Turkish Journal of Computer and Mathematics Education Vol.13 No.01 (2022), 273-286
Research Article

Handling Outlier Data as Missing Values by Imputation Methods: Application of Machine Learning Algorithms

Muhammad Metwally Seliem

Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt.
Department of Business Information System, Higher Institute of Management Sciences, Ministry of Higher Education, New Cairo, Egypt.

Email address:

Seliem.M.M@hims.edu.eg

Abstract: This article is concerned with machine learning (ML) algorithms when data suffers from two common problems namely, missing values and outlies where both problems can be a main cause for efficiency degradation of the ML predictive models. Six ML algorithms are used to predict the Cleveland heart disease dataset where this data suffers from outliers. The performances of these algorithms are measured in terms of metrics namely; accuracy, kappa statistic, F1 score and our suggestion in using the geometric mean. To overcome the negative impacts of outliers and missing values, we proposed a technique called treatment of outlier data as missing values by applying imputation methods (TOMI) instead of the classical method by removing these outliers. Four methods were applied to impute missing data namely, mean, median, K-Nearest Neighbor (KNN) and Random Forest (RF), where the KNN method outperformed the other different methods in terms of mean absolute error (MAE) and root mean square error (RMSE) in imputing the missing values. There are two scenarios on splitting our dataset namely; first: 60-40% and second:75-25%. Based on the first scenario, the Naive Bayes (NB) algorithm showed the highest performance (in all metrics); for instance, in accuracy it achieved 85.12% and 87.6% before and after applying TOMI respectively. While in the second scenario, the Logistic Regression (LR) algorithm showed the highest performance (in all metrics); again for accuracy it achieved 86.67% and 89.33% before and after applying TOMI respectively. To conclude, the NB and LR algorithms predict our dataset better than other used algorithms. Moreover, the applied TOMI technique enhances the efficiency of the entire ML algorithms, which makes the predictive models possess more accurate results.

Key words: Outlier data, Missing values, Machine Learning, Heart Disease, UCI ML Repository, F1 score, Kappa Statistic, Accuracy, Confusion Matrix.

---

1. Introduction

ML or data mining is useful for solving a diverse set of problems. ML involves artificial intelligence, and it is used in solving many problems in data science. ML is a vast interdisciplinary field involves concepts from computer science, statistics, cognitive science, engineering and many other disciplines of mathematics and science. One common application of ML is the prediction of an outcome (i.e., dependent variable) based upon existing data (i.e., independent variables). The machine learns patterns from the existing dataset, and then applies them to an unknown dataset in order to predict the outcome. ML has become one of the mainstays of information technology where its tools are concerned with endowing programs with the ability to learn and adapt. There are numerous applications for ML but data mining is the most significant among all. Classification is one of the most studied problems where it is a powerful ML technique that is commonly used for prediction (Latha and Jeeva, 2019). Three types of ML models researchers need to be familiar with and know the requirements of each: supervised, unsupervised and reinforcement ML.

Supervised learning trains a model on known input and output data so that it can predict future outputs. It is based on "train me" concept (Singh and Kumar, 2020). Its most popular algorithms in the continuous case are: Regression, Decision Tree (DT) and Random Forest while in categorical (classification) case are: Naive Bayes, Support Vector Machine (SVM), K-Nearest Neighbor and Logistic Regression.

Unsupervised learning is used to draw conclusions from datasets which consist of input data with no labelled responses. It can be defined as the learning without a guidance and based on "self-sufficient " concept. In other words, unsupervised learning, finds hidden patterns or intrinsic structures in input data (Singh and Samagh, 2020). Its most common clustering algorithms are: Principal Component Analysis (PCA) and K-Means.

Reinforcement learning: where the algorithm learns a policy of how to act given an observation of the world. Every action has some impact in the environment, and the environment provides feedback that guides the learning algorithm. It is based on "hit and trial" concept. this is frequently used for robotics, gaming and routing (Ayodele, 2010). We can separate our strategy into four primary segments as follows: data Collection, data pre-processing, data training and applications of ML algorithms. An overall work flow of our study has been shown in Fig. 1.

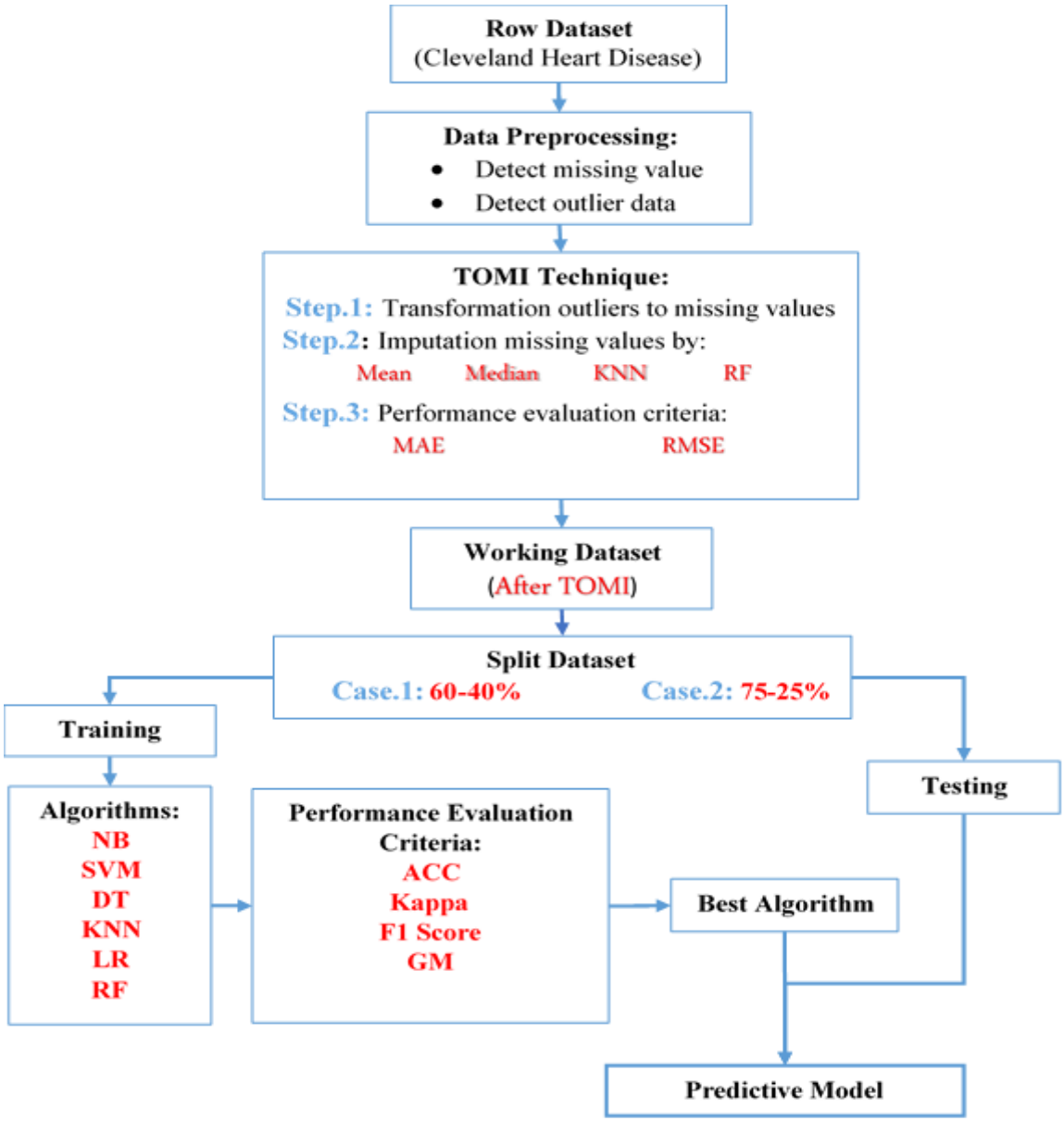

Fig. 1: Methodology of the research work.

ML has been the corner stone in analyzing and extracting information from data and often a problem of missing values is encountered. The missing values is a popular and important topic in statistics. Missing values occur as a result of various factors like missing completely at random, missing at random or missing not at random. Missing data can reduce the statistical power of a study and can produce biased estimates, leading to invalid conclusions. Often, the dataset that used in our analysis is incomplete (includes missing values) in independent and/or dependent variables. Data imputation is defined as a technique of replacing missing data with substituted values. Selection of imputation method usually determined by the mechanism of how the values are missing (Abidin et al., 2018).

An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism. Although outliers are often considered as an error or noise, they may carry important information. Detected outliers are candidates for aberrant data that may otherwise adversely lead to model misspecification, biased parameter estimation and incorrect results. So these data should be excluded from regular

processing of data mining. If these outliers are identified and removed, the predictions made with data-driven ML methods can be more accurate with better predictions. Instead of the classic method by removing these outliers we proposed a new technique called TOMI.

This technique works in two steps, the first is to convert the outliers to missing values, and then imputing them. Then applying to this treated data via TOMI certain supervised ML algorithms (i.e., NB, SVM, DT, KNN, LR and RF). The performance of these proposed algorithms was evaluated using the accuracy, Kappa statistic and geometric mean. The rest of the paper is organized as follows. In the next section, introduces the dataset description and pre-processing. While in Section 3, discusses the different ML algorithms. In Section 4, we illustrate the performance metrics. A real data set is analyzed in Section 5 and concluding remarks are included in Section 6

2. Data Sources

The data used in this study is the Cleveland clinic foundation heart disease dataset available at UCI Machine Learning Repository (Chaki et al., 2015). This dataset has 76 raw attributes, but all published experiments refer to using a subset of 14 of them and it is widely used in the literature as follows in table 1.

Table.1: Selected Cleveland heart disease dataset attributes.

No	Attribute	Value	Type
1	Age	Years: 29 – 77	**Continuous**
2	Sex	1 = male and 0 = female.	**Discrete**
3	CP	Chest pain type: 1 = typical angina, 2 = atypical angina, 3 = non - anginal pain and 4 = asymptomatic.	**Discrete**
4	Trestbps	Resting blood pressure (in mm Hg): 94 – 200	**Continuous**
5	Chol	Serum cholesterol in mg/dl: 126 – 564	**Continuous**
6	Fbs	Fasting blood sugar > 120 mg/dl:1 = true and 0 = false.	**Discrete**
7	Restecg	Resting electrocardiographic: 0 = normal, 1 = having ST-T abnormality and 2 = showing probable or define left ventricular hypertrophy by Estes'criteria.	**Discrete**
8	Thalach	Maximum heart rate achieved : 71 – 202	**Continuous**
9	Exang	Exercise induced angina:1 = Yes and 0 = No.	**Discrete**
10	Oldpeak	Depression induced by exercise relative to rest: 0 – 6.2	**Continuous**
11	Slope	The slope of the peak exercise segment: 1 = up sloping, 2 = flat and 3 = down sloping.	**Discrete**
12	Ca	Number of major vessels colored by fluoroscopy that ranged between 0 to 3.	**Discrete**
13	Thal	3 = normal, 6 = fixed defect and 7 = reversible defect.	**Discrete**
14	Diagnosis	Diagnosis classes: 0 = healthy and 1 = patient who is subject to possible heart disease.	**Discrete**

The data set contains 303 data samples and consists of 13 numeric attributes including age, sex, chest pain type, resting blood pressure, cholesterol, fasting blood sugar, resting ECG, maximum heart rate, exercise induced angina, oldpeak, slope, number of vessels coloured and thal. The classes comprise of integers valued (0): no presence (No Pre.) of heart disease and valued (1): presence (Pre.) of heart disease (Huapaya et al., 2020).

Data Pre-processing (DP): The importance of DP known also as data preparation, is due to several aspects: firstly, the data must be organized into a proper form for data mining algorithms, and, secondly, the data sets used must lead to the best performance and quality for the models obtained by data mining operations. Data pre-processing contains operations which perform data cleaning, data integration, data transformation and data reduction where DP is concerned with transforming the raw data that was collected into a form that can be used in modeling (Danubianu, 2015). Also, the data pre-processing removes the missing and outlier data values from the dataset.

Data Normalization: In most cases, the datasets output its numerical variables have different units and scales, for example, 'Age' in years, 'Income' in dollars and etc. These differences can unduly influence the model. Non-normalization of the data affects the ML algorithm which will be dominated by the variables that use a larger scale, adversely affecting the model performance. Therefore, we need to transform them

3. Algorithms

The supervised ML algorithms which deal more with classification, it includes the following NB, SVM, DT, KNN, LR and RF.

3.1 Naive Bayes

NB is a supervised learning classification algorithm; the NB classifier is based on the bayes theorem. It is a special case of the Bayesian network, and it is a probability based classifier, where the NB classify the data by computing the probability of independent variables. After calculating the probability of each class, the high probability class do assign for the complete transaction. In the NB network, all features are conditionally independent. The changes in one feature therefore does not affect another feature. The NB algorithm is suitable for classifying high dimensional datasets. NB algorithm also useful for classifying different kind of dataset like sentiment analysis and virus detection. It works by using the values for independent variables and predict a pre-defined class for each record. It measures the probability of A given B as shown equation (1). Then working on finding out the distinct class for each attribute (Alabi et al., 2020). NB uses the following equation for measuring the probability:

$$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right)P(A)}{P(B)} \tag{1}$$

3.2 Support Vector Machin

SVM is considered one of the highest prominent and convenient technique for solving problems related to classification of data, learning and prediction. Based on the kernel functions the SVM classifiers are divided into different types such as linear, nonlinear, radial basis function, sigmoid and polynomial. The hyperplane or SVM separates the support vector or data points. The main advantage of SVM is its capability to deal with wide variety of classification problems includes high dimensional and not linearly separable problems. One of the major drawbacks of SVM that it requires a number of key parameters to be set correctly for attaining excellent classification results (Reddy et al., 2019). The general workflow is: Step.1: First, it finds lines or boundaries that correctly classify the training dataset. Step.2: Then, from those lines or boundaries, it picks the one that has the maximum distance from the closest data points (Rawal, 2020).

3.3 Decision Tree

DT is considered the graphical representation of the data. Its supervised learning algorithm is of nonparametric method and It is a powerful prediction model used in classification and regression problems. DT builds classification or regression models in the structure of a tree making it simple to debug and easy for handling both categorical and numerical data. This algorithm divides the population into two or more similar sets based on the most significant predictors. DT algorithm, first calculates the entropy of each and every attribute. Then the dataset is split with the help of the variables or predictors with maximum information gain or minimum entropy. These two steps are performed recursively with the remaining attributes (Ramalingam et al., 2018).

$$Entropy(S) = \sum_{i=1}^{c} - p_i Log_2\, p_i \tag{2}$$

The algorithm works by finding the information gain of the attributes and taking out the attributes for splitting the branches in trees. The information gain for the tree is identified using the given below equation (3):

$$Gain(S, A) = Entropy(S) - \sum_{\vartheta \in values\,(A)} \frac{|S_\vartheta|}{|S|}\, Entropy\,(S_\vartheta) \tag{3}$$

The algorithm for the DT (David and Belcy, 2018) is: Step 1: Identify the information gain for the attributes in the dataset. Step 2: Sort the information gain for the Iris datasets in descending order. Step 3: After the identification of the information gain assign the best attribute of the dataset at the root of the tree. Step 4: Then calculate the information gain using the same formula. Step 5: Split the nodes based on the highest information gain value. Step 6: Repeat the process until each attributes are set as leaf nodes in all the branches of the tree.

3.4 K-Nearest Neighbour (KNN)

KNN classifier is simple but powerful classification algorithm and it is one of the most popular techniques for pattern recognition. Also it is non-parametric algorithm which means that it does not assume anything on the underlying data distribution. In this, the Euclidean distance is calculated between the test data and every sample in

the training data followed by classifying the test data into a class in which most of k-closest neighbors of training data belong to "K "is usually a very small positive integer. As the Value of K increases it becomes increasingly difficult to distinguish between the various classes. Cross-validation technique is used to choose an optimal value of K (Theerthagiri et al., 2021). The idea of the KNN classifier is to take a test data point and compare it with all training data points and to predict the label (class) of the test data point based on the closest training class using the d_1 distance given by:

$$d_1(I_1, I_2) = (\sum_P |I_1^P - I_2^P| \quad (4)$$

where $I_1$and I_2 are the vectors representation of points 1 and 2 respectively, and d_1 denote the distance and Σ is taken over all points (Almustafa, 2020). In general, if the number of neighbors is denoted by N in KNN, then N samples are considered using the following distance metric value

$$d(x, y) = (\sum_{i=1}^{n} |x_i - y_i|^p)^{1/p} \quad (5)$$

where if $p = 1$, then it is Manhattan distance, if $p = 2$, then it is Euclidean distance, and if $p = \infty$ then it is Chebyshev distance. The algorithm for the K-NN as follows (Rawal, 2020): Step.1: Input the dataset and split it into a training and testing set. Step.2: Pick an instance from the testing sets and calculate its distance with the training set. Step.3: List distances in ascending order. Step.4: The class of the instance is the most common class of the three first trainings instances (k=3).

3.5 Logistic Regression

One of the simplest and powerful prediction algorithm is LR; a well-established method for supervised classification. LR measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function, which is the cumulative logistic distribution. LR can be seen as a special case of the generalized linear model (GLM) and thus similar to linear regression. In particular, the key differences between these two models can be seen in the following two features of LR. First, the conditional distribution y | x is a Bernoulli distribution rather than a Gaussian distribution, because the dependent variable is binary. Second, the predicted values are probabilities and are therefore restricted to (0,1) through the logistic distribution function because LR predicts the probability of particular outcomes (Abonazel and Ibrahim, 2018). The equation of LR is as follows:

$$Logit\ (\pi) = \ln(\frac{\pi}{1-\pi}) \quad (6)$$

where (π) is probability of presence of characteristics and $1 - \pi$ is probability of absence of characteristics. The LR assumptions as follow: Step.1: LR does not assume a linear relationship between the dependent and independent variables. Step.2: The dependent variable must be a dichotomy (2categories). Step.3: The independent variables need not to be interval, normally distributed, linearly related, nor with equal variance within each group. Step.4: The categories (groups) must be mutually exclusive and exhaustive; a case can only be in one group and every case must be a member of one of the groups.

3.6 Random Forest

RF is also a popularly supervised ML algorithm. This technique can be used for both regression and classification tasks but generally performs better in classification tasks. Also it works well with large datasets with high dimensionality. The main principle is to assemble several weak learners to create strong learner (Abd El-Salam et al.,2018). RF is an ensemble learning method building models that construct several decision trees at training time, and outputs the modal class out of the classes predicted by individual trees. RF is a combination of tree predictors where each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forest (Kamble et al., 2014).

The algorithm of RF as follows (Islam et al., 2020): Step.1: From the training set, picked K data points randomly. Step.2: From these K data points, generate the decision trees. Step.3: From generated trees, choose the number of N-tree and repeat steps (1) and (2). Step.4: Form the N-tree that predicts the category to which the data points relate for a new data point, and assign the new data point via the category with the highest probability.

4. Performance Evaluation Criteria

Typically, the performance of the ML prediction algorithms is measured by using some metrics based on the classification algorithm. In this work, the prediction results are evaluated by using the metrics such as confusion matrix, accuracy, Kappa statistic. Also, we suggested the geometric mean which uses the followings in its calculation; accuracy, Kappa statistic and F1 score.

4.1 Confusion Matrix

Confusion matrix is a visualization tool commonly used to present the accuracy, sensitivity and specificity of algorithms where it provides a complete insight into the performance of a prediction model. Table (2) summarizes the entries of the confusion matrix used. The matrix gives information about correctly classified as Normal (true) and misclassified as Abnormal (false).

Table (2): Confusion Matrix

Actual	**Predicted**	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN

Where, true positive (TP) is number of positive samples correctly predicted. False negative (FN) is number of positive samples wrongly predicted. False positive (FP) is number of negative samples wrongly predicted as positive. True negative (TN) is number of negative samples correctly predicted.

4.2 Accuracy

The accuracy is a measure of closeness between the target data and the predicted data. It is the ratio of the total number of correct predictions of class to the actual class of the dataset and the equation of accuracy as follows:

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

4.3 Kappa statistic

Estimates the consistency of the prediction model. It compares the result of the predicted model with actual results. It is a statistic value between 0 and 1. A value near 1 might have the great consistency where the higher Kappa value implies a better classification and the equation of Kappa value as follows:

$$Kappa = \frac{\left[\frac{TP+TN}{N}\right] - \left[\frac{(TP+FN)\times(TP+FP)\times(TN+FN)}{N^2}\right]}{1 - \left[\frac{(TP+FN)\times(TP+FP)\times(TN+FN)}{N^2}\right]}$$

4.4 Precision

Precision is the part of significant instances between the retrieved instances. The equation of precision as follows:

Precision = TP/(TP+FP)

4.5 Recall

Recall is the small part of appropriate instances that have been retrieved over the total quantity of relevant instances. The equation of recall as follows:

Recall = TP/ (TP + FN)

4.6 F1 Score:

F1 Score is the measure of the balanced score (harmonic mean) of both precision and recall. It is considered based on the two times the precision times recall divided by the sum of precision and recall and The equation of F1 Score as follows:

F1 Score = 2 × (Precision × Recall) / (Precision + Recall).

4.7 Geometric Mean

$$\text{GM} = \sqrt[3]{\text{Accuracy} \times \text{Kappa statistic} \times \text{F1 score}}$$

5. Real Data Application

5.1 Missing Value

The heart disease dataset included in this research work has no missing values as shown Fig. 2.

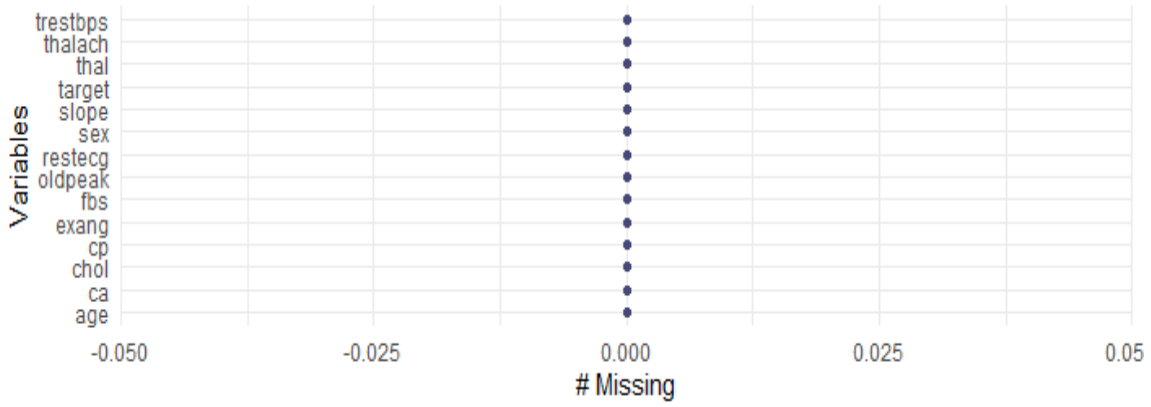

Fig.2: Missing value.

5.2 Outliers data

The heart disease dataset included in this research work has outliers as shown Fig. 3. The box plot of Fig. 3 shows that there are some outliers outside the whiskers of the box plot. However, the Rosner's test Indicates that these outliers are not all significant and thus have no influence on the observations as shown table 3. For examples, the boxplot of "thalach "variable shows that there is one outlier value, but this outlier is not significant as shown table 3 which means, this value does not have a bad effect on the data. Though, the boxplot of "oldpeak " variable shows that there are two outliers value, but these outliers are significant as shown table 3 which means, this value does have a bad effect on the data, therefore, it must be handled. In any case, if the value has a bad effect on the data, we will be handling it, and if it has no a bad effect on the data, we will not be handling it.

Table.3: Results of Rosner's Test for Outliers before handling

Variables	Mean. I	SD.i	Value	Obs.Num	R.i+1	lambda.i+1	Outlier
Thalach	149.6469	22.90516	71	273	3.433587	3.726364	FALSE
Oldpeak	1.039604	1.161075	6.2	205	4.444498	3.726364	TRUE
	1.022517	1.124193	5.6	222	4.071794	3.725431	TRUE
	1.007309	1.094507	4.4	292	3.099742	3.724494	FALSE
	0.996	1.078577	4.2	102	2.970581	3.723555	FALSE
	0.985284	1.064269	4.2	251	3.020586	3.722611	FALSE
Thal	2.313531	0.612277	0	49	3.778573	3.726364	TRUE
	2.321192	0.598571	0	282	3.877891	3.725431	TRUE
Trestbps	131.6238	17.53814	200	224	3.898716	3.726364	TRUE
	131.3974	17.11795	192	249	3.540298	3.725431	FALSE
	131.196	16.78446	180	111	2.907689	3.724494	FALSE
	131.0333	16.5731	180	204	2.954586	3.723555	FALSE
	130.8696	16.35593	180	267	3.00383	3.722611	FALSE
	130.7047	16.13266	178	102	2.93165	3.721664	FALSE
	130.5455	15.92355	178	261	2.980148	3.720713	FALSE
	130.3851	15.70858	174	242	2.776499	3.719759	FALSE
	130.2373	15.52761	172	9	2.689577	3.718801	FALSE

Chol	246.264	51.83075	564	86	6.13026	3.726364	TRUE
	245.2119	48.56788	417	29	3.537072	3.725431	FALSE
	244.6412	47.62357	409	247	3.451207	3.724494	FALSE
	244.0933	46.74335	407	221	3.48513	3.723555	FALSE
	243.5485	45.8576	394	97	3.280841	3.722611	FALSE

It can be observed from table 3 that there are some outliers that have bad influence on the other observations with ultimate incorrect results, where TOMI technique handles these outliers as shown Fig. 5.

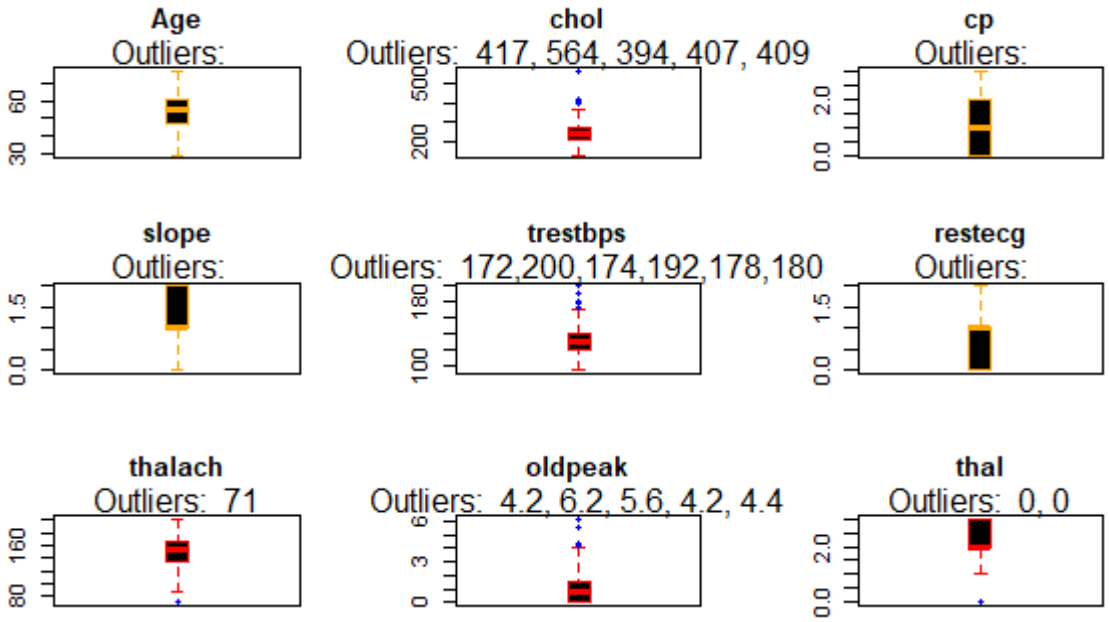

Fig. 3: Boxplot of the raw data before applying TOMI.

5.3 Missing Data Imputation

Missing data imputation techniques based on both statistical and ML algorithms were applied to impute absent values in dataset. The performance evaluation of different missing values approaches in ML problems can be done using different criteria, on this section we discuss the most commonly used which are, MAE and RMSE.

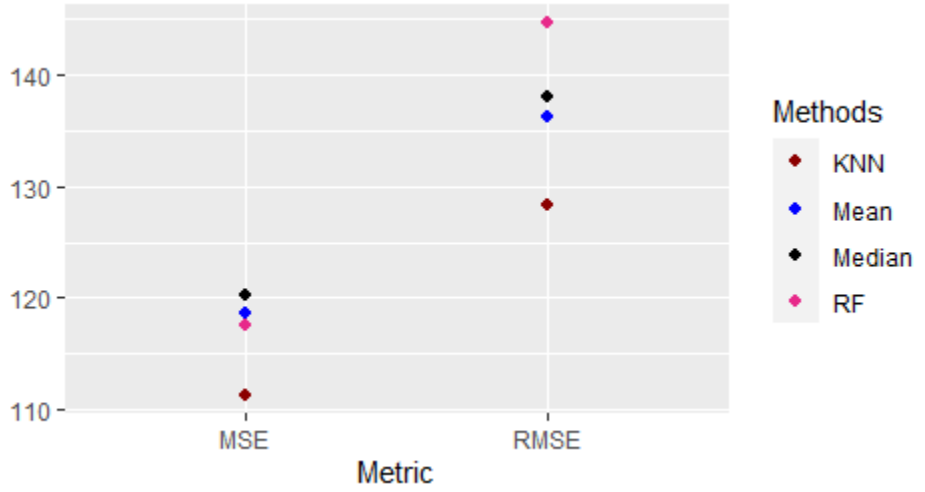

Fig. 4: Imputation Methods.

The imputation methods applying in this paper based on statistical techniques, e.g., mean and median, and ML method, e.g., KNN and RF.

Table.4: Performance Evaluation of Imputation Methods.

Metrics	**Algorithms**			
	Mean	Median	**KNN**	RF
MAE	118.6321	120.1808	**111.3243**	117.5385
RMSE	136.2314	138.0773	**128.4129**	144.6684

As seen in Fig. 4, the KNN method outperformed the different methods in terms of MAE and RMSE. Fig.4 and table 4 also confirms this conclusion. Fig. 5 show that no outlier values are present in the data.

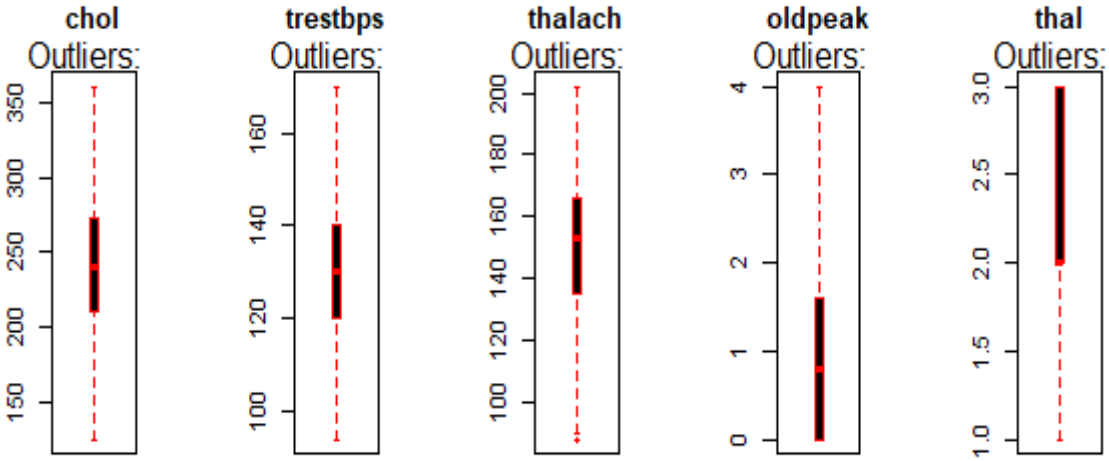

Fig. 5: Boxplot of data after applying TOMI.

5.4 Results

After imputation and data normalization to build a classification model, the combined dataset with 14 attributes is divided into training and testing data with a percentage split of 60–40%, 75–25%. In case 1, data is split below into two subsets: training (60%) and testing (40%) while in case 2, data is split below into two subsets: training (75%) and testing (25%). The confusion matrix obtained by five different supervised ML algorithms is given below. The performance measures are in accordance with the accuracy of each classification algorithm. Ten-fold cross validation was utilized to evaluate the performance of the classification models. In this approach, the entire dataset is divided into ten subsets and processed ten times where, nine subsets are used as testing sets and the remaining subset is used as training. Finally, the results are obtained by averaging each ten iterations.

Table (5): Case 1 splitting data to 60–40%.

Phase Testing	**Algorithms**	**Performances**						
		Confusion matrix			Metric			
		TYPE	**No Pre.**	**Pre.**	**ACC**	**Kapp**	**F1**	**GM**
Before TOMI	**NB**	No Pre.	**44**	**7**	**0.8512**	**0.6982**	**0.8302**	**0.7901**
		Pre.	**11**	**59**				
	SVM	No Pre.	40	5	8347	0.6615	0.8000	0.7615
		Pre.	15	61				

	DT	No Pre.	26	1	0.7521	0.4778	0.6341	0.6107
		Pre.	29	65				
	KNN	No Pre.	42	6	0.843	0.6799	0.8155	0.7760
		Pre.	13	60				
	LR	No Pre.	46	12	0.8264	0.6516	0.8142	0.7596
		Pre.	9	54				
	RF	No Pre.	43	6	0.8512	0.6972	0.8269	0.7887
		Pre.	12	60				
After TOMI	NB	No Pre.	**44**	**4**	**0.876**	**0.7473**	**0.8544**	**0.8239**
		No Pre.	**11**	**62**				
	SVM	Pre.	39	2	0.8512	0.6935	0.8125	0.7827
		No Pre.	16	64				
	DT	Pre.	38	4	0.8264	0.643	0.7835	0.7467
		No Pre.	17	62				
	KNN	Pre.	41	5	0.843	0.679	0.8119	0.7745
		No Pre.	14	61				
	LR	No Pre.	46	10	0.843	0.6838	0.8288	0.7817
		Pre.	9	56				
	RF	No Pre.	42	3	0.8678	0.7292	0.8400	0.8100
		Pre.	13	63				

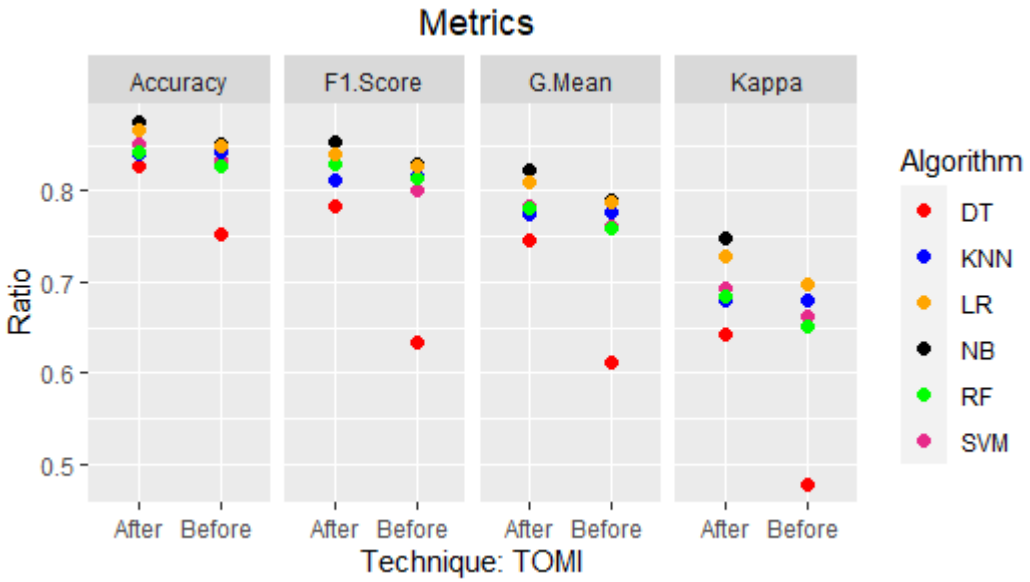

Fig. 6: Performance Evaluation of before and after TOMI technique in case 1.

Table 5 compares the classification metrics of algorithms before and after applying TOMI technique where it showed the effectiveness of the new technique. For instance, the accuracy of NB, SVM, DT, KNN, LR and RF are found in the range of 75.21%–85.12%. But after applying TOMI, the accuracy of algorithms is found in the range of 82.64%–87.6%.

Fig. 6 the NB algorithm exhibits the best performance in all metrics after applying TOMI technique where the accuracy is (87.7%), kappa (74.73%), F1 score (85.44%) and geometric mean (82.39%). Table 6 with before applying TOMI technique in case 75% training and 25% testing, the dataset is classified, the accuracy rates of NB, SVM, DT, KNN, LR and RF are found in the range of 76%–86.67%. But after applying TOMI, the accuracy rates of algorithms are found in the range of 77.33%–89.33% as shown table 6.

Table (6): Case 2 splitting data to 75–25%.

Phase Testing	**Algorithms**	**Performances**						
		Confusion matrix			Metric			
		TYPE	**No Pre.**	**Pre.**	**ACC**	**Kapp**	**F1**	**GM**
Before TOMI	NB	No Pre.	28	8	0.8133	0.6253	0.8000	0.7409
		Pre.	6	33				
	SVM	No Pre.	27	4	0.8533	0.7018	0.8308	0.7923
		Pre.	7	37				
	DT	No Pre.	25	9	0.76	0.5158	0.7353	0.6605
		Pre.	9	9				
	KNN	No Pre.	27	6	0.8267	0.6494	0.8060	0.7563
		Pre.	7	35				
	LR	**No Pre.**	**30**	**6**	**0.8667**	**0.7323**	**0.8571**	**0.8163**
		Pre.	**4**	**35**				
	RF	No Pre.	27	5	0.84	0.6756	0.8182	0.7743
		Pre.	7	36				
After TOMI	NB	No Pre.	28	4	0.8667	0.7296	0.8485	0.8125
		No Pre.	6	37				
	SVM	Pre.	26	1	0.88	0.7536	0.8525	0.8268
		No Pre.	8	40				
	DT	Pre.	26	9	0.7733	0.5438	0.7536	0.6817
		No Pre.	8	32				
	KNN	Pre.	25	4	0.8267	0.6458	0.7937	0.7511
		No Pre.	9	37				
	LR	**No Pre.**	**30**	**4**	**0.8933**	**0.7848**	**0.8824**	**0.8520**
		Pre.	**4**	**37**				
	RF	No Pre.	27	5	0.84	0.6756	0.8182	0.7743
		Pre.	7	36				

Fig. 7 shows a visual representation of the accuracy and Kappa statistic results of the used algorithms before and after applying TOMI technique presented in details in table 6, which concludes that the LR algorithm

outperformed the other algorithms where it achieved (86.67% and 89.33%) for accuracy, (73% and 78.5%) for kappa statistic, (85.7% and 88.24) for F1 statistics and (81.6% and 85.2%) for geometric mean, before and after applying TOMI technique respectively.

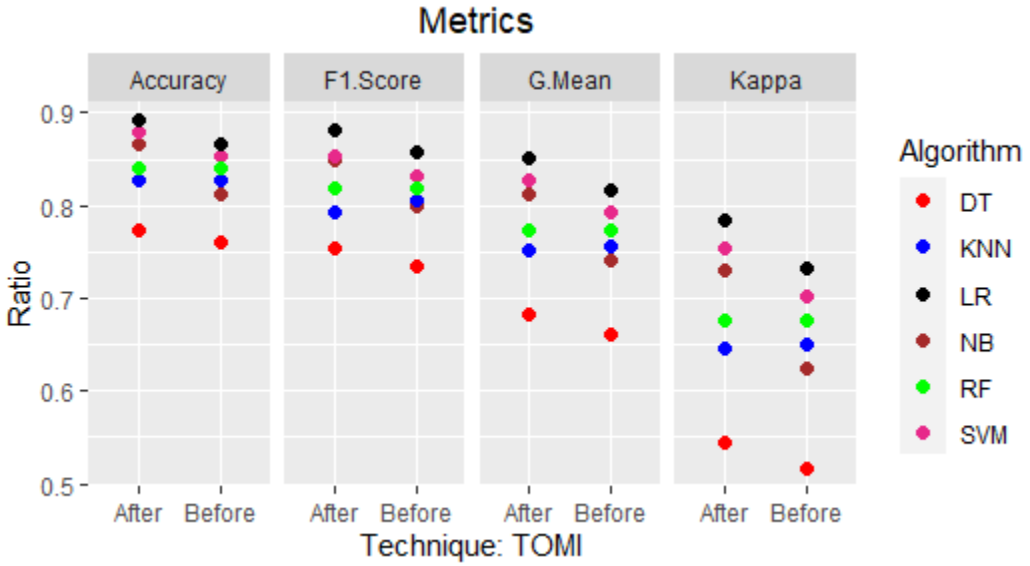

Fig. 7: Performance Evaluation of before and after TOMI technique in case 2.

6. Conclusion

In ML algorithms, the presence of missing values and outlier data can often lead to model misspecification and degrades the efficiency of ML predictive models. Although outliers are often considered as an error or noise, they may carry important information. In this article, supervised ML algorithms namely NB, SVM, DT, KNN, LR and RF are used to predict the Cleveland heart disease dataset where this data suffers from outliers. To overcome the negative impacts of outlier data and missing values we suggest to use TOMI technique. We applied four imputation methods to treat the problem of missing data, which comprised two methods based on statistical analysis and two methods based on ML. The statistical methods included mean and median imputation and two different ML implementation included KNN and RF. The performances of these methods are measured in terms of MAE and RMSE. According to the results of the imputation methods, we apply ML algorithms where the performances of the ML algorithms are evaluated by the following metrics; accuracy, kappa statistic, F1 score and our suggestion in using the geometric mean, which is calculated based on the previous three metrics. In this regard, there are two scenarios on splitting our working dataset. The first scenario is 60-40% and the second scenario is 75-25%.

The Results showed that the KNN method outperformed the other different methods in imputing the missing values according to MAE and RMSE. Based on the first scenario, the NB algorithm showed the highest performance; it achieved (85.12% and 87.6%) for accuracy, (69.8% and 74.73%) for kappa statistic, (83.02% and 85.44%) for F1 score and (79% and 82.4%) for geometric mean, before and after applying TOMI technique respectively. While in the second scenario, the LR algorithm showed the highest performance; it achieved (86.67% and 89.33%) for accuracy, (73% and 78.5%) for kappa statistic, (85.7% and 88.24%) for F1 statistics and (81.6% and 85.2%) for geometric mean, before and after applying TOMI technique respectively. To conclude, the NB and LR algorithms predict our dataset better than other used algorithms. Furthermore, the applied TOMI technique

enhances the efficiency of the whole ML algorithms, which makes the predictive models possess more accurate results.

7. References

[1] Abd El-Salam, S. M., Ezz, M. M., Hashem, S., Elakel, W., Salama, R., ElMakhzangy, H., and ElHefnawi, M. (2019). Performance of machine learning approaches on prediction of esophageal varices for Egyptian chronic hepatitis C patients. Informatics in Medicine Unlocked, 17, 100267.

[2] Abidin, N. Z., Ismail, A. R., and Emran, N. A. (2018). Performance analysis of machine learning algorithms for missing value imputation. International Journal of Advanced Computer Science and Applications (IJACSA), 9(6), 442-447.

[3] Abonazel, M. R., and Ibrahim, M. G. (2018). On estimation methods for binary logistic regression model with missing values. International Journal of Mathematics and Computational Science, 4(3), 79-85.

[4] Alabi, R. O., Elmusrati, M., Sawazaki-Calone, I., Kowalski, L. P., Haglund, C., Coletta, R. D., and Leivo, I. (2020). Comparison of supervised machine learning classification techniques in prediction of locoregional recurrences in early oral tongue cancer. International journal of medical informatics, 136, 104068.

[5] Almustafa, K. M. (2020). Prediction of heart disease and classifiers' sensitivity analysis. BMC bioinformatics, 21(1), 1-18.

[6] Ayodele TO. Types of machine learning algorithms. New advances in machine learning. 2010; 3:19-48.

[7] Chaki, D., Das, A., and Zaber, M. I. (2015). A comparison of three discrete methods for classification of heart disease data. Bangladesh Journal of Scientific and Industrial Research, 50(4), 293-296.

[8] Danubianu, M. (2015). Step by step data preprocessing for data mining. A case study. In Proc. Of the International Conference on Information Technologies (InfoTech-2015) (pp. 117-124).

[9] David, H., & Belcy, S. A. (2018). HEART DISEASE PREDICTION USING DATA MINING TECHNIQUES. ICTACT Journal on Soft Computing, 9(1).

[10] Huapaya, H. D., Rodriguez, C., and Esenarro, D. (2020). Comparative analysis of supervised machine learning algorithms for heart disease detection.

[11] Islam, M. M., Haque, M. R., Iqbal, H., Hasan, M. M., Hasan, M., and Kabir, M. N. (2020). Breast cancer prediction: a comparative study using machine learning techniques. SN Computer Science, 1(5), 1-14.

[12] Kamble, M. S., Desai, M. A., and Vartak, M. P. (2014). Evaluation and Performance Analysis of Machine Learning Algorithms. Neural Networks, 2(3).

[13] Latha, C. B. C., & Jeeva, S. C. (2019). Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. Informatics in Medicine Unlocked, 16, 100203.

[14] Shah, D., Patel, S., and Bharti, S. K. (2020). Heart disease prediction using machine learning techniques. SN Computer Science, 1(6), 1-6.

[15] Rawal, R. (2020). Breast Cancer Prediction using Machine Learning. Journal of Emerging Technologies and Innovative Research (JETIR), 13(24), 7.

[16] Reddy, N. S. C., Nee, S. S., Min, L. Z., and Ying, C. X. (2019). Classification and feature selection approaches by machine learning techniques: Heart disease prediction. International Journal of Innovative Computing, 9(1).

[17] Singh, A., and Kumar, R. (2020). Heart disease prediction using machine learning algorithms. In 2020 international conference on electrical and electronics engineering (ICE3) (pp. 452-457). IEEE.

[18] Singh, D., & Samagh, J. S. (2020). A comprehensive review of heart disease prediction using machine learning. Journal of Critical Reviews, 7(12), 281-285.

[19] Theerthagiri, P., Jeena Jacob, I., Usha Ruby, A., and Yendapalli, V. (2021). Prediction of COVID-19 Possibilities using K-Nearest Neighbour Classification Algorithm. Int J Cur Res Rev| Vol, 13(06), 156.