The effect of logical permutation in 2 satisfiability reverse analysis method

Cite as: AIP Conference Proceedings **2266**, 040013 (2020); https://doi.org/10.1063/5.0019158
Published Online: 06 October 2020

Mohd Shareduwan Mohd Kasihmuddin, Mohd. Asyraf Mansor, Md Faisal Md Basir, et al.

The Effect of Logical Permutation in 2 Satisfiability Reverse Analysis Method

Mohd Shareduwan Mohd Kasihmuddin[1, b)], Mohd. Asyraf Mansor[2, a)], Md Faisal Md Basir[3, c)] Siti Zulaikha Mohd Jamaludin [1, d)] and Saratha Sathasivam[1, e)]

[1]*School of Mathematical Sciences, Universiti Sains Malaysia, 11800 Minden, Pulau Pinang, Malaysia*
[2]*School of Distance Education, Universiti Sains Malaysia, 11800 USM, Pulau Pinang, Malaysia*
[3]*Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 UTM, Johor, Malaysia*

[a)]Corresponding author: asyrafman@usm.my
[b)]shareduwan@usm.my
[c)]mfaisalmbasir@utm.my
[d)]szulaikha.szmj@usm.my
[e)]saratha@usm.my

Abstract. The dynamical behaviors of logic mining in real datasets are strongly dependent by its logical structure. In this case, logical rule that has been embedded to neural network has long suffered from a lack of interpretability and accuracy. This has severely limited the practical usability of logic mining. Logical permutation is a definitive finite arrangement of attributes that makes 2SAT became true. It was believed that the effect of permutation will increase the accuracy of the system. In this paper, we presented the effect of logical permutation in logic mining (2SATRA) integrated with recurrent Hopfield Neural Network (HNN). Several benchmark datasets will be used to validate the effect of logical permutation. It has been shown that 2SATRA with different permutation will results in improvement in terms of accuracy value. This finding will lead to a better understand of 2SATRA in doing real life datasets.

INTRODUCTION

Hopfield Neural Network (HNN) has been invented by Hopfield and Tank [1] in solving extensive optimization problem ranging from constraint optimization [2] to complex system [3]. A typical HNN consist of interconnecting unit called neuron that fires asynchronously according to the learned pattern. Due to several topological issues in HNN such as spurious effect [4] and neuron oscillation [5], logical rule has been embedded to create an optimal synaptic weight management system [6]. The implementation of logic programming in HNN has been a pioneer work of Abdullah [6] and being applied extensively in logic mining via Reverse Analysis method. The flexibility of Reverse Analysis method was further developed by integrating more systematic logical rule called 2 Satisfiability Reverse Analysis Method (2SATRA) [7]. This development leads to another breakthrough in logic mining when the proposed method was reported to extract the optimal logical rule in several established medical data such as diabetes dataset [7] and cardiovascular dataset [8]. The success story continues when the proposed logic mining method has been implemented into social media dataset such as Facebook [9]. All the mentioned works showed acceptable accuracy $(\geq 70\%)$ and the extracted logic from the dataset were considered significant in the field of prediction, classification and knowledge extraction.

Despite these empirical successes, the structure of 2SATRA incorporated with HNN is poorly understood. The development of the optimal logic mining poses an important question; what is the effect of different assignment of attributes in the logical rule towards the accuracy of the logic mining? The answer of this question would increase the

Proceedings of the 27th National Symposium on Mathematical Sciences (SKSM27)
AIP Conf. Proc. 2266, 040013-1–040013-8; https://doi.org/10.1063/5.0019158
Published by AIP Publishing. 978-0-7354-2029-8/$30.00

capability of 2SATRA in choosing the best extracted logical rule for a given dataset. The effect of neuron permutation in HNN provides a fruitful insight in terms of neuron structure. Generally, neuron in HNN will be activated based on the given input and undergoes learning process in order to obtain the synaptic weight. The obtained synaptic weight will be utilized to excite the input neuron based on the embedded pre-determined logical rule. As a result, the induced logical rule will lose flexibility and fail to highlight non-contributing attributes. This paper concerned with the logical permutation effect in HNN toward the capability of 2SATRA in extracting optimal logical rule. In particular, this paper examined the quality of the HNN model for different logical rule in simulated datasets. The result of this paper showed that: by increasing the neuron permutation, 2SATRA will achieve higher accuracy compared to the conventional 2SATRA.

2 SATISFIABILITY REPRESENTATION

Satisfiability (SAT) is a representation of determining the interpretation that satisfy the given Boolean Formula. According to [10], SAT is proven to be NP-complete problem and has been included to cover wide range of optimization problem. Extensive research on SAT leads to the creation of variant SAT which is 2 Satisfiability (2SAT). Generally, 2SAT consist of the following properties [11]:

(a) A set of defined x variables, $q_1, q_2, q_3, ..., q_x$ where $q_i \in \{-1,1\}$ that exemplify False and True respectively.

(b) A set of literals. A literal can be variable or the negation of variable such that $q_i \in \{q_i, \neg q_i\}$.

(c) A set of y definite clauses, $C_1, C_2, C_3, ..., C_y$. Every consecutive C_i is connected to logical AND ($\wedge$). Each 2 literals in (b) is connected by logical OR ($\vee$).

By taking into account property (a) until (c), one can define the explicit definition of P_{2SAT} as follows

$$P_{2SAT} = \bigwedge_{i=1}^{y} C_i \tag{1}$$

where C_i is a list of clause with 2 variables each

$$C_i = \bigvee_{i=1}^{x} (m_i, n_i) \tag{2}$$

By considering the Equation (1) and (2), a simple example of P_{2SAT} can be written as

$$P_{2SAT} = \left(A \vee \neg B\right) \wedge \left(\neg M \vee D\right) \wedge \left(\neg E \vee \neg F\right) \tag{3}$$

where the clauses in Equation (3) are $C_1 = \left(A \vee \neg B\right)$, $C_2 = \left(\neg M \vee D\right)$ and $C_3 = \left(\neg E \vee \neg F\right)$. Note that, each clauses mentioned above must be satisfied with specific interpretations. For example, if the interpretation reads $\left(A, B\right) = \left(-1, 1\right)$, P_{2SAT} will evaluate False or -1. Since P_{2SAT} contains information storage mechanism and easy to classify, we implemented P_{2SAT} into ANN as a logical system.

2 SATISFIABILITY IN HOPFIELD NEURAL NETWORK

Hopfield Neural Network (HNN) [1] is a recurrent neural network that contains interconnected input and output without hidden layers. Each neuron S_i in HNN fires (+1) or inactive (-1) according to the input transferred to the network. The information transferred based on the learned connection called synaptic weight. The connection between S_i and S_j are given as S_{ij} where both neurons are connected by synaptic weight W_{ij}. Synaptic weight is a building block of content addressable memory (CAM) and can be properly trained by using Hebbian learning [12]. In HNN, the network has symmetrical property $w_{ij} = w_{ji}$ and no self-connection $w_{ii} = w_{jj} = 0$. General definition of HNN with i th activation is given as follows

$$S_i = \begin{cases} 1, if \sum_j w_{ij} S_j > \theta_i \\ -1, Otherwise \end{cases} \tag{4}$$

where S_i is a neuron state with pre-defined threshold θ. Structurally, asynchronous update is favored to ensure the final neuron state reach global solution. P_{2SAT} can be implemented in HNN by assigning 2SAT variable to each neuron and the synaptic weight can be derived by using Abdullah method [6]. From the viewpoint of 2SAT programming in HNN, the local field of HNN, $h_i(t)$ is given by:

$$h_i(t) = \sum_j w_{ij}^{(2)} S_j + w_i^{(1)} \tag{5}$$

where $W_{ij}^{(2)}$ and $W_i^{(1)}$ are the second order and first order of P_{2SAT} respectively. Therefore, Equation (5) follows the dynamical rule that reads

$$S_i(t+1) = sgn[h_i(t)] \tag{6}$$

where "sgn" represent the signum function or the squashing function of HNN. Several researchers implemented Sigmoid function to provide the non-linearity effects during neuron classification [13, 14]. Worth mentioning that, the final state retrieved from the Equation (5) and (6) is considered global solution if the neurons states satisfy the embedded P_{2SAT} with zero cost function. Magnitude of the minimum energy corresponds to the final state of neurons can be computed by using Lyapunov energy function:

$$H_{P_{2SAT}} = -\frac{1}{2}\sum_i \sum_j w_{ij}^{(2)} S_i S_j - \sum_i w_i^{(1)} S_i \tag{7}$$

According to [15], the value of $H_{P_{2SAT}}$ in Equation (7) always decrease monotonically with HNN dynamics. By substituting the satisfied interpretation into Equation (7), we can calculate the absolute minimum energy, $H_{P_{2SAT}}^{\min}$ of the HNN model. In this paper, implementation of P_{2SAT} in HNN is abbreviated as HNN-2SAT. The primary goal of HNN models is to find the final state that corresponds to $\left| H_{P_{2SAT}} - H_{P_{2SAT}}^{\min} \right| \le Tol$.

2 SATISFIABILITY BASED REVERSE ANALYSIS METHOD

The primary challenge in evaluating the data with pre-defined goal is how to extract the correct P_{2SAT} so as to evaluate effectively the behavior of the data. The configuration of the optimal P_{2SAT} should make possible tracktable inference, and should be effective enough to classify the outcome of the datasets. A popular strategy is to develop a data mining that capitalize learned P_{2SAT} incorporated with ANN. 2 Satisfiability Reverse Analysis Method (2SATRA) is a paradigm that uses HNN to learn and extract P_{2SAT} from a given dataset. Given a set of data, $S_1, S_2, S_3, S_4, ..., S_N$ where $S_i \in \{1,-1\}$ and N is the number of tested attributes, the task of 2SATRA is to map learning neuron states into final neuron state. During learning phase, the synaptic weight for each dataset will be evaluated by using Abdullah Method [6]. Table 1 summarized all the possible synaptic weight from each dataset

TABLE 1. Synaptic weight of P_{2SAT} according to [6]

Synaptic Weight	$C_1 = S_1 \vee S_2$	$C_2 = \neg S_1 \vee S_2$	$C_3 = S_1 \vee \neg S_2$	$C_4 = \neg S_1 \vee \neg S_2$
W_{S_1}	0.25	-0.25	0.25	-0.25
W_{S_2}	0.25	0.25	0.25	-0.25
$W_{S_1S_2}$	-0.25	0.25	-0.25	-0.25

For example, if the given dataset reads $\left(S_1^1, S_2^1, S_3^1, S_4^1\right) = \left(1, -1, -1, -1\right)$, 2SATRA will translate the interpretation into logical rule $P_{2SAT}^1 = \left(S_1 \vee \neg S_2\right) \wedge \left(\neg S_3 \vee \neg S_4\right)$. By refering to Table 1, the obtained synaptic weight for P_{2SAT}^1 are C_3 and C_4 respectively. In this paper, we implement several possible permutations for P_{2SAT}^1 such as

$$P_{2SAT}^{k_1} = \left(\neg S_1 \vee S_2\right) \wedge \left(\neg S_3 \vee \neg S_4\right) \tag{8}$$

$$P_{2SAT}^{k_2} = \left(\neg S_1 \vee \neg S_4\right) \wedge \left(S_2 \vee \neg S_4\right) \tag{9}$$

In this case, the $P_{2SAT}^{k_i}$ embedded to HNN has different possible attribute arrangement. Note that, only $P_{2SAT}^{k_i} = 1$ will be considered during learning phase of HNN and if the number of logical rule is within the acceptance tolerance range $n\left(P_{2SAT}^{k_i}\right) \leq Tol$, $P_{2SAT}^{k_i}$ will be selected P_{best}. The behaviour of the HNN is based on the logical P_{best} and the scheme will follows the conventional 2SATRA which leads to the induced logical rules P_i^B. To further test the accuracy of 2SATRA, we will compare P_i^B with the testing datasets.

EXPERIMENTAL SETUP

In this section, the effect of permutation in P_{2SAT}^1 in obtaining optimal induced logic will be investigated. The simulation will be divided into two parts: the first simulates effectiveness of $P_{2SAT}^{k_i}$ in achieving global minimum energy, and the other is simulating 2SATRA by using established datasets. Table 2 illustrate the list of parameters for HNN-2SAT. Clausal Noise CN is added to increase the accuracy of the 2SATRA by presenting more randomized 2SAT clause for each attribute. The main drawback for adding more CN is the learning error will increase as the learning of HNN become ineefective [7]. For the choice of activation function, Hyperbolic activation function is used to squash the final state of the neurons due to the contunuity and non-linear behaviour of the function. During retrieval phase of HNN-2SAT, the neuron is initilialized randomly in order to reduce possible biasness of the network. For both simulations, Dev C++ Version 5.11 software will be chosen due to the flexibility of the compiler. All the experiments will be conducted in Intel Core i7 2.5GHz processor, 8GB RAM and Windows 8.1. The threshold CPU time was 24 hours and outputs that exceed the time were all excluded. In terms of performance evaluation, accuracy and precision will be utilized to evaluate the effectiveness and the effect of permutation in 2SATRA. All the datasets were obtained from UC Irvine Machine Learning Repository and the detail of each dataset is stated in Table 3. Following the several literatures [9, 23] this study will take 60% as a learning data and 40% as a testing data.

TABLE 2. List of Parameters in HNN-2SAT [22]

Dataset	Parameters
Neuron Combination	100
Tolerance Value (∂)	0.001 [15]
Number of learning (θ)	100
No_ Neuron String	100
Selection_Rate	0.1
CPU time	24 Hours [9]
Activation Function	Hyperbolic Activation Function [14]
Clausal Noise CN	$1 \leq CN \leq 10$
Permutation	6

TABLE 3. List of Datasets

Datasets	German Credit Data [16]	Congressional Voting Records [17]	Hepatitis [18]
Area	Finance	Social Sciences	Medical
Number of Instances	1000	435	155
Attribute Type	Real	Binary	Real
Number of Attributes	6 out of 20	6 out of 17	6 out of 19
Missing Values	None	Yes	Yes
Learning/Testing dataset	600/400	261/174	93/62
Description	Evaluate credit risk for potential debtors	Classify republican and democrat voter in United State.	Classify the survival of the hepatitis patient with specific attributes

TABLE 4. Attribute Assignments

Datasets	German Credit Data	Congressional Voting Records	Hepatitis
$P_{2SAT}^{k_i}$	Creditability	Republican or Democratic	Die or Live
A	Duration of Credit	Handicapped infant	Sex
B	Payment Status of Previous Credit	Water project cost sharing	Steroid
C	Credit Amount	El Salvador aid	Antiviral
D	Value Savings/Stocks	Religious groups in schools	Fatigue
E	Length of current employment	Aid to Nicaraguan contras	Malaise
F	Instalment	Immigration	Anorexia

RESULT AND DISCUSSION

In this study, 2SATRA incorporated with HNN-2SAT is constructed to simulate the effect of logical permutation. Compared to previous 2SATRA models [7, 8], the arrangement of attributes in each dataset will be permuted randomly. The global minima ratio or *Zm* of 2SATRA with different embedded P_{2SAT}^{i} is reported to be consistently approaching 1. The implementation of HNN in 2SATRA is due to the absence of hidden layer which in our view, will require specific optimization that increase the complexity of the learning phase. According to [14], the value of *Zm* approaching to 1 signifies that all retrieved neuron state is global minimum energy. The neuron state is initialized randomly, and the network solely depends on the local field in Equation (5) and (6). The result also showed that the P_{2SAT}^{i} produced by 2SATRA is always evolve in global minimum solution. Hyperbolic Activation Function (HTAF) is shown to classify the final state that agrees with the "learned" logical rule $P_{2SAT}^{k_i}$. In this case, the proposed logical permutation in 2SATRA does not affect the effectiveness of HNN-2SAT in achieving global solution. The proposed method does not consider other retrieval method such as Boltzmann machine [14] because adding the mentioned optimization layer will result in logical overfitting and reduce the variation of P_{2SAT}^{i}. Table 4 until Table 6 provide a closer look at a portion of P_{2SAT}^{i} produced by HNN-2SAT and we observe that 2SATRA with permutation are more accurate in determining the final outcome of the dataset. Considering the conventional 2SATRA that utilized restricted attribute arrangement, we discovered a new perspective of 2SATRA that outperform the conventional method. The new 2SATRA show significantly higher accuracy and precision compared to the conventional induced logic $P_{2SAT}^{standard}$ for all the datasets.

In Table 7, we compared the performance of classification model with other established model in the literatures. Although the existing algorithm have similar classification potential on these day, they require more attributes and learning data to reach the same amount of accuracy. The main impetus of the study is to explore different connection among attributes that lead to higher accuracy via unsupervised learning neural network. In this case, the proposed 2SATRA is more flexible and managed to create high variation of induced P_{2SAT}^{i}. Various P_{2SAT}^{i} obtained in a subtle balance and their accuracies provide alternative insight in doing approaching different SAT logical rule such as Maximum Satisfiability [24]. For example, the lowest accuracy in Congressional Voting Record (refer Table 4) is considered valuable in creating logical rule that leads to false outcome. One of the limitation of the study is the permutation size. As the number of permutation size increase, the computational time will increase indefinitely. The input layer of 2SATRA will keep on changing until the threshold accuracy is reached which in this case is defined by the user. In this case, the new 2SATRA must commit with high value of permutation size in order to increase the variation of P_{2SAT}^{i}.

Table 8 summarized the performance of our proposed 2SATRA with the accuracy of other existing models. The performance of new 2SATRA in doing German Credit and Congressional Voting Records were significantly better compared to the existing literature. In hepatitis dataset, existing literature [21] is slightly better than our proposed method due to more number of attribute considered in their study. On the other hand, the learning phase of 2SATRA is prone to high value of learning error and accelarating algorithm is required to optimize the performance of the learning. Several studies such as [7] and [22] have proven that metaheuristics algorithm is able to reduce the learning error and reduce the computational time.

TABLE 5. Accuracy and Precision for German Credit Data

No	P_{2SAT}^{i}	Accuracy	Precision
1	$P_{2SAT}^{1}=(A\vee B)\wedge(C\vee D)\wedge(E\vee F)$	88.15%	90.73%
2	$P_{2SAT}^{2}=(A\vee C)\wedge(D\vee B)\wedge(E\vee F)$	86.25%	88.78%
3	$P_{2SAT}^{3}=(E\vee\neg D)\wedge(\neg C\vee A)\wedge(B\vee F)$	77.73%	80.00%
4	$P_{2SAT}^{4}=(\neg B\vee F)\wedge(E\vee D)\wedge(\neg C\vee A)$	80.95%	82.43%
5	$P_{2SAT}^{5}=(E\vee B)\wedge(D\vee C)\wedge(F\vee A)$	**92.42%**	**95.12%**
6	$P_{2SAT}^{6}=(E\vee D)\wedge(C\vee A)\wedge(B\vee F)$	84.83%	87.32%
7	$P_{2SAT}^{Standard}=(A\vee\neg B)\wedge(C\vee D)\wedge(E\vee F)$	76.15%	90.73%

TABLE 6. Accuracy and Precision for Congressional Voting Records

No	P_{2SAT}^{i}	Accuracy	Precision
1	$P_{2SAT}^{1}=(B\vee E)\wedge(\neg D\vee A)\wedge(C\vee F)$	50.98%	36.36%
2	$P_{2SAT}^{2}=(D\vee\neg E)\wedge(B\vee A)\wedge(C\vee\neg F)$	68.63%	90.9%
3	$P_{2SAT}^{3}=(E\vee A)\wedge(\neg F\vee B)\wedge(\neg D\vee\neg C)$	43.14%	22.73%
4	$P_{2SAT}^{4}=(E\vee\neg B)\wedge(A\vee C)\wedge(\neg F\vee D)$	11.76%	13.64%
5	$P_{2SAT}^{5}=(C\vee\neg D)\wedge(E\vee F)\wedge(B\vee A)$	**94.12%**	**100%**
6	$P_{2SAT}^{6}=(F\vee\neg A)\wedge(D\vee C)\wedge(E\vee B)$	56.86%	63.64%
7	$P_{2SAT}^{Standard}=(\neg A\vee B)\wedge(C\vee\neg D)\wedge(\neg E\vee\neg F)$	29.41%	9.1%

TABLE 7. Accuracy and Precision for Hepatitis data

No	P_{2SAT}^{i}	Accuracy	Precision
1	$P_{2SAT}^{1}=(B\vee E)\wedge(D\vee F)\wedge(A\vee C)$	52.38%	85.71%
2	$P_{2SAT}^{2}=(E\vee\neg F)\wedge(B\vee C)\wedge(\neg D\vee A)$	**80.95%**	**95.31%**
3	$P_{2SAT}^{3}=(F\vee D)\wedge(\neg C\vee B)\wedge(E\vee A)$	57.14%	84.71%
4	$P_{2SAT}^{4}=(C\vee A)\wedge(D\vee B)\wedge(F\vee E)$	38.1%	100%
5	$P_{2SAT}^{5}=(C\vee D)\wedge(F\vee B)\wedge(\neg A\vee E)$	52.38%	85.72%
6	$P_{2SAT}^{6}=(B\vee\neg E)\wedge(\neg A\vee F)\wedge(C\vee\neg D)$	51.7%	82.3%
7	$P_{2SAT}^{Standard}=(A\vee\neg B)\wedge(\neg C\vee D)\wedge(E\vee F)$	35.4%	90.2%

TABLE 8. Performance of New 2SATRA with other existing literatures

Dataset	New 2SATRA	Existing Model	Accuracy
German Credit	**92.42%**	Huang *et al*. [19]	78.1%
Congressional Voting Records	**94.12%**	Do *et al*. [20]	83.24%
Hepatitis	80.95%	Cortez [21]	**83.2%**

CONCLUSION

We have shown that accuracy of 2SATRA can be tackled in the context of neuron permutation. The different connection between attribute arrangement has been exploited to obtain P_{2SAT}^{i} with various level of accuracy. By employing different logical arrangement during learning phase of HNN, 2SATRA has achieved the best performance for all 3 datasets compared to standard 2SATRA. Moreover, 2SATRA with permutation effect outperformed most of the existing method who were using similar datasets. Further investigation will be on how to develop a statistical method such as logistic regression analysis in identifying the best pair for 2SAT in each clause.

ACKNOWLEDGMENT

This research was supported by Fundamental Research Grant Scheme (FRGS), Ministry of Education Malaysia, grant number 203/PMATHS/6711804.

REFERENCES

1. J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities", in *National Academy of Sciences of the United States of America*, Proceedings, edited by D. E. Koshland *et al.* (1982), pp. 2554-2558.
2. F. Jolai and A. Ghanbari, Expert Systems with Applications **37**, 5331-5335 (2010).
3. M. Kobayashi, Neurocomputing **240**, 110-114 (2017).
4. C. Gorman, A. Robins and A. Knott, Neural Networks **91**, 76-84 (2017).
5. Q. Xu, Z. Song, H. Bao, M. Chen and B. Bao, AEU-International Journal of Electronics and Communications **96**, 66-74 (2018)
6. W. A. T. W. Abdullah, International Journal of Intelligent Systems **7**, 513-519 (1992).
7. M. S. M. Kasihmuddin, M. A. Mansor and S. Sathasivam, "Satisfiability based reverse analysis method in diabetes detection", in *25th National Symposium on Mathematical Sciences (SKSM25)*, AIP Conference Proceedings, edited by M. Daud *et al.* (American Institute of Physics, NY, 2018), pp. 020020.
8. M. A. Mansor, S. Sathasivam and M. S. M. Kasihmuddin, "3-satisfiability logic programming approach for cardiovascular diseases diagnosis", in *25th National Symposium on Mathematical Sciences (SKSM25)*, AIP Conference Proceedings, edited by M. Daud *et al.* (American Institute of Physics, NY, 2018), pp. 020022.
9. M. A. Mansor, S. Sathasivam and M. S. M. Kasihmuddin, "Artificial immune system algorithm with neural network approach for social media performance metrics", in *25th National Symposium on Mathematical Sciences (SKSM25)*, AIP Conference Proceedings, edited by M. Daud *et al.* (American Institute of Physics, NY, 2018), pp. 020072.
10. S. A. Cook, "The complexity of theorem-proving procedures", in *Third Annual ACM Symposium on Theory of Computing-STOC '71*, Proceedings, edited by M. A. Harrison *et al.* (ACM, NY, 1971), pp. 151-158.
11. M. S. M. Kasihmuddin, M. A. Mansor and S. Sathasivam, Pertanika Journal of Science & Technology **25**, 139-152 (2017).
12. T. J. Sejnowski and G. Tesauro, "The Hebb Rule for Synaptic Plasticity: Algorithms and Implementations" in *Neural Models of Plasticity*, edited by J. Byrne and W. O. Berry (Academic Press, New York, 1989), pp. 94-103.
13. M. A. Mansor and S. Sathasivam, "Performance analysis of activation function in higher order logic programming", in *24th National Symposium on Mathematical Sciences (SKSM24)*, AIP Conference Proceedings, edited by S. Zabidin *et al.* (American Institute of Physics, NY, 2016), pp. 030007.
14. S. Sathasivam and M. Velavan, Modern Applied Science **8**, 140-146 (2014).
15. S. Sathasivam, and W. A. T. W. Abdullah, Computing **91**, 119-133 (2011).
16. H. Hofmann, Credit data, (Institut für Statistik und Ökonometrie Universität Hamburg, 1994).
17. A. Purwar and S. K. Singh, Expert Systems with Applications **42**, 5621–5631 (2015)
18. G. M. Weiss and H. Hirsh, "A quantitative study of small disjuncts", in *Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, Proceeding, (2000), pp. 665-670.
19. C. L. Huang, M. C. Chen, and C. J. Wang, Expert systems with applications **33**, 847–856 (2007).
20. T. Do Van, H. Do Duc and G. C. Nguyen, "Classify high dimensional datasets using discriminant positive negative association rules", in *5th Asian Conference on Defense Technology*, Proceedings (2018), pp. 1-7.
21. P. Cortez, "Advances in data mining: Applications and theoretical aspects" in *10th Industrial Conference on Data Mining*, Proceeding (2010), pp. 572–583.
22. M. S. M. Kasihmuddin, M. Mansor, and S. Sathasivam, Pertanika Journal of Science & Technology **25**, (2017).
23. N. Hamadneh, S. Sathasivam, S. L. Tilahun, and O. H. Choon, Journal of Applied Sciences **12**, 840-847 (2012).
24. C. Luo, S. Cai, K. Su and W. Huang, Artificial Intelligence **243**, 26-44 (2017).