© Universiti Tun Hussein Onn Malaysia Publisher's Office

EmAIT

Journal homepage: http://publisher.uthm.edu.my/ojs/index.php/emait
e-ISSN : 2773-5540

Emerging Advances in Integrated Technology

An Implementation of First and Second Order Neural Network Classification on Potential Drug Addict Repetition

Nazri Mohd Nawi[1*], Eneng Tita Tosida[2], Hamiza Hasbi[1], Norhamreeza Abdul Hamid[1]

[1]Soft Computing and Data Mining Centre (SMC), Faculti Sains Komputer dan Teknologi Maklumat (FSKTM), Universiti Tun Hussein Onn Malaysia (UTHM), 86400, Parit Raja, Batu Pahat, Johor, MALAYSIA

[2]Computer Science Training Ceneter (ComSTraC), Data Science Center (DSC), Computer Science Department, Universitas Pakuan, Jl. Pakuan Po Box 452, Bogor, Jawa Barat, 16143, INDONESIA

*Corresponding Author

DOI: https://doi.org/10.30880/emait.2021.02.01.003
Received 12 April 2021; Accepted 27 Mei 2021; Available online 30 Jun 2021

Abstract: Back propagation (BP) neural network is known for its popularity and its capability in prediction and classification. BP used gradient descent (GD) method as one of the most widely used error minimization methods used to train back propagation (BP) networks. Besides its popularity BP still faces some limitation such as very slow in learning as well as easily get stuck at local minima. Many techniques have been introduced to improve BP performance. This research implements second order method together with gradient descent in order to improve its performance. The efficiency of both methods are verified and compared by means of simulations on classifying drug addict repetition. The results show that the second order methods are more reliable and significantly improves the learning performance of BP.

Keywords: Back propagation, classification, gradient descent, neutral network, second order

1. Introduction

Artificial neural networks (ANN) is a technique that mimic the hardware's system and/or software's system that patterned after the operation of neurons in the human brain. ANN has been used in computer vision, speech recognition, machine translation, social network filtering, playing board and video games, medical diagnosis and in many other domains.

Back propagation is one of the most popular method that is used in ANN to calculate an error. The weights are adjusted to reduce the error. Back propagation learns by iteratively processing a set of training data. In order to minimize the error, back propagation adjusts the weights of the neural networks [1].

One of the popular task that used ANN in data mining is classification task where ANN particularly back propagation algorithm trains a feed-forward multi-layer perceptron neural network for a given set of input patterns. The networks examine its output response to the sample input pattern when each entry of the sample set is presented to the network. The output response and desired output are compared and the value of error is calculated. The weights of connection are adjusted based on the error [2]. The popularity of ANN had gained great attention by researchers and one of the application that used ANN is classifying the potential use of drugs by drug addict.

Drug addiction is defined as a chronic disease, repetition on drug usage. Even though these harmful effects. It is considered as a disease to the brain that can damage the brain by changing the structure and how it works. The changes in the brain can be long-lasting and can lead to many harmful behaviors, often self-destructive. Drug addiction can be considering as a serious problem to Malaysian Government and there is no systematic way to classify the potential drug addict from repeat their addiction.

*Corresponding author: nazri@uthm.edu.my

Therefore, ANN particularly back propagation algorithm is known for its famous capability in prediction and classification is used for this problem. Even though back propagation is a well-known algorithm with trained feed forward neural network. Unfortunately, it is very slow in learning and classifying the result [3]. This is because of gradient descent algorithm where the method easily can get stuck at any point. At this point it becomes necessary to make use of many techniques to improve training efficiency of back propagation algorithm to get better performance. Recent studies also had focus to introduce higher order methods such as second order instead of first order (gradient descent) used by BP. There are many second order methods that can be implemented with BP which significantly improve further the performance of BP.

The aim of this study is to analyze the performance of second order and first order neural network methods in classifying the potential drug addict repetition. Some simulations were performed on those two methods on drug addict datasets. The research used categorical dataset from *Agensi Anti Dadah Kebangsaan Kuala Terengganu*.

The remainder of this paper is organized as follows: In the next section, we present the fundamental concept of gradient descent and second order method. Next section presents the methodology for this research. The experiments and simulation results are discussed in the following section. Finally, the paper is concluded in final section.

2. Literature Review

This section discusses the basic concept of BP algorithm as well as some literature reviews by previous researchers in improving BP algorithm. In addition, some general understanding on potential drug addict will also be cover in this section.

2.1 Drug Addiction

Although Malaysia has achieved a great deal of success in the country's development trend, there are still many challenges and obstacles that can affect the status to move forward as the involvement of people with unhealthy activities such as being trapped by drug abuse. The phenomenon and the problem of drugs have become a tough challenge that can hamper career advancement and the excellence of today's identity. The involvement of a person with drug abuse activities has not only negatively impacted social problems but also able to lead to health problems such as HIV/AIDS infections. But, when a person is caught up with the symptoms of drug addiction, the probability of repeating the drug is very high [4]. The factors that causes drug addicts back to drugs are drug addiction reminders are still not strong, the heart is not strong enough to resist the craving for drugs, yet there is no desire to stop taking drugs, stigma of people who have negative perceptions, lack of work and money to sustain life and get along with friends who are still taking drugs.

In drug addict problem, it is very hard to predict someone from repeating their drug addiction. One of the best way is to classify manually without any appropriate method. Therefore, there is a need for systematic methods to help healthcare to classify on drug addict's repetition. There are several systematic methods that have been used that related to drug. Based on research by Elena Tutubalina and Sergey Nikolenko [5], recent studies on text mining applications increasingly employ nonstandard sources of information to obtain new data related to health condition, the efficiency of treatment, drug reaction, and interactions between different drugs. Traditional channels for identifying the problem which is adverse drug reactions (ADRs) are reliable but very slow and only produce a small amount of data. Text reviews, either on specialized web sites or in general-purpose social network, may lead to a data source of unprecedented size, but identifying ADRs in free-form text is a challenging natural language processing problem. So, they propose a novel model for the problem, using recurrent neural architectures and conditional random fields [5].

Other researches that are related to drug are from *Agensi Anti Dadah Kebangsaan*(AADK). The author used a quantitative cross-sectional survey and analyzed the data using multiple regression statistical tests [6]. Malhotra et al. [7] states that cross-sectional designs involve data collection methods on one type of sample from a population that is studied once only based on the attributes of the existing respondents [7]. Chua [8] others found out that multiple regression analysis is a method that can be used to identify changes in two or more factors (independent variables) that contribute to changes in a dependent variable [8]. In that research, multiple regression analysis using stepwise completion which was a variation of forward settlement procedures was used to measure the objective of the research. Diekhoff [9] states that stepwise approaches have advantages over other regression because through that method, only significant predictor variables will be included in the regression [9]. The second advantage is that stepwise regression can avoid multi-collinearity which is a strong correlation between fortune variables.

Artificial neural networks (ANN) also one of the application that researcher used to apply a multilayered, feed-forward a fully connected network of perceptions. The use of simple neural networks is done because of this type of neural network represent a universal function in the sense that if the topology of the network is allowed to vary freely it can take the shape of any broken curve by the simplicity of the theory, ease of programming as well as good results [10]. Furthermore, researchers can analyze data more efficiently by using ANN.

2.2 Artificial Neural Networks

The functionality of artificial neural networks was created based on model of the human brain. The human brain consists of 10 billion nerve cells or neurons. On average, every neuron is linked by about 10,000 synapses. The structure and function of the human brain is too complex. The input is received and processed by the brain resulting in output. Based on research of human brain function and structure researchers have suggested artificial neural network. ANNs and their techniques are becoming more popular and essential for modelling and optimization in many areas of science and engineering, and this popularity is largely due to their ability to exploit the tolerance imprecision and uncertainty in real world problem, coupled with their firmness and parallelism. Moreover, with its popularity, Artificial Neural Network (ANN) has been implemented for various classification and learning tasks [11].

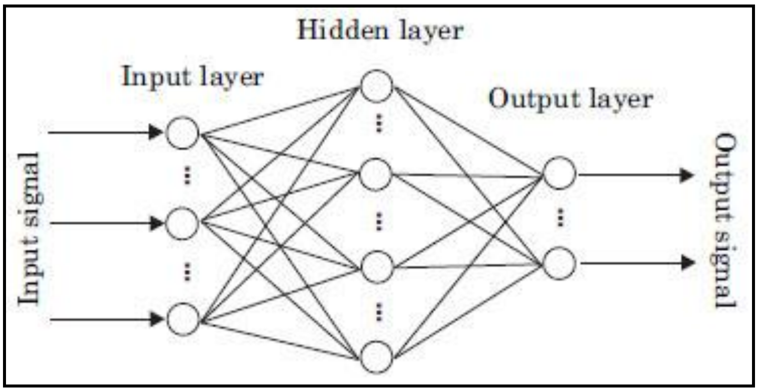


Fig. 1 - Structured chart of neural network

2.3 Related Studies

This section studies some related studies and solutions that been proposed by previous researchers and it will be the guidance for this research in deciding the appropriate improvement on BP. The following studies are some of the researches equivalent that can be compared in this study. A comparison study between the use of first and second order training algorithm for ANN had been done by [12] where the authors compared between back propagation and minimization methods for training feed-forward network. Feed-forward network training is a special case of functional minimization, where no explicit model of the data is assumed. Linearization of the training problem through use of orthogonal basis functions is not justified due to the high dimensionality of the data. A number of methods based on local gradient and Hessian matrices are discussed. This paper also suggested some modifications by implementing methods such as first and second order training methods. Experimentally proved that Conjugate gradient and Quasi Newton's methods outperformed the Gradient Descent methods using share rates data. In addition, Levenberg-Marquardt algorithm is a special interest in financial forecasting.

Since the most popular algorithm to train feed-forward neural networks is the back propagation. Unfortunately, the convergence of this algorithm is slow because of gradient descent algorithm. Research [10] had demonstrated that 'Gain' also referred to the slope of the activation function was directly influenced the performance of BP algorithm. By introducing the adaptive gain of the activation function and the gain values was changed adaptively for each node. The research proposed an improvement on BP algorithm. Furthermore, researchers also introduce the learning rate and weight values and the physical interpretation of the relationship between the gain values. The efficiency of the proposed algorithm and the conventional Gradient Descent Method was compared and verified by means of simulation on four classification problems. In learning the patterns, the simulations result show that the proposed method converged faster on Wisconsin breast cancer with an improvement ratio of nearly 2.8, 1.76 on diabetes problem, 65% better on thyroid data sets and 97% faster on IRIS classification problem. The results of the proposed algorithm significantly improve the learning speed of the conventional back propagation algorithm is shown clearly.

Recently, researches on about diagnosing breast cancer by using various machine learning techniques had gained its popularity [13]. This paper presents a study on breast cancer classification using Feed Forward Artificial Neural Networks. Back propagation algorithm was used to train this network. The performance of the network was evaluated using Wisconsin breast cancer dataset for various training algorithms. The proposed algorithm was tested on real life problem.

In this research, six training algorithms were used, and among these six methods, Levenberg Marquardt method gave the good result of 99.28%. Min-max normalization is used for pre-processing in this diagnosis. However, further work is needed to increase the accuracy of breast cancer classification diagnosis. Table 1 shows the comparison between some authors and their improvement techniques that lead to the motivation of this research.

Table 1- Comparison of research

Author	*Technique/Method/Algorithm*	*Features*
D. V. Nagori [14]	• Back propagation algorithm • Line search	The focus is functional minimization on any basis. It is proved that Conjugate gradient and Quasi Newton's methods outperformed the Gradient Descent methods by using share rates data.
Rehman, M.Z., Nawi, N. M., and Ghazali, R. [15]	• Back propagation algorithm • Activation function • Learning rate	This research proposed an algorithm for improving the performance of the back propagation algorithm by introducing the adaptive gain of the activation function. The gain values change adaptively for each node.
Basheer, I. A., and Hajmeer, M. [16]	• Back propagation algorithm • Quasi Newton	This paper presents a result of direct classification of data after replacing missing values using median method for Wisconsin Breast Cancer Data (WBCD) dataset with various back propagation training algorithms. The training algorithms are compared using accuracy.

3. Methodology

The selection of methodology is important because it affects the selection of the model developed. Therefore, the selection of methodology should meet the research requirements. This section explains the methodology that will be implement to carry out the study by adopting. CRISP-DM model which distinguishes six main phases of a Knowledge Discovery in Database (KDD) process which start from the beginning of understanding the research and ending by reporting writing for the result. Figure 2 shows that there are six phases involved in this methodology which are research understanding, data understanding, data preparation, modelling, evaluation and deployment.

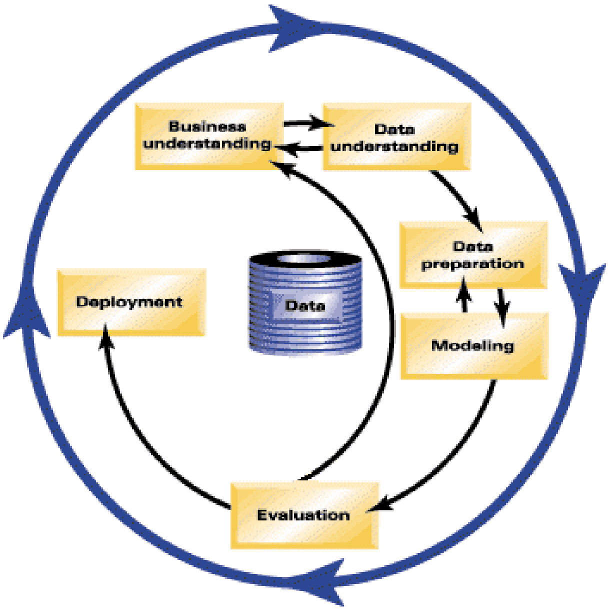


Fig. 2 - Overall process of CRISP-DM

The first phase of the CRISP-DM process is to understand the research objectives and requirements from a research perspective, then converting this knowledge to achieve the objectives [13]. At this phase the objectives of the research are being identified. In addition, all the requirements and methodology that will be used are identified including the initial selection of tools and techniques. During this phase, the background domain of the research also will be reviewed for the introduction for this research. The goal of this phase is to uncover important factors that could influence the result of the research.

The second phase of the CRISP-DM process is the data understanding phase which starts with an initial data collection and cleaning process that include activities such as identifying data quality problems, discover first insights into the data and detect interesting subsets to form hypotheses for hidden information. For this research, the data collection comes from *Agensi Anti Dadah Kebangsaan Daerah Kuala Terengganu.* Data exploration includes understanding or getting familiar with the data. Other than that, data exploration such as viewing summary statistics can occur at the end of this phase. Then identify data quality problems from the data because there could be have some problems that can discovered. Models such as classification can be applied during this phase for the simulation and generating the results. Data verification is important in this phase because it determines whether the data are required or not for all the cases. Examine the quality of the data by checking the correct data errors and also missing values in the data. Research had demonstrated that the quality of the data can significantly improve algorithm performances [17]. Therefore, it must be addressed during this phase in order to ensure that the data used during for later phase of the research are fit for purpose.

The next phase of the CRISP-DM process is to prepare the data for modeling. The data preparation phase covers all activities such as constructing the data that will be fed into the model tools. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. The tasks include table, record, and attribute selection, data cleaning, construction of new attributes, and transformation of data for modelling tools. Once the available data resources are identified, they need to be selected, cleaned, built into the form desired, and formatted. In this research, the drug addict data were selected and was cleaned to make sure every dataset have no missing value. This phase also included recording, scaling and tabulating for the requirements of the research. Then data preparation is executed where many aspects, including searching for and removing duplicate data, searching for and redefining or removing outliers, and redefining variables measurement levels.

After the data have been prepared, modeling the performance is executed in order to support the research objectives. In this phase, various modelling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are some techniques for the same data mining problem type. Some techniques require specific data formats. This research uses some selected algorithms from first order and second order of neural network. All parameters including the learning rate, the momentum rate, gain and the activation function are fixed. In this research, the simulation runs using MATLAB because it is known for its high-performance language for technical computing. Furthermore, it integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

Finally, the evaluation phase where at this phase the performance of each models that been selected are evaluated. It is important to thoroughly evaluate all models and review the steps executed to construct the model in order that it can be certain and properly achieves the research objectives. This phase analyses all simulation results for each training and testing dataset. Besides, this phase demonstrates steps taken to construct the research models.

4. Results and Discussions

ANN particularly the back propagation (BP) algorithm has been widely used in many applications because of its stability and capability in solving non-linear problems. Despite providing successful solutions, BP still suffering from some limitations such as slow convergence and easily getting stuck in local minima. Researchers had introduced many improvements and enhancements by introducing some modifications on the architecture as well as adjusting some parameters that contribute to the performance of BP. Latest researches also demonstrate that the implementation of second order with BP can significantly improve further the speed and the convergence accuracy of BP.

The performances of the second order method implementation are evaluated using real data collected from *Agensi Anti Dadah Kuala Terengganu*. The raw data have gone through some pre-processing phase where the data have been normalized using Microsoft Office Excel and also RapidMiner Studio programs. MATLAB version R2010b is used to run all simulations on Intel(R) Core i5 @2.50GHz, RAM 6.00 GB, Toshiba Satellite L840. Furthermore, this research only focuses on the performance of three algorithms as follows:

1) Gradient Descent (GD) method
2) Conjugate Gradient Fletcher Reeves (CGFR) method
3) Conjugate Gradient Polak Ribiere (CGPR) method

The dataset contains of 587 instances and after pre-processing the datasets is then divided randomly into two parts for training and testing. The data split into two where 70% of data which consists of 411 instances will be used for training and the remaining 176 instances (30%) are used for testing. There are 9 attributes that been identified for this research which consists of 8 input node and only one output node. This research sets different types of hidden nodes architecture which consists of 5, 7 and 10 hidden nodes. Number of iteration is assigned to 1000 epoch and 100 trials are run for the dataset. Mean Square Error (MSE) which is 0.01 is used to measure error.

Table 2 - Fixed parameter

Parameter	***Value***
Momentum α	0.4
Learning rate ŋ	0.5
Adaptive gain	No

Table 2 shows three parameters that required for the network model. Those parameters are learning rate, momentum, and adaptive gain. The sigmoid activation function is used for activation function because its value ranges between 1 and 0.

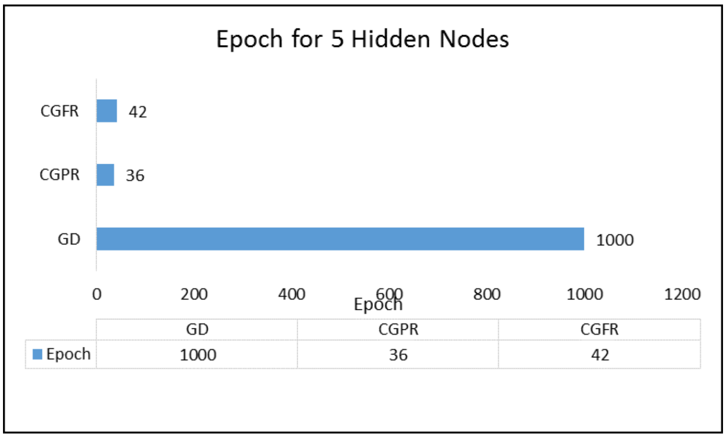
This dataset consists of 8 input attributes and 1 output attributes where the input attributes are: -

a) Educational background,
b) Occupation,
c) Types of drugs,
d) Reasons of addiction,
e) Way to addict,
f) Age of addiction,
g) Period of addiction,
h) Marital status.

The simulation results are recorded and arranged based on different number of hidden nodes architecture. All output attributes for this dataset is to classify the repetition of drug addict. All results are recorded in term of (a) the number of iteration (epochs), (b) CPU time as well as (c) accuracy and stored in the result file that to be measured the performances.

4.1 5 Hidden Nodes Architecture

An epoch is a measurement as the number of cycle of the training vectors needed to update the weights. For batch training all of the training data pass through the learning algorithm simultaneously in one epoch before weights are updated. In addition, the advantage of getting the accurate value is by adding number of iterations (epoch) where for more epochs that are used the better the results are shown in Figure 3.



	GD	CGPR	CGFR
■ Epoch	1000	36	42

Fig. 3 - Epochs of GD, CGPR and CGFR using 5 hidden nodes

Figure 3 shows the number of iterations (epoch) for GD, CGPR and CGFR using 5 hidden nodes. The figure also shows that GD makes full epoch of 1000 epochs as compared to CGPR and CGFR where each making 36 epochs and 42 epochs. If the number of iterations (epoch) is small in the training, the result would be poor and there would lead to under fitting. Therefore, in order to avoid under fitting, the network must have enough hidden units to represent the required mappings and train networks for long periods of time so that functions of square error costs enough to minimize.

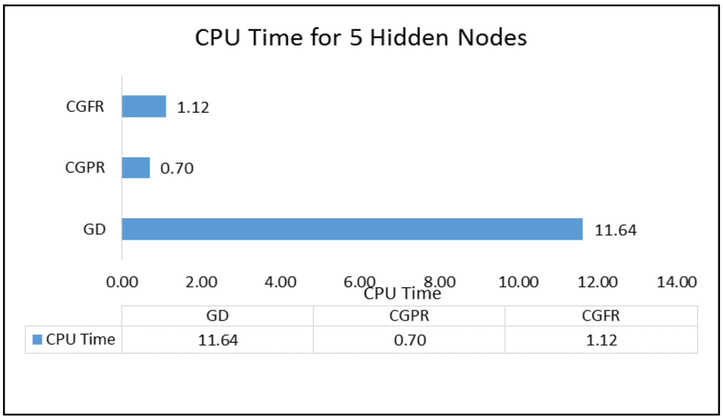


Fig. 4 - CPU Time of GD, CGPR and CGFR using 5 hidden nodes

Figure 4 shows the average of CPU time for GD, CGPR and CGFR using 5 hidden nodes. It shows that CGPR and CGFR are focusing to reach global minima within fewer epochs faster than GD. Each recorded an average time of 11.64 as compared to GD 0.70 and CGPR 1:12 as compared to CGFR. The figure shows the CGPR are faster rather than CGFR and GD.

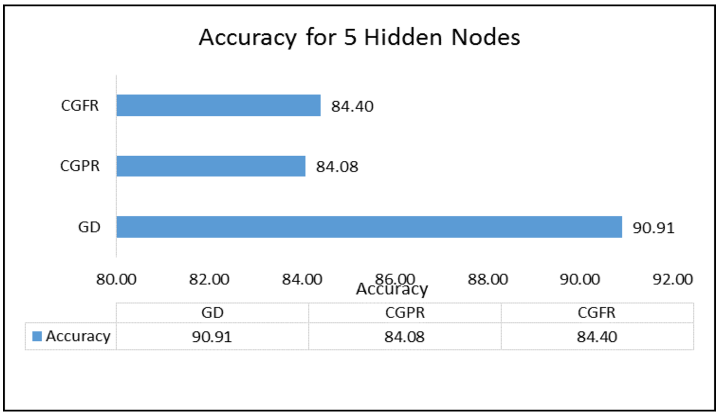


Fig. 5 - Accuracy of GD, CGPR and CGFR using 5 hidden nodes

Figure 5 shows the accuracy percentage between GD, CGPR and CGFR. This shows that GD has the highest average accuracy of CGPR and CGFR of 90.91. Then it follows by CGFR with average accuracy of 84.40 and CGPR of 84.08. The average accuracy for CGPR and CGFR is not much difference with only 0.32. Overall difference in the average accuracy of first order and second order is 6:51.

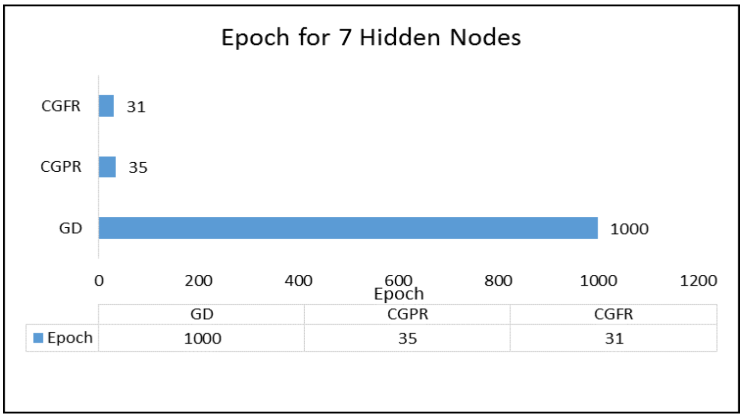
Table 3 - The performance summary of algorithms on Epochs, CPU time and accuracy for drug addict classification dataset (5 Hidden Nodes)

Algorithm	*GD*	*CGPR*	*CGFR*
Epochs	1000	**36**	42
CPU Time	11.64	0.70	1.12
Accuracy %	**90.91**	84.08	84.40

As summarized in the Table 3, GD algorithm performs the best accuracy with 90.91% as compared to CGPR and CGFR. However, GD algorithm took as much as 1000 epochs or 25 times more epochs in achieving the accuracy as compared to CGFR and CGPR. Even though CGFR with the second highest average accuracy of 84.40 and CGPR with the average accuracy of 84.08. Both CGPR and CGFR took less epoch as well as CPU time to converge as compared to GD which took 11.64 per second to converge. It demonstrates that the implementation of second order methods such as CGPR and CGFR reduce significantly CPU time with less epoch which help the simulation process to be fast and save time.

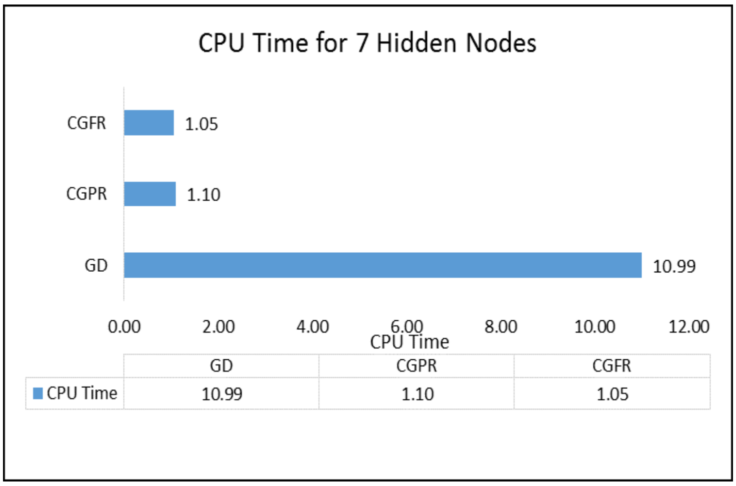
4.2 7 Hidden Nodes Architecture

Figure 6 shows the number of iterations (epoch) of GD, CGPR and CGFR while using 7 hidden nodes. In order to achieve the minimum error, it shows that GD performs with full iteration. The number of iteration (epoch) for CGPR is 35 and CGFR is 31. There not much difference in term of epoch between CGPR and CGFR and they produce a better result. However, as expected that first order method need more epoch to converge as compared to second order methods.



	GD	CGPR	CGFR
■ Epoch	1000	35	31

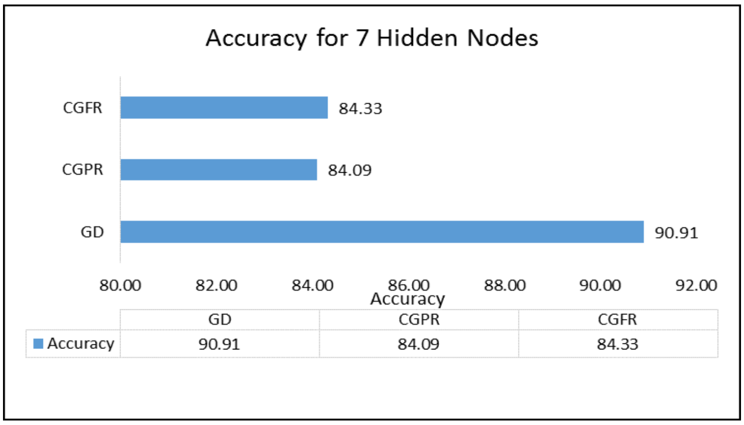
Fig.6-Epochs of GD, CGPR and CGFR using 7 hidden nodes



	GD	CGPR	CGFR
■ CPU Time	10.99	1.10	1.05

Fig. 7 - CPU Time of GD, CGPR and CGFR using 7 hidden nodes

Figure 7 shows the average of CPU Time for GD, CGPR and CGFR that used 7 hidden nodes. When 7 hidden nodes are applied to the network model, GD method records the average of CPU time of 10.99, CGPR took only 1.10 seconds and CGFR only 1.05. In addition, it shows that CPU time records for CGPR and CGFR are not much different and yet they perform better than GD because GD takes longer time to get the minimum error and less average of accuracy.



	GD	CGPR	CGFR
■ Accuracy	90.91	84.09	84.33

Fig.8 - Accuracy of GD, CGPR and CGFR using 7 hidden nodes

Figure 8 shows the average accuracy for GD, CGPR and CGFR using 7 hidden nodes. GD recorded the highest average of accuracy as compared to CGPR and CGFR which is 90.91. Each CGPR and CGFR recorded an average accuracy of 84.09 and 84.33. The average of accuracy results for 5 hidden nodes and 7 hidden nodes are almost the same when round off to two decimal places.

Table 4 - The performance of algorithms on Epochs, CPU Time and accuracy for drug addict classification dataset (7 Hidden Nodes)

Algorithm	*GD*	*CGPR*	*CGFR*
Epoch	1000	35	31
CPU Time	10.99	1.10	1.05
Accuracy %	90.91	84.09	84.33

As summarized in the Table 4, GD algorithm performs the same accuracy as in Table 3. However, there is a variance between CGPR with CGPR at Table 3 which is 0.01 more accurate where the accuracy for CGPR is 84.09. Similarly, there is slightly different result for CGFR of 0.07 percent accurate when the hidden nodes are added to 7 in Table 4. GD can reduce the CPU time of 0.65 from 10.99. However, CGPR takes as much as 1.10 to converge. On the other hand, CGFR saves CPU time as much as 0.07, from 1.12 to 1.05 and decrease the number of iteration (epoch) occurred, as shown in Table 4.

4.3 10 Hidden Nodes Architecture

Figure 9 shows the number of iteration (epoch) for GD, CGPR and CGFR used 10 hidden nodes. Again, GD make use of 1000 epochs to converge whereas CGPR and CGFR just need as much as 34 and 41 epochs to converge. Even though, the results show that there is an increase of epoch for CGFR and reduction of epoch for CGPR and yet they perform better than first order method.

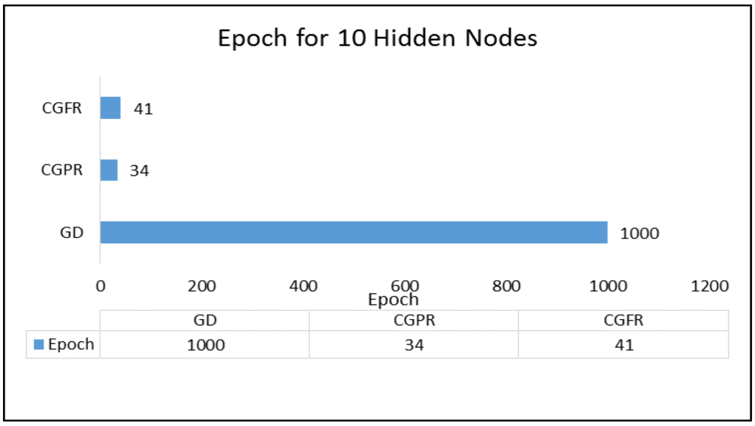


Fig. 9 - Epochs of GD, CGPR and CGFR using 10 hidden nodes

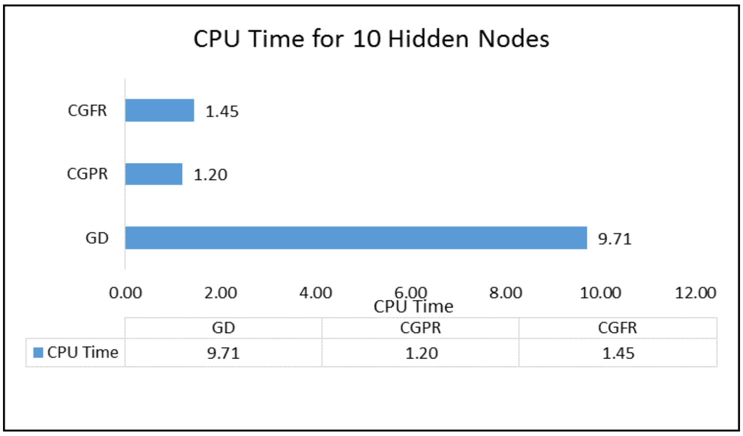


Fig. 10 - CPU Time of GD, CGPR and CGFR using 10 hidden nodes

Figure 10 shows the average of CPU Time for GD, CGPR and CGFR using 10 hidden nodes. The figure shows a slight difference for the CPU Time by GD where GD records faster when using 10 hidden nodes as compared to 5 and 7 hidden nodes used previously. CPU Time recorded by GD is 9.71. As usual, CGPR and CGFR recorded almost the same time with the use of 5 and 7 hidden nodes of 1.20 and 1.45.

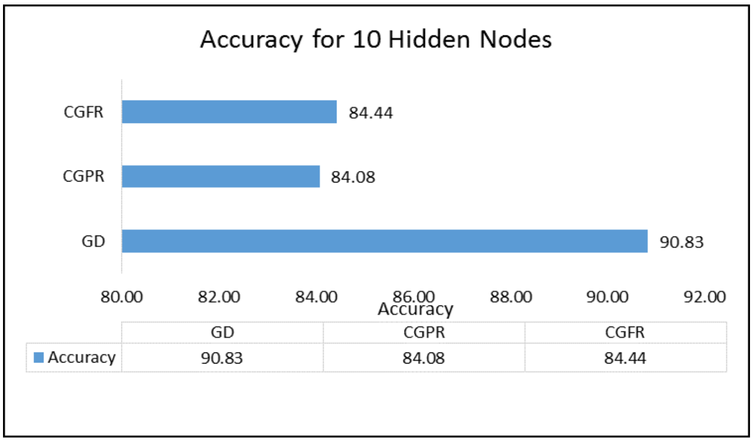


Fig. 11 - Accuracy of GD, CGPR and CGFR using 10 hidden nodes

Figure 11 shows the average accuracy for GD, CGPR and CGFR that used 10 hidden nodes. When the hidden nodes are added, the average accuracy for GD changed to 90.83. CGPR also changed to 84.08 and CGFR to 84.44. CGFR shows an increase of 0.11 percent accuracy when the hidden nodes are added to 10. GD still maintains the average accuracy results with 90.

Table 5 - The performance of algorithms on Epochs, CPU time and accuracy for drug addict classification dataset (10 Hidden Nodes)

Algorithm	*GD*	*CGPR*	*CGFR*
Epoch	1000	34	41
CPU Time	9.71	1.20	1.45
Accuracy %	90.83	84.08	84.44

As summarized in the Table 5, GD algorithm achieves the highest average accuracy as compared to CGPR and CGFR which is 90.83. CGPR performs with 84.08 and the average accuracy for CGFR is 84.44. CPU time recorded by GD for 10 hidden nodes is 9.71 faster than 5 hidden nodes as well as 7 hidden nodes. CGPR takes less CPU time for 1.20 seconds with 34 epochs but CGFR takes more CPU time for 1.45 seconds with 41 epochs. Therefore, the simulations show that in order to get a better accuracy the number of hidden nodes must be added.

4.4 The Summary Result on Testing Datasets

The performances of algorithms on drug addict classification dataset are further evaluate by using testing datasets. 70% of the dataset are reserved for training session. Dataset for testing is selected randomly so bias issues would not arise in these problems.

Table 6 summarizes that GD algorithm performs very well and for the classification problems. However, GD took maximum of epochs to reach 90.91 percent accuracy. Therefore, the results show that CGPR achieves the average accuracy of 84.06 and CGFR of 83.79. Both algorithms record the same CPU time which is 1.05 with 34 epochs and 32 epochs.

Table 6 - Summary Results for Drug Addict using 5 hidden nodes

Algorithm	*GD*	*CGPR*	*CGFR*
Epoch	1000	34	32
CPU Time	13.37	1.05	1.05
Accuracy %	90.91	84.06	83.79

Table 6 shows the average of epochs, CPU time and accuracy of drug addict dataset on testing data. Number of iteration (epoch) made by GD algorithm is still reach maximum of 1000 epochs. However, CGPR made 34 epochs and followed by CGFR which is 32 epochs in average. CGPR and CGFR took less epochs because both of them are second order neural network where they improve the performance of first order neural network, back propagation algorithm. GD algorithm takes 13.37 seconds of CPU time and CGPR and CGFR records the same time which is 1.05 each as shown in Table 6. The average of accuracy for 5 hidden nodes as shown in Table 6 shows that GD algorithm has the nearest error which is 90.91. Then, CGPR have 84.06 and follow by CGFR with 83.79 in average accuracy on drug addict datasets.

Table 7 summarized the performance of GD algorithm, CGPR and CGFR when using 7 hidden nodes for simulation process. By the longest time as compared to CGPR and CGFR namely 11.22 seconds, GD algorithm has recorded the highest average accuracy when hidden nodes are added to 7. CGPR records the fastest CPU time which is 1.14 and followed by CGFR which is 1.25 with an average accuracy of 84.14 and 83.94 respectively.

Table 7 - Summary Result for Drug Addict using 7 hidden nodes

Algorithm	*GD*	*CGPR*	*CGFR*
Epoch	1000	34	32
CPU Time	11.22	1.14	1.25
Accuracy %	90.91	84.14	83.94

Table 7 shows the average of epochs, CPU time and accuracy of drug addict dataset when using 7 hidden nodes. It notices that the number of iteration (epoch) achieve the same result when using 5 hidden nodes. However, some improvements happen at the CPU time where CPU time becomes faster when 7 hidden nodes is used. GD algorithm achieves CPU time of 11.22, then follow by CGPR of 1.14 and last but not least CGFR with 1.25. The average accuracy for GD algorithm is still same which is 90.91 and for CGPR and CGFR was increased to 84.14 and 83.94.

Table 8 shows the summary result of performance of GD, CGPR and CGFR when 10 hidden nodes is used. The GD achieves the same average accuracy of 90.91. The average accuracy for CGPR is 84.20 where it increases up to 0.06. On the other hand, CGFR also increase its accuracy with 0.19 to 84.13. It can be notice that in Table 8, the number of iteration (epoch) for CGPR and CGFR differ only by 1 epoch which is 33 and 32.

Table 8 - Summary result for drug addict using 10 hidden nodes

Algorithm	*GD*	*CGPR*	*CGFR*
Epoch	1000	33	32
CPU Time	10.41	1.17	1.22
Accuracy	90.91	84.20	84.13

Table 8 shows the average of epochs, CPU time and accuracy of drug addict dataset using 10 hidden nodes. In the number of iteration (epoch), CGFR performs 32 epochs while CGPR performs 33 epochs. In facts, GD algorithm performs 1000 epochs because back propagation algorithm need more epoch and time to find the accurate error. The hidden nodes also influence the CPU time so GD algorithm which takes 10.41 seconds to converge. While CGPR takes only 1.17 and CGFR takes 1.22.

5. Conclusion

This research presents the performance comparison of classification between first order neural network known as Gradient Descent and second order neural network such as Conjugate Gradient on drug addict dataset with different hidden nodes architecture. The simulation results demonstrate that Conjugate Gradient clearly is the best algorithm that has the ability to classify dataset with high accuracy by using appropriate inputs and outputs. Moreover, this research is very useful for future researcher which will helps them in doing classification on classification problems with a very good accuracy.

Acknowledgement

The Authors would like to thank Universiti Tun Hussein Onn Malaysia (UTHM) and Faculty of Computer Science and Information Technology (FSKTM) for supporting this Research.

References

[1] Deng, W. J., Chen, W. C., and Pei, W.: Back-propagation neural network based importance-performance for determining critical service attributes, J. of Expert Systems and Applications, vol. (2), pp. 1--26. (2008)

[2] Basheer, I. A., and Hajmeer, M.: Artificial Neural Networks: fundamentals, computing, design and application, J. of Microbiological Methods, vol. 43 (1), pp. 03--31. (2000)

[3] Nazri Mohd Nawi, R.S. Ransing, Mohd Najib Mohd Salleh, Rozaida Ghazali, Norhamreeza Abdul Hamid, "An Improved Back Propagation Neural Network Algorithm on Classification Problems," Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia, (2010)

[4] Fauziah Ibrahim*, "Pengguna Dadah Wanita di Malaysia: Pengalaman Penagihan dan Hubungan Kekeluargaan," Pusat Pengajian Psikologi dan Pembangunan Manusia, Fakulti Sains Sosial dan Kemanusiaan, Universiti Kebangsaan Malaysia, Bangi, Selangor, (2014)

[5] E. T. a. S. Nikolenko, "Combination of Deep Recurrent Neural Networks and Conditional Random Fields for Extracting Adverse Drug Reactions from User Reviews," Hindawi Journal of Healthcare Engineering, Russia, (2017)

[6] Fauziah Ibrahim, "FAKTOR MENYUMBANG KEPADA PENAGIHAN RELAPS DALAM KALANGAN PENAGIH DADAH PUSPEN DI SEMENANJUNG MALAYSIA," AADK, Malaysia, (2015).

[7] N. H. J. S. M. &. C. M. Malhotra, "Marketing Research: Applied Orientation (1st Edition)," Prentice Hall, Sydney, (1996)

[8] C. Y. Piaw, "Kaedah dan Statistik Penyelidikan: Kaedah Pendidikan.," McGraw Hill, Kuala Lumpur, (2006).

[9] G. Diekhoff, "Statistics for the Social and Behavioral Sciences.," Wm. C. Brown Publisher, Dubuque, (1992).

[10] Nazri Mohd Nawi, "The Effect of Pre-Processing Techniques and Optimal Parameters selection on Back Propagation Neural Networks," Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia, (2017)

[11] V. N. a. B. Trivedi, Fundamentals of ANN, Back propagation algorithm and its parameters, International journal of science, technology and management, (2014)

[12] T. A. J. C. A. Syed Muhammad Aqil Burney, "A Comparison of First and Second Order Training Algorithms for Artificial Neural Networks," *World Academy of Science, Engineering and Technology,* vol. 1, p. 1, (2007).

[13] A. F.Paulin, "Classification of Breast cancer by comparing Back propagation training algorithms," *International Journal on Computer Science and Engineering,* vol. 3, p. 1, (2011)

[14] D. V. Nagori, "Fine tuning the parameters of back propagation algorithm for optimum learning performance," 2nd International Conference on Contemporary Computing and Informatics, (2016)

[15] Rehman, M.Z., Nawi, N. M., and Ghazali, R.: Studying the effect of adaptive momentum in improving the accuracy of gradient descent back propagation algorithm on classification problems, J. International Journal of Modern Physics (IJMPCS), vol.1 (1). (2012)

[16] Basheer, I. A., and Hajmeer, M.: Artificial Neural Networks: fundamentals, computing, design and application, J. of Microbiological Methods, vol. 43 (1), pp. 03--31. (2000)

[17] J. H. Rüdiger Wirth, "CRISP-DM: Towards a Standard Process Model for Data Mining," European Commission, Germany