Comparison of Different Classifiers for Diabetes Diagnosis

Shuchang Ye[a], Enqi Liu[b]

The University of Sydney, Department of Information Technology, Sydney, Australia

[a]shye5505@uni.sydney.edu.au, [b]eliu5850@uni.sydney.edu.au

Keywords: Classifier, Machine Learning, Weka, Diabetes Diagnosis

Abstract: Machine learning algorithms provide several indispensable tools for intelligent medical data analysis. The paper provides a macroscopical comparison among different classifiers' performance in diabetes diagnosis. Representative and pervasive classifiers are chosen in several typical classifier categories, which are supported by Waikato Environment for Knowledge Analysis. The dataset used is the Pima Indians Diabetes Database, which is collected by the National Institute of Diabetes and Kidney Diseases in 1990. The high-level overview of the procedure of this study is data preprocessing, applying a classification algorithm, and estimating the performance. The paper briefly introduces the nature of each classifier and its application scenarios. The details of data preprocessing including feature selection are explained and the results of the outcome are discussed. The existing studies leave out the interpretability of classifiers which is crucial in medical prediction. To address the limitation of previous studies, this paper takes interpretability, and domain knowledge into consideration when estimating the performance of each model. The Naïve Bayes classifier achieves relatively high performance in this scenario.

1. Introduction

In machine learning, data are categorized into various classes through classification [1], with which, models are developed that use a set of training data to identify which category a new set of test data belongs to based on a set of predefined categories [2]. Considering diabetes is one of the leading causes of death worldwide, classification and characterization are crucial for treatment strategies. Machine learning is widely employed in the medical field since it aids in better prediction and decision-making. [3]. The importance of this study may include but is not limited to improving the accuracy and diagnosing time of medical diagnosis for this disease, inspiration for the public to prevent diabetes, providing reference advice for medical treatment plans, etc. However, selecting the appropriate classifier for a particular circumstance is not an easy problem to solve. First, there is a growing enormous number of algorithms available that fall into different categories. And there is a lack of methodologies that can help in recommending a given type of algorithm ahead of time for a specific kind of dataset. Besides, Most of these classification algorithms exhibit performance degradation when faced with datasets containing irrelevant and/or redundant features [2]. The quota for the performance of classifiers involves resilience for mistakes, the complexity of classifiers, real-time response accuracy, and so on.

This study aims to perform the medical diagnosis of Pima Indian's diabetes leveraging different machine learning classifiers. Diagnosing the problem with accurate results in an acceptable time is a major challenge in Bioinformatics. Even though the medical techniques for predicting diabetes are mature, they have limitations in effectiveness and popularity. In today's world, where medicine and technology are bumped, the use of computer science is one of the important tools to diagnose problems [4]. Instead of investigating the pathology underneath the disease, machine learning treats the medical diagnosis as a supervised learning problem. In this paper, we present the comparison of different classification techniques using the Waikato Environment for Knowledge Analysis or in short, WEKA, which is open-source software that consists of a collection of machine learning algorithms for data mining tasks [5].

A range of different types of classification algorithms is estimated in this paper. Bayes Classifier

  DOI: 10.25236/iwmecs.2022.012

(Naïve Bayes in this study), which belongs to the family of probabilistic Graphical Models (GM'S), is also known as Brief Network. This is primarily based on the concept of predicting the class in view of the values of members of the features [6]. Function Classifier (Multilayer perceptron and Support Vector Machine using SMO in this study), where the concept of neural network and regression is utilized, by which, the network is able to learn complex tasks by extracting progressively more informative features from the input patterns [7]. Lazy Classifier (K Nearest Neighbor using IBK in this study), demands storing complete training data, which results in high memory cost and inefficient prediction. Rules Classifier (OneR and ZeroR in this study), which has the feature that rules are mutually exclusive, are used to find the best boundary to achieve high accuracy. Trees Classifiers (Decision Tree by J48 and Random Forest in this study), are designed to search for optimal decision methods to minimize the overall probability of misclassification [8]. The accuracy, training time, and prediction time are estimated to weigh the performance of a certain classifier.

2. Data

2.1 Data Collection

The Pima Indian diabetes dataset is used for this project. The original owner for this dataset is the National Institute of Diabetes and Digestive and Kidney Diseases, which is received on 9 May 1990. All patients in this database are Pima-Indian women at least 21 years old and living near Phoenix, Arizona, USA [9]. This dataset contains 8 attributes and has 768 instances in total. The attributes include personal information: Number of times pregnant, Body mass index, Skin thickness, and Age; test measurements: glucose concentration, Diastolic blood pressure, 2-Hour serum insulin, and Diabetes pedigree function. Each row in the dataset relates to a particular patient. There are several modifications from the original data to the data we use in this study. The missing values are replaced with averages, and the class attribute is changed to nominal values. There are two classes in this dataset: yes and no, where 'yes' represents a person who tested positive for diabetes while 'no' represents a person who tested negative for diabetes. We have 500 instances in the 'no' class and 268 instances in the 'yes' class.

2.2 Data Pre-processing

Data pre-processing is a significant step in the knowledge discovery process, as advisable decisions must be based on rigorous data [10], but data in the real world is dirty, incomplete, and noisy [11]. Normalization is an important and necessary data pre-processing step to construct an accurate model. The primary goal is to reduce the bias of those attributes whose numerical value is greater in distinguishing pattern classes. The table, as shown in Fig. 1, illustrates that the mean and standard deviation difference between each attribute is existed and cannot be neglected. In order to avoid greater numeric attribute values dominating the smaller attribute values and overestimation, we need to apply normalization on our dataset to ensure the data is on the same scale. Weka's in-built normalization filter was used in our project to normalize values for each attribute. Weka's normalization function implemented in our study projects the values into the interval from 0 to 1, which scales the un-normalized data to a predefined lower and upper bounds linearly (which is called Minimum-Maximum Value Based Normalization Methods) [12], which means the method does not guarantee normal distribution. After the normalization, the entry becomes more rational and demonstrates an equal numerical contribution between all the attributes, as seen in Fig. 2 below:

STATISTICAL SUMMARY BEFORE NORMALIZATION IS THE ABBREVIATION OF STANDARD DEVIATION

Attribute	Statistical values	
	Mean	***Std***
Number of times pregnant	3.8	3.4
Plasma glucose concentration	121.7	30.4
Diastolic bolld pressure (mm Hg)	72.4	12.1

Attribute	Statistical values	
	Mean	***Std***
Triceps skin fold thickness (mm)	29.1	8.8
2-Hour serum insulin (mu U/ml)	155.3	85.0
Body mass index (weight in kg/m^2)	32.5	6.9
Diabetes pedigree function	0.5	0.3
Age (years)	33.2	11.8

Fig.1 Statistical summary for the Pima dataset before normalization

STATISTICAL SUMMARY AFTER NORMALIZATION

Attribute	Statistical values	
	Mean	***Std***
Number of times pregnant	0.226	0.198
Plasma glucose concentration	0.501	0.196
Diastolic bolld pressure (mm Hg)	0.494	0.123
Triceps skin fold thickness (mm)	0.240	0.096
2-Hour serum insulin (mu U/ml)	0.170	0.102
Body mass index (weight in kg/m^2)	0.292	0.141
Diabetes pedigree function	0.168	0.141
Age (years)	0.204	0.196

Std is the abbreviation of Standard deviation

Fig.2 Statistical summary for the Pima dataset after normalization

2.3 Feature Selection

Feature selection, as a data pre-processing strategy, has been demonstrated to be effective and efficient in preparing data (especially high-dimensional data) for various data mining and machine learning problems [13]. This process accelerates data mining algorithms, improves predictive accuracy, and increases comprehensibility. Irrelevant features are those that do not provide helpful information, and redundant features provide no more improvement to the model than the currently selected features [14].

Correlation-based feature selection evaluates the value of a subset of attributes by considering the individual predictive power of each feature and the degree of redundancy between them. Not only do correlation coefficients estimate the association between subsets of attributes and class, but also inter-correlations between the features [10]. In our study, we adopted CfsSubsetEval as our Attribute Evaluator and BestFirst as the searching method. CfsSubsetEval (CSE) measures the significance of attributes on their predictive power of attributes and their degree of redundancy. Subsets that have less relevance but are highly relevant to the target class have a relatively high priority. The first five attributes ranked are taken into consideration, i.e. check status, duration, history, credit limit, and savings status [15].

The procedure of attribute selection can be divided into two parts: ranking the attribute in the subsets and selecting the best one. BestFirst is a forward search algorithm which expands 5 nodes before terminating. In our study, 38 subsets are evaluated, and the merit of best subset found is 0.173. Finally, 5 attributes of 8 variables are selected:

1) Plasma glucose concentration 2 hours in an oral glucose tolerance test
2) 2-Hour serum insulin (mu U/ml)
3) Body mass index (weight in kg/(height in m)^2)
4) Diabetes pedigree function
5) Age (years)

3. Method

3.1 ZeroR Classifier

ZeroR is a learning algorithm which is used to test the results of the other learners. Rather than carefully selecting an attribute, the most common category is selected by ZeroR all the time. ZeroR learners are utilized to perform the comparison of the results of the other learners and determine whether they are meaningful or not, especially in the presence of one prominent dominating category [16].

3.2 OneR Classifier

OneR classifier, the abbreviation of 'One Rule', is a simple yet accurate classification algorithm. It is based on the calculated gain to generate one rule for each predictor in the data, and then selects the rule with the smallest total misclassification rate as its "one rule". OneR produces rules only slightly less accurate than classification algorithms prevailing now but produces rules that are simple for humans to understand [17].

3.3 K-Nearest Neighbor Classifier

The k-nearest neighbor's algorithm is a technique of classification, which is a very simple concept of machine learning. Generally, the k-nearest neighbor's algorithm is used for classification and regression. It classifies the input data by the method of majority rule. The nearest neighbor depends on the minimum distance point (e.g. Euclidean distance) [18].

3.4 Naïve Bayes Classifier

Bayes' theorem is the fundamental of the Bayesian classifier. Naive Bayesian classifiers have the assumption that the effect of an attribute value on a given pre-defined label is independent of the values of the other attributes, which is called class conditional independence. It is made to simplify the computation involved and release the influence of inter-relation, in this sense, is considered "naive" [19].

3.5 Multilayer Perceptron Classifier

Multilayer Perceptron classifier is an improvement of single neurons. MLP is consist of different layers where neurons are fully connected, which can produce complex boundaries. The backpropagation is leveraged to reduce the loss which is calculated by the loss function. In order to find the global minimum on a topographic map, trapping in a local optimum should be avoided, where come the optimizers? The optimizers do not guarantee to find the global minimum, but in practical use, they can often get a better performance. To approximate the global minimum faster, learning the entire topographic map is required.

3.6 Decision Tree Classifier (J48 Algorithm)

Decision tree classifiers are characterized by the use of one or several decision functions to classify unknown samples into a class in a continuous manner. The J48 algorithm is named as an optimized implementation of the C4.5 or an improved version of the C4.5. The output produced by J48 is the Decision tree. Decision trees have the same structure as trees having different nodes, such as the root node, intermediate nodes, and leaf nodes, which are connected by edges (the process of classification). Each node in the tree represents a condition for a decision and that decision brings about our result as the name is the Decision tree [20].

3.7 Random Forest Classifier

The Random Forest classifier is an ensemble classifier that uses a set of Classification And Regression Trees (CARTs) to perform supervised learning. The trees are created by drawing a subset of training samples with replacement, which is a bagging approach. The method to calculate the error is known as the out-of-bag (OOB) error. Each decision tree is independently generated without any pruning and each node is split using a user-defined partial of features (Mtry), which is selected at

random. Through developing the forest up to a user-defined number of trees (Ntree), the algorithm builds trees that have higher variance and lower bias. The final classification decision is made by estimating the average of (using the arithmetic mean) the class assignment probabilities calculated by all produced trees [21].

3.8 Support Vector Machine Classifier

A support vector machine (SVM) is a machine learning algorithm that is trained by mapping assigned labels to objects given attributes. In essence, an SVM is a mathematical entity, an algorithm (or recipe) for optimizing the trade-off of the margin width and the accuracy [22]. SVM maps input data into a high-dimensional feature space where it is more likely to be a linear separatable [23].

4. Result and Discussion

Cross-validation is the statistical method often used for evaluating and comparing different learning algorithms or classifiers. It is conducted by dividing the dataset into 2 groups: the training set and the testing set. The training set is designed to train the model while the testing set is designed to test and validate the model that is developed by the training set. K-fold cross-validation is a standard form of cross-validation and stratified cross-validation is an extension of regular cross-validation. Stratified cross-validation is leveraged in estimating the functionality of feature selection and the difference among a variety of classification algorithms. For the reproducible purpose, the random seed is set to be 1 and other attributes remain in default values. In this project, 10-fold stratified cross-validation was used to evaluate our implementations of K Nearest Neighbor (KNN) and Naive Bayes (NB). In 10-fold stratified cross validation, the data is divided into 10 folds and the class ratio remains the same as the original dataset. Then, we iterate 10 times of training and testing such that within each iteration a different fold of the data is used for testing while the remaining 9 folds are used for training.

A critical factor influencing whether a model is suitable for a particular scenario is assumption satisfaction. The Naïve Bayes classifier has the assumptions of independence and equality of importance. The attributes selected have no obvious association, yet they are not equally important. Thus, the more relaxed version, Naive Bayes, is applied. KNN assumes the object is predictable at query processing time, which is satisfied. The decision Tree introduces a relaxed assumption of Naïve Bayes independent requirement. It measures whether the ratio of positive prediction and negative prediction is affected by introducing another attribute, which is acceptable in our dataset. Cluster assumption and Manifold assumption are made for SVM, which strictly enforces data that share the same clusters belonging to the same class, which is against our data.

The classification methods selected from the WEKA to measure the performances are ZeroR, OneR, 1 Nearest Neighbor (1NN), 5 Nearest Neighbor (5NN), Naive Bayes (NB), Decision Tree (DT), Multilayer Perception (MLP), Support Vector Machine (SVM) and Random Forest (RF).

4.1 Feature Selection

Feature selection is an important step in data pre-processing since it can ideally prevent overfitting, filter informative attributes, reduce training and prediction time, and enhance performance. Especially when the dimension of the dataset is high, the density of data drops, which may not satisfy the assumptions of some classification algorithms. The accuracies for each classifier with and without feature selection are shown in Fig. 3.

ACCURACY ESTIMATION

Accuracy	**Classification Algorithm**								
	0R	***1R***	***1NN***	***5NN***	***NB***	***DT***	***MLP***	***SVM***	***RF***
CFS	65.1	70.8	67.8	74.5	75.1	71.7	75.1	76.3	75.3
No CFS	65.1	70.8	69.0	74.5	76.3	73.3	76.2	76.2	76.2

Fig.3 Accuracy comparison among different classifiers

Reduction in training time is one of the main reasons why feature selection is applied. The time complexity of training and prediction is estimated in terms of the number of variables. The majority

of running time lies in polynomials, even exponential of the number of attributes. Feature selection can control the time inside acceptable boundaries. The improvement of training time is estimated in our study as shown in Fig. 4.

TRAINING TIME ESTIMATION

Accuracy	Classification Algorithm								
	0R	***1R***	***1NN***	***5NN***	***NB***	***DT***	***MLP***	***SVM***	***RF***
CFS	0	0.02	0	0	0.01	0.09	0.81	0.1	0.18
No CFS	0	0	0	0	0	0.02	0.30	0.01	0.10

0 represents the time is less than 0.01

Fig.4 Training time comparison among different classifiers

The table shows the benefits comparing previously normalized data with after feature selection data. NB, DT, MLP, SVM, RF have remarkable promotions, while other classifiers' decrease in training time is not obvious. The nature of this phenomenon is that rule classifiers focus on a single attribute and lazy classifiers simply store information. Not only does feature selection reduce training time, but also it saves time in predicting new data.

The kappa statistic aims to measure interrater reliability. The importance of rater reliability rests with the fact that it represents the level of which the data collected in this research are proper representations of the attributes measured. This estimation is important when deciding whether a model can be applied to medical applications. Cohen suggested the Kappa result be interpreted as follows Fig. 5 [25].

KAPPA STATISTIC

Kappa statistic	***Reliability***
≤ 0	No agreement
0.01~0.20	None to slight
0.21~0.40	Fair
0.41~0.60	Moderate
0.61~0.80	Substantial
0.81~1.00	Almost perfect agreement

Fig.5 The inpretation of kappa statistic

KAPPA STATISTIC ESTIMATION

Accuracy	Classification Algorithm								
	0R	***1R***	***1NN***	***5NN***	***NB***	***DT***	***MLP***	***SVM***	***RF***
CFS	0	0.32	0.29	0.43	0.44	0.39	0.46	0.44	0.45
No CFS	0	0.32	0.31	0.43	0.41	0.42	0.47	0.45	0.47

Fig.6 Training time comparison among different classifiers

The Fig. 6 records the changes of kappa statistics before and after feature selection. The result demonstrates a non-decreasing variation in Kappa statistics. The Kappa statistic plays an important role in deciding whether the model can be applied to medical studies. Cohen claims the Kappa statistic is supposed to be higher than 0.41. Simple rule classifiers and 1NN fail the requirement of Kappa statistic.

Furthermore, overfitting is another factor that should be taken into consideration. Overfitting is an error occurring when a classification model is too closely aligned to an in-sample class, while reducing the functionality of predicting new data. In our study, an experiment on DT is performed to investigate the impact of feature selection on overfitting problems. The J48 algorithm is adopted in this experiment setting the confidence level to 95%. The accuracy before and after the feature selection is noteworthy:

$$71.3572\% \rightarrow 73.3073\%$$

The feature selection has the capacity of remitting the effect of overfitting to some extent.

The use of machine learning in supporting medical diagnosis is growing in popularity. However,

most of the time, we need not only the result, but also a pathology behind it, in other words, interpretation. The attributes selected by correlation-based feature selection are listed below:

1) glucose concentration: traditional diabetes diagnosis lies in the glucose concentration. The approximate boundaries of glucose concentration in the diagnosis of diabetes are set in Fig. 7 [26].

GLUCOSE CONCENTRATION

Glucose concentration	***Diagnosis result***
$\leq 7.7\ mmol/L$	No diabetes defined
$7.7{\sim}11.1\ mmol/L$	Borderline diabetes
$> 11.1\ mmol/L$	Diabetes

Fig.7 The diagnosis decision on glucose concentration

2) 2-Hour serum insulin: Insulin is a hormone produced in the pancreas, which is associated with glucose tolerance. The role of insulin is to control glucose levels in the blood.

3) Body mass index: Body mass index (BMI) has consistently been associated with adverse health outcomes, including non-insulin dependent diabetes.

4) Diabetes pedigree function: Non-insulin-dependent (type II) diabetes mellitus (NIDDM) is characterized by hyperglycemia and insulin resistance and affects nearly 5% of the general population. Inherited factors are important for its development, which comes from the diabetes pedigree function.

5) Age: Age is not a factor causing diabetes, while diabetes is more common in the elders. Especially type II diabetes has long been regarded as a condition that affects older people.

All the attributes have either been demonstrated as significant parameters in diagnosing insulin-dependent diabetes or non-insulin-dependent diabetes or have been demonstrated to have linkage to.

4.2 Classifier Implementation

The motivation for implementing classifiers is that the training time itself is not enough to estimate the performance of a particular classifier. The time consuming for predicting new data is often considered more important than the duration to build up a model because building models can be completed in advanced, while timely feedback of results is critical in the medical field. Delayed results will lead to symptoms that cannot be effectively treated at an early stage, which will lead to aggravation of symptoms and even loss of the opportunity to save a precious life. Unfortunately, Weka does not support classification time analysis. Therefore, we implement the classifier from two typical categories: lazy learning (KNN) and eager learning (NB) as Fig. 8.

ACCURACY AND TIME

statistic	**My1NN**		**My1NN**		**My1NN**	
	CFS	***No CFS***	***CFS***	***No CFS***	***CFS***	***No CFS***
Accuracy	69.14	69.06	76.04	74.74	76.56	73.17
Training	0	0	0	0	0	0
Predicting	10.53	10.55	7.42	10.73	0.58	0.84

0 represents the time is less than 0.01

Fig.8 Accuracy, training and classification time estimation

The accuracy is estimated with the same strategy of Weka (stratified 10-fold cross validation). Python time module is implemented for calculating the training time and classification time, which is run in Macbook Pro of M1 Pro chip and 32G Random Access Memory. At each iteration of the stratified 10-fold cross validation, the training time and testing time are recorded.

Even though the concepts behind our classifiers and WEKA's classifiers are almost the same, in implementation level, there exists slight differences. For KNN, when dealing with numeric value, the nearest neighbor is determined by the Euclidean distance in our implementation, while WEKA does one step forward. It maps the difference to the normal distribution curve and calculates the probability difference. Because in data preprocessing, the normalization is performed, the approximation releases this operation. For Naïve Bayes, this implementation assumes normal distribution and adopts a

probability density function, however, WEKA estimates the probability based on the real distribution of data. The attributes' values are normalized, which makes this implementation reasonable.

The result is consistent with what we perform by Weka. It can be further confirmed that CFS can greatly improve the operating efficiency without affecting the accuracy. The classification time of lazy learning algorithm is many times longer than that eager learning algorithm while they have almost same efficiency of training. The reason behind the phenomenon is eager learning methods construct classification while lazy learning methods simply store the data and generalizing beyond these data is postponed until an explicit request is made. Besides, lazy classifiers construct different approximations each time while eager classifiers use the same approximation every time. As medical diagnosis is in a complete problem domain with efficient response requirement, the lazy learning is not suitable for our scenario.

4.3 Classification Algorithm Comparison

The performance between classifiers will be assessed from 5 aspects: accuracy, PRC area, F-measure, training time and prediction time, and Kappa statistic. The accuracy for each classifier are rounded to 2 decimal places and recorded in Fig. 9. The graph shows that the ZeroR has the lowest accuracy 65.10%. This is reasonable. ZeroR classifier only look at the target and ignores all other predictors, it predicts the class using the majority vote. Hence, it can only correctly predict the majority class. The classifier that has the highest accuracy 76.69% is the Support Vector Machine (SVM). Moreover, the Nave Bayes (NB) has accuracy of 76.30%, Multilayer Perception (MLP) has accuracy of 76.17% and Random Forest (RF) has accuracy of 76.17%. The difference between the accuracy of these classifier is tiny.

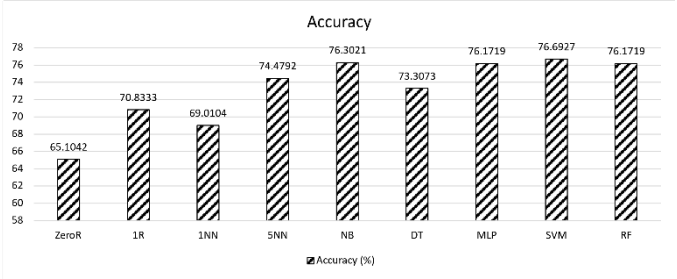


Fig.9 Accuracy estimation among different classifiers

However, there are some limitations involved if evaluating the performance of each classifier is only based on the accuracy. The accuracy is defined as the ratio between the total number of correct predictions and the total number of predictions. This is illustrated in the formula for the Accuracy displayed below.

$$Accuracy = \frac{True\ Positive + True\ Negative}{The\ amount\ of\ samples} \quad (1)$$

Thus, accuracy only cares about the correctness of the class prediction, it is not sensitive to class distribution. There are 500 instances for class 'no' and 268 instances for class 'yes' in diabetes. Thus, this dataset is unbalanced. Hence, more metrics need to consider to be added here to evaluate the performance. PRC area computes the area under the Precision-Recall curve (PRC) and PRC will demonstrate the relationship between Precision and Recall and F-measure computes the harmonic mean of Precision and Recall. The below formula illustrates the definition of Precision and Recall.

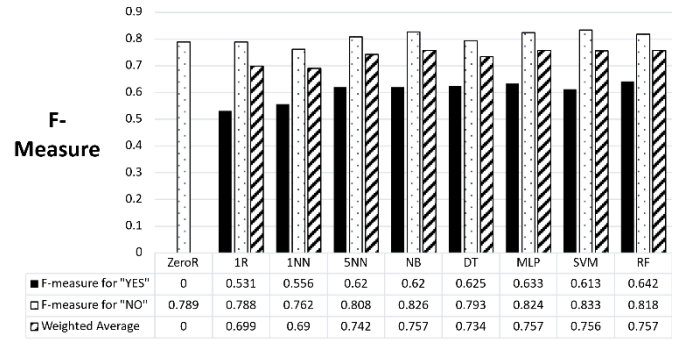
$$Precision = \frac{True\ Positive}{True\ Positve + False\ Positive} \quad (2)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Nagative} \quad (3)$$

In this project, Precision measures the ratio between the number of patients that the classifier correctly predict has diabetes and the number of all the patients that the classifier predict has diabetes. Recall measures the ratio between the number of patients that the classifier correctly predict has diabetes and the number of all the patients who have diabetes. In a real-world context, people are trying

to prevent the cases that a patient having diabetes to be classified as having no diabetes as this may cause the patient's condition worse, which means high recall. However, consider another case that the treatment is only helpful for people who have diabetes and harmful for people who do not have diabetes. Thus, people are also trying not to cure people that do not have the disease, this means high Precision. But the Precision and Recall cannot be satisfied at the same time. As we mentioned earlier, PRC and F-measure will demonstrate the relationship between precision and recall. Therefore, the PRC area and F-measure will then be some good evaluation metrics to use in this scenario.

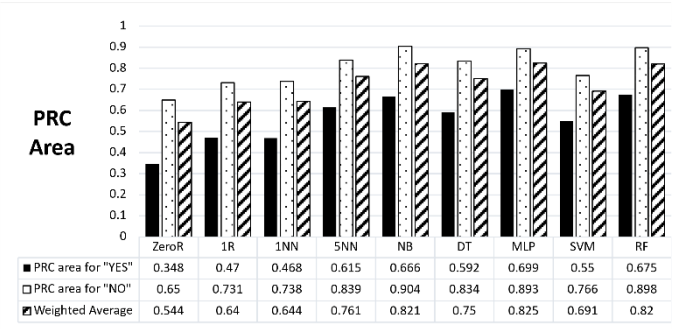
As what shown in Fig. 10, the black column represents the F-measure for class 'yes' for each classifier while the white column represents the F-measure for class 'no' for each classifier. It can be observed that the F-measure for 'no' is higher than the F-measure for 'yes' for every classifier. The reason for this may contribute to the smaller number of 'yes' instances compare to the 'no' instances. The classifiers NB, MLP, and RF have the highest value in F-measure for weighted average.



	ZeroR	1R	1NN	5NN	NB	DT	MLP	SVM	RF
F-measure for "YES"	0	0.531	0.556	0.62	0.62	0.625	0.633	0.613	0.642
F-measure for "NO"	0.789	0.788	0.762	0.808	0.826	0.793	0.824	0.833	0.818
Weighted Average	0	0.699	0.69	0.742	0.757	0.734	0.757	0.756	0.757

Fig.10 F-measure estimation among different classifiers

The black column represents the PRC Area for class 'yes' for each classifier while the white column represents the PRC Area for class 'no' for each classifier, as seen in Fig. 11. The PRC area for 'no' is also higher than the PRC Area for 'yes' for every classifier. The reason for this is the same as situation in F-measure that discussed before. NB has the highest value in PRC Area for class 'no' over these classifiers and MLP has the highest value in PRC Area for class 'yes' and also the weighted average over other classifiers.



	ZeroR	1R	1NN	5NN	NB	DT	MLP	SVM	RF
PRC area for "YES"	0.348	0.47	0.468	0.615	0.666	0.592	0.699	0.55	0.675
PRC area for "NO"	0.65	0.731	0.738	0.839	0.904	0.834	0.893	0.766	0.898
Weighted Average	0.544	0.64	0.644	0.761	0.821	0.75	0.825	0.691	0.82

Fig.11 F-measure estimation under different classifiers

Training time, as an important factors, significantly impacts the usability of models. Weka rounds the training time into 2 decimal places. The bar plot shown in Fig. 12 illustrates the MLP takes the highest training time of 0.28 seconds compared to others. This is reasonable as MLP uses backpropagation to train the data. The time complexity of backpropagation is $O(n \times m \times h^k \times o \times i)$where n represents the number of training samples, represents the number of features, represents the number of hidden layers, represents the number of neurons, represents the number of output neurons and represents the number of iterations. Thus, if there are many hidden layers, the backpropagation will be very time-consuming. The RF has the second-highest training time, this is also reasonable as the RF usually will build a large number of trees which will make the algorithm very slow. The training time for 1NN, 5NN, ZeroR, OneR and NB is expected to be short. 1NN and 5NN are K-Nearest Neighbour classifiers (KNN) with different K. The KNN classifier is known as a Lazy learning classifier which will store all training examples and start building when a new example

needs to be classified. Thus, there is no model built into the training process only storing training examples. The ZeroR takes the majority class as its predictions and OneR generates one rule for each predictor and picks the rule that produces the smallest number of errors. Thus, ZeroR and OneR are both simple and easy to compute algorithms. Moreover, NB has excellent computational complexity as it only requires a single scan of the training data to calculate all the data needed. The actual training time for 1NN, 5NN, ZeroR, OneR and NB are all less than 0.01 seconds which validates the expectation. In addition, the training time for DT and SVM is fair which are 0.02 seconds and 0.01 seconds, respectively.

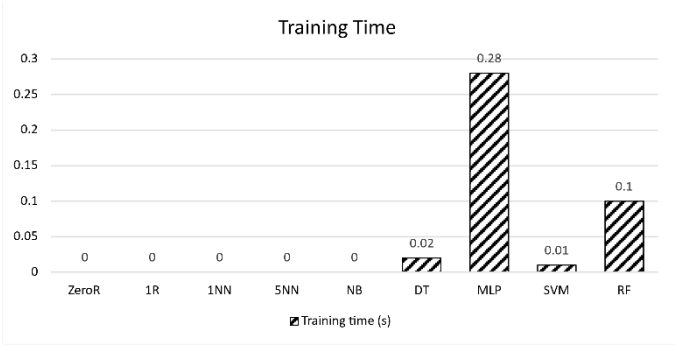


Fig.12 Training time estimation among different classifiers

The result of the kappa statistic for the different classifiers is evaluated, as shown in Fig. 12. The kappa statistic measures the ratio between the difference between observed accuracy and expected accuracy and the difference between 1 and expected accuracy. See formula below:

$$Kappa = \frac{Observed\ Accuracy - Expected\ Accuracy}{1 - Expected\ Accuracy} \quad (4)$$

Thus, the kappa statistic is normalized at the baseline of random chance on our dataset. All the classifiers' kappa statistic should be higher than 0.41 according to Cohen's suggestion. The graph in Fig 13 shows that most of the classifiers have a Kappa statistic that is higher than 0.41 except ZeroR, OneR and 1NN. This also makes sense in real-life, classifiers like ZeroR, OneR and 1NN are meaningless in helping medical treatment. For example, the ZeroR classifier gives predictions based on the majority class. Therefore, the resulting predictions will be all the people has the disease or all the people do not have the disease which is meaningless and useless.

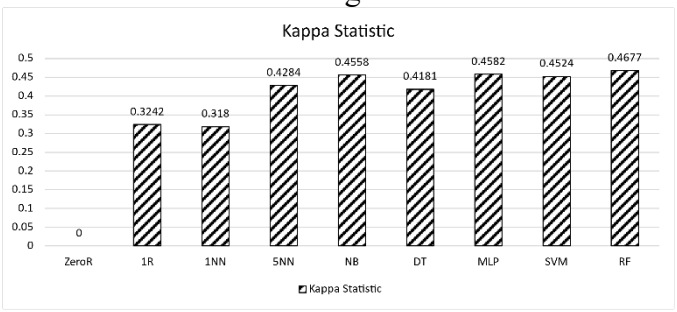


Fig.13 Kappa statistic estimation among different classifiers

In this part, the potential advantages, disadvantages and their potential application scenario is discussed. ZeroR and OneR are very simple and computationally cheap algorithms. They often work well in many real-world problems. For example, in this case, only one aspect is sufficient to predict its class. However, this is not applied to the study of diabetes. Many aspects will determine whether a person has diabetes or not, such as the concentration of serum insulin and Diabetes Pedigree. Thus, the performance of ZeroR and OneR is poor. K-Nearest Neighbors is easy to implement and explain to people who do not understand the technical aspects. However, it is very sensitive to the value of K. Thus, the process of finding the most appropriate 'k' is a big challenge for KNN. This study only investigates 1NN and 5NN for K-Nearest Neighbors. The results show that 5NN performs better than 1NN in every aspect. However, we cannot conclude that 5 is the most appropriate choice for KNN for predicting diabetes, further experiment and validation process is needed. The disadvantage of KNN is

that it is very slow for large datasets. The result from this study do not demonstrate this point as the dataset used in this study is on a small scale. However, diabetes is one of the leading causes of death worldwide, many people suffer from it, and the dataset is huge in real-life. Moreover, it also does not work well on unbalanced data. Our dataset is unbalanced and thus, its prediction performance is not very well compared to others. Naïve Bayes is based on Baye's theorem. It could predict real-time predictions as it only requires a single scan of the training data to calculate all the data needed which is very fast. Moreover, it is scalable to large datasets and still well-performed when data has a large number of features. However, the attributes selected from the diabetes dataset have no obvious association thus they are not equally important. Thus, the more relaxed version of Naive Bays (NB) is used in this study. Correlated attributes will reduce the power of Naïve Bayes. The use of feature selection at the beginning to identify and discard all the correlated attributes is to help reduce the effect caused by the correlated attributes. The results show that the feature selection process did somehow bring a positive impact on the Naive Bayes classifier. The performance of NB is excellent in every aspect compared to other classifiers. The Decision Tree (DT) classifier has many advantages such as explicit classification rules, which people outside the field could also easily understand, and both numeric and categorical data can be used in DT. However, DT also has some disadvantages. Firstly, they are very precarious. A single change in the dataset may lead the tree to grow on a completely different path. Moreover, overfitting easily occurs. This is because the decision tree aimed to perfectly classify every training example and thus it will grow each branch of the tree as deep until the training example is excellently classified. This process will make the tree only memorize the data instead of building patterns. This is also the reason why DT does not work well on a small dataset. In this study, the diabetes dataset is small, thus the DT does not have enough representative examples to construct a model that can well generalize on the new data. Thus, its accuracy and F-measure are not desirable. The Random Forest (RF) classifier is an ensemble classifier and follows a bagging approach. RF will produce numerous decision trees and combines the predictions from them. This overcomes the overfitting problem of the DT and thus its performance should behave better than DT. Our study results validate this expectation. The accuracy, F-measure, and PRC Area all seem better than the DT. However, the drawbacks come out. RF usually will build a large number of trees which will make the algorithm very slow and RF usually hard to interpret as it is a combination of multiple decision trees. The Neural Network (MLP) is designed to draw boundaries without shape limitation to minimize the loss. Almost all the algorithms and classifiers can be approached by two hidden layers, which makes the Neural Network seem to be omnipotent. However, because of the complexity of network and neurons, the neural network can only provide a prediction result rather than give a decision procedure, and both the training time and classification time is much slower than the average. This is fatal to medicine, because diagnosis and treatment need to provide evidence and the rationale behind it. Support Vector Machines (SVM) is a very popular classifier. It can form linear and non-linear decision boundaries. Thus, it is very useful when the classes are separable. Weka has many different kernel tricks for SVM such as Normalized Poly Kernel, Poly Kernel, Precomputed Matrix Kernel, Puk kernel, RBF kernel, and String kernel. Only Normalized Poly Kernel, Poly Kernel, and Puk kernel can handle numeric attributes and thus can be used for the diabetes datasets. Except for the kernel listed above, there is a range of different kernels that can be used in SVM. Therefore, SVM can handle non-linear data using an appropriate kernel such as RBF (Radial basis function) kernel. However, the drawbacks come out. The process is time-consuming, and it is very hard to find the most appropriate kernel function. Moreover, SVM is also hard to interpret by people because it sometimes involves a change in dimensionality.

5. Conclusion

In consequence, this study performed a medical diagnosis of Pima Indian's Diabetes using different machine learning classifiers. Feature selection is an important step in pre-processing the data as it can filter informative attributes, reduce training and prediction time and ideally prevent overfitting. The statistical results have also proven that the use of feature selection in this study has significantly improved the training time and accuracy which coincides with the aim of this study.

These machine learning classifiers are evaluated in terms of accuracy, PRC Area, F-measure, training time and prediction time, and Kappa statistic. According to the statistical summary and analysis discussed above, the best classifier chosen for the Pima Indian's Diabetes dataset is the Naive Bayes. Naive Bayes has the second-highest accuracy over these classifiers. Although Support Vector Machine has the highest accuracy, its PRC area is much lower compared to Naive Bays for both classes positive and negative diagnosis. Naive Bayes also has the highest F-measure, short training time, and acceptable classification time compared to Support Vector Machine and other classifiers as well. Its kappa statistic well satisfies Cohen's requirement.

Except for the Naiver Bayes classifier, Random Forest and Support Vector Machines also perform well in predicting Pima Indian's Diabetes. The accuracy and PRC Area for both of them are high, the training time is acceptable, and the kappa statistic is also satisfied Cohen's requirement. Although the Multilayer perceptron has good results in many aspects such as accuracy, PRC Area, F-measure, and the kappa statistic. Its long training time will be a serious issue here. The two classifiers that do not recommend for helping medical diagnosis are ZeroR and OneR. The use of machine learning in supporting medical diagnosis is growing in popularity. This study helped diagnose diabetes with accurate results in an acceptable time to a certain context and provide complete and important information on relative classifier performance which can be used by others to investigate further.

The future works are investigating the medical interpretation behind the model and building a complete automatic diagnosis and treatment system with higher accuracy. We will study the association of selected features with diabetes and the relationship between features, meanwhile, continue to pay attention to relative medical trends and look for more information which supports the diagnosis of diabetes. The accuracy in this study is approximately 75%, which is the result of lack of various samples and critical factors. In the next step, we will enrich the pima dataset by gathering data from different areas and get model replenished through including more variables. The model is going to be refined through medical experiments and advice from experts in related fields. The speed of response, accuracy in practical application, stability, and ability to deal with extreme cases will be measured in the actual medical circumstance.

References

[1] Priya, S., et al., Chapter 8 - Early detection of Parkinson's disease using data mining techniques from multimodal clinical data, in Advanced Machine Vision Paradigms for Medical Image Analysis, T. Gandhi, et al., Editors. 2021, Academic Press. p. 213-228.

[2] Morán-Fernández, L., V. Bólon-Canedo, and A. Alonso-Betanzos, *How important is data quality? Best classifiers vs best features.* Neurocomputing, 2022. 470: p. 365-375.

[3] Ramzan, M. Comparing and evaluating the performance of WEKA classifiers on critical diseases. in 2016 1st India International Conference on Information Processing (IICIP). 2016.

[4] Chokkalingam, S.P. and K. Komathy. Comparison of different classifier in WEKA for rheumatoid arthritis. in 2013 International Conference on Human Computer Interactions (ICHCI). 2013.

[5] Bin Othman, M.F. and T.M.S. Yau. Comparison of Different Classification Techniques Using WEKA for Breast Cancer. in 3rd Kuala Lumpur International Conference on Biomedical Engineering 2006. Berlin, Heidelberg: Springer Berlin Heidelberg.

[6] Garg, T. and S.S. Khurana. Comparison of classification techniques for intrusion detection dataset using WEKA. in International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014). 2014.

[7] kumari Dash, R., *Selection of the best classifier from different datasets using WEKA.* International Journal of Engineering Research & Technology (IJERT), 2013. 2(3).

[8] Kurzyński, M.W., *The optimal strategy of a tree classifier.* Pattern Recognition, 1983. 16(1): p. 81-87.

[9] Kayaer, K. and T. Yildirim. Medical diagnosis on Pima Indian diabetes using general regression neural networks.

[10] Karegowda, A.G., A.S. Manjunath, and M.A. Jayaram, *Comparative study of attribute selection using gain ratio and correlation based feature selection.* International Journal of Information Technology and Knowledge Management, 2010. 2(2): p. 271-277.

[11] Singhal, S. and M. Jena, *A study on WEKA tool for data preprocessing, classification and clustering.* International Journal of Innovative technology and exploring engineering (IJItee), 2013. 2(6): p. 250-253.

[12] Singh, D. and B. Singh, *Investigating the impact of data normalization on classification performance.* Applied Soft Computing, 2020. 97: p. 105524.

[13] Li, J., et al., *Feature Selection: A Data Perspective.* ACM Comput. Surv., 2017. 50(6): p. Article 94.

[14] Kumar, V. and S. Minz, *Feature selection: a literature review.* SmartCR, 2014. 4(3): p. 211-229.

[15] Gnanambal, S., et al., *Classification Algorithms with Attribute Selection: an evaluation study using WEKA.* International Journal of Advanced Networking and Applications, 2018. 9(6): p. 3640-3644.

[16] Devasena, C.L., et al., *Effectiveness evaluation of rule based classifiers for the classification of iris data set.* Bonfring International Journal of Man Machine Interface, 2011. 1(Special Issue Inaugural Special Issue): p. 05-09.

[17] Lakshmi, S.V. and T.E. Prabakaran, *Performance analysis of multiple classifiers on KDD cup dataset using WEKA tool.* Indian Journal of Science and Technology, 2015. 8(17): p. 1-10.

[18] Bharati, S., M.A. Rahman, and P. Podder. *Breast Cancer Prediction Applying Different Classification Algorithm with Comparative Analysis using WEKA*. in *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*. 2018.

[19] Leung, K.M., *Naive bayesian classifier.* Polytechnic University Department of Computer Science/Finance and Risk Engineering, 2007. 2007: p. 123-156.

[20] Amin, M.N. and A. Habib, *Comparison of different classification techniques using WEKA for hematological data.* American Journal of Engineering Research, 2015. 4(3): p. 55-61.

[21] Belgiu, M. and L. Drăguţ, *Random forest in remote sensing: A review of applications and future directions.* ISPRS Journal of Photogrammetry and Remote Sensing, 2016. 114: p. 24-31.

[22] Noble, W.S., *What is a support vector machine?* Nature Biotechnology, 2006. 24(12): p. 1565-1567.

[23] Acır, N., *A support vector machine classifier algorithm based on a perturbation method and its application to ECG beat recognition systems.* Expert Systems with Applications, 2006. 31(1): p. 150-158.

[24] Refaeilzadeh, P., L. Tang, and H. Liu, *Cross-validation.* Encyclopedia of database systems, 2009. 5: p. 532-538.

[25] McHugh, M.L., *Interrater reliability: the kappa statistic.* Biochemia medica, 2012. 22(3): p. 276-282.

[26] Jouven, X., et al., *Diabetes, glucose level, and risk of sudden cardiac death.* European Heart Journal, 2005. 26(20): p. 2142-2147.