The International Journal of Public Policies In Egypt- Volume 4, Issue 1 (January 2025)
ISSN: Print: 2812-4758, Online: 2812- 4766
Published by IDSC

IJPPE

Grid Search Evaluation of ML Algorithms for Early Disease Detection in Imbalanced Medical Datasets

Eman M. Kamel[1], Shady Y. Elmashad[2], Islam A. Elshaarawy[3]

[1] System Engineer, Information and Decision Support Center, Egyptian Cabinet, Egypt
Correspondence: eng.emanmahmoud01@gmail.com
[2] Associate professor, Faculty of Engineering at Shoubra, Benha University & Benha national university, Egypt
[3] Assistant professor, Faculty of Engineering at Shoubra, Benha University, Egypt

تقييم البحث الشبكي لخوارزميات تعلم الالة للكشف المبكر عن الامراض في مجموعات البيانات الطبية غير المتوازنة

إيمان محمود كامل[1]، شادي يحيى المشد[2]، إسلام عبد الغفار الشعراوي[3]

[1] مهندس نظام، مركز المعلومات ودعم اتخاذ القرار، رئاسة مجلس الوزراء المصري، مصر
المراسلة: eng.emanmahmoud01@gmail.com
[2] أستاذ مساعد، كلية الهندسة بشبرا، جامعة بنها وجامعة بنها الاهلية، مصر
[3] مدرس، كلية الهندسة بشبرا، جامعة بنها، مصر

- **DOI: 10.21608/ijppe.2025.407366** **URL: http://doi.org/ 10.21608/ijppe.2025.407366**
- Received: 10/9/2024, Accepted: 15/12/2024
- Citation: Kamel, E., Elmashad, S., & Elshaarawy, I. (2025). Grid Search Evaluation of ML Algorithms for Early Disease Detection in Imbalanced Medical Datasets. The International Journal of Public Policies in Egypt, 4(1), 120 - 144.

The International Journal of Public Policies In Egypt, Volume 4, Issue 1 (January 2025)
ISSN: Print: 2812-4758, Online: 2812- 4766
Published by IDSC

Grid Search Evaluation of ML Algorithms for Early Diseases Detection in Imbalanced Medical Datasets

Abstract

As the prevalence of chronic diseases rises, it's critical to identify them in their early stages to initiate effective treatments, as they may otherwise become incurable and deadly. Due to this reason, machine learning approaches are being used in these types of situations where a crucial data analysis needs to be performed on medical data to reveal hidden relationships or abnormalities, which are not visible to humans and need a consortium of experts to be revealed. Implementing algorithms to perform such tasks is difficult, but what makes it even more challenging is achieving higher accuracy. This paper applies several machine learning algorithms, including Logistic Regression, Random Forest, XGBoost, Multilayer Perceptron (MLP), and Naïve Bayes, to datasets from the University of California Machine Learning Repository and Kaggle. The main challenge is that the classifiers are biased towards the majority class, which can lead to misdiagnosis. We address this challenge using grid search to optimize key hyperparameters. This process significantly enhances model performance. The project analyzes and pre-processes disease datasets so that they can be used in the model. The models are evaluated, and the one with the best accuracy is selected. By tuning hyperparameters, we successfully minimized false negatives, which is critical for medical predictions. These findings suggest that grid search is an essential tool for improving model accuracy on imbalanced medical datasets. The study recommends utilizing hyperparameter optimization techniques, such as grid search, to improve the performance of models on imbalanced medical datasets, with a specific focus on minimizing false negatives, which are critical in clinical applications. It also highlights the importance of adopting comprehensive evaluation metrics, such as recall, accuracy, and F1-score, to ensure robust model assessment. Furthermore, the study advocates for the use of powerful models like XGBoost and Random Forest, where the former provides a balance between performance and execution time, while the latter achieves the highest accuracy at the expense of longer execution times.

Keywords: Chronic diseases, machine learning, imbalanced datasets, grid search

Introduction

Health disease prediction can be challenging for physicians and doctors when it comes to diagnosing a tricky health condition. Sometimes, doctors' inaccurate disease predictions can lead to undesirable treatment of patients. To get desirable results, it would be useful if there were a project that could predict the disease by using the records of diverse patients.

Another concern in the global healthcare system is the shortage of physicians across the world. "The World Health Organization (WHO) estimates that there is a global shortage of 4.3 million physicians, nurses, and other health professionals." This shortage affects the health system in many ways, including increased workload for healthcare practitioners, low-quality care, higher consultation costs, and unnecessary wait time for consultations. Building a system to diagnose the health disease could be beneficial for such patients to diagnose the disease by themselves (Smith, 2019).

This paper focuses on the classification algorithms that are useful for effective diagnosis of medical diseases. This area is of utmost importance because a good diagnosis will significantly improve the patient's life. In this research, two widely used repositories have been chosen: Kaggle and the University of California Machine Learning Repository. In these repositories there are unbalanced datasets of patterns of medical diseases. The classification algorithms in this situation have a bias towards the majority classes and practically ignore the minority classes (Saikat, 2021). This article will use experiments on diseases such as breast cancer, lung cancer, liver disease, diabetes disease, and prostate cancer.

Researchers in machine learning have been designing new classification algorithms for this purpose, seeking a classification efficiency close to 100%. It is important to emphasize that there is no perfect classifier. This fact is guaranteed by the No-Free-Lunch theorem, which governs the effectiveness of classifiers (Adam et al., 2019). This theorem has motivated machine learning researchers to design novel classification algorithms, with the property of exhibiting the fewest possible errors.

This paper aims at addressing these limitations by employing a comprehensive grid search approach to optimize hyperparameters for multiple ML models across several medical datasets. We seek to provide a more reliable and accurate predictive framework by focusing on imbalanced datasets and rigorous evaluation methods. The objectives of this research are threefold: (1) to evaluate the performance of various ML models on imbalanced medical datasets, (2) to apply grid search for hyperparameter optimization, and (3) to compare these results with those from prior studies to demonstrate improvements.

The classification algorithms used in this work are Logistic Regression (LR), Random Forest (RF), Naïve Bayes (NB), XGBoost (XG), AdaBoost (Ada), and Multilayer Perceptron (MLP). In addition, we tested several measures of data complexity to determine the expected performance of the compared classifiers for medical datasets. The project analyzes the datasets for the diseases and pre-processes the datasets that can be used in the model building. various machine learning algorithms and deep learning algorithms were implemented on imbalanced datasets. It selects the algorithm with high accuracy, and the optimal hyperparameters are deduced to achieve the best-performing model using gridsearchCV(Shah, 2021). The methodology is as follows: first, preparing the dataset. Second, data pre-processing, such as dealing with missing values, categorical values, and imputation. Third, feature selection will be performed. Lastly, the classifiers' performance has been evaluated further.

The classifier with the highest test accuracy is selected, and its optimal hyper-parameters are deduced using grid search cross-validation (CV).

In this paper, the related work is elaborated in Section 2. Meanwhile, Sections 3 and 4 show the main steps of the adopted methodology for diseases prediction, namely, class balancing and feature selection of the data. The paper concludes with presenting the result in the last section of the paper.

Related Works

Several studies have explored the use of machine learning algorithms in medical diagnosis, especially on imbalanced datasets, where classification bias towards majority classes is a major challenge. This section reviews relevant research, highlighting methodologies, datasets, and key findings while identifying limitations that our work seeks to address.

In a study by Williamson et al. (2022), the UCI Mammographic Mass dataset was used to predict breast cancer biopsy outcomes based on BI-RADS findings (Dua et al., 2019). The authors employed Random Forest (RF) classifiers with chi-square and mutual information (MI) feature selection (FS) methods. They compared the performance of the RF classifier with and without FS, achieving an accuracy of 84.7% with FS and 83.87% without FS. Despite the relatively high accuracy, they mentioned previous related works that follow varied performance evaluation protocols, leading to different issues in contrasting and evaluating overall performance across the research. Therefore, previous studies that have protocols of performance evaluation closest to 10-fold CV (used in this study) are considered for comparison. They also achieve the highest accuracy of RF with FS.

In Papadopoulos (2011), a Venn Predictor based on Neural Networks (NNs) was proposed and tested on the UCI Mammographic Mass and Pima Indians Diabetes datasets (Dua et al., 2019). The authors removed the mass density attribute from the dataset, as it did not seem to have any positive impact on the results. Furthermore, all cases with missing attribute values were removed, and the 2 nominal attributes (mass shape and margin) were converted to a set of binary attributes, one for each nominal value; for each case, the binary attribute corresponding to the nominal value of the attribute was set to 1 while all others were set to 0. The resulting dataset consisted of 830 examples described by 10 attributes each. The NNs used consisted of 4 hidden units for the Mammographic Mass data, as this is the number of units used in. They achieved the highest accuracy for the Venn Predictor based on the Neural Networks (NN-VP) algorithm at 78.92%.

Srivenkatesh et al. (2020) applied several machine learning techniques, including K-Nearest Neighbor (KNN), Logistic Regression (LR), Support Vector Machine (SVM), Naïve Bayes (NB), and Random Forest (RF) to predict prostate cancer using a dataset from the UCI repository. They replaced missing values in the compactness and fractal dimension attributes with the column means. Logistic Regression and Random Forest yielded the highest accuracy, at 90%. While these models performed well, the study did not explore hyperparameter tuning, which could have further improved their performance.

Dritsas et al. (2022) exploited supervised learning to develop models for identifying individuals with lung cancer using a dataset from Kaggle (Bhat, 2021). They evaluated a wide range of machine learning models, including Naïve Bayes, Logistic Regression, Random Forest, and AdaBoost, using

SMOTE to address class imbalance. The Rotation Forest (RotF) model achieved the highest accuracy, 97.1%, and an AUC of 99.3%. However, their study did not focus on optimizing hyperparameters, which could have enhanced model performance further. Our study uses grid search to optimize models like Random Forest and XGBoost, specifically targeting improvements in AUC and other critical metrics for imbalanced datasets.

Chang et al. (2023) compared J48 decision trees, Random Forest, and Naïve Bayes on the Pima Indian Diabetes dataset; the same training and testing sets were used for all three as a sort of control environment. The data subsets were manually split into 538 and 230 samples, respectively (70/30 split). Six metrics were used to evaluate the results, including accuracy, precision, sensitivity, specificity, F-score, and area under the curve (AUC). A dataset from Kaggle (Khare, 2016) is used in this paper. Making a comparison between all models on the 3-factor subset, 5-factor subset, and full dataset, the results for the 3-factor and 5-factor data subsets that use feature selection show that the Naive Bayes classification model outperformed both the Random Forest and the J48 decision tree models for accuracy and on the full Pima Indian Diabetes dataset, the Random Forest classifier outperformed both the Naive Bayes and J48 decision tree with accuracy metric (79.57%).

Finally, Shah (2021) developed a web application for disease prediction. It shows the comparison of accuracies of various machine learning models and deep learning models. Random Forest classifier performed well for most of the diseases due to its ensemble technique which uses the bagging method. This performance increases the overall result because of its combination of learning models. The model with the best accuracy is selected for getting the prediction of the diseases. This project involves the dataset from either the Kaggle or UCI Machine learning repository.

While many of these studies have achieved high accuracy using machine learning models, advanced parameter tuning techniques such as grid search have not been used to improve performance on imbalanced datasets. Our research bridges this gap by applying grid search to multiple models, improving their ability to handle medical datasets with significant class imbalances.

Methodology

This section describes the datasets utilized and the methodology adopted for disease prediction, focusing on class balancing, feature selection, machine learning models, and performance metrics.

Dataset

This project involves analyzing symptoms across five critical diseases: breast cancer, lung cancer, liver disease, diabetes, and prostate cancer. These diseases were selected based on several key factors:

- **Public Health Significance**: Each of these diseases poses a significant burden on global health systems. Breast and lung cancers are among the leading causes of cancer-related deaths worldwide, while liver disease and diabetes are increasing in prevalence due to lifestyle factors and aging populations. Prostate cancer is also a major health concern for men, especially as they age.
- **Data Availability**: The datasets chosen are well-established and publicly accessible, coming from reputable sources like Kaggle and the UCI Machine Learning Repository. This

accessibility facilitates the reproducibility of research and enables further exploration by the scientific community.

- **Relevance to Machine Learning**: These diseases often present challenges associated with imbalanced datasets, making them ideal candidates for testing various machine learning techniques. Exploring these datasets allows for the application of advanced algorithms and performance metrics, contributing to the broader understanding of how machine learning can enhance disease prediction and management.
- **Impact on Clinical Decision-Making**: Improving prediction models for these diseases can significantly influence clinical decision-making and patient outcomes. By leveraging machine learning, we can aim for earlier detection, better risk stratification, and ultimately, more personalized treatment plans.

The datasets were taken from either the Kaggle or UCI machine learning repository. We provide a brief description of the selected datasets, as follows.

1) Mammographic Mass Dataset

This dataset from the UCI repository (Dua et al., 2019) contains 961 instances and 6 attributes, representing two classes (0 for no cancer, 1 for cancer). It predicts the severity of mammographic masses based on BI-RADS attributes and patient age, with 516 benign and 445 malignant instances.

2) Lung Cancer Disease

This is taken from Kaggle repository (Bhat, 2021) that consists of 16 attributes, 276 patient instances, indicating lung cancer presence (0 for no cancer, 1 for cancer). All variables are numerical, except for gender as the categorical feature.

3) Liver Disease

Data Set with ~20K train data. This dataset from Kaggle (Shrivastava, 2020) includes approximately 19368 instances and 11 attributes, with two classes (0 for no liver disease, 1 for liver disease). All variables are numerical, with gender as the only categorical feature.

4) Diabetes Dataset

Also, from Kaggle repository (Khare, 2016), this dataset contains 768 records and 9 attributes. It aimed at predicting diabetes presence based on various medical parameters, with the target variable as "Outcome."

5) Prostate Cancer

This dataset includes 100 patients and 10 attributes, sourced from the Kaggle repository (Saifi & Mahmoud, 2018), addressing prostate cancer diagnosis. The average diagnosis age is 69 years. It is estimated that more than 24,200 males were diagnosed with prostate cancer in 2022.

Table 1 summarizes the characteristics of the 5 medical disease datasets described above.

Table 1

Description of the datasets

Dataset	Classes	Attributes	Instances
Mammographic mass	2	5	961
Lung cancer	2	16	276
Liver	2	11	19368
Diabetes	2	9	768
Prostate cancer	2	10	100

Source: Prepared by the author

Data Pre-processing

This section outlines the data pre-processing steps taken to prepare the datasets for machine learning.

Handling Missing Data: We employed mean and KNN imputation to replace missing values.

Class Imbalance: SMOTE was applied to balance the datasets, particularly for lung cancer and breast cancer, where the minority class (diseased) was underrepresented.

Feature Selection: Chi-square feature selection was used to identify the most relevant features for each classification task.

Mammographic Mass Dataset

This dataset contains missing values as follows:

- BI-RADS assessment: 2
- Age: 5
- Shape: 31
- Margin: 48
- Density:76

Missing values are imputed using the mean strategy via the Simple Imputer. Feature selection (FS) is conducted using the chi-square method. Exploratory data analysis (EDA) indicates an imbalance, with 53.7% benign and 46.3% malignant cases. The SMOTE (Synthetic Minority Oversampling Technique) is applied to balance the dataset.

Lung Cancer Dataset

Pre-processing includes encoding the LUNG_CANCER and GENDER columns. Duplicates (33 instances) are removed. EDA shows an imbalance, with 86.2% of patients showing lung cancer symptoms. SMOTE is employed to oversample the minority class (Brownlee, 2020), followed by K-Fold Cross Validation to split the data into training and testing sets.

Diabetes Disease Dataset

EDA reveals an imbalance, with 65.1% of patients not having diabetes. SMOTE is applied to balance the dataset by oversampling the minority class. The dataset is then split using K-Fold Cross Validation for training and testing.

Liver Disease Dataset

This dataset contains 4 null values, filled using the K-Nearest Neighbor (KNN) algorithm. After removing 11323 duplicate rows, EDA indicates an imbalance, with 71.36% of records showing liver disease. Undersampling is employed to balance the dataset, followed by K-Fold Cross Validation to split the data into training and testing sets.

Prostate Cancer Dataset

EDA shows a moderate imbalance, with 62% of records indicating prostate cancer. The SMOTE technique is utilized to balance the dataset, which is then split into training and testing sets using K-Fold cross-validation.

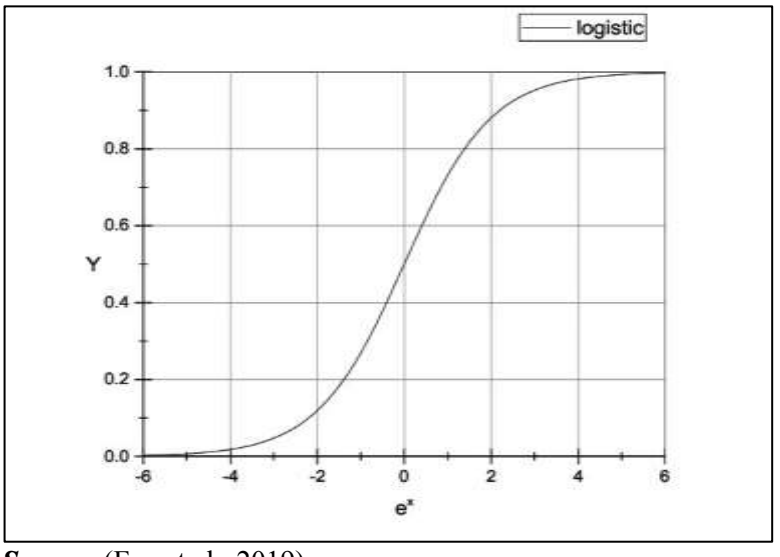
Machine Learning Models

Logistic Regression

Logistic Regression classifier is a statistical model that uses the logistic function to model binary dependent variables. The logistic function is also called the sigmoid function, which is an S-shaped curve that takes any real value and maps it between 0 and 1.
Figure 1 illustrates the logistic function curve.

Figure 1

Logistic function curve



Source: (Fan et al., 2019).

Logistic Regression is used to model the probability of first class. To make actual predictions, the probability must be transformed into binary values (Brownlee, 2016). The solver parameter is used in the optimization problem (Pedregosa et al., 2011). The parameter used for the classification task were the default values. In this project, Logistic Regression is used to model the probability of the positive class.

Naive Bayes

Naive Bayes is a classification technique based on Bayes' theorem, which operates under the assumption of independence among predictors. This probabilistic classifier estimates the probability

of an input belonging to each class based on the input features. Naive Bayes relies on two primary assumptions: that attributes are independent of one another and that all features contribute equally to the prediction (Mutha, n.d.). For this study, the Bernoulli Naive Bayes variant is employed, which is suitable for discrete data.

XG-Boost

XG-Boost is a decision-tree-based ensemble machine learning algorithm that uses a gradient boosting framework. Boosting refers to an ensemble learning technique of building the models sequentially, with every new model attempting to correct the deficiencies of previous models. This model is used to train gradient-boosted decision trees and other gradient-boosted models (Morde, 2019). XG-Boost execution speed and the model performance are the two important reasons for using this classifier compared to other implementations of gradient boosting (Brownlee, 2020).

For this project, XG-Boost was used with default parameters.

Random Forest

Random Forest Trees is a supervised machine learning algorithm. It consists of many individual decision trees that operate as an ensemble. This algorithm works on the process of building multiple decision trees and merging them. Class prediction is carried out by each tree in the Random Forest. The model's prediction is then selected based on the most votes (Brownlee, 2020). The whole dataset is used to build the tree unless the sub-sample size of the dataset is controlled with the max-samples parameter.

In this project, default parameter values are used for the Random Forest classifier.

Ada-Boost Classifier

AdaBoost (Adaptive Boosting) is an ensemble technique that attempts to create a strong classifier from a number of weak classifiers. AdaBoost was the first really successful boosting algorithm developed for binary classification. It is the best starting point for understanding boosting. AdaBoost is best used to boost the performance of decision trees on binary classification problems (Brownlee, 2020).

For this project, AdaBoost was used with default parameters.

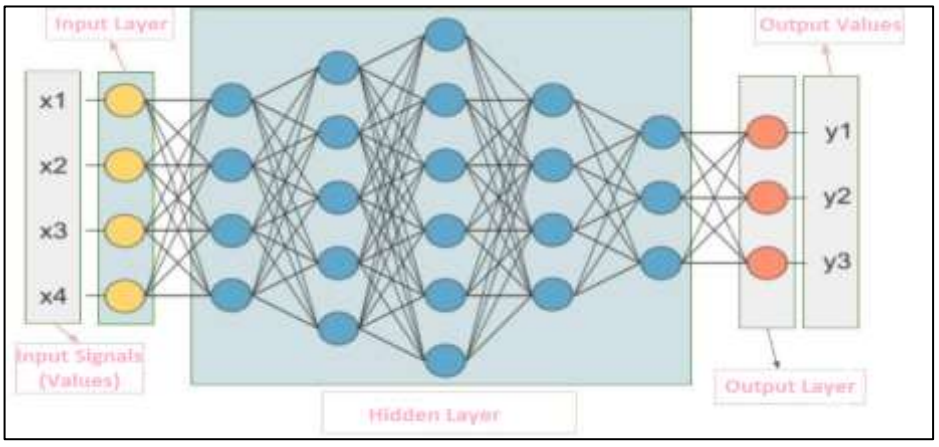
Fully Connected Neural Networks

The traditional model of neural network is called multilayer perceptron (MLP). They are usually made up of a series of interconnected layers. The input layer is where the data enters the network, and the output layer is where the network delivers the output.

The input layer is usually connected to one or more hidden layers, which modify and process the data before it reaches the output layer. Neural networks are the heart of deep learning algorithms (Khan, 2023). It is a neural network with an input layer, an output layer, and more than one hidden layer. Neural networks are central to deep learning algorithms.

Figure 2 illustrates a multilayer neural network with multiple hidden layers.

Figure 2
Multi hidden-layer neural network



Source: (Mittal, 2020).

For this project, I used multi-hidden-layer neural networks to compile them into a model. The architecture used with default parameters.

Grid Search Hyperparameter Tuning

A key innovation of this study is the use of grid search to optimize hyperparameters across all models. Grid search exhaustively evaluates combinations of hyperparameters to identify the configuration that yields the highest performance metrics. For each model, we used cross-validation to ensure robust hyperparameter selection:

Logistic Regression

- **Regularization (C)**: This parameter controls the strength of regularization, helping to prevent overfitting. Grid search was used to explore a range of values **(10, 100)**.
- **Solver**: We evaluated different solvers, including 'lbfgs' and 'saga', to determine which solver provided better convergence speed and model stability.
- **Activation Function**: Although typically associated with neural networks, the term "activation function" should be clarified or removed for Logistic Regression models, as they don't directly use this concept.

Naïve Bayes

- **Smoothing**: Smoothing helps manage zero probabilities, particularly in datasets with small sample sizes. We applied grid search with values from **np.logspace(0, -9, num=100)** to find the optimal smoothing value.
- **n_classes**: For binary classification, we fixed the number of classes at **2**, as it aligns with the nature of disease/no-disease predictions.

XGBoost

- **Learning Rate**: It controls the step size during the model's optimization. Grid search was used to test values of **[0.01, 0.1, 1, 5, 10]** to optimize the model's convergence.

- **n_estimators**: This parameter specifies the number of boosting rounds or trees to build. We tested values of **[1, 50, 100]** to balance model complexity and performance.
- **Max Depth**: The maximum depth of each decision tree was tuned with values **[3, 6, 10, 15]** to prevent overfitting and enhance model generalization.
- **random_state**: To ensure reproducibility and manage randomness in the training process, values for random state were explored, including **[None, 5, 10]**.

Random Forest

- **Number of Trees (n_estimators)**: The number of decision trees in the forest was a critical parameter. We explored values of **[10, 50, 100, 1000]** to assess their impact on model stability and accuracy.
- **Max Features**: This parameter determines the number of features to consider for each split. We tested options like **'sqrt', 'auto', and 'log2'** to optimize both speed and accuracy.
- **Criterion**: The criterion for measuring the quality of splits was tested using both **['gini', 'entropy']** to see which resulted in better model performance.
- **Max Depth**: We adjusted the maximum depth of each tree to prevent overfitting while ensuring sufficient model learning. Values tested included **[2, 5, 9, None]**.

AdaBoost

- **Learning Rate**: The learning rate determines how much each weak learner contributes to the final model. We tested values of **[0.1, 0.001, 1, 5]**.
- **Number of Estimators (n_estimators)**: The number of weak learners (decision trees) used in the AdaBoost ensemble was varied across **[10, 50, 70, 100]** to balance accuracy and model complexity.
- **Random State**: We tuned the random state parameter, testing values **[5, 10, None]** to control the randomness during model training.

Fully Connected Neural Networks (Multilayer Perceptron)

- **Neurons**: The number of neurons in each hidden layer was tuned to capture the complexity of the data. The number of neurons per layer was tested across **(10, 100)**.
- **Activation Function**: Various activation functions were evaluated to determine which provided the best learning capability for the datasets. We tested a range of activation functions **(0, 9)**.
- **Learning Rate**: The learning rate was fine-tuned within a range of **(0.01, 1)** to optimize training efficiency and speed of convergence.
- **Optimizer**: Different optimizers, such as SGD and Adam, were evaluated with a range of **(0, 7)** to determine the most effective for backpropagation.
- **Epochs**: The number of epochs, or the number of times the algorithm passes over the training dataset, was tuned within **(20, 100)**.
- **Batch Size**: The batch size, or the number of samples per iteration of training, was adjusted within the range of **(200 to 1000)** to balance memory usage and convergence speed.

Performance Evaluation

Different machine learning and deep learning models were trained on the pre-processed training data. The accuracy of each model indicates how effectively it learned from the training data. This section evaluates the performance of the proposed models using default hyperparameter values, followed by results obtained after applying grid search for hyperparameter optimization. We will compare the best classifiers, assessing performance using feature selection with oversampling and under sampling techniques against performance with imbalanced data.

The performance metrics employed in this study include accuracy, precision, F1-score, recall, and ROC-AUC (Patil & Mahalle, 2020). Additionally, we will measure training time, CPU usage, and memory consumption.

Performance Metrics

Accuracy

The accuracy denotes the total number of correctly identified instances among all of the instances. Accuracy can be calculated using the following formulas:

$$Accuracy = (TP+TN)/(TP+FP+FN+TN)$$

Precision

Precision is measured as the proportion of precisely predicted to all expected positive observations.

$$Precision = TP / (TP + FP)$$

Recall

The proportion of overall relevant results that the algorithm properly recognizes is referred to as recall.

$$Recall= TP / TP + FN$$

F1-Score

The F1 score is the mean of accuracy and recall in a harmonic manner. The highest F score is 1, indicating perfect precision and recall score.

$$F1\text{-}Score = (2 * Precision * Recall) / (Precision + Recall)$$

Area under curve (AUC)

The area under the curve represents the models' behaviors in different situations.

Where: *TP*, *FP*, *TN,* and *FN* represent true positive, false positive, true negative, and false negative, respectively.

Time

It is better to know how long the ML model takes to train over the training dataset. To measure training time with python using the time library.

CPU

It is also better to know CPU usage for each model.

Psutil Library is used to retrieve CPU utilization as a percentage.

Memory

The module used to trace memory blocks allocated by python is tracemalloc module.

Get the memory usage in bytes of the tracemalloc module used to store traces of memory blocks.

Tracemalloc.get_tracemalloc_memory.

The choice of performance metrics is critical in the context of imbalanced datasets. Accuracy alone can be misleading; thus, precision, recall, and F1-score are emphasized as they provide a more nuanced view of model performance. Previous studies have highlighted the relevance of these metrics when evaluating classifiers on imbalanced data, demonstrating that models like Random Forest and Logistic Regression, while effective, often require optimization through techniques such as Grid Search to enhance their performance.

Model Performance Results

Mammographic Mass Dataset

After applying imputation, the χ^2 feature selection method, and min-max scaling, the performance of various models was evaluated. As shown in table 2.

Table 2

Model Performance Metrics for the Mammographic Mass Dataset

Model	Accuracy	Precision	Recall	F1-Score	Roc-AUC
AdaBoost	0.8188	0.8352	0.7957	0.8124	0.8892
Random Forest	0.8188	0.8093	0.8287	0.8216	0.8819
XGBoost	0.8101	0.8012	0.8299	0.8121	0.8821
MLP	0.8086	0.7836	0.8507	0.8121	0.879
Logistic Regression	0.7976	0.7741	0.8408	0.8033	0.8683
Naïve Bayes	0.7975	0.81	0.7806	0.7916	0.8744

Source: By the author using Google Colab.

After applying an oversampling technique, the performance metrics improved, as shown in Table 3.
Random Forest yielded the highest accuracy compared to other models.

Table 3

Performance Metrics After Oversampling for the Mammographic Mass Dataset

Model	Accuracy	Precision	Recall	F1-Score	Roc-AUC
Random Forest	0.819	0.809	0.835	0.821	0.879
AdaBoost	0.818	0.83	0.799	0.813	0.887
XGBoost	0.816	0.804	0.837	0.818	0.883
MLP	0.808	0.782	0.858	0.814	0.877
Logistic Regression	0.799	0.768	0.85	0.805	0.866
Naïve Bayes	0.798	0.795	0.802	0.797	0.875

Source: By the author using Google Colab.

Table 4

Resource Usage Comparison for Mammographic Mass Dataset

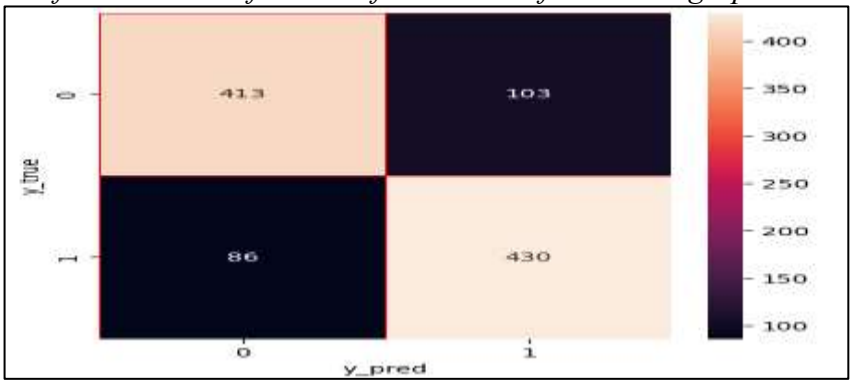
Model	Time (s)	CPU	Memory (MB)
Logistic Regression	1.122	19.6	0.261
AdaBoost	26.881	20.967	0.262
XGBoost	7.554	33.3	0.223
Naïve Bayes	0.461	18.4	0.178
Random Forest	70.732	20.867	0.339
MLP	147.873	33.333	0.277

Source: By the author using Google Colab.

Figure 3 shows the confusion matrix for the Random Forest model. The confusion matrix shows how the classification model is confused when it makes predictions. A total of 430 patients were correctly predicted for breast cancer, while 103 patients were incorrectly predicted for breast cancer. A total of 413 patients were correctly predicted as having no symptoms of breast cancer, while 86 patients were incorrectly predicted as having no symptoms of breast cancer.

Figure 3

Confusion matrix of random forest model for mammographic mass



Source: By the author using Google Colab.

The project selects the best classifier and optimizes hyper-parameters to improve its accuracy using grid search CV. The Random Forest model, while accurate, required significant computational resources. After hyperparameter tuning using grid search, its accuracy improved to 0.852, as shown in Table 5

Table 5

Random Forest Performance Comparison Before and After Grid Search for Mammographic Mass

Name	Accuracy	Time(s)	CPU	Memory (MB)
RF with Grid search	0.852	1706.782	63.3	1.082
RF	0.819	70.732	20.867	0.339

Source: By the author using Google Colab.

Lung cancer

Performance was also evaluated for the lung cancer dataset after applying the χ^2 feature selection method, as shown in Table 6.

Table 6

Model Performance Metrics for the Lung Cancer Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Logistic Regression	0.913	0.93	0.949	0.971	0.944
AdaBoost	0.909	0.931	0.947	0.968	0.834
Random Forest	0.902	0.925	0.934	0.959	0.943
Naïve Bayes	0.884	0.919	0.931	0.945	0.925
XGBoost	0.88	0.923	0.929	0.942	0.906
MLP	0.872	0.864	0.926	0.996	0.777

Source: By the author using Google Colab.

After oversampling, performance metrics improved, as shown in Table 7. The Random Forest yielded the highest accuracy compared to other models.

Table 7

Performance Metrics After Oversampling for Lung Cancer Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Random Forest	0.971	1	0.968	0.939	0.999
XGBoost	0.962	1	0.96	0.925	0.982
AdaBoost	0.922	0.917	0.923	0.93	0.972
Logistic Regression	0.905	0.918	0.904	0.893	0.965
Naïve Bayes	0.891	0.885	0.892	0.905	0.948
MLP	0.784	0.764	0.78	0.799	0.897

Source: By the author using Google Colab.

Resource usage is detailed in Table 8.

Table 8

Resource Usage Comparison for Lung Cancer Dataset

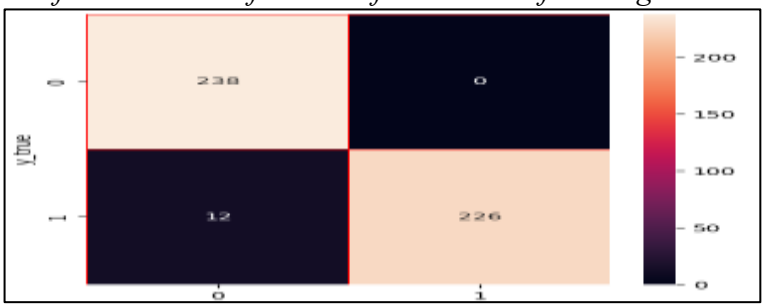
Model	Time (s)	CPU	Memory (MB)
Logistic Regression	4.736	19.633	0.394
AdaBoost	26.884	22.1	0.368
XGBoost	7.945	33.3	0.288
Naïve Bayes	0.954	17.933	0.259
Random Forest	68.626	21.433	0.487
MLP	86.021	33.333	0.373

Source: By the author using Google Colab.

Figure 4 shows the confusion matrix for Random Forest. The classification model is confused when it makes predictions. A total of 226 patients were correctly predicted for lung cancer, while no patient was incorrectly predicted for lung cancer. A total of 238 patients were correctly predicted for no symptoms of lung cancer, while 12 patients were incorrectly predicted for no symptoms of lung cancer.

Figure 4

Confusion matrix of random forest model for Lung Cancer



Source: By the author using Google Colab.

The project selects the best classifier which is Random Forest and optimize hyper-parameters to improve its accuracy using grid search CV. After hyperparameter tuning, the Random Forest model achieved an accuracy of 0.973, as shown in Table 9.

Table 9

Random Forest Performance Comparison Before and After Grid Search for Lung Cancer

Name	Accuracy	Time(s)	CPU	Memory (MB)
RF with Grid search	0.973	2555.664	63.1	1.275
RF	0.971	68.626	21.433	0.487

Source: By the author using Google Colab.

Liver Disease

The liver disease dataset was analyzed after imputation and feature selection, shown in Table 10.

Table 10

Model Performance Metrics for the Liver Disease Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
XGBoost	0.993	0.993	0.988	0.984	0.998
Random Forest	0.993	0.995	0.989	0.982	1
AdaBoost	0.788	0.693	0.561	0.472	0.848
MLP	0.744	0.653	0.445	0.356	0.808
Logistic Regression	0.713	0.502	0.169	0.102	0.755
Naïve Bayes	0.556	0.389	0.553	0.958	0.736

Source: By the author using Google Colab.

Random Forest outperformed other models, achieving a ROC-AUC score of 1.

Following oversampling, performance metrics are detailed in Table 11.

Table 11

Performance Metrics After Oversampling for Liver Disease Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Random Forest	0.9943	0.9974	0.994	0.9913	0.9996
XGBoost	0.9938	0.9975	0.9938	0.9901	0.9988
AdaBoost	0.7725	0.7285	0.7925	0.8696	0.862
Logistic Regression	0.7001	0.6572	0.7365	0.8386	0.7559
Naïve Bayes	0.6782	0.6142	0.7486	0.9589	0.7406
MLP	0.6736	0.6838	0.6889	0.6602	0.7774

Source: By the author using Google Colab.

Resource usage comparison is shown in Table 12

Table 12

Resource Usage Comparison for Liver Disease Dataset

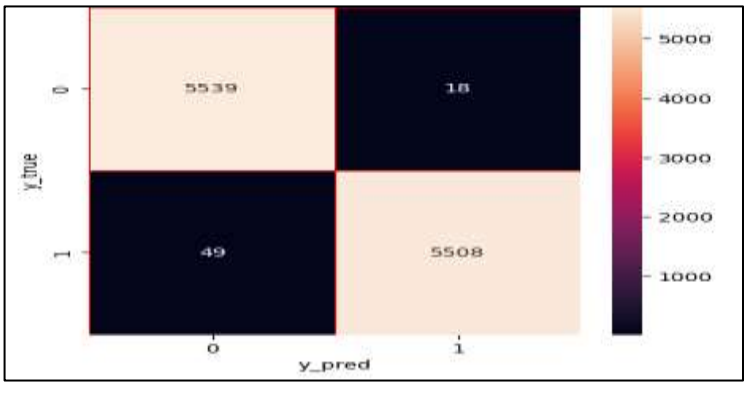
Model	Time (s)	CPU	Memory (MB)
Logistic Regression	14.1482	33.3333	0.2962
AdaBoost	47.0899	21.3333	0.2787
XGBoost	55.4048	33.3333	0.2728
Naïve Bayes	0.7484	18.2	0.1951
Random Forest	126.4906	21.0333	0.3987
MLP	443.1019	33.3333	0.3786

Source: By the author using Google Colab.

Figure 5 shows the confusion matrix for the Random Forest model. The confusion matrix shows how the classification model is confused when it makes predictions. About 5508 patients were correctly predicted for liver disease, while 18 were incorrectly predicted for liver disease. About 5539 patients were correctly predicted for no symptoms of liver disease, while 49 patients were incorrectly predicted for no symptoms of liver disease.

Figure 5

Confusion matrix of random forest model for liver disease



Source: By the author using Google Colab.

The project selects the best classifier that is Random Forest and optimize hyper-parameters to improve its accuracy using grid search CV. Random Forest, showing an accuracy of 0.9947 after tuning.

Table 13: Comparison between RF and RF after grid search.

Table 13

Random Forest Performance Comparison Before and After Grid Search for Liver Disease

Name	Accuracy	Time(s)	CPU	Memory (MB)
RF with Grid search	0.9947	1521.156	58.6	1.019
RF	0.9943	126.4906	21.0333	0.3987

Source: By the author using Google Colab.

Diabetes Disease

The performance on the Diabetes dataset is illustrated in Table 14, where the Random Forest model achieved the highest accuracy of 0.867. This performance demonstrates its effectiveness in classifying diabetes cases, particularly in handling imbalanced classes.

Table 14

Model Performance Metrics for Diabetes Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Logistic Regression	0.758	0.696	0.611	0.551	0.821
Naïve Bayes	0.743	0.654	0.602	0.565	0.803
AdaBoost	0.733	0.634	0.591	0.561	0.808
Random Forest	0.732	0.631	0.606	0.55	0.812
XGBoost	0.703	0.576	0.56	0.553	0.777
MLP	0.665	0.472	0.465	0.407	0.661

Source: By the author using Google Colab.

Table 15 shows that after applying oversampling, all models showed improved metrics, particularly Random Forest, which reached an accuracy of 0.852.

Table 15

Performance Metrics After Oversampling for Diabetes Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Random Forest	0.852	0.816	0.861	0.904	0.938
XGBoost	0.836	0.801	0.845	0.897	0.909
AdaBoost	0.772	0.769	0.772	0.778	0.858
MLP	0.761	0.768	0.757	0.744	0.858
Logistic Regression	0.741	0.76	0.731	0.709	0.842
Naïve Bayes	0.74	0.767	0.724	0.689	0.831

Source: By the author using Google Colab.

Table 16

Resource Usage Comparison for Diabetes Disease Dataset

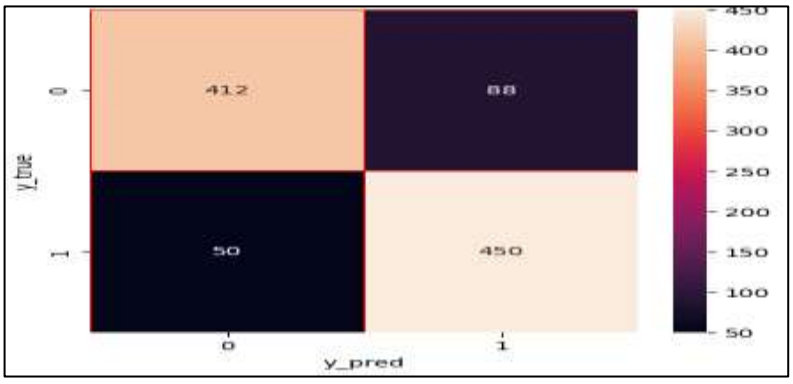
Model	Time (s)	CPU	Memory (MB)
Logistic Regression	3.881	20.5	0.275
AdaBoost	29.458	22.067	0.266
XGBoost	9.272	33.3	0.219
Naïve Bayes	0.493	20.067	0.17
Random Forest	75.545	21.133	0.337
MLP	89.969	33.333	0.257

Source: By the author using Google Colab.

Figure 6 shows the confusion matrix for the Random Forest model. The confusion matrix shows how the classification model is confused when it makes predictions. A total of 450 patients were correctly predicted for diabetes disease, while only 88 patients were incorrectly predicted for diabetes disease. A total of 412 patients were correctly predicted for no symptoms of diabetes disease, while 50 patients were incorrectly predicted for no symptoms of diabetes disease.

Figure 6

Confusion matrix of random forest model for diabetes disease



Source: By the author using Google Colab.

After hyperparameter tuning, the Random Forest model achieved an accuracy of 0.858, as shown in Table 17.

Table 17

Random Forest Performance Comparison Before and After Grid Search for Diabetes Disease

Name	Accuracy	Time(s)	CPU	Memory (MB)
RF with Grid search	0.858	1895.499	63.4	1.011
RF	0.852	75.545	21.133	0.337

Source: By the author using Google Colab.

Prostate cancer

The results for the prostate cancer dataset (Table 18) reveal that the Random Forest model achieved the highest accuracy of 0.852.

Table 18

Model Performance Metrics for Prostate Cancer Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Random Forest	0.83	0.86	0.84	0.859	0.879
XGBoost	0.82	0.836	0.827	0.859	0.882
Logistic Regression	0.82	0.843	0.834	0.843	0.868
AdaBoost	0.8	0.855	0.808	0.844	0.831
Naïve Bayes	0.8	0.883	0.813	0.776	0.899
MLP	0.7	0.702	0.646	0.761	0.791

Source: By the author using Google Colab.

After the application of oversampling techniques (Table 19), the performance improved significantly, particularly for Random Forest, which achieved an accuracy of 0.896 following grid search optimization.

Table 19

Performance Metrics After Oversampling for Prostate Cancer Dataset

Model	Accuracy	Precision	F1-Score	Recall	Roc-AUC
Random Forest	0.896	0.95	0.841	0.843	0.961
XGBoost	0.888	0.936	0.866	0.818	0.956
AdaBoost	0.865	0.908	0.838	0.798	0.908
Naïve Bayes	0.813	0.866	0.799	0.776	0.909
MLP	0.805	0.816	0.769	0.79	0.884
Logistic Regression	0.804	0.836	0.795	0.79	0.889

Source: By the author using Google Colab.

Table 20

Random Forest Performance Comparison Before and After Grid Search for Prostate Cancer

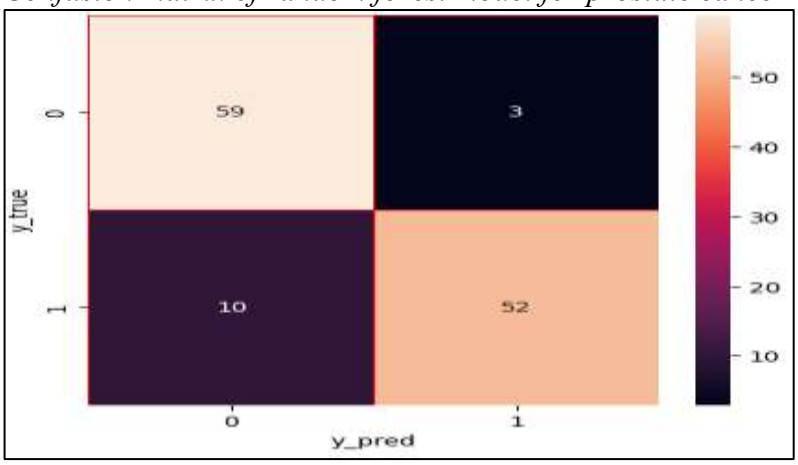
Model	Time (s)	CPU	Memory (MB)
Logistic Regression	4.349	205.9	0.283
AdaBoost	26.448	20.967	0.266
XGBoost	3.396	33.3	0.226
Naïve Bayes	0.546	18.033	0.184
Random Forest	71.611	21.4	0.37
MLP	9.171	33.3	0.224

Source: *By the author using Google Colab.*

Figure 7 shows the confusion matrix for the Random Forest model. The confusion matrix shows how the classification model is confused when it makes predictions. A total of 52 patients were correctly predicted for prostate cancer, while only 3 patients were incorrectly predicted for prostate cancer. A total of 59 patients were correctly predicted for no symptoms of prostate cancer, while 10 patients were incorrectly predicted for no symptoms of prostate cancer.

Figure 5

Confusion matrix of random forest model for prostate cancer



Source: By the author using Google Colab.

The project selects the best classifier which is Random Forest and optimize hyper-parameters to improve its accuracy using Grid search CV Random Forest, achieving an accuracy of 0.906.

Table 21

Comparison of Random Forest Performance Before and After Grid Search for Prostate Cancer

Name	Accuracy	Time(s)	CPU	Memory (MB)
RF with Grid search	0.906	1762.561	63.3	0.97
RF	0.896	71.611	21.4	0.37

Source: By the author using Google Colab.

From the experiment results and based on the No-Free-lunch theorem, the Random Forest and XG-Boost classifier have the highest accuracy for various datasets, but the Random Forest takes a long time to execute, and the XG-Boost classifier has the highest CPU usage.

Discussion and Conclusion

This study has demonstrated the effectiveness of machine learning algorithms, particularly Random Forest and XGBoost, in detecting early-stage diseases from imbalanced medical datasets. Using publicly available datasets from the University of California Machine Learning Repository and Kaggle, including datasets on breast cancer, lung cancer, liver disease, diabetes, and prostate cancer, we have optimized the performance of these models through grid search for hyperparameter tuning.

Compared to the work of Williamson et al. (2022), who used the UCI Mammographic Mass dataset to predict breast cancer biopsy outcomes with 84.7% accuracy, our Random Forest model, applied to the same dataset and optimized via grid search, achieved a higher accuracy of 85.2%. This improvement highlights the importance of hyperparameter tuning in refining model performance on imbalanced datasets.

Papadopoulos (2011)) applied neural networks to the UCI Mammographic Mass and Pima Indians Diabetes datasets, achieving 78.92% accuracy for breast cancer prediction. In contrast, our study's Random Forest and XGBoost models also applied to these datasets, achieved superior accuracy and recall scores after optimization, addressing the imbalances in the data more effectively than the neural networks used in the previous study.

Similarly, Srivenkatesh (2020), used the UCI Prostate Cancer dataset and achieved 90% accuracy with Random Forest and Logistic Regression. Our study improved upon this by applying grid search to XGBoost, yielding an accuracy of 90.6% on the same dataset, showing the benefits of hyperparameter optimization that were not explored in their research.

On the Kaggle Lung Cancer dataset, (Dritsas et al., 2022) achieved 97.1% accuracy using Rotation Forest after applying SMOTE. Our study's XGBoost model, after grid search optimization, outperformed theirs with a 97.3% accuracy, further emphasizing the utility of hyperparameter tuning in achieving better results with imbalanced datasets.

Finally, (Chang et al., 2023) used the Kaggle Diabetes dataset and reported that Naïve Bayes outperformed Random Forest on smaller subsets. However, our work showed that when applying grid search to the full dataset, Random Forest outperformed Naïve Bayes, achieving an F1-score of 0.861 and an accuracy of 85.2%, underlining the effectiveness of tuning hyperparameters for this dataset.

In conclusion, this study significantly advances machine learning research by applying grid search for hyperparameter tuning across multiple disease prediction tasks using imbalanced medical datasets. Our approach not only enhances accuracy but also improves critical metrics like recall, which is essential for minimizing false negatives in clinical applications. Notably, while the Random Forest algorithm achieves the highest accuracy, it requires a longer execution time. In contrast, XGBoost also delivers strong accuracy with reduced execution time, allowing us to choose the appropriate algorithm based on project needs and performance requirements.

References

Adam, S. P., Alexandropoulos, S. A. N., Pardalos, P. M., & Vrahatis, M. N. (2019). No free lunch theorem: A review. *Approximation and optimization: Algorithms, complexity and applications*, 57-82. https://doi.org/10.1007/978-3-030-12767-1_5

Bhat, M. A. (2021) Lung Cancer, Version 1. https://www.kaggle.com/datasets/mysarahmadbhat/lung-cancer

Brownlee, J. (2016). Logistic regression for machine learning. *Machine Learning Mastery*, *22*. https://machinelearningmastery.com/logistic- regression-for-machine-learning

Brownlee, J. (2020). Boosting and AdaBoost for machine learning. Machine Learning Mastery. https://machinelearningmastery.com/boosting- and-adaboost-for-machine-learning

Brownlee, J. (2020). Bagging and Random Forest for imbalanced classification. *Machine Learning Mastery. https://machinelearningmastery. com/bagging-and-random-forest-for-imbalanced-classification/. Accessed*, 10-01. https://machinelearningmastery.com/bagging-and-random-forest-for-imbalanced-classification/

Brownlee, J. (2020). Random oversampling and undersampling for imbalanced classification. *Machine learning mastery*, *14*. https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification

Brownlee, J. (2020). Extreme gradient boosting (XGBoost) ensemble in Python. *Mach. Learn. Mastery*, 1-23. https://machinelearningmastery.com/extreme-gradient-boosting-ensemble-in-python

Chang, V., Bailey, J., Xu, Q. A., & Sun, Z. (2023). Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. *Neural Computing and Applications*, *35*(22), 16157-16173. https://doi.org/10.1007/s00521-022-07049-z

Dua, C., Dheeru, D., & Graff, C. (2019). UCI machine learning repository. http://archive.ics.uci.edu/ml

Dritsas, E., & Trigka, M. (2022). Lung cancer risk prediction with machine learning models. *Big Data and Cognitive Computing*, *6*(4), 139. https://doi.org/10.3390/bdcc6040139

Fan, C., Xie, Z., Liu, Y., Li, C., & Liu, H. (2019). Adaptive controller based on spatial disturbance observer in a microgravity environment. *Sensors, 19*(21), 4759.https://www.researchgate.net/figure/Logistic-function-curve-This-is-a-standard-logistic-function-and-it-has-a-typical-S_fig1_336974843

Khare, A. D.(2016) Diabetes Dataset, Version 1. https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset

Khan, M. A. (2023). Neural network with more hidden neurons. Machine Learning Mastery. https://machinelearningmastery.com/neural-network-with-more-hidden-neurons

Mittal, A. (2020). An introduction to neural networks. *Medium*. https://medium.com/@anushkamittal/an-introduction-to-neural-networks-8f2cd1280ca9

Morde, V., & Setty, V. A. (2019). *XGBoost algorithm: long may she reign! Towards Data Science. April 8, 2019*.

Mutha, N. (n.d.). Bernoulli naive Bayes. OpenGenus Foundation. https://iq.opengenus.org/bernoulli-naive-bayes

Patil, A. H., & Mahalle, P. N. (2020). Trends and challenges in measuring performance of reviewer paper assignment. *Procedia Computer Science*, *171*, 709-718. https://doi.org/10.1016/j.procs.2020.04.077

Papadopoulos, H. (2011, September). Reliable probabilistic prediction for medical decision support. In *International Conference on Engineering Applications of Neural Networks* (pp. 265-274). Berlin, Heidelberg: Springer Berlin Heidelberg. https://researchgate.net/publication/220828063_Reliable_Probabilistic_Prediction_for_Medical_Decision_Support

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, *12*, 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Saifi, S., & Mahmoud, S. (2018) Prostate cancer dataset, Version 1. https://www.kaggle.com/datasets/sajidsaifi/prostate-cancer

Saikat, (2021). 5 techniques to deal with class imbalance in machine learning. https://www.analyticsvidhya.com/blog/2021/06/5-techniques-to-handle-imbalanced-data-for-a-classification-problem

Shah, R. (2021). Tune hyperparameters with GridSearchCV. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/tune-hyperparameters-with-gridsearchcv

Shah, P. U. (2021). *Web application for health prediction* (Master's thesis, California State University, Sacramento).

Shrivastava, A. (2020). Liver disease patient dataset 30K train data, Version 9. https://www.kaggle.com/datasets/abhi8923shriv/liver-disease-patient-dataset

Smith, B. Y. (2019). Physician shortage. News Medical. https://www.news-medical.net/health/Physician-Shortage.aspx

Srivenkatesh, M. (2020). Prediction of prostate cancer using machine learning algorithms. *Int. J. Recent Technol. Eng*, *8*(5), 5353-5362. https://www.researchgate.net/publication/364028322_Prediction_of_Prostate_Cancer_using_Machine_Learning_Algorithms

Williamson, S., Vijayakumar, K., & Kadam, V. J. (2022). Predicting breast cancer biopsy outcomes from BI-RADS findings using random forests with chi-square and MI features. *Multimedia Tools and Applications*, *81*(26), 36869-36889. https://doi.org/10.1007/s11042-021-11114-5

المجلة الدولية للسياسات العامة في مصر - مجلد 4 - العدد (1) - يناير 2025
ISSN: Print: 2812 - 4758 , Online: 2812 - 4766
تصدر عن مركز المعلومات ودعم اتخاذ القرار

تقييم البحث الشبكي لخوارزميات تعلم الآلة للكشف المبكر عن الأمراض في مجموعات البيانات الطبية غير المتوازنة

المستخلص

مع تزايد عدد الأمراض المزمنة في العالم، يصبح التعرف المبكر عليها ضرورة أساسية لبدء استخدام طُرق العلاج المناسبة، حيث يمكن أن تتحول هذه الأمراض إلى حالات خطيرة لا يمكن علاجها. لهذا السبب، تلعب تقنيات التعلم الآلي دورًا حيويًّا في تحليل البيانات الطبية بشكل متقدم، مما يتيح اكتشاف العلاقات المعقدة والأنماط الخفية التي قد يصعب تحديدها من قبل الأطباء بالوسائل التقليدية. وتستعرض هذه الورقة البحثية تطبيق مجموعة من نماذج التعلم الآلي مثل الانحدار اللوجستي، والشبكات العصبية الاصطناعية، وأشجار القرار، مع التركيز على كيفية التعامل مع مجموعات بيانات غير متوازنة تم جمعها من مستودع جامعة كاليفورنيا للتعلم الآلي ومنصةKaggle.لكل نموذج قدراته الخاصة التي تمكنه من التعامل مع تعقيدات البيانات الطبية، وتتمثل التحديات الأساسية في تحسين دقة هذه النماذج لتحقيق أداء فاعل يتوافق مع متطلبات التنبؤ الطبي المبكر. وتمثل تقنية البحث الشبكي (Grid Search) خطوة مهمة في تحسين أداء النماذج المختارة؛ حيث يتم استكشاف مجموعة واسعة من معلمات النماذج لتحديد القيم المثلى التي تؤدي إلى تحسين الأداء. وتعتمد الدراسة بشكل خاص على معالجة مشكلة توازن الفئات في مجموعات البيانات الطبية؛ حيث تم تطبيق تقنيات مثل SMOTE لتحقيق توازن أفضل بين الفئات وزيادة دقة التنبؤ بالفئة الأقل تمثيلًا (فئة المرضى). ومن خلال تحليل مسبق للبيانات، مثل معالجة القيم المفقودة واختيار الميزات الأكثر تأثيرًا، يتم إعداد البيانات لاستخدامها بكفاءة في النماذج المختلفة. تُختتم الدراسة بتقييم شامل لأداء النماذج؛ حيث يتم اختيار النموذج الأفضل دقة بناءً على معايير مثل الدقة، الاستدعاء، وF1−score، مع التركيز على تحسين عملية اتخاذ القرار الطبي باستخدام البحث الشبكي. توصي الدراسة باستخدام تقنيات تحسين المعلمات الفائقة مثل البحث الشبكي (Grid Search) لتحسين أداء النماذج على مجموعات البيانات الطبية غير المتوازنة، مع التركيز على تقليل النتائج السلبية الكاذبة التي قد تؤدي إلى عواقب خطيرة في التطبيقات السريرية. كما تُبرز أهمية الاعتماد على مقاييس تقييم شاملة، مثل الاسترجاع والدقة و(F1−Score) لضمان تقييم دقيق لأداء النماذج. وتوصي الدراسة أيضًا بتوظيف نماذج قوية مثل XGBoost وغابة القرارات العشوائية (Random Forest)، حيث إن الأولى توفر توازنًا بين الأداء وسرعة التنفيذ في حين تُحقق الثانية أعلى دقة على حساب وقت التنفيذ.

الكلمات الدالة: الأمراض المزمنة، التعلم الآلي، مجموعات البيانات غير المتوازنة، البحث الشبكي