A Review of Supervised Machine Learning Algorithms

Amanpreet Singh
Bharati Vidyapeeth's College of
Engineering, A-4, Paschim Vihar,
Rohtak Road, New Delhi – 110063, India
Email Id: mngujral@gmail.com

Narina Thakur
Bharati Vidyapeeth's College of
Engineering, A-4, Paschim Vihar,
Rohtak Road, New Delhi – 110063, India
Email Id: narinat@gmail.com

Aakanksha Sharma
Bharati Vidyapeeth's College of
Engineering, A-4, Paschim Vihar,
Rohtak Road, New Delhi – 110063, India
Email Id:
er.akanksha0711@gmail.com

***Abstract* — Supervised machine learning is the construction of algorithms that are able to produce general patterns and hypotheses by using externally supplied instances to predict the fate of future instances. Supervised machine learning classification algorithms aim at categorizing data from prior information. Classification is carried out very frequently in data science problems. Various successful techniques have been proposed to solve such problems viz. Rule-based techniques, Logic-based techniques, Instance-based techniques, stochastic techniques. This paper discusses the efficacy of supervised machine learning algorithms in terms of the accuracy, speed of learning, complexity and risk of over fitting measures. The main objective of this paper is to provide a general comparison with state of art machine learning algorithms.**

***Keywords* — Artificial Neural Networks (ANN), Bayesian Network (BN), Decision Trees (DT), k-Nearest Neighbors (k-NN), Logistic Regression (LR), Random Forests (RF), Supervised Machine Learning, Support Vector Machine (SVM)**

NOMENCLATURE

Artificial Neural Networks (ANN), Bayesian Network (BN), Decision Trees (DT), k-Nearest Neighbors (k-NN), Logistic Regression (LR), Random Forests (RF), Machine Learning (ML), Support Vector Machine (SVM)

I. INTRODUCTION

Classification is essential to data analytics, pattern recognition and machine learning. It is a supervised learning technique, since it categorizes data from the prior information. The class of each testing instance is decided by combining the features and finding patterns common to each class from the training data. Classification is done in two phases. First, a classification algorithm is applied on the training data set and then the extracted model is validated against a labeled test data set to measure the model performance and accuracy. Applications of classification include document classification, spam filtering, image classification, fraud detection, churn analysis, risk analysis, etc.

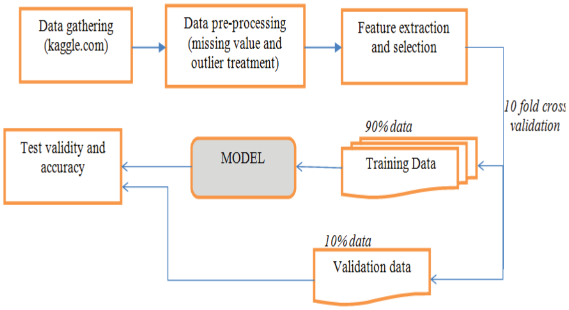
The next section describes the methodology adapted for the study. We go on to explore the related work in the third section. It includes a short description of the algorithms and discusses its variants, if any, and their applications. The fourth section covers the results obtained by studying the application of each algorithm. Conclusion and future scope is covered in the fifth section.

II. METHODOLOGY

We compared the algorithms on the basis of the following factors: Accuracy- the proportion of correct classifications, Speed- computation time required, Comprehensibility- how complex an algorithm is, Speed of learning- important in a real time system where a classification rule must be learned quickly or adjustments are to be made.

The overall procedure for studying the classification algorithms has been shown in Fig.1.

The first step is collecting the dataset. We have obtained the data from KAGGLE which is an online data science competition platform cum community. Titanic data set has been chosen because this data set is very easy to comprehend and easy to work on. Characteristics of the dataset - Number of observations (N) = 891, Number of attributes (p) = 12, Number of classes (q) = 2.



Data gathering (kaggle.com)
Data pre-processing (missing value and outlier treatment)
Feature extraction and selection
10 fold cross validation
90% data
Test validity and accuracy
MODEL
Training Data
10% data
Validation data

Fig. 1. Methodology adapted

The second step is the data preparation and data pre-processing. We have to deal with missing values and outliers which affect performance of various algorithms. Missing values are replaced by mean for continuous and by mode for categorical data. If class value is missing, then whole observation is omitted. Values lying outside (1.5 * Inter-Quartile range) are considered outliers and are also removed.

978-9-3805-4421-2/16/$31.00 ©2016 IEEE

Third step is Feature Engineering. It can be divided into two parts – feature selection and feature extraction. “Feature Selection is the process of identifying and removing as many irrelevant and redundant features as possible” [1].This enables algorithms to operate faster and more effectively. The fact that many features depend on one another often unduly influences the accuracy of supervised machine learning classification models. This problem can be addressed by constructing new features from the basic feature set. This technique is called Feature Extraction or Feature Transformation [3].

Importance of features is shown in Fig. 2.

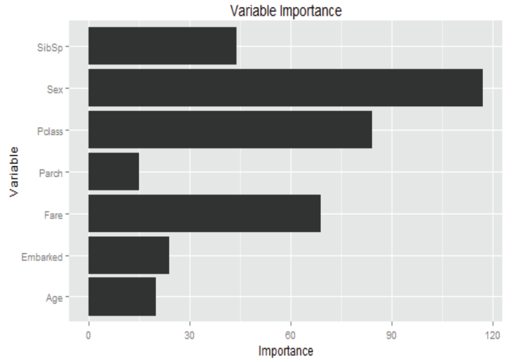


Fig. 2.Variable Importance for Titanic Dataset

We create models with different algorithms and then we compare their accuracy based on cross-validation and by KAGGLE evaluation criteria. The parameters for algorithms can be chosen through resampling methods like cross validation, boosting etc. We apply cross-validation for selecting optimum parameters here.

III. RELATED WORK

Bayesian Networks – This is a graphical model for representing probability relationships among set of variables. Initially, the network (Directed Acyclic Graph) has to be depicted then parameters are determined which makes it difficult to implement without an expert opinion. Prior information about the problem can be represented as structural relationship among its features [32]. However, BN is not very successful with high dimensional data sets because large networks are not feasible in terms of time and space.

Naïve Bayes – It is a BN with only one parent and several children with a strong assumption of independence among its child nodes (*Class Conditional Independence*). If this assumption holds, this type of classifier converges faster than discriminative models (such as Logistic Regression). NB takes less computational time for training. Unlike Neural Networks or SVM, there are no free parameters to be set which greatly simplifies NB [14]. It returns probability which makes it simpler to apply NB to wide variety of tasks. It is not applicable when one needs to take the interactions between features into consideration [7, 11, 13].

Logistic Regression – Statistical models in which a logistic curve is fitted to the dataset [21]. This technique is applied when the dependent variable or target variable is dichotomous. Unlike Decision Trees or SVM’s, there is nice probabilistic interpretation and model can be updated to take new data easily (using online gradient descent method). Since it returns probability, the classification thresholds can be easily adjusted. The logistic model can be an alternative for Discriminant Analysis [10]. It has fewer assumptions - no assumption on the distribution of the independent variables, no linear relationship between the predictors and target variable has to be assumed. It can handle interaction effect, nonlinear effect and power terms. However, it requires large sample size to achieve stable results.

Decision Trees and Random Forests – DT’s are easy to interpret and explain, can easily handle interactions between features. Since, it is non-parametric, outliers don’t affect the model much and it can deal with linearly inseparable data. Some famous algorithms are: ID3, C4.5, C5.0 and CART according to different splitting criteria such as Gini Coefficient, Gain Ratio and Info Gain [18]. Decision Trees can handle variety of data (nominal, numeric, textual), missing values, and redundant attributes; have good generalization ability; are robust to noise, provide high performance for relatively small computational effort. However, it is difficult to handle high dimensional data with DTs. Though computational time is less but considerable time is taken to build the tree. These use divide and conquer approach which performs well if few highly relevant attributes exist but not very well if many complex interactions are present. Errors propagate through trees, which becomes a serious problem as number of classes increase [11, 13]. Furthermore, as tree grows, the number of records in the leaf nodes may be too small to make statistically significant decisions about the class representation. This is called the *Data Fragmentation Problem*. It can be avoided by disallowing further splitting when number of records falls below certain threshold. Also, without proper pruning DTs can easily over fit, which is why an ensemble model Random Forest was developed.

Random Forest is an ensemble method that operates by training a number of decision trees and returning the class with the majority over all the trees in the ensemble [21]. RFs, usually slightly ahead of SVMs, are the winner of many problems in classification. They are fast, scalable, robust to noise, do not over fit, easy to interpret and visualize with no parameters to manage. However, as number of trees increases, algorithm becomes slow for real-time prediction. Several attempts have been made at improving RFs such as decreasing the correlation between the trees, using several attribute evaluation measures in split selection. Another mechanism proposes first to estimate the average margin of the trees on the instances most similar to the new instance and then, after discarding the trees with negative margin, weight the trees’ votes with the margin [24].

Support Vector Machine – It is a complex algorithm, but can provide high accuracy. It also prevents theoretical guarantees regarding over fitting. And with an appropriate kernel, they can work well even if your data isn’t linearly separable in base feature space. They are based on the concept of maximizing the minimum distance from hyperplane to the nearest sample point [26]. Unlike k-NN, accuracy and performance are independent of size of data but on number of training cycles. Especially popular in text classification problems where very high dimensional spaces are the norm. Complexity remains unaffected by number of features. It is robust to high dimensional data and has good generalization ability. But, training speed is less and its performance is dependent on choice of parameters [10].

Since, selection of parameters affects the performance, a technique known as Particle Swarm Optimizer (PSO) is used for selection of optimum parameters. This hybrid model is known as PSO-SVM [30]. Applications include feature selection and Image Classification. Another such model is the evolutionary SVM which is employed to solve the dual optimization problem of SVM. It not only makes an efficient classifier but an adaptive feature extractor as well.

K Nearest Neighbor – It is a non-parametric classification algorithm. It assigns to an unlabeled sample point, the class of the nearest of a set of previously labeled points [21]. The rule is independent of the joint distribution of the sample points and their classifications. It is well suited for multi-modal classes as well as applications where object can have many labels. It is a simple lazy learning method, it has lower efficiency. Also, the performance is dependent on selecting good value of ‘k’. There is no principled way to choose ‘k’, except through computationally expensive techniques like cross validation. It is affected adversely by noise and it is sensitive to irrelevant features also. Performance also varies according to size since all data must be revisited [32, 7].

Neural Networks – They are computational devices that are loosely based on the neuronal structure, processing method and learning ability of the human brain but on much smaller scales. This technique is applicable to problems where the relationships may be non-linear or quite dynamic. NNs provide a powerful alternative to conventional techniques which are often limited by strong assumptions of normality, linearity, variable independence etc. Because a NN can capture many kinds of relationships it allows the user to quickly and relatively easily model phenomena which otherwise may have been very difficult or impossible to explain otherwise. Various variants include Back Propagation Neural Network (BPNN), Probabilistic NN (PNN), Radial Basis Function NN (RBNN) and Complementary NN (CMPNN) classified on the basis of different method implemented to train the network. Most commonly used is BPNN. However, BPNN tends to be slower to train than others, which can be problematic in very large networks with a large amount of data.

A Perceptron is the simplest form of NN, used for classification of patterns said to be linearly separable. It consists of single neuron with adjusted weights. Presence of irrelevant features makes training very inefficient and impractical. A Multi-Layer Perceptron (MLP), is the most widely used NN classifier, capable of modeling complex functions and is robust to irrelevant input and noise [3]. It is hard to train except if one chooses weights closer to optimum weights initially. Generally, determining the size of the hidden layer is a problem. An underestimation will lead to poor approximation whereas overestimation will lead to over fitting and generalization error. Similarly, performance is also sensitive to chosen parameter values. A variant known as Auto-MLP deals with this problem. It is a self-tuning MLP classifier i.e. it automatically adjusts leaning parameters and trains a population of MLPs in parallel. Also, Genetic Algorithms are being used to train the network optimally. [16, 26]

Discriminant Analysis – It combines variables in such a **w**ay that the differences between predefined groups are maximized. Variables can be combined in linear or quadratic fashion giving Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) respectively. LDA is applied when it is assumed that the predictors are normally distributed and covariance of each class is same. QDA has no such assumptions. QDA separates the classes using a quadratic surface (i.e. a conic section). Both are used as classifiers (linear and quadratic). However, LDA is more commonly used as a Dimension Reduction technique.

IV. EXPERIMENTAL RESULTS

We see that tree-based algorithms have performed better than others. This might be because there are certain variables such as sex, pclass (Fig 2.) which are very efficient in dividing the population into classes survived = {0, 1}. Discriminant Analysis is also among the leaders since it combines the variables in such a way that the difference in the population is maximized, and here some variables clearly make the difference.

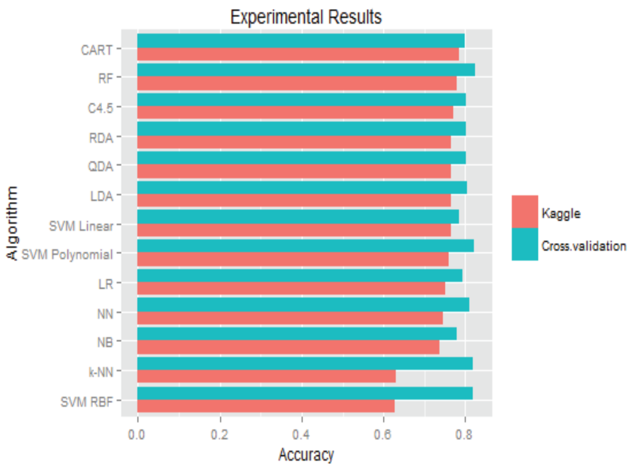


Fig. 3. Experimental Results

Other complex algorithms have underperformed. The reason to which can be thought of to be poor choice of different parameters involved in the algorithm (k in k-NN, initial weights in Neural Networks, etc.). The difference in the two measures (Cross-validation and testing accuracy) indicates the degree of risk of overfitting the model. Here, we see k-NN and SVM-RBF have a difference greater than 10%.

The results obtained are shown in Table 1. as well as Fig. 3.

TABLE 1.Experimental Results

Sr. No.	Results based on Accuracy		
	Algorithm	*Test Accuracy*	*Cross-validation*
1.	CART	0.78469	0.7987090
2.	C4.5	0.76555	0.872267
3.	k-NN	0.63158	0.8194631
4.	SVM RBF kernel	0.62679	0.8189787
5.	SVM Polynomial kernel	0.76077	0.8209189
6.	SVM Linear kernel	0.76555	0.784853
7.	Random Forests	0.77990	0.8249994
8.	Naïve Bayes	0.73684	0.7779668
9.	Logistic Regression	0.75120	0.7928374
10.	Neural Network	0.74641	0.8113895
11.	LDA	0.76555	0.8044938
12.	QDA	0.76555	0.8020449
13.	RDA	0.76555	0.8008372

TABLE II Pros-Cons and applications of supervised ML Classification Algorithms

Algorithm	Advantages	Disadvantages	Application Areas
Bayesian Network	ability to interpret problem in terms of structural relationship among predictors, takes less computational time for training, no free parameters to be set	performance decreases as data grows, cannot deal with high dimensional data	document classification, medical diagnostic systems
Logistic Regression	output interpreted as probability hence wide applications, can handle nonlinearity, interaction effect and power terms	large sample size to achieve stable results, suffer multicollinearity	Crash types,injury severity, voters' types
Decision Trees	non-parametric, handles feature interactions, can deal with linearly inseparable data, handle variety of data, missing values, and redundant attributes; have good generalization ability, robust to noise, provide high performance for relatively small computational effort	difficult to deal with high dimensional data, can easily over fit, considerable time taken to build the tree, can't deal with complex interactions, errors propagate through trees, problem of data fragmentation	Implantable devices, welding quality, drug analysis, remote sensing
Random Forests	fast, scalable , robust to noise, does not over fit, offers explanation and visualization of its output without any parameters to worry about	the algorithm slows down as the number of trees increases	To find cluster of patients, classification of microarray data, object detection
SVM	high accuracy, avoids over fitting, flexible selection of kernels for nonlinearity, accuracy and performance are independent of number of features, good generalization ability	complex, training speed is less and its performance is dependent on choice of parameters	Text classification

k-NN	well suited for multi-modal classes, independent of the joint distribution of the sample points and their classification	lower efficiency, dependent on selecting good value of 'k', affected adversely by noise and irrelevant features, performance also varies according to size of data	Density estimation, vision,computational geometry
Neural Networks	deals with relationships which may be non-linear or dynamic, not restricted by strong assumptions of linearity, normality, variable independence etc., robust to irrelevant input and noise	generally slower to train, performance is sensitive to the size of the hidden layer and chosen parameter values, difficult to interpret	Image classification

V. CONCLUSION AND FUTURE SCOPE

This paper discusses the most commonly used supervised machine learning algorithms for classification. Our aim was to prepare a comprehensive review of the key ideas, drawing out pros and cons and useful variants of the discussed algorithms. The paper shows that every algorithm differs according to area of application and it is not the case that a single algorithm is superior in every scenario. The decision of choosing an appropriate algorithm is based on the type of problem and the data available. Again, by choosing two or more suitable algorithm and creating an ensemble, the accuracy can be increased. We hope that the references cited cover the major drawbacks, guiding the researcher in interesting research directions

REFERENCES

[1] L., Liu, H. Yu, "Efficient Feature Selection via Analysis of Relevance and Redundancy," *JMLR*, pp. 1205-1224, 2004.

[2] S. B. Kotsiantis, P. E. Pintelas, and I. D. Zaharakis, "Machine learning: a review of classification and combining techniques," *Artificial Intelligence Rev* vol. 26, pp. 159–190, 2006.

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

[4] K. Wisaeng, "A Comparison of Different Classification Techniques for Bank Direct Marketing," *International Journal of Soft Computing and Engineering (IJSCE)*, Volume-3, Issue-4, pp. 116-119, 2013.

[5] Ch. M. Bishop, *Pattern Recognition and Machine Learning*.: Springer, 2006.

[6] T. Mitchell, *Machine Learning*.: MITPress and McGraw-Hill, 1997.

[7] M. J. Islam, Q. M. J. Wu, M. Ahmadi, and M. A. Sid-Ahmed, "Investigating the Performance of Naïve- Bayes Classifiers and K-Nearest Neighbor Classifiers," *Journal of Convergence Information Technology*, vol. 5, no. 2, 2010.

[8] C.S. Trilok and J. Manoj, "WEKA Approach for Comparative Study of Classification Algorithm," *International Journal of Advanced Research in Computer and Communication Engineering*, pp. 1925-1931, 2013.

[9] J. Demsar, "Statistical comparisons of classifiers over multiple datasets," *Journal of Machine Learning Research*, vol. 7, pp. 1-30, 2006.

[10] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," *in 23rd international conference on Machine learning*, 2006.

[11] C. J. Hinde, and R. G. Stone D. Xhemali, "Naïve Bayes vs. Decision Trees vs. Neural Networks in the Classification of Training Web Pages," *International Journal of Computer Science Issue*, vol. 4, no. 1, 2009.

[12] R. Entezari-Maleki, A. Rezaei, and B. Minaei-Bidgoli, "Comparison of Classification Methods Based on the Type of Attributes and Sample Size," *Journal of Convergence Information Technology*, vol. 4, no. 3, pp. 94-102, 2009, Available at: http://www4.ncsu.edu/~arezaei2/paper/JCIT4184028_Camera%20Ready.pdf.

[13] N. B. Amor, S. Benferhat, and Z. Elouedi, "Naive bayes vs decision trees in intrusion detection systems," *in ACM symposium on Applied computing*, 2004, pp. 420-424.

[14] L. I. Kuncheva, "On the optimality of naive bayes with dependent binary features," *Pattern Recognition Letters*, vol. 27, pp. 830-837, 2006.

[15] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491-502, 2005.

[16] Z. H. Zhou, "Rule extraction: Using neural networks or for neural networks?," *Journal of Computer Science and Technology*, vol. 19, no. 2, pp. 249-253, 2004.

[17] K. Mollazade, M. Omid, and A. Arefi, "Comparing data mining classifiers for grading raisins based on visual features," *Computers and electronics in agriculture*, vol. 84, pp. 124-131, 2012.

[18] 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, 2005.

[19] N. Williams, S. Zander, and G. Armitage, "A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification," *SIGCOMM Comput. Commun. Rev.* vol 36, pp. 5-16, 2006.

[20] M. Kuhn, "Building Predictive Models in R Using the caret Package," *Journal of Statistical Software*, Volume 28, Issue 5, vol. 28, no. 5, pp. 1-26, 2008.

[21] A. C. Lorena et al., "Comparing machine learning classifiers in potential distribution modelling," *Expert Systems with Applications*, vol. 38, pp. 5268 – 5275, 2011.

[22] S. Alelyani, J. Tang, and H. Liu, "Feature Selection for Classification: A Review," *Data Clustering: Algorithms and Applications*, vol. 29, 2013.

[23] Y. Wu, Y. Zhuang, X. Long, F. Lin, and W. Xu, "Human Gender Classification: A Review," *IEEE Sensors Journal*, 2015, Available at: http://arxiv.org/pdf/1507.05122v1.pdf.

[24] M. Robnik-Sikonja, "Improving Random Forests*," in Machine Learning, ECML, Berlin*, 2004.

[25] C. Zhong, D. Miao, and P.i Fränti, "Minimum spanning tree based split-and-merge: A hierarchical clustering method," *Information Sciences 181*, pp. 3397–3410, 2011.

[26] M. Aly, "Survey on multiclass classification methods," *Neural Network*, pp. 1-9, 2005.

[27] H. Hormozi, E. Hormozi, and H. R. Nohooji, "The Classification of the Applicable Machine Learning Methods in Robot Manipulators," *International Journal of Machine Learning and Computing*, vol. 2, no. 5, pp. 560-563, 2012.

[28] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *The Journal of Machine Learning Research*, vol. 5, pp. 1205-1224, 2004.

[29] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 3, pp. 247-259, 2011.

[30] S. W. Lin, K. C. Ying, S. C., Chen, and Z. J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert systems with applications*, vol. 35, no. 4, pp. 1817-1824, 2008.

[31] Ian H. Witten, Eibe Frank, and Mark A. Hall, *Data Mining - Practical Machine Learning Tools and Techniques*, 3rd ed.: Elsevier, 2014.

[32] F. Pernkopf, "Bayesian network classifiers versus selective k-NN classifier," *Pattern Recognition*, vol. 38, no. 1, pp. 1-10, 2005.