Proceedings of Second Shri Chhatrapati Shivaji Maharaj QIP Conference on Engineering Innovations
Organized by Shri Chhatrapati Shivaji Maharaj College of Engineering, Ahmednagar
In Association with Novateur Publications JournalNX-ISSN No: 2581-4230
February, 22nd and 23rd, 2019

A SURVEY ON VARIOUS MACHINE LEARNING ALGORITHMS

Mr.Jayant K.Shimpi
Computer Engineering
D.Y.Patil College of Engg,Akurdi
Pune ,India
jayant.shimpi06@gmail.com

Ms.Pooja M.Deshpande
Computer Engineering
D.Y.Patil College of Engg,Akurdi
Pune ,India
Poojadeshpande20@gmail.com

***Abstract*—** Machine Learning is the field of computer science that gives computers the capability to learn without being explicitly programmed. Machine learning provides the learning ability to computers which invoke human kind of behavior in computers. Machine learning includes number of algorithms which are used to accomplish the specified tasks.

Keywords— machine; learning; algorithms; pseudocode

I. INTRODUCTION

Machine learning is one of the most interesting fields the technology has witnessed. It teaches computers to invoke humanly behaviour. It primarily focuses on teaching machines how to handle data more efficiently. Sometimes the situation may arise when we fail to analyse some data or to study the pattern, there we can effectively use machine learning algorithms. Machine learning functions similar to the human brain. The more humans know, the more easily we can predict. Machines make an accurate prediction using training sequence. When we give the machine a similar example, it can figure out the outcome. Machine learning is broadly categorized into two learning methods:

Types of Learning:

1. Supervised Learning

In Supervised learning, an input consisting of training dataset and test dataset is to be given to the algorithm wherein input is mapped to output using different algorithms. Test dataset will be processed by algorithms which use some kinds of patterns from training dataset to get the prediction or classification.[2]

Some of the Supervised Learning Algorithms are as follows:

1. Decision tree: Decision tree is the most powerful algorithm used for classification. Each tree consists of nodes and branches. Nodes represent attributes to be classified and branches represents value the node can take.
The pseudo code for decision tree is as follows:

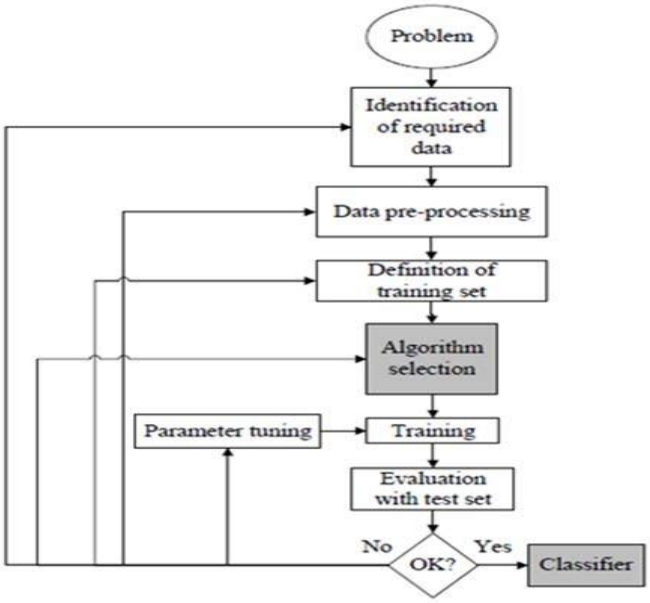

Fig.1: Workflow of supervised machine learning algorithm

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 values such that splitting S according to $f(A)$'s outcomes $(v_1, \ldots, v_n)$ gains the best splitting metric.
6: **if** best splitting metric $\geq$ treshold **then**
7: Label the root node in T as $f(A)$
8: **for** each outcome v_i of f(A) **do**
9: $Subtree_i = TreeGrowing(\sigma_{f(A)=v_i} S, A, y)$.
10: Connect the root node of T to $Subtree_i$ with an edge that is labelled as v_i
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** $t \neq \emptyset$ **then**
4: $T = pruned(T, t)$
5: **end if**
6: **until** t=Ø
7: Return T

Fig. 2. Pseudo Code for Decision Tree Algorithm

Example of Decision tree classification:

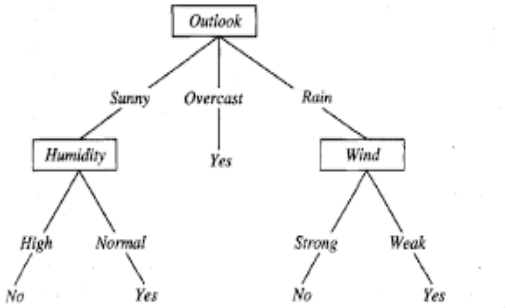

Fig 3:Decision tree Example

2. Naive Bayes: Naive Bayes is a machine learning algorithm used to construct classifiers which will allocate class labels to problem instances. This algorithm follows Bayes probabilistic approach. Naive Bayes model considers two approaches: Class Probability and conditional probability. It is used for binary or multiclass classification.
The pseudo code for Naive Bayes is as follows:

INPUT: training set T, hold-out set H, initial number of components k_0, and convergence thresholds δ_{EM} and δ_{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 δ_{EM}.
  $M \leftarrow M_{best}$
  Prune low weight components of M.
  $k \leftarrow 2k$
until $\log P(H|M)$ fails to improve by ratio δ_{Add}.
Execute E-step and M-step twice more on M_{best}, using examples from both H and T.
Return M_{best}.

Fig 4:Pseudo code for Naive Bayes

Example of Naïve Bayes:
Following is the set of observations:

chills	runny nose	headache	fever	flu?
Y	N	Mild	Y	N
Y	Y	No	N	Y
Y	N	Strong	Y	Y
N	Y	Mild	Y	Y
N	N	No	N	N
N	Y	Strong	Y	Y
N	Y	Strong	N	N
Y	Y	Mild	Y	Y

Do I believe that a patient with the following symptoms has the flu?

chills	runny nose	headache	fever	flu?
Y	N	Mild	Y	?

Fig. 5:Naïve Bayes Example

3. Support Vector Machine:
Another most widely used Supervised Machine Learning algorithm which is used for classification and regression is Support Vector Machine(SVM).It is a non-probabilistic binary linear classifier. At the same time, SVM also performs non-linear classification. It uses a terminology called Hyperplane. Margin calculation is done by drawing margins between the classes. The margins are drawn such that the distance between the margin and the classes is maximum and classification error is minimized.

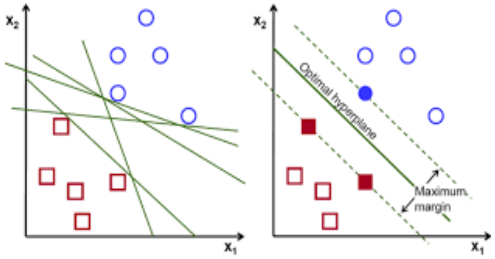

Fig 6:Support Vector Machine

The pseudo code for Support Vector Machine (SVM) is as follows:

INPUT: S, λ, T, k
INITIALIZE: Choose $\mathbf{w}_1$ s.t. $\|\mathbf{w}_1\| \leq 1/\sqrt{\lambda}$
FOR $t = 1, 2, \ldots, T$
  Choose $A_t \subseteq S$, where $|A_t| = k$
  Set $A_t^+ = \{(\mathbf{x}, y) \in A_t : y \langle \mathbf{w}_t, \mathbf{x} \rangle < 1\}$
  Set $\eta_t = \frac{1}{\lambda t}$
  Set $\mathbf{w}_{t+\frac{1}{2}} = (1 - \eta_t \lambda)\mathbf{w}_t + \frac{\eta_t}{k} \sum_{(\mathbf{x},y) \in A_t^+} y\,\mathbf{x}$
  Set $\mathbf{w}_{t+1} = \min\left\{1, \frac{1/\sqrt{\lambda}}{\|\mathbf{w}_{t+\frac{1}{2}}\|}\right\} \mathbf{w}_{t+\frac{1}{2}}$
OUTPUT: $\mathbf{w}_{T+1}$

Fig. 7:Pseudo code for Support Vector Machine

1. Unsupervised Learning
Unsupervised learning is the place you just have input information (X) and no comparing yield factors. The objective for unsupervised learning is to display the basic structure or conveyance in the information so as to get familiar with the information. These are called unsupervised learning on the grounds that dissimilar to managed learning above there is no right answers and there is no instructor. Calculations are left to their very own devises to find and present the fascinating structure with regards to the information. Unsupervised learning issues can be additionally gathered into bunching and affiliation issues.

Unsupervised learning problems can be further grouped into clustering and association problems.

- Clustering: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.
- Association: An association rule learning problem is where you want to discover rules that describe large

portions of your data, such as people that buy X also tend to buy Y.

Unsupervised Machine Learning tries to find hidden structure/ pattern in unlabeled data.

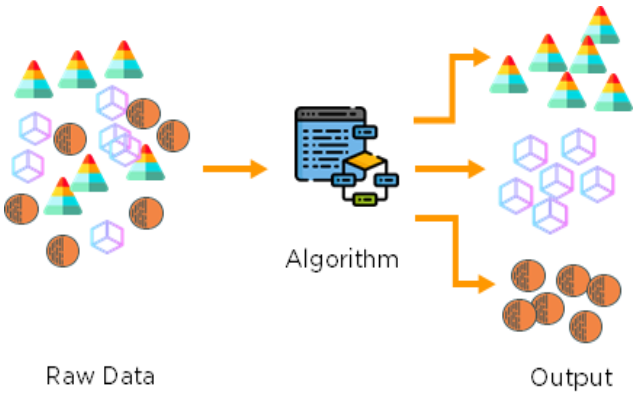

Fig. 8:Unsupervised Learning[2]

Types Of Unsupervised Learning:

Unsupervised learning has two types. They are:

- **Clustering:** Clustering is used for analyzing and grouping data which does not include pre-labeled class or class attributes.

for example, age and sex, or as unpredictable as persona and buy process. Unsupervised learning can enable you to achieve this assignment naturally.

K-Means Clustering – clustering your data points into a number (K) of mutually exclusive clusters. A lot of the complexity surrounds how to pick the right number for K.

Hierarchical Clustering – clustering your data points into parent and child clusters. You might split your customers between younger and older ages, and then split each of those groups into their own individual clusters as well.

Probabilistic Clustering – clustering your data points into clusters on a probabilistic scale.

K-Means Clustering

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity. The results of the K-means clustering algorithm are:

1. The centroids of the K clusters, which can be used to label new data
2. Labels for the training data (each data point is assigned to a single cluster)

Algorithmic steps for k-means clustering:

Let X = {x1,x2,x3,……..,xn} be the set of data points and V = {v1,v2,…….,vc} be the set of centers.

1) Randomly select ‘c’ cluster centers.

2) Calculate the distance between each data point and cluster centers.

3) Assign the data point to the cluster center whose distance from the cluster center is minimum of all the

cluster centers..

4) Recalculate the new cluster center using:

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$$

where, ‘ci’ represents the number of data points in ith cluster.

5) Recalculate the distance between each data point and new obtained cluster centers.

6) If no data point was reassigned then stop, otherwise repeat from step 3).

Hierarchical clustering:

Hierarchical clustering, also known as hierarchical cluster analysis, is an algorithm that groups similar objects into groups called clusters. The endpoint is a set of clusters, where each cluster is distinct from each other cluster, and the objects within each cluster are broadly similar to each other.

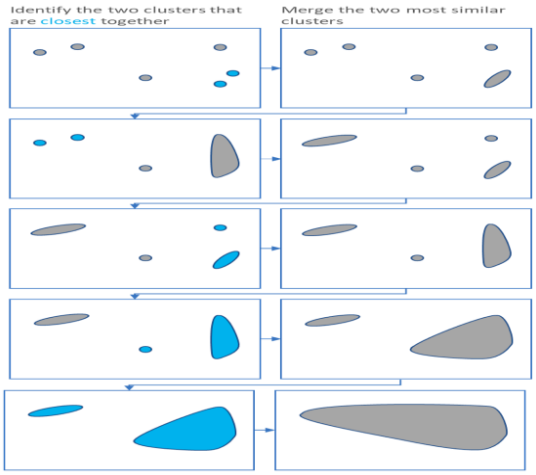

Fig. 9:Hierarchical Clustering

The main output of Hierarchical Clustering is a dendrogram, which shows the hierarchical relationship between the clusters:

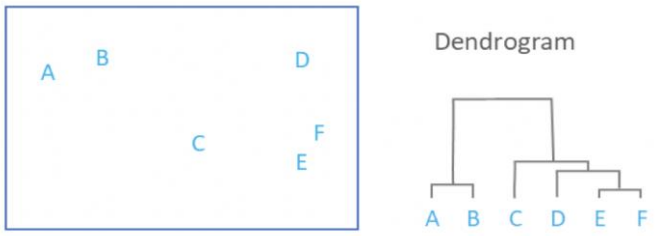

Fig. 10:Dendrogram

Probabilistic Clustering
Clustering your data points into clusters on a probabilistic scale.

Data Compression:

Indeed, even with real advances over the previous decade in figuring force and capacity costs, despite everything it bodes well to keep your informational indexes as little and proficient as would be prudent. That implies just running calculations on fundamental information and not preparing on excessively. Unsupervised learning can help with that through a procedure called dimensionality decrease.

Principal Component Analysis (PCA) –
finds the linear combinations that communicate most of the variance in your data.

$$
\begin{aligned}
&\mathbf{R} \leftarrow \mathbf{X} \\
&\text{for}(k = 0, \ldots, K-1)\ \text{do} \\
&\quad \{ \\
&\quad \lambda = 0 \\
&\quad \mathbf{T}^{(k)} \leftarrow \mathbf{R}^{(k)} \\
&\quad \text{for}(j = 0, \ldots, J)\ \text{do} \\
&\quad\quad \{ \\
&\quad\quad \mathbf{P}^{(k)} \leftarrow \mathbf{R}^T \mathbf{T}^{(k)} \\
&\quad\quad \mathbf{P}^{(k)} \leftarrow \mathbf{P}^{(k)} \left\| \mathbf{P}^{(k)} \right\|^{-1} \\
&\quad\quad \mathbf{T}^{(k)} \leftarrow \mathbf{R}\mathbf{P}^{(k)} \\
&\quad\quad \lambda' \leftarrow \left\| \mathbf{T}^{(k)} \right\| \\
&\quad\quad \text{if}(|\lambda' - \lambda| \leq \varepsilon)\ \text{then break} \\
&\quad\quad \lambda \leftarrow \lambda' \\
&\quad\quad \} \\
&\quad \mathbf{R} \leftarrow \mathbf{R} - \mathbf{T}^{(k)} (\mathbf{P}^{(k)})^T \\
&\quad \} \\
&\text{return } \mathbf{T}, \mathbf{P}, \mathbf{R}
\end{aligned}
$$

Fig. 11:Pseudo code for Principal Component Analysis

Singular-Value Decomposition (SVD) – factorizes your data into the product of three other, smaller matrices.

References

[1] www.towardsdatascience.com
[2] Ayon Dey, "Machine Learning Algorithms: A Review",IJCSIT, ISSN:0975-9646,Vol.7,pages:117-122.
[3] https://blog.algorithmia.com/introduction-to-unsupervised-learning/
[4] https://www.displayr.com/what-is-hierarchical-clustering/
[5] L. Rokach, O. Maimon, "Top – Down Induction of Decision TreesClassifiers – A Survey", IEEE Transactions on Systems,
[6] D. Lowd, P. Domingos, "Naïve Bayes Models for Probability Estimation"
[7] https://webdocs.cs.ualberta.ca/~greiner/C 651/Homework2_Fall2008.html
[8] D. Meyer, "Support Vector Machines – The Interface to libsvm in package e1071", August 2015
[9] S. S. Shwartz, Y. Singer, N. Srebro, "Pegasos: Primal Estimated sub - Gradient Solver for SVM", Proceedings of the 24 th International Conference on Machine Learning, Corvallis, OR, 2007
[10] http://www.simplilearn.com/what-is-machine-learning-and-why-itmatters-article
[11] P. Harrington, "Machine Learning in action", Manning Publications Co., Shelter Island, New York, 2012