JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-7, Issue-3, Mar 2019 E-ISSN: 2347-2693

A Deep Analysis and Efficient Implementation of Supervised Machine Learning Algorithms for Enhancing The Classification Ability of System

Sandeep Kumar Verma[1*], Turendar Sahu[2], Manjit Jaiswal[3]

[1,2,3]Department of Computer Science and Engineering, Institute of Technology, Guru Ghasidas University, Bilaspur, India

[*]*Corresponding Author: skv007bond@gmail.com, Tel.: +91-70897-18773*

DOI: https://doi.org/10.26438/ijcse/v7i3.10941101 | Available online at: www.ijcseonline.org

Accepted: 19/Mar/2019, Published: 31/Mar/2019

***Abstract*—** Machine Learning, a subset of Artificial Intelligence is very popular and emerging field of science which basically focus on designing the approach enabling the computer or machines to learn from the input provided or past experience. There are lots of application of Machine Learning in day-to-day life such as face detection and recognition, decision making in business forecasting, etc. It is also becoming the business subject to various giant enterprises like Amazon, Google, FaceBook, etc. In this paper, we have focused our discussion to some popular Supervised Machine Learning algorithms that are SVM, logistic regression, Multinomial Naive Bayes, KNN apart from some other supervised Machine Learning algorithms like Linear Regression, Linear Discriminant Analysis, Decision Tree, Random Forest, Naïve Bayes, etc. and we determine the most efficient classification algorithm based on the data set which is multiclass dataset. This research paper gives some clarity to the selection of algorithm specific to some application. And we have shown the comparative results.

***Keywords*—**Machine Learning (ML), K-nearest neighbours (KNN), Logistic Regression (Log), Multinomial Naïve Bayes (MulNB), Support Vector Machine (SVM), Classifier

I. INTRODUCTION

Machine Learning (ML) is a field of computer science that evolved from study of pattern recognition and computational learning theory in artificial intelligence (AI).

It is the learning and building of algorithms that can learn and improve themselves from experience of past data and make predictions on data sets [1]. Today, Machine learning is pervasive in business industry science government in solving Worlds important tasks. It is a technology in commercial, medical, financial and scientific applications. Machine Learning is closely related to statistical computing, Natural Language Processing (NLP), Pattern Recognition which focuses on making decisions using computers. As in [2], every instance of dataset, which are used in machine learning algorithm is represented using same set of features. The nature of features of the dataset could be continuous, categorical or binary. In the supervised learning input variables as well as output variables all are known. It is based on the comparison of computed output and expected output [3]. The goal is approximate the mapping functions such that when you have new input data to predict the output variables for the data. Supervised machine learning algorithms are used to solve classification or regression problems. Unsupervised learning algorithms take input and learn some features from dataset when new input data it uses previously learned features to predict the class of the data [4]. This algorithm is used for clustering and feature reduction. The main goal of this algorithm is to learn more about the data find structures in data.

Many real-time problems (Natural Language Processing, Pattern Recognition, classification problem, etc.) involve task that need to be handle in supervised manner. In section 2, we have mainly concentrated on some of the popular supervised techniques like k-nearest neighbours (KNN), logistic regression, multinomial Naive Bayes, support vector machines (SVMs). Section 3, includes implementation and comparisons of supervised machine learning techniques SVM, MulNB, Log and KNN, graphical and tabular comparisons of training time, training accuracy, testing time and testing accuracy on IRIS dataset. Section 4, contains result and analysis of SVM, MulNB, Log and KNN supervised ML algorithms. Section 5 includes conclusions of whole discussion about supervised ML algorithm.

II. RELATED WORK

We have focused our discussion on four algorithms: KNN, Log, MulNB, SVM.

2.1 K-nearest neighbours (KNN)

K-nearest neighbours (KNN) is most fundamental and highly efficient classification algorithm which falls under supervised learning, due to its features likes easy to use, requires low training time, robust to noisy training data, easy

to implement. But this algorithm has some shortcomings as computational complexity, large memory requirement for large dataset, 'curse of dimensionality' can't be ignored and it also gives the equal weights to all attributes or features of dataset [5]. This supervised machine learning algorithm can be used to solve both regression and classification problems.
A classification problem has a categorical output, such as "blue" or "red", "disease" or "no disease" or "dog" or "cat". A classification problem takes some conclusion from observed values. For given one or more inputs a classification model will try to predict the value of one or more outcomes.

A regression problem has a real or contentious output value, such as "salary" or "weight". There are many different techniques apart from KNN, the simplest one is the linear regression. Linear regression tries to fit the data with the best hyper plane which goes through the data points.

Algorithm

K-nearest neighbour (KNN) is based on the principle that the instance of the dataset will generally exist in close impendency to other instances that have similar properties.
Figure 1 shows the pseudo code example for instance based learning method.

```
Procedure InstanceBaseLearner (Testing Instances)
        For each testing instance
        {
Find the k most nearest instance of the training set according to a distance metric
Resulting class = most frequent class label of the k nearest instances
}
```

Figure 1: Pseudo code for instance based learners [2]

Instances can be considered as data points in an n-dimensional instance space or feature space where each of the n dimensions is used to describe one of the 'n' features of the instances or data points. The relative distance of the instances can be determine using the distance metric. The distance metric between two instances must be minimize the distance between two similar classified instances.
The distance metric must minimize the distance between two similar classified instances, while maximizing the distance between instances of different classes. There are many different metric have been presented, some of distance functions are:
To measure the distance in feature space between two points A and B, the **Euclidean distance function** is most widely used.
The **normalized Euclidean metric** between A and B is given by

$$distance(A,B) = \sqrt{\frac{\sum_{i=1}^{m}(x_i - y_i)^2}{m}}$$

Where,

A = (X_1, X_2, X_3, …, X_m), B = (Y_1, Y_2, Y_3,……Y_m) and **m** is the dimensionality of feature space [6].

Manhattan Distance

It is the distance travelled to get from one data point to the other data point [7], and this distance measured along axes at right angles. The name indicates the Manhattan street in greed layout which gives the shortest path between two points. The equation for distance between two points is (X, Y) is given by

$$\text{distance}(\text{X}, \text{Y}) = \sum_{i=1}^{m} |X_i - Y_i|$$

Where,

X = (X_1, X_2,…,X_m), Y = (Y_1,Y_2,…,Y_m) and **m** is the dimensional space.
For example, filtering emails "spam" or "not spam", Classification either predicts classified data or categorical class label based on the training data set and values in classifying attributes and uses in classifying new data. There are various classification models. Some of the classification models are logistic regression, decision tree, random forest, multilayer perceptron and Naïve Bayes.

2.2 Logistic Regression (Log)

Logistic regression also called as Sigmoid Function, is the appropriate regression analysis to conduct when dependent variable is binary or dichotomous i.e. Positive or Negative Class. It is the predictive modelling technique. It estimate the relationship between a dependent variable (target) and an independent variable (predictor). This method is used when dependent variables are binary, tertiary, ternary and quaternary. It is better than statistical regression because simple and multiple regression analyses are used in examine mathematical correlation between expressive variable(s) and dependent variable and for implementing simple and multiple regression analyses in data sets, the dependent variable should follow normal distribution and should be singled from community(es) that shows a normal distribution of parametric type and if such conditions do not met in data sets these regression analyses will not be implemented but logistic regression can handle this situation because there no need of precondition such as a variance, normal distribution or continuous variance. The logistic Regression produces results in a binary (or dichotomous) format which is used to predict the outcome of a categorical dependent variable. So the outcome should be discrete/ categorical such as: True or False, Yes or No, Positive or Negative, High or Low, etc.
The effects of expressive variables are estimated as probability on the dependent variables. According to the type of dependent variable there are three methods of logistic regression analysis[8]:

- Binomial logistic regression (BLOGREG)

This logistic regression method is used when the result variable is in binary form like Yes or No, True or False, High or Low, etc.

- Ordinal logistic regression (OLOGREG)
  This method is used when the result variable is ordinal (i.e. result variable should include values that are observed at least in three categories).These categories are in natural order.
- Nominal logistic regression (NLOGREG)
  It is used when the result variable is nominal (i.e. result variable should include values that are observed at least in three categories). These categories do not have to be in order.

The underlying mathematical concept of logistic regression is the logit which is the natural logarithm of an odds ratio.

2.2.1 Odds

To understand the odds we will think it as the ratio of probability of an event will occur to the probability that the event will not occur.

$$\text{Odd of}\{\text{Event}\} = \frac{\text{p}}{1-\text{p}}$$

Where p is the probability of an event occurring and $(1-\text{p})$ is the probability of the event not occurring.

Since logistic regression estimate the probability of an event occurring over the probability of an event not occurring that is ratio of these, the effect of the independent variable is expressed in terms of odds. With logistic regression p and x (where p is response variable and x is explanatory variable) are modelled by the equation $\text{p} = \alpha + \beta\chi$. There is a problem with this equation which is if the value of χ is very high then the value of p will not fall in between 0 and 1. To solve this issue we take the natural logarithm of odds [9]. Now we model the natural logarithm odds s a linear function of the explanatory variable:

$$\text{logit}(\text{y}) = \ln(\text{odds}) = \ln\left(\frac{\text{p}}{1-\text{p}}\right) = \alpha + \beta\chi$$

Where p is probability of interested outcome, χ is the explanatory variable and α and β are logistic regression parameters.

To get the desired outcome, we take antilog and finally obtained :

$$p = \frac{1}{1 + e^{-(\alpha+\beta_1\chi_1+\,\dots\,+\,\beta_k\chi_k)}}$$

2.2.2 Odds ratio

In [9][10], an odds ratio (OR) is a measurement of association between an exposure and an outcome. Let us consider two events A and B which independent to each other, the corresponding measuring of odds of A occurring relative to B occurring is given by

$$\text{odds ratio}\{\text{A vs B}\} = \frac{\text{odds(A)}}{\text{odds(B)}} = \frac{\text{P}_\text{A}/(1-\text{P}_\text{A})}{\text{P}_\text{B}/(1-\text{P}_\text{B})}$$

Odds ratios are generally used in case-control studies, however they can also be used in cross-sectional and cohort study designs as well (with some correction and/or assumptions). Other probability based classification is Multinomial Naïve Bayes which sometimes gives better performance than logistic regression in some specific applications like Text Classification, etc.

2.3 Multinomial Naïve Bayes (MulNB)

Multinomial Naïve Bayes is very useful supervised machine learning technique which is variant of Naïve Bayes technique [11]. Naïve Bayes is also a simple classification technique which is based on Bayesian theorem [12].Similar to simple Naïve Bayes technique Multinomial Naïve Bayes is mostly used in text classification problems. It is used in many real-time applications such as filtering email as 'spam' or 'non spam', speech recognition where it is called 'unigram language model' [13], language detection and sentiment detection [14], document classification based on defined vocabulary, sorting emails into various folders, documents filtering, topic identification, etc. Generally MulNB is robust even when input statements are dishonoured.

It is based on probabilistic technique. The way to implement this model or classifier are as follows [15]:

1. Discriminating model: Main focus in this model is to obtain the function which calculate the class posterior p(y|x). As name suggest it discriminates among different classes given the input.
2. Generative Model: Main focus in this model is to obtain class conditional density p(x|y) for each value of y, and also learn class priors p(y) and then by applying Bayes formula to obtain posterior:

$$p(y|x) = \frac{p(x|y).\,p(y)}{p(x)}$$

As we've mentioned above, that Multinomial Naïve Bayes is based on Naïve Bayes Classifier. There are two different generative models commonly used in text classification. First model represents the document by a vector in binary attributes indicating whether the particular word has occurred in the statements provided or do not occur in the document. Therefore, filling the vector either by True i.e. 1 if the word occurs in the document or False i.e. 0 if the word do not occur in the document, and the second model represents the document by the set of word occurrences i.e. it counts the number of times a word occurs in the documents.

The working of the MulNB can be simply understand by the following steps:

1) Prob(C|D) = Prob(D|C) Prob(C)/Prob(D).
2) Let T_1, T_2 … T_m be the sequence of lexical terms in D. Assume that the occurrence of a term T in the i^{th} place of word. And D depends only on the category C and given C, is conditionally independent of all the other terms in D and of the position i.

Therefore **Prob (D|C) = Prob(T_1|C) × Prob(T_2|C) × … ×Prob(T_m|C).**

Where Prob(Ti|C) means the probability of term being in C.

3) Estimate Prob(Ti|C) and Prob(C) using the training set. Prob (Ti|C) is estimated as the relative frequency of Ti in document of category C (number of occurrences of T in C)/(total number of words in C).

Prob (C) is estimated as fraction of total number of occurrence of terms and total terms.

4) Prob(D) is independent of the category . Calculate the **product,**

Prob(Ci) × Prob(T1|Ci) × Prob(T2|Ci) × … × Prob(Tm|Ci) for each category Ci and choose the category. Finally, it maximizes product.

Figure 2: Pseudo code Multinomial Naïve Bayes [14]

2.4 Support Vector Machines (SVMs)

Support vector machines (SVMs), is a new and prevalent object (or data) classification approach emerged from Statistical Learning Theory (STL) [16][18] which truly fall under Supervised Machine Learning but some of SVMs are used in semi supervised machine learning [17]. It has also proved efficient and effective technique which is used in Data Mining (DM). There are various fields implementing SVMs for their own benefits [16][17][18]. Some of them are Pattern Recognition, fraud detection in credit card transactions, face recognition, Text classification [19], speech recognition, etc.; Initially SVMs were used for binary classification (Positive and Negative) of data points. SVMs had gone through so many modifications and after so many modifications standard version of SVMs with soft-margin called as C-SVM was published [16]. Now a days SVMs are able to distinguish among various class objects. For better solution, SVMs optimize 'margin' between various data points of different classes in feature space (which may be of high dimension. There are so many modifications made in SVM to improve the solutions for various types of problems such as convex optimization problems like linear programming, convex quadratic programming, second order cone programming, semi-definite programming , bi-level programming, etc., non-convex problems [20] and general optimization problems and many modified SVMs such as v-SVM, linear programming SVM, least squares SVM, proximal SVM, twin SVM, multi-kernel SVM, AUC maximizing SVM, localized SVM, cost sensitive SVM, fuzzy SVM, Crammer-Singer SVM, K-support vector classification regression (K-SVCR), Transductive SVM (TSVM) etc. But we will limit our discussion to the standard SVM which is C-SVM. There are three important key point in the working of SVM

i. Separating Hyper Plane
ii. principle of maximum margin
iii. dual theory
iv. kernel function

Suppose we have training dataset

$$T = \{(x_1, y_1), (x_2, y_2), (x_3, y_3),\ldots,(x_n, y_n)\} \in (R^d \times Y)^n,$$

Where $x_i \in R^d$, $y_i \in Y = \{1, -1\}$, i = 1,2,3,…,n. These data points are mapped in feature space $R^d \times Y$ similarly Figure 1.

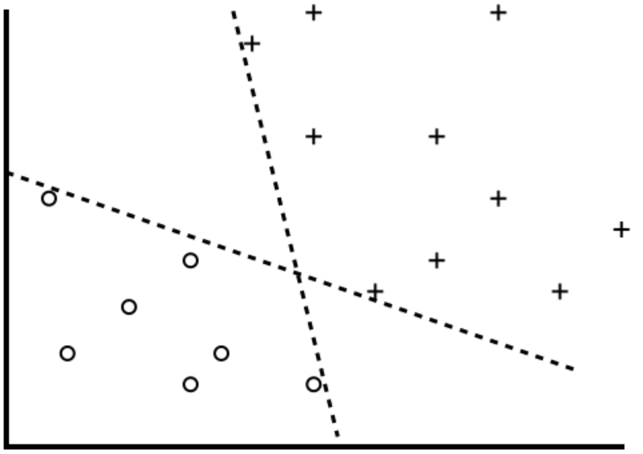

Figure 3: Possible Hyperplanes

Suppose we have various data points (x_i, y_i) belonging to two different class (**P**ositive and **N**egative) in the feature space as shown in Figure 1. It is possible to draw various hyperplanes with the help of some data points called support vectors, segregating (distinguishing) both class in feature space.

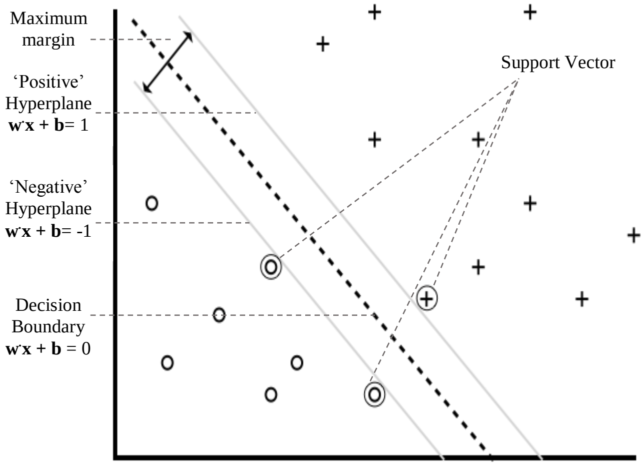

Figure 4: Maximum Margin Hyperplane

If we train our model with any of the hyperplane possible then it is possible that we will end up with inefficient model, predicting accurate class for some data points but wrong for most of the data points. So that we have to consider the principle of maximum margin as we have discussed above. For better solution, maximum margin hyperplane (i.e. optimal hyperplane) is selected during the training of the model as shown in Figure 2.

The following equation,

$$w.x + b = 0,$$

represents a class of hyperplanes from which two parallel hyperplanes are selected which is having maximum margin between them as shown in Figure 2. It is easy to find the optimal hyperplane, when possible to linearly separate two classes by minimizing the square norm of separating hyperplane. The minimization can be found as a convex quadratic programming (QP) problem:

$$\underset{w,b}{\text{Minimize}}\, \Phi(w) = \frac{1}{2}\|w\|^2 \qquad (1)$$

Once the optimal hyperplane is found, data points in feature space that lie on the boundry of the hyperplane is termed as support vector and the optimal hyperplane is obtained by these support vectors.

Algorithm

```
1) Introduce positive Lagrange multipliers, one for each of the inequality constraints (1). This gives Lagrangian:
   $L_P \equiv \frac{1}{2}\|W\|^2 - \sum_{i=1}^{N} \alpha_i y_i (x_i \cdot w - b) + \sum_{i=1}^{N} \alpha_i$
2) Minimize LP with respect to w,
       b. This is a convex quadratic programming problem.
3) In the solution, those points for which $\alpha_i > 0$ are called "support vectors".
```

Figure 5: pseudo code for SVM [2]

III. Implementation And Comparision

3.1 Hardware Specification

We have done our analysis on the following architecture:

Table 1: Hardware Specification

S. N.	Specification type	Description
1.	Manufacturer	Apple
2.	Processor	dual-core Intel Core i5 processor with 3MB shared L3 cache
3.	Processor speed	2.9 GHz
4.	Memory	1866MHz LPDDR3 onboard memory
5.	Memory Capacity	8GB
6.	Storage Media type	SSD (Solid State Drive), PCIe-based flash storage
7.	Storage Capacity	512GB

3.2 Software Specification

3.2.1 Operating System: macOS Mojave ver. 10.14.3

3.2.2 IDE(Integrated Development Environment): Jupyter Notebook (ver. 5.7.0)

3.2.3 Programming Language: Python (ver. 3.0)

3.2.4 Dataset [21][22]:

We have used the following dataset in our analysis

Name: IRIS

No. of objects = 150

Total no. of Features/Input attributes = 4 (sepal length in cm, width in cm, petal length in cm and petal width in cm)

No. of target Classes = 3 (0,1 and 2 for setosa, versicolor and virginica)

Type of Dataset = Numerical only.

3.3 Comparison

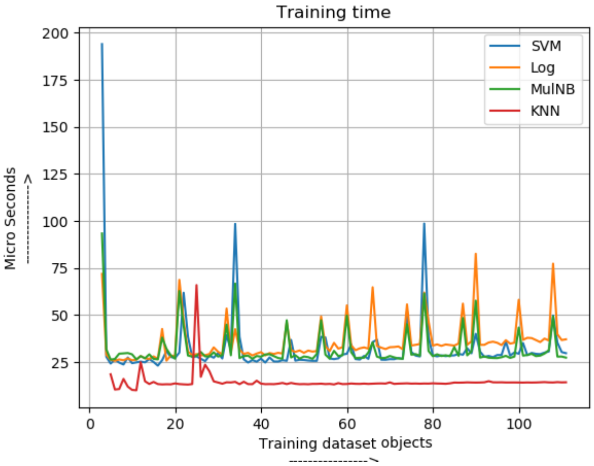

Figure 6: Comparison of SVM, Log, MulNB, KNN based on training time

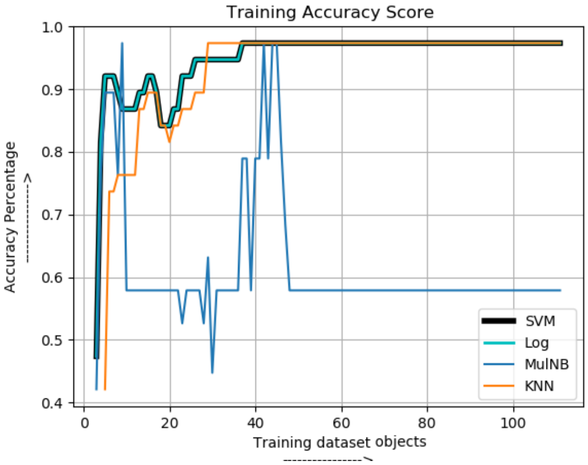

Figure 7: Comparison of SVM, Log, MulNB, KNN based on accuracy

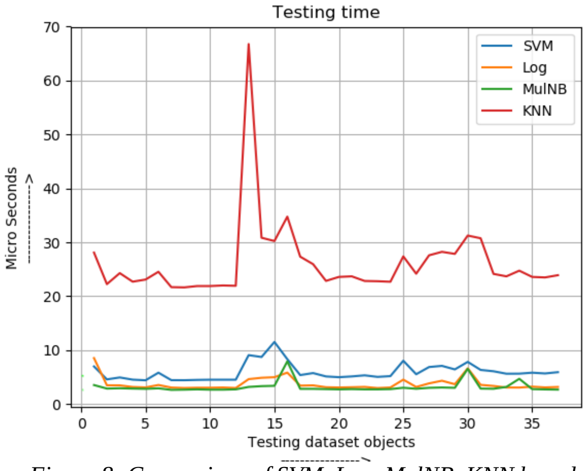

Figure 8: Comparison of SVM, Log, MulNB, KNN based on accuracy

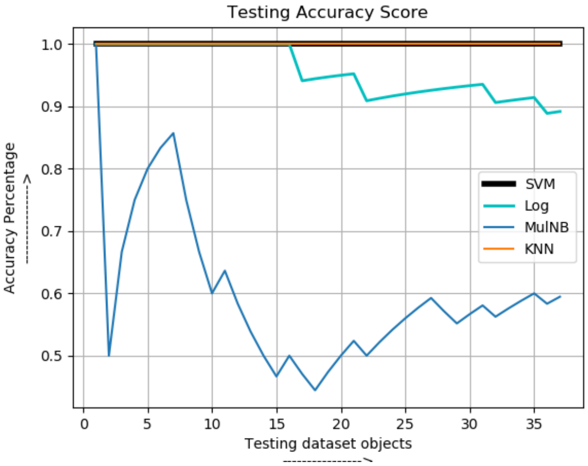

Figure 9: Comparison of SVM, Log, MulNB, KNN based on accuracy

Table 2: Training of various algorithms in micro seconds

No. of Object	SVM	Log	MulNB	KNN
5	25	27	28	18
20	33	33	32	13
40	30	35	33	13
60	36	46	44	13
80	33	35	33	13
100	34	50	38	14

Table 3: Training accuracy of various algorithm in Percentage

No. of Object	SVM	Log	MulNB	KNN
5	92.5	92.5	89	50
20	84	84	57.5	81
40	97.5	97.5	78	97.5
60	97.5	97.5	58	97.5
80	97.5	97.5	58	97.5
100	97.5	97.5	58	97.5

Table 4: Testing Time of various algorithm in micro seconds

No. of Object	SVM	Log	MulNB	KNN
5	4	3.5	3	23
10	4	3.5	3	22
15	11	5	3.75	30
20	3	3.5	2.5	23.5
30	8	7	7	31
35	5.5	3	3	23

Table 5: Testing Accuracy of various algorithm in Percentage

No. of Object	SVM	Log	MulNB	KNN
5	100	100	80	100
10	100	100	60	100
15	100	100	48	100
20	100	95	50	100
30	100	94	57	100
35	100	92	60	100

IV. RESULTS AND ANALYSIS

According to our analysis as shown in Table 2 if the no. of objects are 20 then the training time of SVM is 33, Log is 33, MulNB is 32 and KNN is 13. If the no. of objects are 60 then the training time of SVM is 36, Log is 46, MulNB is 44, KNN is 13 and if the no. of object is 100 then the training time of SVM, Log, MulNB, KNN is 34, 50, 38 and 14 respectively. From these premises we can say that KNN takes least time to train its classifier after this SVM takes less time to be trained. MulNB takes more time than SVM but less time than Log to be trained and Log takes highest time among all the mentioned methods. So, the order of training time can be represented as:

KNN < SVM < MulNB < Log

In Table 3, the training accuracy of various methods have been mentioned. The training accuracy means that how many object hane been used from the training data set to train the various classifiers. If no of objects are 20 then training accuracy time of SVM, Log, MulNB and KNN are 84, 84, 57.5 and 81 respectively. if no of objects are 60 then testing time of SVM, Log, MulNB and KNN are 97.5, 97.5, 58 and 97.5 respectively. if no of objects are 100 then testing time of SVM, Log, MulNB and KNN are 97.5, 97.5, 58 and 97.5 respectively. Hence, we can specify the order of training accuracy as:

KNN = SVM = Log > MulNB

In Table 4, if no. of objects are 10 then testing time of SVM, Log, MulNB and KNN are 4, 3.5, 3 and 22 respectively. If no. of objects are 20 then testing time of SVM, Log, MulNB and KNN are 3, 3.5, 2.5 and 23.5 respectively. if no. of objects are 35 then testing time of SVM, Log, MulNB and

KNN are 5.5, 3, 3 and 23 respectively. From above discussion we can say that Log and MulNB are better than SVM and KNN for testing time, KNN takes more testing time than other algorithm. So, the order can be considered as:

MulNB = Log < SVM < KNN

In Table 5 the testing accuracy of various methods have been mentioned. Testing accuracy means that the percentage of the accurate classified data objects from the given dataset to test the model. If we observe the table, we can see that if the no. of objects is 10 then percentage testing accuracy of SVM, Log, MulNB and KNN are 100, 100, 60 and 100 respectively. If the no. of objects is 20 then percentage testing accuracy of SVM, Log, MulNB and KNN are 100, 95, 50 and 100 respectively. And if the no. of objects is 35 then percentage testing accuracy SVM, Log, MulNB and KNN are 100, 92, 60 and 100 respectively. So, we can represent testing accuracy of various methods as:

SVM = KNN > Log > MulNB

Here SVM and KNN shows best testing accuracy.

V. CONCLUSION

There are various methods in supervised machine learning. It is very difficult to say that whether a particular method is better than other method. Every method works better in some specific type of application. The nature of problem domain affects the selection of ML algorithm. We have discussed SVM, logistic regression, Multinomial Naïve Bayes and KNN. The overall discussion reveals that in terms of training time, training accuracy, testing time and testing accuracy, the best algorithm is SVM. As the mentioned algorithms are supervised these are not able to handle the problems like decision making, clustering (like grouping of different animals), etc.

We have used default value for various parameters like the n_neighbors value is 5 in the KNN algorithms which is used for specifying number of candidates in voting, C (inverse of regularization which is used for handling the overfitting of classifiers) value is 1.0 in Logistic Regression and SVM. We have used numerical type dataset having only four different feature. If we deal with curse of dimensionality (i.e. dataset having large number of features) then the performance of these algorithms also decreases.

These algorithms can be modified to increase their performance by using Grid SearchCV. We can change the parameters of various algorithms to obtain better performance. We are trying to implement these algorithms in other fields like Deep learning and neural network to enhance their performance.

REFERENCES

[1] A. Simon, M. S. Deo, S. Venkatesan, D.R. Ramesh Babu, "*An Overview of Machine Learning and its Applications*", International Journal of Electrical Sciences and Engineering, **2015**.

[2] S. B. Kotsiantis, "*Supervised Machine Learning: A Review of Classification Techniques*", Informatica **31** (2007), pp.**249-268**, **2007**.

[3] S. Das, A. Dey, A. Pal, N. Roy, "*Applications of Artificial Intelligence in Machine Learning: Review and Prospect*", International Journal of Computer Applications (0975 – 8887), Vol. **115**, No. **9**, **2015**.

[4] A. Dey, "*Machine Learning Algorithms: A Review*", International Journal of Computer Science and Information Technologies, Vol. **7 (3)**, pp.**1174-1179**, **2016**.

[5] P. Rani, "*A Review of various KNN Techniques*", International Journal for Research in Applied Science & Engineering Technology, Vol. **5**, Issue **8**, **2017**.

[6] Li-Yu Hu, Min-Wei Huang, Shih-Wen Ke, and Chih-Fong Tsai "*The distance function effect on k-nearest neighbor classification for medical datasets*", Springerplus, Vol. **5(1)**, **2016**.

[7] Chich-Min Ma, Wei-Shui Yang and Bor-wen Cheng, "*How the parameter of k-nearest neighbours Algorithm impact on the Best Classification Accuracy: In case of Parkinson Dataset*", Journal of applied sciences, Vol. **14 (2)**, pp.**171-176**,**2014**.

[8] Murat KORKMAZ, Selami GÜNEY, Şule Yüksel YİĞÎTER, "*The importance of logistic regression implementations in the turkish livestock sector and logistic regression implementations/fields*", J.Agric. Fac. HR.U., **2012**.

[9] C-Y J. Peng, K. L. Lee, G. M. Ingersoll, "*An introduction to logistic regression analysis and reporting*". The Journal of Educational Research, **96(1)**, pp.**3-14**, **2002**.

[10] M. Szumilas, "*Explaining Odds Ratios*", Journal of the Canadian Academy of Child and Adolescent Psychiatry,Vol. **19(3)**, **2010**.

[11] A. M. Kibriya, E. Frank, B. Pfahringer, G. Holmes, "*Multinomial Naive Bayes for Text Categorization Revisited*", G.I. Webb and Xinghuo Yu (Eds.): AI 2004, LNAI **3339**, Springer-Verlag Berlin Heidelberg, pp. **488–499**, **2004**.

[12] D. Mallampati, "*An Efficient Spam Filtering using Supervised Machine Learning Techniques*", International Journal of Scientific Research in Computer Science and Engineering, Vol. **6**, Issue **2**, pp.**33-37**, **2018**.

[13] A. McCallum, K. Nigam, "*A Comparison of Event Model for Naïve Bayes Text Classification*", AAAI Technical Report WS-98-05, 752, **1998**.

[14] R. Mohana, S. Sumathi, "*Document classification using Multinomial Naïve Bayesian classifier*", International Journal of Science, Engineering and Technology Research, Vol. **3**, Issue **5**, **2014**.

[15] H. Doshi, M. Zalte, "*Performance of Naïve Bayes Classifier – Multinomial Model on Different Categories of Documents*", International Journal of Computer Applications, ETCSIT, **2011**.

[16] Y. Tian, Y. Shi, X. Liu, "*Recent advances on support vector machines research*", Technological And Economic Development OF Economy, Vol. **18 (1)**, pp.**5–33**, **2012**.

[17] V. Sindhwani, S. S. Keerthi, "*Large scale semi-supervised linear SVMs*", Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, **477-484, 2006**.

[18] M. A. Hearst, S.T. Dumais, E. Osman, J. C. Platt, B. Schölkopf, "*Support vector machines*", Intelligent Systems and their Applications, IEEE, **1998**.

[19] G. Kaur, K. Kaur, "*Sentiment Detection from Punjabi Text using Support Vector Machine*", International Journal of Scientific

Research in Computer Science and Engineering, Vol. **5**, Issue **6**, pp.**39-46**, **2017**.

[20] Y. E. M. Idris, Li Jun, "*Single to multiple kernel learning with four popular SVM kernels (survey)*", International Journal of Research in Engineering and Technology, Vol. **5**, Issue **3**, **2016**.

[21] R. A. FISHER, "*The use of multiple measurements in taxonomic problems*", Annual Eugenics, Vol. **7**, Part **2**, pp.**179-188**, **1936**.

[22] E. Anderson. "*The species problem in Iris". Annals of the Missouri Botanical Garden*", Vol. **23 (3)**, pp.**457–509**, **1936**.

Authors Profile

Mr. Sandeep Kumar Verma is currently pursuing his Bachelor's degree in Computer Science and Engineering from Institute of Technology, Guru Ghasidas Vishwavidyalaya A Central University, Bilaspur, Chhattisgarh, 495009, India. He has qualified GATE in 2019. His key research field is Grid Computing. He is currently working on Artificial Intelligence, IOT, python, java, Grid Computing, Algorithms.

Mr. Turendar Sahu is currently pursuing his Bachelor's degree in Computer Science and Engineering from Institute of Technology, Guru Ghasidas Vishwavidyalaya A Central University. Bilaspur, Chhattisgarh, 495009, India. He has qualified GATE in 2019. He is currently working on Artificial Intelligence, Grid Computing, java, python, Machine Learning, Artificial Intelligence, IOT.

Mr. Manjit Jaiswal is currently working as an Assistant professor in Computer Science and engineering Department, Institute of technology, Guru Ghasidas Vishwavidyalaya A Central University Bilaspur,Chhattisgarh, 495009, India. He received Master of Technology (M-Tech) degree in 2012 from MANIT, Bhopal, Madhya Pradesh, India. He has published more than 10 papers in reputed journals and conference like IEEE, UGC approved, etc. His research interest fields are Parallel Computing, Firewall, Network Security, Distributed System, Algorithms, Routing Algorithm, Machine Learing, IOT etc. He has 6 years of teaching experience.