Comparison of Classification Algorithms for Predicting Autistic Spectrum Disorder Using WEKA Modeler

SITI FAIRUZ MOHD RADZI (✉ fairuzradzi@usm.my)

Universiti Sains Malaysia

MOHD SAYUTI HASSAN

Universiti Sains Malaysia

Muhammad Abdul Hadi Mohd Radzi

Universiti Sains Malaysia

Abstract

Background: In healthcare area, big data, if integrated with machine learning, enables health practitioners to predict the result of a disorder or disease more accurately. In Autistic Spectrum Disorder (ASD), it is important to screen the patients to enable them to undergo proper treatments as early as possible. However, difficulties may arise in predicting ASD occurrences accurately, mainly caused by human errors. Data mining, if embedded into health screening practice, can help to overcome the difficulties. This study attempts to evaluate the performance of six best classifiers, taken from existing works, at analysing ASD screening training dataset.

Result: We tested Naive Bayes, Logistic Regression, KNN, J48, Random Forest, and Deep Neural Network (DNN) algorithms to ASD screening dataset and compared the classifiers' based on significant parameters; sensitivity, specificity, accuracy, receiver operating characteristic (ROC), area under the curve (AUC), and runtime, in predicting ASD occurrences. We also found that most of previous studies focused on classifying health-related dataset while ignoring the missing values which may contribute to significant impacts to the classification result which in turn will impact the life of the patients. Thus, we addressed the missing values by implementing imputation method where they are replaced with the mean of the available records found in the dataset.

Conclusion: We found that J48 produced promising results as compared to other classifiers when tested in both circumstances, with and without missing values. The outcome is hoped to assist health practitioners in making accurate diagnosis of ASD occurrences in patients.

Introduction

Autistic Spectrum Disorder (ASD), which was first described in 1940, is a group of neuro-developmental disorders that commonly occurs in boys (Bhaumik et, al, 2010). Children with ASD will commonly develop a number of peculiar behavioural changes and will have difficulties in social interaction and communication which will become more apparent when the children reach six months of age (Gold et. al, 2006; CDC, 2020; Bolton, 2012). There is a need to ensure that ASD is diagnosed at the earliest stage in order to facilitate and prepare the parents or caregivers of the affected children with appropriate interventions as soon as possible (Bolton, 2012). Early interventions for patients diagnosed with ASD are important so as to ensure that the patients and caregivers are able to deal with the environmental circumstances during the growing-up phase of the children. Thus, prompt classification of ASD in children is very important which makes the integration of screening tool with data mining tool crucial. A systematic review conducted by Marlow et. al (2019) showed that the screening tools that have been used by healthcare practitioners have yet to incorporate machine learning intervention intensively. According to Pinto-Martin et. al (2005), there is a need to ensure that a tool is incorporated in the development of ASD screening tool for rapid and accurate diagnosis since medical practitioners may probably have difficulties to keep up with the screening process all by themselves which can lead to human errors and in this case, machine learning method can help to solve the issue (Akay, 2009).

There is an urgent need to ensure that machine learning is incorporated in healthcare area to assist the practitioners, however, not many studies have been conducted to examine the sensitivity, specificity, and accuracy of machine learning techniques for ASD screening process which makes the integration nearly impossible. This has been mentioned by Thabtah (2019) who compiled a number of ASD screening-related datasets which can be analysed through machine learning for improving the sensitivity, specificity, and accuracy in diagnosing ASD patients. In other similar research, Thabtah (2017) found that the algorithms in machine learning can assist practitioners in expediting the process of screening and diagnosing a patient with ASD. Thus, we attempt to fill in the gap by conducting an experiment to compare the best classification algorithms to be used for ASD screening process.

1.1 Related Works

Ramotra et. al (2020) conducted a study to predict the presence of heart disease by utilizing classification techniques namely Decision Tree, Naïve Bayes, Support Vector Machines, and Artificial Neural Networks classifiers. The study concluded that Naive Bayes classifier presented the highest accuracy among other classifiers being tested. A study conducted by Kibis et. al (2017) utilized artificial neural network (ANN), classification and regression tree (C&RT), Logistic Regression, and Bayesian Belief Network (BBN) to investigate breast cancer survivability with respect to the classifiers' accuracy, sensitivity, specificity, and area under the curve (AUC) metric. The study found that Logistic Regression method presented better performance with regard to AUC metrics. Alaiad et. al (2020) tested five classification algorithms, namely, Naive Bayes, Decision Tree, Support Vector Machine, K-Nearest Neighbour (KNN), and JRip in predicting chronic kidney disease. The study found that KNN achieved high performance in term of the accuracy as compared to other

classification algorithms. In a study conducted by Dawngliani et. al (2020) to classify benign and malignant tumors for breast cancer screening dataset, J48 decision tree outperformed other decision tree algorithms in terms of the accuracy. The study tested J48 with other decision tree algorithms namely Decision Stump, Random Forest Tree, REP tree, Hoeffding Tree and Logistic Model Tree (LMT). In other study conducted by Mun and Jumadi (2020), Naïve Bayes, Decision Table and Random Forest were tested to compare which algorithm produced the best statistical performance in classifying dyslexia dataset. The study has shown that Random Forest produced 100% accuracy and specificity in classifying dyslexia subject. Raj and Masood (2020) classified ASD screening dataset by using Naïve Bayes, Support Vector Machine, Logical Regression, KNN, Neural Network, and Deep Convolutional Neural Network (CNN). The study concluded CNN to be the best algorithm for handling classification as it has the highest accuracy as compared to other algorithms being tested.

The related works mentioned above showed that different classification techniques outperformed other classification techniques when tested to health related datasets. In relation to the mentioned related works, this study is intended to test the best classification technique from each existing study to the the ASD screening dataset in order to investigate which classification technique can produce better performance in screening the ASD class. We propose to investigate the performance of Naive Bayes, Logistic Regression, KNN, J48, Random Forest, and Deep Neural Network (DNN) algorithms by comparing their accuracy, specificity, sensitivity, receiver operating characteristic (ROC), and AUC results. We argue that Deep Neural Network (DNN) is most suitable to be used in replacement of CNN due to the fact that the ASD screening dataset contained no visual images or text embedding which does not fulfil the requirement for utilizing CNN. It is evident that CNN is mainly used to analyse and recognize patterns in images and text embeddings of a dataset by utilizing the input data's collocated grids (Lang et. al, 2019; Segal-Rozenhaimer et. al, 2020; Panahi et. al, 2020). Thus, we replaced CNN with DNN instead.

Missing values are common in data analysis especially when big data from different sources is involved. Missing values or data inconsistency can potentially affect how data is analysed, and thus, it should be addressed by embedding several methods; deletion or data imputation (Brown and Kros, 2003; Shahzad et. al, 2017; Hruschka et. al, 2005; Pampaka et. al, 2016). Data imputation is a method whereby missing values are replaced with meaningful values such as the mean of the available records. This study aims at investigating the effect of ignoring missing values in the ASD dataset to the sensitivity, specificity, and accuracy of each classification algorithm used in this study since there exists a lack of studies being conducted to investigate this matter. Data imputation method has been utilized to address the missing values and to compare the sensitivity, specificity, and accuracy before and after the method was applied to the ASD dataset.

This paper is organized as follows; research method section explains the approaches that we used to classify the ASD screening dataset including the classification process, classification techniques, performance matrix, and dataset involved in this study, result and discussion section presents the results of the classification process using WEKA Explorer and WEKA Experimenter to the dataset in two conditions; with and without missing values, conclusion section concludes the findings and suggests several potential studies that can be conducted in the future.

Method

The main purpose of this study is to analyse a set of ASD screening data by using six classification algorithms to help in improving the diagnosis process of ASD in healthcare practices. This section demonstrates the approaches that we used to classify the ASD screening dataset including the classification process, classification techniques, performance matrix, and dataset involved in this study.

Since the output of the training dataset is known which is the screening class, classification technique is the most suitable one to be used in this case. Classication can help to improve screening process and assist at reducing any possible errors caused by inexperienced health practitioners (Akay, 2009; Al-Askar, 2016). Figure 1 illustrates the process of classifying the dataset by using six chosen classifiers. The process began by classifying the ASD screening dataset in two conditions; with and without missing values, using six classifiers individually with Weka Explorer. As proposed by Tsai and Chang (2016), data imputation method was used to deal with incomplete data by replacing the missing values with reasonable values. Once the results were obtained, we furthered the experiment by running all six classifiers simultaneously in Weka Experimenter and the results were analysed and compared for both conditions.

2.1 Classification techniques

For the purpose of this study, six classifiers techniques have been chosen to analyse their performances in classifying ASD screening dataset. The classifiers used include Naive Bayes, Logistic Regression, KNN, J48, Random Forest, and CNN. Each of these classifiers have shown the highest performance when tested with other classifiers in previous studies.

Naïve Bayes uses Bayesian techniques to build a simple network in which it is assumed that the probability that one new example belongs to a class based on the assumption that all attributes are self-reliant from each other (Chen et. al, 2020; van der Heide et. al, 2019). According to Lewis (1998), Naive Bayes algorithm can produce an optimal classifier in different situations where the assumptions are wildly violated. Logistic Regression is a multivariate statistical method which requires less assumption. The method is useful for evaluating the relationship between independent variables and dependent variables as well as for predicting the risk of a disease based on predictor variables built within the model (Ghazvini et. al, 2019).

K-nearest Network (KNN), or known as IBk in WEKA modeler environment, is an instance-based learning algorithm that is capable at classifying many types of dataset as well as capable of performing distance weighting. In KNN, when k samples belong to a type of category and it is found that the samples of the dataset are closely identical to other k samples, the samples will be classified into the category (Chen et. al, 2020). The classifier will examine the distance of the samples to the sample of the nearest neighbouring training in the feature space (Madeti and Singh, 2018). J48 is one of the decision tree techniques implemented from C4.5 algorithm. It consists of four main programmes; the decision tree generator, production rule generator, decision tree interpreter, and production rule interpreter. The decision tree produced by C4.5 is utilized for classification and grouping purposes. This algorithm works by producing a decision tree based on the dataset by recursive division of the data followed by the development of the decision tree using Depth-First strategy (Quinland, 1993).

Random Forest, used for classifying and regressing an observation, is one of the decision tree techniques which is constructed by using a different bootstrap sample from the original data. The technique which roots from decision tree algorithm consists of a huge number of individual decision trees which work as an ensemble (Yiu, 2019). Approximately one-third of the data will be left out of the bootstrap sample and not utilized in the construction of the kth tree (Breiman and Cutler, 2004). Similar to other classifiers that have been previously mentioned, Deep Neural Network (DNN) is one of the types of neutral network algorithms which can also be used for both classification and regression cases. It has been widely used as a tool to classify data in many crucial fields which can be extended from speech recognition to computational medicine (Rad et. al, 2020). The neural network incorporates the training of neural network with an amplified cross-entropy error function (Ganapath et. al, 2013).

2.2 Dataset

This study utilized a set of ASD screening dataset provided by Dr. Fadi Fayez Thabtah from the Department of Digital Technology, Manukau Institute of Technology, New Zealand (Thabtah, 2017). The researcher developed a screening app used to screen the symptoms of ASD in patients and the data of the patients, which have been proven important in screening ASD occurrence (Thabtah, 2018), were collected and stored in a designated database.

The dataset is made up of 21 attributes, which includes 292 instances or records. The attributes or data fields include age, gender, ethnicity, jaundice status, existence of autism among family member, relation, country of residence, used screening app before, type of screening method, questions about the performance or ability of the patient (A1_Score, A2_Score, A3_Score, A4_Score, A5_Score, A6_Score, A7_Score, A8_Score, A9_Score, A10_Score), screening score, and screening class. The diagnosis result from the screening test is categorized into "Yes" for a patient diagnosed with ASD and "No" for a patient not diagnosed with ASD and due to the categorization, the dataset is categorized as asymmetric dataset. The dataset contains 90 or 30.82% missing values in the age, ethnicity, and relation attributes. Age is one of the most significant information needed in ASD screening as it assists practitioners to determine the type of early treatment that the patients should be undergoing if they are diagnosed with ASD (Delehanty et. al, 2020; Jones et. al, 2014; Thabtah, 2018). Thus, it is important to address the missing value. Table 1 shows the list of the attribute names and types contained in the ASD screening dataset.

Table 1. Attributes of ASD screening dataset

Attributes name	Attribute code	Description	Attributes type
Age	age	The child's age	Number
Gender	gender	The child's gender (Male or Female)	String
Ethnicity	ethnicity	List of common ethnicities in text format	String
Born with jaundice	jaundice	Whether the child was born with jaundice	Boolean (yes or no)
Family member with ASD	autism	Whether any immediate family member has ASD	Boolean (yes or no)
Relation	relation	Parent, self, caregiver, medical staff, etc.	String
Country of residence	contry_of_res	Name of countries in text format	String
Used the app before	used_app_before	Whether the user has used a screening app	Boolean (yes or no)
Screening Method Type	age_desc	The type of screening methods chosen based on age category (0=toddler, 1=child, 2= adolescent, 3= adult)	Integer (0,1,2,3)
Question 1	A1_Score	Question asking "Does your child look at you when you call his or her name?"	Binary (0, 1)
Question 2	A2_Score	"How easy is it for you to get eye contact with your child?"	Binary (0, 1)
Question 3	A3_Score	"Does your child point to something to indicate that he or she wants something?"	Binary (0, 1)
Question 4	A4_Score	Does your child point to something to share interest with you?	Binary (0, 1)
Question 5	A5_Score	Does your child pretend? (eg: talk on the toy phone)	Binary (0, 1)
Question 6	A6_Score	Does your child follow where you are looking at?	Binary (0, 1)
Question 7	A7_Score	If you or someone else in the family is visibly upset, does your child show sign of wanting to comfort them?	Binary (0, 1)
Question 8	A8_Score	How would you describe your child's first word?	Binary (0, 1)
Question 9	A9_Score	Does your child use simple gesture? (example: waving for goodbye)	Binary (0, 1)
Question 10	A10_Score	Does your child stare at nothing with no apparent purpose?	Binary (0, 1)
Screening Score	result	The final score obtained based on the scoring algorithm of the screening method used. This was computed in an automated manner	Integer
Screening Class	Class/ASD	Whether the child has ASD or not	Boolean (yes or no)

When the ASD screening dataset is loaded to WEKA modeler environment, the modeler will recognize the attributes and during the scanning process of the dataset, it will compute basic statistics on each attribute in the dataset. The left panel highlighted in green

(Fig. 2) shows the list of attributes for the dataset. The modeler shows that the dataset contains 21 attributes and 292 instances in total.

2.3 Performance metrics

In this study, the accuracy result of each classifier is used as the performance metrics for the purpose of analysing each classifier's performance more systematically. The AUC metric is also examined for all classifiers used in this study. The metric is used to measure the performance of each classifier over the full range of sensitivities and specificities and is not affected by the trade-off between accuracy and specificity (van der Heide et. al, 2019).

i) Confusion matrix

Confusion matrix is used to evaluate the performance of the classification algorithms. It is a visualization tool used to illustrate the accuracy, specificity, and sensitivity of the classifiers in classification process. The matrix illustrates the relationships between the outcome and prediction of classes shown in Table 2.

Table 2
Confusion matrix

Actual		Predicted	
		Positive	Negative
	Positive	a	b
	Negative	c	d

The description of the confusion matrix shown in Table 2 is shown as follows:

- a is True Positive (TP) where it is predicted as having ASD and having ASD in actual situation
- b is False Negative (FN) where it is predicted as not having ASD but having ASD in actual situation
- c is False Positive (FP) where it is predicted as having ASD but not having ASD in actual situation
- d is True Negative (TN) where it is predicted as not having ASD and not having ASD in actual situation

In data mining, accuracy, calculated using the formula; $(TP + TN)/(TP + FP + TN + FN)$, is defined as the proportion of the total number of data that is correctly classified. It is computed by dividing the total sum true positive and true negative with the total sum of all positive and negative data. Sensitivity or recall or true positive rate, calculated using the formula; $TP/(TP + FN)$, calculates the proportion of actual positives that are correctly classified, for instance, the percentage of patients that are correctly identified as having ASD. To calculate accuracy from the confusion matrix, the total number of data categorized as true positive is divided with the total sum of true positive and false negative data. Specificity or true negative rate calculates the proportion of actual negatives that are correctly classified, for instance, the percentage of patients correctly identified as not having ASD. The formula used is $TN/(TN + FP)$. Sensitivity is calculated from true negative and false positive data. Specificity also allows practitioners to determine the probability of false alarm.

ii) ROC and AUC

The performance of each algorithm can also be measured by comparing the result of ROC and AUC. ROC plots the classification algorithm's sensitivity and specificity at different classification thresholds in a visual format. Meanwhile, AUC measures the area below the ROC curve which is depicted as the probability that the classification algorithm ranks a random positive data more highly than a random negative data.

2.4 Pre-processing and processing measures

The ASD screening dataset contains missing values in three attributes; age, ethnicity, and relation, thus the study tested the classifiers on the dataset with and without missing values to investigate the effect of both conditions on the classifiers' performance result. The missing values were addressed during the pre-processing phase by deploying the most used imputation method which is imputation

using mean values. Imputation using mean values is relevant in this study since the missing values in the ASD screening dataset are categorized as missing completely at random (MCAR), thus reducing the biasness to the classification result (Bennett, 2001).

After the pre-processing stage, cross-validation mode was performed to the dataset by splitting the data into k-block or a chunk of objects and this process was performed before classification process took place. Once the modeler split the dataset, the data was tested by each classifier in which it was trained to use K-1 block and the process was reiterated for all the blocks. For this study, the ASD screening dataset was split into ten blocks.

Results And Discussion

This section presents the results of the classification process when we tested the classifiers using WEKA Explorer and WEKA Experimenter to the dataset in two conditions; with and without missing values by which the results are presented and discussed accordingly.

3.1 Testing classifier performance using WEKA Explorer

The study tested six different types of classifiers which include Naïve Bayes, Logistic Regression, KNN, J48, Random Forest, and DNN. Table 3 illustrates the performance of the classifiers tested in WEKA Explorer by deploying cross-validation test mode. For classification with missing values, the area under ROC curve (AUC) of each classifier has also been illustrated for comparison as shown in Fig. 5.

Table 3
Weighted average of classifiers for ASD dataset with missing values

Classifier	Training runtime (sec)	Correctly Classified Instance % (Accuracy)	Incorrectly Classified Instance %	Kappa Statistic	TP Rate	FP Rate	Precision (Sensitivity)	Recall (Specificity)	ROC	AUC
Naïve Bayes	0.03	98.9726	1.0274	0.9794	0.990	0.010	0.990	0.990	1.000	0.9996
Logistic Regression	0.51	95.2055	4.7945	0.904	0.952	0.048	0.952	0.952	0.991	0.9893
KNN	0.00	88.3562	11.6438	0.7674	0.884	0.114	0.886	0.884	0.894	0.8939
J48	0.08	100.0000	0.0000	1.000	1.000	0.000	1.000	1.000	1.000	1.000
Random Forest	0.76	100.0000	0.0000	1.000	1.000	0.000	1.000	1.000	1.000	1.000
DNN (2-layer)	28.93	86.9863	13.0137	0.7393	0.870	0.131	0.870	0.870	0.928	0.9282

We furthered the study by classifying the ASD screening dataset through the removal of the missing values in the dataset. This was conducted by using one of the imputation methods which is imputation using mean values. In WEKA modeler environment, the said imputation method is represented as "Replace Missing Value" filter (Fig. 3). In the pre-processing phase, the missing values were replaced with the mean value of the attribute that contained missing values, before being classified using the classifiers. The result is organized in Table 4. We also illustrated the AUC of each classifier for comparison in Fig. 4.

Table 4
Weighted average of classifiers for ASD dataset without missing values

Classifier	Training runtime (sec)	Correctly Classified Instance % (Accuracy)	Incorrectly Classified Instance %	Kappa Statistic	TP Rate	FP Rate	Precision (Sensitivity)	Recall (Specificity)	ROC	AUC
Naïve Bayes	0.03	98.9726	1.0274	0.9794	0.990	0.010	0.990	0.990	1.000	0.9997
Logistic Regression	0.11	95.2055	4.7945	0.9040	0.952	0.048	0.952	0.952	0.991	0.9893
KNN	0.00	89.3836	10.6164	0.7878	0.894	0.105	0.895	0.894	0.897	0.8969
J48	0.01	100.0000	0.0000	1.0000	1.000	0.000	1.000	1.000	1.000	1.000
Random Forest	0.27	100.0000	0.0000	1.0000	1.000	0.000	1.000	1.000	1.000	1.000
DNN (2-layer)	26.29	86.9863	13.0137	0.7393	0.870	0.131	0.870	0.870	0.928	0.9282

From the experiments, all classifiers obtained different results when classifying the ASD screening dataset with and without missing values. Naïve Bayes classifier produced slightly different AUC score when classifying the dataset with and without imputation, dismissing the previous studies (Patil, 2016; Kanchana, 2016) which claimed that Naïve Bayes is insensitive to missing values. Based on the classification result of all the classifiers being tested in Table 3, when missing values were ignored, J48 and Random Forest have shown 100% performance as compared to other classifiers in terms of the accuracy, sensitivity, and specificity. However, J48 only required 0.08 second to classify the dataset as compared to Random Forest (0.76 seconds). Meanwhile, DNN produced the lowest accuracy result (86.9863%) in classifying the dataset with missing values in comparison to KNN (88.3562%), Logistic Regression (95.2055), and Naïve Bayes (98.9726%). In terms of the computational complexity to build the classifying model, DNN has shown the worst performance in which it required 28.93 seconds to build the model followed by Random Forest (0.76 seconds), Logistic Regression (0.51 seconds), J48 (0.08 seconds), and Naïve Bayes (0.03 seconds). Despite the fastest run time (0.00 second) in classifying the dataset, KNN failed to classify 11.6438% of instances correctly when missing value were ignored, the second highest after DNN.

Based on Table 4, when the missing values were replaced with the mean values, both J48 and Random Forest classifiers have shown constant performance similar to when missing values were ignored except for the time taken to build the model which has increased for both classifiers; 0.01 seconds for J48 and 0.27 seconds for Random Forest. The time taken for all classifiers to build the model has also increased with KNN producing the fastest speed to build the model (0.00 seconds) followed by J48 (0.01 seconds), Naïve Bayes (0.03 seconds), Logistic Regression (0.11 seconds), Random Forest (0.27 seconds), and DNN (26.29 seconds). However, KNN failed to classify 10.6164% of instances correctly as compared to other classifiers that required more time to classify the instances but can produce better accuracy.

Further, we compared the ROC score to study the performance of each classifier. ROC enables the demonstration of the diagnosis ability of a classifier in which the main goal is to have the curve closer to the value 1 on the Y-axis of the graph which indicates the ability of a classifier to classify a given dataset accurately. Simply put, the closer the curve to the value 1, the better the performance of the classifier at classifying a dataset. J48, Naïve Bayes, and Random Forest produced the best ROC curve which indicates that the classifiers can predict ASD occurrences accurately as opposed to Logistic Regression, KNN, and DNN when classifying the dataset in both conditions. From the ROC curve, we further illustrated the AUC of each classifier shown in Figs. 3 and 4. The experiment showed that when the dataset with missing values is classified, J48 and Random Forest produced the best AUC with 1.000 score followed by Naïve Bayes (0.9996), Logistic Regression (0.9893), DNN (0.9282), and KNN (0.8939). When testing the dataset without the missing values, Naïve Bayes and KNN increased their AUC performances by 0.0001 from 0.9996 to 0.9997 and by 0.03 from 0.839 to 0.869 respectively. The result indicates that the classifiers that could not produce a perfect score of 1.000 for AUC are unable to predict ASD occurrences accurately, this in turn, may affect the life of a patient being wrongly diagnosed.

3.1 Verifying Classifier Performance using WEKA Experimenter

We further the experiment by classifying the dataset with and without missing values by utilizing WEKA Experimenter in order to validate the results produced by the classifiers when tested in WEKA Explorer environment. WEKA Experimenter allows more than one classifier to be tested simultaneously to classify the dataset. We tested the dataset by running 10-fold cross-validation test mode.

Corrected Paired T-Test mode was utilized to verify the performance of each classifier by comparing the accuracy and AUC results of each of the classifiers when tested simultaneously. The test works by comparing each classifier in pairs and making reasonable assumptions with regards to the distribution of the results collected. The test used J48 classifiers as the test base with 0.05 two-tailed confidence level and the result is shown in Table 5.

Table 5
Accuracy and AUC score of all classifiers tested simultaneously in WEKA Experimenter (with missing values)

Classifier	Correctly Classified Instance % (Accuracy)	AUC
Naïve Bayes	98.84	1.00
Logistic Regression	95.51	0.99
KNN	88.42	0.89
J48	100.00	1.00
Random Forest	99.97	1.00
DNN (2-layer)	94.52	0.98

Based on the test conducted using WEKA Experimenter, 600 data were loaded since each classifier was evaluated 100 times (10-fold cross validation multiplied by 10 repetitions). We compared the accuracy and AUC score of each classifier when tested in both WEKA Explorer and WEKA Experimenter. The result shows that J48 classifier outperformed other classifiers in terms of its accuracy and AUC readings; 100% and 1.00. On the other hand, the accuracy of Naïve Bayes and Random Forest decreased while Logistic Regression, KNN, and DNN improved their accuracy especially DNN that produced 7.5337% of improvement. We furthered the test by using J48 as the test base and the result shows that no classifiers outperformed the test base, except Random Forest, which generated 1.00 of AUC result, akin to J48.

In addition, the accuracy of Logistic Regression (95.51%), DNN (94.52), and KNN (88.42%) as well as the AUC of KNN (0.89) and DNN (0.98) have shown to be significantly different than that of J48. Based on the test using both WEKA Explorer and WEKA Experimenter, it is evident that J48 is the best classifier among the other five classifiers with 95% of confidence level when tested with the dataset containing missing values. This shows that J48 is capable at handling dataset that contains missing values. Further, a study conducted by Aziz et. al (2017) tested Naïve Bayes, J48, and Random Forest to classify a non-medical dataset and the classification results were compared. Based on the result, they found that Naïve Bayes produced the best accuracy in classifying the instances as opposed to this study. Thus, we can suggest that a classifier's classifying performance is affected by the type of dataset as well as the number of instances involved in the experiment.

We then addressed the missing values by using imputation method and tested the dataset in WEKA Experimenter. We tested the dataset without missing values by deploying the same method through which we used Corrected Paired T-Test by using 0.05 significance level and compared the performance of each classifier in terms of the accuracy and AUC score. Since J48 turned out to be the best classifier among other classifiers being tested, we used the classifier as the test base for the experiment. The result of the accuracy and AUC of the classifiers is illustrated in Table 6.

Table 6. Accuracy and AUC score of all classifiers tested simultaneously in WEKA Experimenter

(without missing values)

Classifier	Correctly Classified Instance % (Accuracy)	AUC
Naïve Bayes	98.91	1.00
Logistic Regression	95.51	0.99
KNN	89.08	0.90
J48	100	1.00
Random Forest	100	1.00
DNN (2-layer)	94.52	0.98

As shown in Table 6, we found that J48 and Random Forest, both of which have 100% accuracy and 1.00 AUC score, have outperformed other classifiers when missing values were replaced with the mean values of the attributes containing missing values. The experiment has illustrated that Logistic Regression, KNN, and DNN to be significantly different than that of the test base in terms of the accuracy. Meanwhile, in terms of the result of AUC, KNN and DNN have shown to be significantly different than the test base. By comparing the classification result of ASD screening dataset in both conditions, the study shows that Random Forest can perform better when the dataset contains no missing values since the classifier has shown an improvement from 99.97% ("with missing values" condition) to 100.00% ("without missing values" condition) of reading when classifying the ASD class correctly. This is due to its nature of being capable at providing the best estimates of the variables that are deemed important in the classification. Besides, the classifier is also capable at estimating the missing values and maintaining its accuracy when the amount of missing values is massive. On the other hand, Logistic Regression and DNN are not affected by the existence of missing values in the dataset. We also found that Naive Bayes improved its accuracy when missing values were replaced with the mean value of the attribute. This study has somewhat debunked the study which showed that Naïve Bayes has low sensitivity towards the existence of missing values in a dataset (Radzi et. al, 2018). However, a further study is sought to test its sensitivity using other datasets.

d) Confusion Matrix of the Best Classifier

As mentioned previously, confusion matrix is a visualization tool that allows researchers to illustrate the accuracy, specificity, and sensitivity of a given classifier. From the experiment, confusion matrix is produced to illustrate the accuracy, specificity, and sensitivity of J48 classifier, the best classifier among other classifiers being tested. The confusion matrix of J48 classifier is shown in Fig. 5 and Fig. 6 for both conditions; ASD screening dataset with and without missing values, respectively.

The confusion matrix which is produced when J48 is used to classify ASD screening dataset with and without missing values illustrates that in both conditions, J48 could classify 151 instances with 'yes' class category and 141 instances with 'no' class category correctly. The matrix also shows that neither false positive nor false negative classification was produced by J48, which proves that J48 can perform better in classifying the database in both conditions as compared to other classifiers especially Random Forest which also produced the same accuracy score as J48 in WEKA Explorer but can only produce 99.97% of accuracy when tested in WEKA Experimenter environment when missing values were present in the ASD dataset.

Future Work & Conclusion

Not many studies have utilized classification algorithms to particularly analyse ASD screening dataset. Most researchers that we found through literature review utilized other most common datasets such as breast cancer and heart disease datasets, thus there exists difficulties in analysing and comparing the performance of classifiers being tested to ASD screening dataset. This is important in order to assist in selecting the best classification methods for screening and diagnosing a patient with ASD, thus we attempted to fill in the gap by analysing and classifying ASD screening dataset by using six classifiers. Based on the result obtained through the tests with both Weka Explorer and Weka Experimenter, we can conclude that J48 outperformed other classifiers in terms of the time taken to classify the class as well as the accuracy and AUC produced by the classifier when tested to the ASD dataset with and without missing values. On the other hand, KNN performed poorly in classifying the dataset as compared to other classifiers even though the classifier took 0.00 second of runtime to classify the dataset.

The accuracy produced by J48 can assist health practitioners to make better decision when analysing datasets in both circumstances, with and without missing values. Making a right decision at the right time is imperative in crucial industries especially health-related field for it to perform effectively and efficiently as well as to reduce human errors which can affect the life of a patient. Moreover, we

compared our results with the previous works mentioned in this paper and found that a classifier's classifying performance is affected by the type of dataset as well as the number of instances involved in the experiment. However, a specific study is required to test the classification algorithms to classify other health related datasets with different volume of instances in order to investigate the effect of the volume on the performance of the algorithms. Future work can include utilizing other classification algorithms such as Simple Cart, Ada Boost, Logit Boost, Bagging, and Decision Stump. This study can also be further extended by deploying multi-modal deep learning to predict ASD occurrences as well as the abnormalities of other related neurological diseases such as amnestic mild cognitive impairment and multiple sclerosis (Jacini et. al, 2018; Liparoti et. al, 2019). Simulating missing values existed in other significant attribute fields should also be considered for future work.

Declarations

Ethics approval and consent to participate

Not applicable. This study reported public dataset.

Consent for publication

Not applicable.

Availability of data and materials

The datasets used in this study is a public dataset provided by *Dr. Fadi Fayez Thabtah from the Department of Digital Technology, Manukau Institute of Technology, New Zealand. The dataset is available publicly at this website:* https://archive.ics.uci.edu/ml/datasets/Autism+Screening+Adult.

Competing interests

All authors have no competing interests.

Authors' contributions

SFMR and MSH developed the concept for the study. SFMR and MAHMR retreived dataset and performed data analysis, classification, and evaluations. MSH provided supervision throughout the study. SFMR, MSH, and MAHMR contributed to the manuscript writing and approved the final manuscript.

References

[1] Bhaumik, S., Tyrer, F., Barrett, M., Tin, N., McGrother, C. W., & Kiani, R. (2010). The relationship between carers' report of autistic traits and clinical diagnoses of autism spectrum disorders in adults with intellectual disability. In Research in Developmental Disabilities, 31 (3), 705–712. doi:10.1016/j.ridd.2010.01.012

[2] Gold, C., Wigram, T., & Elefant, C. (2006). Music therapy for autistic spectrum disorder. In Cochrane Database of Systematic Reviews. DOI:10.1002/14651858.cd004381.pub2

[3] Center for Disease Control and Prevention (CDC). (2020). What is Autism Spectrum Disorder? Retrieved on 3 June 2020 from https://www.cdc.gov/ncbddd/autism/facts.html

[4] Bolton, P. F., Golding, J., Emond, A., & Steer, C. D. (2012). Autism Spectrum Disorder and Autistic Traits in the Avon Longitudinal Study of Parents and Children: Precursors and Early Signs. In Journal of the American Academy of Child & Adolescent Psychiatry, 51(3), 249–260.e25. doi:10.1016/j.jaac.2011.12.009

[5] Marlow, M., Servili, C. Tomlinson, M. (2019). A Review of Screening Tools for the Identification of Autism Spectrum Disorders and Developmental Delay in Infants and Young Children: Recommendations for Use in Low- and Middle-Income Countries. In Autism Research. DOI:10.1002/aur.2033

[6] Pinto-Martin, J. A., Souders, M. C., Giarelli, E., & Levy, S. E. (2005). The Role of Nurses in Screening for Autistic Spectrum Disorder in Pediatric Primary Care. In Journal of Pediatric Nursing, 20 (3), 163–169. doi:10.1016/j.pedn.2005.01.004

[7] Akay, M.F. (2009). Support vector machines combined with feature selection for breast cancer diagnosis. In Expert System Application, 36 (2). https://doi.org/10.1016/j.eswa.2008.01.009

[8] Thabtah, F. (2017). Autism Spectrum Disorder Screening: Machine Learning Adaptation and DSM-5 Fulfillment. In Proceedings of the 1st International Conference on Medical and Health Informatics 2017, pp.1-6. Taichung City, Taiwan, ACM. https://doi.org/10.1145/3107514.3107515

[9] Thabtah, F. (2019). An accessible and efficient autism screening method for behavioural data and predictive analyses. In Health Informatics Journal, 25(4), 1739–1755. Doi:10.1177/1460458218796636

[10] Ramotra, A.K., Mahajan, A., Kumar, R., Mansotra, V. (2020). Comparative Analysis of Data Mining Classification Techniques for Prediction of Heart Disease Using the Weka and SPSS Modeler Tools. In Smart Innovation, Systems and Technologies, 165, 89-97. DOI:10.1007/978-981-15-0077-0

[11] Kibis, E.Y., Büyüktahtakin, I.E., Dag, A. (2017). Data analytics approaches for breast cancer survivability: Comparison of data mining methods. In 67th Annual Conference and Expo of the Institute of Industrial Engineers, 591-596. https://search.proquest.com/docview/1951123056?accountid=14645

[12] Alaiad, A., Najadat, H., Mohsen, B., Balhaf, K. (2020). Classification and Association Rule Mining Technique for Predicting Chronic Kidney Disease. In Journal of Information and Knowledge Management. DOI: 10.1142/S0219649220400158

[13] Dawngliani, M.S., Chandrasekaran, N., Lalmawipuii, R., Thangkhanhau, H. (2020). Comparison of Decision Tree-Based Learning Algorithms Using Breast Cancer Data. In Lecture Notes on Data Engineering and Communications Technologies, 49, 885-896. DOI: 10.1007/978-3-030-43192-1_96

[14] Mun, N. L, and Jumadi, N. A. (2020). Statistical Evaluation on the Performance of Dyslexia Risk Screening System Based Fuzzy Logic and WEKA. In International Journal of Advanced Science and Technology, 29 (7), 638-649. http://sersc.org/journals/index.php/IJAST/article/view/9946/5413

[15] Raj, S., Masood, S. (2020) Analysis and Detection of Autism Spectrum Disorder Using Machine Learning Techniques. In Procedia Computer Science 167, 994–1004. DOI: 10.1016/j.procs.2020.03.399

[16] Lang, S., Bravo-Marquez, F., Beckham, C., Hall, M., Frank, E. (2019). WekaDeeplearning4j: A deep learning package for Weka based on Deeplearning4j. In Knowledge-Based Systems, 178, 48-50. https://doi.org/10.1016/j.knosys.2019.04.013

[17] Segal-Rozenhaimer, M., Li, A., Das, K., & Chirayath, V. (2020). Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN). In Remote Sensing of Environment, 237. doi:10.1016/j.rse.2019.111446

[18] Panahi, M., Sadhasivam, N., Pourghasemi, H. R., Rezaie, F., Lee, S. (2020). Spatial prediction of groundwater potential mapping based on convolutional neural network (CNN) and support vector regression (SVR). In Journal of Hydrology. https://doi.org/10.1016/j.jhydrol.2020.125033

[19] Brown, M. L., Kros, J. F. (2003). Data mining and the impact of missing data. In Industrial Management & Data Systems, 103. 611-621. DOI: 10.1108/02635570310497657

[20] Shahzad, W., Rehman, Q., Ahmed, E. (2017). Missing Data Imputation using Genetic Algorithm for Supervised Learning. In International Journal of Advanced Computer Science and Applications, 8 (3), 438-445. DOI: 10.14569/IJACSA.2017.080360

[21] Hruschka, E. R., Hruschka Jr., E. R., Ebecken, N. F. F. (2005). Missing Values Imputation for a Clustering Genetic Algorithm. In Lecture Notes in Computer Science, 3612, 245-254. https://link.springer.com/chapter/10.1007/11539902_29

[22] Pampaka, M., Hutcheson, G.,Williams, J. (2016). Handling missing data: analysis of a challenging data set using multiple imputation. In International Journal of Research & Method in Education, 39 (1), 19–37. http://dx.doi.org/10.1080/1743727X.2014.979146

[23] Al-Askar,H., Radi, M. MacDermott, A. (2016). Chapter 7 - Recurrent Neural Networks in Medical Data Analysis and Classifications. In Emerging Topics in Computer Science and Applied Computing, Applied Computing in Medicine and Health, 147-165. https://doi.org/10.1016/B978-0-12-803468-2.00007-2

[24] Tsai, C., Chang, F. (2016). Combining instance selection for better missing value imputation. In The Journal of Systems and Software, 122, 63-71. http://dx.doi.org/10.1016/j.jss.2016.08.093

[25] Chen, S., Webb, G. I., Liua, L., Ma, X. (2020). A novel selective naïve Bayes algorithm. In Knowledge-Based Systems, 192. https://doi.org/10.1016/j.knosys.2019.105361

[26] van der Heide, E.M.M., Veerkamp, R.F., van Pelt, M.L., Kamphuis, C., Athanasiadis, I., Ducro, B. J. (2019). Comparing regression, naive Bayes, and random forest methods in the prediction of individual survival to second lactation in Holstein cattle. In Journal of Dairy Science, 102 (10), 9409-9421. https://doi.org/10.3168/jds.2019-16295

[27] Lewis, D.D. (1998) Naive (Bayes) at forty: The independence assumption in information retrieval. In: Nédellec C., Rouveirol C. (eds) Machine Learning: ECML-98. ECML 1998. In Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), 1398. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0026666

[28] Ghazvini, K., Yousefi, M., Firoozeh, F., & Mansouri, S. (2019). Predictors of tuberculosis: Application of a logistic regression model. In Gene Reports. doi:10.1016/j.genrep.2019.100527

[29] Chen, L., Wang, C., Chen, J., Xiang, Z., Hu, X. (2020). Voice Disorder Identification by using Hilbert-Huang Transform (HHT) and K Nearest Neighbor (KNN). In Journal of Voice. https://doi.org/10.1016/j.jvoice.2020.03.009

[30] Madeti, S. R., Singh, S. N. (2018). Modeling of PV system based on experimental data for fault detection using kNN method. In Solar Energy, 173, 139-151. https://doi.org/10.1016/j.solener.2018.07.038

[31] Quinland, J. R. (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo.

[32] Yiu, T. (2019). Understanding Random Forest: How the Algorithm Works and Why it Is So Effective. Retrieved on 4 June 2020 from https://towardsdatascience.com/understanding-random-forest-58381e0602d2

[33] Breiman, L., Cutler, A. (2004). Interface Workshop-April 2004. RFtools for Predicting and Understanding Data. https://www.stat.berkeley.edu/~breiman/RandomForests/interface04.pdf

[34] Rad, M. T., Viardin, A., Schmitz, G. J., Apel, M. (2020). Theory-training deep neural networks for an alloy solidification benchmark problem. In Computational Materials Science, 180. https://doi.org/10.1016/j.commatsci.2020.109687

[35] Ganapathy, S., Kulothungan, K., Muthurajkumar, S. (2013). Intelligent feature selection and classification techniques for intrusion detection in networks: a survey. In Journal of Wireless Computer Network, 271. https://doi.org/10.1186/1687-1499-2013-271

[36] Delehanty, A., Lee, J., Hooker, J. L., Cortese, J., & Woods, J. (2020). Exploring message framing to engage parents in early screening for autism spectrum disorder. In Patient Education and Counseling. doi:10.1016/j.pec.2020.06.024

[37] Jones, E. J. H., Gliga, T., Bedford, R., Charman, T., & Johnson, M. H. (2014). Developmental pathways to autism: A review of prospective studies of infants at risk. In Neuroscience & Biobehavioral Reviews, 39, 1–33. doi:10.1016/j.neubiorev.2013.12.001

[38] Thabtah, F. F. (2017). Autistic Spectrum Disorder Screening Data for Children Data Set. [Data File]. Retrieved from https://archive.ics.uci.edu/ml/datasets/Autism+Screening+Adult
Thabtah, F. (2018). An accessible and efficient autism screening method for behavioural data and predictive analyses. In Health Informatics Journal. doi:10.1177/1460458218796636

[39] Bennett, D. A. (2001). How can I deal with missing data in my study? In Australian and New Zealand Journal of Public Health, 25 (5), 464-469. https://onlinelibrary.wiley.com/doi/epdf/10.1111/j.1467-842X.2001.tb00294.x

[40] Patil, T. (2016). Systematic Mapping Study of Missing Values Techniques using Naive Bayes. In International Research Journal of Engineering and Technology (IRJET), 3 (3). https://www.irjet.net/archives/V3/i3/IRJET-V3I3385.pdf

[41] Kanchana, S., Thanamani, A.S. (2016). Elevating the Accuracy of Missing Data Imputation Using Bolzano Classifier. In International Journal of Engineering and Technology (IJET), 8 (1). http://www.enggjournals.com/ijet/docs/IJET16-08-01-028.pdf

[42] Aziz, A. S. A., Hanafi, S. E., Hassanien, A. E. (2017). Comparison of classification techniques applied for network intrusion detection and classification. In Journal of Applied Logic, 24, Part A, 109-118. https://doi.org/10.1016/j.jal.2016.11.018

[43] Radzi, S.F.M., Abidin, S.S.Z., Aziz, A. A., Damandhoori F.S. H., Singh, M.M. (2017). Breast Cancer Cells Predictions with Classification Algorithms, In Advanced Science Letter, 23 (5), 3817-5015. https://doi.org/10.1166/asl.2017.8316

[44] Jacini,F., Sorrentino, P., Lardone, A., Rucco, R., Baselice, F., Cavaliere,C., Aiello, M., Orsini, M., Iavarone,A., Manzo, V., Carotenuto, A., Granata, C., Hillebrand, A., Sorrentino, G.. (2018). Amnestic mild cognitive impairment is associated with frequency-specific brain network alterations in temporal poles. In Frontiers in Aging Neuroscience 10, 400. https://dx.doi.org/10.3389%2Ffnagi.2018.00400

[45] Liparoti M, Della Corte M, Rucco R. (2019). Gait abnormalities in minimally disabled people with Multiple Sclerosis: A 3D-motion analysis study. In Multiple Sclerosis and Related Disorders, 29, 100-107. doi:10.1016/j.msard.2019.01.028

Figures
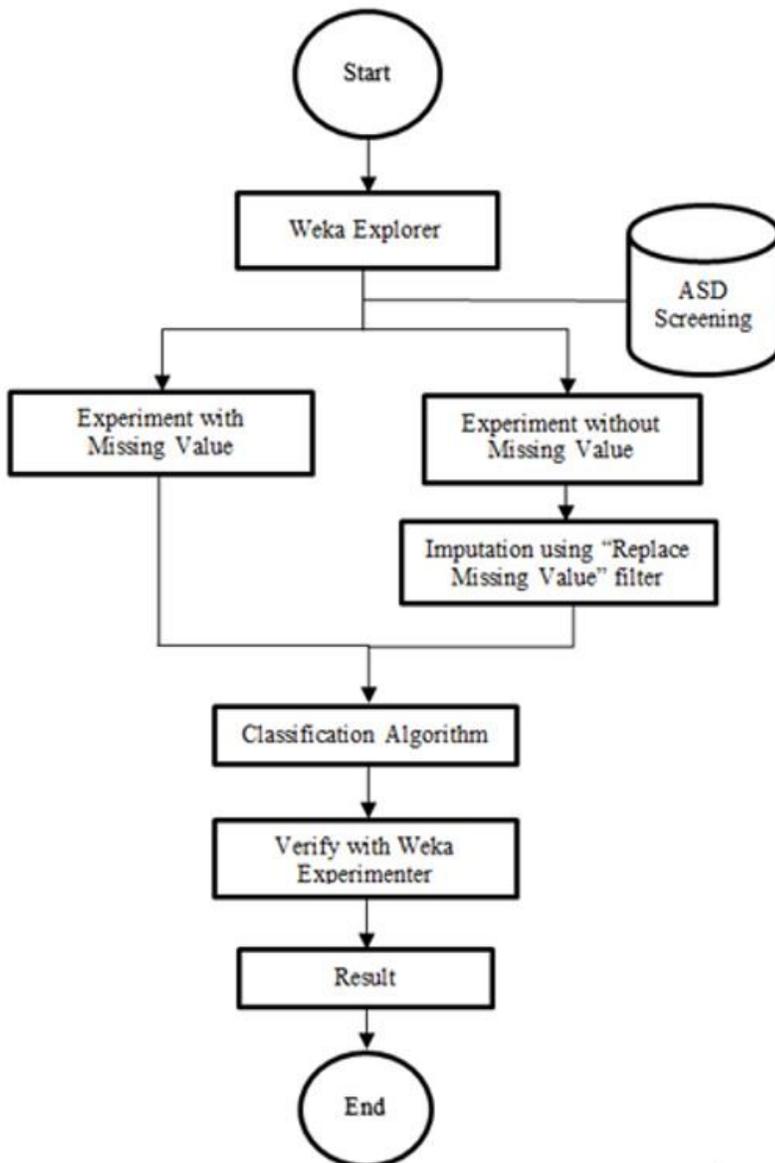
Figure 1

Classification process of ASD screening dataset
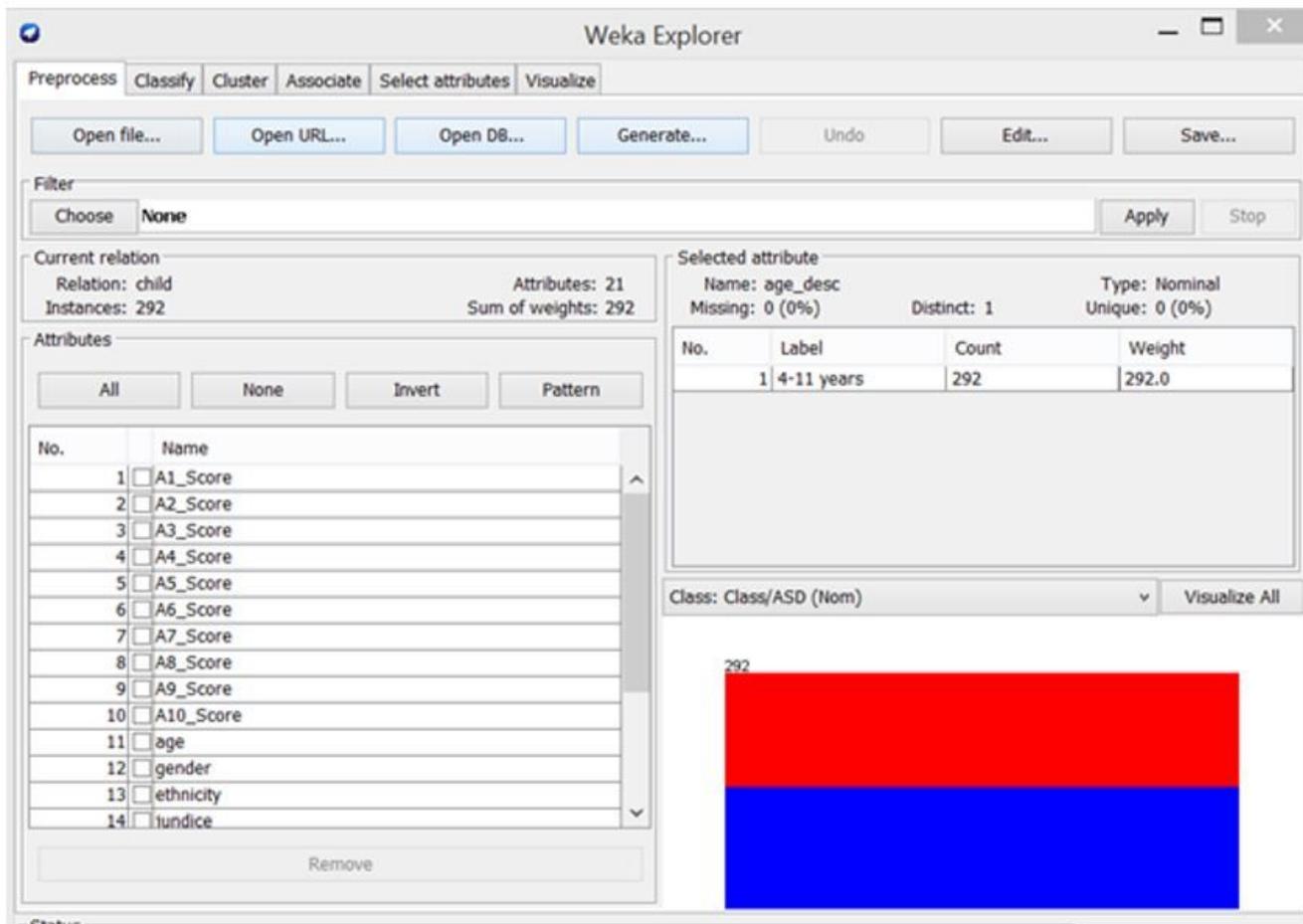
Figure 2

Description of ASD screening dataset in WEKA
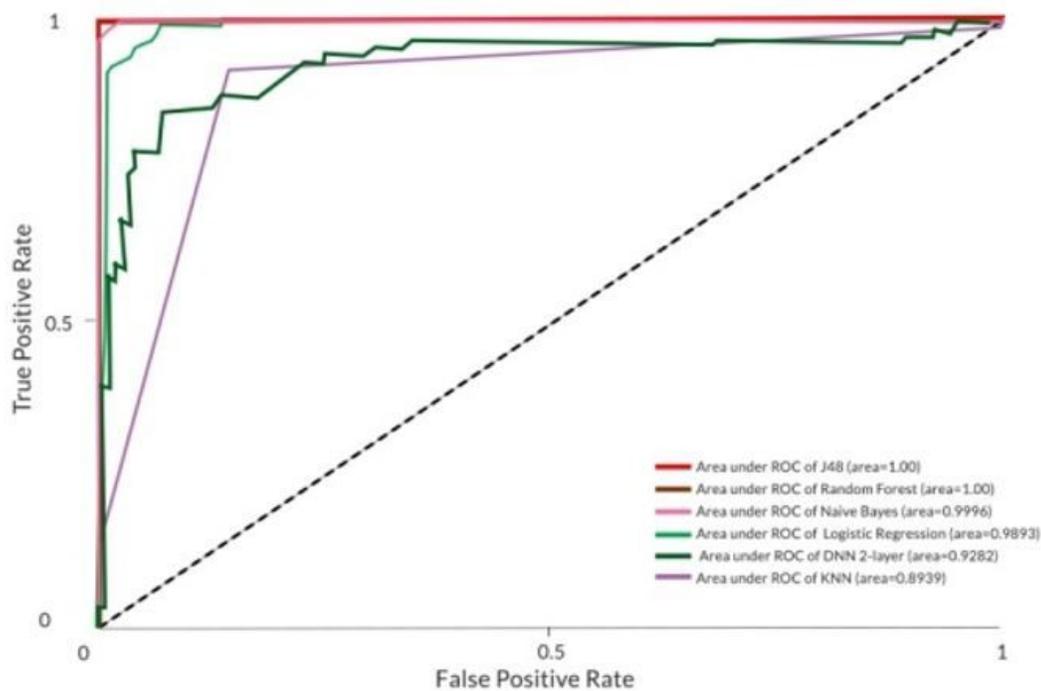
Figure 3

AUC of classifiers for ASD 'yes' class (with missing values)
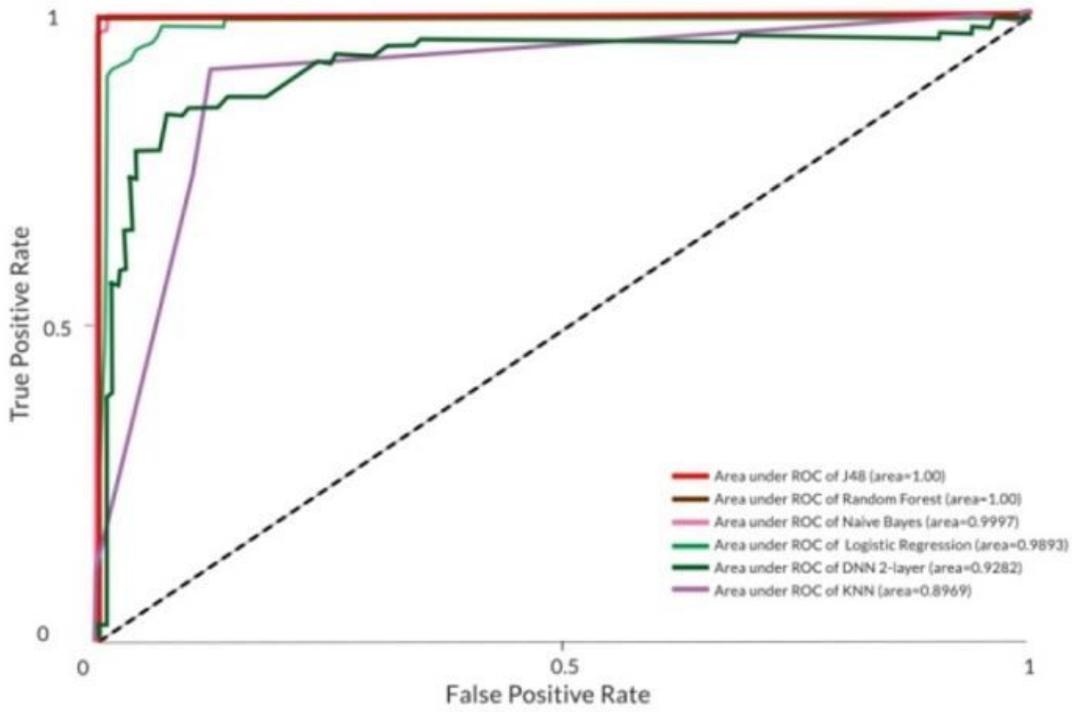

Figure 4

AUC of each classifier for ASD 'yes' class (without missing values)
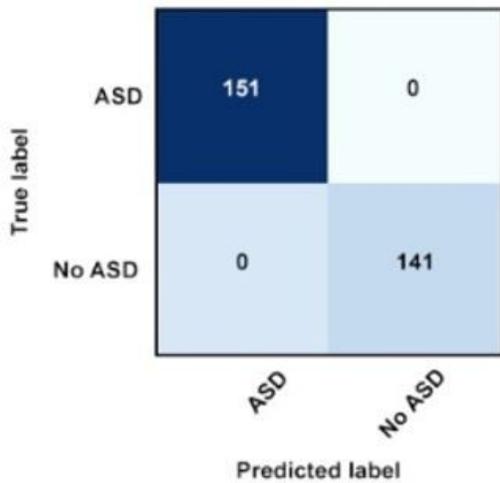

Figure 5

Confusion Matrix of J48 for ASD screening dataset with missing values
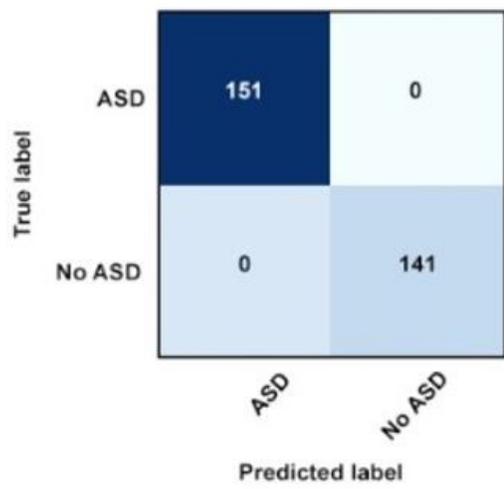
Figure 6

Confusion Matrix of J48 for ASD screening dataset without missing values