Disease Prediction using SVM & Machine Learning

S. Harshita[1] Deekshita S.[2] Divya Balaji[3] Karthiyayini J.[4]
[1,2,3]Student [4]Senior Assistant Professor
[1,2,3,4]Department of Information Science & Engineering
[1,2,3,4]New Horizon College of Engineering, Bangalore, India

Abstract— Disease prediction has been a revolutionary achievement as it reduced the risk of late stage diagnosis by a drastic margin. Several traditional methods have been implemented earlier as guessing criterion. But, the advancement in technology has provided the facility to predict diseases that cannot be detected in early stages and are not visible in traditional diagnosis methods. Using software algorithms and tools has made it convenient for patients and researchers to predict and experiment on available recorded datasets for the improvement and achieving an error free disease prediction method.

Key words: Disease Prediction, SVM, PCA, Clustering

I. INTRODUCTION

A disease or medical condition is an abnormal condition of an organism that impairs bodily functions, associated with specific symptoms and signs. Disease prediction has long been considered a critical topic. Several technical advancements in the field of Artificial intelligence and Machine Learning have already been developed to solve this type of medical care problem. Diseases like heart diseases are a major cause of death, affecting over one-third of the world's population. All over the world, 17.5 million people die of heart disease every year. By prediction and diagnosis diseases in patients, the number of deaths can be reduced caused by diseases. A promising technique of screening heart diseases is through data mining. By extracting common physical examination indicators, we can build a reliable prediction model for each patient. Data mining is non- trivial extraction of implicit, previously unknown and potential useful information about data. This in turn, is used to analyse the rich collection of data from different perspectives and deriving information. However, the performance of multiple classifiers in disease prediction is not fully understood. The major purpose of this study is to investigate the performance of different classifiers. In addition, we use various evaluation criteria to examine the performance of these classifiers with real-life datasets. Finally, we also use statistical testing to evaluate the significance of the difference in performance among the three classifiers.

A mechanized framework in therapeutic analysis would upgrade medicinal consideration and it can likewise lessen costs.

– By extracting common physical examination indicators, we can build a reliable prediction model for each patient.
– However, the performance of multiple classifiers in disease prediction is not fully understood.
– The major purpose of this study is to investigate the performance of different classifiers & data to use different evaluation criteria such as clustering for prediction.

II. PROPOSED SYSTEM

Here, we propose a cost-effective method for heart disease prediction. The proposed scheme consists of the following four mechanisms:

1) PCA Dimensionality reduction: a statistical method that converts data from high dimensional to low dimensional space.
2) Linear SVM: Support Vector Machine is a supervised machine learning algorithm to solve multi-class classification problem.
3) RBF Kernel SVM: used for a non-linearly separable problem.
4) Stratified k-fold cross verification: rearranging data to ensure each is a representation of the whole.

For the experiment purpose, the problem is divided into two. For each of the two, classifiers are run as 60/40 and 80/20 splits respectively, where 60% and 80% are used for training the classifiers and 40% and 20% are used for testing the predictions respectively. Cleveland dataset is used which has 303 instances and 14 attributes. Dimensionality reduction is applied followed by PCA and the feature set to apply the algorithm on the datasets. This type of a proposed system is beneficial, since a comparative study is involved with different classifiers, the most effective ones can be established using this system to help predict diseases with given datasets. The proposed system is flexible to work with different datasets.

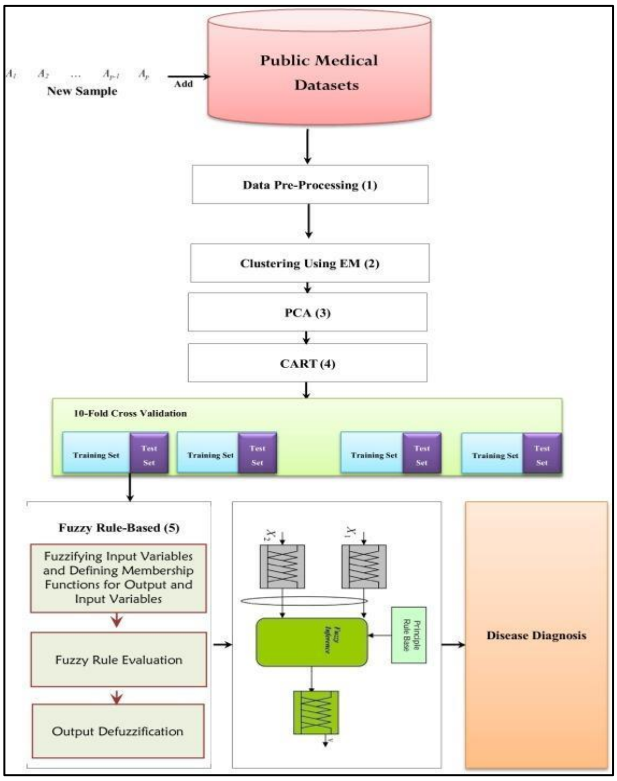


Fig. 1:

III. Implementation

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for either classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well.

A Support Vector Machine (SVM) performs classification by finding the hyper-plane that maximizes the margin between the two classes. The vectors (cases) that define the hyper-plane are the support vectors.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification, implicitly mapping their inputs into high-dimensional feature spaces. For implementing support vector machine on a dataset, we can use libraries. There are many libraries or packages available that can help us to implement SVM smoothly. We just need to call functions with parameters according to our need. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. Here, we propose a cost-effective method for heart disease prediction.

IV. Specifications

We implement the code in Python for its various advantages as, the diverse application of the Python language is a result of the combination of features which give this language an edge over others. Some of the benefits of programming in Python include.

A. *Extensive Support Libraries*

Python provides a large standard library which includes areas like internet protocols, string operations, web services tools and operating system interfaces. Many high use programming tasks have already been scripted into the standard library which reduces length of code to be written significantly.
Here, we use certain Python prerequisites:

- Numpy
- matplot-lib
- scikit-learn
- PCA

B. *Open Source and Community Development*

Python language is developed under an OSI-approved open source license, which makes it free to use and distribute, including for commercial purposes.

C. *Learning Ease & Support Available*

Python offers excellent readability and uncluttered simple-to-learn syntax which helps beginners to utilize this programming language. The code style guidelines, PEP 8, provide a set of rules to facilitate the formatting of code. Additionally, the wide base of users and active developers has resulted in a rich internet resource bank to encourage development and the continued adoption of the language.

D. *User-friendly Data Structures*

Python has built-in list and dictionary data structures which can be used to construct fast runtime data structures. Further, Python also provides the option of dynamic high-level data typing which reduces the length of support code that is needed.

E. *Productivity & Speed*

Python has clean object-oriented design, provides enhanced process control capabilities, and possesses strong integration and text processing capabilities and its own unit testing framework, all of which contribute to the increase in its speed and productivity. Python is considered a viable option for building complex multi-protocol network applications.

F. *Productivity & Speed*

Python has clean object-oriented design, provides enhanced process control capabilities, and possesses strong integration and text processing capabilities and its own unit testing framework, all of which contribute to the increase in its speed and productivity. Python is considered a viable option for building complex multi-protocol network applications.

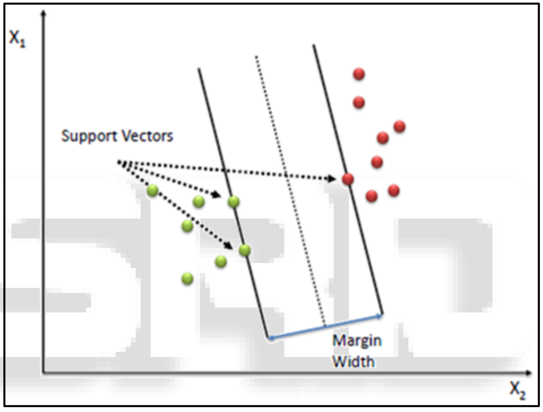
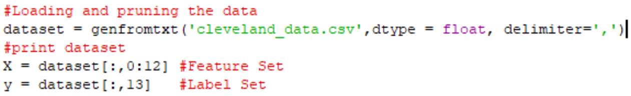


Fig. 2:

V. Datasets

To predict whether the disease exists or not, we use cross validation method which is also available in scikit-learn.
After including the libraries, we include the dataset file which is in .csv format as shown below.



```
#Loading and pruning the data
dataset = genfromtxt('cleveland_data.csv',dtype = float, delimiter=',')
#print dataset
X = dataset[:,0:12] #Feature Set
y = dataset[:,13]   #Label Set
```

The feature set 'x' represents the first twelve columns in the above figure while the thirteenth column indicates the condition of the person, that is, it represents whether the person has heart disease or not and is hence denoted by label set 'y'. After choosing the feature set and label set, we print the instances of all the values of the .csv dataset file in matrix form which shows all the label set values indicating whether the person has disease or not.

VI. Conclusion

The main idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of many variables correlated with each other, either heavily or lightly, while retaining the variation present in the dataset, up to the maximum extent. The same is done by transforming the variables to a new set of variables, which are known as the principal components (or simply, the PCs) and are orthogonal, ordered such that the retention of variation present in the original variables decreases as we move down in the order. So, in this way, the 1st principal component retains maximum variation that was present in the original components.

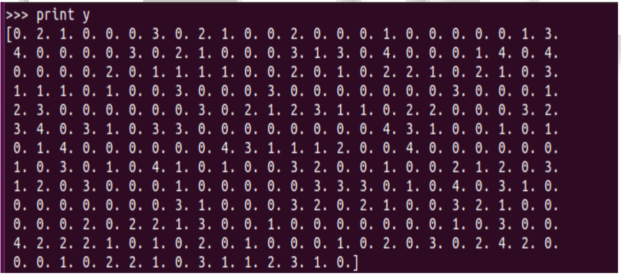
Once the features are extracted, they are stored in a new variable called x_new. The reason why extraction is needed is because, in the original datasets, there were twelve columns of data which had to be kept in record to predict whether the person has disease or not. But after performing Feature Extraction, the number of columns is diminished to two which makes the process easier to perform the disease prediction.

By running the entire Python code after splitting the data into training set and test set, we get an accumulated plotted graph which represents the vectors denoting whether the disease exists or not. The above step shows the entire pool of data which denotes the presence and absence of disease.

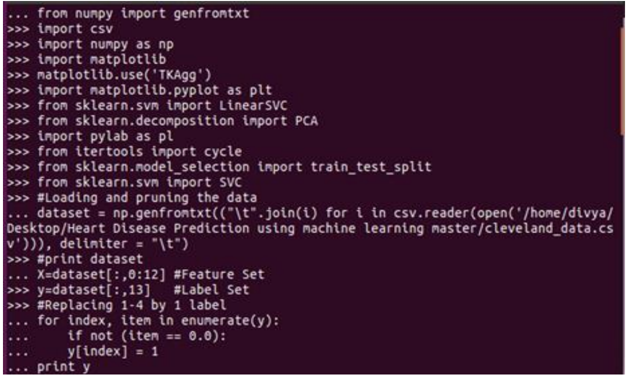
By the use of Support Vector Machine(SVM), we can predict the accuracy and the plotted graph denotes the green line which separates the non disease predicted data from the disease predicted data and also denotes the people who are prone to have a disease too, which is represented by the dots on the border line.

VII. Result

The following snapshots define the results or outputs that we will get after step by step execution of all the modules of the system.

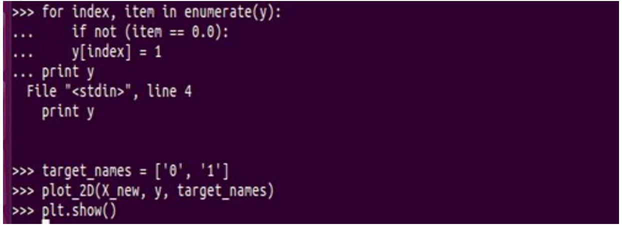


```
>>> print y
[0. 2. 1. 0. 0. 0. 3. 0. 2. 1. 0. 0. 2. 0. 0. 0. 1. 0. 0. 0. 0. 0. 1. 3.
 4. 0. 0. 0. 0. 3. 0. 2. 1. 0. 0. 0. 3. 1. 3. 0. 4. 0. 0. 0. 1. 4. 0. 4.
 0. 0. 0. 0. 2. 0. 1. 1. 1. 1. 0. 0. 2. 0. 1. 0. 2. 2. 1. 0. 2. 1. 0. 3.
 1. 1. 1. 0. 1. 0. 0. 3. 0. 0. 0. 3. 0. 0. 0. 0. 0. 0. 0. 3. 0. 0. 0. 1.
 2. 3. 0. 0. 0. 0. 0. 0. 3. 0. 2. 1. 2. 3. 1. 1. 0. 2. 2. 0. 0. 0. 3. 2.
 3. 4. 0. 3. 1. 0. 3. 3. 0. 0. 0. 0. 0. 0. 0. 0. 4. 3. 1. 0. 0. 1. 0. 1.
 0. 1. 4. 0. 0. 0. 0. 0. 0. 4. 3. 1. 1. 1. 2. 0. 0. 4. 0. 0. 0. 0. 0. 0.
 1. 0. 3. 0. 1. 0. 4. 1. 0. 1. 0. 0. 3. 2. 0. 0. 1. 0. 0. 2. 1. 2. 0. 3.
 1. 2. 0. 3. 0. 0. 0. 1. 0. 0. 0. 0. 0. 3. 3. 3. 0. 1. 0. 4. 0. 3. 1. 0.
 0. 0. 0. 0. 0. 0. 0. 3. 1. 0. 0. 0. 3. 2. 0. 2. 1. 0. 0. 3. 2. 1. 0. 0.
 0. 0. 0. 2. 0. 2. 2. 1. 3. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 1. 0. 3. 0. 0.
 4. 2. 2. 2. 1. 0. 1. 0. 2. 0. 1. 0. 0. 0. 1. 0. 2. 0. 3. 0. 2. 4. 2. 0.
 0. 0. 1. 0. 2. 2. 1. 0. 3. 1. 1. 2. 3. 1. 0.]
```



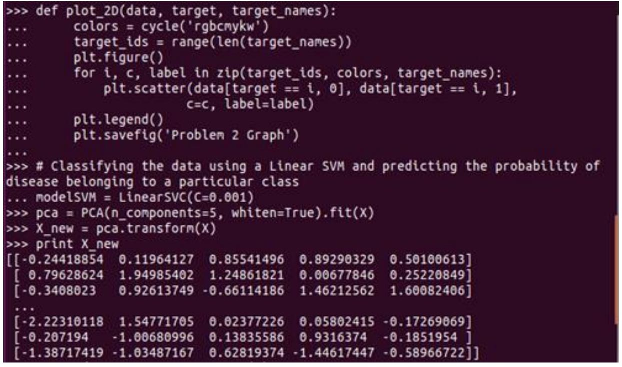
```
... from numpy import genfromtxt
>>> import csv
>>> import numpy as np
>>> import matplotlib
>>> matplotlib.use('TKAgg')
>>> import matplotlib.pyplot as plt
>>> from sklearn.svm import LinearSVC
>>> from sklearn.decomposition import PCA
>>> import pylab as pl
>>> from itertools import cycle
>>> from sklearn.model_selection import train_test_split
>>> from sklearn.svm import SVC
>>> #Loading and pruning the data
... dataset = np.genfromtxt(("\t".join(i) for i in csv.reader(open('/home/divya/
Desktop/Heart Disease Prediction using machine learning master/cleveland_data.cs
v'))), delimiter = "\t")
>>> #print dataset
... X=dataset[:,0:12] #Feature Set
>>> y=dataset[:,13]   #Label Set
>>> #Replacing 1-4 by 1 label
... for index, item in enumerate(y):
...     if not (item == 0.0):
...     y[index] = 1
... print y
```

The next step is to perform the Feature Extraction on the dataset to diminish the size of the datasets. This is done by using the transform() method. The result of this operation is shown. Further, the step is to save these data values in variable target names and then plot the graph based on these values. The plotting of the graph is done by using the Feature Extracted values stored in x_new, the 'y' feature and the target names variable.



```
>>> for index, item in enumerate(y):
...     if not (item == 0.0):
...     y[index] = 1
... print y
  File "<stdin>", line 4
    print y


>>> target_names = ['0', '1']
>>> plot_2D(X_new, y, target_names)
>>> plt.show()
```



```
>>> def plot_2D(data, target, target_names):
...       colors = cycle('rgbcmykw')
...       target_ids = range(len(target_names))
...       plt.figure()
...       for i, c, label in zip(target_ids, colors, target_names):
...           plt.scatter(data[target == i, 0], data[target == i, 1],
...                       c=c, label=label)
...       plt.legend()
...       plt.savefig('Problem 2 Graph')
...
>>> # Classifying the data using a Linear SVM and predicting the probability of
disease belonging to a particular class
... modelSVM = LinearSVC(C=0.001)
>>> pca = PCA(n_components=5, whiten=True).fit(X)
>>> X_new = pca.transform(X)
>>> print X_new
[[-0.24418854  0.11964127  0.85541496  0.89290329  0.50100613]
 [ 0.79628624  1.94985402  1.24861821  0.00677846  0.25220849]
 [-0.3408023   0.92613749 -0.66114186  1.46212562  1.60082406]
 ...
 [-2.22310118  1.54771705  0.02377226  0.05802415 -0.17269069]
 [-0.207194   -1.00680996  0.13835586  0.9316374  -0.1851954 ]
 [-1.38717419 -1.03487167  0.62819374 -1.44617447 -0.58966722]]
```

Once the graph is plotted, we can open it using the command plt.show().

The plotted graph on opening looks like the figure shown below.

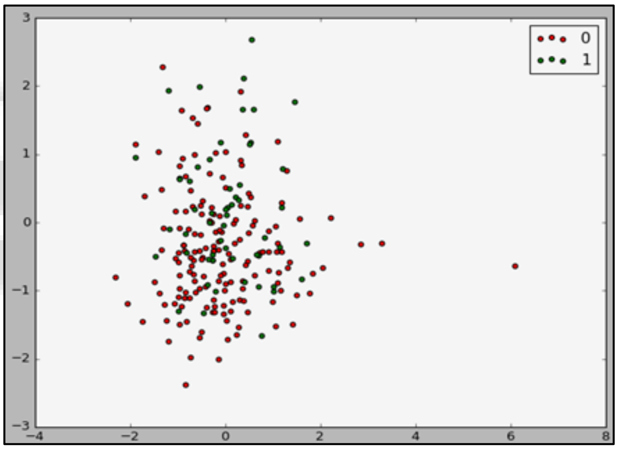


Fig. 3:

References

[1] University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.1

[2] University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.1

[3] V.A. Medical Center, Long Beach and Cleveland Clinic Foundation:Robert Detrano, M.D., Ph.D.1 72

[4] http://scikit-learn.org/0.13/index.html1 73

[5] http://archive.ics.uci.edu/ml/datasets/Heart+Disease1 74

[6] Jolliffe, I. T. (1986). Principal Component Analysis. Springer-Verlag.1 75 p. 487. doi:10.1007/b98835. ISBN 978-0-387-95442-41 76

[7] Andreas M체ller (2012). Kernel Approximations for Efficient SVMs (and other feature extraction1 77 methods)1 78

[8] Hsu, Chih-Wei; Chang, Chih-Chung; and Lin, Chih-Jen (2003). APractical Guide to Support1 79 Vector Classification (Technicalreport).

[9] http://www.cs.cmu.edu/~schneide/tut5/node42.html1 Silver, M. Sakata, T. Su, H.C. Herman, C. Dolins, S.B. & O'Shea, M.J. (2001).

[10] Case study: how to apply data mining techniques in a healthcare data warehouse. Journal of Healthcare Information Management, 15(2), 155-164. 3. Benko, A. & Wilson, B. (2003).

[11] Online decision support gives plans an edge. Managed Healthcare Executive, 13(5), 20 4. Heon Gyu Lee, Ki Yong Noh, Keun Ho Ryu, "Mining Biosignal Data: Coronary Artery Disease Diagnosis using Linear and Nonlinear Features of HRV," LNAI 4819:

[12] Cody, W.F. Kreulen, J.T. Krishna, V. & Spangler, W.S. (2002). The integration of business intelligence and knowledge management. IBM Systems Journal, 41(4), 697-713 6. Ceusters, W. (2001).

[13] Medical natural language understanding as a supporting technology for data mining in healthcare. In Medical Data Mining and Knowledge Discovery, Cios, K. J. (Ed.), PhysicaVerlag Heidelberg, New York

[14] Megalooikonomou, V. & Herskovits, E.H. (2001). Mining structure function associations in a brain image database.

[15] Chhikara, S & Sharma,P Data Mining Techniques on Medical Data for Finding Locally Frequent Diseases, I JRASET 2014,PP 396-402.

[16] Tallón-s, Antonio J., César Hervás- Martínez, JoséC. Riquelme, and Roberto Ruiz. (2013)"Feature selection to enhance a two-stage evolutionary algorithm in product unit neural networks for complex classification problems",Neurocomputing J.R. Quinlan. 1993, C4.5:

[17] Programs for Machine Learning. Morgan Kauffman Publishers, San Mateo-California. 10. J.R. Quinlan. 1995, MDL and Categorical Theories (Continued). In Machine Learning: Proceedings of the Twelfth International Conference.

[18] Eibe Frank and Ian H. Witten. 1998 Generating accurate rule sets without global optimization. In Proc 15th International Conference on Machine Learning, Madison, Wisconsin, pages 144-151.

[19] Morgan Kaufmann. Heart attack dataset from http://archive.ics.uci.edu/ml/datasets/Heart Disease 13. J. Alcalá-Fdez, L. Sánchez, S. García, M.J. del Jesus, S. Ventura, J.M. Garrell, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, J.C. Fernández, F. Herrera, 2009 KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems. Soft Computing 307-318 14. J.

[20] Alcalá-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera. KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework. Journal of Multiple-Valued Logic and Soft Computing 17:2-3 (2011) 255-287

[21] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten 2009; The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1. 16. Sharma Purushottam, Dr Kanak Saxena, Richa Sharma" Efficient Heart Disease Prediction System using Decision Tree" in IEEE International Conference on Computing Communication and Automation (ICCCA2015),May 2015 17.

[22] Sharma Purushottam, Dr Kanak Saxena, Richa Sharma" Heart Disease Prediction System Evaluation Using C4.5 Rules and Partial Tree" in Springer, Computational Intelligence in Data Mining, 2015, pp-285-294, DOI 10.1007/978-81-322-2731-1_26.

[23] Ang JC, Mirzal A, Haron H, Hamed H. Supervised, unsupervised and semi-supervised feature selection: a review on gene selection. IEEE/ACM Trans Comput Biol Bioinform 2015;PP(99).

[24] Balakrishnan S, Narayanaswamy R, Savarimuthu N, Samikannu R. SVM ranking with backward search for feature selection in type II diabetes databases. In: Systems, man and cybernetics, 2008. SMC 2008. IEEE international conference on. IEEE; 2008. p. 2628–33.

[25] Nagpal, D. Gaur ModifiedFAST: a new optimal feature subset selection algorithm J Inform Commun Convergence Eng, 13 (2) (2015), pp. 113-122 CrossRef[14]R. Battiti Using mutual information for selecting features in supervised neural net learning IEEE Trans Neural Networks, 5 (4) (1994), pp. 537-550