Application of Machine Learning Algorithms on Diabetic Retinopathy

Ridam Pal
Computer Science
& Engineering Dept.
Techno India University, West Bengal
Email:ridam101@gmail.com

Dr.Jayanta Poray
Computer Science
& Engineering Dept.
Techno India University, West Bengal
Email:jayanta.poray@gmail.com

Mainak Sen
Computer Science
& Engineering Dept.
Techno India University, West Bengal
Email:mainaksen.1988@gmail.com

***Abstract*—Diabetic Retinopathy (DR) is one of the leading cause of sight inefficiency for diabetic patients. The clinical diagnostic results and several outcome of eye testing methods reviled a set of observations that eases the decision making in the case of diabetic retinopathy for the doctor, therapist. Machine learning, a branch of artificial intelligence is applied in clinical data analytic as it can detect patterns in data, and then use these uncovered patterns to predict future data or perform some kind of decision making under uncertainty. In case of DR finding the co-relation between the depth of affection and the clinical result is very much critical, as several parameters are need to be taken into consideration for optimal decision making by the therapist. In this paper we have reviewed the performance of a set of machine learning algorithms and verify their performance for a particular DR data set.**

***Index Terms*—Machine learning algorithms in medical diagnostics, Diabetic retinopathy, Support vector machine, WEKA simulator, Python in machine learning algorithms.**

I. Introduction

Diabetic Retinopathy(DR) is a diabetic eye disease which affects light sensitive tissues like Retina, that lines the back of the eye [1]. It is a common cause of vision loss among people with diabetes and one of the leading causes of blindness among the working age adults. It actually refers to retinal changes in patients suffering with diabetes mellitus. With increase in the life expectancy of diabetes, the incidence of Diabetic Retinopathy has also increased. Diabetic Retinpathy is a leading cause of blindness. This disease is also known as diabetic eye disease. Diabetic Retinopathy affects the retinal blood vessels and causes them to bleed or leak fluids,thus distorting the vision. Diabetic Molecular Edema is a consequence of Diabetic Retinopathy causes the swelling in the are of the retina called the Macula. Thus early detection, timely treatment and follow up care is essential to protect against vision loss. Poor metabolic control is less important than duration,but is nevertheless relevant to the development and progression of Diabetic Retinopathy. Sex ratio is more in female than in males (4:3). Pregnancy may accelerate the changes in of Diabetic Retinopathy. Hypertension, when associated may accelerate the changes of diabetic Retinopathy. Other risk factors includes smoking,anaemia,obesity and hyperlipidemia. Thus to diagnose a disease it takes multiple tests and long time is used in this process. So in order to make the decision system fast data mining techniques can be used to predict the disease at faster rate with enhanced efficiency.

In this paper, we have deployed machine learning[18] techniques to predict whether an image contains signs of Diabetic Retinopathy or not. The data set was obtained from University of California, Irvine Machine Learning Repository(UCI), which contains features extracted from the Messidor image set. All features represent either a detected lesion, a descriptive feature of a anatomical part or an image-level descriptor. The next section of this paper explains the Machine Learning classification models used to conduct prediction, present experimental results and finally the conclusions.

II. Literature Review and Past works

A review showed that several studies has been conducted to in the field of medical science to predict the outcome of the disease from datasets using statistical approaches,artificial neural network and various other methods. Clinical diagnose becomes very critical due to robustness and variation of patients clinical attributes. This becomes tedious task for doctors to manually investigate and diagnose all the result and get solution for the therapy. There are several approaches to support doctors in their medical aid to deal with medical problem case to case basis. One such approach is done in diagnosing cancer. Several dataset of cancer has been retrieved to predict recurrence of various types of cancer and also to predict the survival rate of patients. Ahmad used three machine learning techniques Decision Tree, Support Vector Machine(SVM) and Artificial Neural Netwrok(ANN) to predict the recurrence of breast cancer among patients[2]. Shiv Shakti worked on various data mining techniques to predict weather a patient is suffering from benign or malignant cancer[4]. Delen worked with three techniques logistic regression,artificial neural network(ann) and decision tree to predict breast cancer survival rate [5]. Data mining techniques were used by Salha for diagnosis of heart disease[8]. Rupali and Medhekar implemented Naive Bayes classifier for predicting heart disease[9][7], Shouman and Akhil Jabbar used k-Nearest Neighbour for diagnosing heart disease[10][11] whereas Support vector Machine was used by Suruchi Pimple and Anuja Kumari to diagnose diabetes [16] [13]. Support

978-1-5090-3704-9/17/$31.00 © 2017 IEEE

vector machine was also used by Nasser H. Sweilam for diagnosis cancer disease[12]. Here we will be working on how diabetic retinopathy can be predicted from a image extracted file converted into dataset.

III. Data set information

The dataset contains 1151 instances with 19 numeric valued attributes and one numeric valued outcome variable. The raw data obtained from the UCI repository was first stored in a csv file. Since none of the columns contained missing values all of the instances could be used as a potential training example. The Machine Learning algorithms were deployed using the popular Python based Scikit-Learn libraries. Before implementing any such algorithms the data set was preprocessed using Pandas and Numpy. During the preliminary state,lesions are developed within the retina known as microaneurysm (MA)which indicates about the disease. To determine any statistical test, alpha level is determined which is also known as significance level which helps in determining the confidence interval. The alpha level lies between 0 to 1. Depending on the confidence level of alpha number of MA are detected starting from 0.5 upto 1 in this dataset.

Attribute 0 states about the binary quality assessment,1 stating about the good quality and 0 stating bad quality. Attribute 1 states about the binary result of pre-screening, where 1 indicates severe retinal abnormality and 0 its lack. Attributes 2-7 states about the results of MA detection. Each feature value stand for the number of MAs found at the confidence levels alpha = 0.5,0.7,0.8,0.9,.0. Attributes 8-15 contain the same information as 2-7 for exudates. However, as exudates are represented by a set of points rather than the number of pixels constructing the lesions, these features are normalized by dividing the number of lesions with the diameter of the ROI to compensate different image sizes. Attribute 16 states the euclidean distance of the center of the macula and the center of the optic disc to provide important information regarding the patients condition. This feature is also normalized with the diameter of the ROI. Attribte 17 states the diameter of the optic disc. Attribte 18 states the binary result of the AM/FM-based classification. Attribte 19 states class label. 1 = contains signs of DR (Accumulative label for the Messidor classes 1, 2, 3), 0 = no signs of DR.

TABLE I
Attributes information

No	Variable Name	Definition
1	Attribute 0	Binary result of image quality
2	Attribute 1	Binary result of severe retinal abnormality
3	Attribute 2	MA detection for alpha=0.5
4	Attribute 3	MA detection for alpha=0.6
5	Attribute 4	MA detection for alpha=0.7
6	Attribute 5	MA detection for alpha=0.8
7	Attribute 6	MA detection for alpha=0.9
8	Attribute 7	MA detection for alpha=1.0
9	Attribute 8-15	Attributes 8-15 contains the same info as 2-7 for exudate's.
10	Attribute 16	Euclidean distance from centre of macula to centre of optic disc
11	Attribute 17	Diameter of optic disc
12	Attribute 18	Binary result of AM/FM based classification

IV. Machine Learning Algorithm

A. Naive Bayes Classifier

The Naive Bayes classification algorithm works by reducing the number of parameters to be estimated by assuming conditional independence of the different attributes present in any instance. The parameters are then estimated using the Bayesian Maximum a Posteriori estimate or by using the Maximum Likelihood estimate. In a Gaussian Naive Bayes classifier, we also assume that the underlying distribution of output when boolean, is a Bernoulli distribution. And the attributes are assumed to follow a Gaussian distribution. The assumption of conditional independence is not always valid but it works well nonetheless in most cases. We assign that label to a new instance which has the maximum probability for the given set of attribute values.

This classifier deals with two class problem. A decision rule is implemented to find in which category the object will lie either in class W_1 or W_2. This is done with the help of the previous history of the classes and based on that two probability is generated P(W_1) and P(W_2). With the help of this probability simple decision rule is implemented which states if,

$$P(W_1) > P(W_2)$$

then W_1 class is in favour.

else if,

$$P(W_1) < P(W_2)$$

then W_2 class is in favour. Naive Bayes classifier uses the concept of apriori probability. Some feature vector is combined with this apriopri probability to make Naive Bayes classifier work in correct logical form. Let us assume the feature vector to be x. Now along with apriori probability, feature vector is used to make decision whether the object is in class W_1 or W_2. Again in this case supervised learning technique is implemented. Some objects from W_1 and some from W_2 is taken. Now the feature vector x is measured for the class W_1 and W_2. The probabilistic function or probability density measure for variable x is found for the objects belonging to class W_1 and W_2.

P(x/W_1)= Probability density function of x taking the objects from class C_1.

P(x/W_2)= Probability density function of x taking the objects from class C_2.

The decision should be based on P(x/W_1) stating that this observation for x helps to take decision in favour of W_1 and P(x/W_2) vice versa.

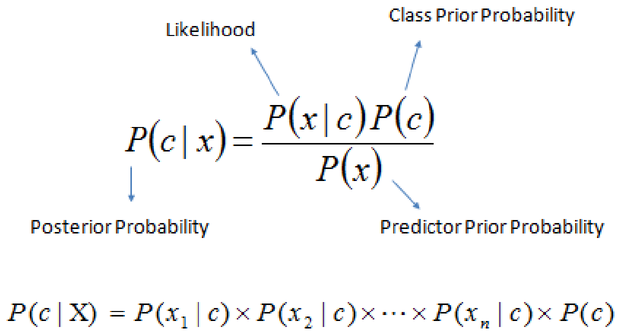

Fig. 1. Naive Bayes Classifier.

If,

$$P(x/W_1) > P(x/W_2)$$

then W_1 is in favour,
else if,

$$P(x/W_1) < P(x/W_2)$$

then W_2 is in favour.

For joint probability density function,

$$P(W_i, x) = P(W_i/x).P(x) = P(x/W_i).P(Wi) \quad (1)$$

P(W_i,x) states an object taken from W_i and has the feature vector x.

$$Now, P(W_i/x).P(x) = P(x/W_i).P(W_i) \quad (2)$$

$$or, P(W_i/x) = P(x/W_i).P(W_i)/P(x) \quad (3)$$

where, P(W_i) is apriori probability based on history,
P(x/W_i) is probability density function of x taking class W_i,
P(W_i/x) is called aposteriori probability.
So if,

$$P(W_1/x) > P(w_2/x)$$

then W_1 is in favour else if,

$$P(W_1/x) < P(W_2/x)$$

then W_2 is in favour. For this decision the help of aposteriori probability is taken which is calculated with the help of apriori probability and probability density function of x.

B. Decision Tree

Decision tree[3][17] learning is a method by which discrete valued target functions are approximated and the learned function is represented by a decision tree. Learned trees are a series if-then statements which classifies instances by sorting them down the tree from the root node to the leaf which then assigns the classification or label for than instance. Each node in a Decision Tree specifies some test on a particular attribute of the instance and each branch descending from the node is one of the possible values of that attribute. The attribute to be tested is chosen on the basis of Information Gain. The test is made on that attribute which best classifies the training examples and thus have the largest Information Gain. To quantitatively describe Information Gain we take the help of Entropy, which is the measure of purity of a sample. We then define Information Gain as the expected reduction in entropy caused by partitioning the examples based on a particular attribute.

C. K-Nearest Neighbors

K-Nearest Neighbors(KNN) is a simple classification algorithm which stores the training examples on the training stage and provides classification to a new instance based on a similarity measure during the prediction stage. It computes the distance of the new example from all other training examples based on some distance measure like Euclidean distance, Minkowski distance or Mahalanobis distance. The algorithm then looks at k closest examples and finds out the dominant class of these examples. The dominant class is assigned to as the class of the new instance. Some tweaks are often implemented rather than using the standard algorithm like providing larger weights to the closer examples. A small value of k effectively captures the fine structures of the the problem (if such structures exists). This is often needed when the size of the training set is small. A larger valued k in K-Nearest Neighbors(KNN) algorithm is however less sensitive to noise in the output class and thus provides us better estimates for the test example.The training examples are maintained in the memory so that it can be referred in future to classify the training set[11]. It is broadly classified into two techniques

1. Structureless NN techniques.
2. Structure based NN techniques.

In this technique the whole dataset is divided into two parts training dataset and test dataset. The distance between the training point and sample point is calculated and the lowest distance evaluated is called as the nearest neighbour. This technique is used when the dataset is mostly continuous but it can also deal with discrete attributes. The algorithm can be set accordingly,

1. Finding the Kth training instance closest to the unknown instance which needs to be classified. This can be done by calculating the euclidean distance among the attributes is $\sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 .... + (a_n - b_n)^2}$ where a1,a2,... belongs to known class and b1,b2,b3 .... belongs to unknown class. The closest instance will be stored from the K instances.

2. Classifying the K instances into most commonly occuring classification.

D. Support vector machine

Support vector machine(SVM) allows us to classify well beyond the training set and prevents overfitting[6]. It has the ability to use many features without much computations. If the data is linearly separable then it constructs a line or a hyperplane which maximizes the minimum margin. This maximum margin is obtained from a subset of points of the training set called the Support Vectors. When a new instance is found we

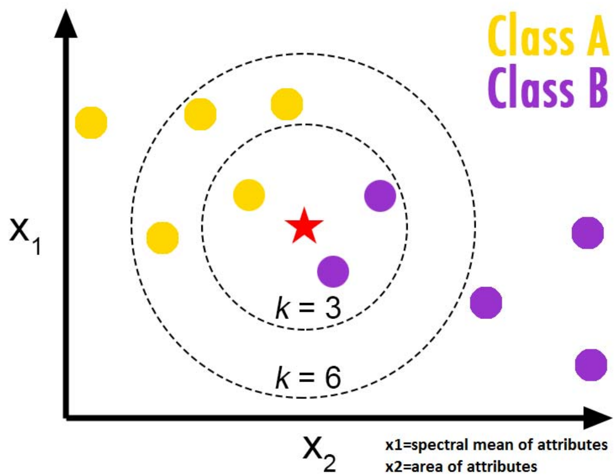

Fig. 2. K-Nearest Neighbors.

use the support vectors to decide which side of the line it may lie on. The larger the distance the support vectors have from the margin the higher is our confidence to classify the new test case. If the training data points are not linearly separable then the input space is transformed to a higher dimension. After transformation, the computation in the feature space can be costly because of its higher dimensional. This problem is then solved using the Kernel trick. The kernel function is an infinite dimension space but the similarity can be easily computed.

Support vector machine is defined with the help of linear discriminant function.It is defined with the help of two class problem stated as class C_1 and C_2 and with the help of support vector machine it is classified whether it belongs to class C_1 or class C_2.

$$g(x) = W^t x + W_0 \tag{4}$$

Here g(x) defines a linear discriminant function ,
W being the weight vector,
X being the input feature vector,
W_0 being the threshold(bias term).
The solution of the function will be,

$$g(x) = W^t x + W_0 = 0 \tag{5}$$

W will represent the orientation of the hyperplane in D-dimensional space where D being the dimension of the feature vector.W_0 will represent the position of the hyperplane in the D-dimensional space.

For every feature of x we need to compute g(x).
g(x_1)=$W^t x_1 + W_0$ > 0 then x_1 belongs to class c_1

In this case X_1 represents the positive side of the hyperplane.
g(x_2)=$W^t x_2 + W_0$ > 0 then x_1 belongs to class c_2

In this case X_2 represents the positive side of the hyperplane.
g(x)=$W^t x + W_0$=0,then the point is on the hyperplane.

This is a hyperplane which divides the D-dimensional space into two equal halves. In one half g(x) will be positive and in the other half g(x) will be negative. Thus it is inferred in following fashion that if g(x)>0; then it belongs to class C_1 or if g(x)<0; then it belongs to class C_2.

In figure 3, the straight line correctly classifies the given feature vectors. But then this straight line is not at all desirable since it gives a large bias in favour of C_2 whereas it puts penalty for C_1 since the margin for C_2 is greater than C_1. The dotted line will be preferred further since this case the classifier classifies the feature vectors in equal halves that is the line will be equally apart from the two feature vector classes.

A support vector machine tries to find a classifier which will be in this kind of form. It tries to minimise the distance of the separating boundary between the two classes by maximising the distance of the separating plane from each of the feature vector. If,

$W^t x_i + W_0$ >0 then x_i belongs to class C_1
$W^t x_i + W_0$ <0 then x_i belongs to class C_2

For every x_i we can find an equivalent y_i, y_i=+1,-1. Now if we multiply y_i with Wx_i+W_0 it will be always greater than 1. Using this concept we can go for designing the classifier.

Let us assume p to be an unknown vector which has to be classified using W and W_0, which can obtained after designing the classifier. The computation will be done in the following fashion.

W^t.p+W_0>0, then p belongs to class C_1 else it belongs to class C_2.

This equation can be modified to
$W^t x_i$+W_0>γ
where γ is referred as margin (Margin is the measure of distance of x_i from separating plane.)

If W^t.x+W_0=0 refers to a hyperplane then the distance of x form the hyperplane W.x+W_0 will be

$$W^t.x + W_0/mod(W) >= \gamma \tag{6}$$

$$or, W^t.x + W_0 >= \gamma mod(W) \tag{7}$$

With proper scaling,
$W^t x_1 + W_0$>=1 if it belongs to class C_1
$W^t x_1 + W_0$<=-1 if it belongs to class C_2.

For each x_i if y_i is multiplied during the learning process then,

$$y_i(W^t x_1 + W_0) >= 1 \tag{8}$$

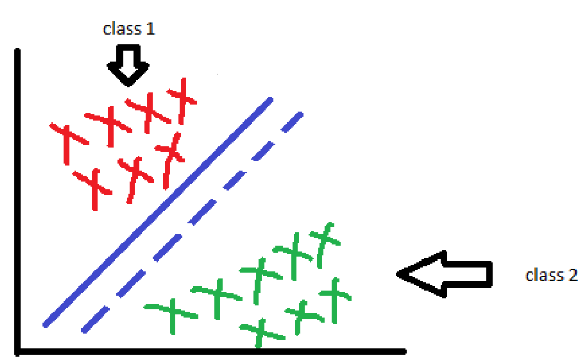

Fig. 3. Support Vector Machine.

The equation is equal to 1 if it is a support vector. The equation is greater than 1 if it is not a support vector.

A support vector machine is influenced by the position of the support vectors. It is not influenced by the feature vectors since feature vectors are not support vectors. Equation(8) states whether the x_i will belong to support vector or not.Maximising ofγ can be done by minimising mod(W) and maximising W_0. For minimising W,

ϕ(W)=W^t.W=W.W

Now,ϕ(W) needs to minimised.Since its a constraint problem, it can be converted to unconstrained problem using Lagrange's multiplier.

$$L(W, W_0) = 1/2(W.W) - \sum \alpha_i[y_i[W.x_i + W_0] - 1] \quad (9)$$

where α_i is the Lagrange's multiplier.

For optimising W_0, the derivative of equation 9 is calculated with respect to W_0.

$$L(W, W_0) = 1/2(W.W) - \sum \alpha_i y_i (W.x_i) + \sum \alpha_i y_i W_0 + \sum \alpha_i$$

$$\frac{\partial L}{\partial W_0} = \sum \alpha_i y_i = 0$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \quad (10)$$

where m is the number of feature vectors given for designing the classifier.

For optimising the W, the derivative of equation 9 is calculated with respect to W.

$$\frac{\partial L}{\partial W} = W - \sum \alpha_i y_i x_i$$

$$W - \sum \alpha_i y_i x_i = 0$$

$$\sum_{i=1}^{m} \alpha_i y_i x_i = W \quad (11)$$

Substituting the value of equation 10 and equation 11 in equation 9.

$$L(W, W_0) = 1/2(W.W) - W \sum \alpha_i y_i x_i + \sum \alpha_i$$

$$L(W, W_0) = 1/2(W.W) - W.W + \sum \alpha_i$$

$$L(W, W_0) = \sum_{i=1}^{m} \alpha_i - 1/2(W.W) \quad (12)$$

where α_i>=0. Thus, this equation have to be maximised with the following constraints. For an unknown feature vector Z, the sign specified by the equation will denote whether that unknown vector lies in class C_1 or C_2. If the value is positive then the feature vector lies in class C_1 else the feature vector lies in class C_2.

V. Result and discussion

This paper has been used to predict the occurrence Diabetic Retinopathy(DR) using data mining techniques. Every algorithm has its own limitations and strength specified to the type of application it performs.In case of our dataset Support Vector Machine worked more efficiently than other algorithms. We deployed a Support vector machine(SVM) with a 'linear' kernel and penalty parameter C of the error term was set to 10. This combination of parameters was obtained using the Grid-Search CV function of the Scikit-Learn libraries. Finally, we obtained an accuracy of 74.65% on the testing set.
The K-Nearest Neighbours(KNN)algorithm was used with k=15 and tweaked so that closer examples had higher weights and it provided an accuracy of 67.71%.
By using WEKA[19][20]simulation software the same algorithms were implemented on this dataset to predict the outcome of the disease where Support Vector machine gave a maximum accuracy of 67.54%. The comparative study of WEKA simulation kit along with Python code implemented in these algorithms depicts that Python code has higher efficiency than the WEKA simulation kit in case of every algorithms operated on this dataset which has been depicted in Figure 4.

TABLE II
Comparison of Algorithms Accuracy Using Python And Weka

No	Algorithm	Accuracy(Python)	Accuracy(WEKA)
1	Naive Bayes Classifier	65.97	56.64%
2	Decision Tree	65.45	63.51%
3	K-Nearest Neighbors	67.71	60.03%
4	Support Vector Machine	74.65	67.85%

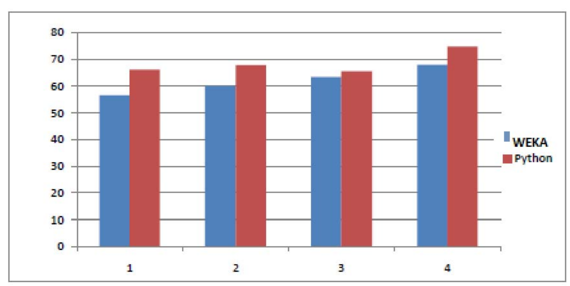

Fig. 4. Comparative study of accuracy between Python and WEKA.

VI. Conclusion

Machine Learning algorithms have been employed to predict the occurrence of Diabetic Retinopathy (DR). In this work among many possible classification algorithms,we have used four algorithms namely, Naive Bayes (NB), Decision Tree (DT), K-Nearest Neighbor (KNN) and Support Vector Machine (SVM). Each of the algorithms have certain advantages and disadvantages specific to the type of applications.

However, the results indicated the SVM had the highest accuracy among the rest, closely followed by KNN. The performance of the rest of the algorithms were average on the test data set with a few selected parameters. There were a few limitations in the approach in terms of the number of attributes used and the way the attributes were represented. Further studies can be conducted using an Image dataset with Deep Neural Networks rather than extracting features and representing them in numeric form.

REFERENCES

[1] Balint Antal, Andras Hajdu, *" An ensemble-based system for automatic screening of diabetic retinopathy, Knowledge-Based Systems 60",2014, 20-27.*

[2] Ahmad LG, Eshlaghy AT, Poorebrahimi A, Ebrahimi M, Razavi AR, *"Using Three Machine Learning Techniques for Predicting Breast Cancer Recurrence", J Health Med Inform 4:124. doi:10.4172/2157-7420.1000124,2013.*

[3] Tom Mitchell. *Machine Learning*. McGraw Hill, 1997.

[4] Shiv Shakti Shrivastava,Anjali Sant,Ramesh Prasad Aharwal, *"An Overview on Data Mining Approach on Breast Cancer data", International Journal of Advanced Computer Research, Volume-3, Number-4, Issue-13,2013.*

[5] Dursun Delen, Glenn Walker, Amit Kadam, *"Predicting breast cancer survivability: a comparison of three data mining methods", Artificial Intelligence in Medicine ,vol. 34, PP. 113-127 ,2005.*

[6] Pang-Ning Tan,Vipin Kumar,Michael Steinbach . *Introduction to Data mining.*

[7] Dhanashree S.Medhekar, Mayur P.Bote, Shruti D.Deshmmukh, *"Heart Disease Prediction using NaÃŕve Bayes", International Journal Of Enhanced Research In Science Technology and Engineering Vol.2 Issue 3,2013.*

[8] Salha M. Alzahani, Afnan Althopity, Ashwag Alghamdi, Boushra Alshehri, Suheer Aljuaid, *"An Overview of Data Mining Techniques Applied for Heart Disease Diagnosis and Prediction",Lecture Notes on Information Theory, Vol. 2, No. 4, pp. 310-315,2014. doi: 10.12720/lnit.2.4.310-315.*

[9] Rupali R.Patil, *"Heart disease prediction system using NaÃŕve Bayes and Jelinek-mercer smoothing", International Journal Advanced Research in Computer and Communication Engineering, Vol.3, Issue 5,2014.*

[10] Mai Shouman, Tim Turner, Rob Stocker, *"Applying k-Nearest Neighbour in Diagnosing Heart Disease Patients", International Journal of Information and Education Technology, Vol. 2, No. 3,2012.*

[11] M.Akhil Jabbar, B.L Deekshatulua, Priti Chandra, *"Classification of Heart Disease Using K- Nearest Neighbor and Genetic Algorithm",CoRR, volume abs/1508.02061, arXiv:1508.02061,2015.*

[12] Nasser H. Sweilam, A.A. Tharwat, N.K. Abdel Moniem, *Support vector machine for diagnosis cancer disease: A comparative study", Egyptian Informatics Journal, Volume 11, Issue 2, PP. 81-92, 2010. doi:10.1016/j.eij.2010.10.005.*

[13] Suruchi Pimple, *"Application of Support Vector Machine for Diagnosis of Diabetes: A Systematic review", IOSR Journal of Computer Engineeering , e-ISSN: 2278-0661,p-ISSN: 2278-8727, PP 13-17.*

[14] Sumit Bhatia, Praveen Prakash, G.N. Pillai, *"Svm Based Decision Support System for Heart Disease Classification with Integer-Coded Genetic Algorithm to Select Critical Features", Proceedings of the World Congress on Engineering and Computer Science, October 22 - 24, San Francisco, USA, 2008.*

[15] Mihaela GHEORGHE, *"A Support Vector Machine Approach For Developing Telemedicine Solutions: Medical Diagnosis", Network Intelligence Studies, Issue 5, pp.43-48, 2015.*

[16] V. Anuja Kumari, R.Chitra, *"Classification of Diabetes Disease Using Support Vector Machine",International Journal of Engineering Research and Applications. Vol. 3, pp. 1797-1801, ISSN: 2248-9622,2013.*

[17] M. CerÄŰak *"A Comparison of Decision Tree Classifiers for Automatic Diagnosis of Speech Recognition Errors",Computing and Informatics, vol. 29, no. 3, pp. 489-501, 2012.*

[18] Quinlan, R.J. *"C4.5: Programs for Machine Learning",Morgan Kaufmann Publishers, San Meteo, CA, USA, 1993.*

[19] David, Satish Kumar, Amr TM Saeb, and Khalid Al Rubeaan *"Comparative Analysis of Data Mining Tools and Classification Techniques using WEKA in Medical Bioinformatics",Computer Engineering and Intelligent Systems 4, no. 13: 28-38,2013.*

[20] M. Othman,T. Yau *"Comparison of different classification techniques using WEKA for breast cancer",International Conference on Biomedical Engineering, vol. 15,pp. 520-523, 2007.*