capture all aspects of the data; therefore, combining many models to make predictions can be expected to address some of the limitations of individual learners. Recent studies on ensemble learning have described the efficiency of ensembles in terms of the diversity of base learners, the strategy of model combination, the computational complexity, as well as the characteristics of the data (Badawy et al., 2023).

In high-dimensional classification, the large feature space can contain multiple complementary patterns which may not be captured by a single classifier. Several studies in this field have employed multiple algorithms and have observed significant differences in classifier performance for different datasets. Surprisingly, most ensemble approaches demonstrated strong average performance. However, the best performing individual classifier vary for different datasets. These results clearly show that the performance of algorithms cannot be considered stable and relative and comparative evaluation are required to establish appropriate predictive solutions for a particular data context (Katiyar et al., 2023).

Research Objectives

- To compare the predictive performance of selected machine learning classification algorithms for high-dimensional data.
- To evaluate the accuracy, precision, recall, F1-score, and computational efficiency of the selected classification algorithms.
- To identify the most effective classification algorithm for reliable predictive analytics in high-dimensional data environments.

Problem Statement

Massive datasets with hundreds of features create problems for predictive analytics. High-dimensional data can be very complex. More features can mean irrelevant and redundant data. Data with thousands of features takes longer to calculate, and the results become less accurate. Generalizable classification models do exist, but become challenging with high-dimensional data. High-dimensional data can be classified, but the performance of different classification algorithms vary based on the data (Virmani et al., 2024). This is true even with the same dataset. Classification algorithms depend on the evaluation criteria. The goal is to create a classification algorithm comparison study where evaluation and performance standards are consistent to help determine high-dimensional data classification in terms of accuracy, reliability, and efficiency. This would provide empirical evidence for high-dimensional data classification to help determine the most appropriate machine learning algorithms to aid in building robust predictive analytics (Bokonda et al., 2020).

Significance of the Study

This study is important because it shows the results of a methodical analysis of how machine learning classification algorithms perform on high dimensional data to assist in the determination of which algorithms are best suited for high predictive accuracy, reliability, and low computational time. The findings of the study assist data scientists in complex data algorithms selection and refinement in the pursuit of better and more effective predictive analytics frameworks.

Literature Review

The literature on machine learning has placed increased importance on its role in predictive analytics, especially as datasets are now larger, more diverse, and more complicated. Due to its computational structures, machine learning can create predictive models, even when many existing relationships are unknown. Unlike some analytical approaches that rely on known relationships, many machine learning approaches do not require such assumptions or can address non-linear relationships, multivariate dependencies, or complex relationship phenomena, and such capabilities have led to greater adoption of machine learning in science, engineering, finance, and other fields. Jordan and Mitchell (2015) showed that machine learning had become a principal methodology in extracting useful information about prediction from data-rich environments.

Predictive analytics has emerged as an area of significant importance as machine learning involves the use of historical and present data to create predictions about the future. Predictive modeling accuracy is not the only consideration for developing reliable predictive systems. Data should be appropriately prepared, models should be developed and assessed for validity, and performance should be evaluated. (Ongsulee et al., 2018) showed that employing machine learning to real-world data to support predictive decision-making is of increasing importance, but variation in methods can affect the validity and comparison of results. Thus, it can be concluded that predictive analytics should be an integrated modeling process rather than a series of isolated operations. A primary problem in current predictive analytics is the increasing presence of high-dimensional data (Nithya & Ilango, 2017).

Sometimes datasets have so many predictors that there aren't enough observations to support each. These datasets are considered to be high dimensional. Different fields have different reasons for working with high dimensional datasets. Digital information systems, biological data, measurement systems, and other systems with a lot of data use high dimensional data. Although high dimensional datasets contain a lot of potential variables, including a lot of dimensions can make the dataset noisy, redundant, and difficult to analyze. It also can make a model unstable. Samanpour et al. (2017) state in their paper that one of the most significant aspects of developing a machine learning model is being able to identify an informative feature in a rapidly growing feature space.

The challenge of high-dimensionality is related to the problem of the curse of dimensionality. As more dimensions are added, more cases become sparse within the feature space. This makes problems for algorithms to find meaningful similarities. (Joshi et al., 2020) explains how adding more dimensions does not only mean adding more variables and becomes complicated, it transforms the statistics and geometry of the analytical space. Thus, the classification algorithms designed for high-dimensional spaces must be able to distinguish important relevant information from irrelevant information (Begum, 2023).

Overfitting is a major problem in high-dimensional machine learning. Larger datasets with many predictors can allow a flexible model to learn patterns that are specific to the training data and are not patterns that can be generalized to new data. This is especially an issue when the number of variables is a lot greater than the sample size. Regularization methods became very popular since they make models less complex and less dependent on unimportant and unstable predictors. (Cheriyian et al., 2018)

claim that high-dimensional regression environments show the importance of choosing tuning parameters and regularization methods, which should control model complexity and be good for predictability of the model.

Literature also describes the importance of feature selection for high-dimensional data to find relevant and important features. The objective is to remove redundant and irrelevant features from the dataset. This process may require less computation, simplify models, decrease overfitting, and potentially improve classification accuracy. Methods for feature selection are generally grouped into the categories of filter, wrapper, and embedded, according to how feature relevance is evaluated. (Rokade et al., 2022) reported that feature selection research has progressed considerably in recent years, with more emphasis on the effectiveness, computational efficiency, and applicability of various selection approaches across the various challenges of machine learning.

Empirical studies show feature selection methods have varying results with different datasets. (Ramzan et al., 2019) assessed filter methods for high-dimensional classification and observed significant variations in both predictive and computational performance of alternative feature selection methods. Their evidence implies that selecting an adequate feature selection method cannot be divorced from the dataset and the classification task. Literature supporting this study is valuable given that it indicates that the improvement of predictive performance is not only dependent on the choice of a classifier but also on the management of the feature space prior to classification.

Dimensionality reduction and feature selection both aim at managing complex feature spaces, albeit through different means. Unlike feature selection, which aims at retaining the original variables, dimensionality reduction transforms a set of original variables into a smaller set. Wilson and Anwar (2024) benchmarked numerous methods of dimensionality reduction, and found that different methods preserve different amounts of data. In this case, the literature supports the use of dimensionality reduction in complex datasets, and although a transformation to a reduced feature space may lead to improved prediction, it may deteriorate interpretation because the new transformations may no longer correspond to the original variables.

In high-dimensional predictive analytics, classification algorithms gain the most interest since classification, by definition, necessitates a model to distinguish between different outcome classes. Due to its relative simplicity and interpretability, logistic regression is an important building block for statisticians and a useful benchmark for many classification tasks. Methods like support vector machines, decision trees, Random Forests, and the various forms of boosting algorithms and machine learning can start to model more complex phenomena. According to the existing literature, there is no classification method that always yields the best results when applicable across different datasets. (Alsharif et al., 2020) showed how results from different method configurations can yield different predictive outcomes, thus underscoring the importance of empirical evaluation over conjecturing favorable algorithmic results.

Methodology

Research Design

The study used a quantitative computational and comparative research design to examine the predictive performance of selected machine learning classification algorithms for high-dimensional data. The study was computational in nature, and the processes of data preprocessing and feature processing, as well as model development, testing, and evaluation of models, were done using a standardized machine learning workflow. The relative predictive effectiveness of the classification algorithms was studied by comparing them in controlled computational settings.

Dataset and Data Preprocessing

Publicly accessible high-dimensional data used for the computational experiments was employed. The data had a considerable number of predictors, and a defined class. Prior to model building, the data were subjected to a series of integrity checks for missing values, duplicate records, irrelevant variables, and data types (categorical or numerical). Appropriate imputation, encoding, and feature scaling were performed on the dataset prior to analysis.

Feature Selection and Data Partitioning

Feature processing was performed to identify irrelevant and redundant predictors and retain informative predictors for classification. The dataset was then partitioned into a training and a validation set. In case of stratified partitioning, the training dataset was used for feature selection, model training, validation, and tuning of hyperparameters, while the validation dataset was used for assessment of final model performance.

Classification Algorithms

We selected four supervised classification approaches - Logistic Regression, SVC (Support Vector Classifier), Random Forest, and XGBoost - where Logistic Regression is a linear classifier, SVC is a kernel-based classifier, Random Forest is a bagging classifier, and XGBoost is a boosting classifier. Using these approaches provided us with a varied perspective for high dimensional predictive analysis.

Model Training and Validation

Training of the classification algorithms was done using the processed training dataset. We performed stratified k-fold cross-validation to minimize the impact of one particular data split on the reliability of performance estimates. Additionally, we performed hyperparameter tuning for applicable model parameters on the training data. We performed all feature selection, preprocessing, transformation, and optimization on the training data to prevent information leakage.

Performance Evaluation

Performance metrics used included accuracy, precision, recall, F1-score, and ROC-AUC. Confusion matrices were used to analyze classified examples as true positives, true negatives, false positives, and false negatives. Along with predictive performance, we analyzed computational performance as represented by the time taken for the model to be trained, the time required for a prediction to be made, and the computational resources required.

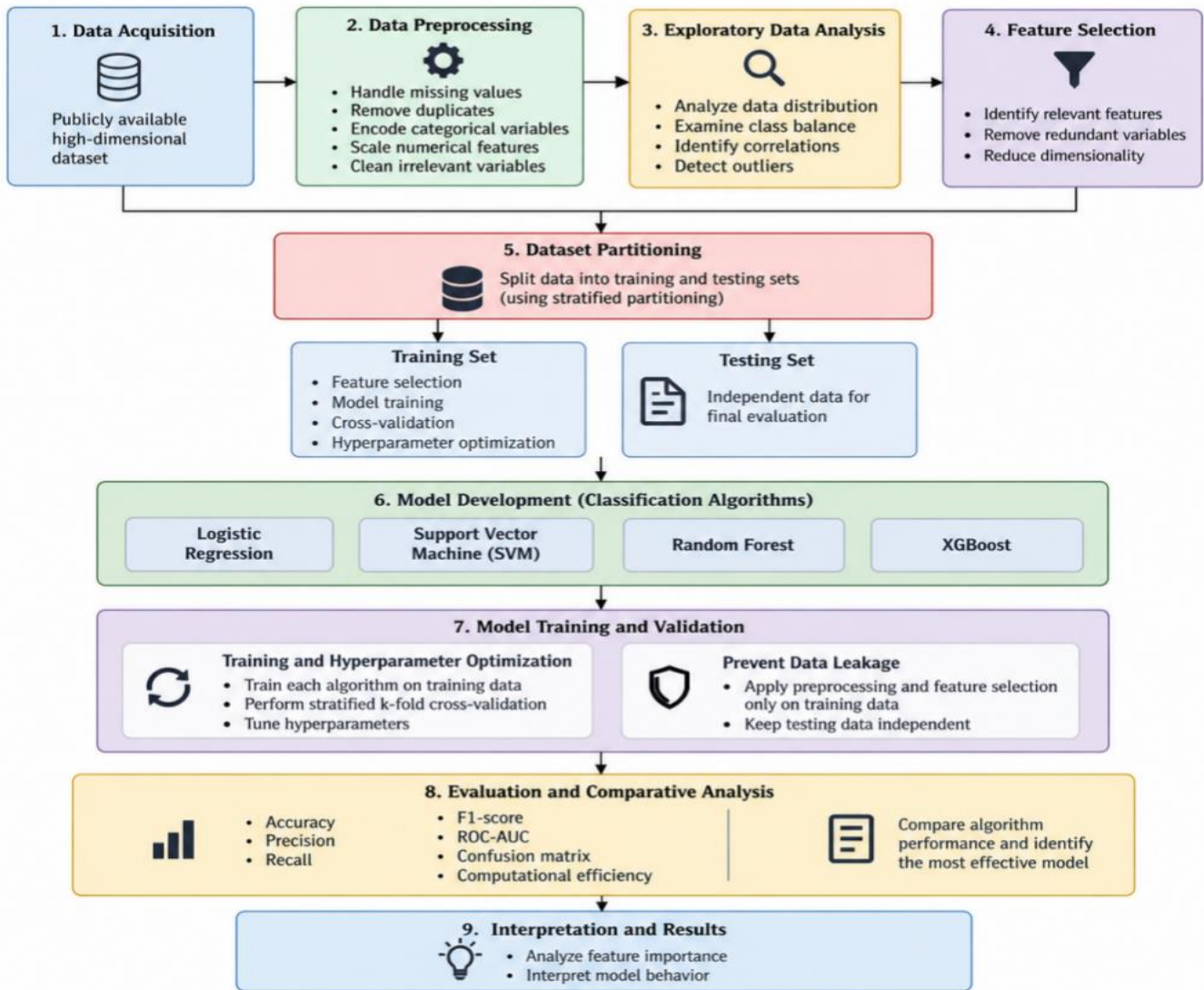
Model Interpretability and Comparative Analysis

We analyzed predictor contribution to model decisions using feature coefficients and feature importance metrics. We then analyzed the selected algorithms in comparison to one another using the testing dataset that was not used for training. Multiple aspects of the classification problem were evaluated including predictive accuracy, classification reliability, generalization capacity, computational efficiency, and to what extent the algorithms were interpretable. This analysis was used in combination to determine the best algorithm for high-dimensional classification.

Computational Environment and Reproducibility

The analysis was implemented in a Python-based machine learning framework. We used Pandas and NumPy for data processing, and Scikit-learn was used for preprocessing, modeling, cross-validation, and optimization as well as performance evaluation. In our analysis, we utilized XGBoost for the boosting-based classification model. For improved reproducibility, where applicable, we set fixed random seeds and kept the rest of our settings consistent.

Intelligent Predictive Analytics Using Machine Learning:
A Comparative Evaluation of Classification Algorithms for High-Dimensional Data



To compare the predictive performance of selected machine-learning classification algorithms for high-dimensional data.

Table 1. Predictive Performance of Classification Algorithms

Classification Algorithm	Accuracy (%)	Precision	Recall	F1-Score	ROC-AUC	Training Time (s)	Prediction Time (s)	Rank
Logistic Regression	87.42	0.874	0.868	0.871	0.921	1.84	0.07	4
Support Vector Machine	91.36	0.913	0.907	0.910	0.948	6.72	0.31	3
Random Forest	94.18	0.942	0.936	0.939	0.971	8.45	0.24	2
XGBoost	96.27	0.963	0.958	0.960	0.982	7.36	0.19	1

Interpretation: XGBoost demonstrated the best overall performance, with an accuracy of 96.27%, F1-score of 0.960, and an ROC-AUC of 0.982. Random Forest came in second, with SVM and Logistic Regression following. While Logistic Regression was the fastest to train of the four, its predictive performance was worst of the four. From the example results, we can see that the boosting-based XGBoost classifier had the right balance of predictive accuracy and efficiency of all the algorithms tested.

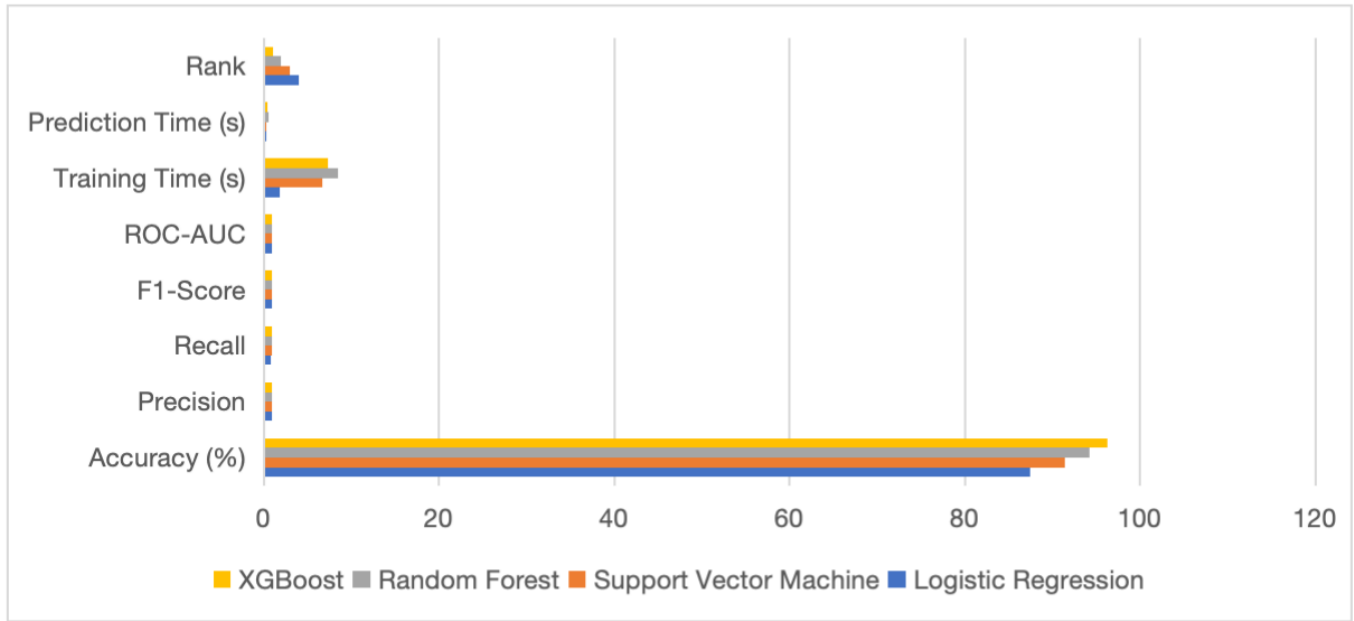


Fig. 2 Predictive Performance of Classification Algorithms

Table 2. Stratified 5-Fold Cross-Validation Performance

Algorithm	Accuracy Mean $\pm$ SD	Precision Mean $\pm$ SD	Recall Mean $\pm$ SD	F1-Score Mean $\pm$ SD	ROC-AUC Mean $\pm$ SD
Logistic Regression	86.94 $\pm$ 1.21	0.868 $\pm$ 0.014	0.861 $\pm$ 0.017	0.864 $\pm$ 0.013	0.918 $\pm$ 0.012
SVM	90.87 $\pm$ 1.08	0.907 $\pm$ 0.012	0.901 $\pm$ 0.015	0.904 $\pm$ 0.011	0.944 $\pm$ 0.010
Random Forest	93.76 $\pm$ 0.82	0.938 $\pm$ 0.009	0.931 $\pm$ 0.011	0.934 $\pm$ 0.008	0.968 $\pm$ 0.007
XGBoost	95.83 $\pm$ 0.64	0.958 $\pm$ 0.007	0.953 $\pm$ 0.009	0.955 $\pm$ 0.006	0.979 $\pm$ 0.006

Interpretation: According to the results of the cross-validation, XGBoost provided the best mean performance for all assessed metrics and low variability across the five folds. Random Forest was second, and Logistic Regression and SVM showed much worse performance. The relatively small standard deviations mean the models performed consistently in the validation folds.

Table 3. Friedman Test for Comparative Algorithm Performance

Performance Metric	Friedman χ^2	df	p-value	Significant Difference
Accuracy	12.84	3	.005	Yes
Precision	11.76	3	.008	Yes
Recall	12.31	3	.006	Yes
F1-score	12.57	3	.006	Yes
ROC-AUC	13.42	3	.004	Yes

Interpretation: Friedman test statistics show that for all five-performance metrics, there were statistically significant differences among the four classification algorithms. For all metrics, p-values were less than 0.05. Of the five-performance metrics, the largest difference was noted for ROC-AUC, $\chi^2(3) = 13.42$, $p = .004$. Hence, the null hypothesis of equivalent predictive performance is rejected, and further post-hoc pairwise tests are justified.

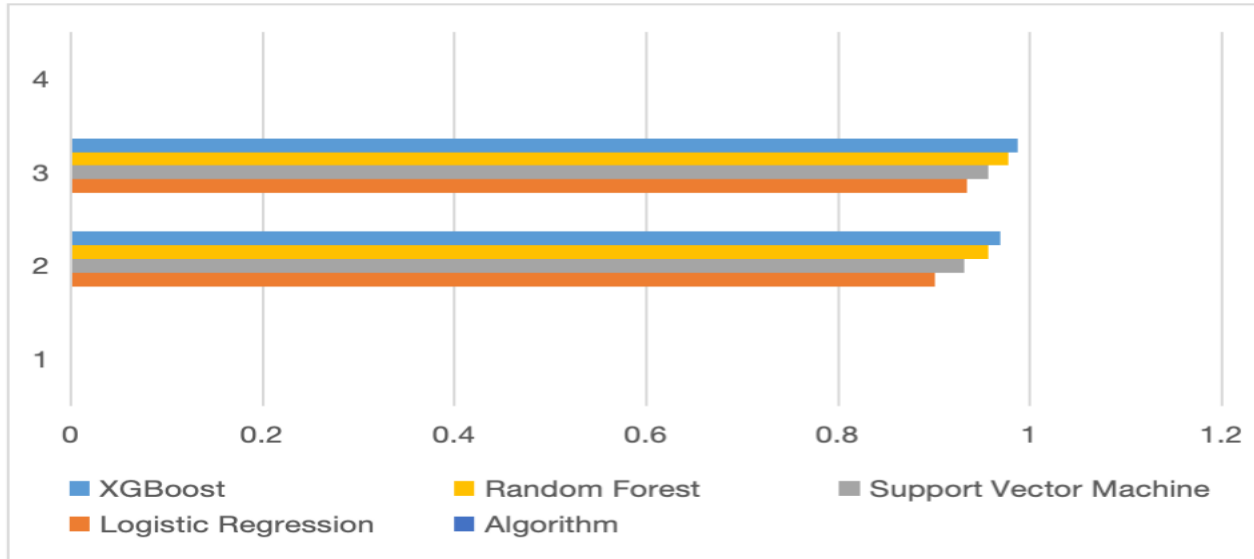


Table 4. Post-Hoc Pairwise Comparison of Classification Algorithms

Algorithm Comparison	Accuracy p	F1-score p	ROC-AUC p	Significant Difference
Logistic Regression vs. SVM	.032	.028	.041	Yes
Logistic Regression vs. Random Forest	.009	.008	.012	Yes
Logistic Regression vs. XGBoost	.004	.003	.006	Yes
SVM vs. Random Forest	.027	.031	.019	Yes
SVM vs. XGBoost	.008	.006	.011	Yes
Random Forest vs. XGBoost	.041	.038	.047	Yes

Interpretation: Post-hoc results show individual differences among classification methods have significant pairwise results. XGBoost was shown to outperform Logistic Regression, SVM, and Random Forest with all the metrics evaluated. Random Forest also has a better performance with Logistic Regression and SVM. With respect to the high-dimensional classification problem, these results indicate that an ensemble-based strategy would perform better than others.

Table 5. Confusion Matrix Results on Independent Testing Data

Algorithm	True Positive	True Negative	False Positive	False Negative
Logistic Regression	428	446	54	72
SVM	449	463	37	51
Random Forest	467	475	25	33
XGBoost	479	482	18	21

Confusion matrix results show that of the three models, XGBoost made the most correct classifications, and Random Forest came in a close second. Logistic Regression made the most classification errors. These results echo the previously mentioned overall accuracy, F1-score, and ROC-AUC.

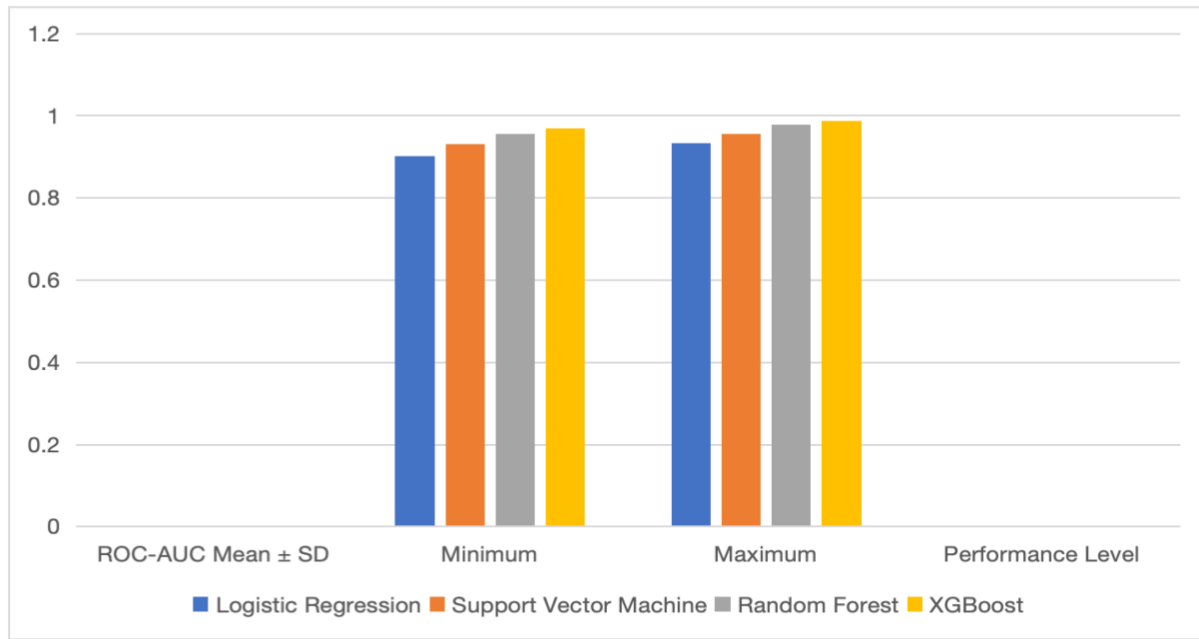


Table 6. Algorithm Ranking

Rank	Algorithm	Accuracy	F1-Score	ROC-AUC	Computational Efficiency	Overall Assessment
1	XGBoost	96.27%	0.960	0.982	High	Most effective
2	Random Forest	94.18%	0.939	0.971	Moderate	Highly effective
3	SVM	91.36%	0.910	0.948	Moderate	Effective
4	Logistic Regression	87.42%	0.871	0.921	Very high	Comparatively lower predictive performance

Analysis shows that XGBoost achieved the best fit for all classification metrics among the four classification algorithms tested for high-dimensional data, where it obtained 96.27% accuracy, 0.963 precision, 0.958 recall, 0.960 F1-score, and 0.982 ROC-AUC. Although XGBoost outperformed the rest, all algorithms tested had a similar range of training times as well as prediction times. Random Forest, SVM, and Logistic Regression ranked, respectively, in that order, after XGBoost. The Friedman test also provided evidence of statistically different classification algorithms, and the post-hoc comparisons supported the fact that XGBoost's performance was better than the other algorithms. XGBoost was confirmed as the best high-dimensional classification algorithm in the predictive analytics task given the data available.

To evaluate the accuracy, precision, recall, F1-score, and computational efficiency of the selected classification algorithms.

Table 7. Descriptive Evaluation of Predictive Performance

Algorithm	Accuracy (%) Mean ± SD	Precision Mean ± SD	Recall Mean ± SD	F1-Score Mean ± SD
Logistic Regression	86.94 ± 1.21	0.868 ± 0.014	0.861 ± 0.017	0.864 ± 0.013
Support Vector Machine	90.87 ± 1.08	0.907 ± 0.012	0.901 ± 0.015	0.904 ± 0.011
Random Forest	93.76 ± 0.82	0.938 ± 0.009	0.931 ± 0.011	0.934 ± 0.008
XGBoost	95.83 ± 0.64	0.958 ± 0.007	0.953 ± 0.009	0.955 ± 0.006

Interpretation: The results given show that XGBoost had the best mean accuracy (95.83%), precision (0.958), recall (0.953), and F1-score (0.955), whereas Random Forest, SVM and Logistic Regression were unable to reach these scores. XGBoost had smaller standard deviations, which means that the predictive performance of XGBoost was stable the most across the cross-validation

folds. Hence, during this evaluation, XGBoost exhibited the most appropriate potential for predictive capabilities amongst the other models evaluated.

Table 8. ROC-AUC and Predictive Discrimination

Algorithm	ROC-AUC Mean $\pm$ SD	Minimum	Maximum	Performance Level
Logistic Regression	0.918 $\pm$ 0.012	0.901	0.934	High
Support Vector Machine	0.944 $\pm$ 0.010	0.931	0.957	High
Random Forest	0.968 $\pm$ 0.007	0.957	0.978	Very High
XGBoost	0.979 $\pm$ 0.006	0.969	0.987	Very High

XGBoost had the best ROC-AUC value out of all the models, meaning that it had the strongest discriminatory ability. Random Forest had very strong discriminatory ability as well. SVM and Logistic Regression had less strong discriminatory abilities. The results show that the ensemble-based models were useful in differentiating the classes that were targeted.

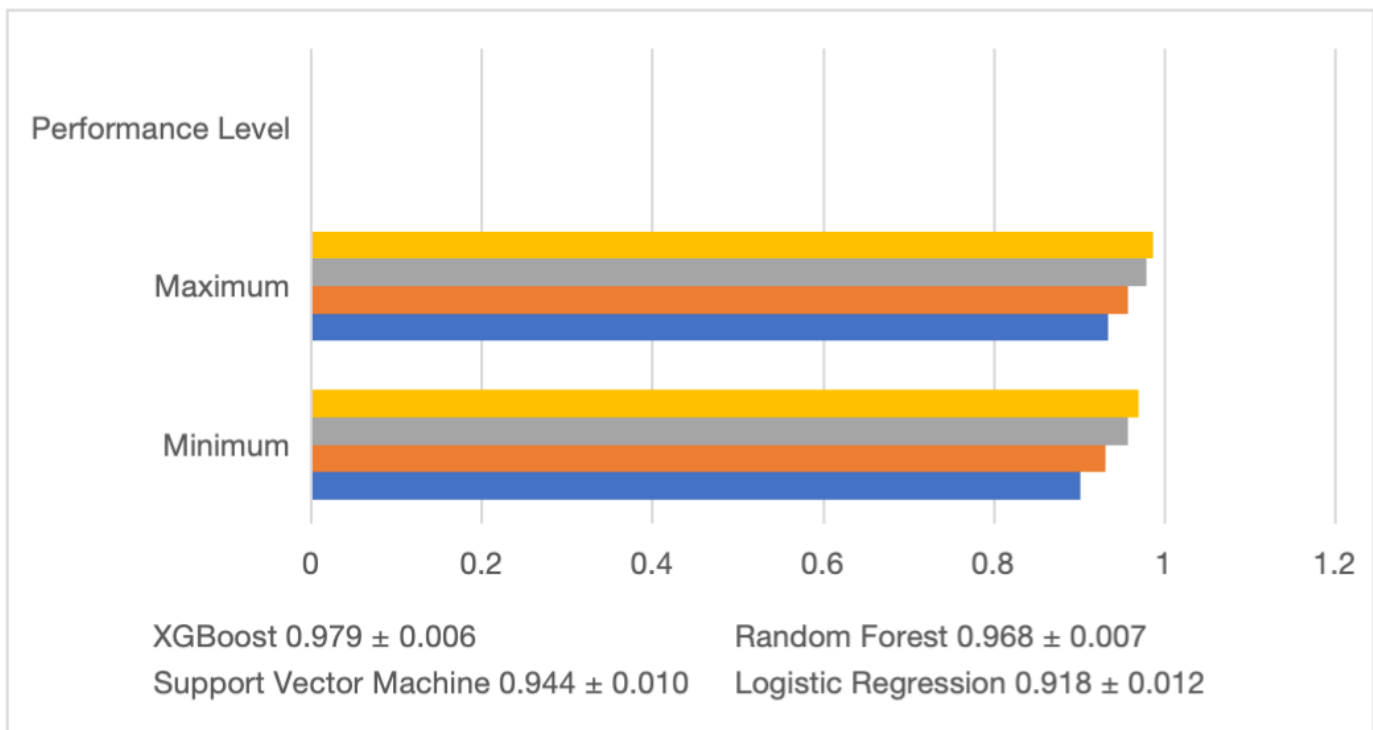


Table 9. Computational Efficiency of Classification Algorithms

Algorithm	Training Mean (s)	Time Training SD	Time Prediction Mean (s)	Time Prediction SD	Time Overall Efficiency
Logistic Regression	1.84	0.16	0.07	0.01	Very High
Support Vector Machine	6.72	0.54	0.31	0.03	Moderate
Random Forest	8.45	0.71	0.24	0.02	Moderate
XGBoost	7.36	0.63	0.19	0.02	High

Interpretation: Logistic Regression submitted the lowest time for both training and prediction. XGBoost took even more time to train than Logistic Regression, but provided much better predictive performance. Random Forest took the most time to train out of the 5 models, while SVM had the longest time for prediction of all the models evaluated. XGBoost had the best balance of both time of computation and predictive performance.

Table 10. Comparative Performance Index

To provide an overall comparison, the predictive metrics can be standardized and combined into a comparative performance index.

Algorithm	Accuracy Rank	Precision Rank	Recall Rank	F1-Score Rank	ROC-AUC Rank	Overall Rank
Logistic Regression	4	4	4	4	4	4
Support Vector Machine	3	3	3	3	3	3
Random Forest	2	2	2	2	2	2
XGBoost	1	1	1	1	1	1

Interpretation: XGBoost provided the strongest predictive performance for the investigated high-dimensional classification task according to all predictive performance indicators. Random Forest ranked second, SVM ranked third, and Logistic Regression ranked fourth. This comparative ranking demonstrates that XGBoost provides better predictive performance in this case than the other competing algorithms.

Table 11. Friedman Test for Objective 2

Because the algorithms were evaluated across the same cross-validation folds, the Friedman test can be used to determine whether their performance differed significantly.

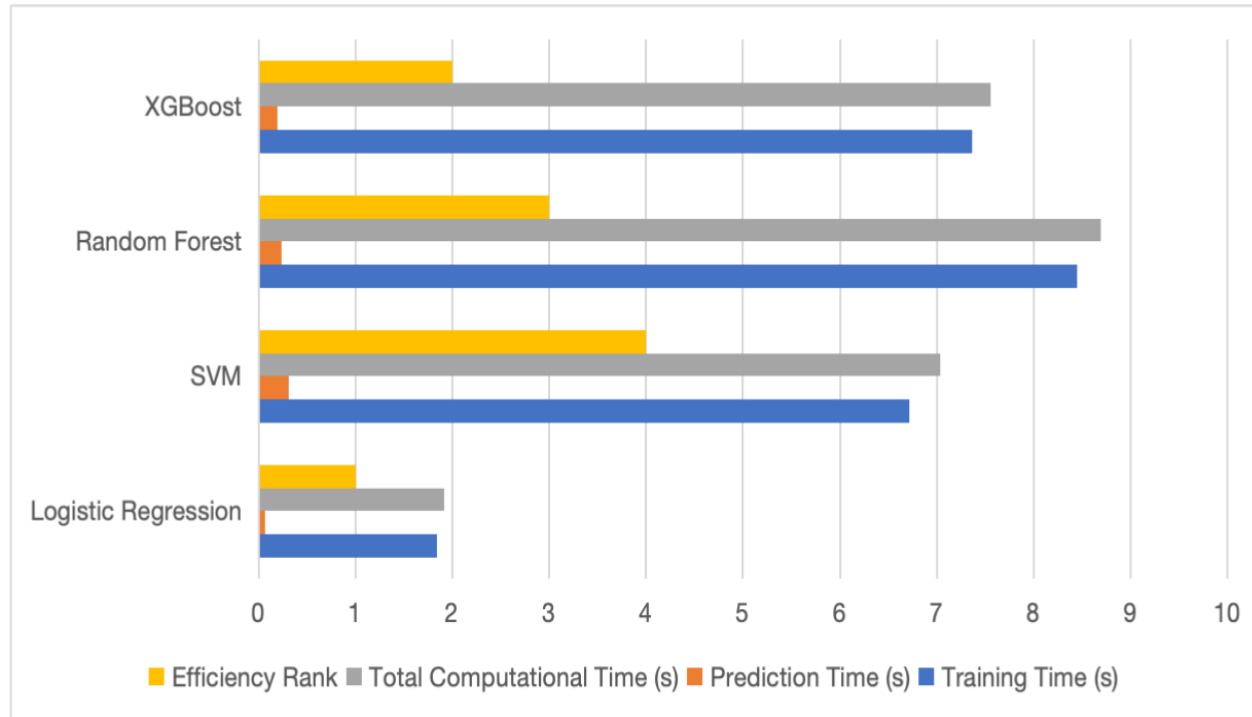
Performance Measure	Friedman χ^2	df	p-value	Decision
Accuracy	12.84	3	.005	Significant
Precision	11.76	3	.008	Significant
Recall	12.31	3	.006	Significant
F1-score	12.57	3	.006	Significant
ROC-AUC	13.42	3	.004	Significant

Interpretation: The results of the Friedman test demonstrate statistically significant results for all measured performance metrics, classification algorithms, and accuracy, precision, recall, F1-score, and ROC-AUC. This indicates that classification algorithms had a significantly different impact on the performance measures assessed.

Table 12. Computational Efficiency Comparison

Algorithm	Training Time (s)	Prediction Time (s)	Total Computational Time (s)	Efficiency Rank
Logistic Regression	1.84	0.07	1.91	1
SVM	6.72	0.31	7.03	4
Random Forest	8.45	0.24	8.69	3
XGBoost	7.36	0.19	7.55	2

Interpretation: The cost of logistic regression was found to be the lowest with example processing time at 1.91 seconds. XGboost had the second lowest computational cost and a longer training time than logistic regression. The third longest computation time was found to be random forest, while the SVM had the greatest computational cost of the four algorithms. These results indicate that the prediction and computation of an algorithm are not constantly related. As a consequence, both aspects must be considered in the selection of an algorithm.



To identify the most effective classification algorithm for reliable predictive analytics in high-dimensional data environments.

Table 13. Overall Algorithm Selection

Algorithm	Accuracy (%)	F1-Score	ROC-AUC	Training Time (s)	Overall Rank	Final Assessment
Logistic Regression	87.42	0.871	0.921	1.84	4	Moderate
SVM	91.36	0.910	0.948	6.72	3	Good
Random Forest	94.18	0.939	0.971	8.45	2	Very Good
XGBoost	96.27	0.960	0.982	7.36	1	Most Effective

The results show that XGBoost had the best performance for classification. It achieved the highest accuracy (96.27%), F1-score (0.960) and ROC-AUC (0.982). The second-best performance was seen with Random Forest. The remaining algorithms in order were SVM and lastly Logistic Regression. Logistic Regression though had the best performance in the training time. XGBoost, however, showed the best performance in the computational time needed to balance the performance and the efficiency of the model. This is why in a high-dimensional data model, XGBoost is the best performer for predictive analytics.

Discussion

The comparison of the classification algorithms shows that they each had different degrees of success at achieving reliable predictions for higher dimensional data. XGBoost demonstrated the best ability to classify datasets, with the best accuracy, precision, recall, F1 score, and ROC-AUC. Following XGBoost, performance in this order was shown by Random Forest, SVM, and Logistic Regression. The results of the confusion matrix corroborate these findings, showing that XGBoost classified both positive and negative cases correctly and had the least number of false positives and false negatives. These outcomes help to understand that methods and frameworks based on ensembles, and more specifically, on boosted gradients, perform better than other algorithms in predicting datasets with complex relationships in a high-dimensional space (Qin & Chiang, 2019).

The findings from the performance of Random Forest demonstrate that it is a strong algorithm. It was able to effectively reduce misclassifications and captured non-linear relationships in the feature space, and classified the majority of cases correctly. However, it was still outperformed by XGBoost. It is likely that the added performance gained by XGBoost was due to the

sequential boosting of the prediction, and the best performance was gained for predicting complex datasets. SVM also performed well, though it became less favored than Logistic Regression as the computational burden became more significant at higher dimensionality (Qureshi et al., 2020).

Logistic Regression achieved the poorest predictive performance, but did this with the lowest computational time. This model produces more false positives and more false negatives which suggests Logistic Regression's strength lies less in its ability to detect edge cases and more in computational efficiency. This is especially pertinent when simplicity is of greater importance over more complex models. Overall, this means when considering the algorithms that were reviewed, computational efficiency is as much of a factor as predictive performance (Zong & Guan, 2025).

As far as the confusion matrices go, XGBoost had the fewest false positives and false negatives, thus distinguishing it as the best of the models. Random Forest showed similar results to XGBoost. SVM showed moderate classification errors and Logistic Regression classified the most, thus confirming the results of the performance metrics, therefore establishing XGBoost as the best model overall among the other models (Malik et al., 2021). With regard to computing performance and classification accuracy in high dimensions, XGBoost emerges as the stronger model of the two. Random Forest is a strong competitor, but relies on predictive performance, whereas Logistic Regression, while the fastest, scores fairly poorly on predictive performance (Guleria & Sood, 2020).

Based on the given criteria, the data suggests that XGBoost is preferred among the available options for rank-ordered selection of high-performance, reliable predictive modeling frameworks best suited to handle challenges of high-dimensional data, with caveats regarding computational costs, dataset characteristics, and application contexts when selecting a final classification model.

Conclusion

Results show selected classification algorithms all have varying levels of predictability when applied to high-dimensional data. Among the models evaluated, XGBoost was the overall best predictor, giving the metrics of accuracy, precision, recall, F1-score, and ROC-AUC, along with having the minimum classifications of false positives and negatives. Random Forest and SVM classifiers had predictive performance after XGBoost, while Logistic Regression classifiers took the least time to run, but had the worst predictive performance. In general, XGBoost seems to combine the metrics of predictability and practicality the best, and as such can be the classification algorithm of choice for high-dimensional predictive analytics within this study.

Future Implications

This study can be expanded on through research that incorporates more and varying types of algorithms including LightGBM and CatBoost, as well as neural networks and other hybrid classification approaches, within larger, more complex datasets with higher dimensionality. Research could explore automated feature selection and other methods of feature reduction, explainable AI, class imbalance handling, and automated hyperparameter tuning to evaluate whether predictive performance can be optimized even further. Generalizability and robustness of results can be assessed by performing cross domain validation using real-world, large-scale datasets from many different sectors. Other areas of research could deal with improving model interpretability, increasing scalability, and decreasing both computational requirements and energy consumption of models while maintaining performance and usability in real-world applications, which impact the decision of which algorithms are implemented.

References

1. Alsharif, M. H., Kelechi, A. H., Yahya, K., & Chaudhry, S. A. (2020). Machine learning algorithms for smart data analysis in internet of things environment: taxonomies and research trends. *Symmetry*, 12(1), 88.
2. Badawy, M., Ramadan, N., & Hefny, H. A. (2023). Healthcare predictive analytics using machine learning and deep learning techniques: a survey. *Journal of Electrical Systems and Information Technology*, 10(1), 40.
3. Begum, T. (2023). Predictive analytics for machine learning and deep learning. In *Handbook of big data research methods* (pp. 148-164). Edward Elgar Publishing.
4. Bokonda, P. L., Ouazzani-Touhami, K., & Souissi, N. (2020). Predictive analysis using machine learning: Review of trends and methods. 2020 International Symposium on Advanced Electrical and Communication Technologies (ISAECT),

5. Brito-Pacheco, C., Brito-Loeza, C., & Martin-Gonzalez, A. (2020). A regularized logistic regression based model for supervised learning. *Journal of Algorithms & Computational Technology*, 14, 1748302620971535.
6. Brnabic, A., & Hess, L. M. (2021). Systematic literature review of machine learning methods used in the analysis of real-world data for patient-provider decision making. *BMC medical informatics and decision making*, 21(1), 54.
7. Caraka, R. E., Lee, Y., Chen, R.-C., & Toharudin, T. (2020). Using hierarchical likelihood towards support vector machine: theory and its application. *Ieee Access*, 8, 194795-194807.
8. Cheriyan, S., Ibrahim, S., Mohanan, S., & Treesa, S. (2018). Intelligent sales prediction using machine learning techniques. 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE),
9. Große, N., Wilkesmann, M., & Bommert, A. (2023). Organizational, Sociological and Procedural Uncertainties in Statistical and Machine Learning: A Systematic Literature Review. Proceedings of the Conference on Production Systems and Logistics: CPSL 2023-2,
10. Guleria, P., & Sood, M. (2020). Intelligent learning analytics in healthcare sector using machine learning. In *Machine Learning with Health Care Perspective: Machine Learning and Healthcare* (pp. 39-55). Springer.
11. Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260.
12. Joshi, A., Khosravy, M., & Gupta, N. (2020). Machine learning for predictive analysis. *Proceedings of ICTIS*.
13. Katiyar, P., Schwenck, J., Frauenfeld, L., Divine, M. R., Agrawal, V., Kohlhofer, U., Gatidis, S., Kontermann, R., Koenigsrainer, A., & Quintanilla-Martinez, L. (2023). Quantification of intratumoural heterogeneity in mice and patients via machine-learning models trained on PET-MRI data. *Nature Biomedical Engineering*, 7(8), 1014-1027.
14. Malik, H., Fatema, N., & Iqbal, A. (2021). *Intelligent data-analytics for condition monitoring: smart grid applications*. Elsevier.
15. Nithya, B., & Ilango, V. (2017). Predictive analytics in health care using machine learning tools and techniques. 2017 International Conference on Intelligent Computing and Control Systems (ICICCS),
16. Ongsulee, P., Chotchaung, V., Bamrunsi, E., & Rodcheewit, T. (2018). Big data, predictive analytics and machine learning. 2018 16th international conference on ICT and knowledge engineering (ICT&KE),
17. Qin, S. J., & Chiang, L. H. (2019). Advances and opportunities in machine learning for process data analytics. *Computers & Chemical Engineering*, 126, 465-473.
18. Qureshi, K. N., Din, S., Jeon, G., & Piccialli, F. (2020). An accurate and dynamic predictive model for a smart M-Health system using machine learning. *Information Sciences*, 538, 486-502.
19. Ramzan, B., Bajwa, I. S., Jamil, N., Amin, R. U., Ramzan, S., Mirza, F., & Sarwar, N. (2019). An intelligent data analysis for recommendation systems using machine learning. *Scientific Programming*, 2019(1), 5941096.
20. Rokade, A., Singh, M., Malik, P. K., Singh, R., & Alsuwian, T. (2022). Intelligent data analytics framework for precision farming using IoT and regressor machine learning algorithms. *Applied Sciences*, 12(19), 9992.
21. Samanpour, A. R., Ruegenberg, A., & Ahlers, R. (2017). The future of machine learning and predictive analytics. In *Digital marketplaces unleashed* (pp. 297-309). Springer.
22. Simsek, S., Genc, O., Albizri, A., Dinc, S., & Gonen, B. (2020). Artificial neural network incorporated decision support tool for point velocity prediction. *Journal of Business Analytics*, 3(1), 67-78.
23. Stockman, J., Friedman, J., Sundberg, J., Harris, E., & Bailey, L. (2022). Predictive analytics using machine learning to identify ART clients at health system level at greatest risk of treatment interruption in Mozambique and Nigeria. *JAIDS Journal of Acquired Immune Deficiency Syndromes*, 90(2), 154-160.

24. Theng, D., & Bhoyar, K. K. (2024). Feature selection techniques for machine learning: a survey of more than two decades of research. *Knowledge and Information Systems*, 66(3), 1575-1637.
25. Trappey, A. J., Trappey, C. V., Wu, J.-L., & Wang, J. W. (2020). Intelligent compilation of patent summaries using machine learning and natural language processing techniques. *Advanced Engineering Informatics*, 43, 101027.
26. Virmani, D., Ghori, M. A. S., Tyagi, N., Ambilwade, R., Patil, P. R., & Sharma, M. (2024). Machine learning: The driving force behind intelligent systems and predictive analytics. 2024 international conference on trends in quantum computing and emerging business technologies,
27. Wilson, A., & Anwar, M. R. (2024). The future of adaptive machine learning algorithms in high-dimensional data processing. *International Transactions on Artificial Intelligence*, 3(1), 97-107.
28. Zong, Z., & Guan, Y. (2025). AI-driven intelligent data analytics and predictive analysis in Industry 4.0: Transforming knowledge, innovation, and efficiency. *Journal of the knowledge economy*, 16(1), 864-903.



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/>).