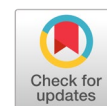


Constructing decision rules from naive bayes model for robust and low complexity classification



Nabeel Hashim Al-A'araji ^{a,1}, Safaa Obayes Al-Mamory ^{b,2}, Ali Hashim Al-Shakarchi ^{c,3,*}

^a Ministry of Higher Education, Baghdad, Iraq

^b University of Information Technology and Communications, Collage of Business Informatics, Baghdad, Iraq

^c University of Babylon, Collage of Information Technology, Hillah, Iraq

¹ nhkaghed@itnet.uobabylon.edu.iq; ² salmamory@uoitc.edu.iq; ³ ali.al-shakarchi@itnet.uobabylon.edu.iq

* corresponding author

ARTICLE INFO

ABSTRACT

Article history

Selected paper from The 2020 Global Research Conference (GRaCe'20), Trengganu-Malaysia (Virtually), 16-18 October 2020, <https://terengganu.uitm.edu.my/grace2020/>. Peer-reviewed by GRaCe'20 Scientific Committee and Editorial Team of IJAIN journal.

Received October 26, 2020

Revised November 10, 2020

Accepted March 15, 2021

Available online March 31, 2021

Keywords

Classification
Evaluation metrics
Naïve Bayes
Rule-Based classifier
Rule-set

A large spectrum of classifiers has been described in the literature. One attractive classification technique is a Naïve Bayes (NB) which has been relayed on probability theory. NB has two major limitations: First, it requires to rescan the dataset and applying a set of equations each time to classify instances, which is an expensive step if a dataset is relatively large. Second, NB may remain challenging for non-statisticians to understand the deep work of a model. On the other hand, Rule-Based classifiers (RBCs) have used IF-THEN rules (henceforth, rule-set), which are more comprehensible and less complex for classification tasks. For elevating NB limitations, this paper presents a method for constructing a rule-set from the NB model, which serves as RBC. Experiments of the constructing rule-set have been conducted on (Iris, WBC, Vote) datasets. Coverage, Accuracy, M-Estimate, and Laplace are crucial evaluation metrics that have been projected to rule-set. In some datasets, the rule-set obtains significant accuracy results that reach 95.33 %, 95.17% for Iris and vote datasets, respectively. The constructed rule-set can mimic the classification capability of NB, provide a visual representation of the model, express rules infidelity with acceptable accuracy; an easier method to interpreting and adjusting from the original model. Hence, the rule-set will provide a comprehensible and lightweight model than NB itself.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



1. Introduction

Classification is a data mining branch that is used by a variety of applications. Decision Tree (DT), Neural Network (NN), Support Vector Machine (SVM), Rule-Based Classifier (RBC), and Bayesian Classifiers (BC) are used for classification and pattern recognition tasks in many enterprises and industries [1]–[7]. These techniques can be grouped according to two major criteria. The first is the way the model has been built. For example, in DT, a model is derived and expressed as a tree. On the other hand, in RBC, the model is represented by decision rules, and BC represents a model via a set of equations or graphs (in the case of Bayesian Network). The second is the

model's complexity because some classifiers provide explanation capability for decision-making processes like (DT, RBC), while others produce black-box models such as (SVM, NN). This is so because they cannot explain how a learning result was reached [8].

Nowadays, we bear contemporary a real coup of data, which has wholly been pushed by way of superior storage technology and the Internet's increasing popularity. Finding actionable knowledge

within that huge data is humanly no longer possible, yet it demands sophisticated statistical and data mining techniques. SVM, NN, and NB are currently the most attractive classification techniques. Most benchmarking researches reveal that these models, in general, perform well among the trends of classification techniques due to capturing nonlinearities and probability membership in the case of SVM, NN [9], and NB, respectively. However, for SVM and NN, their strength is also their weakness, since they are regarded as black-box models [10]. In fact, from a knowledge representation perspective, classification models can be grouped into two main categories: white-box and black-box models [11]. The first provides a comprehensible form with the explanation capability about how the decision-making mechanism is handled. RBC and DT fall under this category. The second, which is an incomprehensible model, does not explain the decision-making process; SVM and NN return to this category [12].

NB [13] is a probabilistic model. This model makes its prediction depend on probability membership. It reflects the relationship between the independent and dependent variables to predict [14] [15]. A major drawback of this model is that, although the construction of a classifier is relatively easy and can be built without complicated parameters, it is time-consuming if applied on a large dataset. This is because every time NB needs to classify a new instance, the whole dataset requires to be reviewed for applying statistical equations to conduct classification. Rescanning the entire dataset is a very intensive step, especially if the dataset is relatively big [16]. Moreover, NB may remain difficult for non-statisticians in understanding the details of a model's operation and classification mechanism. Thus, it could be considered as a black-box model from the perspective of those who are not specialists in statistics.

On the other hand, RBC relies on its prediction on a more comprehensible rule-set for humans (i.e., White-box model) and easy to implement and tuned by an expert in a specific domain. If we employ the pros of both techniques (i.e., NB, RBC), we can get the classification power of NB and elevate its limitation. This is accomplished by taking the classification behavior of NB in a form classification rule (i.e., rule-set) which can be served as RBC. So, by considering the above-mentioned perspectives and to overcome the NB limitations, this paper attempts to present a useful method of constructing a rule-set, which can serve as RBC, from the NB model. The construction process is done in two steps. The first includes using a training dataset to train the NB model, while the second step uses the learned model and constructs a rule-set from it. The constructed rule-set can mimic the classification capability of NB, provides a visual representation of the model, expresses in fidelity form, shows acceptable accuracy, which is easier to interpret and adjust from the original model. Moreover, it (i.e., rule-set) averts the rescanning dataset and applies a set of equations to classify instances, which is an expensive step if the dataset is relatively large. When a new instance is considered for classification, a rule-set is scanned to determine the triggered rules that are satisfied by instance. After that, a rule engine will fire the appropriate rule to get the instance's class.

The experiments have conducted three different datasets (Iris, WBC, Vote). The first two of which have numerical attributes, while the last one is restricted with nominal attributes. Four vital rule evaluation metrics have been projected to the constructed rule-set, named (Coverage, Accuracy, M-Estimate, and Laplace). The experimental results show that the generated rule-set has simple and pure formalism, which could be used as classification rules (i.e., RBC) with relatively high accuracy. In some datasets with specific experimental settings, the rule-set gets surprised accuracy results for Iris and vote datasets, respectively.

This paper is organized as follows: Section Two discusses related works and gives an overview of rule construction methods in the literature. Section Three demonstrates short preliminaries of NB and RBC. The proposed rule construction method will be described in Section Four. Experiments, datasets, results, and other configurations have been demonstrated in section five. Finally, the paper ends with conclusions and suggestions for future works.

2. Method

This section provides a simple overview of two attractive classifiers that are relevant to the proposed work. This first one is the probabilistic Naïve Bayes Classifier, and the second is Rule-Based Classifier. A brief account of the work's mechanism, Pros, and cons for each model will be provided.

2.1. Related works

To the best of our knowledge, Andrews, Diederich, and Tickle [17] is the first work that divides rule extraction method from ANN, that can be also applied to other models into two main modes: decompositional and pedagogical (also called learning-based). The principle of discriminate these modes relies on how to extract rules from the learned model. In decompositional mode, the focus is on extracting rules by browsing the deeply of model components. While in pedagogical mode, the concentration will be to cooperate with another machine learning algorithm that has facilitation of explanation ability which means that the original model (i.e. black-box model) has been used to produce elements to be used in the second model that generating the rules as for output. The difference between decompositional and pedagogical is schematically illustrated in Fig. 1.

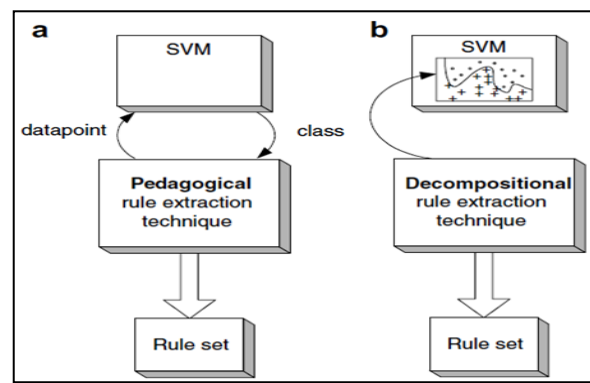


Fig. 1. SVM rule extraction technique, Pedagogical (a) and Decompositional (b) [18]

Rule extraction from classification models has been covered in different publications. Table 1 shows a summarized overview rule of extraction techniques proposed by the authors. The first set of works shows the decompositional (D) approached to rule construction. Then, the pedagogical (P) rule construction approaches are listed.

Table 1. Overview of Rule Extraction Algorithms

Ref	Algorithm/Approach	Summary
[19]	NOFM/D	NOFM is a method that extracts an accurate and comprehensible rule set from learned KNNs
[20]	KL/D	KL is able to deal with single and multi-layer NN.
[21]	-/D	transform decision trees into a set of production rules
[22]	SVM+Prototypes/D	Clustering algorithm has been used to determine prototype vectors
[23]	-/D	Extract rule-set from NB model
[16]	RNBC/D	Rules are generated based on Naïve Bayesian classifier
[24]	-/P	Extracting rule set from Support Vector Machine (SVM)
[25]	Re-RX/P	Extract rule from NN, working with discrete and continuous attributes without a need for discretization
[26]	TREPAN/P	Extracting comprehensible, symbolic representations from trained neural networks.
[27]	GEX/P	Extracting rule from trained NN.
[28]	-/P	Extracting rule from SVM

Towell and Shavlik [19] propose and evaluate a method for extracting expert- comprehensible rule set from learning NNs, called NOFM. The name NOFM comes from the concept of *IF (M of antecedents are true) THEN Results obtained by extracting rules closely mimic the original network's accuracy.*

Fu [20] presents an algorithm called Knowledgetron (KT), which extracts the rule set from a backpropagation network, and it can deal with single-layer and multi-layer networks. Another work was done by Quinlan [21], which aimed to transform decision trees into production rules. The proposed method makes use of a training set from which the decision tree was built. The author finds out the produced rules are more accurate when predicting unseen instances. Núñez, Angulo, and Català [22] propose a method to extract a rule from the Support Vector Machine named SVM+ Prototypes method. The procedure of extracting rules uses support vector and geometric operation to define ellipsoids in the input space, which are converted later into a rule-set.

Few works in the literature are concerned with construct a rule-set from Naïve Bayes. An attractive work close to our work was conducted by Śnieżyński [23], which presents a new method that converts the Naïve Bayes model to a rule-set. The researcher's major concern is to prove the possibility of converting the NB model into rules with relatively high accuracy. Another paper was presented by Alashqur [16] in which he proposes a new approach called Rule-based Naïve Bayesian Classifier (RNBC). The author uses a simple three steps for constructing a classification rule set. Diederich and Barakat [24] propose a novel approach for extracting the rule set from SVM. Their method simulates the original model via a learning task that utilized a training dataset and updated or different datasets to convert SVM into a set of non-overlapping rules. Setiono *et al.* [25] introduce a rule extraction algorithm called Recursive-Rule eXtraction (Re-RX), which aims to extract rules from NN. This algorithm can deal with discrete and continuous variables and work efficiently with large datasets. Credit risk datasets have been used to validate the extracted rules.

2.2. Naïve Bayes Classifier

NB considered amongst the foremost efficient classification models depended on the Bayesian Theorem [29][30]. The NB classifier focuses the conditional probability of instances in the training data; that is, for each attribute A_i of target C , the likelihood of the target is calculated based on its attributes $A_1, A_2 \dots A_n$, hence, the target has been predicted along with maximum posterior probability[31].

The key idea of NB classifier is that it relies on the independent postulate for the attributes given the target class C . hence, all attributes of a certain instance are conditionally independent given target C [32], and this trait is considered as a weakness criterion if the hypothesis is not verified. Fig. 2 is illustrated the relationship between attributes and the target class of the NB classifier.

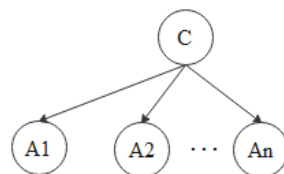


Fig. 2. Naive Bayes Structure

The Bayes theorem demonstrates in (1).

$$P(A|B) = P(B|A).P(A)/P(B) \quad (1)$$

Whereas P represents the probability, $P(A|B)$ represents a posterior probability of the target when the attribute is specified. $P(A)$ is the preceding probability of a target, and $P(B|A)$ represents a Likelihood of instance A given target C , whereas $P(B)$ represents the past probability of the predictor. The divisor $P(B)$ is eliminated when calculating $P(A|B)$ because it considers as fixed (constant) to all targets, as shown in (2) [24][25].

$$P(A|B) \sim P(B|A).P(A) \quad (2)$$

The symbol "-" refers to that LHS (i.e., Left Hand Side) is prorated to the RHS (i.e., Right Hand Side). The target that has a maximum posterior probability will be considered as a final class for the overall prediction targets [16][31][33]. It is worth mentioning that NB is computationally fast if it is applied to relatively small data, yet it works well with large data at the expense of the execution time.

2.3. Rule-Based Classifier

A pivotal means of interpreting data or information is via rules. RBC utilizes a set of rules for classification tasks. A body of the rule is formed as *IF condition THEN conclusion*.

The IF part (i.e., LHS) refers to the rule antecedent (which may have one or more attributes as a condition); on the other hand, THEN part (i.e., RHS) represents the rule eventual (rule consequent that represents the target label) [34]. A rule covers an example whenever the rule antecedent is met. The coverage rule is calculated by dividing the number of instances satisfied by the overall number of instances in (3).

$$\text{Coverage } (R) = \frac{\text{Number of records satisfied by rule}}{\text{Total number of records}} \quad (3)$$

If the same example triggers no two rules in the data, the rule set is mutually exclusive. If it exists, it indicates a single rule for each feature combination in the feature space. The rule set is considered exhaustive, which means that each feature space segment has a rule. The preceding criteria guarantee that each pattern is covered by a single rule [35]. Nevertheless, this is frequently an ideal situation, and not all rule-based classifiers exhibit this property. Instances in the dataset that are not launched via any rule may be created due to a violation of the exhaustive property. However, if the mutually exclusive property is violated, multiple rules may be triggered by the same instance, and once the triggered rules generate a different target label, a rule conflict will occur. In this case, a strategy for resolving conflict should be issued.

2.4. Rule of construction methodology

A derivation set of rules from the NB classifier is applicable. For constructing the rules, the unpretentious method is to extract a rule utilizing each attribute of instances. The straightforward form could be illustrated as follows:

$$\text{Rule } (R): X_i \text{====} > Y_k : L$$

$X_i(X_1, X_2, X_3 \dots, X_n)$ refers to the instance with its attributes, and $Y_k(Y_1, Y_2, \dots, Y_k)$ refers to the target label. The ruling power (i.e., rule strength) is represented by the notation (L). If the dataset is sufficiently large, this rule extraction method will generate a large number of rules. A parameter controls rule construction, and the L parameter is required to reduce a large number of rules. By hiring L, it is possible to eliminate weak rules that degrade the model's performance, thereby increasing the classification model's accuracy. As a result of the above, the rule construction algorithm-based [23] could be viewed in Fig. 3.

```

Algorithm: The rule resulted from the NB
Input: The probability of NB conditional
       p(Xi|C), threshold T
Output: R(Rule-set)
1   Begin
2   for each instance Xi do
3   for each value of class C do
4   P: = p(Xi|Ck)
5   if (P>=T) OR (P<=1-T) then
6   R=R ∪ (Xi -->Ck: f(p))
7   end
8   end
9   End

```

Fig. 3. The constructed Rule Set algorithm based on NB

The algorithm takes a Naive Bayes conditional probability $P(X|C)$ and a threshold T as inputs and produces a set of IF-Then rules as for output. The number of produced rules will be limited by the threshold (T). The probabilistic is defined by the function $f(p)$ in line (6). It can take two different forms. The first is a probabilistic mark, which is $f(P)=P$; in this case, if many rules cause an example and the rules' consequents belong to the same class, the consequents should be aggregated by multiplying the labels of the rules. Another alternative is to discretize the labels by applying the $f(p)=\text{Round}(p)$ [23].

The condition in line (5) is used in the preceding algorithm when the input domain is constrained to two values, namely X_i^j values with a conditional probability close to either 1 or 0. If the domain contains more than two values, the condition should then be substituted by the Entropy.

For instance, Entropy condition X_i for class Y_k can be defined in (4) as follows:

$$E(i,k) = \sum_{j=1}^{|D_{X_i}|} -P(X_i = x_j^i | Y = y_i) \log_2 P(X_i = x_j^i | Y = y_i) \quad (4)$$

Where $E(i,k)$ refers to the probability distribution entropy, D_{X_i} is input domain. The condition in line (5) must take the form: $E < T$.

The constructed rules with discrete strength are considered more comprehensible than those obtained via a probabilistic one. Rules that have the exact consequent and exact strengths can be aggregated in one classification rule by connecting the OR operation rules.

3. Results and Discussion

3.1. Datasets

This research was conducted on three different datasets: Iris, Wisconsin Breast Cancer (WBC), and Congressional Voting Records Data Set (Voting). These datasets are available at the UCI repository for machine learning tasks [36]. The Voting dataset includes nominal attributes whereas the remainder includes numerical attributes. Table 2 epitomizes these datasets.

Table 2. Datasets Summary

Dataset	# Cass	Missing value	# Nominal attribute	# Numeric attribute	# Instance
Iris	3	None	0	4	150
WBC	2	Several	0	9	699
Voting	2	Many	19	0	435

The first famed dataset contains three classes, each of which represents the species of the iris. This dataset is considered a balanced dataset since classes are equally distributed. WBC includes patient information represented by two classes that take either benign or malign of cancer. Whereas in a Voting dataset, the target label refers to the party of voter records of Congressmen. The class distribution of WBC and Vote datasets reveal that they are unbalanced datasets.

Weka is a tool that is oriented to solve data mining issues. It includes an aggregation of machine learning algorithms related to classification, clustering, regression, feature selection, etc. Weka is fully written in Java and provides Java API, hence, their algorithms could be applied straightway to the selected dataset or called based on the Java code [37]. The accuracy of NB for our datasets on training data is 96%, 96.56%, and 90.34% for Iris, WBC, and Vote, respectively. Preprocessing operations are conducted on Iris and WBC. In these two datasets, the unsupervised discretization (binning) process on the numeric attributes to divide attributes into groups has been applied, then convert them to discrete counterparts and facilitates the process of rule construction.

3.2. Experiments

Equal width(EW) and *Equal frequency(EF)* are two unsupervised binning methods that have been used in the discretization process. In each of these methods, a different value of an important parameter called bin (which describe the interval of data, subdivision of equal size in the case of equal width or divide the data into K groups, each of them contain approximately the same number of values in the case of equal frequency) that control discretization process. In [23], just *EW* with two bins and a class-based ordering conflict resolution strategy is employed. The proposed work in this research has adapted class-based ordering and size-ordering rule conflict strategies.

Applying the proposed algorithm, rule sets for every dataset using thresholds with a range (0.6 - 0.9), discrete $f(p)=\text{Round}(p)$ as a rule labeling (strength) have been constructed. Indeed, threshold T can range between (0.1- 0.9). The lower value produces maximum rule-set with low accuracy measure, while the upper values gave the smallest count of rules with comparatively rising accuracy. Table 3 shows the number of constructing rules for each dataset with two binning values (2 and 3) and threshold (T) with ranges (0.6 - 0.9). It is worth noting that only the most powerful rules will be taken when applying rules for classification, i.e., the rules with strength equal (1.0).

Table 3. Number of Constructed Rules for Dataset (Without Default Rules)

T	Iris, bin=2		WBC, bin=2		Voting
	<i>EW</i>	<i>EF</i>	<i>EW</i>	<i>EF</i>	
0.6	22	18	26	34	49
0.7	18	16	21	32	43
0.8	14	16	20	28	30
0.9	12	12	18	12	10

T	Iris, bin=3		WBC, bin=3	
	<i>EW</i>	<i>EF</i>	<i>EW</i>	<i>EF</i>
0.6	33	35	49	48
0.7	30	31	39	41
0.8	26	25	30	33
0.9	24	24	27	18

The plot in Fig. 4 shows the relationship between the probability threshold T and the number of constructing rules with two binning techniques (Equal width, Equal frequency). From the sketch, it is clear that the relationship between threshold T and generating rules is reversible. That is, whenever increase T, the number of generating rules will decrease.

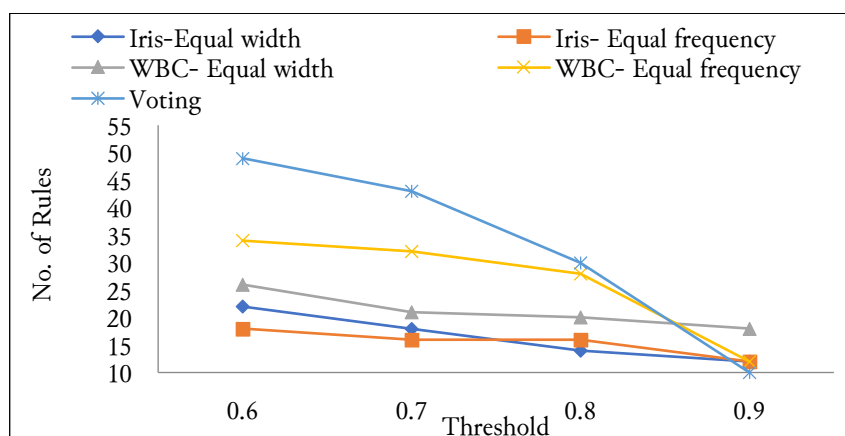


Fig. 4. Number of Generated Rules Per Threshold T With Bin=2

The constructed rule-sets have been applied as classification rules (i.e., RBC) to the three datasets mentioned in Table 2 utilizing software which the authors develop. If no rule matches an instance during classification, this case is handled as missing-instance and can be solved by adding a default rule.

As mentioned in subsection (3.2), not all RBCs are mutually exclusive, which leads to the conflict of rules. This implies the use of some conflict resolution strategies. The implementation of classification rules will adopt two different strategies. They are (A) *class-based ordering* and (B) *size-ordering*. The former relies on class, i.e., the class of higher priority is executed first. The latter assigns the highest priority to the triggering rules with the "toughest" requirement (i.e., with the most attribute test). Table 4 highlights the accuracy after applying the rule-set on Iris and WBC datasets, with T ranged (0.6_0.9), using bin equal to 3 and two different rule conflict resolution strategies.

Table 4. Accuracy of Generated Rule-Set for (Iris, WBC) in Different Configuration

Iris Dataset				
<i>T</i>	<i>Binning</i>	<i># Rules</i>	<i>Conflict Strategy</i>	<i>Accuracy</i>
0.6	EW	10	A	49.33%
			B	71.33%
	EF	10	A	74.67%
			B	78.67%
0.7	EW	9	A	49.33%
			B	71.33%
	EF	9	A	80.0%
			B	80.0%
0.8	EW	7	A	90.67%
			B	94.67%
	EF	7	A	91.33%
			B	95.33%
0.9	EW	6	A	90.67%
			B	94.67%
	EF	7	A	91.33%
			B	95.33%
WBC Dataset				
0.6	EW	11	A	68.24%
			B	68.24%
	EF	12	A	74.11%
			B	74.11%
0.7	EW	10	A	68.24%
			B	68.24%
	EF	11	A	76.97%
			B	76.97%
0.8	EW	9	A	68.53%
			B	68.53%
	EF	8	A	76.82%
			B	76.82%
0.9	EW	8	A	68.53%
			B	68.53%
	EF	2	A	78.97%
			B	78.97%

The bin number is a vital parameter that affects the accuracy of RBC. The experiments adapted two values of this parameter (bin=2 or 3). The reason for choosing these values is that they produce a rule set with high accuracy metric when used for classification. What is remarkable in the results is that the accuracy of RBC increases with the increase of T. In the Iris dataset, significant results have been obtained in accuracy metrics that reach 94.67% and 95.33% in EW and EF, respectively, with T=0.9. If the previous configuration has directly been applied on Naïve Bayes (i.e., discretize with bin=3), the

obtained accuracy is 94%. The conflict resolution strategy also has an apparent effect on accuracy metrics. Fig. 5 reveals that size ordering is more accurate than the class-based ordering strategy when applying to the Iris dataset.



Fig. 5. Accuracy Comparison Between Class-Based Ordering And Size Ordering With EW Binning on Iris Dataset

Equal frequency discretization produces a rule set that shows the superior accuracy compared to the Equal width in all datasets for both class-based and size ordering conflict resolution strategies. The plot in Fig. 6 shows this situation in the WBC dataset.

Table 5 summarizes the results of applying a constructed rule set on different threshold T with class-based and size-ordering strategies on the Vote dataset. In Vote, which is a nominal attributed dataset, the class-based ordering shows high classification accuracy than size-ordering. With T=0.9, which produces just 4 strength rules, a significant accuracy that reached 95.17% has been obtained, compared with 90.34 %, which have been obtained from applying NB directly to the dataset.

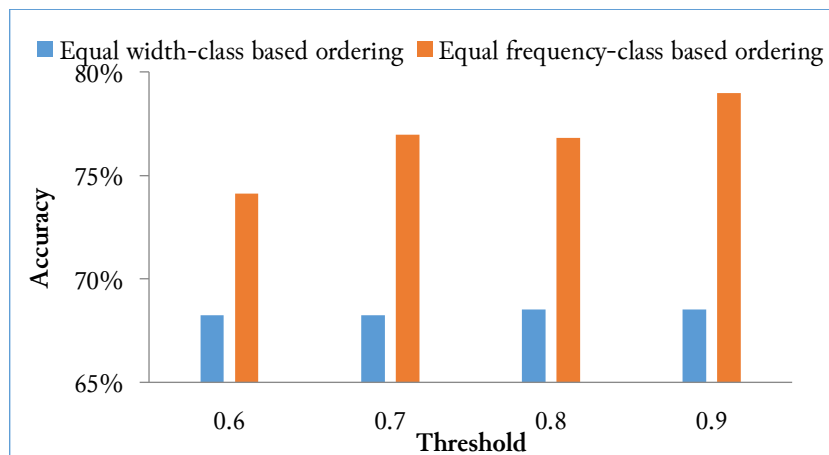


Fig. 6. Accuracy Comparison Between Equal Width and Equal Frequency on WBC Dataset

Table 5. The Accuracy of Generated Rule Set For Voting Dataset in Different Configuration

Vote Dataset			
T	# Rules	Conflict strategy	Accuracy
0.6	23	A	75.86%
		B	43.45%
0.7	20	A	87.13%
		B	43.45%
0.8	13	A	90.34%
		B	46.21%
0.9	4	A	95.17%
		B	76.32%

Table 6 demonstrates the evaluation of the rule set for the Iris dataset that is constructed with the configuration of bin=3, T=0.9, and (EF) discretization scheme. The evaluation is done via famed metrics (Coverage, Accuracy, Laplace, M-Estimate).

Table 6. Evaluation of iris Rule-Set

Rules	Coverage	Accuracy	Laplace	M_Estimate
if (sepalength<5.5) =====>Iris-setosa:1.0	34.67%	86.54%	83.64%	81.82%
if (petallength<2.5) =====>Iris-setosa:1.0	33.33%	100.0%	96.23%	94.34%
if (petalwidth<0.8) =====>Iris-setosa:1.0	33.33%	100.0%	96.23%	94.34%
if (petallength>=2.5 && petallength <4.8) =====>Iris-versicolor:1.0	30.0%	97.78%	93.75%	91.67%
if (petalwidth>=0.8&& petalwidth <1.6) =====>Iris-versicolor:1.0	32.0%	93.75%	90.2%	88.24%
if (petallength>=4.8) =====>Iris-virginica:1.0	36.67%	89.09%	86.21%	84.48%
if (petalwidth>=1.6)=====>Iris-virginica:1.0	34.67%	34.67%	87.27%	85.45%

The set of rules shown in Table 6 for the Iris dataset that serves as an RBC is more comprehensible to humans to interpret than the original NB probability model. A comparison of accuracy related to RBC constructed from NB is plotted in Fig. 7.

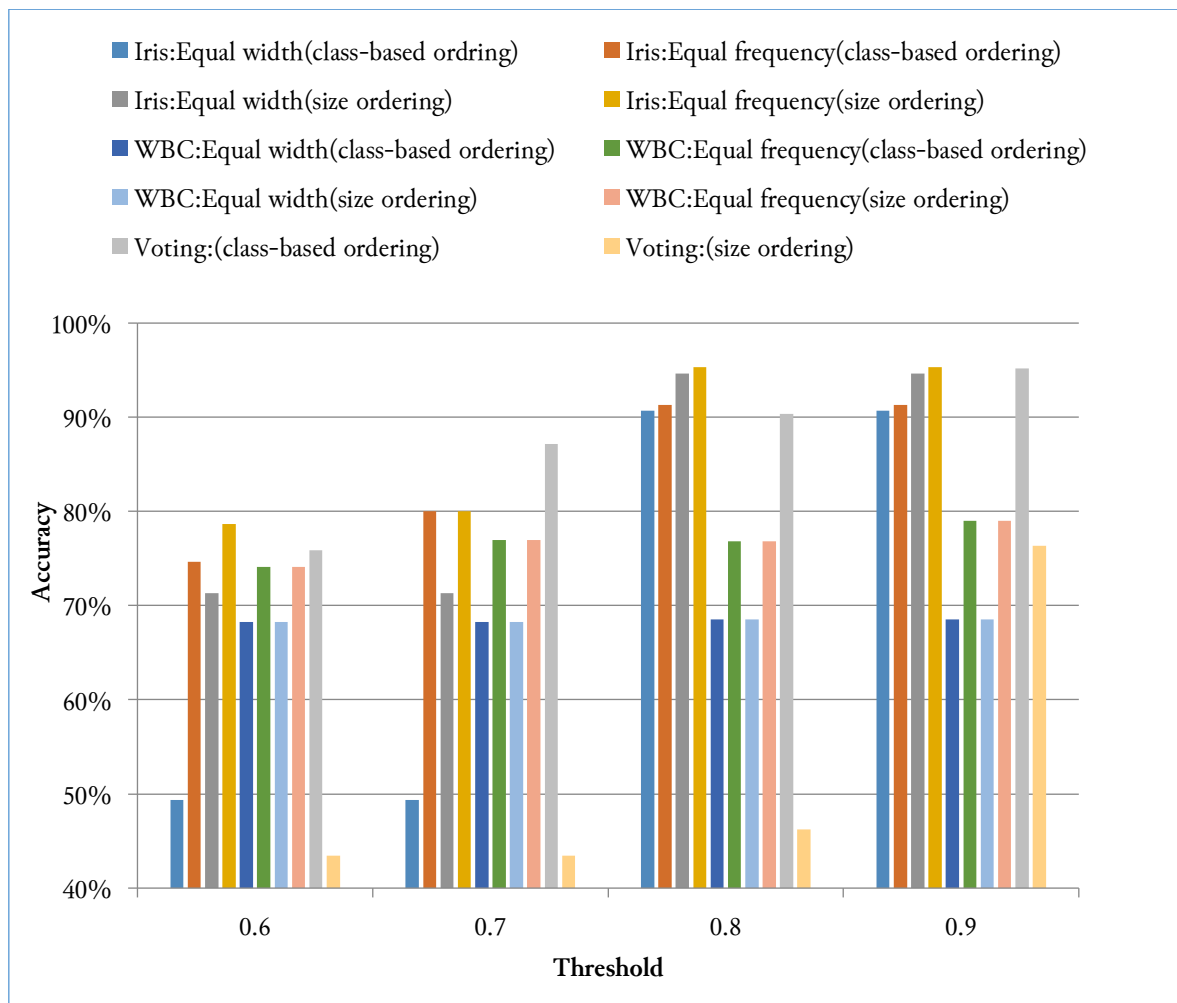


Fig. 7. Accuracy Comparisons For Constructed Rule-Set

4. Conclusion

This paper represents an attempt to initiate a method for constructing a decision rule-set from the Naïve Bayes classifier that is easily interpreted and is more comprehensible by a human. Two discretization methods (Equal width, Equal frequency) have been employed for preprocesses numerical datasets to facilitate rules construction. Experimental results show that the rule set constructed from Naïve Bayes is pure, with classification accuracy relatively high compared to the original model. Implement the rule construction method onto the different problem domains has been planned. Also, improving the accuracy of a constructed rule set via proposing a new strategy of implementing the classification rules has been attempted.

Acknowledgment

This research is supported by the College of Information Technology/University of Babylon.

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding statement. The authors received no specific funding for this work

Conflict of interest. The authors declare no conflict of interest.

Additional information. The datasets used in this research are available at [UCI](#) repository.

References

- [1] S. Garg and S. Batra, "A novel ensemble technique for anomaly detection," *Int. J. Commun. Syst.*, vol. 30, no. 11, p. e3248, 2017, doi: [10.1002/dac.3248](#).
- [2] R. González Perea, E. Camacho Poyato, P. Montesinos, and J. A. Rodríguez Díaz, "Prediction of irrigation event occurrence at farm level using optimal decision trees," *Comput. Electron. Agric.*, vol. 157, pp. 173–180, 2019, doi: [10.1016/j.compag.2018.12.043](#).
- [3] M. A. Ferrag, L. Maglaras, A. Ahmim, M. Derdour, and H. Janicke, "RDTIDS: Rules and Decision Tree-Based Intrusion Detection System for Internet-of-Things Networks," *Futur. Internet*, vol. 12, no. 3, p. 44, 2020, doi: [10.3390/fi12030044](#)
- [4] J. Nalepa and M. Kawulok, "Selecting training sets for support vector machines: a review," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 857–900, 2019, doi: [10.1007/s10462-017-9611-1](#)
- [5] Q. Wu, Z. Ma, G. Xu, S. Li, and D. Chen, "A novel neural network classifier using beetle antennae search algorithm for pattern classification," *IEEE access*, vol. 7, pp. 64686–64696, 2019, doi: [10.1109/ACCESS.2019.2917526](#)
- [6] T. Alasalmi, J. Suutala, H. Koskimäki, and J. Röning, "Better Classifier Calibration for Small Data Sets," *arXiv Prepr. arXiv2002.10199*, 2020, doi: [10.1145/3385656](#)
- [7] J. Li, M. Gao, and R. D'Agostino, "Evaluating classification accuracy for modern learning approaches," *Stat. Med.*, vol. 38, no. 13, pp. 2477–2503, 2019, doi: [10.1002/sim.8103](#)
- [8] N. Barakat and J. Diederich, "Eclectic Rule-Extraction from Support Vector Machines," *Eng. Technol.*, vol. 2, no. 1, pp. 331–334, 2006, doi: [10.1080/00029157.1964.10402393](#).
- [9] D. Martens, B. B. Baesens, T. Van Gestel, and T. Van Gestel, "Decompositional rule extraction from support vector machines by active learning," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 2, pp. 178–191, 2009, doi: [10.1007/978-3-540-75390-2](#).
- [10] C. Panigutti, R. Guidotti, A. Monreale, and D. Pedreschi, "Explaining multi-label black-box classifiers for health applications," in *International Workshop on Health Intelligence*, 2019, pp. 97–110, doi: [10.1109/TKDE.2008.131](#)
- [11] G. L. Pappa and A. Freitas, *Automating the Design of Data Mining Algorithms*, 2010, doi: [10.1007/978-3-642-02541-9](#), doi: [10.1007/978-3-030-24409-5_9](#)
- [12] O. Loyola-González, "Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view," *IEEE Access*, vol. 7, pp. 154096–154113, 2019, doi: [10.1007/978-3-642-02541-9](#)

- [13] M. M. Saritas and A. Yasar, "Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification," *Int. J. Intell. Syst. Appl. Eng.*, vol. 7, no. 2, pp. 88–91, 2019, doi: [10.1109/ACCESS.2019.2949286](https://doi.org/10.1109/ACCESS.2019.2949286)
- [14] A. Sahasrabudhe, S. Naikade, A. Ramaswamy, B. Sadliwala, and P. Futane, "Survey on Intrusion Detection System using Data Mining Techniques," *Int. Res. J. Eng. Technol.*, vol. 4, no. 5, pp. 1780–1784, 2017, doi: ISSN: 2395 -0056, doi: [10.18201/ijisae.2019252786](https://doi.org/10.18201/ijisae.2019252786)
- [15] K. Kumar, R. Jha, and S. Afroz, "Data Mining Techniques for Intrusion Detection : A Review," *Int. J. Adv. Researcg Comput. Commun. Engineering*, vol. 3, no. 6, pp. 6938–6942, 2014, Available at: [Google Scholar](#)
- [16] A. Alashqur, "A Novel Methodology for Constructing Rule-Based Naïve Bayesian Classifiers," *Int. J. Comput. Sci. Inf. Technol.*, vol. 7, no. 1, pp. 139–151, 2015, doi: [10.5121/ijcsit.2015.7114](https://doi.org/10.5121/ijcsit.2015.7114).
- [17] R. Andrews, J. Diederich, and A. B. Tickle, "Survey and critique of techniques for extracting rules from trained artificial neural networks," *Knowledge-Based Syst.*, vol. 8, no. 6, pp. 373–389, 1995, doi: [10.1016/0950-7051\(96\)81920-4](https://doi.org/10.1016/0950-7051(96)81920-4).
- [18] D. Martens, B. Baesens, T. Van Gestel, and J. Vanthienen, "Comprehensible credit scoring models using rule extraction from support vector machines," *Eur. J. Oper. Res.*, vol. 183, no. 3, pp. 1466–1476, 2007, doi: [10.1016/j.ejor.2006.04.051](https://doi.org/10.1016/j.ejor.2006.04.051).
- [19] G. Towell and J. W. Shavlik, "Interpretation of artificial neural networks: Mapping knowledge-based neural networks into rules," in *Advances in neural information processing systems*, 1992, pp. 977–984, Available at: [Google Scholar](#)
- [20] L. Fu, "Rule learning by searching on adapted nets," *Proc. 9th Natl. Conf. Artif. Intell. - Vol. 2*, vol. 91, pp. 590–595, 1991. Available at: [Google Scholar](#)
- [21] J. R. Quinlan, "Generating production rules from decision trees," in *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 1987, vol. 30107, pp. 304–307. Available at: [Google Scholar](#)
- [22] H. Núñez, C. Angulo, and A. Català, "Rule extraction from support vector machines.," in *Esann*, 2002, pp. 107–112. Available at: [Google Scholar](#)
- [23] B. Śnieżyński, "Converting a Naive Bayes Models with Multi-valued Domains into Sets of Rules," *Int. Conf. Database Expert Syst. Appl.*, vol. 229, pp. 634–643, 2006, doi: [10.1007/11827405_62](https://doi.org/10.1007/11827405_62).
- [24] J. Diederich and N. Barakat, "Hybrid rule-extraction from support vector machines," in *2004 IEEE Conference on Cybernetics and Intelligent Systems*, 2004, vol. vol.2, pp. 1270–1275, doi: [10.1109/ICCIS.2004.1460774](https://doi.org/10.1109/ICCIS.2004.1460774)
- [25] R. Setiono, C. Mues, and B. Baesens, "Risk management and regulatory compliance: A data mining framework based on neural network rule extraction," *ICIS 2006 Proc. - Twenty Seventh Int. Conf. Inf. Syst.*, pp. 71–86, 2006. Available at: [Google Scholar](#)
- [26] M. W. Craven and J. W. Shavlik, "Extracting Tree-Structured Representations of Trained Networks," *Adv. Neural Inf. ...*, vol. 8, p. 7, 1996. Available at: [Google Books](#)
- [27] U. Markowska-Kaczmar and M. Chumieja, "Discovering the mysteries of neural networks," *Int. J. Hybrid Intell. Syst.*, vol. 1, no. 3–4, pp. 153–163, 2004, doi: [10.3233/HIS-2004-13-404](https://doi.org/10.3233/HIS-2004-13-404)
- [28] F. Chen, "Learning Accurate and Understandable Rules From Svm Classifiers," Ph.D. thesis, Science: School of Computing Science, 2004. Available at: [Google Scholar](#)
- [29] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *J. Inf. Sci.*, vol. 44, no. 1, pp. 48–59, 2018, doi: [10.1177/0165551516677946](https://doi.org/10.1177/0165551516677946).
- [30] S. A. Mostafa *et al.*, "Examining multiple feature evaluation and classification methods for improving the diagnosis of Parkinson's disease," *Cogn. Syst. Res.*, vol. 54, pp. 90–99, 2019, doi: [10.1016/j.cogsys.2018.12.004](https://doi.org/10.1016/j.cogsys.2018.12.004)
- [31] M. Goldszmidt, "Bayesian Network Classifiers," *Wiley Encycl. Oper. Res. Manag. Sci.*, vol. 29, no. 2–3, pp. 131–163, 2011, doi: [10.1002/9780470400531.eorms0099](https://doi.org/10.1002/9780470400531.eorms0099), doi: [10.1002/9780470400531.eorms0099](https://doi.org/10.1002/9780470400531.eorms0099)
- [32] M. Možina, J. Demšar, M. Kattan, and B. Zupan, "Nomograms for Visualization of Naive Bayesian Classifier," in *European Conference on Principles of Data Mining and Knowledge Discovery*, 2010, vol. 3202,

pp. 337–348, doi: [10.1007/978-3-540-30116-5_32](https://doi.org/10.1007/978-3-540-30116-5_32).

- [33] I. Technologies, A. S. Kapse, V. P. Kshirsagar, A. Kapse, and M. B. Nagori, "A Survey on Intrusion Network Detection System Using Data Mining Techniques," vol. 2, no. 3, pp. 1253–1256, 2011. Available at: ijcsit.com
- [34] M. R. Karlsen and S. Moschoyiannis, "Learning condition--action rules for personalised journey recommendations," in *International Joint Conference on Rules and Reasoning*, 2018, pp. 293–301, doi: [10.1007/978-3-319-99906-7_21](https://doi.org/10.1007/978-3-319-99906-7_21)
- [35] A. R. Webb and K. D. Copsey, *Statistical Pattern Recognition*. WILEY, 2011, doi: [10.1002/9781119952954](https://doi.org/10.1002/9781119952954)
- [36] C. L. Blake and C. J. Merz, "UCI Repository of machine learning databases," *University of California*. Accessed, p. <http://archive.ics.uci.edu/ml/>, 2018. Available at: [Google Scholar](https://scholar.google.com/)
- [37] E. Frank, M. A. Hall, and I. H. Witten, *The WEKA workbench*, 2017, doi: [10.1016/b978-0-12-804291-5.00024-6](https://doi.org/10.1016/b978-0-12-804291-5.00024-6).