Disease Prediction Using Machine Learning (Heart, Diabetes, Breast Cancer, Kidney)

Mr. R.S. Bodare, Mr. P.N .Ghorpade, Mr. V.S., Pisal, Ms. V.M. Ghadge,
Prof. P.N. Shendage.

1. *Mr. V. S. Pisal, Computer Department, COE Phaltan, Maharashtra, India.*
2. *Ms. V. M. Ghadge, Computer Department, COE Phaltan, Maharashtra, India.*
3. *Mr. P. N. Ghorpade, Computer Department, COE Phaltan, Maharashtra, India.*
4. *Mr. R. S. Bodare, Computer Department, COE Phaltan, Maharashtra, India.*
5. *Prof. P.N. Shendage, Computer Department, COE Phaltan, Maharashtra, India.*

ABSTRACT

Now- a -days people are facing so many diseases due to environmental changes and wrong life style [1]. Due to this it has become crucial task to detect that disease at early stage to prevent future loss. To do an accurate prediction within limited time using proposed data has become a challenge to the doctors. Data mining is very important task. In every field data is growing exponentially so that medical field is also not an exception. With the help of data collected from medical field, data preprocessing and machine learning, deep learning algorithms we can find out the hidden patterns. We represented a general disease prediction model to predict diseases like diabetes, breast cancer, heart attack, kidney and liver disease [1]. For these disease forecasts we used a pruning tree, K-Nearest Neighbor (KNN), retrieval machine and vector support (SVM) machine. Accuracy of breast cancer model is 97.36% using KNN which is a very popular algorithm with a noticeable efficiency. The accuracy of liver disease is 74.2% using logistic regression which is more than decision tree and KNN. Accuracy of heart disease is 86.9% using KNN while accuracy of diabetes is 81.1% using SVM.

Keyword: - *Machine learning, SVM, KNN, data processing, TP, TN, FP, FN, DL and ML.*

Introduction

Artificial Intelligence has occupied each and every part of life due to its usage. AI plays a key role in the decision-making process and in prediction [1]. Machine learning is the science of teaching machines to teach them how to learn by themselves. Machine learning is superset of deep learning, DL mimics the human mind. Supervised learning, unsupervised learning, reinforcement learning are a variety of machine learning algorithms [1]. Logistic Regression, SVM, KNN, Decision Tree, Naïve Bayes become supervised machine learning algorithms. K-means clustering, association are unsupervised machine learning algorithms. Medical data is growing very rapidly that it has become the need to use that data efficiently and wisely [1]. To use the data efficiently is crucial task but before using it the data should be preprocessed. Data is converted into understandable format. Health care industry is developing a lot of data like clinical evaluation, medical tests, doctors meet up, medicines history of any patient. In real world, data is impure. Sometimes some information may not be present in any patient's history. Or sometimes someone has filled wrong information, because of that prediction may not be correct one. This may lead to false treatment and sometimes it may lead to severe problem. Raw data cannot be used directly and must be processed

first. Data is random, noisy and unambiguous. The datasets are downloaded from Kaggle and UCI [2, 5]. For this project we have extensively used scikit Learn which is a module that is built on top of scipy library in python version 2 onwards [3]. Data sets can have missing values and that can impact the working efficiency of machine learning algorithm. So to handle missing values we can use mean imputation, median imputation, mode imputation, random sample imputation and so on. Flooring-capping, trimming, IQR, Logarithmic Technique these all are techniques that are used to handle outliers. To convert a categorical value into numerical values we can use frequency encoding, target guided encoding, one hot encoding and so on. After gathering and cleaning the data, the data is ready and can be used to train a machine learning model. This sophisticated, previously acquired data was used to train support vector classifiers, logistic regression, and the closest neighborhood of K. Deployment part is the final step where webapp is deployed on Heroku platform for users. [4].

Introduction of disease prediction using machine learning algorithms is explained in section I and Section II represents proposed system architecture Section III represents experimental analysis, results Section IV concludes our proposed system while at the end of this reference papers are listed.

II. SYSTEM ARCHITECTURE

A. Architecture Overview

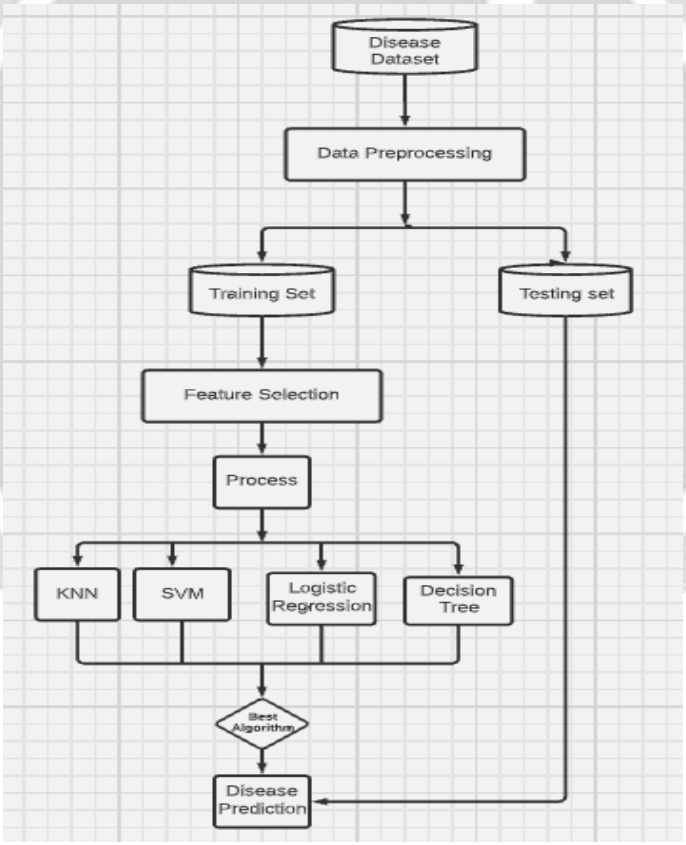


Fig: System Architecture

The data required for machine learning algorithms are taken from Kaggle [5] website for prediction of diseases. The downloaded data includes so many independent columns which represents symptoms and one dependent column which represents results initially data is not in well format so it need to be processed by handling missing values, handling outliers, handling skewness in the features, etc. Data

preparation is the foundation step for any machine learning algorithm. This pre-processed data is divided into a training set and a test set. Unnecessary and unused features have been removed, and all remaining features are provided to machine learning algorithms such as KNN, SVM, logistic regression, and decision trees. The best out of all those results is chosen and testing data is passed to check whether training is done appropriately or not.

B. Algorithms used:

1. K nearest neighbor (KNN) [6]

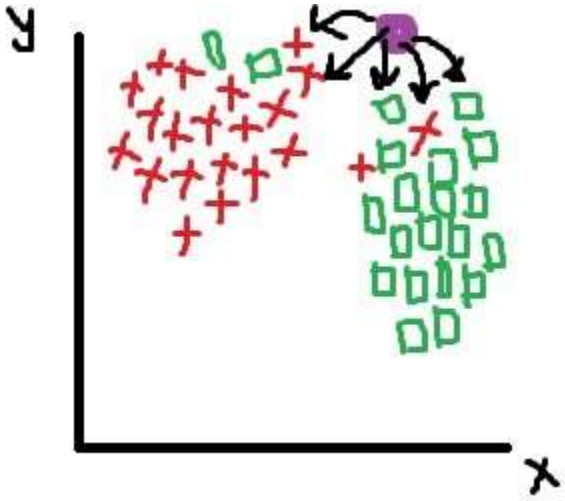


Fig a. K nearest neighbor [4]

1.1**. Algorithm**

1. First find out the value for k i.e. nearest neighbors from the query.

2. Now calculate the distance from your nearest neighbor's k.

3. Check how many belonging to category 1 and category 2.

4. If count of category 1 is greater than category 2 then query belongs to category 1 or vice versa.

1.2. **Calculates the question point distance from each location**

There are 2 techniques to calculate distances.

a. Euclidean distance

$$D=\text{sqrt}\left((x2-x1)^2 + (y2-y1)^2\right)$$

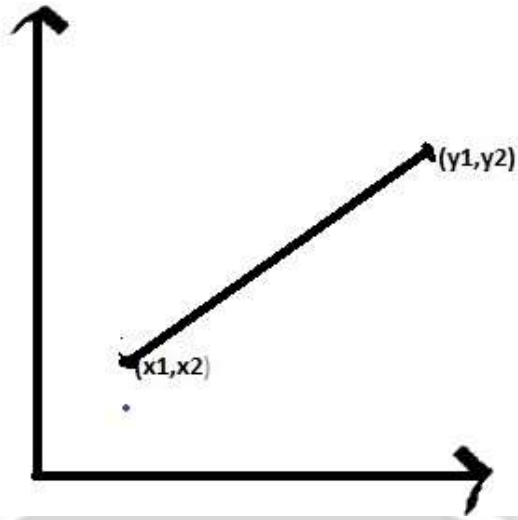


Fig b. Distance formulae [6]

1.3. Use the elbow method to pick a good K Value:

Here we can see that after K> 23 the error rate is usually 0.06-0.05

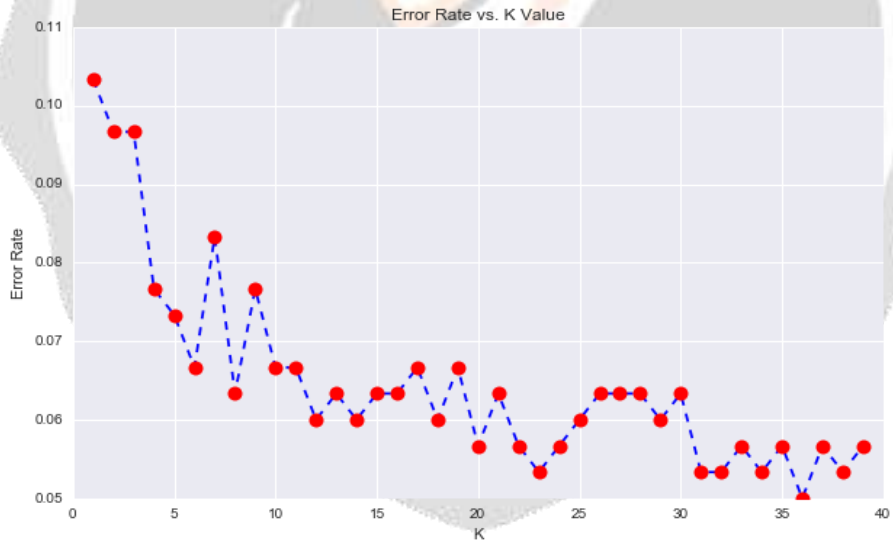


Fig c. Elbow method

1.4. Time and space complexities of KNN [8]:

(n: Number of training examples, d: Number of dimensions of the data, k:Number of neighbors)

Time complexity: 0(k*n*d)

Space complexity: 0(n*d)

2. Support Vector Machine (SVM) [7]

2.1. Introduction

a. Both classification and regression problems are solved by SVM.
b. SVM always tends to better accuracy.

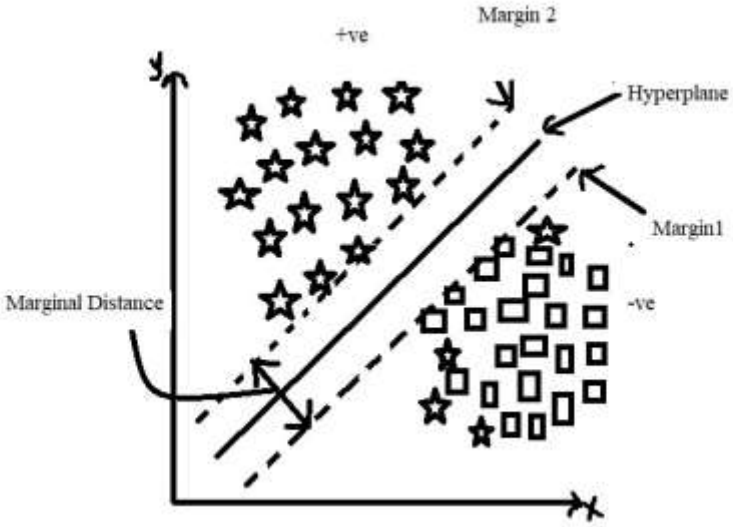


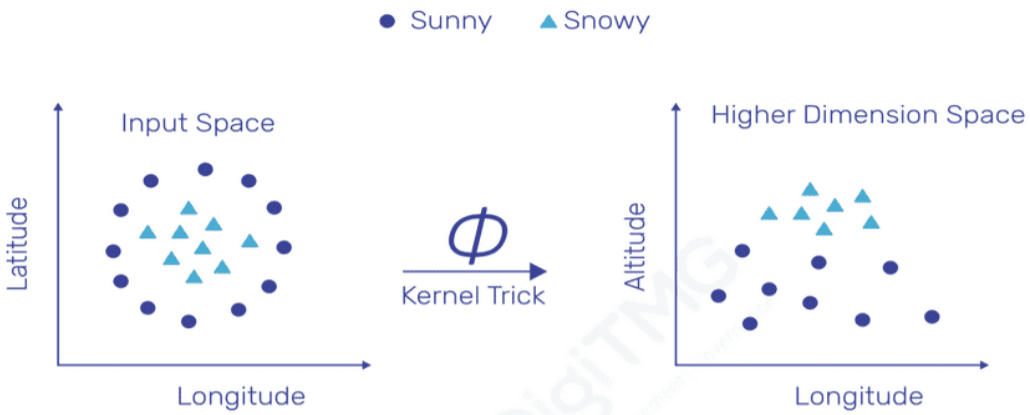
Fig d. Support Vector Machine [6]

In above figure data points are linearly separable. Select the marginal lines having maximum marginal distance

2.2. Important terminologies

a. Marginal lines:-Two parallel lines that are passing through two or more support vectors and parallel to hyperplane are called marginal lines.

b. Marginal distance: The distance between the two lateral lines is called the marginal distance. For more accuracy



it has to be high.

If the data points are not linearly separable, the kernel is used to convert the lower dimensions of the data to the higher dimensions.

Fig e: kernel tricks [7]

There are different kernel tricks [7].

1. Linear kernel: It is the dot product of all the features which don't transfer the data.
2. Polynomial Kernel: It is nonlinear transformation of the data.
3. Gaussian Kernel: It is widely used kernel for nonlinear data.
4. Sigmoid Kernel: It is similar to sigmoid activation function.

Linear kernel: $k(x1,x2) = x1.x2$

Polynomial Kernel: $k(x1,x2) = (\gamma\, x1.x2 + c)\text{^}d$

Gaussian Kernel: $k(x1,x2) = \exp(-\gamma\, |x1 - x2|\text{^}2)$

Sigmoid kernel: $k(x1,x2) = \tanh(\gamma\, x1.x2 + c)$

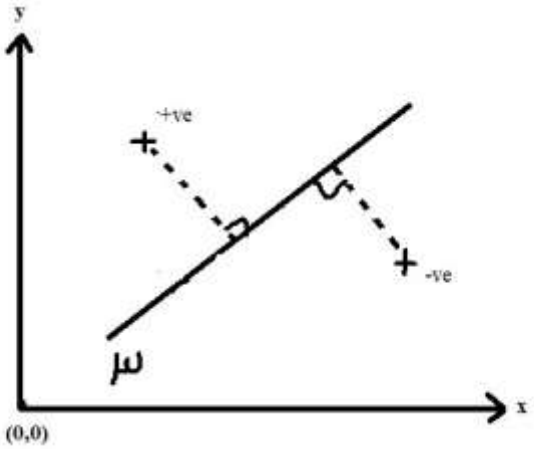
3. Time and space complexities of SVM [8]:

(n: Number of training examples, m: features ,k: number of support vectors)

Time complexity: $0(n^2)$

Space complexity: 0(mk)

3. Logistic Regression [6]



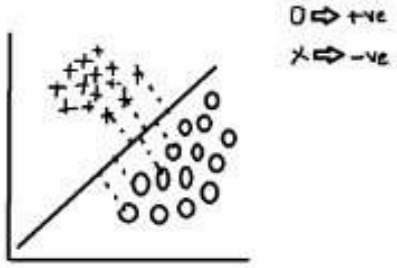


fig f:-Logistic regression [6]

Case 1:

y01=w^tx1>0

y01 =y*w^x>0 (correctly classified as +ve)

Case 2:

y02=w^x2<0

y02=y*w^x>0 (correctly classfied as –ve)

from above here is the final equation that should be maximized

$$\sum \mathbf{yi} * \mathbf{w}^{t} * \mathbf{xi}$$

Minimizing external influences on measuring logistic regression parameters as

$$\mathbf{max} \sum \mathbf{f(yi} * \mathbf{w}^{t}\mathbf{x)}$$

Cost function for logistic regression

Sigmoid activity ranges in range from 0 to 1.

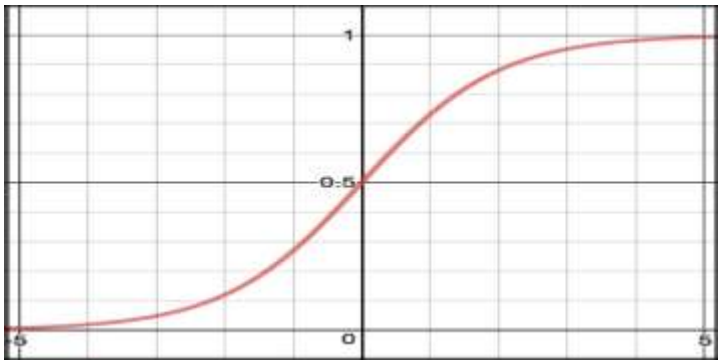


Fig i. sigmoid function [8]

3.2 Time and Space complexities for logistic regression [8]:

(n: number of data points ,d: dimensions)

Time complexity: 0(nd)

Space complexity: 0(d)

4. Decision Tree [8]

4.1 .Entropy

Entropy is a measure of randomness. In other words, it's a measure of unpredictability [7]. Entropy tells the purity of the split.

$$H(s) = -p(+).\log2(p+) – p(-).\log2(p-)$$

(s=sample, p (+): percentage of positive class, p (-): percentage of negative class).

Entropy always lies between 0 and 1. If entropy is approximately equal to 1 then it is completely worst split or impure split. If entropy is near to 0 then it is pure or best split [6].

4.2 Gini Impurity

It tells how good your split is. Gini impurity is used more than entropy because it is computationally efficient with faster execution than entropy [6].

$$GI= 1-\sum(p)^2$$

$$GI=1- [(p (+) ^2+p (-) ^2)]$$

(GI: Gini impurity, p (+): percentage of positive class, p (-): percentage of negative class)

When p (+) increases entropy also increases but gini impurity decreases. The maximum increase of gini impurity is +0.5

4.3 Information Gain

Best classification is done using information gain. Information gain is used to reduce the entropy value by choosing the optimal division of the decision tree [7].

$$Gain(S)= H(S)-\sum|Sv|/|S|*H(Sv)$$

(S: total sample size, H(S): entropy of sample, Sv: Current sample size)

If information gain value is high then that split is accepted as best split [6].

4.4 Time and Space complexity for decision tree [8]:

(n: number of data points, d: number of dimension, k:number of nodes)

Time complexity: 0(n*logn*d)

Space complexity: 0(k)

I. RESULT AND DISCUSSIONS

A. Experimental Setup

All the experimental cases are implemented in python programming using variety of libraries like NumPy, pandas, sklearn, sunbird, etc. With react js for webapp and react native for android application and in environment with System having configuration of Intel processor.

Confusion matrix:

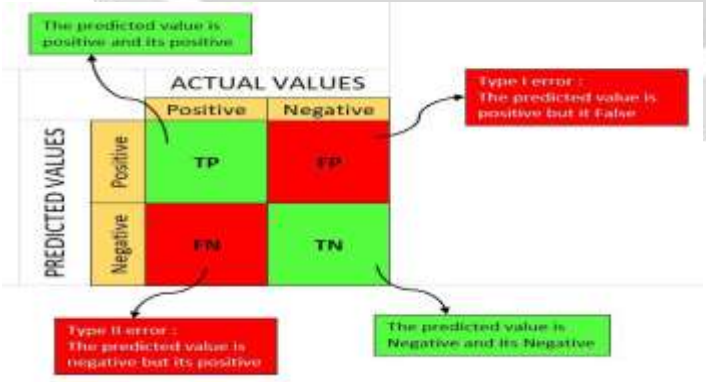


Fig j. Confusion matrix [7]

B. Results

This section compares the performance of each model. Demonstrates accuracy, precision, recall and f1 score of each machine learning model in relation to the disease.[1]

Precision:

The model is out of those total predicted positive and how many of them are actually positive. [6]

$$Precision=TP/ (TP+FP)$$

Recall:

Recall tells us how many of the actual positive cases we were able to predict correctly

$$Recall=TP/ (TP+FN)$$

F1-score:

Since both precision and recall are important for our task, F1 measure was used to [6].

$$F1\text{-}score= 2(recall * precision)/ (recall + precision)$$

Accuracy:

Accuracy of your estimate of both actual and predicted values on all values [6].

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

Comparison of machine learning algorithms for the prediction are shown in the following table.

	KNN				SVC				Logistic Regression				Decision Tree			
	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score	Accuracy
Liver	0.72	0.72	0.71	72	0.76	0.64	0.58	65.2	0.76	0.75	0.74	74.8	0.69	0.69	0.69	70.8
Heart	0.86	0.86	0.86	86.9	0.84	0.83	0.84	84.75	0.82	0.82	0.82	83.15	0.79	0.8	0.79	79.81
Diabetes	0.7	0.69	0.69	72.2	0.8	0.77	0.78	81.1	0.79	0.74	0.75	79.2	0.77	0.75	0.76	79.2
Breast Cancer	0.98	0.97	0.97	97.36	0.95	0.96	0.95	95.61	0.97	0.97	0.97	97.36	0.9	0.91	0.9	90.35

Conclusions

Machine learning models are implemented using Logistic Regression, Decision Tree, KNN and SVM. We used ML algorithms as it enables systems to learn and improve from past experiences and in medical field data is growing rapidly .KNN, Logistic Regression, Decision Tree, and SVM algorithm compared based on accuracy.
Logistic regression gives 74% accuracy for liver disease which is good compared to other models implemented in this paper, KNN gives 86% accuracy for heart disease which is good compared to other models implemented in this paper, SVM gives 81% accuracy for diabetes disease which is good compared to other models implemented in this paper and for the breast cancer KNN gives 97% accuracy which is good compared to other models implemented in this paper. SVM has high time complexity than other used machine learning algorithms, while Logistic Regression has lowest time complexity than other used machine learning algorithms.

References

[1]. D Dhiraj Dahiwade, Prof. Gajanan Patle, Prof. Ektaa Meshram," Designing Disease Prediction Model Using Machine Learning Approach ," IEEE Trans. Neural Netw., vol. 14, no. 1, pp.195-200, Jan. 2019.

[2]. "UCI Machine Learning Repository." [Online]. Available: https://archive.ics.uci.edu/ml/index.php.

[3]. Allen Daniel Sunny1, Sajal Kulshreshtha, Satyam Singh3, Srinabh, Mr. Mohan Ba, Dr. Sarojadevi H " Disease Diagnosis System By Exploring Machine Learning Algorithms", International Journal of Innovations in Engineering and Technology (IJIET) Volume 10 Issue 2 May 2018.

[4]. https://www.heroku.com/

[5]. https://www.kaggle.com/

[6] . https://www.youtube.com/user/krishnaik06

[7]. https://www.analyticsvidhya.com/

[8]. https://medium.com/