

Machine Learning Algorithms: A Review

Darpan Pandey¹, Kamal Niwaria², Bharti Chourasia³

¹M.E (D.C.) Student, Dept of Electronics & Communication, RKDFIST (Bhopal), M.P., India

²Assistant Professor, Dept. of Electronics & Communication, RKDFIST (Bhopal), M.P., India

³Associate Professor, Dept. of Electronics & Communication, RKDFIST (Bhopal), M.P., India

Abstract - Nowadays artificial intelligence is the trendiest research area and one of its most popular sub-domains is machine learning. Its mechanism is based on making the system intelligent by past experiences. There are a wide range of applications of machine learning methods such as pattern recognition, image classification, model prediction, data mining, search engines, sentiment analysis, time series forecasting, structural health monitoring, virtual personal assistants - Siri, Alexa, Google Now, etc. The major goal of this paper is to provide the overview of different machine learning approaches.

Key Words: component, formatting, style, styling, insert

1. INTRODUCTION

Machine Learning is presently most popular and complex computer vision approach that became most prominent in the research and industry. It is presently most favorite innovative area in health care, finance, security, data management, trend analysis and prediction. Machine learning is based on making a system so intelligent that it can perform decision making task itself without being an external programme [1, 2].

First, in 1946, ENIAC – the first computer system was invented. The first machine learning thought rises with the first self -playing game program written by Arthur Samuel (IBM) in 1952. Then in 1957 Rosenblatt proposed the perceptron model, a simplex approach to neuron model. In 1967, nearest neighbor approach was modeled to recognize patterns a.k.a. pattern recognition. In 1990s, the machine learning field became subject of interest in various fields due to merger of computer science and statistical methods [3]. Today's world is the world of artificial intelligence, so the machine learning and its allied branches are most favorable research area for researchers in the current era.

2. MACHINE LEARNING MODEL

The method of machine learning basically involves the automatic learning of computer system or machine through the given data a.k.a. raw data or referred as datasets. This process generally includes two major steps-

- (I) Training of model
- (II) Decision making or Testing of model

For this purpose we have two types of datasets i.e. training data and test data. Training data is used to train model involving data preprocessing and feature extraction before training of model. The obtained trained model is used for determining the result of unknown datasets or a.k.a. test data. The accuracy of model can also be determined by decision making of test data for which we already know the result [4].

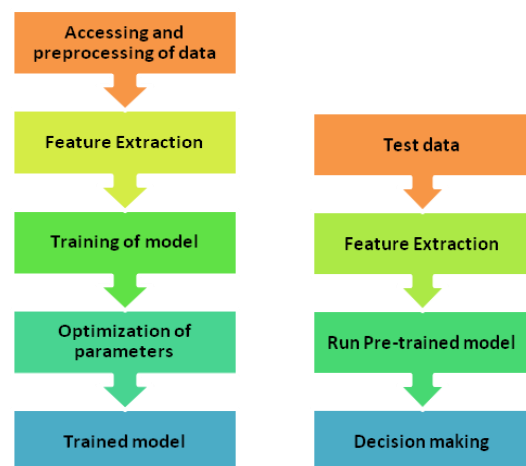


Fig -1: Training and testing of machine learning model

3. CLASSIFICATION OF MACHINE LEARNING ALGORITHMS

Machine learning methods are broadly classified into three learning methods - Supervised, unsupervised and reinforcement learning. We discuss these approaches one by one in detail below [4].

3.1 Supervised Learning

As the name suggests, this algorithm requires supervision for prediction or decision making. Here the input dataset is partitioned into training and test data. In supervised learning, target or output values are already assigned to training data for training of the model. Supervised learning is most popularly used to find the solution for classification and regression problems. Classification methods are used for discrete-value problems while regression methods are used for decision making in continuous value problems [4, 5].

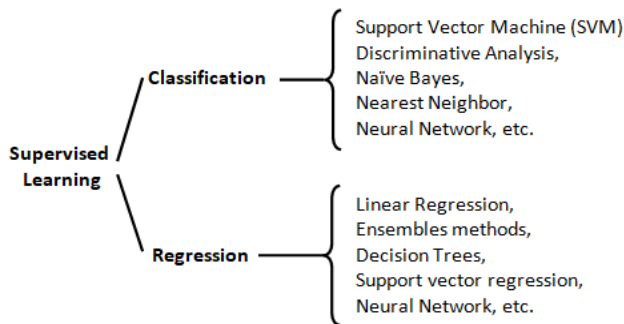


Fig-2: Classification of Supervised Learning algorithms

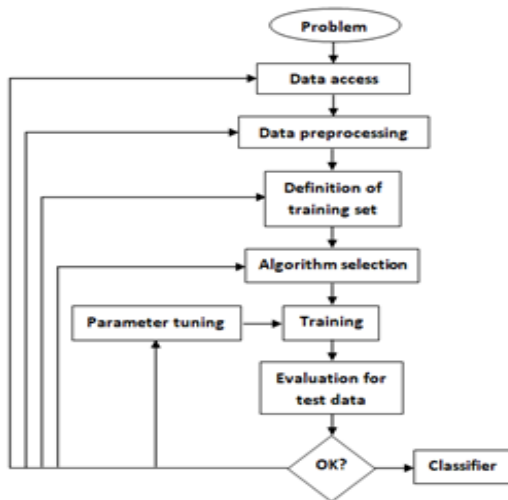


Fig-3: Workflow of supervised machine learning algorithm

Fig-4: Working of Support Vector Machine

```

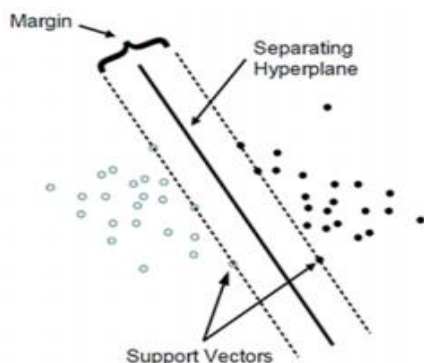
INPUT:  $S, \lambda, T, k$ 
INITIALIZE: Choose  $w_1$  s.t.  $\|w_1\| \leq 1/\sqrt{\lambda}$ 
FOR  $t = 1, 2, \dots, T$ 
  Choose  $A_t \subseteq S$ , where  $|A_t| = k$ 
  Set  $A_t^+ = \{(x, y) \in A_t : y \langle w_t, x \rangle < 1\}$ 
  Set  $\eta_t = \frac{1}{|A_t^+|}$ 
  Set  $w_{t+\frac{1}{2}} = (1 - \eta_t \lambda)w_t + \frac{\eta_t}{k} \sum_{(x,y) \in A_t^+} y x$ 
  Set  $w_{t+1} = \min \left\{ 1, \frac{1/\sqrt{\lambda}}{\|w_{t+\frac{1}{2}}\|} \right\} w_{t+\frac{1}{2}}$ 
OUTPUT:  $w_{T+1}$ 

```

Fig-4: Pseudo code for Support Vector machine [8]

3.1.1 Support vector machine (SVM)

This is a most popular method for classification problems. It basically works on the method of calculation of margin. This approach uses hyperplane or a set of hyperplanes to separate features having high dimensional data to discriminate the objects between different classes. Initially SVM was only able to handle two-class problems or binary-classification problems but later the approach to solve multi-class problems were also developed by using a set of hyperplanes [6, 7, 8]



3.1.2 Discriminant analysis

A constructive way to solve the dimensionality issue is to reduce the dimension of data. As the dimensions enhance in number, there is positive increment in space volume that causes sparsity in the provided dataset resulting in problems by statistical computations. Discriminant analysis involves clustering of objects on the basis of similarity in their features to classify the data into various classes. This is a perfect example of dimensionality reduction approach [5]. Popular discriminative approaches based on dimensional reduction are [5, 9]:

- Multidimensional Scaling (MDS)
- Linear Discriminant Analysis (LDA)
- Mixture Discriminant Analysis (MDA)
- Quadratic Discriminant Analysis (QDA)
- Flexible Discriminant Analysis (FDA)

3.1.3 Naïve Bayes

Naïve Bayes is another classification based machine learning approach which involves mainly the conditional probability method to determine whether the object belongs to particular class or not. Based on the probability of belonging to particular class a tree network is created a.k.a. Bayesian network [10].

```

INPUT: training set  $T$ , hold-out set  $H$ , initial number of components  $k_0$ , and convergence thresholds  $\delta_{EM}$  and  $\delta_{Add}$ .

Initialize  $M$  with one component.
 $k \leftarrow k_0$ 
repeat
  Add  $k$  new mixture components to  $M$ , initialized using  $k$  random examples from  $T$ .
  Remove the  $k$  initialization examples from  $T$ .
  repeat
    E-step: Fractionally assign examples in  $T$  to mixture components, using  $M$ .
    M-step: Compute maximum likelihood parameters for  $M$ , using the filled-in data.
    If  $\log P(H|M)$  is best so far, save  $M$  in  $M_{best}$ .
    Every 5 cycles, prune low-weight components of  $M$ .
  until  $\log P(H|M)$  fails to improve by ratio  $\delta_{EM}$ .
   $M \leftarrow M_{best}$ 
  Prune low weight components of  $M$ .
   $k \leftarrow 2k$ 
until  $\log P(H|M)$  fails to improve by ratio  $\delta_{Add}$ .
Execute E-step and M-step twice more on  $M_{best}$ , using examples from both  $H$  and  $T$ .
Return  $M_{best}$ .

```

Fig-5: Pseudo code for Naïve Bayes [11]

3.1.4 Nearest neighbor

Nearest neighbor is a instance based algorithm generally referred as K-nearest neighbor (KNN) algorithm. In this approach basically the prediction for test data is made based on the object having nearest similar feature than the other available in training datasets. Hence larger the training dataset, result accuracy is higher [4, 5].

3.1.5 Regression algorithms

Regression analysis is based on fitting of a curve on the basis of training data available to identify the decision for continuous value problems. It models a mutual relationship among the parameters on the basis of error predicted by the trained model. Regression methods are used for those problems where prediction is to be made have continuous nature i.e. large or infinite solutions. Well known regression algorithms used are [2, 5]:

- Ordinary Least Squares Regression (OLSR)
- Linear Regression
- Logistic Regression
- Support Vector Regression
- Multivariate Adaptive Regression Splines (MARS)
- Locally Estimated Scatterplot Smoothing (LOESS)

```

Let  $W = \{x_1, x_2, \dots, x_n\}$  be a set of  $n$  labeled samples. The algorithm is as follows:
BEGIN
  Input  $y$ , of unknown classification.
  Set  $K, 1 \leq K \leq n$ .
  Initialize  $i = 1$ .
  DO UNTIL ( $K$ -nearest neighbors found)
    Compute distance from  $y$  to  $x_i$ .
    IF ( $i \leq K$ ) THEN
      Include  $x_i$  in the set of  $K$ -nearest neighbors
    ELSE IF ( $x_i$  is closer to  $y$  than any previous nearest neighbor) THEN
      Delete farthest in the set of  $K$ -nearest neighbors
      Include  $x_i$  in the set of  $K$ -nearest neighbors.
    END IF
    Increment  $i$ .
  END DO UNTIL
  Determine the majority class represented in the set of  $K$ -nearest neighbors.
  IF (a tie exists) THEN
    Compute sum of distances of neighbors in each class which tied.
    IF (no tie occurs) THEN
      Classify  $y$  in the class of minimum sum
    ELSE
      Classify  $y$  in the class of last minimum found.
    END IF
  ELSE
    Classify  $y$  in the majority class.
  END IF
END
  
```

Fig-6: Pseudo code for Nearest Neighbor [12]

3.1.6 Ensembles methods

Till 1990s, the individual learning methods remained much popular but later it has been observed that use of combination of various learning methods a.k.a. ensemble learning approach provides much better result for overall problems instead of a particular learning algorithm [5]. The two most prominent ensembles techniques are:

a) Boosting: It is based on reduction of the variance and bias. An effective learning approach is obtained by combining a number of weak learning approaches that rarely produce an efficient and accurate result.

b) Bagging: A.k.a. bootstrapping used to obtain more accurate and stable result for a machine learning method. This can be used in both supervised methods i.e. in classification problems and also in regression problems. Bootstrapping is also used to reduce the invariance problem and also handles the overfitting issue [5, 13].

```

Input: Data set  $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$ ;
       Base learning algorithm  $\mathcal{L}$ ;
       Number of learning rounds  $T$ .

Process:
   $D_1(i) = 1/m$ .
  for  $t = 1, \dots, T$ :
     $h_t = \mathcal{L}(\mathcal{D}, D_t)$ ;
     $\epsilon_t = \Pr_{i \sim D_t}[h_t(\mathbf{x}_i) \neq y_i]$ ;
     $\alpha_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$ ;
     $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \exp(-\alpha_t) & \text{if } h_t(\mathbf{x}_i) = y_i \\ \exp(\alpha_t) & \text{if } h_t(\mathbf{x}_i) \neq y_i \end{cases}$ 
     $= \frac{D_t(i) \exp(-\alpha_t y_i h_t(\mathbf{x}_i))}{Z_t}$ 
  end.

Output:  $H(\mathbf{x}) = \text{sign}(f(\mathbf{x})) = \text{sign} \sum_{t=1}^T \alpha_t h_t(\mathbf{x})$ 
  
```

Fig-7: Pseudo code for AdaBoost [13]

```

Input: Data set  $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$ ;
       Base learning algorithm  $\mathcal{L}$ ;
       Number of learning rounds  $T$ .

Process:
  for  $t = 1, \dots, T$ :
     $\mathcal{D}_t = \text{Bootstrap}(\mathcal{D})$ ;
     $h_t = \mathcal{L}(\mathcal{D}_t)$ 
  end.

Output:  $H(\mathbf{x}) = \text{argmax}_{y \in \mathcal{Y}} \sum_{t=1}^T 1(y = h_t(\mathbf{x}))$ 
  
```

Fig-8: Pseudo code for Bagging [13]

The other popular ensembles methods are [3]:

- Random Forest
- Bootstrapped Aggregation
- Gradient Boosted Regression Trees (GBRT)
- Stacked Generalization (blending)
- AdaBoost
- Gradient Boosting Machines (GBM)

3.1.7 Decision trees

Learning in decision tree involves prediction of a model that creates mapping between observations and conclusions for an object to its target value. The models of decision trees that have fixed target values are generally known as classification tree models. The decision tree contains internal nodes which represent the feature for a particular object while the leaf nodes represent class labels. The regression trees are those for which the target values are continuous in nature. Hence decision trees can be used for both type of problems i.e. classification and regression [4, 14].

Some of the important decision tree methods are [14]:

- Classification and Regression Tree (CART)
- Iterative Dichotomiser 3 (ID3)
- C4.5 and C5.0
- Chi-squared Automatic Interaction Detection (CHAID)
- Decision Stump
- Conditional Decision Trees

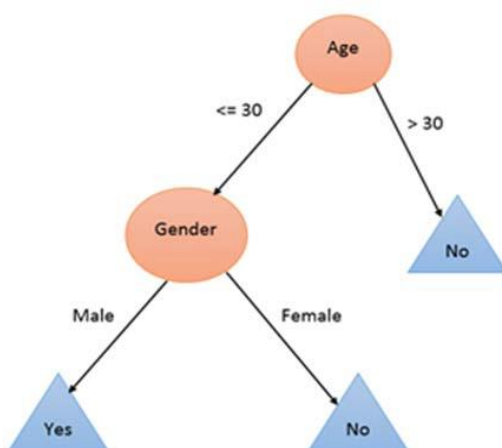


Fig-9: Decision Tree

```

procedure DTInducer(S, A, y)
1: T = TreeGrowing(S, A, y)
2: Return TreePruning(S, T)
procedure TreeGrowing(S, A, y)
1: Create a tree T
2: if One of the Stopping Criteria is fulfilled then
3:   Mark the root node in T as a leaf with the most common
   value of y in S as the class.
4: else
5:   Find a discrete function f(A) of the input attributes val-
   ues such that splitting S according to f(A)'s outcomes
   (v1, ..., vn) gains the best splitting metric.
6:   if best splitting metric ≥ threshold then
7:     Label the root node in T as f(A)
8:     for each outcome vi of f(A) do
9:       Subtreei = TreeGrowing(σf(A)=vi, S, A, y).
10:      Connect the root node of T to Subtreei with an
      edge that is labelled as vi
11:    end for
12:  else
13:    Mark the root node in T as a leaf with the most
    common value of y in S as the class.
14:  end if
15: end if
16: Return T
procedure TreePruning(S, T, y)
1: repeat
2:   Select a node t in T such that pruning it maximally
   improve some evaluation criteria
3:   if t ≠ ∅ then
4:     T = pruned(T, t)
5:   end if
6: until t = ∅
7: Return T
  
```

Fig-10: Pseudo code for Decision Tree [14]

3.1.8 Neural networks

The artificial neural network model is based on biological theory of neurons (Primary unit of a cell in human brain). Similar to human brain, it consists of interconnection of a number of nodes (neurons) in the form of input layer, hidden layer and output layer. The neurons of one layer are fully connected to neurons of next layer in the neural network [4, 5, 15].

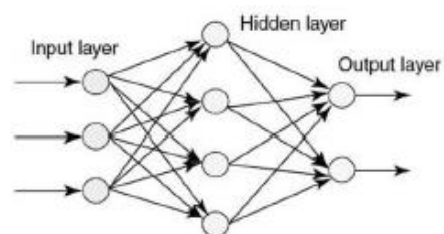


Fig-11: Structure of an Artificial Neural Network

Neural network model can be used for both of the classification and regression analysis. Supervised neural model is based on error correction of weights associated with the neuron interconnection based on the predicted output and actual target value so that associated weights can

be updated to further improve the performance. The most popular methods are Perceptron model, Back-Propagation model, Hopfield Network, Radial Basis Function Network (RBFN), etc.

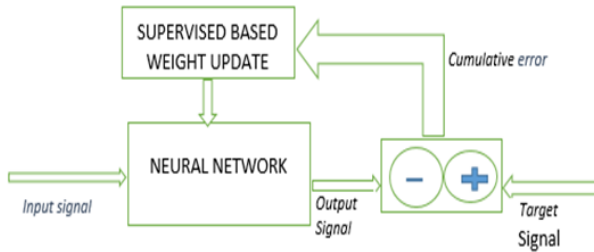


Fig-12: Neural Network for Supervised Learning

3.2. Unsupervised Learning

Unsupervised learning is a class of machine learning that perform inferred tasks to determine the obscured arrangement from unlabeled dataset. Here model is trained based on the features of unlabeled data and then decision is made for the test data available. Unsupervised learning generally focuses on clustering and dimensional reduction problems [3, 16].

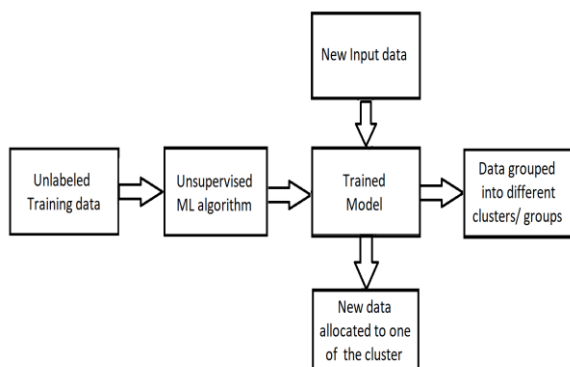


Fig-12: Implementation of Unsupervised Learning algorithm

The most popular unsupervised algorithms are K-means clustering, Principal component analysis (PCA), Hierarchical model, Hidden Markov model, neural network, etc [4].

3.2.1 K-means clustering

Clustering algorithm is based on the grouping of objects based on similar features into various clusters. This distributes the available data into various groups or clusters, in the manner that each group contains the data with less distinguish features within the range of a definite distance. K-means model is a centroid based approach having K-clusters so that the mean value lie at the centre of its cluster [4, 17].

```
function Direct-k-means()
Initialize k prototypes (w1, ..., wk) such that wj = il, j ∈ {1, ..., k}, l ∈ {1, ..., n}
Each cluster Cj is associated with prototype wj
Repeat
for each input vector il, where l ∈ {1, ..., n},
do
Assign il to the cluster Cj, with nearest prototype wj.
(i.e., |il - wj*| ≤ |il - wj|, j ∈ {1, ..., k})
for each cluster Cj, where j ∈ {1, ..., k}, do
Update the prototype wj to be the centroid of all samples currently in Cj, so that wj = ∑il ∈ Cj il / |Cj|
Compute the error function:
E = ∑j=1k ∑il ∈ Cj |il - wj|2
Until E does not change significantly or cluster membership no longer changes
```

Fig-13: Pseudo code for K-means clustering [17]

3.2.2 Principal component analysis (PCA)

PCA is fastest and easy to implement efficient approach among others. PCA is a dimensionality reduction based approach that transforms high dimensional data to low dimensional data however it has limited ability of linear transformation from one space into another space. It focuses on highly important traits while ignoring less important. That means retaining the features of objects that have distinct information than the others while similar features are eliminated to reduce dimensions of data [4, 18].

```
R ← X
for(k = 0, ..., K - 1) do
{
λ = 0
T(k) ← R(k)
for(j = 0, ..., J) do
{
P(k) ← RTT(k)
P(k) ← P(k) / ||P(k)||-1
T(k) ← RP(k)
λ' ← ||T(k)||
if(|λ' - λ| ≤ ε) then break
λ ← λ'
}
R ← R - T(k)(P(k))T
}
return T, P, R
```

Fig-14: Pseudo code for PCA algorithm [18]

3.2.3 Neural networks

Neural networks can be also be used in unsupervised learning approach. In this the neural model doesn't have any idea about the target output. This network classifies the data into various class based on the similarity of parameters in the data. The unsupervised neuron model correlates the input parameters of various objects and groups them into various classes [16].

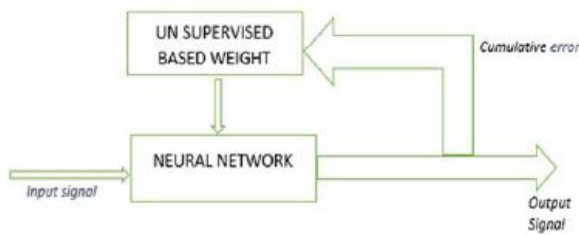


Fig-15: Unsupervised Learning model

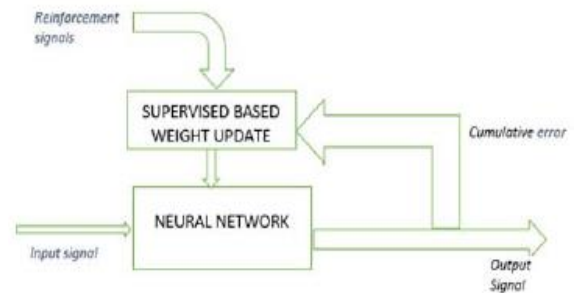


Fig-16: Reinforced Neural Network

3.3 Reinforcement Learning

This type of learning is preferably used in robotics and automation, gaming world, for navigation purpose and various other applications. This is a dynamic learning approach that creates an environment to achieve a definite outcome without being explicitly informing how near the target is. Reinforcement model is a class of learning approach that is based on try and hit method and selects the action that gives more positive (more close to target). It is a kind of composition of both supervised and unsupervised learning models. It initiates the learning with some random actions and then provides feedback to correct the action in positive manner [19].

Reinforcement learning model is exclusively based on two functions: trial and error finding and delayed outcome [20]. The reinforced model: the agent creates a behavior based on the input i from the input function I and state transition r both from current state s . Due to the behavior B , an action a is generated to modify the environment.

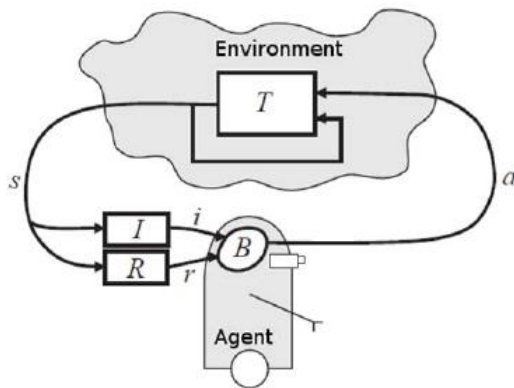


Fig-16: Reinforcement Learning Mechanism

Reinforcement neuron model is based on the feedback that the action taken by the network is corrective or not. If correction is positive then the network takes decision in positive forward direction to provide more nearest outcome to target. The thing to be noted that the network contains no prior knowledge regarding target [21].

4. CONCLUSION

In this paper we reviewed various machine learning models that are most popular. In today's era, humans are using machine learning based intelligent systems in daily life in direct or indirect purpose. This paper provides an insight overview of these trending machine learning models that can be used in various applications and advancement to these approaches.

REFERENCES

- [1] Munoz, and Andres, "Machine Learning and Optimization," Retrieved 2017-06-01 from https://www.cims.nyu.edu/~munoz/files/ml_optimization.pdf
- [2] Hosch, and L. William, "Machine Learning," Retrieved 2017-06-01 from <http://www.britannica.com/EBchecked/topic/116194/machinelearning>
- [3] Ron Kohavi, Foster Provost, "Glossary of terms: Machine Learning," pp. 271-274, 1998
- [4] N. K. Chauhan, K. Singh, "A Review on Conventional Machine Learning vs Deep Learning," International Conference on Computing, Power and Communication Technologies (GUCON), Greater Noida, India. Sep 28-29, 2018
- [5] S.B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques", Informatica 31 (2007) 249-268
- [6] D. Meyer, "Support Vector Machines – The Interface to libsvm in package 1071", August 2015
- [7] Y. Lee, Y. Lin, and G. Wahba, "Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data," Journal of the American Statistical Association, 99(465):67-81, March 2004.

- [8] S. S. Shwartz, Y. Singer, N. Srebro, "Pegasos: Primal Estimated sub-Gradient Solver for SVM", Proceedings of the 24th International Conference on Machine Learning, Corvallis, OR, 2007
- P. Harrington, "Machine Learning in action", Manning Publications Co., Shelter Island, New York, 2012
- [9] M. J. Islam, Q. M. J. Wu, M. Ahmadi, and M. A. Sid-Ahmed, "Investigating the Performance of Naive- Bayes Classifiers and K- Nearest Neighbor Classifiers," Journal on Convergence Information Technology, vol. 5, no. 2, 2010.
- [10] D. Lowd, P. Domingos, "Naïve Bayes Models for Probability Estimation," ICML '05 Proceedings of the 22nd international conference on Machine learning, 2005
- [11] J. M. Keller, M. R. Gray, J. A. Givens Jr., "A Fuzzy K-Nearest Neighbor Algorithm", IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-15, No. 4, August 1985
- [12] Z.H.Zhou, "Ensemble Learning", National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China, 2013
- [13] L. Rokach and O. Maimon, "Top-down induction of decision trees classifiers - a survey," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 35, no. 4, pp. 476-487, Nov. 2005.
- [14] V. Sharma, S. Rai, A. Dev, "A Comprehensive Study of Artificial Neural Networks", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, October 2012
- [15] <https://www.simplilearn.com/what-is-machine-learning-and-why-it-matters-article>
- [16] K. Alsabati, S. Ranaka, V. Singh, "An efficient k-means clustering algorithm", Electrical Engineering and Computer Science, 1997.
- [17] M. Andrecut, "Parallel GPU Implementation of Iterative PCA Algorithms", Institute of Biocomplexity and Informatics, University of Calgary, Canada, 2008.
- [18] R.S. Sutton, "Introduction: The Challenge of Reinforcement Learning", Machine Learning, 8, Page 225-227, Kluwer Academic Publishers, Boston, 1992
- [19] L.P. Kaelbling, M. L. Littman, A. W. Moore, "Reinforcement Learning: A Survey", Journal of Artificial Intelligence Research, 4, Page 237-285, 1996
- [20] S.B. Hiregoudar, K. Manjunath, K. S. Patil, "A Survey: Research Summary on Neural Networks", International Journal of Research in Engineering and Technology, Volume 03, Special Issue 03, pages 385-389, May, 2014