Applied Medical Informatics
Vol. 43, No. 3/2021, pp: -

Interpretable Machine Learning Model Selection for Breast Cancer Diagnosis Based on K-means Clustering

Dieudonne N. OUEDRAOGO*

Binghamton University, Department of System Science and Industrial Engineering, State University of New York 4400 Vestal Pkwy E, Binghamton, NY 13902, USA.
E-mail: douedra1@binghamton.edu

* Author to whom correspondence should be addressed; Tel.: +1 646-342-7841

Received: 14 June 2021/Accepted: 19 September 2021/ Published online: 27 September 2021

Abstract

Background: Breast cancer affects millions of women; with the increasing growth in data collection in recent years, machine learning models are used in the diagnosis phase. While the accuracy of the models plays a significant role in choosing a model, the interpretability of the model for doctors and decision-makers is crucial in understanding and building trust in breast cancer diagnosis. In practice, it is challenging for researchers and practitioners to select the optimal model based on multiple objectives such as accuracy, interpretability, and computation runtime. We proposed a model selection technique unifying various objectives based on K-means clustering. This study's main contribution is the use of interpretable machine learning techniques such as LIME, ELI5, and SHAP and machine learning algorithms to predict the tumor type. ***Materials and Methods:*** The data used in this study were collected by Dr. William H. Wolberg, W. Nick Street, Olvi L. Mangasarian from the University of Wisconsin Hospitals, and donated to the UCI machine learning repository by Nick Street. Forty-three models are built using the dataset. The runtime for each model is recorded in seconds, and the Accuracy, Balanced Accuracy, the AUC-ROC, the F1-score, and the interpretability tool are compiled. A K-means clustering algorithm is applied to the resulting outputs. Through the elbow method, three categories of clusters are selected. ***Results:*** The proposed method showed high performance, as well as ease in interpreting the model. The K-means clusters' characteristics show that models in cluster number 2 have low and medium interpretability and low computation runtime. ***Conclusion:*** AdaBoost and XGBoost Classifiers with ELI5 interpretability are the most performant and most explainable models. They show the highest accuracy and the lowest computation runtime, and each prediction is explained by a linear combination of the top features.

Keywords: Breast cancer diagnosis; Interpretable machine learning; Interpretability; Explainability; Model selection; K-means clustering

Introduction

Machine learning (ML) and artificial intelligence (AI) are sets of applications, tools, and processes used to learn patterns from data sets and make predictions and decisions without being explicitly programmed [1]. The predictions are based on a finalized model and selected amount to a large number of others. Choosing a model based on a single objective such as accuracy is relatively easy; a selection based on multiple goals such as interpretability and usage of computing resources is challenging. Black-box models [2], such as neural networks, ensemble boosting, and stacked models, are widely used to improve accuracy, but a new challenge arises in the form of *trust*. Modeling in healthcare domains such as breast cancer diagnosis often carries multiple objectives; models need to be explainable and trustable for doctors and decision-makers. Lundberg [3] and Ribeiro [4] highlight those challenges in their articles and propose interpretability tools such as SHAP (SHapley Additive exPlanations), LIME (Local Interpretable Model-Agnostic Explanations), and ELI5 (Explain Like I'm 5) for black-box models. The goal of using those techniques is to understand the prediction made by the black-box model in terms of the initial input variables.

Interpretability

Machine learning interpretation and explanation are crucial to ensure no bias in the modeling process; it provides trust and transparency. The lack of model transparency and the inability to understand large and complex models lead to trust issues in Machine Learning and Artificial Intelligence systems. Deep learning and neural network models, gradient boosting, and random forest algorithms are widely used in breast cancer diagnosis. The predictions with those models have shown excellent results [5-7]; however, the lack of understanding of the rules behind those predictions generates trust issues. Ribeiro [8], through their article, explained the importance of using interpretable models and interpretability tools on building trust. When a model's prediction can be decomposed into a combination of the model input variables, it changes the black-box model into a transparent model. It improves trust in the predictions being made. Interpretability [9] is currently a hot topic as many healthcare domains require accurate and explainable models; Doshi [10] described a scenario where interpretability is needed and pointed to fairness, trust, and reliability are the main reasons for using interpretability. Deep learning models' interpretation techniques are scarce due to the complexity of deep neural networks and the number of parameter features used. For the interpretability of Convolution Neural Networks (CNN) models, Bau [11] proposed using a Network Dissection attribution of latent representations by estimating the alignment between hidden units and the semantic concepts. Many systems, such as image classifiers, operate on low-level features rather than high-level concepts. The Concept Activation Vectors (CAVs) are introduced to address these challenges, where the neural net's internal state is interpreted in terms of human-friendly concepts.

- ***SHAP:*** The most widely used interpretation technique is SHAP which stands for SHapley Additive exPlanation. SHAP was proposed by Lundberg [3] and defined as a unified approach for interpretability; it is a technique of attribution of feature importance based on Shapley game theory developed by Shapley [12] to interpret and explain complex machine learning models. However, SHAP's general framework is slow and often not usable for deep learning models; gradient-based algorithms often replace SHAP for deep learning models.
- ***LIME:*** Ribeiro [6] proposed using model-agnostic techniques to interpret machine learning predictions. All models are treated as black-box, which generates flexibility in choosing models, interpretations, and representations, that improve the debugging, comparison, and interfaces for various techniques. There are many challenges in building such a framework as outlined by Ribeiro [6]; to mitigate those challenges, they introduced LIME, a model-agnostic explanation technique.
- ***ELI5*:** ELI5, often called ELI, is a local explanation approach built on the same principle as LIME, which approximates the complex model locally to a linear model where the output is similar to regression with coefficients and bias.

We proposed a model selection technique unifying multiple objectives performance metrics, computation runtime, and interpretability based on K-means clustering. This study's main contribution is the use of interpretable machine learning techniques such as LIME, ELI5, and SHAP and machine learning algorithms to predict and explain the tumor type.

Material and Method

Dataset

The dataset is the breast cancer data from the UC Irvine Machine Learning Repository it is used to illustrate the process. The data was collected in the University of Wisconsin Hospitals, Madison Dr. William H. Wolberg, W. Nick Street, Olvi L. Mangasarian, and donated to the UCI machine learning repository by Nick Street [15-17]. The data has 569 instances; two Classes, 212 - malignant, 357 – benign, and 30 columns comprised of the following attributes: Radius, which is the mean of distances from the center to the points on the perimeter; Texture, which represents the standard deviation of the gray-scale values; Smoothness, that defines the local variation in radius lengths; Area; Perimeter; Compactness defines as the Perimeter^2/Area - 1.0; Concavity, which expresses the severity of concave portions of the contour; Concave points, the number of concave portions of the contour; Symmetry; Fractal dimension, which is defined as the coastline approximation – 1. There are 30 features since each image has a mean, a standard error, and a worst/largest (mean of the three worst/largest values) computed. For instance, field 0 is Mean Radius; field 10 is Radius SE, field 20 is Worst Radius. All the features are numeric, and the dataset contains no missing values.

The data set is unbalanced, two Classes, 212 - Malignant, 357 – Benign, as depicted in Figure 1.

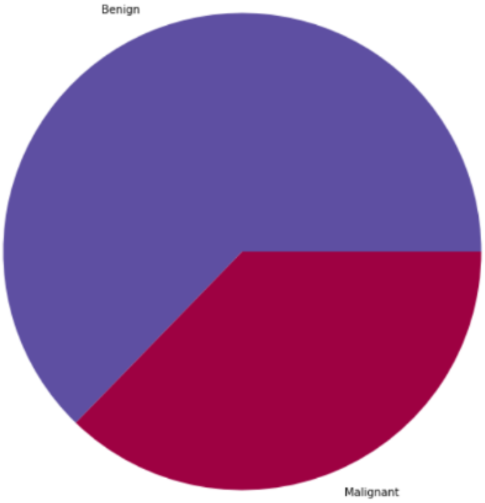


Figure 1. The distribution of data based on classes

Modeling Phase

The dataset was stratified split into 75 percent for training and 25 percent for testing to assess the performance of each model. The stratification goal was to ensure that the percentage of benign and malignant cases remained the same in training and testing sets. All 30 features of the dataset were used in building the models.

The algorithms, classifiers, and models implemented in this study include a broad set of commonly used techniques by data scientists found in the Scikit-learn python package to simulated the variate of choices often faces by modelers. The methods considered are: AdaBoost [18], Adaptive Boosting proposed by Yoav Freund and Robert Schapire that won the 2003 Gödel Prize, an ensemble of decision trees (weak learners) where their outputs are weithed sum to get the final output of a strong learner (the final boosted output); CatBoost [19], Categorical Boosting a machine learning technique based on gradient boosting on decision trees; XGBoost [20] which stands for eXtreme Gradient Boosting, an optimized gradient boosting method that allow boosting to be implemented in parallel in order to improve speed; Random Forest [21] a frequently used ML method that use bagging techniques over decision trees, the method was proposed by Leo Breiman and Adele Cutler; LightGBM [22] an alogrithm widely used by data scientists and developed by Microsoft based on distributed gratient boosting techniques; Linear Discriminant Analysis [23], Support Vector Machine [24], Stochastic Gradient Classifier [25], the Perceptron [26], Quadratic Discriminant Analysis [27], Logistic Regression [28], Label Propagation [29], Label Spreading [30], Ridge Classifier CV [31], Ridge Classifier [32], Extremely Randomized Trees (Extra Trees Classifier) [33], Passive Aggressive Classifier [34-35], Linear SVC [36], Calibrated Classifier CV [37], K-Nearest Neighbors [38], Bagging Classifier [39], Bernoulli NB [40], Nu SVC [41], Nearest Centroid [42], Gaussian NB [43], Decision Tree [44]. Python and Scikit-learn [45] are used in this study. All models were built using the training dataset, and the performance metrics were assessed on the test set.

Performance

The performance metrics used in this study are *Accuracy, Balanced-Accuracy, AUC-ROC, F1-score*, the computation *runtime*.

A breast cancer diagnosis is a binary classification with *benign* as a Negative class and *malignant* as a Positive class. There are four basic combinations in a binary classification of actual data category and assigned category: *true positives* - *TP* (correct positive assignments), *true negatives* - *TN* (correct negative assignments), *false positives* - *FP* (incorrect positive assignments), and *false negatives* - *FN* (incorrect negative assignments). Furthermore, *True Positive Rate TPR* is defined as *TP/(TP+FN)* and referred to as sensitivity or recall, while *True Negative Rate TNR* is defined as *TN/(TN+FP)* is named specificity.

The *Accuracy* or *Fraction Correct (FC)* measures the fraction of all instances that are correctly categorized.

We also compute the *Balanced-Accuracy* for unbalance datasets such as breast cancer data as the average recall obtained on each class. *Balanced-Accuracy = (TPR+TNR)/2*

ROC, known as the Receiver Operating Characteristic curve, is a plot of the performance of a classifier at all thresholds. It represents the *TPR* versus the *FPR* at all possible classification thresholds. *AUC-ROC,* the Area Under the *ROC* curve, measures the surface underneath under the *ROC* curve. The *AUC-ROC* is the probability that the classifier ranks a random positive sample more highly than a random negative one; a higher value indicates a better classifier.

F1-score often called balanced F-score, and *F-measure* is the harmonic mean of the Precision and the Recall:

*F1-score = 2(Precision*Recall)/(Precision + Recall)*
where: *Precision is TP/(TP+FP), and Recall is TP/(TP+FN)*

K-Means Clustering

K-means clustering [13] is a vector quantization method that divides the dataset's M observations and N features into K clusters to minimize the sum of squares of the within-cluster distances. This study uses the algorithm to find natural groups of models to be chosen based on the *Accuracy, Balanced-Accuracy, ROC-AUC, F1-score*, the *runtime*, and the *Interpretability* technique. The number of clusters is determined by using the elbow method [14].

The elbow method consists of running successive K-means clustering on the dataset for incremental K values and then computing the sum of squared distances from each point to its assigned cluster center. The total sum of squared distances, referred to as *explained variance,* is plotted against the number of clusters, and the elbow of the curve is used to define the number of clusters.

K-means Silhouette Score

The clustering performance is assessed by computing the average silhouette coefficient [45] for the entire dataset. The mathematical expression of the silhouette score is *(b-a)/max (a, b)*; where *a* represents the mean of the intra-cluster distances and *b* the distance between the data point and the closest cluster that the data point is not belonging. Values are between -1 and 1. The perfect silhouette score being 1. Values around zero indicate overlapping clusters, while negative implies that the datapoint belongs to the wrong group.

Results

Models Performance Metrics and Interpretability Tools

The performance metrics and the interpretability tools used are compiled in Table 1. In white-box models, such as Logistic Regression, there is no need for an external interpretability tool. The interpretability is referred to as *ITSELF* as the model explains itself. Table 1 shows the results obtained by using each model ID.

Table 1. Models performance metrics and Interpretability tools

Model ID	Accuracy	Balanced Accuracy	AUC ROC	F1 Score	Run Time	Inter-pretability
AdaBoost Classifier	0.979021	0.979455	0.979455	0.979060	0.232496	SHAP
CatBoost Classifier	0.979021	0.979455	0.979455	0.979060	7.602528	SHAP
XGBoost Classifier	0.979021	0.975577	0.975577	0.978979	0.173537	SHAP
Random Forest Classifier	0.972028	0.973899	0.973899	0.972130	0.174532	SHAP
LightGBM Classifier	0.972028	0.966143	0.966143	0.971913	0.158578	SHAP
Linear Discriminant Analysis	0.972028	0.962264	0.962264	0.971784	0.041888	ITSELF
Support Vector Classifier	0.965035	0.960587	0.960587	0.964965	0.012964	ITSELF
Stochatic Gradient Classifier	0.965035	0.960587	0.960587	0.964965	0.008975	ITSELF
Perceptron	0.965035	0.960587	0.960587	0.964965	0.010971	ITSELF
Quadratic Discriminant Analysis	0.958042	0.958910	0.958910	0.958195	0.011968	ITSELF
Logistic Regression	0.958042	0.955031	0.955031	0.958042	0.017953	ITSELF
Label Propagation	0.958042	0.955031	0.955031	0.958042	0.026802	ITSELF
Label Spreading	0.958042	0.955031	0.955031	0.958042	0.024933	ITSELF
Ridge Classifier CV	0.965035	0.952830	0.952830	0.964642	0.011969	ITSELF
Ridge Classifier	0.965035	0.952830	0.952830	0.964642	0.015958	ITSELF
Extra Trees Classifier	0.951049	0.945597	0.945597	0.950951	0.099735	SHAP
Passive Aggressive Classifier	0.944056	0.943920	0.943920	0.944260	0.011968	ITSELF
Linear Support Vector Calssifier	0.944056	0.943920	0.943920	0.944260	0.012965	ITSELF
Calibrated Classifier CV	0.958042	0.943396	0.943396	0.957460	0.030918	ITSELF
Extra Tree Classifier	0.937063	0.942243	0.942243	0.937581	0.008975	SHAP
K Nearest Neighbors Classifier	0.951049	0.937841	0.937841	0.950499	0.016955	ITSELF
Bagging Classifier	0.930070	0.936688	0.936688	0.930737	0.048868	SHAP
Bernoulli Naive Bayes	0.930070	0.932809	0.932809	0.930547	0.017946	ITSELF
Nu Support Vector Classifier	0.944056	0.932285	0.932285	0.943567	0.023936	ITSELF
Nearest Centroid	0.937063	0.926730	0.926730	0.936662	0.010971	ITSELF
Gaussian Naive Bayes	0.916084	0.910063	0.910063	0.916084	0.008977	ITSELF
Decision Tree Classifier	0.895105	0.905031	0.905031	0.896449	0.011968	ITSELF
AdaBoost Classifier	0.979021	0.979455	0.979455	0.979060	0.232496	ELI
CatBoost Classifier	0.979021	0.979455	0.979455	0.979060	7.602528	ELI
XGBoost Classifier	0.979021	0.975577	0.975577	0.978979	0.173537	ELI
Random Forest Classifier	0.972028	0.973899	0.973899	0.972130	0.174532	ELI
LightGBM Classifier	0.972028	0.966143	0.966143	0.971913	0.158578	ELI
Extra Trees Classifier	0.951049	0.945597	0.945597	0.950951	0.099735	ELI
Extra Tree Classifier	0.937063	0.942243	0.942243	0.937581	0.008975	ELI
Bagging Classifier	0.930070	0.936688	0.936688	0.930737	0.048868	ELI
AdaBoost Classifier	0.979021	0.979455	0.979455	0.979060	0.232496	LIME
CatBoost Classifier	0.979021	0.979455	0.979455	0.979060	7.602528	LIME
XGBoost Classifier	0.979021	0.975577	0.975577	0.978979	0.173537	LIME
Random Forest Classifier	0.972028	0.973899	0.973899	0.972130	0.174532	LIME
LightGBM lassifier	0.972028	0.966143	0.966143	0.971913	0.158578	LIME
Extra Trees Classifier	0.951049	0.945597	0.945597	0.950951	0.099735	LIME
Extra Tree Classifier	0.937063	0.942243	0.942243	0.937581	0.008975	LIME
Bagging Classifier	0.930070	0.936688	0.936688	0.930737	0.048868	LIME

Interpretability Based on SHAP

Figure 2. shows the output of the XGBoost model using SHAP as an interpretability tool.

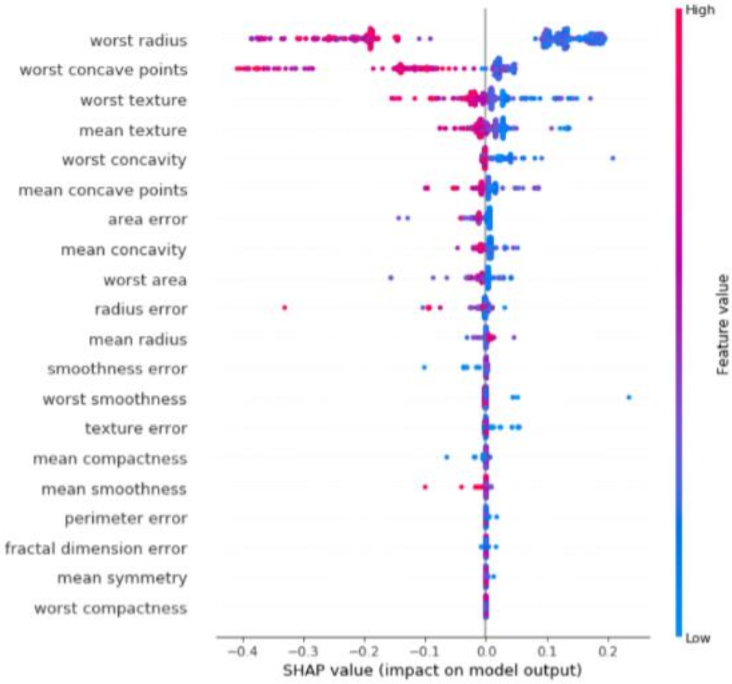
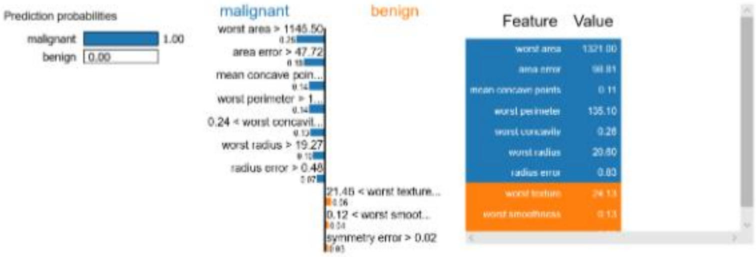


Figure 2. Interpretation of breast cancer prediction using SHAP with XGBoost algorithm

Interpretability Based on LIME

LIME, known as Local Interpretable Model-agnostic Explanation, is a surrogate diagnostic interpretability technique. LIME takes the surroundings and fits a basic interpretable linear model for a particular data point in the feature space. Figure 3 depicts the interpretability of LIME on the XGboost model predictions

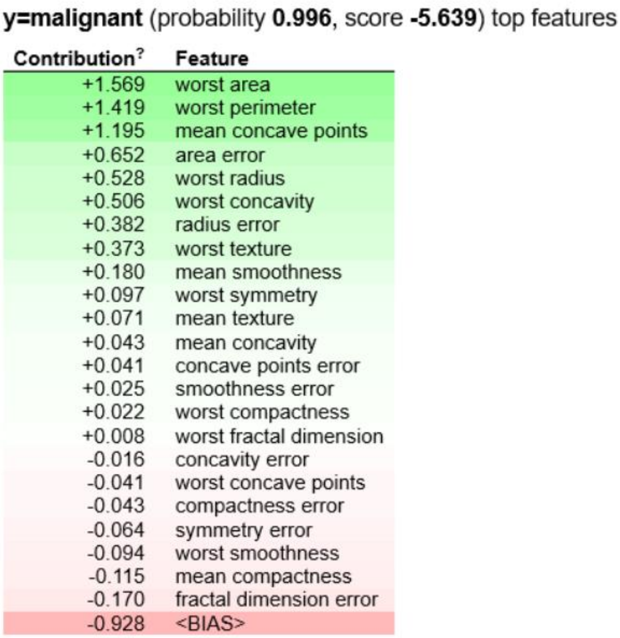


Feature	Value
worst area	1321.00
area error	98.81
mean concave points	0.11
worst perimeter	135.10
worst concavity	0.26
worst radius	20.80
radius error	0.83
worst texture	24.13
worst smoothness	0.13

Figure 3. Interpretation of breast cancer prediction using LIME

Interpretability Based on ELI5

ELI5 (ELI) is also a surrogate modeling technique used to debug machine learning classifiers and explain their top prediction via an easy-to-understand and good-visual way. However, it is not a complete model-agnostic explanations technique, mainly tree-based and other parametric linear models can only be used. The prediction is displayed as the sum of the top features plus a bias term.



y=malignant (probability **0.996**, score **-5.639**) top features

Contribution?	Feature
+1.569	worst area
+1.419	worst perimeter
+1.195	mean concave points
+0.652	area error
+0.528	worst radius
+0.506	worst concavity
+0.382	radius error
+0.373	worst texture
+0.180	mean smoothness
+0.097	worst symmetry
+0.071	mean texture
+0.043	mean concavity
+0.041	concave points error
+0.025	smoothness error
+0.022	worst compactness
+0.008	worst fractal dimension
-0.016	concavity error
-0.041	worst concave points
-0.043	compactness error
-0.064	symmetry error
-0.094	worst smoothness
-0.115	mean compactness
-0.170	fractal dimension error
-0.928	<BIAS>

Figure 4. Interpretation of breast cancer prediction using ELI

The Optimal Number of Clusters from the Elbow Method

The optimum number of clusters is 3; it is determined through the elbow method, as depicted in Figure 5 - the plot shows the sum of squared distances versus the number of clusters. The value at K = 3 represents the elbow of the curve, which is the optimum number of clusters.

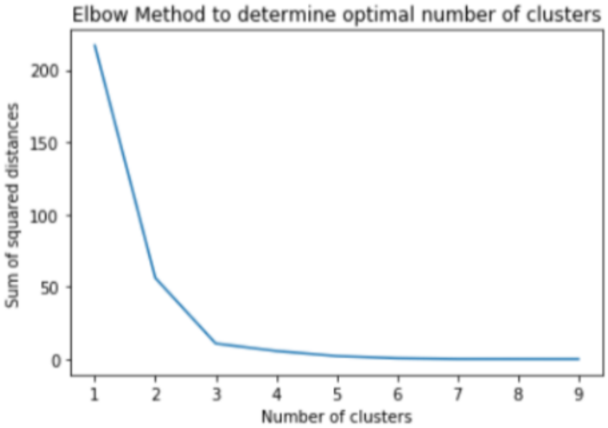


Figure 5. Sum of squared distances versus the number of clusters

The obtained silhouette score is 0.756, meaning good clustering performance

Clustering Outputs

Using the K-means clustering, we infer three types of model clusters, 0, 1, and 2. K is obtained through the elbow method using the above data. A closer look at those clusters' characteristics shows that models in cluster number 2 have low and medium interpretability and low computation. AdaBoost Classifier, XGBoost Classifier with ELI interpretability are the most performant and easily explainable modeling techniques for this diagnosis. The complete results are in Table 2. The clustering results show models falling into three clusters, 1, 2, or 3.

Table 2. K-means clustering of Breast Cancer Data Classification models

Model ID	Accuracy	Balanced Accuracy	AUC ROC	F1 Score	Run Time	Inter-pretability	Model Cluster
AdaBoost Classifier	0.979021	0.979455	0.979455	0.979060	0.232496	SHAP	1
CatBoost Classifier	0.979021	0.979455	0.979455	0.979060	7.602528	SHAP	2
XGBoost Classifier	0.979021	0.975577	0.975577	0.978979	0.173537	SHAP	1
AdaBoost Classifier	0.979021	0.979455	0.979455	0.979060	0.232496	ELI	3
CatBoost Classifier	0.979021	0.979455	0.979455	0.979060	7.602528	ELI	2
XGBoost Classifier	0.979021	0.975577	0.975577	0.978979	0.173537	ELI	3
AdaBoost Classifier	0.979021	0.979455	0.979455	0.979060	0.232496	LIME	1
CatBoost Classifier	0.979021	0.979455	0.979455	0.979060	7.602528	LIME	2
XGBoost Classifier	0.979021	0.975577	0.975577	0.978979	0.173537	LIME	1
Random Forest Classifier	0.972028	0.973899	0.973899	0.972130	0.174532	SHAP	1
LightGBM Classifier	0.972028	0.966143	0.966143	0.971913	0.158578	SHAP	1
Linear Discriminant Analysis	0.972028	0.962264	0.962264	0.971784	0.041888	ITSELF	3
Random Forest Classifier	0.972028	0.973899	0.973899	0.972130	0.174532	ELI	3
LighGBM Classifier	0.972028	0.966143	0.966143	0.971913	0.158578	ELI	3
Random Forest Classifier	0.972028	0.973899	0.973899	0.972130	0.174532	LIME	1
LightGBM Classifier	0.972028	0.966143	0.966143	0.971913	0.158578	LIME	1
Support Vector Classifier	0.965035	0.960587	0.960587	0.964965	0.012964	ITSELF	3
Stochastic Gradient Classifier	0.965035	0.960587	0.960587	0.964965	0.008975	ITSELF	3
Perceptron	0.965035	0.960587	0.960587	0.964965	0.010971	ITSELF	3
Ridge Classifier CV	0.965035	0.952830	0.952830	0.964642	0.011969	ITSELF	3
Ridge Classifier	0.965035	0.952830	0.952830	0.964642	0.015958	ITSELF	3
Quadratic Discriminant Analysis	0.958042	0.958910	0.958910	0.958195	0.011968	ITSELF	3
Logistic Regression	0.958042	0.955031	0.955031	0.958042	0.017953	ITSELF	3
Label Propagation	0.958042	0.955031	0.955031	0.958042	0.026802	ITSELF	3
Label Spreading	0.958042	0.955031	0.955031	0.958042	0.024933	ITSELF	3
Calibrated ClassifierCV	0.958042	0.943396	0.943396	0.957460	0.030918	ITSELF	3
Extra Trees Classifier	0.951049	0.945597	0.945597	0.950951	0.099735	SHAP	1
K Nearest Neighbors Classifier	0.951049	0.937841	0.937841	0.950499	0.016955	ITSELF	3
Extra Trees Classifier	0.951049	0.945597	0.945597	0.950951	0.099735	ELI	3
Extra Trees Classifier	0.951049	0.945597	0.945597	0.950951	0.099735	LIME	1
Passive Aggressive Classifier	0.944056	0.943920	0.943920	0.944260	0.011968	ITSELF	3
Linear SVC	0.944056	0.943920	0.943920	0.944260	0.012965	ITSELF	3
Nu SVC	0.944056	0.932285	0.932285	0.943567	0.023936	ITSELF	3
Extra Tree Classifier	0.937063	0.942243	0.942243	0.937581	0.008975	SHAP	1
Nearest Centroid	0.937063	0.926730	0.926730	0.936662	0.010971	ITSELF	3
Extra Tree Classifier	0.937063	0.942243	0.942243	0.937581	0.008975	ELI	2
Extra Tree Classifier	0.937063	0.942243	0.942243	0.937581	0.008975	LIME	1
Bagging Classifier	0.930070	0.936688	0.936688	0.930737	0.048868	SHAP	1
Bernoulli Naive Bayes	0.930070	0.932809	0.932809	0.930547	0.017946	ITSELF	3
Bagging Classifier	0.930070	0.936688	0.936688	0.930737	0.048868	ELI	3
Bagging Classifier	0.930070	0.936688	0.936688	0.930737	0.048868	LIME	1
Gaussian Naive Bayes	0.916084	0.910063	0.910063	0.916084	0.008977	ITSELF	3
Decision Tree Classifier	0.895105	0.905031	0.905031	0.896449	0.011968	ITSELF	3

Cluster 1 shows low computation runtime models while cluster 2 is the opposite; it offers high computation models. Cluster 3 depicts models with low runtime and self-explainable or relatively easy to explain interpretability tools.

Validation of Methodology

The technique described in this study assigns a cluster label to each model. Another XGBoost model is used to assess the validity of the labeling where the models' performance metrics represent the features, and the cluster categories represent the target (output). The XGBoost's model accuracy on a test set estimates the consistency of this labeling technique. The clustered dataset is randomly split into the train and test 80 and 20; the input variables represent the models' metrics. The target variable is the Cluster category. The distribution of clusters categories is shown in Figure 6. The accuracy on the test set points to perfect results 100%, as depicted in Figure 7

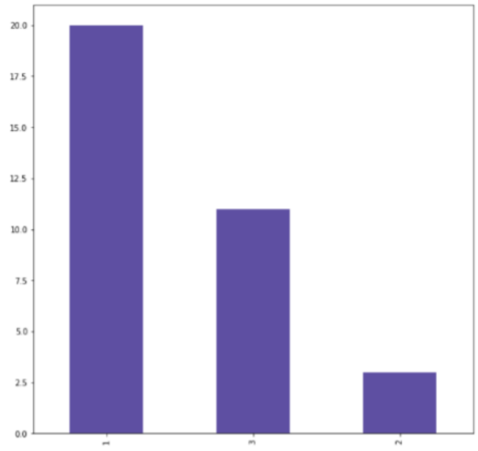


Figure 6. Test set's distribution of Clusters' categories

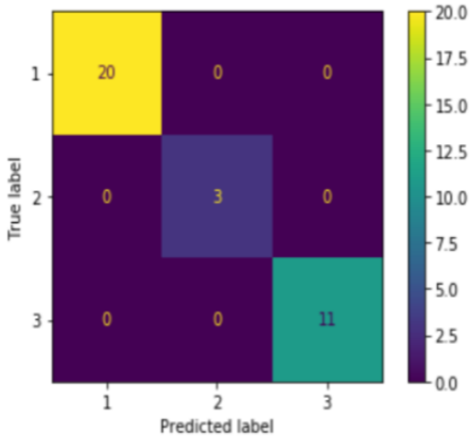


Figure 7. Confusion Matrix of Xgboost performance on the test set (Accuracy = 100%)

Discussion

The results show that a model can be chosen based on multiple criteria. While in the literature, many studies on Breast cancer diagnosis using machine learning focus on getting more accurate models based on a single criterion such as Accuracy [46] or F1-score, or AUC-ROC values; this study focuses on multiple criteria. In breast cancer diagnosis, accuracy is essential, and the prediction's explanation is also crucial. While we have used a particular dataset here and selected criteria, the technique applies to any dataset.

This study clearly shows distinct clusters; making sense of those clusters can become challenging when the number of constraints grows and could be a limitation of this proposed method. For a relatively low number of criteria or objectives, less effort is needed to characterize those clusters. In practice, decision-makers and modelers value fewer constraints, so the technique is beneficial. Using the K-Means clustering on this data is well justified as the average silhouette score and the elbow method output support it; however, it does not prevent the usage of other clustering techniques. Future studies should expand into using different clustering techniques. This technique could be perceived as a labeling technique. Essentially, it assigns a label that defines the cluster category of using a particular model. If the labeling is consistent, it must be learnable in terms of machine learning. A sophisticated machine learning algorithm such as XGBoost can be used to assess the technique. The models' performance metrics represent the features, and the cluster categories represent the target (output). The XGBoost's model accuracy on a test set estimates the consistency of this labeling technique. The final results dataset is split into train and test 80 and 20, respectively. The input variables represent the models' metrics. The target variable is the Cluster category—the accuracy on the test set point to perfect results 100%.

The K-means clusters' characteristics show that models in cluster number 2 have low and medium interpretability and low computation runtime. AdaBoost and XGBoost Classifiers with ELI5 interpretability are the most performant and most explainable models.

The proposed method shows a model could be selected based on multiple objectives by clustering the objectives. Those objectives could be numerical such as Accuracy, F1-score, Balance -Accuracy, or AUC-ROC. The objective could also be categorical such as the interpretability technique used on the model. This flexibility makes the method valuable and applicable to many domains where in practice, a single criterium is not enough to validate the selection of a machine learning model.

List of abbreviations

ML = Machine Learning
AI = Artificial Intelligence
SHAP = SHapley Additive exPlanations
LIME = Local Interpretable Model-Agnostic Explanations
ELI5 = Explain Like I'm 5
ELI = Explain Like I'm 5
NB= Naive Bayes
AUC = Area Under Curve
ROC = Receiver Operating Curve
CV = Cross-Validation
CART = Classification and Regression Trees
SVC = Support Vector Classifier
CNN = Convolution Neural Network

Conflict of Interest

The author declares that they have no conflict of interest.

Acknowledgments

This research was supported by the Watson School's System Science and Industrial Engineering Department at the State University of New York at Binghamton.

References

1. Samuel AL. Some studies in machine learning using the game of checkers. IBM Journal of Research and Development 1959;3(3):210-229.
2. Guidotti R, Monreale A, Ruggieri S, Turini F, Giannotti F, Pedreschi D. A survey of487methods for explaining black-box models. ACM Computing Surveys 2018:51(5):1-42.
3. Lundberg SM, Lee S-I. A unified approach to interpreting model predictions. NIPS'17 Proceeding of the 31st International Conference on Neural Information Processing Sysetem 2017, pp. 4765-4774.
4. Ribeiro MT, Singh S, Guestrin C. Anchors: High-precision model-agnostic explanations. Thirty-Second AAAI Conference on Artificial Intelligence 2018;32(1):1527-1535.
5. Abdolmaleki P, Buadu LD, Murayama S, Murakami J, Hashiguchi N, Yabuuchi H, Masuda K. Neural network analysis of breast cancer from MRI findings. Radiation Medicine 1997;15(5):283-294.
6. Abdolmaleki P, Buadu LD, Naderimansh H. Feature extraction and classification of breast cancer on dynamic magnetic resonance imaging using artificial neural network. Cancer Letters 2001;171(2):183-191.
7. Burke HB, Goodman PH, Rosen DB, Henson DE, Weinstein JN, Harrell Jr FE, Marks JR, Winchester DP, Bostwick DG. Artificial neural networks improve the accuracy of cancer survival prediction. Cancer 1997;79(4):857-862.
8. Ribeiro MT, Singh S, Guestrin C. "why should I trust you?": Explaining the predictions of any classifier. InProceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, 2016, pp. 1135-1144, New York, NY, USA.
9. Kaur H, Nori H, Jenkins S, Caruana R, Wallach H, Wortman Vaughan J. Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning. Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, CHI '20, 2020, pp. 4971-4914, New York, NY, USA.
10. Doshi-Velez F, Kim B. Towards a rigorous science of interpretable machine learning. 2017, arxiv.org/abs/1702.08608
11. Bau D, Zhou B, Khosla A, Oliva A, Torralba A. Network dissection: Quantifying interpretability of deep visual representations. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017, pp. 3319-3327, doi: 10.1109/CVPR.2017.354.
12. Shapley LS. A value for n-person games. Contributions to the Theory of Games 1953;2(28):307-317.
13. Hartigan JA. Clustering algorithms. John Wiley & Sons, Inc., 1975.
14. Kodinariya TM, Makwana PR. Review on determining number of cluster in k-means clustering.International Journal 2013;1(6):90-95.
15. Street WN, Wolberg WH, Mangasarian OL. Nuclear feature extraction for breast tumor diagnosis. IST/SPIE1993 International Symposium on Electronic Imaging: Science and Technology 1993;1905:861-870.
16. Mangasarian OL, Street WN, Wolberg WH. Breast cancer diagnosis and prognosis via linear programming. Operations Research 1995;43(4):570-577.
17. Wolberg WH, Street WN, Mangasarian OL. Machine learning techniques to diagnose breast cancer from fine-needle aspirates. Cancer Letters 1994;77:163-171.
18. Freund Y, Schapire RE, Naoki Abe. A short introduction to boosting. Journal-Japanese Society For Artificial Intelligence 1999;14(5):771-780.
19. Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A. CatBoost: unbiased boosting with categorical features. arXiv preprint (arXiv:1706.09516) 2017.
20. Chen T, Guestrin C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data 2016, pp. 785-794, doi:10.1145/2939672.2939785.
21. Breiman L. Random forests. Maching Learning, 2001;45(1):5-32.

22. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu TY . Lightgbm: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems 2017;30:3146-3157.
23. Balakrishnama S, Ganapathiraju A. Linear discriminant analysis-a brief tutorial. Institute for Signal and information Processing 1998, pp. 1-8.
24. Vapnik V, Guyon I, Hastie T. Support vector machines. Mach. Learn 1995;20(3):273-297.
25. Friedman J. Stochastic gradient boosting. Computational Statistics & Data Analysis 2002;38(4):367-378.
26. Utgoff PE. Perceptron trees: A case study in hybrid concept representations. Connection Science 1989:1(4):377-391.
27. Tharwat A. Linear vs. quadratic discriminant analysis classifier: a tutorial. International Journal of Applied Pattern Recognition 2016;3(2):145-180.
28. DeMaris A. Logistic regression. Handbook of Psychology 2003, pp. 509-532.
29. Raghavan UN, Réka A, Soundar K. Near linear time algorithm to detect community structures in large-scale networks. Physical Review E 2007;76(3):036106.
30. Zhou D, Bousquet O, Lal TN, Weston J, Scholkopf B. Learning with local and global consistency. Advances in Neural Information Processing Systems 16 (NIPS 2003) 2004;16:321-328.
31. Kim SH, Cho DH, Seok KH. Study on the ensemble methods with kernel ridge regression. Journal of the Korean Data and Information Science Society 2012;23(2):375-383.
32. Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Machine Learning 2006;63:3-42.
33. Crammer K, Dekel O, Keshet J, Shalev-Shwartz S, Singer Y. Online passive aggressive algorithms. Journal of Machine Learning Research 2006;7:551-585.
34. Herbster M. Learning additive models online with fast evaluating kernels. Proceedings of the Fourteenth Annual Conference on Computational Learning Theory, 2001, pp. 444-460.
35. Rejani Y, Thamarai Selvi S. Early detection of breast cancer using SVM classifier technique. International Journal on Computer Science and Engineering 2009;1(3):127-130.
36. Kull M, Silva Filho TM, Flach P. Beyond sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration. Electronic Journal of Statistics 2017;11(2):5052-5080.
37. Kramer O. K-nearest neighbors. Dimensionality reduction with unsupervised nearest neighbors. Springer, Berlin, Heidelberg, 2013, pp. 13-23.
38. Breiman L. Bagging predictors. Machine Learning 1996;24(2):123-140.
39. McCallum A, Nigam K. A comparison of event models for naive Bayes text classification. Proc. AAAI/ICML-98 Workshop on Learning for Text Categorization, 1998, pp. 41-48.
40. Platt JC. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. 1999 Available from: http://citeseer.ist.psu.edu/viewdoc/download?doi=10.1.1.41.1639&rep=rep1&type=pdf
41. Tibshirani R, Hastie T, Narasimhan B, Chu G. Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proceedings of the National Academy of Sciences of the United States of America, 2002;99(10):6567-6572.
42. Chan TF, Gene HG, LeVeque RJ. Updating formulae and a pairwise algorithm for computing sample variances. COMPSTAT 1982 5th Symposium held at Toulouse 1982. Physica, Heidelberg, 1982.
43. Breiman L, Friedman J, Olshen R, Stone C. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.
44. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011;12:2825-2830.
45. Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 1987;20:53-65.
46. Gupta P, Garg S. Breast Cancer Prediction using varying Parameters of Machine Learning Models. Procedia Computer Science 2020;171:593-601.