History of Medicine, 2022, 8(2): 528-542

DOI: 10.17720/2409-5834.v8.2.2022.51

Diabetes Prediction Using Extreme Learning Machine: Application of Health Systems

G Mounika, N Prashanthi, G Vidyulatha
Assistance professor, Assistance professor, Assistance professor

Department Of CSE
Sree dattha institute of engineering and science

Abstract

Recently, machine learning has become a hot research topic. Therefore, this study investigates the interaction between software engineering and machine learning within the context of health systems. We proposed a novel framework for health informatics: the framework and methodology of software engineering for machine learning in health informatics (SEMLHI). The SEMLHI framework includes four modules (software, machine learning, machine learning algorithms, and health informatics data) that organize the tasks in the framework using a SEMLHI methodology, thereby enabling researchers and developers to analyze health informatics software from an engineering perspective and providing developers with a new road map for designing health applications with system functions and software implementations. Our novel approach sheds light on its features and allows users to study and analyze the user requirements and determine both the function of objects related to the system and the machine learning algorithms that must be applied to the dataset. Our dataset used in this research consists of real data and was originally collected from a hospital run by the Palestine government covering the last three years. The SEMLHI approach considers the Indian Diabetes dataset to perform the simulations, and extreme learning machine is outperformed as compared to state of art approaches.

Keywords: Diabetes prediction, Extreme learning machine, Health system.

1. INTRODUCTION

1.1 Overview

Human body needs energy for activation. The carbohydrates are broken down to glucose, which is the important energy source for human body cells. Insulin is needed to transport the glucose into body cells. The blood glucose is supplied with insulin and glucagon hormones produced by pancreas. Insulin hormones produced by the beta cells of the islets of Langerhans and glucagon hormones are produced by the alpha cells of the islets of Langerhans in the pancreas. When the blood glucose increases, beta cells are stimulated and insulin is given to the blood. Insulin enables blood glucose to get into the cells and this glucose is used for energy.

So blood glucose is kept in a narrow range. Diabetes is a chronic disease with the potential to cause a worldwide health care crisis. According to International Diabetes Federation 382 million people are living with diabetes across the whole world. By 2035, this will be doubled as 592 million [1]. However, early prediction of diabetes is quite challenging task for medical practitioners due to complex interdependence on various factors. Diabetes affects human organs such as kidney, eye, heart, nerves, foot etc. Data mining is a process to extract useful information from large database, as there are very large and enormous data available in hospitals and medical related diabetes. It is a multidisciplinary field of computer science which involves computational process, machine learning, statistical techniques, classification, clustering and discovering patterns.

Recently, Data mining techniques have been widely used in predicting the data like time-series [2, 3]. A number of data mining algorithms have been proposed for early prediction of disease with higher accuracy in order to save human life and reduce the treatment cost [4]. Thus, applying these algorithms to predict diabetes should be done. In our work, we used five different supervised learning methods to conduct our experiment.

1.2 Motivation

The field of health informatics (HI) aims to provide a largescale linkage among disparate ideas. Normally, a healthcare dataset is found to be incomplete and noisy; as a result, reading data from dataset linkage traditionally fails within the discipline of software engineering. Machine learning (ML) is a rapidly maturing branch of computer science since it can store data on a large scale. Many ML tools can be used to analyze data and yield knowledge that can improve the quality of work for both staff and doctors; however, for developers, there is currently no methodology that can be used. Regarding software engineering, there has been a lack of approaches to evaluating which software engineering tasks are better performed by automation and which require human involvement or human-in-the-loop approaches [1]. Big data has many challenges regarding analysis challenges for real-world big data [2], including OLAP mass data, mass data protection, mass data survey and mass data dissemination. Recently, a set of frameworks have been used to develop data analysis tools such as Win-CASE [3] and SAM [4]. The market has vast data analysis tools that can discover interesting patterns and hidden relationships to support decision makers [5]. BKMR used the R package as a statistical approach on health effects to estimate the multivariable exposure-response function [6]. Augmentor included the Python image library for augmentation [7], while for the visualization of medical treatment plans and patient data, CareVis was used [8], as it was designed for this task. Other applications require a visual interface using COQUITO [9]. For health-care data analytics, the widely known 3P tools [10] were used. Many simple applications, such as WEKA, which provided a GUI for many machine learning algorithms [11], while Apache Spark was used for the cluster computing framework [12], are powerful systems that can used in various applications for solving problems using big data and machine learning [13]. Table 1 summarizes the main tools used for big data in analytics according with respect to the task. Software engineering for machine learning applications (SEMLA) discusses the challenges, new insights, and practical ideas regarding the engineering of ML and artificial engineering (AI) [14]. NSGA-II proposed algorithms for real-world applications that include more than one objective function for enhancing performance in terms of both diversity and convergence [15]. ML algorithms in clinical genomics generally come in three main forms: supervised, unsupervised and semi-supervised [16]. Interflow system requirement analysis (ISRA) has been used to determine the system requirements [17].

1.3 Problem statement

Electronic healthcare (eHealth) frameworks have replaced traditional medical frameworks to improve mobile healthcare (mHealth) and enable patient-to-physician and patientto-patient interactions to achieve improved healthcare and quality of life (QoL) [18]. Big data and IoT have been used for improving the efficiency of m-health systems by predicting potential life-threatening conditions during the early stages [19]. Intelligent IoT eHealth solutions enable healthcare professionals to monitor health-related data continuously and provide real-time actionable insights used to support decision making [20]. Machine learning is a field of software engineering that frequently utilizes factual procedures to enable PCs to ‘‘learn’’ by using information from saved datasets. Unsupervised or information mining focuses more on exploratory information investigation and is known as learning supported by data analytics. Patient laboratory test queue management and wait time prediction are a challenging and complicated job. Because each patient might require different phase operations (tasks), such as a check-up, various tests, e.g., a sugar level test or blood test, X-rays or surgery, each task can consider different medical tests, from 0 to N, for each patient according to their condition.

2. LITERATURE SURVEY

K.Vijiya Kumar et al. [17] proposed random Forest algorithm for the Prediction of diabetes develop a system which can perform early prediction of diabetes for a patient with a higher accuracy by using Random Forest algorithm in ma- chine learning technique. The proposed model gives the best results for diabetic prediction and the result showed that the prediction system is capable of predicting the diabetes disease effectively, efficiently and most importantly, instantly.

Muhammad Azeem Sarwar et al. [18] proposed study on prediction of diabetes using machine learning algorithms in healthcare they applied six different machine learning algorithms Performance and accuracy of the applied algorithms is discussed and compared. Comparison of the different machine learning techniques used in this study reveals which algorithm is best suited for prediction of diabetes. Diabetes Prediction is becoming the area of interest for researchers in order to train the program to identify the patient are diabetic or not by applying proper classifier on the dataset. Based on previous research work, it has been observed that the classification process is not much improved. Hence a system is required as Diabetes Prediction is important area in computers, to handle the issues identified based on previous research.

Tejas N. Joshi et al. [19] presented Diabetes Prediction Using Machine Learning Techniques aims to predict diabetes via three different supervised machine learning methods including: SVM, Logistic regression, ANN. This project pro- poses an effective technique for earlier detection of the diabetes disease.

Nonso Nnamoko et al. [20] presented predicting diabetes onset: an ensemble supervised learning approach they used five widely used classifiers are employed for the ensembles and a meta-classifier is used to aggregate their outputs. The results are presented and compared with simi- lar studies that used the same dataset within the literature. It is shown that by using the proposed method, diabetes onset prediction can be done with higher accuracy.

Deeraj Shetty et al. [21] proposed diabetes disease prediction using data mining assemble Intelligent Diabetes Disease Prediction System that gives analysis of diabetes malady utilizing diabetes patients database. In this system, they propose the use of algorithms like Bayesian and KNN (K-Nearest Neighbor) to apply on diabetes patients database and analyze them by taking various attributes of diabetes for prediction of diabetes disease.

3. PROPOSED METHOD

3.1 Proposed Framework

In proposed system the combining Software Engineering and Machine Learning algorithms to improve disease prediction in health care systems and to minimize time taken to predict disease as we don’t have enough hospitals or bed to accommodate growing number of patients and we can solve this problem of predicting disease with less time by employing software and machine learning algorithms. Proposed method concept is known as SEMLHI Software Engineering with Machine Learning for Health Data).

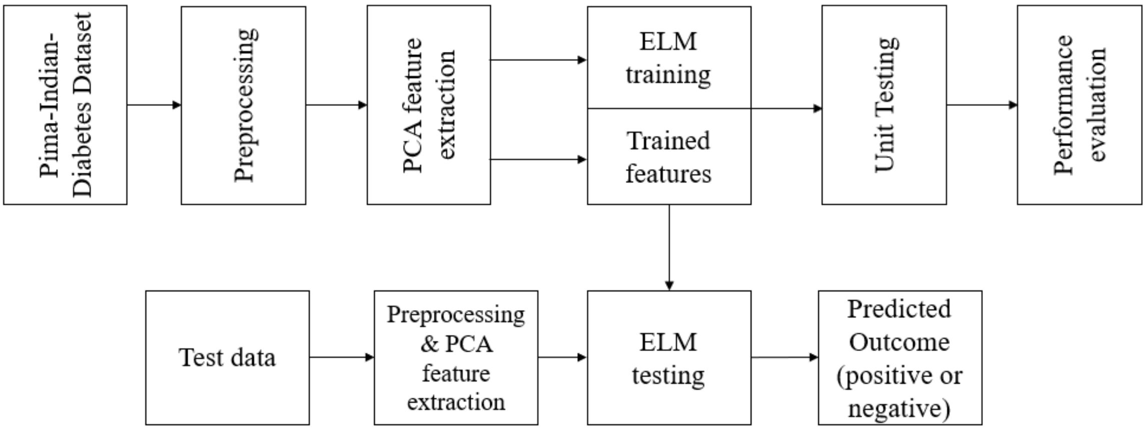


Fig. 1: Proposed framework.

Fig. 1 shows the proposed framework. Propose SEMLHI consists of 4 components.

- **Health Informatics Data:** To predict any disease we need to build Machine Learning models by using datasets and this dataset often contains missing data, null and non-numeric data and this type of data could degrade ML prediction accuracy and to overcome from this problem this work is applying PREPROCESSING on health care data to remove all missing and null values and then convert non-numeric data to numeric data by applying python SKLEARN PREPROCESSING classes. Often this dataset may contain unnecessary columns or attributes and to remove this attribute here applying dimensionality reduction algorithm called PCA. The PCA (principal component analysis) remove unnecessary attributes from dataset and maintain only important attributes necessary to make correct prediction.
- **ML Algorithms:** In this module we are using various machine learning algorithms such as Linear SVC, Multinomial Naïve Bayes, Random Forest, Logistic Regression, KNN and Extreme Learning Machine (ELM). This algorithm train itself with available datasets and then generate a train model and then this train model will be applied on new test data to perform prediction. By using above algorithms, we can make machine to learn and perform prediction without any human supports.
- **Machine Algorithm Model:** Once after building above models then we can apply new test data on this model to predict whether patient lab reports are positive or negative.
- **Software:** This module used by developers to check reliability of above modules by applying software quality check, UNIT TESTING and software verification.

In proposed work by using various size of dataset we are applying classification, clustering ore regression and to implement this concept is using Palestine Hospital dataset and this dataset not available on internet and also not publish this dataset on internet so using INDIAN DIABETES dataset. we will use this dataset to train above ML algorithms and then perform UNITTESTING to check all ML algorithms are giving accurate accuracy values.

3.2 Dataset

This data set contains 416 liver patient records and 167 non liver patient records collected from North East of Andhra Pradesh, India. The "Dataset" column is a class label used to divide groups into liver patient (liver disease) or not (no disease). This data set contains 441 male patient records and 142 female patient records. Any patient whose age exceeded 89 is listed as being of age "90".

Columns**:**

- Age of the patient
- Gender of the patient
- Total Bilirubin
- Direct Bilirubin
- Alkaline Phosphotase
- Alamine Aminotransferase
- Aspartate Aminotransferase
- Total Protiens
- Albumin
- Albumin and Globulin Ratio

Dataset: field used to split the data into two sets (patient with liver disease, or no disease)

Get the Dataset: To create a machine learning model, the first thing we required is a dataset as a machine learning model completely works on data. The collected data for a particular problem in a proper format is known as the dataset.

Dataset may be of different formats for different purposes, such as, if we want to create a machine learning model for business purpose, then dataset will be different with the dataset required for a liver patient. So each dataset is different from another dataset. To use the dataset in our code, we usually put it into a CSV file. However, sometimes, we may also need to use an HTML or xlsx file. sCSV stands for "Comma-Separated Values" files; it is a file format which allows us to save the tabular data, such as spreadsheets. It is useful for huge datasets and can use these datasets in programs.

3.3 Preprocessing

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data preprocessing task.

3.4 Splitting the Dataset

In machine learning data preprocessing, we divide our dataset into a training set and test set. This is one of the crucial steps of data preprocessing as by doing this, we can enhance the performance of our machine learning model. Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models. If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:

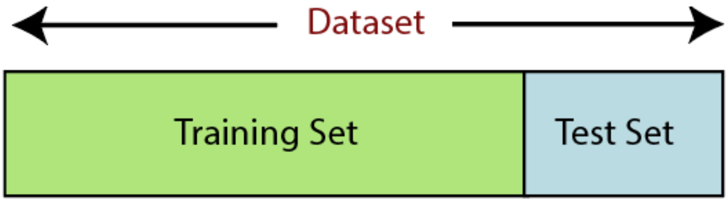


Fig. 2: Splitting the dataset.

3.5 PCA feature reduction

The Principal Component Analysis is a popular unsupervised learning technique for reducing the dimensionality of data. It increases interpretability yet, at the same time, it minimizes information loss. It helps to find the most significant features in a dataset and makes the data easy for plotting in 2D and 3D. PCA helps in finding a sequence of linear combinations of variables.

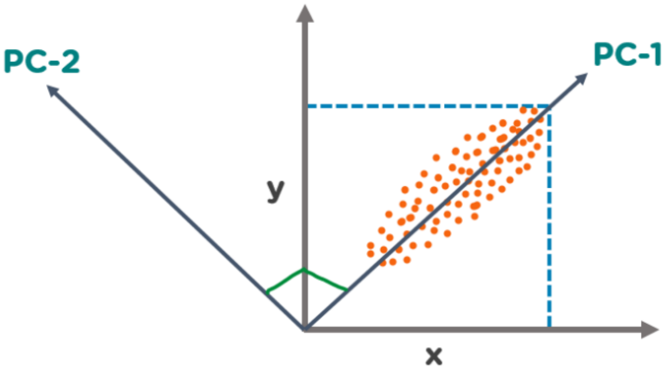


Fig. 3: PCA analysis.

In the above figure, we have several points plotted on a 2-D plane. There are two principal components. PC1 is the primary principal component that explains the maximum variance in the data. PC2 is another principal component that is orthogonal to PC1.

Fig. 4: Applications of PCA in machine learning.

- PCA is used to visualize multidimensional data.
- It is used to reduce the number of dimensions in healthcare data.
- PCA can help resize an image.
- It can be used in finance to analyze stock data and forecast returns.
- PCA helps to find patterns in the high-dimensional datasets.

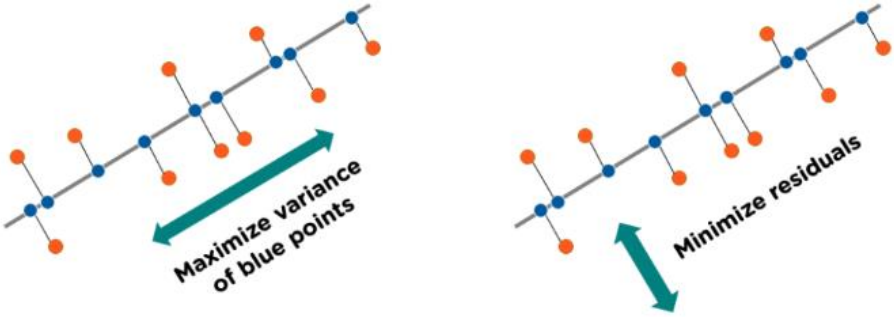


Fig. 5: PCA working.

Step 1: Normalize the data: Standardize the data before performing PCA. This will ensure that each feature has a mean = 0 and variance = 1.

$$Z = \frac{x - \mu}{\sigma}$$

Step 2: Build the covariance matrix: Construct a square matrix to express the correlation between two or more features in a multidimensional dataset.

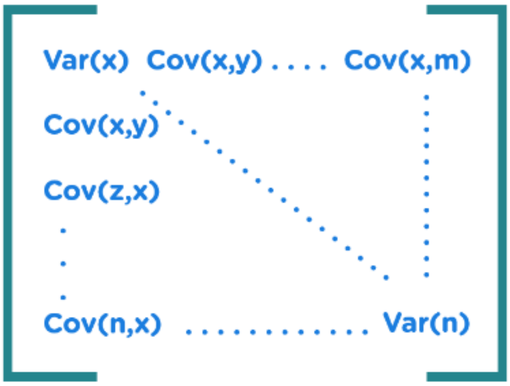


Fig. 6: Covariance matrix formulation.

Step 3: Find the Eigenvectors and Eigenvalues: Calculate the eigenvectors/unit vectors and eigenvalues. Eigenvalues are scalars by which we multiply the eigenvector of the covariance matrix.

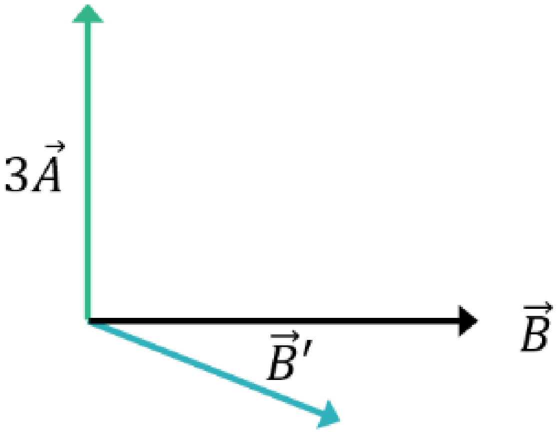


Fig. 7: PCA dimension reduction.

Step 4: Sort the eigenvectors in highest to lowest order and select the number of principal components.

3.6 ELM Prediction

ELM is a kind of advanced neural network, consists of three layers such as input layer, hidden layer (number of neurons) and an output layer. The input layer captures the input variable, hidden layers make a linear relationship among the variables and the output layer presents the predicted value. The following principle that differentiates ELM from other traditional NN is based on the parameters of the feed-forward network, inputs weights and biases provided to the hidden layer. In ELM, the bias of the hidden layer and input weight are randomly generated and the output is calculated by the Moore–Penrose generalized inverse of the hidden layer output matrix. The randomly chosen input weight and hidden layer biases learn the training samples with minimum error. After randomly choosing the input weights and the hidden layer biases, SLFNs can be simply considered as a linear system. The main advantage of ELM is, its structure does not depend on network parameters which produce stability. Hence it is useful for classification, regression, and clustering.

Therefore, we adopted ELM as a classification model in predicting the software quality. Figure 2 shows the architecture of ELM with four input layers, ten hidden layers, and three output layers. The process of training and testing the ELM contains a network with two vectors of input vector and target output vector. The ELM prediction model used for classification form a function FðxÞ, considering ða1; a2; :::; anÞ attributes as input vector and ðzjÞ as output classification label target vector. In this proposed work, the attributes ISC, BICM, SCCR, and SCCP are considered as input metrics to predict software quality in terms of Maintainability, Independency and Portability. The classification labels zj vector denotes this software quality factors. Each of the input vector attributes describes the software components in the JavaBean software system. The objective consider is, to design an optimal input weight in SLFN with minimum error rate. Therefore, the evaluation of function FðxÞ is performed on the given dataset (S), where S is grouped into two parts of the training set ðSTrainÞ and testing set ðSTestÞ. The learning process uses a back propagation algorithm for SLFN. Finally, the output classification label is evaluated on STest. Consider L training samples be aj;zj , where aj ¼ aj1; aj2; ...; ajm T 2 Rm aj ¼ aj1; aj2; :::; ajm T 2 Rm indicates the input vector of jth samples with m-dimensional attributes and zj ¼ zj1;zj2; ...zjn T 2 Rn indicates the jth output (target) vector with n-dimension. In ELM, the bias to the hidden layer and input weights are generated randomly instead of tuning the network parameter. Therefore, the nonlinear system are transformed into linear system. The output function of SLFN are defined in mathematically as follows,

$$F(x) = \sum_{j=1}^{N} \beta_j G\left(\alpha_j x_j + c_j\right) = o_j \quad j = 1, 2, 3, \ldots, L \qquad (1)$$

where oj denotes the jth sample output value, bj ¼ ðbj1; bj2; ...; bjmÞ bj ¼ ðbj1; bj2; ...; bjmÞ denotes the output weight vector connection between the jth hidden neurons to the output neurons, cj 2 R denotes the randomly assigned bias vector to the hidden neurons and the output neurons, and aj 2 R denotes the random input weight vector assigned between the input neurons and hidden neurons. The whole representation of Gðajxj þ cjÞ indicates the output of jth hidden neurons with xj input samples.

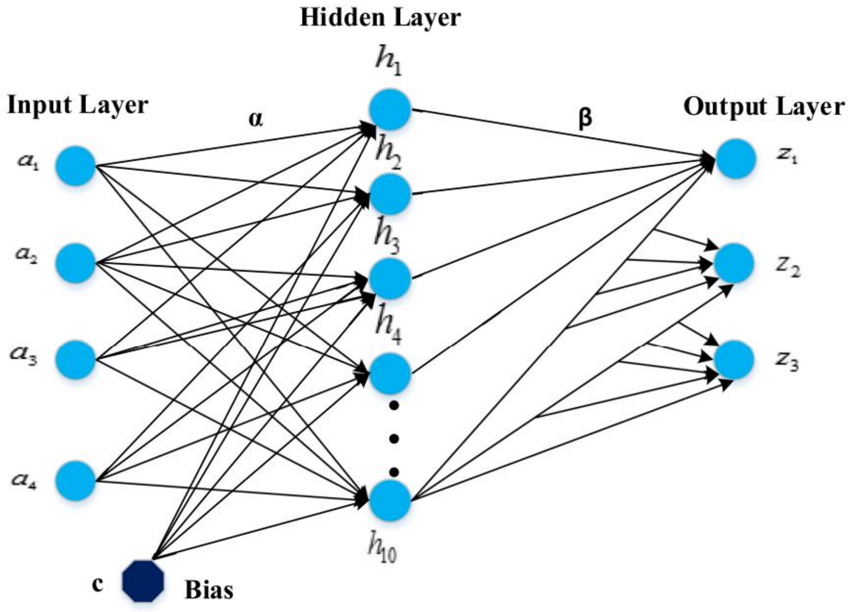


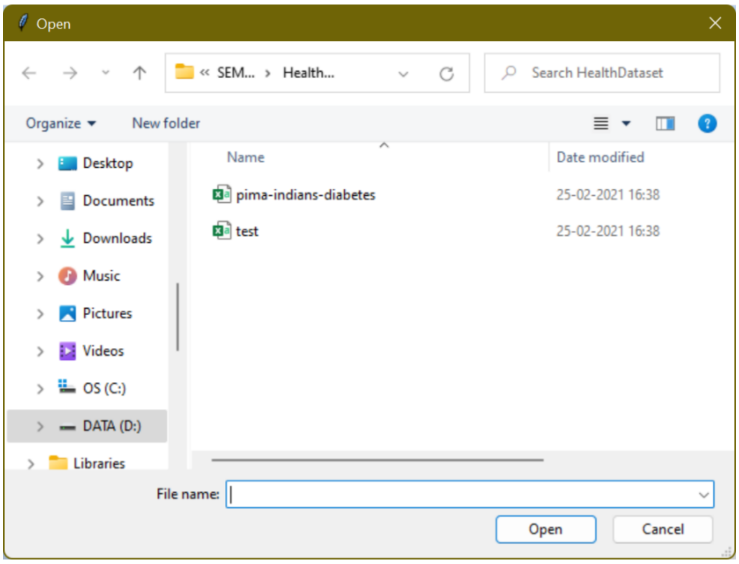
Fig. 8: The architecture of the extreme learning machine.

4. RESULTS AND DISCUSSION

This proposed implementation concept consists of 4 components

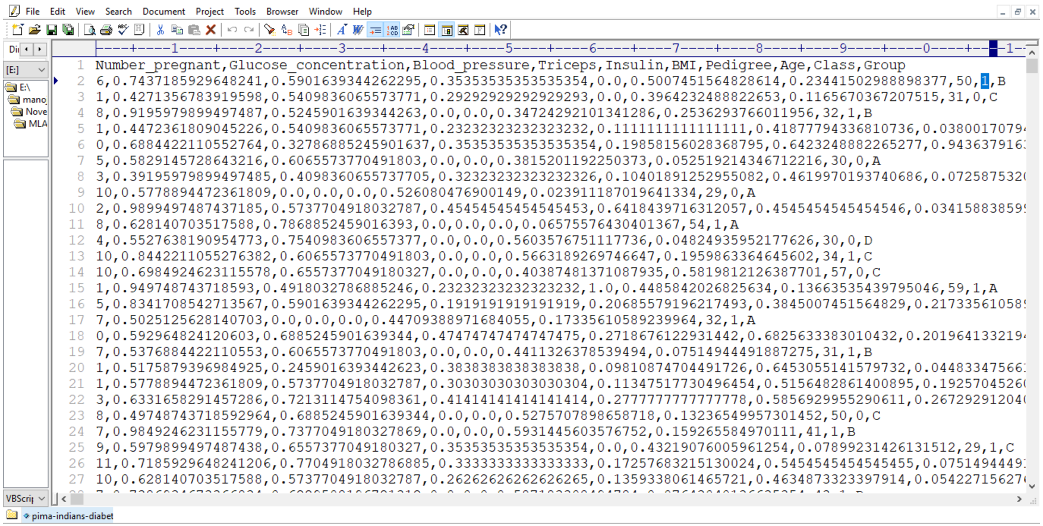
1) Health Informatics Data: To predict any disease we need to build Machine Learning models by using datasets and this dataset often contains missing data, null and non-numeric data and this type of data could degrade ML prediction accuracy and to overcome from this problem author is applying PREPROCESSING on health care data to remove all missing and null values and then convert non-numeric data to numeric data by applying python SKLEARN PREPROCESSING classes. Often this dataset may contain unnecessary columns or attributes and to remove these attributes author applying dimensionality reduction algorithm called PCA. PCA (principal component analysis) removes unnecessary attributes from dataset and maintains only important attributes necessary to make correct prediction.
2) ML Algorithms: This module uses various machine learning algorithms such as Linear SVC, Multinomial Naïve Bayes, Random Forest, Logistic Regression, KNN and extreme learning machine (ELM). This algorithm trains itself with available datasets and then generates a train model and then this train model will be applied on new test data to perform prediction. By using the above algorithms, we can make machines learn and perform predictions without any human supports.
3) Machine Algorithm Model: Once after building the above models then we can apply new test data on this model to predict whether patient lab reports are positive or negative.
4) Software: This module is used by developers to check reliability of above modules by applying software quality check, UNIT TESTING and software verification.

In above screen click on 'Upload Healthcare Dataset' button to upload dataset



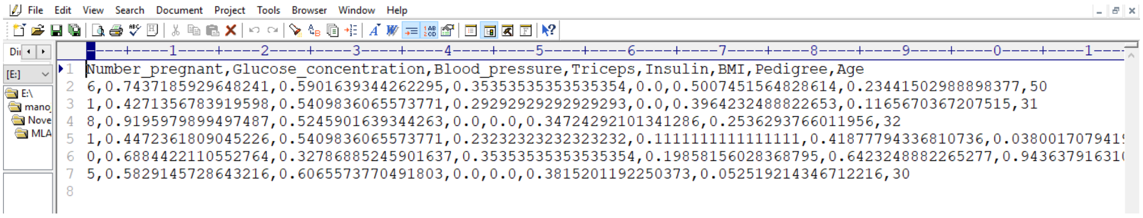
In above screen selecting and uploading diabetes dataset and then click on 'Open' button to load dataset.

In below screen of dataset we can see there is last label 'Class' which contains values as 0 and 1 where 0 means that lab values contains no disease and 1 means that lab values contains disease



```
Number_pregnant,Glucose_concentration,Blood_pressure,Triceps,Insulin,BMI,Pedigree,Age,Class,Group
6,0.7437185929648241,0.5901639344262295,0.35353535353535354,0.0,0.5007451564828614,0.23441502988898377,50,1,B
1,0.4271356783919598,0.5409836065573771,0.29292929292929293,0.0,0.3964232488822653,0.1165670367207515,31,0,C
8,0.9195979899497487,0.5245901639344263,0.0,0.0,0.34724292101341286,0.2536293766011956,32,1,B
1,0.4472361809045226,0.5409836065573771,0.23232323232323232,0.1111111111111111,0.41877794336810736,0.0380017079
0,0.6884422110552764,0.32786885245901637,0.35353535353535354,0.19858156028368795,0.6423248882265277,0.943637916
5,0.5829145728643216,0.6065573770491803,0.0,0.0,0.3815201192250373,0.05251921434671216,30,0,A
3,0.39195979899497485,0.4098360655737705,0.32323232323232326,0.10401891252955082,0.4619970193740686,0.072587532
10,0.5778894472361809,0.0,0.0,0.0,0.526080476900149,0.023911187019641334,29,0,A
2,0.9899497487437185,0.5737704918032787,0.45454545454545453,0.6418439716312057,0.4545454545454546,0.034158838
8,0.628140703517588,0.7868852459016393,0.0,0.0,0.0,0.06575576430401367,54,1,A
4,0.5527638190954773,0.7540983606557377,0.0,0.0,0.5603576751117736,0.04824935952177626,30,0,D
10,0.8442211055276382,0.6065573770491803,0.0,0.0,0.5663189269746647,0.1959863364645602,34,1,C
10,0.6984924623115578,0.6557377049180327,0.0,0.0,0.4038748137108 7935,0.5819812126387701,57,0,C
1,0.949748743718593,0.4918032786885246,0.23232323232323232,1.0,0.4485842026825634,0.13663535439795046,59,1,A
5,0.8341708542713567,0.5901639344262295,0.1919191919191919,0.20685579196217493,0.3845007451564829,0.217335610
7,0.5025125628140703,0.0,0.0,0.0,0.44709388971684055,0.17335610589239964,32,1,A
0,0.592964824120603,0.6885245901639344,0.47474747474747475,0.2718676122931442,0.6825633383010432,0.201964133219
7,0.5376884422110553,0.6065573770491803,0.0,0.0,0.4411326378539494,0.07514944491887275,31,1,B
1,0.5175879396984925,0.2459016393442623,0.38383838383838383,0.09810874704491726,0.6453055141579732,0.0448334756
1,0.5778894472361809,0.5737704918032787,0.30303030303030304,0.11347517730496454,0.5156482861400895,0.192570452
3,0.6331658291457286,0.7213114754098361,0.41414141414141414,0.2777777777777778,0.5856929955290611,0.267292912
8,0.49748743718592964,0.6885245901639344,0.0,0.0,0.5275707898658718,0.13236549957301452,50,0,C
7,0.9849246231155779,0.7377049180327869,0.0,0.0,0.5931445603576752,0.159265584970111,41,1,B
9,0.5979899497487438,0.6557377049180327,0.35353535353535354,0.0,0.43219076005961254,0.07899231426131512,29,1,C
11,0.7185929648241206,0.7704918032786885,0.3333333333333333,0.17257683215130024,0.5454545454545455,0.075149444
10,0.628140703517588,0.5737704918032787,0.26262626262626265,0.1359338061465721,0.4634873323397914,0.05422715627
```

In above dataset screen all values are the lab report values and 'Class' value contains 0 or 1 and ML algorithm will train with above lab report values and Class Value and then generate a model. Generated train model we will apply on below test data to predict class label. In below test dataset we can see there is no Class label column and ML will predict Class label by using alone lab values. See below test values



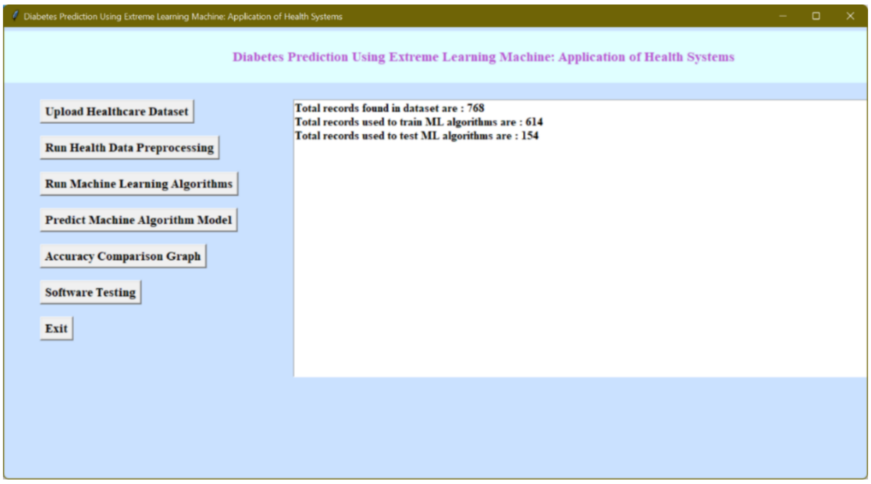
```
Number_pregnant,Glucose_concentration,Blood_pressure,Triceps,Insulin,BMI,Pedigree,Age
6,0.7437185929648241,0.5901639344262295,0.35353535353535354,0.0,0.5007451564828614,0.23441502988898377,50
1,0.4271356783919598,0.5409836065573771,0.29292929292929293,0.0,0.3964232488822653,0.1165670367207515,31
8,0.9195979899497487,0.5245901639344263,0.0,0.0,0.34724292101341286,0.2536293766011956,32
1,0.4472361809045226,0.5409836065573771,0.23232323232323232,0.1111111111111111,0.41877794336810736,0.038001707941
0,0.6884422110552764,0.32786885245901637,0.35353535353535354,0.19858156028368795,0.6423248882265277,0.94363791631
5,0.5829145728643216,0.6065573770491803,0.0,0.0,0.3815201192250373,0.05251921434671216,30
```

In above test lab report values there is no Class label. Now go back to output screen and then click on 'Run Health Data Preprocessing' button to remove missing values and then apply PCA dimensionality algorithm to get below graph

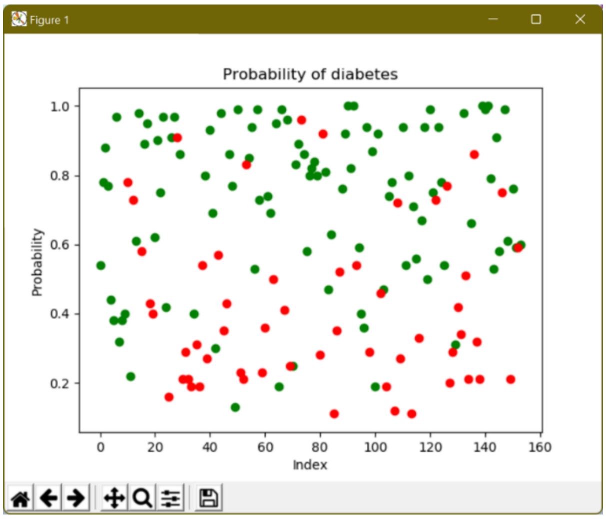
	Number_preg	Glucose_conc	Blood_pressu	Triceps	Insulin	BMI	Pedigree	Age	Class
	1.00								
	0.13	1.00	0.15	0.06	0.33	0.22	0.14	0.26	0.47
	0.14	0.15	1.00	0.21	0.09	0.28	0.04	0.24	0.07
	-0.08	0.06	0.21	1.00	0.44	0.39	0.18	-0.11	0.07
Insulin	-0.07	0.33	0.09	0.44	1.00	0.20	0.19	-0.04	0.13
BMI	0.02	0.22	0.28	0.39	0.20	1.00	0.14	0.04	0.29
Pedigree	-0.03	0.14	0.04	0.18	0.19	0.14	1.00	0.03	0.17
Age	0.54	0.26	0.24	-0.11	-0.04	0.04	0.03	1.00	0.24
Class									1.00

In above graph in top we can see names of columns and in boxes values with minus symbols are not important and only positive column values are important and ML algorithm will train only with positive values and now close above graph to get below screen.

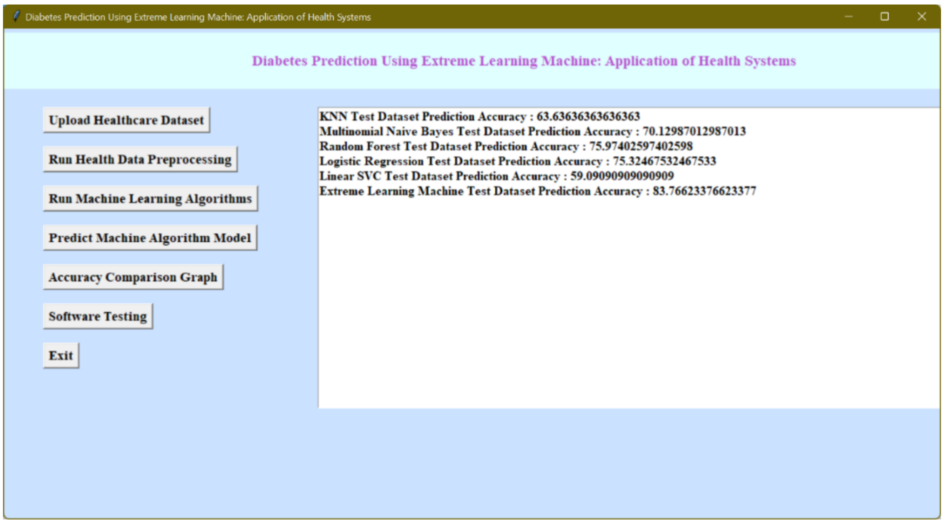
In below screen after applying pre-processing and PCA we got total records as 768 and application using 614 records to train ML algorithms and to generate model and then used 154 records to test that trained model and to calculate prediction accuracy. Now both train and test data are ready and now click on 'Run Machine Learning Algorithms' button to start training all ML algorithms on train and test data



```
Total records found in dataset are : 768
Total records used to train ML algorithms are : 614
Total records used to test ML algorithms are : 154
```

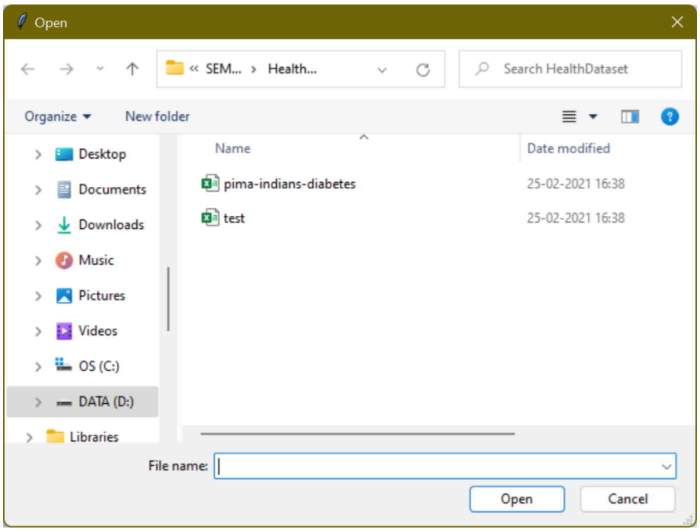


In above graph green colour dots are the records which contains no disease and red colour dots are the records which contains disease and this graph generated for all 154 test records. Now close above graph to see all ML prediction accuracy

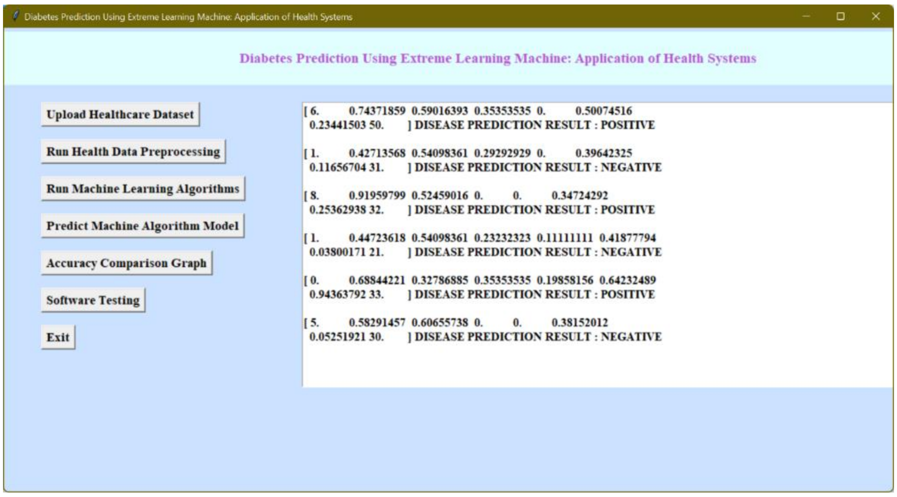


```
KNN Test Dataset Prediction Accuracy : 63.63636363636363
Multinomial Naive Bayes Test Dataset Prediction Accuracy : 70.12987012987013
Random Forest Test Dataset Prediction Accuracy : 75.97402597402598
Logistic Regression Test Dataset Prediction Accuracy : 75.32467532467533
Linear SVC Test Dataset Prediction Accuracy : 59.09090909090909
Extreme Learning Machine Test Dataset Prediction Accuracy : 83.76623376623377
```

In above screen we can see prediction accuracy of each algorithm and from all algorithms ELM is giving good prediction accuracy and now all ML algorithms are ready with trained model and now click on ‘Predict Machine Algorithm Model’ button to upload new test records and then ML will predict whether new test records contain positive or negative disease

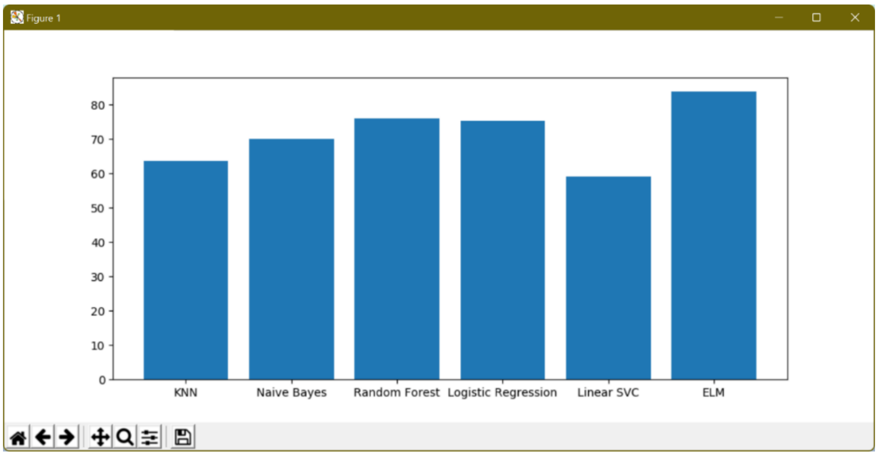


In above screen selecting and uploading ‘test.csv’ file and then click on ‘Open’ button to load test data and to get below prediction result.

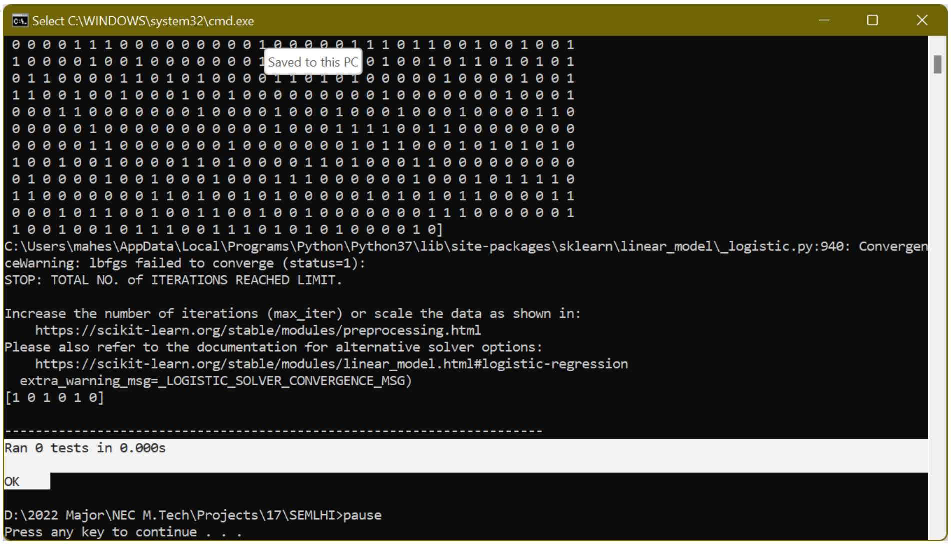


```
[ 6.        0.74371859 0.59016393 0.35353535 0.         0.50074516
  0.23441503 50.       ] DISEASE PREDICTION RESULT : POSITIVE

[ 1.        0.42713568 0.54098361 0.29292929 0.         0.39642325
  0.11656704 31.       ] DISEASE PREDICTION RESULT : NEGATIVE

[ 8.        0.91959799 0.52459016 0.         0.         0.34724292
  0.25362938 32.       ] DISEASE PREDICTION RESULT : POSITIVE

[ 1.        0.44723618 0.54098361 0.23232323 0.11111111 0.41877794
  0.03800171 21.       ] DISEASE PREDICTION RESULT : NEGATIVE

[ 0.        0.68844221 0.32786885 0.35353535 0.19858156 0.64232489
  0.94363792 33.       ] DISEASE PREDICTION RESULT : POSITIVE

[ 5.        0.58291457 0.60655738 0.         0.         0.38152012
  0.05251921 30.       ] DISEASE PREDICTION RESULT : NEGATIVE
```

In above screen for each test lab record ML predict whether disease is positive or negative. Now click on ‘Accuracy Comparison Graph’ button to get below graph



In above graph x-axis represents ML algorithm names and y-axis represents accuracy of all those algorithms and from above graph we can conclude that extension ELM is giving better accuracy and now click on ‘Software Testing’ to check all ML are working properly and to get below screen



```
 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 0 1 1 0 0 1 0 0 1 0 0 1
 1 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 Saved to this PC 0 1 0 0 1 0 1 1 0 1 0 1 0 1
 0 1 1 0 0 0 0 1 1 0 1 0 1 0 0 0 0 1 1 0 1 0 1 0 0 0 0 0 1 0 0 0 0 1 0 0 1
 1 1 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 1
 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 0
 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 1 0 0 1 1 0 0 0 0 0 0 0 0
 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0 0 1 0 1 0 1 0 1 0
 1 0 0 1 0 0 1 0 0 0 0 1 1 0 1 0 0 0 0 1 1 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0
 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 0 1 1 1 1 0
 1 1 0 0 0 0 0 0 0 1 1 0 1 0 0 1 0 1 0 0 0 0 0 1 0 1 0 1 0 1 1 0 0 0 0 1 1
 0 0 0 1 0 1 1 0 0 1 0 0 1 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1
 1 0 0 1 0 0 1 0 1 1 1 0 0 1 1 1 0 1 0 1 0 1 0 0 0 0 1 0]
C:\Users\mahes\AppData\Local\Programs\Python\Python37\lib\site-packages\sklearn\linear_model\_logistic.py:940: Convergen
ceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
[1 0 1 0 1 0]

----------------------------------------------------------------------
Ran 0 tests in 0.000s

OK

D:\2022 Major\NEC M.Tech\Projects\17\SEMLHI>pause
Press any key to continue . . .
```

In above screen if all ML algorithms are working properly then we will get tests result as OK.

5. CONCLUSION

This research introduced a new methodology, that can develop health informatics application using machine learning. Our methodology used the grounded theory methodology to develop SEMLHI framework. Developers use SEMLHI methodology to analyzing and developing software for the HI model and create a space in which SE and ML experts could work on the ML model lifecycle. Proposed framework includes a theoretical framework to support research and design activities that incorporates existing knowledge. Our work introduces a new approach form clustering and classification for ML in HI. SEMLHI methodology includes seven-phase, designing (encode data and Define outlier and cleaning up the data), implementing (Verification & Validation), maintaining and defined Workflows, structured Information, security and privacy, testing and performance, and reusing software applications. SEMLHI framework includes four modules that organize the tasks for each module, and introduce a SEMLHI Methodological that enable researchers and developer to analyze health informatics software from an engineering perspective. The ultimate goal from a SEMLHI Methodological is to define a standardized methodology for software development in the Health area and include all stages from defining the problem until developing the application and get the result with the test stage.

REFERENCES

[1] Devi, M. Renuka, and J. Maria Shyla. "Analysis of Various Data Mining Techniques to Predict Diabetes Mellitus." International Journal of Applied Engineering Research 11.1 (2016): 727-730.

[2] Berry, Michael 1., and Gordon Linoff. Data mining techniques: for marketing, sales, and customer support. John Wiley & Sons, Inc., 1997.

[3] Witten, Ian H., et al. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016.

[4] Emoto, Takuo, et al. "Characterization of gut microbiota profiles in coronary artery disease patients using data mining analysis of terminal restriction fragment length polymorphism: gut microbiota could be a diagnostic marker of coronary artery disease." Heart and vessels 32.1 (2017): 39-46.

[5] Giri, Donna, et al. "Automated diagnosis of coronary artery disease affected patients using LDA, PCA, ICA and discrete wavelet transform." Knowledge-Based Systems 37 (2013): 274-282.

[6] Fatima, Meherwar, and Maruf Pasha. "Survey of Machine Learning Algorithms for Disease Diagnostic." Journal of Intelligent Learning Systems and Applications 9.01 (2017): 1.

[7] Huang, Guang-Bin, Qin-Yu Zhu, and Chee-Kheong Siew. "Extreme learning machine: theory and applications." Neurocomputing 70.1 (2006): 489-501.

[8] Huang, Guang-Bin, Qin-Yu Zhu, and Chee-Kheong Siew. "Extreme learning machine: theory and applications." Neurocomputing 70.1 (2006): 489-501.

[9] Tiwari, Mukesh, Jan Adamowski, and Kazimierz Adamowski. "Water demand forecasting using extreme learning machines." Journal of Water and Land Development 28.1 (2016): 37-52.

[10] U-;;ar, Ay�egUI, Yakup Demir, and CUneyt GUzeli�. "A new facial expression recognition based on curvelet transform and online sequential extreme learning machine initialized with spherical clustering." Neural Computing and Applications 27.1 (2016): 131-142.

[11] Boyd, C. R.; Tolson, M. A.; Copes, W. S. (1987). "Evaluating trauma care: The TRISS method. Trauma Score and the Injury Severity Score". The Journal of trauma. 27 (4): 370 - 378. doi: I 0.1097/00005373-198704000-00005. PMID 3106646.

[12] Kologlu M., Elker D., Altun H., Sayek I. Validation of MPI and OIA II in two different groups of patients with secondary peritonitis II Hepato-Gastroenterology. - 2001. - Vol. 48, N2 37. - pp. 147-151

[13] Kologlu M., Elker D., Altun H., Sayek 1. Validation of MPI and OIA II in two different groups of patients with secondary peritonitis II Hepato-Gastroenterology. - 2001. - Vol. 48, N2 37. - pp. 147-151

[14] Laura Aurialand Rouslan A. Moro2, "Support Vector Machines (SVM) as a Technique for Solvency Analysis " .Symp. Computational Intelligence in Scheduling (SCIS 07), ASME Press, Dec. 2007, pp. 57-64, doi: 1 0.11 09/SCIS.2007.357670.

[15] Zissis, Dimitrios (October 2015). "A cloud based architecture capable of perceiving and predicting multiple vessel behaviour". Applied Soft Computing. 35: 652-661. doi:10.1016/j.asoc.2015.07.002.

[16] Graves, Alex; and Schmidhuber, JUrgen; Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks, in 1010 Bengio, Yoshua; Schuurmans, Dale; Lafferty, John; Williams, Chris K. /.; and Culotta, Aron (eds.), Advances in Neural Information Processing Systems 22 (NlPS'22), December 7th-10th, 2009, Vancouver, BC, Neural Information Processing Systems (NIPS) Foundation, 2009, pp. 545-552.

[17] K.VijiyaKumar, B.Lavanya, I.Nirmala, S.Sofia Caroline, "Random Forest Algorithm for the Prediction of Diabetes ".Proceeding of International Conference on Systems Computation Automation and Networking, 2019.

[18] Md. Faisal Faruque, Asaduzzaman, Iqbal H. Sarker, "Perfor- mance Analysis of Machine Learning Techniques to Predict Diabetes Mellitus". International Conference on Electrical, Computer and Communication Engineering (ECCE), 7-9 Feb- ruary, 2019.

[19] Tejas N. Joshi, Prof. Pramila M. Chawan, "Diabetes Prediction Using Machine Learning Techniques".Int. Journal of Engineer- ing Research and Application, Vol. 8, Issue 1, (Part -II) Janu- ary 2018, pp.-09-13

[20] Nonso Nnamoko, Abir Hussain, David England, "Predicting Diabetes Onset: an Ensemble Supervised Learning Approach ". IEEE Congress on Evolutionary Computation (CEC), 2018.

[21] Deeraj Shetty, Kishor Rit, Sohail Shaikh, Nikita Patil, "Diabe- tes Disease Prediction Using Data Mining ".International Con- ference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2017.