International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

Optimizing Supervised Classification Algorithms for Real - World Data

G. Divya[1], Dr. V. Maniraj[2]

[1]Research Scholar, Department of Computer Science, A. V. V. M Sri Pushpam College (Autonomous), Poondi, Thanjavur (Dt), Affiliated to Bharathidasan University, Thiruchirappalli, Tamilnadu, India
Email: *gdivya19.mca[at]gmail.com*

[2]Associate Professor, Research Supervisor, Head of the Department, Department of Computer Science, A. V. V. M Sri Pushpam College (Autonomous), Poondi, Thanjavur (Dt), Affiliated to Bharathidasan University, Thiruchirappalli, Tamilnadu, India
Email: *manirajv61[at]gmail.com*

Abstract: ***Supervised Machine Learning (SML) involves the identification of algorithms that learn from externally provided data instances to generate generalized models, which are then used to make predictions about future data. Among the various tasks performed by intelligent systems, supervised classification is one of the most commonly executed. This paper outlines several Supervised Machine Learning (ML) classification techniques, compares different supervised learning algorithms, and identifies the most effective classification method based on dataset characteristics, including the number of instances and features. Seven distinct machine learning algorithms were examined: Decision Table, Random Forest (RF), Naïve Bayes (NB), Support Vector Machine (SVM), Neural Networks (Perceptron), JRip, and Decision Tree (J48), utilizing the Waikato Environment for Knowledge Analysis (WEKA) tool. For the analysis, a Diabetes dataset containing 786 instances with eight independent attributes and one dependent attribute was used for classification. The findings reveal that SVM achieved the highest precision and accuracy. Naïve Bayes and Random Forest algorithms followed as the next most accurate methods after SVM. The study highlights that the time required to develop a model, as well as the precision (accuracy), is a critical factor, while Kappa statistic and Mean Absolute Error (MAE) also play significant roles. Thus, for effective supervised predictive machine learning, algorithms must prioritize precision, accuracy, and minimal error to have supervised predictive machine learning.***

Keywords: Machine Learning, Classifiers, Data Mining Techniques, Data Analysis, Learning Algorithms, Supervised Machine Learning

1. Introduction

Machine learning is one of the fastest growing areas of computer science, with far - reaching applications. It refers to the automated detection of meaningful patterns in data. Machine learning tools concerned with endowing programs with the ability to learn and adapt [19].

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 a good reason to believe that smart data analysis will become even more pervasive as a necessary ingredient for technological progress.

There are several applications for Machine Learning (ML), the most significant of which is data mining. People are often prone to making mistakes during analyses or, possibly, when trying to establish relationships between multiple features [9].

Data Mining and Machine Learning are Siamese twins from which several insights can be derived through proper learning algorithms. There has been tremendous progress in data mining and machine learning as a result of evolution of smart and Nano technology which brought about curiosity in finding hidden patterns in data to derive value. The fusion of statistics, machine learning, information theory, and computing has created a solid science, with a firm mathematical base, and with very powerful tools.

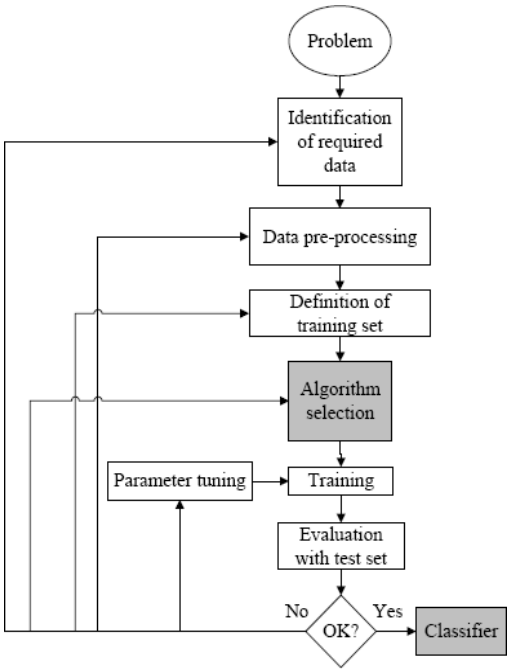
Machine learning algorithms are organized into a taxonomy based on the desired outcome of the algorithm. Supervised learning generates a function that maps inputs to desired outputs. Unprecedented data generation has made machine learning techniques become sophisticated from time to time. This has called for utilization for several algorithms for both supervised and unsupervised machine learning. Supervised learning is fairly common in classification problems because the goal is often to get the computer to learn a classification system that we have created [21].

ML is perfectly intended for accomplishing the accessibility hidden within Big Data. ML hand over's on the guarantee of extracting importance from big and distinct data sources through outlying less dependence scheduled on individual track as it is data determined and spurts at machine scale. Machine learning is fine suitable towards the intricacy of handling through dissimilar data origin and the vast range of variables as well as amount of data concerned where ML prospers on increasing datasets. The extra data supply into a ML structure, the more it be able to be trained and concern the consequences to superior value of insights. At the liberty from the confines of individual level thought and study, ML is clever to find out and show the patterns hidden in the data [15].

One standard formulation of the supervised learning task is the classification problem: The learner is required to learn (to approximate the behavior of) a function which maps a vector into one of several classes by looking at several input - output examples of the function. Inductive machine learning is the process of learning a set of rules from instances (examples in a training set), or more generally speaking, creating a classifier that can be used to generalize

from new instances. The process of applying supervised ML to a real - world problem is described in Figure 1.

Figure 1: The Processes of Supervised Machine Learning



This work focuses on the classification of ML algorithms and determining the most efficient algorithm with highest accuracy and precision. As well as establishing the performance of different algorithms on large and smaller data sets with a view classify them correctly and give insight on how to build supervised machine learning models. The remaining part of this work is arranged as follows: Section 2 presents the literature review discussing classification of different supervised learning algorithms; section 3 presents the methodology used, section 4 discusses the results of the work while section 5 gives the conclusion and recommendation for further works.

2. Literature Review

a) Classification of Supervised Learning Algorithms

According to [21], the supervised machine learning algorithms which deals more with classification includes the following: Linear Classifiers, Logistic Regression, Naïve Bayes Classifier, Perceptron, Support Vector Machine; Quadratic Classifiers, K - Means Clustering, Boosting, Decision Tree, Random Forest (RF); Neural networks, Bayesian Networks and so on.

1) Linear Classifiers: Linear models for classification separate input vectors into classes using linear (hyperplane) decision boundaries [6]. The goal of classification in linear classifiers in machine learning, is to group items that have similar feature values, into groups. [23] stated that a linear classifier achieves this goal by making a classification decision based on the value of the linear combination of the features. A linear classifier is often used in situations where the speed of classification is an issue, since it is rated the fastest classifier [21]. Also, linear classifiers often work very well when the number of dimensions is large, as in document classification, where each element is typically the number of counts of a word in a document. The rate of convergence among data set variables however depends on the margin. Roughly speaking, the margin quantifies how linearly separable a dataset is, and hence how easy it is to solve a given classification problem [18].

2) Logistic regression: This is a classification function that uses class for building and uses a single multinomial logistic regression model with a single estimator. Logistic regression usually states where the boundary between the classes exists, also states the class probabilities depend on distance the boundary, in a specific approach. This moves towards the extremes (0 and 1) more rapidly when data set is larger. These statements about probabilities which make logistic regression more than just a classifier. It makes stronger, more detailed predictions, and can be fit in a different way; but those strong predictions could be wrong. Logistic regression is an approach to prediction, like Ordinary Least Squares (OLS) regression. However, with logistic regression, prediction results in a dichotomous outcome [13]. Logistic regression is one of the most commonly used tools for applied statistics and discrete data analysis. Logistic regression is linear interpolation [11].

3) Naive Bayesian (NB) Networks: These are very simple Bayesian networks which are composed of directed acyclic graphs with only one parent (representing the unobserved node) and several children (corresponding to observed nodes) with a strong assumption of independence among child nodes in the context of their parent [7]. Thus, the independence model (Naive Bayes) is based on estimating [14]. Bayes classifiers are usually less accurate that other more sophisticated learning algorithms (such as ANNs). However, [5] performed a large - scale comparison of the naive Bayes classifier with state - of - the - art algorithms for decision tree induction, instance - based learning, and rule induction on standard benchmark datasets, and found it to be sometimes superior to the other learning schemes, even on datasets with substantial feature dependencies. Bayes classifier has attribute - independence problem which was addressed with Averaged One - Dependence Estimators [8].

4) Multi - layer Perceptron: This is a classifier in which the weights of the network are found by solving a quadratic programming problem with linear constraints, rather than by solving a non - convex, unconstrained minimization problem as in standard neural network training [21]. Other well - known algorithms are based on the notion of perceptron [17]. Perceptron algorithm is used for learning from a batch of training instances by running the algorithm repeatedly through the training set until it finds a prediction vector which is correct on all of the training set. This prediction rule is then used for predicting the labels on the test set [9].

5) Support Vector Machines (SVMs): These are the most recent supervised machine learning technique [24]. Support Vector Machine (SVM) models are closelyrelated to classical multilayer perceptron neural networks. SVMs revolve around the notion of a ―margin‖—either side of a hyperplane that separates two data classes. Maximizing the margin and thereby creating the largest possible distance between the separating hyperplane and the instances on either side of it has been proven to reduce an upper bound on the expected generalisation error [9].

6) K - means: According to [2] and [22]K - means is one of the simplest unsupervised learning algorithms that solve the well - known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. K - Means algorithm is be employed when labeled data is not available [1]. General method of converting rough rules of thumb into highly accurate prediction rule. Given ―weak‖ learning algorithm that can consistently find classifiers (―rules of thumb‖) at least slightly better than random, say, accuracy _ 55%, with sufficient data, a boosting algorithm can provably construct single classifier with very high accuracy, say, 99% [16].

7) Decision Trees: Decision Trees (DT) are trees that classify instances by sorting them based on feature values. Each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. Instances are classified starting at the root node and sorted based on their feature values [9]. Decision tree learning, used in data mining and machine learning, uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. More descriptive names for such tree models are classification trees or regression trees [20]. Decision tree classifiers usually employ post - pruning techniques that evaluate the performance of decision trees, as they are pruned by using a validation set. Any node can be removed and assigned the most common class of the training instances that are sorted to it [9].

8) Neural Networks: [2] opined Neural Networks (NN) that can actually perform a number of regression and/or classification tasks at once, although commonly each network performs only one. In the vast majority of cases, therefore, the network will have a single output variable, although in the case of many - state classification problems, this may correspond to a number of output units (the post - processing stage takes care of the mapping from output units to output variables). Artificial Neural Network (ANN) depends upon three fundamental aspects, input and activation functions of the unit, network architecture and the weight of each input connection. Given that the first two aspects are fixed, the behavior of the ANN is defined by the current values of the weights. The weights of the net to be trained are initially set to random values, and then instances of the training set are repeatedly exposed to the net. The values for the input of an instance are placed on the input units and the output of the net is compared with the desired output for this instance. Then, all the weights in the net are adjusted slightly in the direction that would bring the output values of the net closer to the values for the desired output. There are several algorithms with which a network can be trained [12].

9) Bayesian Network: A Bayesian Network (BN) is a graphical model for probability relationships among a set of variables (features). Bayesian networks are the most well - known representative of statistical learning algorithms [9]. The most interesting feature of BNs, compared to decision trees or neural networks, is most certainly the possibility of taking into account prior information about a given problem, in terms of structural relationships among its features [9]. A problem of BN classifiers is that they are not suitable for datasets with many features [4]. This prior expertise, or domain knowledge, about the structure of a Bayesian network can take the following forms:

- Declaring that a node is a root node, i. e., it has no parents.
- Declaring that a node is a leaf node, i. e., it has no children.
- Declaring that a node is a direct cause or direct effect of another node.
- Declaring that a node is not directly connected to another node.
- Declaring that two nodes are independent, given a condition - set.
- Providing partial nodes ordering, that is, declare that a node appears earlier than another node in the ordering.
- Providing a complete node ordering.

b) Features of Machine Learning Algorithms

Supervised machine learning techniques are applicable in numerous domains. A number of Machine Learning (ML) application oriented papers can be found in [18], [25]. Generally, SVMs and neural networks tend to perform much better when dealing with multi - dimensions and continuous features. On the other hand, logic - based systems tend to perform better when dealing with discrete/categorical features. For neural network models and SVMs, a large sample size is required in order to achieve its maximum prediction accuracy whereas NB may need a relatively small dataset.

There is general agreement that k - NN is very sensitive to irrelevant features: this characteristic can be explained by the way the algorithm works. Moreover, the presence of irrelevant features can make neural network training very inefficient, even impractical. Most decision tree algorithms cannot perform well with problems that require diagonal partitioning. The division of the instance space is orthogonal to the axis of one variable and parallel to all other axes.

Therefore, the resulting regions after partitioning are all hyper rectangles. The ANNs and the SVMs perform well when multi - co linearity is present and a nonlinear relationship exists between the input and output features.

Naive Bayes (NB) requires little storage space during both the training and classification stages: the strict minimum is the memory needed to store the prior and conditional probabilities. The basic kNN algorithm uses a great deal of storage space for the training phase, and its execution space is at least as big as its training space. On the contrary, for all non - lazy learners, execution space is usually much smaller than training space, since the resulting classifier is usually a highly condensed summary of the data. Moreover, Naive Bayes and the kNN can be easily used as incremental learners whereas rule algorithms cannot. Naive Bayes is naturally robust to missing values since these are simply ignored in computing probabilities and hence have no impact on the final decision. On the contrary, kNN and neural networks require complete records to do their work.

Finally, Decision Trees and NB generally have different operational profiles, when one is very accurate the other is not and vice versa. On the contrary, decision trees and rule classifiers have a similar operational profile. SVM and ANN have also a similar operational profile. No single learning algorithm can uniformly outperform other algorithms over all datasets.

Different data sets with different kind of variables and the number of instances determine the type of algorithm that will perform well. There is no single learning algorithm that will outperform other algorithms based on all data sets according to no free lunch theorem. [10] Table 1 presents the comparative analysis of various learning algorithms.

Table 1: Comparing learning algorithms (**** stars represent the best and * star the worst performance) [9]

	Decision Trees	Neural Networks	Naïve Bayes	kNN	SVM	Rule-learners
Accuracy in general	**	***	*	**	****	**
Speed of learning with respect to number of attributes and the number of instances	***	*	****	****	*	**
Speed of classification	****	****	****	*	****	****
Tolerance to missing values	***	*	****	*	**	**
Tolerance to irrelevant attributes	***	*	**	**	****	**
Tolerance to redundant attributes	**	**	*	**	***	**
Tolerance to highly interdependent attributes (e.g. parity problems)	**	***	*	*	***	**
Dealing with discrete/binary/continuous attributes	****	***(not discrete)	***(not continuous)	***(not directly discrete)	**(not discrete)	***(not directly continuous)
Tolerance to noise	**	**	***	*	**	*
Dealing with danger of overfitting	**	*	***	***	**	**
Attempts for incremental learning	**	***	****	****	**	*
Explanation ability/transparency of knowledge/classifications	****	*	****	**	*	****
Model parameter handling	***	*	****	***	*	***

Research Methodology

The research data was obtained from National Institute of Diabetes and Digestive and Kidney Diseases which was made available online at University of California, Irvive website: https: //archive. ics. uci. edu/ml/machine - learning databases/pima - indians - diabetes/ (2017). This data was chosen because of its accuracy and has also been anonymized (de - identified), therefore confidentiality is ensured. The number of Attributes is 8 with one class making it 9. All attributes are numeric - valued as follows:

1) Number of times pregnant
2) Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3) Diastolic blood pressure (mm Hg)
4) Triceps skin fold thickness (mm)
5) 5.2 - Hour serum insulin (mu U/ml)
6) Body mass index (weight in kg/ (height in m) ^2)
7) Diabetes pedigree function
8) Age (years)
9) Class variable (0 or 1)

Table 2: Class Distribution: (class value 1 is interpreted as "tested positive for diabetes") and (class value 0 is interpreted as "tested negative for diabetes")

Class	Value Number of instances	Converted Value (attribute)
0	500	NO
1	268	YES

Table 2 shows 768 as the total number of instances used for this research work with 500 tested positive for diabetes and 268 tested negative for diabetes.

The comparative analysis among various supervised machine learning algorithms was carried out using WEKA 3.7.13 (WEKA - Waikato Environment for Knowledge Analysis). Thed ata set was trained to reflect one nominal attribute column as the dependent variable. The values 1's for class distribution (class variable) were changed to YES which means tested POSITIVE for DIABETS and values 0s for class distribution (class variable) were changed NO which means tested NEGATIVE for DIABETES. This is essential because most of the algorithms require that there must be at least one nominal variable column. Seven classification algorithms were used in the course of this research namely: Decision Table, Random Forest, Naïve Bayes, SVM, Neural Networks (Perceptron), JRip and Decision Tree (J48). The following attributes were considered for the comparative analysis: Time, Correctly Classified, Incorrectly Classified, Test Mode, No of instances, Kappa statistic, MAE, Precision of YES, Precision of NO and Classification.

In order to predict the accuracy and ensure precision for different machine learning algorithms, this research work was carried out by tuning the parameters with two different sets of number of instances. The first category was 768 instances and 9 attributes as follows (Number of times pregnant, Plasma glucose concentration a 2 hours in an oral glucose tolerance test, Diastolic blood pressure (mm Hg), Triceps skin fold thickness (mm), 2 - Hour serum insulin (mu U/ml), Body mass index (weight in kg/ (height in m) ^2), Diabetes pedigree function, Age (years) and Class variable (0 or 1)) with one dependent variable and eight independent variables. The second category of data set was 384 instances and 6 attributes as follows (Number of times pregnant, Plasma glucose concentration a 2 hours in an oral glucose tolerance test, 2 - Hour serum insulin (mu U/ml), Diabetes pedigree function, Age (years) and Class variable (0 or 1)) with one dependent variable and five independent variables.

3. Results and Discussion

3.1 Results

WEKA was used in the classification and comparison of the various machine leaning algorithms. Table 3 shows the results with 9 attributes as well as parameters considered MAE (Mean Absolute Error) is a measure of how close forecast or predictions are to the eventual outcome. Kappa Statistic is a metric that compares an observed accuracy with an expected accuracy (Random Chance) YES means tested positive to diabetes. NO means tested negative for diabetes.

Table 3: Comparison of various classification algorithms with large data set and more attributes

Algorithm	Time	Correctly Classified %	Incorrectly Classified %	TestMode	Attributes	No of instances	Kappa statistic	MAE	Precision of YES	Precision of NO	Classification
DecisionTable	0.23	72.3958	27.6042	10- foldcross - validation	9	768	0.3752	0.341	0.619	0.771	Rules
RandomForest	0.55	74.7396	25.2604	10- foldcross - validation	9	768	0.4313	0.3105	0.653	0.791	Trees
NaïveBayes	0.03	76.3021	23.6979	10- foldcross - validation	9	768	0.4664	0.2841	0.678	0.802	Bayes
SVM	0.09	77.3438	22.6563	10- foldcross - validation	9	768	0.4682	0.2266	0.740	0.785	Functions
Neural Networks	0.81	75.1302	24.8698	10- foldcross - validation	9	768	0.4445	0.2938	0.653	0.799	Functions
JRip	0.19	74.4792	25.5208	10- foldcross - validation	9	768	0.4171	0.3461	0.659	0.780	Rules
Decision Tree (J48)	0.14	73.8281	26.1719	10- foldcross - validation	9	768	0.4164	0.3158	0.632	0.790	Tree

Table 4 shows the results with 6 attributes of the classification and comparison of the various machine leaning algorithms and parameters considered.

T**able 4:** Comparison of various classification algorithms with smaller data set and less attributes

Algorithm	Time	Correctly Classified %	Incorrectly Classified %	TestMode	Attributes	No of instances	Kappa statistic	MAE	Precision of YES	Precision of NO	Classification
DecisionTable	0.09	67.9688	32.0313	10-foldcross - validation	6	384	0.3748	0.3101	0.581	0.734	Rules
RandomForest	0.42	71.875	28.125	10-foldcross - validation	6	384	0.3917	0.3438	0.639	0.761	Trees
NaïveBayes	0.01	70.5729	29.4271	10-foldcross - validation	6	364	0.352	0.3297	0.633	0.739	Bayes
SVM	0.04	72.9167	27.0833	10- foldcross - validation	6	384	0.3837	0.2708	0.711	0.735	Functions
Neural Networks (Perceptron)	0.17	**59**	**41**	10- foldcross - validation	6	384	**0.1156**	**0.4035**	0.444	0.672	Functions
JRip	0.01	**64**	**36**	10-foldcross -	6	384	**0.2278**	**0.4179**	0.514	0.714	Rules

				validation							
Decision Tree (J48)	0.03	**64 %**	**36**	10-foldcross - validation	6	384	**0.1822**	**0.4165**	0.519	0.685	Tree

Time is the TIME taking to build the model. MAE (Mean Absolute Error) is a measure of how close forecast or predictions are to the eventual outcome. Kappa Statistic is a metric that compares an observed accuracy with an expected accuracy (Random Chance) YES means tested positive to diabetes. NO means tested negative for diabetes.

Table 5 and 6: Ranking of Precision of Positive Diabetes and Negative Diabetes using different algorithms showing smaller and larger data sets respectively

Smaller Dataset 384		
Algorithm	Precision of YES (Positive Diabetes)	Precision of NO (Negative Diabetes)
SVM	0.711	0.735
RandomForest	0.639	0.761
NaïveBayes	0.633	0.739
DecisionTable	0.581	0.734
DecisionTree (J48)	0.519	0.685
JRip	0.514	0.714
Neural Networks (Perceptron)	0.444	0.672

Large DataSet768		
Algorithm	Precision of YES (Positive Diabetes)	Precision of NO (Negative Diabetes
SVM	0.74	0.785
NaïveBayes	0.678	0.802
JRip	0.659	0.78
RandomForest	0.653	0.791
NeuralNetworks (Perceptron)	0.653	0.799
DecisionTree (J48)	0.632	0.79
DecisionTable	0.619	0.771

Table 7 and 8: Ranking of Correctly Classified and Incorrectly Classified with the time to build the model showing smaller and larger data sets respectively using different algorithm.

SmallerDataset384			
Algorithm	Time	Correctly Classified	Incorrectly Classified
SVM	0.04sec	72.92%	27.08%
Random Forest	0.42sec	71.88%	28.13%
Naïve Bayes	0.01sec	70.57%	29.43%
Decision Table	0.09sec	67.97%	32.03%
JRip	0.01 sec	**64%**	**36%**
Decision Tree (J48)	0.03 sec	**64%**	**36%**
Neural Networks (Perceptron)	0.17 sec	**59%**	**41%**

LargeDataSet768			
Algorithm	Time	Correctly Classified	Incorrectly Classified
SVM	0.09sec	77.34%	22.66%
NaïveBayes	0.03sec	76.30%	23.70%
Neural Networks (Perceptron)	0.81sec	75.13%	24.87%
Random Forest	0.55sec	74.74%	25.26%
JRip	0.19sec	74.48%	25.52%
Decision Tree (J48)	0.14sec	73.83%	26.17%
DecisionTable	0.23sec	72.40%	27.60%

Table 9: Descriptive Analysis of various Dataset attributes

Attribute number	Mean	Standard Deviation
1	3.8	3.4
2	120.9	32.0
3	69.1	19.4
4	20.5	16.0
5	79.8	115.2
6	32.0	7.9
7	0.5	0.3
8	33.2	11.8

3.2 Discussion

Table3 shows the comparison of the result for 768 instances and 9 attributes. It was observed that all the algorithms have higher Kappa statistic compared to MAE (Mean Absolute Error). Also, correctly classified instances are higher than incorrectly classified instances. This is an indication that with higher data sets, the predictive analysis is more reliable. SVM and NB require large sample size in order to achieve maximum prediction accuracy as shown in the table 3, while Decision Tree and Decision Table have the least precision.

Table4 shows the comparison of the result for 384 instances and 6 attributes. The Kappa statistics for Neural Networks, JRip and J48 are lower compared to MAE and this does not portray precision and accuracy. This shows that with smaller datasets Neural Networks, JRip and J48 shows drastic reduction in the percentage of correctly classified instances in comparison to incorrectly classified instances. However, with smaller data set SVM and RF shows high accuracy and precision. Whereas Decision Table built the model with more time compare to JRip and Decision Tree. Therefore, less time does not guarantee accuracy. If Kappa Statistic is less than Mean Absolute Error (MAE), the algorithm will not show precision and accuracy. It follows that, the algorithm which such characteristics cannot be used for that data set as it will not show precision and accuracy.

Table 6shows precision for larger data set and smaller data set with SVM reflecting the algorithm with highest prediction. Also, table 5 shows SVM being the algorithm with highest precision. Smaller data sets.

Tales 7 and 8 shows the comparison of percentage of correctly classified and incorrectly classified for smaller and large datasets respectively with the time to build the model. From Table 7, the results reveal Naive Bayes and JRip as the algorithms with fastest time to build, however the percentage of correctly classified is lower in JRip which shows that Time to build as model is not tantamount to accuracy. In the same vein, SVM has the highest level of accuracy with time of 0.04 seconds. Comparing this results with Table 8Neural Networks (Perceptron) was the third correctly classified algorithm. This means that Neural Network performs well with large dataset as compared to small data set. Also, the results shows that Decision Table does not perform well with large dataset. By and large,

SVM algorithm shows the highest classification and the larger the dataset, the higher the precision.

Table 9 shows the mean and standard deviation of all the attributes used in this research reveals that Plasma glucose concentration (attribute 2) has the highest mean as well as Diabetes pedigree function (attribute 7) with the lowest mean which is an indication of strong influence on small data set. However, a lower Standard Deviation (SD) is not necessarily more desirable which means Diabetes pedigree function (attribute 7) might not be of significance vale when analyzing large data set.

4. Conclusion and Recommendation for Further Works

ML classification requires thorough fine tuning of the parameters and at the same time sizeable number of instances for the data set. It is not a matter of time to build the model for the algorithm only but precision and correct classification. Therefore, the best learning algorithm for a particular data set, does not guarantee the precision and accuracy for another set of data whose attributes are logically different from the other. However, the key question when dealing with ML classification is not whether a learning algorithm is superior to others, but under which conditions a particular method can significantly outperform others on a given application problem. Meta - learning is moving in this direction, trying to find functions that map datasets to algorithm performance [12]. To this end, meta - learning uses a set of attributes, called meta - attributes, to represent the characteristics of learning tasks, and searches for the correlations between these attributes and the performance of learning algorithms. Some characteristics of learning tasks are: the number of instances, the proportion of categorical attributes, the proportion of missing values, the entropy of classes, etc.

[3] provided an extensive list of information and statistical measures for a dataset. After a better understanding of the strengths and limitations of each method, the possibility of integrating two or more algorithms together to solve a problem should be investigated. The objective is to utilize the strengths of one method to complement the weaknesses of another. If we are only interested in the best possible classification accuracy, it might be difficult or impossible to find a single classifier that performs as well as a good ensemble of classifiers. SVM, NB and RF machine learning algorithms can deliver high precision and accuracy regardless of the number of attributes and data instances. This research shows that time to build a model is one factor on one hand; and precision with kappa statistic while MAE is another factor on the other hand. Therefore, ML algorithms requires precision, accuracy and minimum error to have supervised predictive machine learning.

This work recommends that for large data sets, a distributed processing environment should be considered. This will create room for high level of correlation among the variables which will ultimately make the output of the model more efficient.

References

[1] Alex, S., & Vishwanathan, S. V. N. (2008). *Introduction to Machine Learning*. Cambridge University Press. ISBN: 0 - 521 - 82583 - 0. Available at KTH website: https: //www.kth. se/social/upload/53a14887f276540ebc81aec3/online. pdf Retrieved from website: http: //alex. smola. org/drafts/thebook. pdf

[2] Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Clarendon Press. ISBN: 0198538642. Available at: http: //cs. du. edu/~mitchell/mario_books/Neural_Networks_for_Pattern_Recognition_ - _Christopher_Bishop. pdf

[3] Brazdil, P., Soares, C., & da Costa, J. (2003). Ranking learning algorithms: Using IBL and meta - learning on accuracy and time results. *Machine Learning*, 50 (3), 251–277. doi: 10.1023/A: 1021713901879. Available at Springer website: https: //link. springer. com/content/pdf/10.1023%2FA%3A1021713901879. pdf

[4] Cheng, J., Greiner, R., Kelly, J., Bell, D., & Liu, W. (2002). Learning Bayesian networks from data: An information - theory based approach. *Artificial Intelligence*, 137, 43 - 90. Available at: http: //www.sciencedirect. com/science/article/pii/S0004370202001911

[5] Domingos, P., & Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero - one loss. *Machine Learning*, 29, 103–130. Available at: http: //disi. unitn. it/~p2p/RelatedWork/Matching/domingos97optimality. pdf

[6] Elder, J. (n. d.). Introduction to Machine Learning and Pattern Recognition. Available at LASSONDE University EECS Department York website: http: //www.eecs. yorku. ca/course_archive/2011 - 12/F/4404 - 5327/lectures/01%20Introduction. pd

[7] Good, I. J. (1951). Probability and the weighing of evidence. *Philosophy*, 26 (97), 163 - 164. doi: https: //doi. org/10.1017/S0031819100026863. Available at: https: //www.cambridge. org/core/journals/philosophy/article/probability - and - the - weighing - of - evidence - by - goodi - j - london - charles - griffin - and - company - 1950 - pp - viii - 119 - price - 16s/7D911224F3713FDCFD1451BBB2982442

[8] Hormozi, H., Hormozi, E., & Nohooji, H. R. (2012). The classification of the applicable machine learning methods in robot manipulators. *International Journal of Machine Learning and Computing (IJMLC)*, 2 (5), 560–563. doi: 10.7763/IJMLC.2012. V2.189. Available at IJMLC website: http: //www.ijmlc. org/papers/189 - C00244 - 001. pdf

[9] Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. *Informatica*, 31, 249–268. Retrieved from IJS website: http: //wen. ijs. si/ojs - 2.4.3/index. php/informatica/article/download/148/140

[10] Lemnaru, C. (2012). *Strategies for dealing with real - world classification problems* (Unpublished PhD thesis). Faculty of Computer Science and

Automation, Universitatea Technica, Din Cluj - Napoca. Available at: http: //users. utcluj. ro/~cameliav/documents/TezaFinalLemnaru. pdf
[11] Logistic regression pp.223–237. Available at: https: //www.stat. cmu. edu/~cshalizi/uADA/12/lectures/ch12. pdf
[12] Neocleous, C., & Schizas, C. (2002). Artificial neural network learning: A comparative review. In I. P. Vlahavas & C. D. Spyropoulos (Eds.), *Methods and Applications of Artificial Intelligence* (pp.300–313). Springer. doi: 10.1007/3 - 540 - 46014 - 4_27. Available at: https: //link. springer. com/chapter/10.1007/3 - 540 - 46014 - 4_27
[13] Newsom, I. (2015). Data Analysis II: Logistic Regression. Available at: http: //web. pdx. edu/~newsomj/da2/ho_logistic. pdf
[14] Nilsson, N. J. (1965). *Learning Machines*. McGraw - Hill. Published in: *Journal of IEEE Transactions on Information Theory*, 12 (3), 407 - 407. doi: 10.1109/TIT.1966.1053912. Available at: http: //dl. acm. org/citation. cfm?id=2267404
[15] Pradeep, K. R., & Naveen, N. C. (2017). A collective study of machine learning (ML) algorithms with big data analytics (BDA) for healthcare analytics (HcA). *International Journal of Computer Trends and Technology (IJCTT)*, 47 (3), 149–155. ISSN: 2231 - 2803. doi: 10.14445/22312803/IJCTT - V47P121. Available from IJCTT website: http: //www.ijcttjournal. org/2017/Volume47/number - 3/IJCTT - V47P121. pdf
[16] Schapire, R. (n. d.). Machine learning algorithms for classification.
[17] Rosenblatt, F. (1962). *Principles of Neurodynamics*. Spartan, New York.
[18] Setiono, R., & Loew, W. K. (2000). FERNN: An algorithm for fast extraction of rules from neural networks. *Applied Intelligence*.
[19] Shalev - Shwartz, S., & Ben - David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*.
[20] Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Verlag.
[21] Taiwo, O. A. (2010). Types of machine learning algorithms. In Yagang Zhang (Ed.), *New Advances in Machine Learning* (pp.3–31). InTech. ISBN: 978 - 953 - 307 - 034 - 6. Available at InTech open website: http: //www.intechopen. com/books/new - advances - in - machine - learning/types - of - machine - learning - algorithms
[22] Kanungo, T., & D. M. (2002). A local search approximation algorithm for k - means clustering. In *Proceedings of the Eighteenth Annual Symposium on Computational Geometry* (pp.1 - 8). Barcelona, Spain: ACM Press.
[23] Shepard, T. J., & P. J. (1998). Decision fusion using a multi - linear classifier. In *Proceedings of the International Conference on Multisource - Multisensor Information Fusion*.
[24] Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory* (2nd ed.). Springer Verlag.
[25] Witten, I. H., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques* (2nd ed.). Morgan Kaufmann Publishers. ISBN: 0 - 12 - 088407 - 0. Available from: [ftp: //93.63.40.27/pub/manuela. sbarra/Data Mining Practical Machine Learning Tools and Techniques - WEKA. pdf