IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A SURVEY OF ISSUES AND CHALLENGES OF DEVELOPING SMART DEVICES USING MACHINE LEARNING ALGORITHMS

Parshva Jain[*1] & Rahul Vijayvargiya[2]
[*1&2]Information Technology, Medicaps, Indore (M.P.)-452001, India

DOI: 10.5281/zenodo.1042070

ABSTRACT

In the digital world everything is developing faster and powerfully. New technologies will be introducing by the IT expects to make the devices more user friendly and more intelligent by modifying the functionality in such manner, that the work of individuals will become cost effective and efficient. More complex problem can be easily tackled and solved. The algorithm which helps to developed intelligent devices that can handled the complex and unsolvable problem is known as machine learning algorithm. In this paper, we will discuss the issues and challenges that can be occur to developed smart devices and also we will study the different algorithms provided by machine learning.

KEYWORDS: Machine Learning, Machine Learning Algorithms, Smart Devices, Issues, Challenges.

I. INTRODUCTION

Over the past two decades Machine Learning has become one of the mainstays of information technology and with that, a rather central, albeit usually hidden, part of our life. With the ever increasing amounts of data becoming available there is good reason to believe that smart data analysis will become even more pervasive as a necessary ingredient for technological progress. The goal of machine learning is to program computers to use example data or past experience to solve a given problem. Many successful applications of machine learning exist already, including systems that analyze past sales data to predict customer behavior, optimize robot behavior so that a task can be completed using minimum resources, and extract knowledge from bioinformatics data.

Like many other areas of Artificial Intelligence, the technical capabilities of machine learning approaches are regularly oversold and this hype overshadows the real advances. Machine learning algorithms have become an increasingly important part of our lives. They are integral to all sorts of applications from the speech recognition technology in Siri to Google's search engine. Unfortunately machine learning systems are often more noticeable in our lives because of failures rather than successes. We come face-to-face with the limitations of auto-text recognition daily, while spam filtering algorithms quietly remove mass mail from our inboxes completely unnoticed. Improvements to machine learning algorithms are allowing us to do more sophisticated computational tasks. But it is often unclear exactly what these tools can do, their limitations and the implications of their use - especially in such a fast moving field.

II. ISSUES AND CHALLENGES

There are many potential benefits to using machine learning techniques and plenty of future applications but there are also a number of pitfalls and challenges which will impact if and how widely the benefits are felt. These are not technical challenges but the way in which the learning algorithm is deployed and the ultimate goal of its use.

1. Understanding assumptions for better outcomes

Even with good quality, unbiased data the conclusions drawn by different machine learning approaches can be widely different. Variations in approaches to generating a model can be enough to determine whether a significant correlation is seen or not. This variation arises in part because of the different assumptions algorithms make but no matter what machine learning algorithm is used, analysis of data will always require some difficult subjective decisions to be made about the best models or variables to use.

2. Machines as collaborators

Machine learning algorithms are rarely set up to give a reason for a particular decision or output. This perception of machine learning as an opaque decision–making tool instils a level of mistrust in its outputs. For the likes of physicians or policymakers it is important to have clear justifications for a decision, it is not good enough to rely on the supposed quality of the algorithm. This is particularly important when systems may be prone to errors or the decisions behind the choice of model is not known. People understandably place more trust in humans than machines but this reluctance to trust these learning systems is a big challenge in realizing their full potential.

3. Skills and understanding

If these techniques are to be utilized effectively we need to cultivate informed workers at all levels who can correctly deploy and interrogate the outcomes of algorithmic processes. The 'Crowd sourcing Analytics' experiment highlights the importance of having people who are able to understand the machine learning process and will not simply take its conclusions as correct. In many cases this will require a different kind of data scientist, one that does not have the core technical ability to write code but enough of a general understanding of what can and cannot be achieved using machine learning approaches to effectively evaluate its outputs.

4. Coding errors and quality assurance

Algorithm programming errors are not uncommon. In the majority of cases these bugs are only a nuisance and can be fixed easily after identification. The more we rely on learning systems for important tasks such as driving cars or medical diagnosis the outcomes of these errors might be far more serious. Progress is being made in the methods of software behavior 'verification' but it is particularly challenging to guarantee that a system built automatically via machine learning methods will behave properly in unpredictable real–world environments.

5. Speed of change and regulation

Maintaining an adequate understanding of capabilities and limitations in the long run will be challenging as the field of machine learning is advancing quickly. Ensuring regulation is able to stay up to date with these advances while not standing in the way of progress is only going to become more difficult. Both of these challenges are closely linked and require the co-ordinate efforts of business, academia, government and broader society to overcome them.

A deeper understanding of how to effectively use these systems for policy would be a hugely valuable asset, particularly for recognizing where regulation might be needed and what it should look like. The first steps will be to try and understand where regulation might be important, how the implementation of these techniques may impact society and how to deal with the rapid growth in capabilities. For driverless cars or high-frequency trading many of the potential pitfalls and dangers of using machine learning algorithms are well known, but creating frameworks to mitigate these challenges is not straightforward. It will require a great deal of thought and testing. More broadly there may be common issues such as verification of software behavior or responsibility that will need to be regulated but for many areas the potential importance of regulation is unclear. More focused discussions between academia, industry and government paired with more speculative foresight exercises would be an effective first step in addressing these challenges.

III. MACHINE LEARNING ALGORITHMS

1. Linear Regression

It is used to estimate real values (cost of houses, number of calls, total sales etc.) based on continuous variable(s). Here, we establish relationship between independent and dependent variables by fitting a best line. This best fit line is known as regression line and represented by a linear equation Y= a *X + b.

The best way to understand linear regression is to relive this experience of childhood. Let us say, you ask a child in fifth grade to arrange people in his class by increasing order of weight, without asking them their weights! What do you think the child will do? He / she would likely look (visually analyze) at the height and build of people and arrange them using a combination of these visible parameters. This is linear regression in real life! The child has actually figured out that height and build would be correlated to the weight by a relationship, which looks like the equation above.

In this equation:
Y – Dependent Variable
a – Slope
X – Independent variable
b – Intercept

These coefficients a and b are derived based on minimizing the sum of squared difference of distance between data points and regression line. Look at the below example. Here we have identified the best fit line having linear equation y=0.2811x+13.9. Now using this equation, we can find the weight, knowing the height of a person.

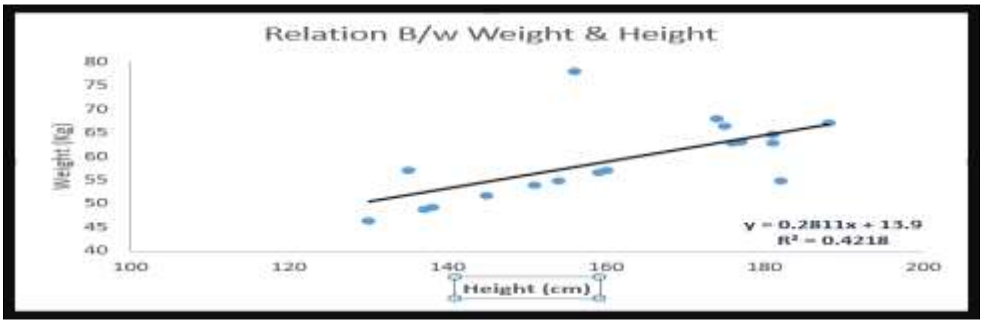


Figure 1: Relation Between Weight and Height

Linear Regression is of mainly two types: Simple Linear Regression and Multiple Linear Regression. Simple Linear Regression is characterized by one independent variable. And, Multiple Linear Regression(as the name suggests) is characterized by multiple (more than 1) independent variables. While finding best fit line, you can fit a polynomial or curvilinear regression. And these are known as polynomial or curvilinear regression.

2. Logistic Regression

Don't get confused by its name! It is a classification not a regression algorithm. It is used to estimate discrete values (Binary values like 0/1, yes/no, true/false) based on given set of independent variable(s). In simple words, it predicts the probability of occurrence of an event by fitting data to a logit function. Hence, it is also known as **logit regression**. Since, it predicts the probability, its output values lies between 0 and 1 (as expected).

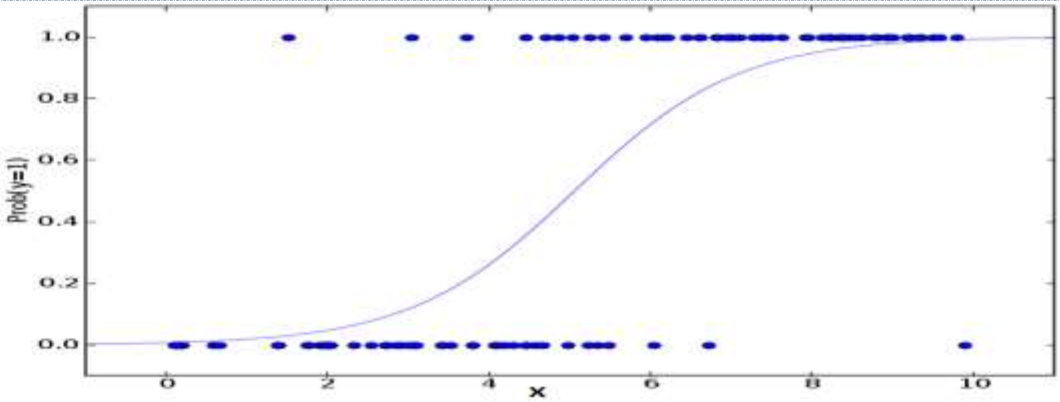
Let's say your friend gives you a puzzle to solve. There are only 2 outcome scenarios – either you solve it or you don't. Now imagine, that you are being given wide range of puzzles / quizzes in an attempt to understand which subjects you are good at. The outcome to this study would be something like this – if you are given a trigonometry based tenth grade problem, you are 70% likely to solve it. On the other hand, if it is grade fifth history question, the probability of getting an answer is only 30%. This is what Logistic Regression provides you.

Coming to the math, the log odds of the outcome is modeled as a linear combination of the predictor variables.
odds= p/ (1-p) = probability of event occurrence / probability of not event occurrence
ln(odds) = ln(p/(1-p))
logit(p) = ln(p/(1-p)) = b0+b1X1+b2X2+b3X3....+bkXk

Above, p is the probability of presence of the characteristic of interest. It chooses parameters that maximize the likelihood of observing the sample values rather than those minimize the sum of squared errors (like in ordinary regression).

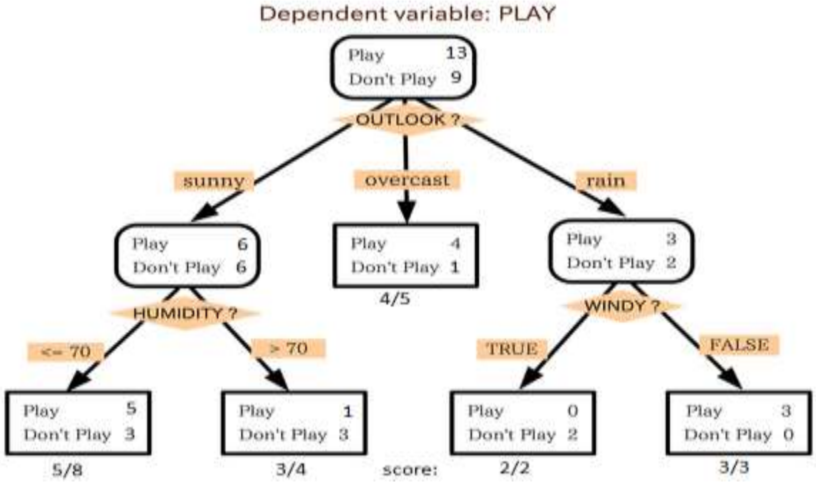
Now, you may ask, why take a log? For the sake of simplicity, let's just say that this is one of the best mathematical ways to replicate a step function. I can go in more details, but that will beat the purpose of this research paper.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7



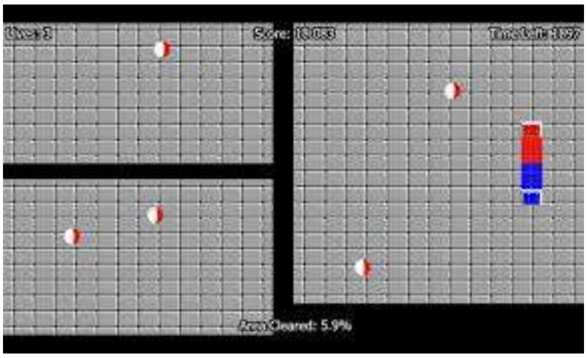
3. Decision Tree

This is one of my favorite algorithms and I use it quite frequently. It is a type of supervised learning algorithm that is mostly used for classification problems. Surprisingly, it works for both categorical and continuous dependent variables. In this algorithm, we split the population into two or more homogeneous sets. This is done based on most significant attributes/ independent variables to make as distinct groups as possible.



In the image above, you can see that population is classified into four different groups based on multiple attributes to identify ‘if they will play or not’. To split the population into different heterogeneous groups, it uses various techniques like Gini, Information Gain, Chi-square, entropy.

The best way to understand how decision tree works is to play Jazz ball – a classic game from Microsoft (image below). Essentially, you have a room with moving walls and you need to create walls such that maximum area gets cleared off without the balls.

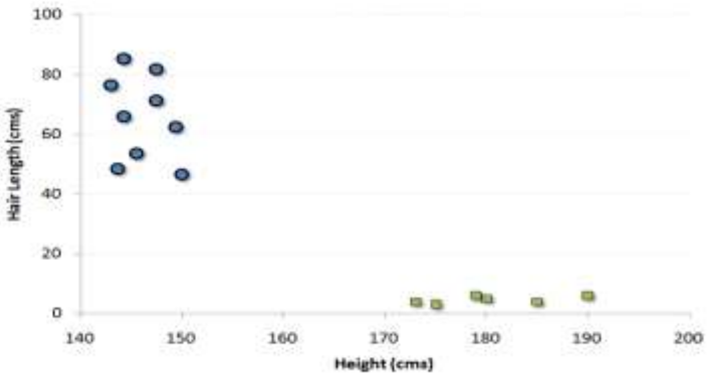


So, every time you split the room with a wall, you are trying to create 2 different populations with in the same room. Decision trees work in very similar fashion by dividing a population in as different groups as possible.

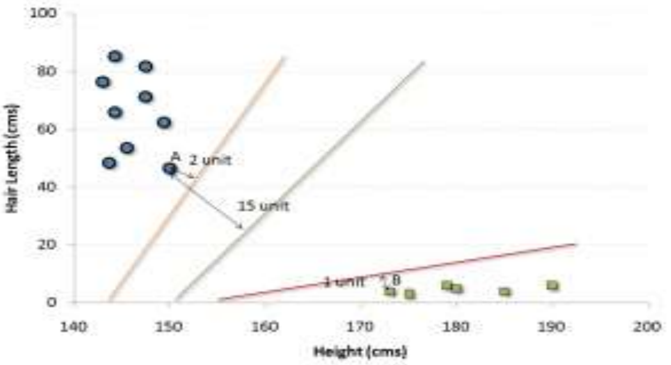
4. SVM (Support Vector Machine)

It is a classification method. 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.

For example, if we only had two features like Height and Hair length of an individual, we'd first plot these two variables in two dimensional space where each point has two co-ordinates (these co-ordinates are known as **Support Vectors**)



Now, we will find some *line* that splits the data between the two differently classified groups of data. This will be the line such that the distances from the closest point in each of the two groups will be farthest away.



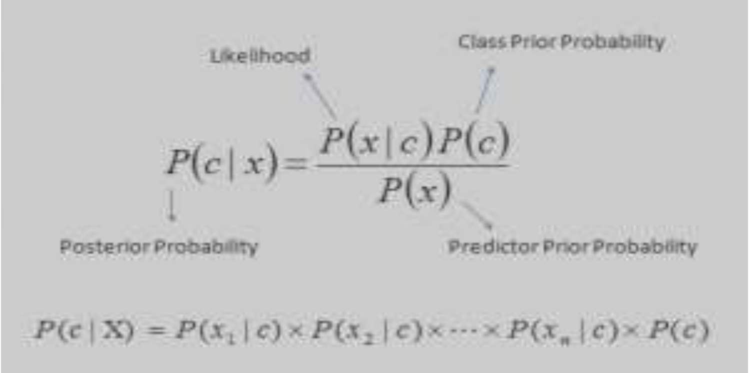
In the example shown above, the line which splits the data into two differently classified groups is the *black* line, since the two closest points are the farthest apart from the line. This line is our classifier. Then, depending on where the testing data lands on either side of the line, that's what class we can classify the new data as.

5. Naive Bayes

It is a classification technique based on Bayes' theorem with an assumption of independence between predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier would consider all of these properties to independently contribute to the probability that this fruit is an apple.

Naive Bayesian model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:



$$P(c\mid x)=\frac{P(x\mid c)P(c)}{P(x)}$$

$$P(c\mid X)=P(x_1\mid c)\times P(x_2\mid c)\times\cdots\times P(x_n\mid c)\times P(c)$$

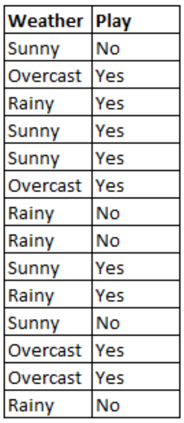
Here,

- *P*(*c*|*x*) is the posterior probability of *class* (*target*) given *predictor* (*attribute*).
- *P*(*c*) is the prior probability of *class*.
- *P*(*x*|*c*) is the likelihood which is the probability of *predictor* given *class*.
- *P*(*x*) is the prior probability of *predictor*.

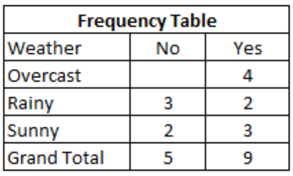
Example: Let's understand it using an example. Below I have a training data set of weather and corresponding target variable 'Play'. Now, we need to classify whether players will play or not based on weather condition. Let's follow the below steps to perform it.

Step 1: Convert the data set to frequency table

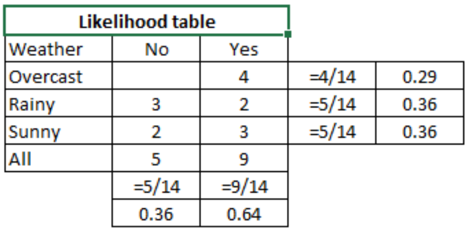
Step 2: Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.



Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No



Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9



Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

Step 3: Now, use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

Problem: Players will pay if weather is sunny, is this statement is correct?

We can solve it using above discussed method, so P(Yes | Sunny) = P(Sunny | Yes) * P(Yes) / P (Sunny)

Here we have P (Sunny |Yes) = 3/9 = 0.33, P(Sunny) = 5/14 = 0.36, P(Yes)= 9/14 = 0.64

Now, P (Yes | Sunny) = 0.33 * 0.64 / 0.36 = 0.60, which has higher probability.

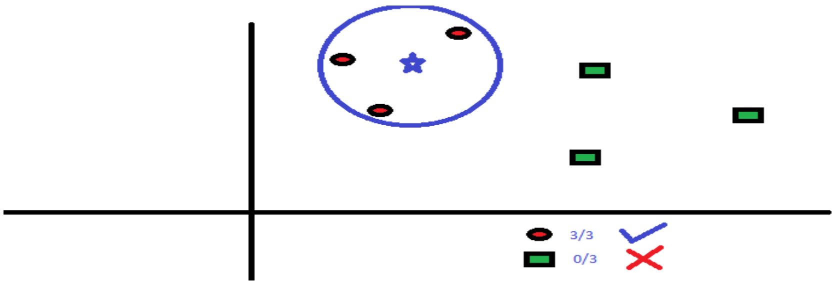
Naive Bayes uses a similar method to predict the probability of different class based on various attributes. This algorithm is mostly used in text classification and with problems having multiple classes.

6. KNN (K- Nearest Neighbors)

It can be used for both classification and regression problems. However, it is more widely used in classification problems in the industry. K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its k neighbors. The case being assigned to the class is most common amongst its K nearest neighbors measured by a distance function.

These distance functions can be Euclidean, Manhattan, Minkowski and Hamming distance. First three functions are used for continuous function and fourth one (Hamming) for categorical variables. If K = 1, then the case is simply assigned to the class of its nearest neighbor. At times, choosing K turns out to be a challenge while performing KNN modeling.

More: Introduction to k-nearest neighbors : Simplified.



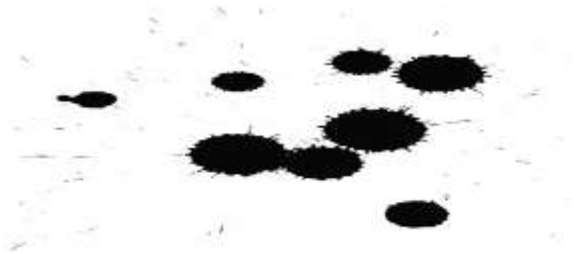
KNN can easily be mapped to our real lives. If you want to learn about a person, of whom you have no information, you might like to find out about his close friends and the circles he moves in and gain access to his/her information!

Things to consider before selecting KNN

- KNN is computationally expensive
- Variables should be normalized else higher range variables can bias it
- Works on pre-processing stage more before going for KNN like outlier, noise removal

7. K-Means

It is a type of unsupervised algorithm which solves the clustering problem. Its procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters). Data points inside a cluster are homogeneous and heterogeneous to peer groups.
Remember figuring out shapes from ink blots? k means is somewhat similar this activity. You look at the shape and spread to decipher how many different clusters / population are present!

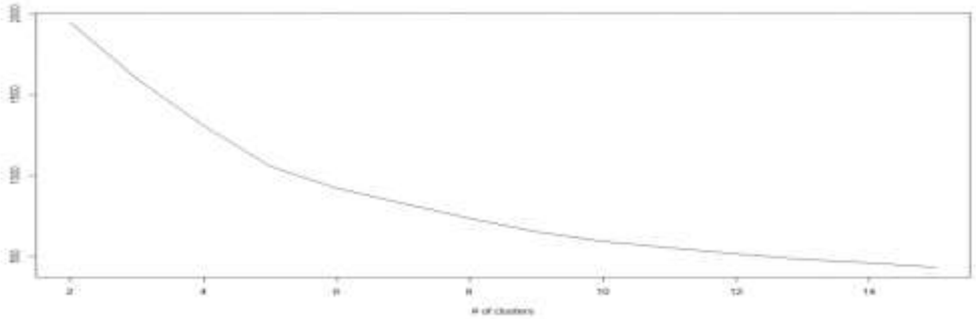


How K-means forms cluster:

1. K-means picks k number of points for each cluster known as centroids.
2. Each data point forms a cluster with the closest centroids i.e. k clusters.
3. Finds the centroid of each cluster based on existing cluster members. Here we have new centroids.
4. As we have new centroids, repeat step 2 and 3. Find the closest distance for each data point from new centroids and get associated with new k-clusters. Repeat this process until convergence occurs i.e. centroids does not change.

How to determine value of K:
In K-means, we have clusters and each cluster has its own centroid. Sum of square of difference between centroid and the data points within a cluster constitutes within sum of square value for that cluster. Also, when the sum of square values for all the clusters are added, it becomes total within sum of square value for the cluster solution.

We know that as the number of cluster increases, this value keeps on decreasing but if you plot the result you may see that the sum of squared distance decreases sharply up to some value of k, and then much more slowly after that. Here, we can find the optimum number of cluster.



8. Random Forest

Random Forest is a trademark term for an ensemble of decision trees. In Random Forest, we've collection of decision trees (so known as "Forest"). To classify a new object based on attributes, each tree gives a classification and we say the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest).

Each tree is planted & grown as follows:

1. If the number of cases in the training set is N, then sample of N cases is taken at random but *with replacement*. This sample will be the training set for growing the tree.
2. If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
3. Each tree is grown to the largest extent possible. There is no pruning.

IV. CONCLUSION AND FUTURE ENHANCEMENT

We propose an efficient use of machine learning algorithms and techniques which can be implemented on smart systems for their construction and effective usage. The supervised and unsupervised machine learning techniques are discussed in this paper. The main issues in smart systems using machine learning algorithms for fast processing is task scheduling. The various issues and challenges are discussed in the latter half of the survey. The paper also outlines the mechanisms and equations of ML algorithms.

Many investigators and stakeholders withdraw into their private studies with a copy of the data set and work in isolation to solidify algorithmic performance. Publishing results to the ML community is the end of the process. The worlds of finance, politics, education, medicine, law, and more stand to benefit from systems that can adapt, analyze, and take actions. This paper identifies the problems in constructing smart systems by implementing machine learning algorithms on it. Aiming for real impact does not just increase our job satisfaction, it is the only way to get the rest of the world to notice, recognize, value, and adopt ML solutions. The paper also outlines the possibilities and advanced enhancements of machine learning and techniques in fields of pattern recognition, image processing, text processing in nearby future

V. REFERENCES

[1] Sally Goldman; Yan Zhou, "Enhancing Supervised Learning with Unlabeled Data", Department of Computer Science, Washington University, St.Louis, MO 63130 USA.
[2] Y. Bengio. Learning deep architectures for AI. Foundations and Trends in Machine Learning, 2:1–127, 2009

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

[3] Rich Caruana; Alexandru Niculescu- Mizil,"An Empirical Comparison of Supervised Learning Algorithms", Department of Computer Science, Cornell University, Ithaca, NY 14853 USA.

[4] Niklas Lavesson,"Evaluation and Analysis of Supervised Learning Algorithms and Classifiers", Blekinge Institute of Technology Licentiate Dissertation Series No 2006:04,ISSN 1650-2140,ISBN 91-7295-083-8.

[5] Types of Machine Learning Algorithms, Taiwo Oladipupo Ayodele, University of Portsmouth, United Kingdom.

[6] V. N. Vapnik. The Nature of Statistical Learning Theory. Springer, New York, NY, 1995.

[7] I. Witten, E. Frank, and M. Hall. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Mateo, CA, 3rd edition, 2011

[8] Warrick, P. A., Hamilton, E. F., Kearney, R. E., and Precup, D. A machine learning approach to the detection of fetal hypoxia during labor and delivery. In Proc. of the Twenty-Second Innovative Applications of Artificial Intelligence Conf., pp. 1865–1870, 2010

[9] https://en.wikipedia.org/wiki/Carbonell.

CITE AN ARTICLE

Jain, P., & Vijayvargiya, R. (2017). A SURVEY OF ISSUES AND CHALLENGES OF DEVELOPING SMART DEVICES USING MACHINE LEARNING ALGORITHMS. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, 6*(11), 22-30.