Iraqi Journal of Information and Communications Technology(IJICT)
Conference Series: The 1st Conference of Applied Researches in Information Engineering(ARIE2021), 2021

ISSN:2222-758X
e-ISSN:2789-7362

HEART DISEASE PREDICTION SYSTEM USING MACHINE LEARNING ALGORITHM

Israa Nadheer [1], Mohammad Ayache [2], Hussein Kanaan [3]
[1,2,3] Faculty of Engineering, Islamic University of Lebanon, Lebanon
israanadheer@gmail.com [1] , {mohammad.ayache, hussein.kanaan}@iul.edu.lb [2,3]
Received:11/4/2021, Accepted:10/6/2021

***Abstract*- Information decision support systems are becoming more in use as we are living in the era of digital data and the rise of artificial intelligence. Heart disease is one of the most known and dangerous is getting very important attention, this attention is translated into digital and prediction system that detects the presence of disease according to the available data and information. In this paper we propose a Heart Disease Prediction System using Machine Learning Algorithms, in terms of data we used the Cleveland dataset, this dataset is normalized then divided into three scenarios in terms of training and testing respectively, 80%-20%, 50%-50%, 3%-70%. In each case of the dataset if it is normalized or not we will have these three scenarios. We used three machine learning algorithms for every scenario mentioned before which support- vector machine (SVM), Sequential minimal optimization (SMO) and multilayer perceptron (MLP), in these algorithms we've used two different kernels to test the results upon that. These two types of simulation are added to the collection of scenarios mentioned above to become like the following we have at the main level two types normalized and unnormalized dataset, then for each one we have three types according to the amount of training and testing dataset, for each of these scenarios we have two scenarios according to the type of kernel to become 30 scenarios in total, our proposed system have shown dominance in terms of accuracy over the other previous works.**

keywords: **Heart disease, Machine learning, Cleveland dataset.**

I. INTRODUCTION

The problem of heart disease is widely popular among all cultures, it is the leading cause of death for men, women, and people of most racial and ethnic groups in the world, according to CDC (Centers od Disease Control and Prediction) 1 person dies every 36 seconds in the united states from heart disease, 655000 Americans die from heart disease every year, it costs the United States about 219 billion dollars each year which includes the cost of health care service, medicines and lost productivity due to death. Heart disease prediction systems are usually composed of 3 layers, dataset layer, feature-selection layer and classification layer. Dataset layer is designed in a way to import the appropriate dataset with some filtering or processing implemented, some kind of filtering or processing is normalization which is an effective way to formulate the data which will facilitate the classification process. We divide the dataset into 70% to tell the model what are the facts to base its decision on as much as we increase the training dataset we will enrich the model with a lot of facts and therefore better decisions will be given, on the other hand, testing data is used to test our model before putting it into use. The next level is feature selection, in this level we choose a machine learning algorithm that analyzes them and selects the most appropriate and the most valuable features. Datasets have a large number of features which is not good for analysis since a large number of features increase the complexity of the model and thus the model will focus on several dimensions of data while abstracting the dimensions into smaller numbers will help more coring into a successful result. The final layer is the analysis part, it is the most important component in such a system, at this layer the model which is used for analysis is built and it is the core of the system. Statistical algorithms use probability concepts to state the result without feeding the model with data, these types are called non-supervised algorithms. Supervised algorithms which are called machine

ISSN:2222-758X
e-ISSN:2789-7362

learning algorithms, after choosing the dataset, dividing it into training and testing and apply a feature-selection algorithm on it we start modelling the algorithm, we start picking up the small details of it to fit our expectations. Several known algorithms are used in the terms of supervised like SVM, Neural Network, SMO, Decision Tree, Naive Bayes, Random Forest and Apriori, these are the most effective in terms of prediction for prevention. The main objective of this study is to develop an accurate and efficient heart disease hybrid prediction system using machine learning algorithms.

TABLE I
Heart Disease Factors

Risk factors	Description	General Symptom
Age	Old people are more suffers from heart disease	Chain pain Shortness of breath Irregular heartbeat Fatigue Fainting Swollen feet
Sex	Males are at greater risk than females	
Family history	If relatives have heart disease the probability of a person to have cardiovascular disease is high	
Smoking	Heart disease higher in smokers than nonsmokers people	
Poor diet	Diet food is essential for development of heart	
Blood pressure	Blood pressure can effect in narrowing hardening arteries, as well as thickening blood vessels[1], [2].	
High blood cholesterol levels	It increases formation of plaques	
Diabetes	It is the disease as a result of sugar in our body	
Obesity	Overweight body is one of the cause for heart diseases	
Physical inactivity	Physical activity helps heart to function properly	
Stress	Damage arteries	
Poor hygiene	It increases heart disease	

In the above table, we will find that almost all the key factors for heart disease are features in the dataset that we will use, we will talk about that in the next chapter. There exist some factors in the table that we will not find it directly in the dataset but it will be an image for a numerical value that represents its existence in the heart disease factor. Many machine learning algorithms equips data decision systems with logical and typical behaviour depending on the dataset used. The algorithm's usage is derived by the classification, as classification guides the user for business decisions. In Fig. 1 we can see a general architecture of data-decision systems for heart disease prediction, as we can see several sandard components may see in different names in different instances but the main goal remains the same. Starting with the dataset, it is imported for the next phase, after that comes data processing where data is cleaned and a feature selection process is applied to the dataset to extract the best features that have an impact on the classification process. In the next phase, we have to choose which machine learning algorithm we have to use. In the next phase, what we have is an empty model of machine learning choose and a dataset, what happens in this layer is that we train our model using the processed dataset, in this case, we will fill our model with logical use cases and which will make it prepared to the next layer. In the final phase, we can test the model we train, so after dividing the dataset in the first stages into training and testing, we use the testing part of the dataset to test the algorithm.

ISSN:2222-758X
e-ISSN:2789-7362

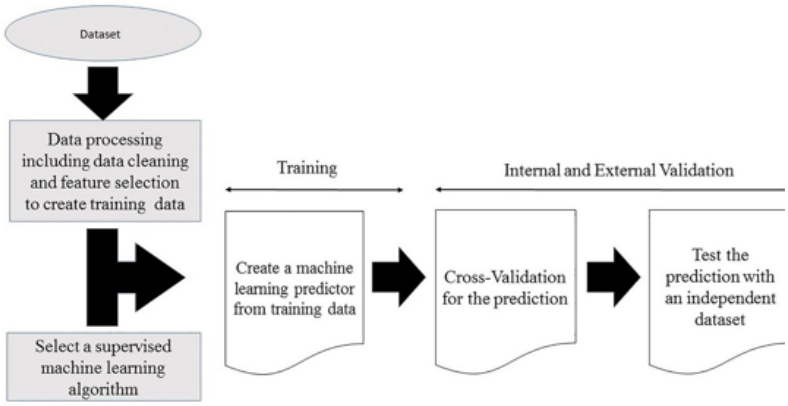

Figure 1: Heart disease prediction system design

II. Related Work

In the literature review, we've done a lot of research in that field and extracted their results, a lot of works and papers have worked in predicting and solving the problem of heart disease. In [1] , the author exploited the fast correlation-based feature selection (FCBF) method to filter redundant features to improve the quality of heart disease classification. Then, we perform a classification based on different algorithms such that K-Nearest Neighbor, Support Vector Machine, Naive Bayes, Random Forest and Multilayer Perception Artificial Neural Network optimized by Particle Swarm Optimization (PSO) combined with Ant Colony Optimization (ACO) approaches. The proposed mixed approach is applied to the heart disease dataset and the results demonstrate the efficacy and robustness of the proposed hybrid method in processing various types of data for heart disease classification. Finally, A prototype heart disease prediction system is developed using three data mining techniques, the most effective model to predict patients with heart disease appears to be Naive Bayes followed by Neural Network and then Decision Tree. In [2] , the author proposed a heart disease diagnosis system to predict more accurately the disease using genetic algorithms as feature selection and Naive Bayes, Clustering and Decision Tree as classification algorithms, the system used "Sellapan et al" as a dataset with 909 records and 13 attributes, after applying the genetic algorithm on the dataset 6 attributes were left. As for the results, the observations exhibit that the Decision Tree algorithm outperforms the other two techniques after incorporating feature subset selection with relatively high model construction time. Naive Bayes performs consistently before and after the reduction of attributes with the same model construction time. Classification via clustering performs poorly compared to the other two methods. In [3] , the author proposed an efficient associative classification algorithm using the genetic approach for heart disease prediction. The main motivation for using genetic algorithms in the discovery of high-level prediction rules is that the discovered rules are highly comprehensible, having high predictive accuracy and of high interestingness values. Experimental results show that most of the classifier rules help in the best prediction of heart disease which even helps doctors in their diagnosis decisions, this system showed dominance over all other algorithms and systems in different datasets which is a very important case. In [4] , the author proposed an MDSS heart disease classification based on sequential minimal optimization (SMO)

ISSN:2222-758X
e-ISSN:2789-7362

technique in support vector machine (SVM). We illustrated the UCI machines learning repository of the Cleveland heart disease database, we trained SVM by using the SMO technique and this training requires the solution for a very large QP optimization problem. SMO algorithm breaks this large optimization problem into small sub-problems. The results proved that MDSS can carry out heart disease diagnosis accurately in a fast way and on a large dataset in other words it shows the good ability of prediction. In [5] , the author proposed an approach that combines KNN and genetic algorithms to improve the classification accuracy of the heart disease dataset. They used genetic search as a good measure to prune redundant and irrelevant attributes and to rank the attributes which contribute more towards classification. The least ranked attributes are removed, and the classification algorithm is built based on evaluated attributes. This classifier was trained to classify the heart disease data set as either healthy or sick. As a result, the performance of our proposed approach has been tested with 6 medical data sets and 1 non-medical dataset. Out of 7 datasets, 6 datasets were chosen from the UCI repository and heart disease A.P was taken from various corporate hospitals in Andhra Pradesh, finally, the comparison of the author proposed algorithm with 4 algorithms showed its dominance in terms of accuracy. In [6], the author aims to design and implement an automatic heart disease diagnosis system using Matlab. The Cleveland dataset for heart disease was used as the main database for training and testing the developed system. To train and test the system, two sub-systems were developed.

1) The first system is based on the Multilayer Perceptron (MLP) structure on the Artificial Neural Network (ANN) .
2) The second system is based on the Adaptive Neuro-Fuzzy Inference Systems (ANFIS) approach. Each system has two main modules, namely, training and testing, where 80% and 20% of the Cleveland data set were randomly selected for training and testing purposes respectively.

In [7] , the author proposed an enhanced framework for heart disease prediction, the object was to predict more accurately the presence of heart disease with the reduced number of attributes. Thirteen attributes were reduced to 6 attributes using genetic search. Subsequently, three classifiers like Naive Bayes, Clustering and Decision Tree were used to predict the diagnosis of patients with the same accuracy as obtained before the reduction of several attributes. As a result, the attribute filtering method of PCA selects the 6 attributes and Information gain filters 8 attributes out of 14 attributes. The classifiers accuracy with the full dataset is more for ANN than SVM, classification tree and Naive Bayes algorithm. The classification accuracy for ANN is 83.70% and Naive Bayesian Classification is 82.75% than the other methods. And from table V, the Information gain and PCA attribute algorithm gives more reduced attributes than the CFS and Gain ratio attribute filtering methods. In [8] , the author proposed a heart disease diagnosis system using rough sets based attribute reduction for feature selection and interval type-2 fuzzy logic system for classification. The system uses the dataset "SPECTF" , it applies normalization before using the rough sets based attribute reduction algorithm for feature selection, after that a subset of the dataset is created. In the second layer, the interval type-2 fuzzy logic system was applied to training data while the fuzzy logic system was applied to the testing data. As a result, we can see that in the heart disease dataset the author's approach shows higher accuracy than when applied to SPECTF dataset. In [9] , the author proposed a novel method to determine the disease using Cleveland Heart Disease Dataset by combining the computational power of various ML and DM algorithms and concluded that among all the algorithms, K-Nearest Neighbors gives the highest accuracy of 87%. Along with this,

ISSN:2222-758X
e-ISSN:2789-7362

a web app is developed using a flask in python with which the user can enter the attributes and predict heart disease. The system uses Cleveland Heart Disease Dataset without normalization where no attribute reduction algorithm for feature selection was used, dataset has been divided into training and testing 70% is used for training and 30% for testing. In the second layer training data is applied to Decision Trees, K-Nearest Neighbor, Support Vector Machine and Random Forest. Decision Tree gave an accuracy of 79%, K-Nearest Neighbor gives an accuracy of 87%, Support vector Machines gives an accuracy of 83%, and Random Forest Gives an accuracy of 84%. In [10] , the author proposed an effective heart disease prediction model (HDPM) for a CDSS which consists of Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to detect and eliminate the outliers, a hybrid Synthetic Minority Over-sampling Technique-Edited Nearest Neighbor (SMOTE-ENN) to balance the training data distribution and XGBoost to predict heart disease. The system uses two publicly available datasets (Statlog and Cleveland) without normalization where no attribute reduction algorithm for feature selection was used, dataset has been divided into training and testing 70% is used for training and 30% for testing. In the second layer training data is applied to (naive bayes (NB), logistic regression (LR), multilayer perceptron (MLP), support vector machine (SVM), decision tree (DT). Finally, the overall designed and developed HDCDSS in this study can be used as a practical guideline for healthcare practitioners. In [11] , the author proposed a deep learning classification system using different layers of convolution, rectifier and pooling operations that can be used to increase feature extraction of ECG signal. We have proposed two models, one is used 1-D signal, in which we designed a model for classification csv type of data for ECG signal, while in the second proposed system, we used model for 2-D signal after convert it from its csv type . 2-D signal (ECG image) is used to augment the two-dimensional signal with different methods to increase the accuracy of the model by training it with the geometric transformation of the original input images such as rotation, shearing etc. The results are compared with AlexNet and other models based on different metrics, which are used to measure the performance of the proposed work, the result shows that the proposed models improve the efficiency of the classification in the two systems.

III. PROPOSED MODEL

We are living in the era of digital data and decision systems, systems that make business decisions based on data are very attractive and implemented in several industries, these systems are support to the human decisions, in big companies decisions are not based on just machine system, they came as a support for the human decision finally decides the pattern or the way the company has to follow. Heart disease is a very popular disease and is an example of an industry that needs such systems, the critical point here is that the problem is very sensitive where the decision may decide a person will live or die which is very important to the healthcare system, at the same time this makes the need for a data-decision system very important and necessary. In our approach we worked on a prediction data-decision system that is based on the Cleveland dataset, a prediction system that is composed of several algorithms combined to form one system, the aim of this system is the classification of heart disease data to predict which records or patients are cure and which of them has the disease. Our system is based on 3 machine learning algorithms SMO, SVM and MLP, the dataset used in Cleveland.

ISSN:2222-758X
e-ISSN:2789-7362

TABLE II
Comparison Between Algorithms in Previous Work

Year	Title	Author	Method	Advantages	Disadvantages
2010	Enhanced Prediction of Heart Disease with Feature Subset Selection using Genetic Algorithm	• M.ANBARASI • E.ANUPRIY • N.CH.S.N.IYENGAR	• Genetic algorithm • Naïve Bayes • CVC • Decision Tree	• Predict more accurately heart disease • 13 attributes are reduced to 6 • High model construction time	• Inconsistencies and missing values • Intensity of the disease based on the results was unpredictable
2012	Heart Disease Prediction System using Associative Classification and Genetic Algorithm	• M.Akhil jabbar • Dr.Priti Chandra • Dr.B.L Deekshatulu	• Associative Classification • Genetic algorithm	• Achieves maximum accuracy • Rules are highly comprehensible, • having high predictive accuracy and of high interestingness values	• Attributes are not reduced
2013	DECISION SUPPORT SYSTEM FOR HEART DISEASE BASED ON SEQUENTIAL MINIMAL OPTIMIZATION IN SUPPORT VECTOR MACHINE	• Deepti Vadicherla • Sheetal Sonawane	• SVM • SMO	• Results are accurately in fast way and on a large dataset • Use of SMO to solve very large QP optimization problem	• No feature-selection is used • Absence of shrinking and Kernel caching in SMO
2013	Classification of Heart Disease Using K-Nearest Neighbor and Genetic Algorithm	• M.Akhil jabbar • B.L Deekshatulu • Priti Chandra	• KNN • Genetic algorithm	• Highly efficient and effective algorithm for pattern recognition	• Redundant and irrelevant attributes produce less accurate result
2014	Automatic Heart Disease Diagnosis System Based on Artificial Neural Network (ANN) and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) Approaches	• Mohammad A. M. Abushariah • Assal A. M. Alqudah • Omar Y. Adwan • Rana M. M. Yousef	• MLP • Neuro-fuzzy	• 2 separated systems with testing and training module • Each system has case-based module	• No feature-selection algorithm is used • Weak MLP architecture
2015	Propose a Enhanced Framework for Prediction of Heart Disease	• K. Sudhakar • Dr. M. Manimekalai	• PCA • Info-gain • Naïve Bayes • Neural network • CFS • SVM • Classification tree	• Multi-algorithms used in feature-selection layer • Classification layer based on feature-selection layer that produces high accuracy	• Time consumer
2015	A highly accurate firefly based algorithm for heart disease prediction	• Nguyen Cong Long • Phayung Meesad • Herwig Unger	• Rough sets based attribute reduction • Chaos Firefly • Fuzzy Logic • Interval Type-2 Fuzzy Logic System	• Handle with high-dimensional dataset • Reduces computational burden and enhances performance	• Learning process is computationally expensive • Rough sets based attribute reduction is unmanageable when NB of attributes are large
2018	Heart Disease Prediction and Classification Using Machine Learning Algorithms Optimized by Particle Swarm Optimization and Ant Colony Optimization	• Youness Khourdifi • Mohamed Bahaj	• FCBF • K-Nearest Neighbor • SVM • Naïve Bayes • Random Forest • MLP • PSO • ASO	• High efficacy and robustness • Hybrid system in terms of feature-selection and classification layers	• Very high processing power • Large time consumer
2020	Heart Disease Prediction using Machine Learning and Data Mining	• Keshav Srivastava • Dilip Kumar Choubey	• DT • KNN • SVM • Random forest	• Web app interface	• Absence of feature-selection algorithm
2020	HDPM: An Effective Heart Disease Prediction Model for a Clinical Decision Support System	• NORMA LATIF FITRIYANI • MUHAMMAD SYAFRUDIN • GANJAR ALFIAN • JONGTAE RHEE	• Over-sampling Technique-Edited nearest Neighbor • XGBoost	• Density-Based Spatial Clustering of Applications • Noise (DBSCAN) to detect and eliminate the outliers	• Absence of different datasets • Absence of privacy and security properties
2020	ECG SIGNAL CLASSIFICATION BASED ON DEEP LEARNING BY USING CONVOLUTIONAL NEURAL NETWORK (CNN)	• Aqeel M.hamad alhussainy • Ammar D.Jasim	• CNN(1-D vector) • CNN(2-D vector)	• Usage of ECG signal as a metric for detecting heart disease • Use of rotation shearing of input images to increase the accuracy of the model	• Absence of feature-selection algorithm • Datasets used are weak

ISSN:2222-758X
e-ISSN:2789-7362

The block diagram below explains at the abstract level the flow of our systems in addition to the relation between different blocks of the system. The diagram starts with importing the dataset into our system, then we start normalizing the dataset by running Matlab code that outputs the normalized dataset. The next step is dataset separation, in this step, we divide our dataset into three sub-datasets in parallel, in other words, we create three parallel scenarios in each scenario we modify our dataset and separate it into two parts. In the first scenario we separated the dataset into 80% training and 20% testing, in this scenario we applied the machine learning algorithms all to training and then to testing to get the results. In the second scenario, we separated the dataset into 50% training and 50% testing, then we repeated the steps as in the previous scenario. In the third scenario we separated the dataset into 30% training and 70% testing, also we repeated the steps as in the previous scenarios. All these scenarios are executed separately which means the execution of a scenario will not affect the other scenario, these are different executions

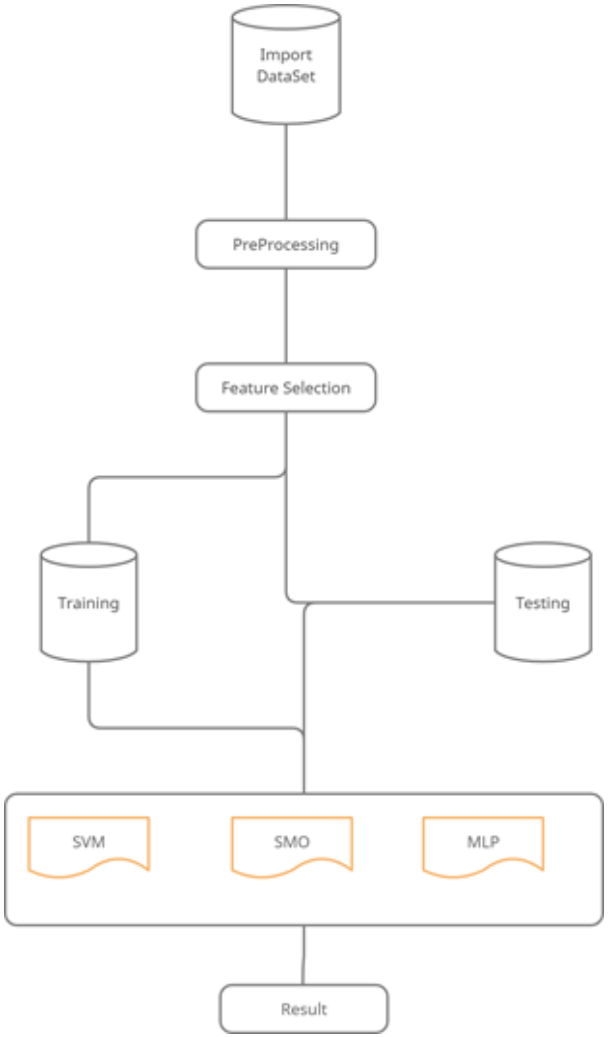

Figure 2: Block diagram

ISSN:2222-758X
e-ISSN:2789-7362

A. Database

In this work we used the Cleveland dataset, this dataset can be found on this link https://archive.ics.uci.edu/ml/datasets/heart+disease, this dataset is widely used among heart disease systems, it contains a lot of use cases and features for better classification, the number of records we used in our system is 303 and the number of features was 14, below we will list all the public datasets for heart disease with the number of records:

1) Cleveland: 303
2) Hungarian: 294
3) Switzerland: 123
4) Long Beach VA: 200

TABLE III
Dataset Features

No	Attributes	Description
1	Age	Age in year
2	Sex	0 for female and 1 for male
3	Cp	Chest pain type Value 1: typical angina Value 2: atypical angina Value 3: non-anginal pain Value 4: asymptomatic
4	Trestbps	Resting blood sugar in mm Hg on admission to the hospital
5	Chol	Serum cholesterol in mg/dl
6	Fbd	(Fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)
7	Restecg	Resting ECG result
8	Thalach	Maximum heart rate achieved
9	Exang	Exercise induced angina
10	Oldpeak	ST depression induced by exercise relative to rest
11	Slope	Slope or peak exercise ST segment
12	Ca	Number of major vessels colored by fluoroscopy
13	Thal	Defect type
14	num	The predicted attribute

Above we saw 14 attributes with their description, these attributes are common between all the datasets we've listed above. As all the attributes are clearly explained in the description, the only attributes that we need to explain its importance in the classification process and which plays an important role in the system is the last attribute, this attribute is the predicted attribute which comes with a value 0 or 1 which explains if this record holds information for a cured patient or sick patient, if the value is 1 then the patient is sick if it is 0 then the patient is not sick. This attribute is automatically filled with the dataset, the values of this attribute are real and are used as a base for the training of the algorithms and which form a base for the classification criteria, without it the dataset has no meaning and next steps which is classification and prediction will fail. After the step of exposing the dataset, we are now ready to describe what is the ext operation applied to the dataset. Normalization is the method applied to the dataset, we need to formulate the values of the dataset to facilitate the

ISSN:2222-758X
e-ISSN:2789-7362

mission of machine learning algorithms. We used the "Standardization" method for normalization, in other words, ZScore, a random variable X with mean μ and standard deviation σ, the Z-Score of a value x is:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

x is the value of a specific column in a specific record. μ is the mean of each column in the dataset. σ is the standard deviation of each column in the dataset. Standardization measures the distance of a data point from the mean in terms of the standard deviation, as a result, the dataset has a mean 0 and STD equal to 1. The shape properties of the original data will be retained (same skewness and kurtosis). Standardization can be used to put data on the same scale before further analysis.

B. Classification

In our work, we worked on three machine learning algorithms SVM, SMO and MLP, in this section we are going to explain the implementation and properties of these algorithms.

1) MLP: Starting with MLP we used three scenarios, the first one 80% for training and 20% testing from the dataset, the second one 50% train 50% test, the third one 30% train 70% test, the dataset contains 303 records, Two hidden layers were used, the first one contains 3 neurons and the second one contains 4 neurons, the input layer has 14 neurons, while the output layer contains 1 neuron, below is the architecture of the neural network we have used:

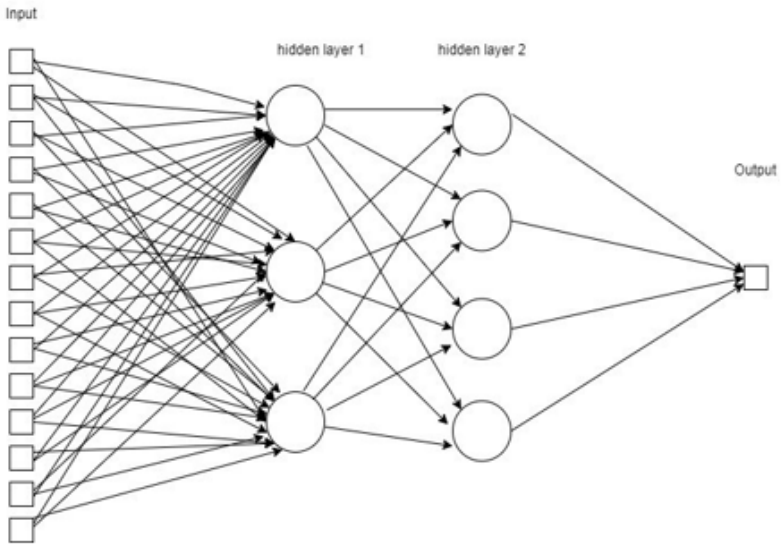

Figure 3: Multilayer perceptron architecture

As we can see that we have the input, output and two hidden layers. The input is composed of 14 neurons which is the number of features in the Cleveland dataset, these 14 neurons were chosen all since we need to count all the features value of the patient, and since we didn't apply the feature selection algorithm before we apply classification,

ISSN:2222-758X
e-ISSN:2789-7362

besides we need to analyze all the record for the patient to get a precise result. At the level of hidden layers we had two, the first hidden layer contains three neurons, each neuron is connected to all neurons of the input vector, this is done to count all the feature values into our model calculations, and no connection is done between the neurons at the same level. In the second hidden layer, we had four neurons that are also connected to all the neurons coming from the first hidden layer that is done to count all the results from the first layer and to be calculated as inputs in the second layer. At the level of output, we have one neuron, since the output is a Boolean result which means the patient is sick or not sick so the result will be one element. Our MLP network contains elements and details more than layers and neurons, it also contains activation function, several activation functions exist in a neural network in general and thus in MLP like:

a) Sigmoid function
b) Hyperbolic tangent function
c) Tangent Function
d) Softmax Function
e) Softsign function
f) Rectified Linear Unit (ReLU) function
g) Exponential Linear Units (ELUs) Function

The one used in MLP is the Hyperbolic tangent function (tanh). The hyperbolic tangent function the tanh function is another type of AF. It is a smoother, zero-centred function having a range between -1 to 1.

2) SVM: In SVM we used 3 scenarios, the first one 80% for training and 20% testing from the dataset, the second one 50% train 50% test, the third one 30% train 70% test, the dataset contains 303 records. For each scenario of these, we use RBF and Linear Kernel, both of them were applied to each scenario. The aim of this multiple implementations of kernels is that we wanted to make a descriptive comparison between the two kernels according to each scenario. Regarding feature selection, we used sequential feature selection in Matlab in SVM to reduce the dimensionality of data. One more point to be focused on is that we tested SVM with and without Normalization. Results and graphs will be showed in chapter 4.
3) SMO: In SMO we used SMO in three scenarios, the first one 80% for training and 20% testing from dataset, the second one 50% train 50% test, the third one 30% train 70% test, besides the dataset contains 303 records. For each scenario of these, we applied SMO using RBF and Linear kernel functions. The modification was applied to the dataset before applied to the algorithm, we split data into two parts the first part contains all the data columns instead of the last column, the second part contains only the last column which represents the actual result of the record (1 cure, 0 sick). Several parameters were used with SMO:

a) Max iterations was specified to 105.
b) The regularization parameter was set to 10 (tells the SVM optimization how much you want to avoid misclassifying).
c) Tolerance was specified to $1e^{-4}$ (Higher tolerance means high precision).

ISSN:2222-758X
e-ISSN:2789-7362

d) $EPS = 1e^{-3}$ (Epsilon), (The value of epsilon determines the level of accuracy of the approximated function. It relies entirely on the target values in the training set. If epsilon is larger than the range of the target values we cannot expect a good result. If epsilon is zero, we can expect to overfit. Epsilon must therefore be chosen to reflect the data in some way).

e) Type = 'R' (Type variable determines the type of kernel that will be used in SMO, there are four types which are 1, r, p or s) . Below is the flow diagram of the SMO algorithm: The block diagram above explains the flow of logic in the SMO algorithm, the algorithm starts with consuming the value of data then starting a while loop that repeats itself for a given number N, this number of iterations is given as a parameter for SMO, this parameter is called Max iterations. Inside the loop, we check the quadratic equation of SVM, if it needs more optimization we continue and take a random value from the dataset fitting the quadratic equation, after that we check the quadratic function applied on the dataset if it is optimized or not, if not we repeat the steps from the beginning of the loop, if yes optimized then while loop is ended and results are displayed.

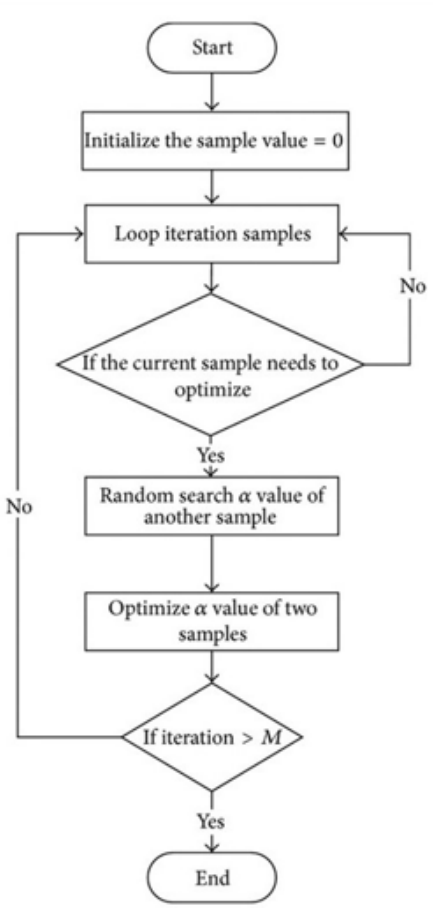

Figure 4: SMO flow chart

ISSN:2222-758X
e-ISSN:2789-7362

IV. SIMULATION

A. Simulation Scenario

In the simulation we tested three algorithms SVM, SMO and MLP, in each of the algorithms we formed 12 scenarios, these scenarios are formed at the first level between normalized and unnormalized, then at the second level for RBF and linear kernels, at the last level, we have the three scenarios of dataset distribution on training and testing which are 80%-20%, 50%-50% and 30%-70% this distribution of scenarios was applied on SVM and SMO except MLP since it does not have kernel so i total we will have 30 scenarios.

B. MLP

In this section, we will expose all the results of MLP algorithm, including normalized and unnormalized datasets in 80-20%, 50-50 % and 30-70 %:

TABLE IV
Simulation Parameters

Parameters	Value
Simulator	Matlab 2017
Platform	Windows 10
PC Type	ASUS GL752 VW
Processor	Intel(R) Core(TM) i7-6700HQ CPU @ 2.6 GHz
RAM	16 GB RAM
Graphics	NIVIDIA GeForce GTX 960M / 8 GB

In Fig. 5 we can see that the classification in MLP using the unnormalized dataset through 80-20 training and testing scenario has given a great result, 86.9565 is the accuracy of the algorithm, in Fig. 5 we can see the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2. In Fig. 6 we can see the results from training and testing MLP using the unnormalized dataset through 50-50 training and testing scenario has given a great result, 84.7068 is the accuracy of the algorithm, in Fig. 6 we can see the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2. In this case, lower accuracy indicates that the training dataset has a very important role in increasing the accuracy, this increase is forced by the important role that is played during the training of our model. This role is explained by enriching our model by logical scenarios that will feed the testing scenario which will result in final decision about the status of the patient. Finally this improves the dominance of increasing training dataset over the testing in a balanced percentage to improve the output of our model. In Fig. 7 our result proposed an Accuracy of 81.7073, Fig. 7 shows the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2. In this case lower accuracy indicates that the training dataset has a very important role in increasing the accuracy, this increasing is forced by the important role that is played during the training of our model. This role is explained by enriching our model

ISSN:2222-758X
e-ISSN:2789-7362

by logical scenarios that will feed the testing scenario which will result in final decision about the status of the patient. Finally this improves the dominance of increasing training dataset over the testing in a balanced percentage to improve the output of our model.

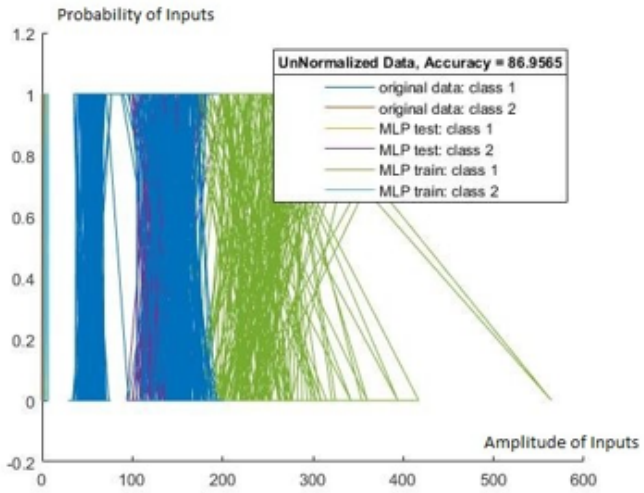

Figure 5: Accuracy in MLP using unnormalized dataset (80%-20%)

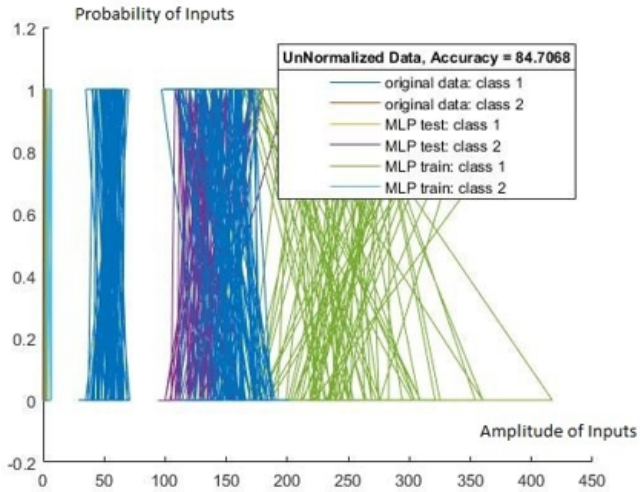

Figure 6: Accuracy in MLP using unnormalized dataset (50%-50%)

In Fig. 8 we can see the large increase in accuracy (94.89%) for 80%-20% training and testing using the normalized Cleveland dataset, this increase is derived mainly by the normalization of dataset which plays an important role in reformatting the data into more simple values, moreover it formulate the data in a way where the mean is equal to 0

ISSN:2222-758X
e-ISSN:2789-7362

and standard deviation equal to 1, this formulation improves the classification process, since normalization facilitates the mathematical model applied in machine learning algorithms.

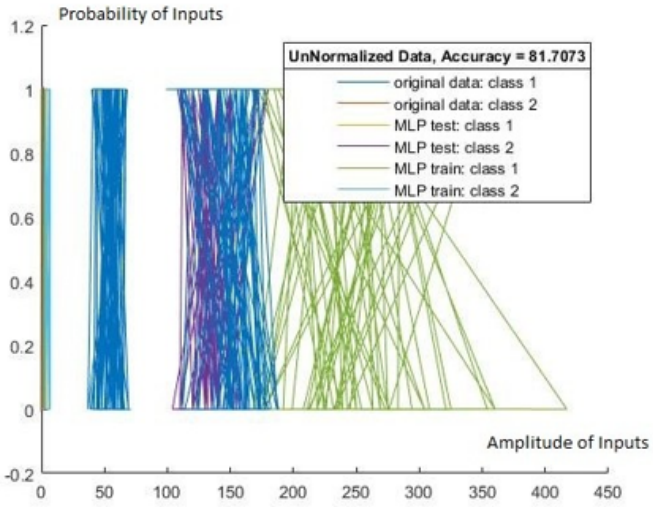

Figure 7: Accuracy in MLP using unnormalized dataset (30%-70%)

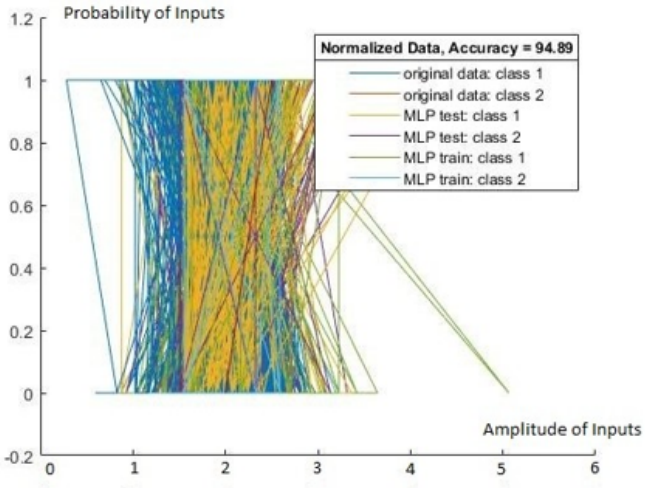

Figure 8: Accuracy in MLP using normalized dataset (80%-20%)

In Fig. 9 we can see the results from training and testing MLP using normalized dataset through 50-50 training and testing scenario has given a great result, 90.458 is the accuracy of the algorithm, this result can be expressed by the increasing number of yellow and purple data since these portions express the testing cases in the simulation. As a result, we have lower accuracy than 80%-20% but we kept on the dominance over the 50%-50% of the unnormalized dataset, this dominance is very important as it points to the importance of normalizing the data before using the MLP algorithm.

ISSN:2222-758X
e-ISSN:2789-7362

In Fig. 10 we can see our results with an Accuracy of 89.5833, the figure shows the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2. In this case, lower accuracy indicates that the training dataset has a very important role in increasing the accuracy, this increase is forced by the important role that is played during the training of our model. As we notice that 89.5833 accuracy is better than all the results using Unnormalized dataset even 80%-20% which is very interesting and shows the importance of normalization on the classification output.

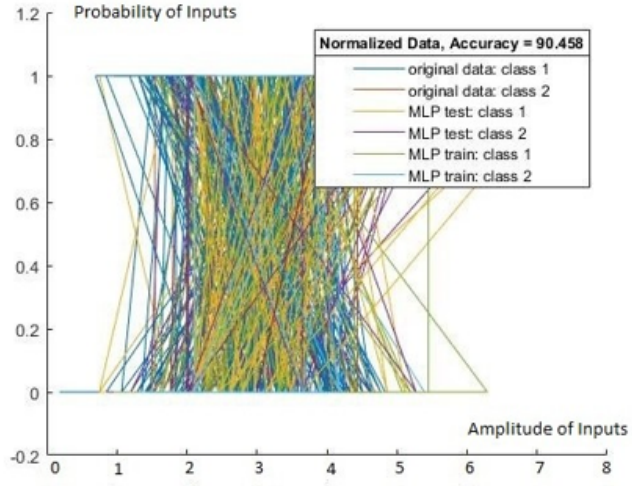

Figure 9: Accuracy in MLP using normalized Dataset (50%-50%)

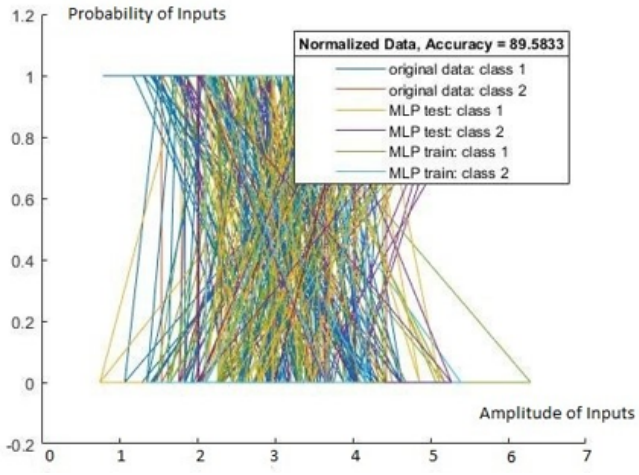

Figure 10: Accuracy in MLP using normalized Dataset (30%-70%)

ISSN:2222-758X
e-ISSN:2789-7362

C. SMO

in this section we will show the figures of the SMO algorithm for unnormalized and normalized datase:
In Fig. 11 we can see the result of applying SMO on the unnormalized dataset using linear kernel in the 80-20 % scenario, as we can see the different colors of points refers to the different class that this point refers to, also different colors will refer to different process which means training or testing. From the figure we can conclude that the data in the testing process are more classified as class 2. The accuracy in the figure which is 81.4734 refers to the accuracy in the testing phase.

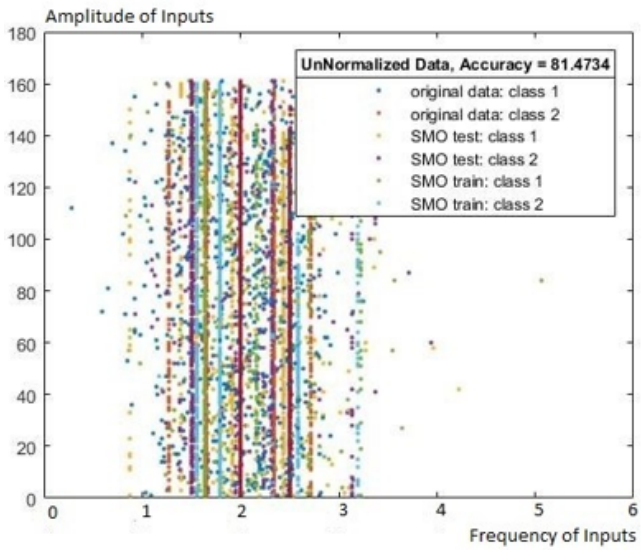

Figure 11: Accuracy in SMO with linear kernel using unnormalized dataset (80%-20%)

In Fig. 12 we can see the results from training and testing SMO using unnormalized dataset through 50-50 training and testing scenario has given a good result, 75.3497 is the accuracy of the algorithm, this result can be noticed throught the yellow and purple portion of data. In this case lower accuracy indicates that the training dataset has a very important role in increasing the accuracy. Finally this improves the dominance of increasing training dataset over the testing in a balanced percentage to improve the output of our model. In Fig. 13 we will analyze the results generated from applying the SMO algorithm on the unnormalized dataset using the linear kernel in the 30-70 % scenario, a different result is obtained from the figure. First of all the accuracy of this scenario is 72.5128% which is smaller than that of 50-50 % and 80-20 %, this is related to the decreasing number of training data, which plays important role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number. In Fig. 14 we can see the result of applying SMO on the unnormalized dataset using RBF kernel in the 80-20 % scenario, as we can see the different colors of points refers to the different class that this point refers to, also different colors will refer to a different process which means training or testing. From the figure above we can conclude that the data in the testing process are

ISSN:2222-758X
e-ISSN:2789-7362

more classified as class 2. The accuracy is pointing at 83.8508 which is greater than the accuracy in the linear kernel for the same conditions, this greater accuracy shows the dominance of using RBF kernel over the linear kernel.

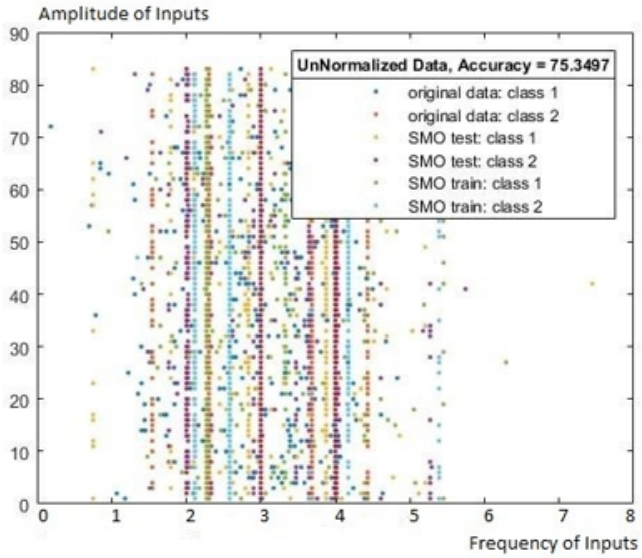

Figure 12: Accuracy in SMO with linear kernel using unnormalized Dataset (50%-50%)

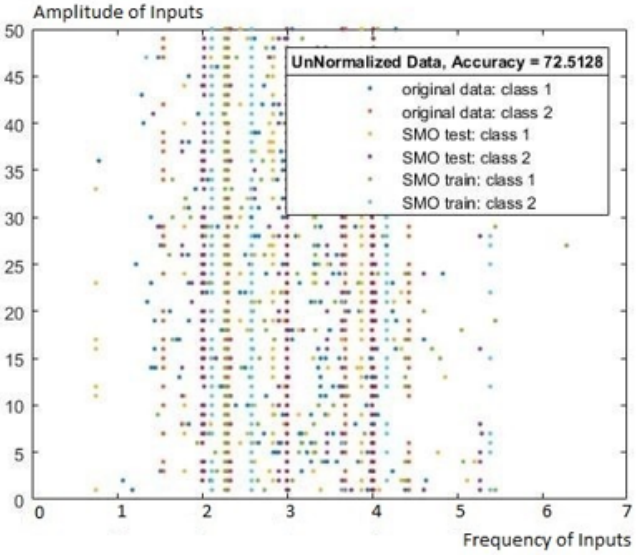

Figure 13: Accuracy in SMO with linear kernel using unnormalized dataset (30%-70%)

In Fig. 15 we can see the results from training and testing SMO using unnormalized dataset through 50-50 training and testing scenario has given a good result, 76.1269 is the accuracy of the algorithm which is lower than the 80-20 %

ISSN:2222-758X
e-ISSN:2789-7362

scenario which normal and expected according to the previous experiments, besides it is greater than 75.3497 which is the percentage using linear kernel which more and more proves the superiority of using RBF kernel over that of linear, in the figure we can obtain the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2.

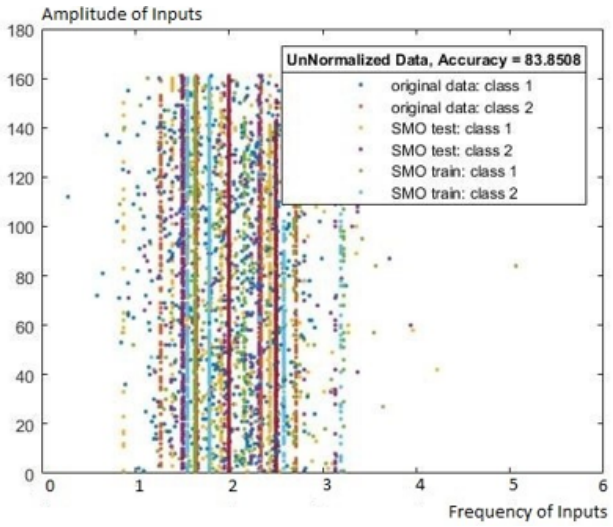

Figure 14: Accuracy in SMO with RBF Kernel using unnormalized dataset (80%-20%)

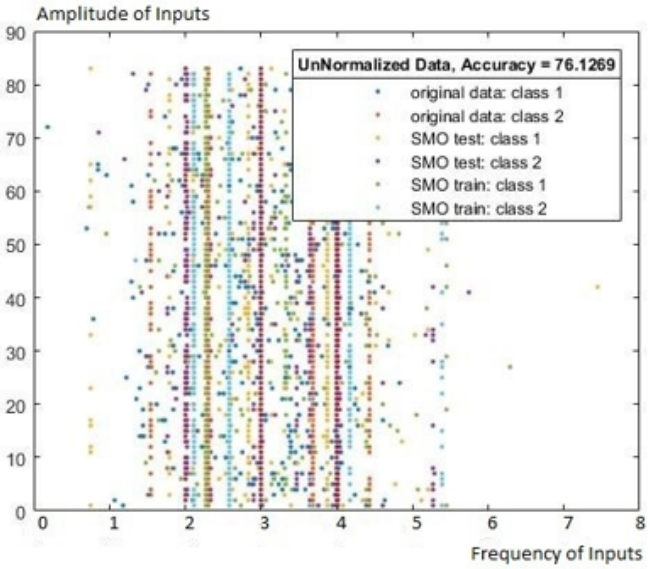

Figure 15: Accuracy in SMO with RBF Kernel using unnormalized dataset (50%-50%)

In Fig. 16 we will analyze the results generated from applying SMO algorithm on unnormalized dataset using RBF

ISSN:2222-758X
e-ISSN:2789-7362

kernel in the 30-70 % scenario, a different result is obtained from the figure. First of all the accuracy of this scenario is 74.6606 which is better than that of the linear kernel (72.5128%) which is also smaller than that of 50-50 % and 80-20 %, this is related to the decreasing number of training data, which plays important role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number.

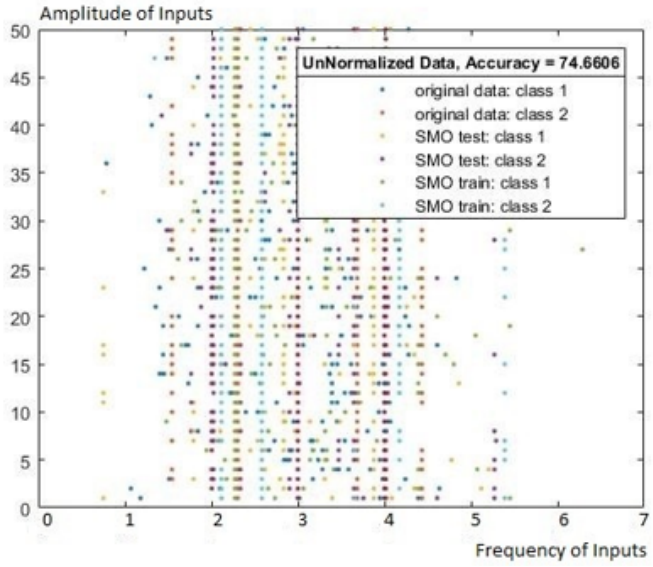

Figure 16: Accuracy in SMO with RBF kernel using unnormalized dataset (30%-70%)

In Fig. 17 we can see the result of applying SMO on the normalized dataset using the linear kernel in the 80-20 % scenario, as we can see the different colors of points refers to the different class that this point refers to, also different colors will refer to a different process which means training or testing. From the figure above we can conclude that the data in the testing process which are colored in purple are more classified as class 2 than class 1 which are colored in yellow. The accuracy in the figure is 93.5709 which is greater than that of RBF and linear in the unnormalized scenario, this shows a greater superiority by using a normalized dataset over that of unnormalized. In Fig. 18 we can see the results from training and testing SMO using normalized dataset through 50-50 training and testing scenario has given a great result, 91.2952 which is smaller than that of 80-20 % which is an expected result, but the huge dominance was over the unnormalized scenarios it shows more than normalized dataset interfere positively in the classification process to produce incredible results, in the figure above we can obtain the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2. In Fig. 19 we will analyze the results generated from applying SMO algorithm on the normalized dataset using the linear kernel in the 30-70 % scenario, a different result is obtained from the figure. First of all the accuracy of this scenario is 89.0394% which is smaller than that of 50-50 % and 80-20 %, this can be indicated by data portions colored in yellow and purple which also shows domination for class 1 represented by yellow data over class 2, this is related to the decreasing number of training data, which plays important

ISSN:2222-758X
e-ISSN:2789-7362

role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number.

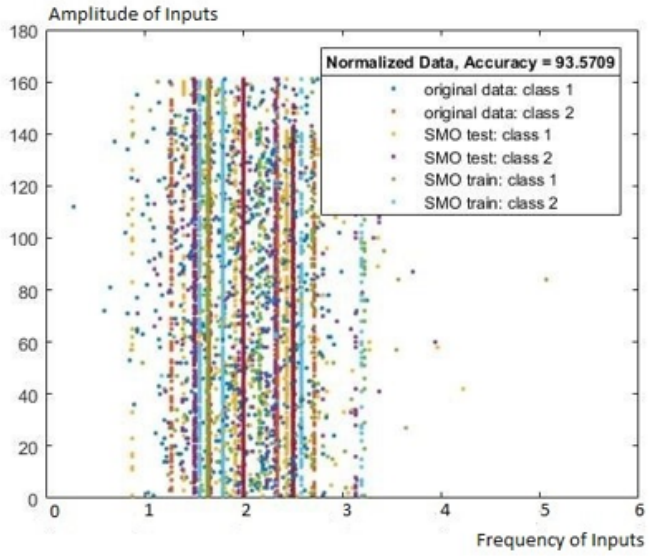

Figure 17: Accuracy in SMO with linear kernel using normalized dataset (80%-20%)

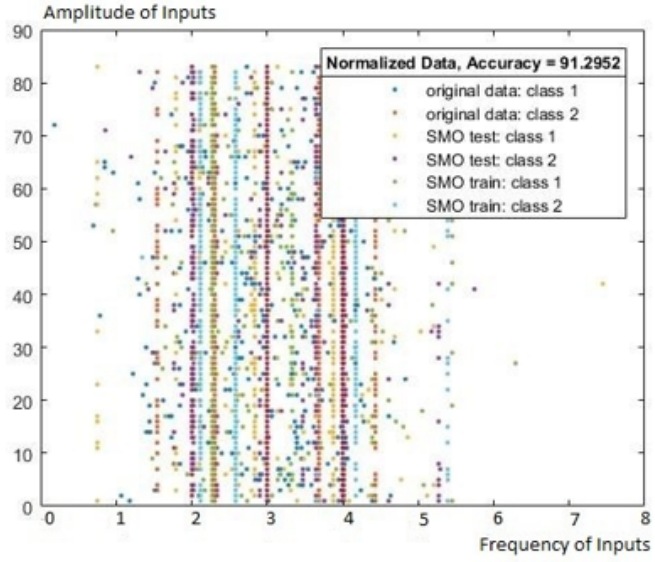

Figure 18: Accuracy in SMO with linear kernel using normalized dataset (50%-50%)

In Fig. 20 we can see the result of applying SMO on normalized dataset using RBF kernel in the 80-20 % scenario, as we can see the different colors of points refers to the different class that this point refers to, also different colors will refer

ISSN:2222-758X
e-ISSN:2789-7362

to different process which means training or testing. From the figure above we can conclude that the data in the testing process are more classified as class 2. The accuracy is pointing at 95.2139 which is greater than the accuracy in the linear kernel for the same conditions, this greater accuracy shows the dominance of using RBF kernel over the linear kernel.

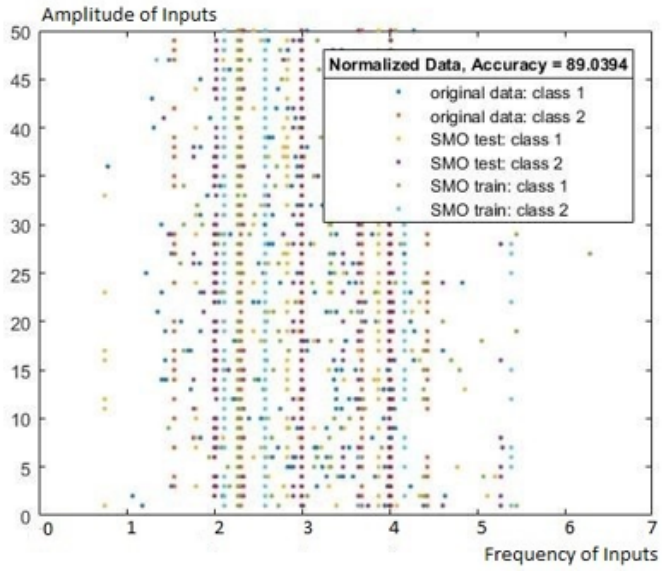

Figure 19: Accuracy in SMO with linear kernel using normalized dataset (30%-70%)

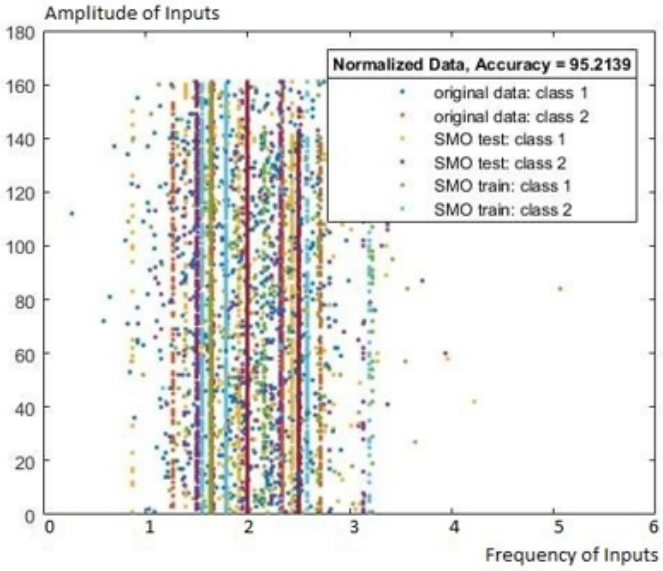

Figure 20: Accuracy in SMO with RBF kernel using normalized dataset (80%-20%)

In Fig. 21 we can see the results from training and testing SMO using normalized dataset through 50-50 training and

ISSN:2222-758X
e-ISSN:2789-7362

testing scenario has given a good result, 92.0942 is the accuracy of the algorithm which is lower than the 80-20 % scenario which normal and expected according to the previous experiments, besides it is greater than that of the unnormalized dataset which more and more proves the superiority of using normalized over the unnormalized and RBF kernel over that of linear, in the figure above we can obtain the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2.

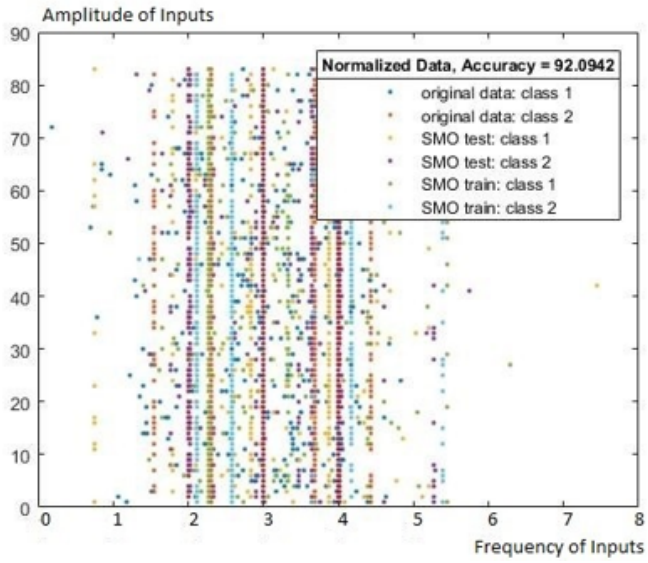

Figure 21: Accuracy in SMO with RBF kernel using normalized dataset (50%-50%)

In Fig. 22 we will analyze the results generated from applying the SMO algorithm on the normalized dataset using RBF kernel in the 30-70 % scenario, a different result is obtained from the figure. First of all the accuracy of this scenario is 89.97 which is better than that of the linear kernel (89.03%) which is also smaller than that of 50-50 % and 80-20 %, this is related to the decreasing number of training data, which plays important role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number.

D. SVM

in this section we will show the figures of the SVM algorithm for unnormalized and Normalized dataset, besides we will analyze each figure by figure and show the advantages and disadvantages of each this approach:

In Fig. 23 above we can see the result of applying SVM on the unnormalized dataset using the linear kernel in the 80-20 % scenario, as we can see the different colors of points refers to the different class that this point refers to, also different colors will refer to a different process which means training or testing. From the figure above we can conclude that the data in the testing process are more classified as class 2. The accuracy in the figure which is 79.9920 refers to the accuracy in the testing phase.

ISSN:2222-758X
e-ISSN:2789-7362

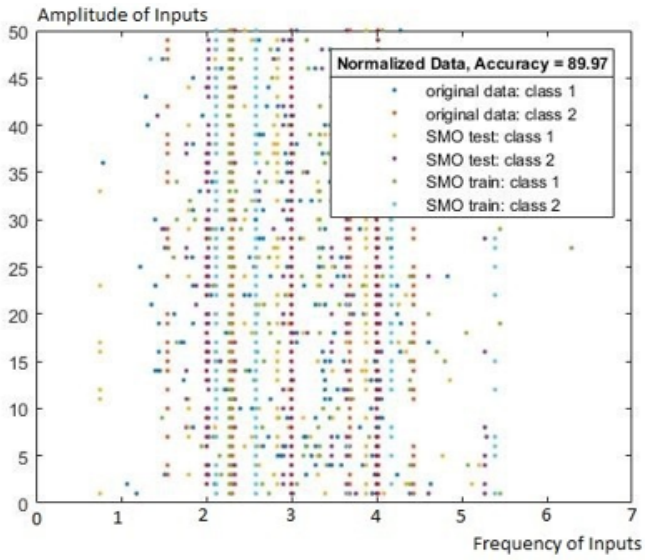

Figure 22: Accuracy in SMO with RBF kernel using normalized dataset (30%-70%)

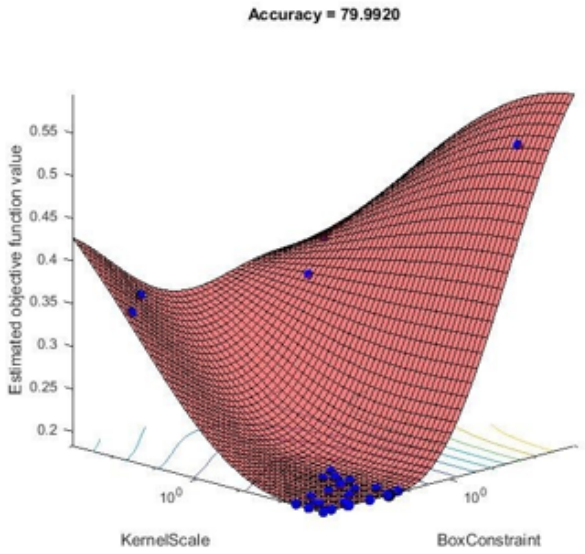

Figure 23: Accuracy in SVM with linear kernel using unnormalized dataset (80%-20%)

In Fig. 24 we can see the results from training and testing SVM using Unnormalized dataset through 50-50 training and testing scenario has given a good result, 76.1468 is the accuracy of the algorithm, in the figure we can obtain the different stages of the classification process, we can see plotting the original data, training and testing phase for both class 1 and 2.

ISSN:2222-758X
e-ISSN:2789-7362

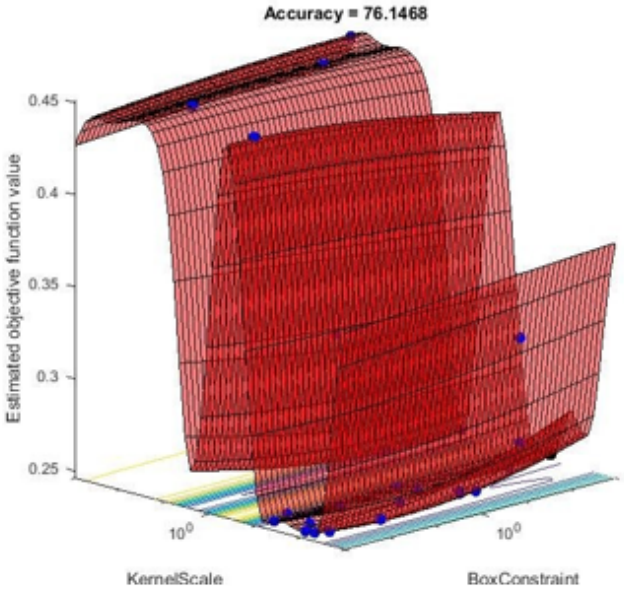

Figure 24: Accuracy in SVM with linear kernel using unnormalized dataset (50%-50%)

In Fig. 25 we will analyze the results generated from applying the SVM algorithm on the Unnormalized dataset using the linear kernel in the 30-70 % scenario, a different result is obtained from the figure. First of all the accuracy of this scenario is 74.2286 % which is smaller than that of 50-50 % and 80-20 %, this is related to the decreasing number of training data, which plays important role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number. In Fig. 26 we can see the result of applying SVM on unnormalized dataset using RBF kernel in the 80-20 % scenario. The accuracy is pointing at 92.8824 which is greater than the accuracy in the linear kernel for the same conditions, this greater accuracy shows the dominance of using RBF kernel over the linear kernel.

ISSN:2222-758X
e-ISSN:2789-7362

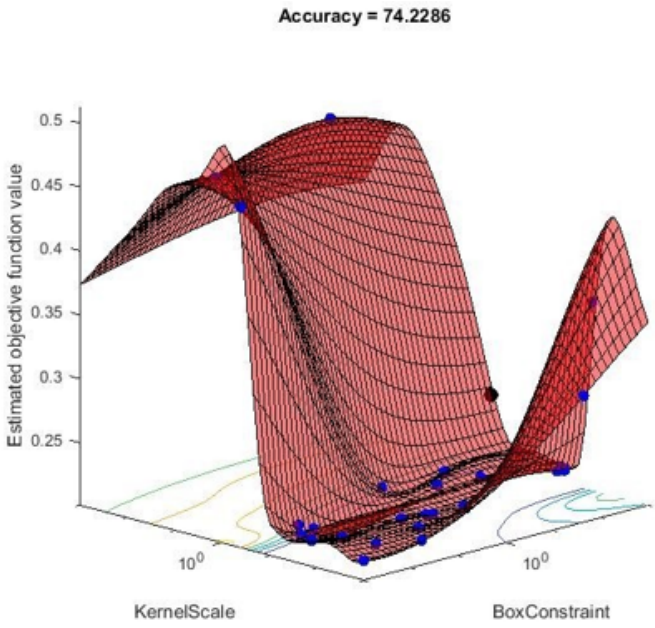

Figure 25: Accuracy in SVM with linear Kernel using unnormalized dataset (30%-70%)

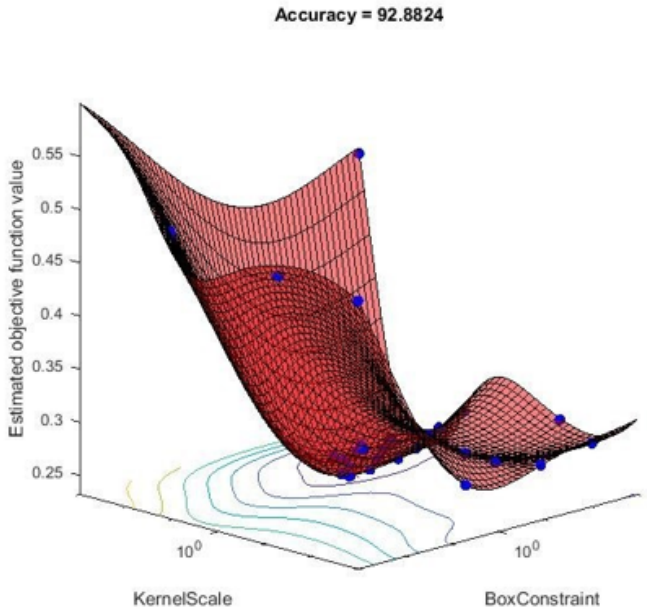

Figure 26: Accuracy in SVM with RBF kernel using unnormalized dataset (80%-20%)

In Fig. 27 we can see the results from training and testing SVM using unnormalized dataset through 50-50 training and testing scenario has given a good result, 90.9327 is the accuracy of the algorithm which is lower than the 80-20 %

ISSN:2222-758X
e-ISSN:2789-7362

scenario which is normal and expected according to the previous experiments, besides it is greater than 76.1468 which is the percentage using a linear kernel which more and more proves the superiority of using RBF kernel over that of linear.

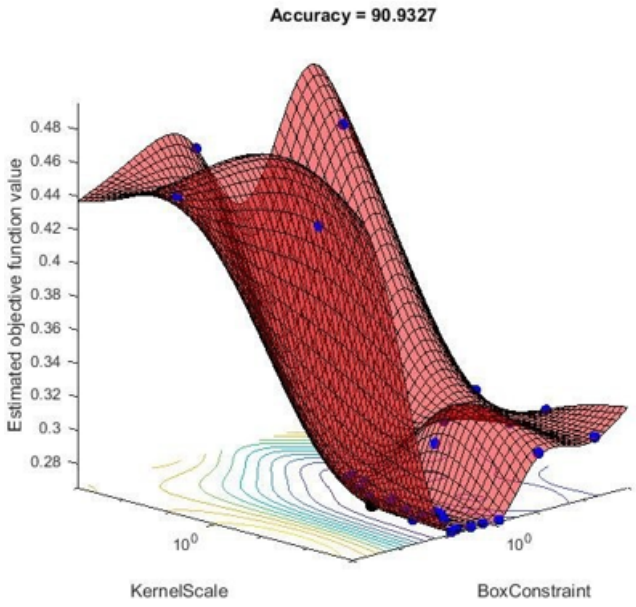

Figure 27: Accuracy in SVM with RBF kernel using unnormalized dataset (50%-50%)

In Fig. 28 we will analyze the results generated from applying SVM algorithm on unnormalized dataset using RBF kernel in the 30-70 % scenario, a different result is obtained from the figure. Finally the accuracy of this scenario is 87.1374 which is better than that of linear kernel (74.2286%) which is also smaller than that of 50-50 % and 80-20 %. In Fig. 29 we can see the result of applying SVM on the normalized dataset using the linear kernel in the 80-20 % scenario, as we can see the different colors of points refers to the different class that this point refers to, also different colors will refer to a different process which means training or testing. The accuracy in the figure is 91.0426 which is greater than that of RBF and linear in the unnormalized scenario, this shows a greater superiority by using a normalized dataset over that of unnormalized. In Fig. 30 we can see results from training and testing SVM using normalized dataset through 50-50 training and testing scenario has given a great result, 88.4329 which is smaller than that of 80-20 % which is an expected result, but the huge dominance was over the unnormalized scenarios it shows more that normalized dataset interfere positively in the classification process to produce incredible results. In Fig. 31 we will analyze the results generated from applying SVM algorithm on the normalized dataset using the linear kernel in the 30-70 % scenario, a diffeent result is obtained from the figure. First, the accuracy of this scenario is 85.3961 % which is smaller than that of 50-50 % and 80-20 %, this is related to the decreasing number of training data, which plays important role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number.

ISSN:2222-758X
e-ISSN:2789-7362

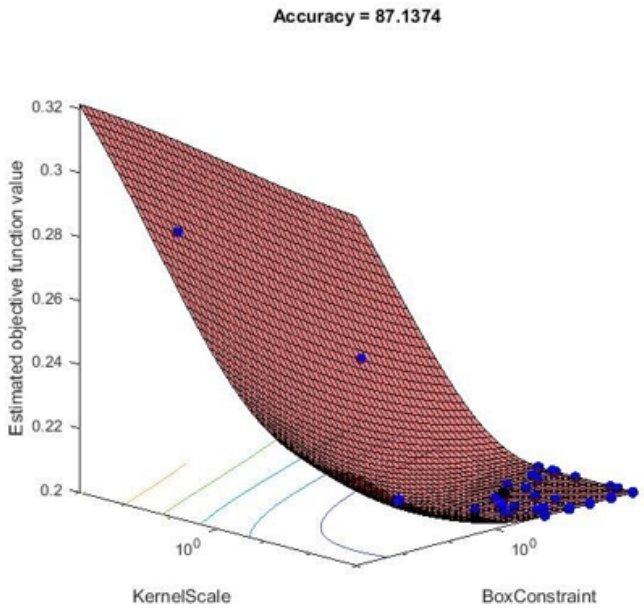

Figure 28: Accuracy in SVM with RBF Kernel using unnormalized dataset (30%-70%)

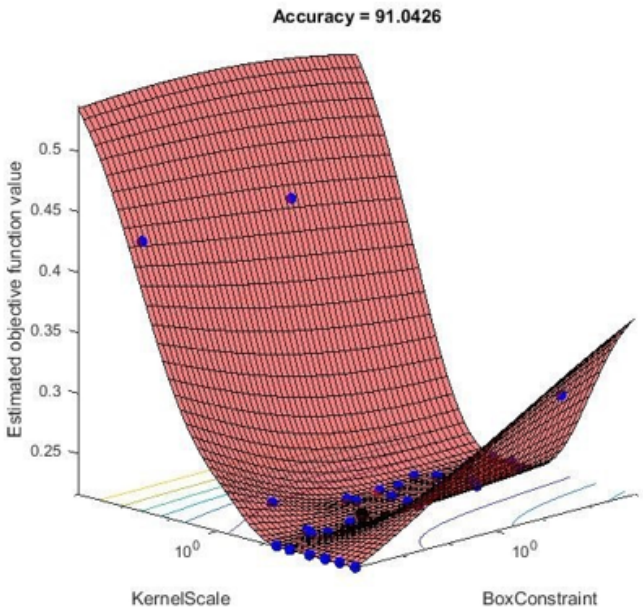

Figure 29: Accuracy in SVM with linear kernel using normalized dataset (80%-20%)

ISSN:2222-758X
e-ISSN:2789-7362

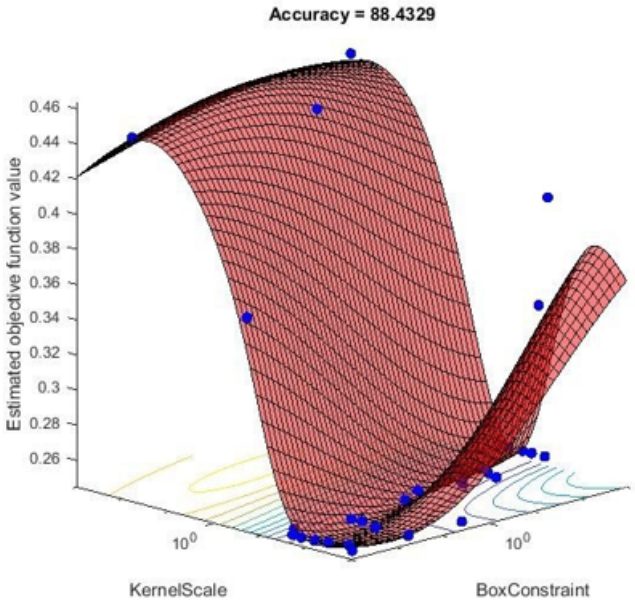

Figure 30: Accuracy in SVM with linear kernel using normalized dataset (50%-50%)

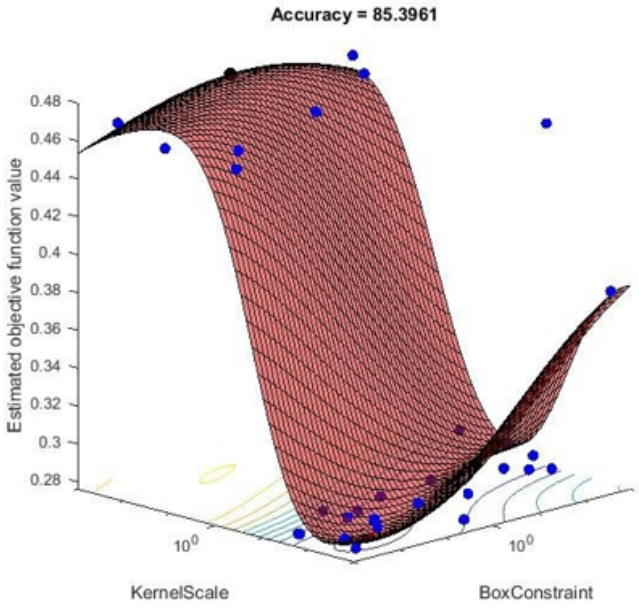

Figure 31: Accuracy in SVM with linear kernel using normalized dataset (30%-70%)

In Fig. 32 above we can see the result of applying SVM on normalized dataset using RBF kernel in the 80-20 % scenario. From the figure we can conclude that the data in the testing process are more classified as class 2. The accuracy

ISSN:2222-758X
e-ISSN:2789-7362

is pointing at 96.5870 which is greater than the accuracy in the linear kernel for the same conditions, this greater accuracy shows the dominance of using RBF kernel over the linear kernel.

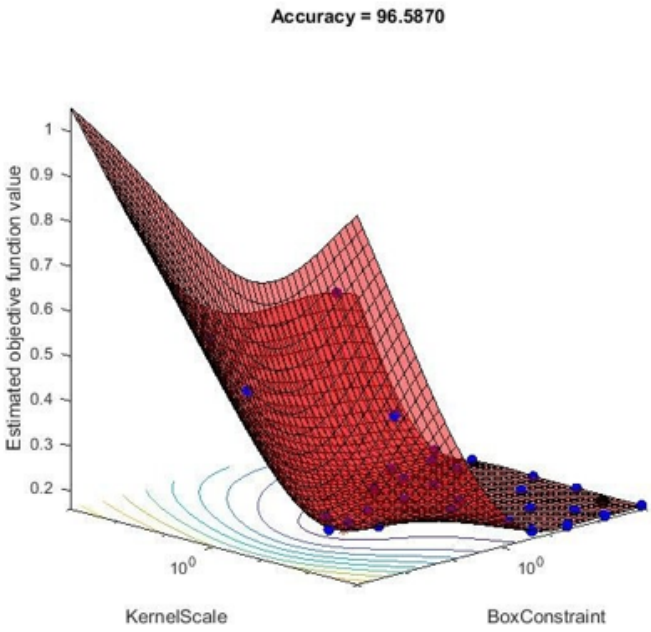

Figure 32: Accuracy in SVM with RBF kernel using normalized dataset (80%-20%)

In Fig. 33 we can see the results from training and testing SVM using normalized dataset through 50-50 training and testing scenario has given a good result, 93.8451 is the accuracy of the algorithm which is lower than the 80-20 % scenario which normal and expected according to the previous experiments, besides it is greater than that of the unnormalized dataset which more and more proves the superiority of using normalized over the unnormalized and RBF kernel over that of linear. In Fig. 34 we will analyze the results generated from applying SVM algorithm on normalized dataset using RBF kernel in the 30-70 % scenario, a different result is obtained from the figure. First of all the accuracy of this scenario is 90.3755 which is better than that of the linear kernel (85.3961%) which is also smaller than that of 50-50 % and 80-20 %, this is related to the decreasing number of training data, which plays important role in classification through feeding the model by logical patterns, this feed has been decreased by decreasing the training dataset records number.

ISSN:2222-758X
e-ISSN:2789-7362

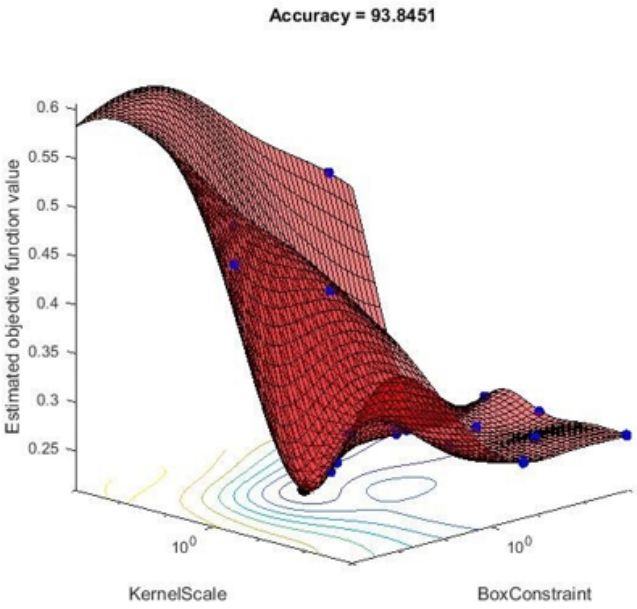

Figure 33: Accuracy in SVM with RBF kernel using normalized dataset (50%-50%)

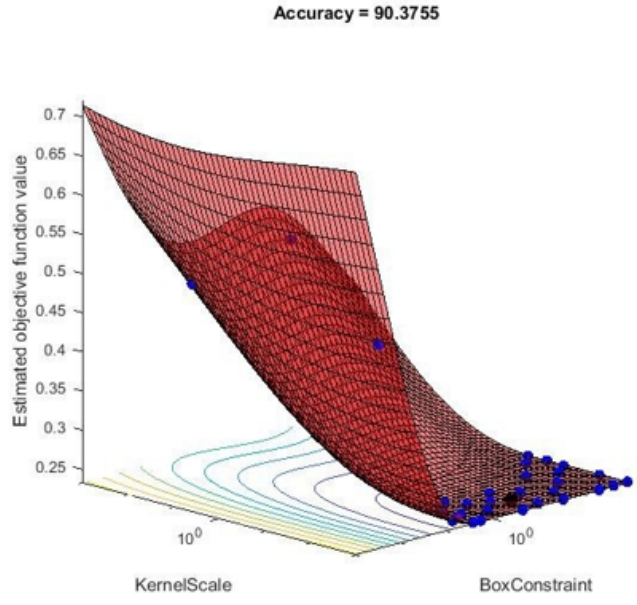

Figure 34: Accuracy in SVM with RBF kernel using normalized dataset (30%-70%)

V. COMPARISON

In this section we will show a comparative analysis for the algorithms concerning the accuracy, below is the table of comparison:

ISSN:2222-758X
e-ISSN:2789-7362

TABLE V
Comparison Between Algorithms in Our Work

	norm						Un-norm					
MLP	80-20			50-50	30-70		80-20	50-50	30-70			
	94.40			90	89.5833		86.9565	84.7068	81.7073			
	rbf			linear			rbf			linear		
	80-20	50-50	30-70	80-20	50-50	30-70	80-20	50-50	30-70	80-20	50-50	30-70
SVM	91.4810	88.4206	83.2594	85.4364	80.3547	78.3105	83.8710	76.4706	73.2394	77.4194	65.3595	62.9108
SMO	95.2139	92.0942	89.0394	93.5709	90.2952	89.0394	83.8508	76.1269	74.6606	81.4734	75.3497	72.5128

VI. CONCLUSION

In this work, we have to make a Classification system where several algorithms are combined to make a descriptive analysis between them. We used SMO, SVM and MLP algorithms along with Cleveland dataset. The dataset used was normalized, and the results that were extracted were being separated between normalized and unNormalized. The results are shown in chapter 4 we incredible and show the dominance of our approach over the other approaches, also we made a descriptive analysis in the normalized and unNormalized cases, the normalized dataset shows a very great improvement of the algorithms result with every large margin In all the scenarios. Finally, the enhancements and future work we want to do is to set a feature selection algorithm to be added to the system, also we are planning to use hybrid datasets, which means merge datasets to benefit from the variety of datasets of nature to improve the classification.

REFERENCES

[1] Y. Khourdifi and M. Bahaj, "Heart Disease Prediction and Classification Using Machine Learning Algorithms Optimized by Particle Swarm Optimization and Ant Colony Optimization" , International Journal of Intelligent Engineering and Systems, Vol. 12, No. 02, 2019.
[2] A. Masilamani, Anupriya, and N. C. S. N. Iyenger, "En Hanced Prediction of Heart Disease with Feature Subset Selection Using Genetic Algorithm" , International Journal of Engineering Science and Technology, Vol. 2, No. 10, 2010.
[3] J. Akhil, B. Deekshatulu, and P. Chandra, "Heart Disease Prediction System Using Associative Classification and Genetic Algorithm" , Artificial Intelligence, Vol. 1, No. 03, 2013.
[4] D. Vadicherla and S. S. Sonawane, "Sequential Minimal Opti- Mization in Support Vector Machine" , 2013.
[5] J. Akhil, B. Deekshatulu, and P. Chandra, "Classification of Heart Disease Using K- Nearest Neighbor and Genetic Algorithm" , Procedia Technology, Vol. 10, No.12, pp. 85-94, 2013.
[6] M. Abushariah, A. Mustafa, O. Adwan, and R. Yousef, "Auto- Matic Heart Disease Diagnosis System Based on Artificial Neural Network (ANN) and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) Approaches" , Journal of Software Engineering and Applications, Vol. 7, No. 11, pp. 1055-1064, 2014.
[7] M. Sudhakar, "Propose a Enhanced Framework for Prediction of Heart Disease" , International Journal of Engineering Research and Applications, Vol. 5, No. 04, 2015.
[8] L. Nguyen Cong, P. Meesad, and H. Unger, "A Highly Accurate Firefly Based Algorithm for Heart Disease Prediction" , Expert Systems with Applications,Vol. 06, 2015.
[9] K. Srivastava and D. Choubey, "Heart Disease Prediction Using Machine Learning and Data Mining" , Vol. 9, No. 05, pp. 212-219, 2020.
[10] N. Fitriyani, M. Syafrudin, G. Alfian, and J. Rhee, "Hdpm: An Effective Heart Disease Prediction Model for A Clinical Decision Support System" , IEEE Access, Vol. 8, No. 07, pp. 133034- 133050, 2020.
[11] A. M. Hamad alhussainy and A. D. Jasim, "Ecg Signal Classification Based on Deep Learning by Using Convolutional Neural Network (CNN) " , Iraqi Journal of Information & amp; Communications Technology, Vol. 3, pp. 12-23, Dec. , 2020.