Auto-associative Networks and Genetic Algorithms for Classification

Rui Chu
Research School of Computer Science, Australian National University

Abstract. In this paper, genetic algorithms for feature selection combined with an artificial neutral network is conducted to solve a real-world binary classification problem with Statlog(German Credit Data) Data Set [1]. In addition, an auto-associative network is used to create a compression representation of the input patterns and then to solve the issue. These two methods and a combined feature selection based on decision trees and neural networks method on the same data set published in the research paper are compared. It can be observed that the test accuracy of the auto-encoder approach is higher than the accuracy of other two combined feature selection and neural networks methods. Moreover, the test accuracy of genetic algorithms is slightly higher than that of decision trees method.

Keywords: Auto-associative Neural Networks, Genetic Algorithms, Feature Selection, Classification

1 Introduction

The data set used in the paper is Statlog(German Credit Data) Data Set [1]. The data set contains information on 1000 loan applicants and each applicant is described by a series of 24 different numerical attributes. The problem is to classify applicants as good or bad credit risks. The reason why this data set is chosen is that the issue is closely related to our lives. In addition, the classification is significant because it is worse to classify a customer as good when they are bad or classify a customer as bad when they are good. Using loan classification technique, the business flexibility of companies is realized, and its business process is informatized. In this way, the loan management level and economic benefits of companies can be improved, and the loan process can be standardized. Furthermore, it reduces costs and improves business efficiency.

Feature selection is a task to select a minimum number of required features to represent a data that can be able to distinguish from each class. Since necessary and relevant features are selected and irrelevant ones are eliminated, classification accuracy can be improved and learning time can be shortened [2].

To model the problem, genetic algorithms for feature selection combined with an artificial neutral network is conducted to solve the credit classification problem. In the second part, an auto-encoder is used on the German Credit data set to create a compressed representation and then combine it with the original output column to form a new data set. A backpropagation feedforward network with batch training approach is built to run the new data set for the binary classification. K-fold cross validation method is used to evaluate the model.

2 Method

Since normalizing input data prior to training can improve the performance of neural network and speed up calculations, the input data is normalized [3]. To improve data quality, duplication information and missing values in the data set are checked during the preprocessing phase.

2.1 Genetic Algorithms

Genetic algorithms are inspired by the process of Darwinian natural selection and biological evolution. It is widely used for finding a near optimal solution to optimization problems. Genetic algorithms use binary strings to represent individuals in the problem domain. In the feature subset selection problem, each candidate solution (individual) would represent a feature subset [4]. For example, let N be the total number of features, there exist 2^N possible feature subsets. Each candidate solution is represented by a N-bit binary string.

The quality of each individual in the population can be evaluated using a fitness function. Those individuals that represent better solutions are given opportunities to “reproduce”, producing chromosomes for the next generation.

Chromosomes also undergo mutation in order to introduce new genetic material into the population and then ensure genetic diversity from one population to the next, which is similar to biological mutation [5]. Mutation is randomly changing the values of genes in a chromosome. For instance, a string 10010 may randomly changes to 11010 [4]. Crossover, on the other hand, operates on two parent strings to produce two offspring. Crossover is the process of producing one or more new individuals through the combination of genetic material randomly selected from two or more parents.

Since a genetic algorithm is a stochastic search for an optimal solution to a given problem, it is used to obtain the feature set that minimizes redundant and irrelevant attributes and maximizes the relevant ones. As a result, classification accuracy can be improved and learning process can be shortened.

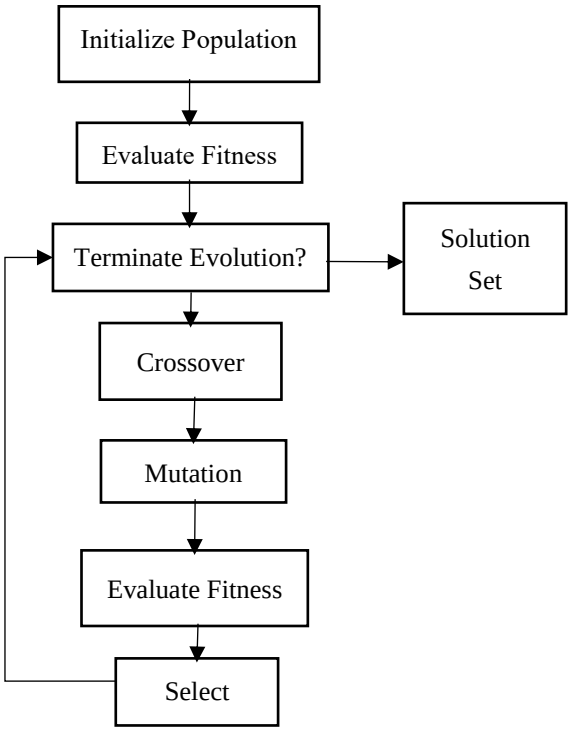

Fig. 1. Flowchart of a basic genetic algorithm for feature selection.

2.2 Auto-associative Topology

The dataset is encoded using an auto-encoder. The auto-associative network contains same number of input units and output units, which is 24, and 10 hidden units. The desired outputs in the network are the same as the inputs, using back-propagation training algorithm [6]. To build the network, weights are initialized to small random values and calculate hidden activations and output activations [7]. The activation function used is Sigmoid Function. Weights are adapted recursively from the output layer backwards towards the input layer and repeat calculations. The final hidden layer activations are the compression representation for the input data. The compressed representation is combined with the original data targets to form a new data set with 11 different attributes.

The auto-associative network is to map the input to the output. As fewer number of hidden neurons than input neurons are used, it can reduce dimensionality. As a result, the following calculation becomes faster. Furthermore, pruning redundant or useless hidden units from backpropagation training neural networks is beneficial, as it can improve generalization performance [8].

2.3 Back-propagation Training

Error back-propagation training algorithm is used for the neutral network model. Since the issue is binary classification, the Sigmoid Function is used as activation function. Sigmoid Function is defined as:

$$f(x) = \frac{1}{1+e^{-x}} \quad (1)$$

The cross-entropy loss is used as loss function.

2.4 Stochastic Gradient Descent with Momentum

The optimization method used is Stochastic Gradient Descent(SGD) with momentum. SGD has trouble navigating ravines and may cause local minimum [9]. Momentum is an approach that can help accelerate SGD in the relevant direction and increase the speed of convergence rate and learning rate [10].

Since too many number of epochs to train will lead to overfitting, early stopping is used to find the optimal number.

2.5 Batch Training

The backpropagation feedforward network using batch training method is built to run the new data set with 10 input units. Mini-batch training splits the training data set into small batches and calculates model error. However, it updates weights and biases after each batch size of examples have been evaluated.

As to the batch size, the large mini-batch size can reduce the communication cost but may slow down in convergence [11]. Moreover, generalization performance degrades as the min-batch size increases [12]. Nevertheless, since model is updated after each propagation, the network will be trained slower with small mini-batches. In addition, small batch size will lead to less accurate estimation of the gradient [13]. Therefore, the batch size is initialized to 100 in the case and it has a good performance in practice.

2.6 K-fold Cross Validation

To evaluate the network, 5-fold cross validation method is used. The data set is randomly divided into 5 subsets. Each time, one of the five subsets is used as the test set the other four subsets are formed as the training set. The model is learned from the training set and applied to the test set to calculate error. Repeat the procedure on all data set and compute the average error to compute the model.

K-fold cross validation method is selected because it can address overfitting problem. In addition, the variance of evaluation is relatively small and is not very expensive to compute.

3 Results and Discussion

5-fold cross validation method is used to evaluate the model. Cross validation method is selected because it can address overfitting problem. As the model has good prediction in new data sets, it is not overfitting. Holdout method, which the data set is randomly separated into two sets, called the training set and the test set is also tried in the experiment. However, even if holdout method takes less time to compute, its evaluation has a high variance. The test accuracy is unstable. In addition, leave-one-out method is very expensive to compute. Therefore, k-fold cross validation is used for evaluation. As to the reason why k is selected as 5, it will waste too much data if k is too small. Furthermore, if k is too large, it is expensive to compute.

In the experiment, the test accuracy is between 60 and 70 percent without genetic algorithms, auto-encoder and batch training. However, the accuracy increases to more than 70 percent after using mini-batch training. With the use of mini-batching training, the results of the auto-associative network are evaluated. Table 1 shows average and standard deviation of testing accuracy before and after using the genetic algorithm and auto-associative network. It is obvious that the testing accuracy increases to 80.6% using auto-encoder. Moreover, the stability of the accuracy is improved. However, the test accuracy of genetic algorithm only increases to 77.20% and becomes more unstable.

Table 1. *Results with and without genetic algorithm and auto-encoder*

Method	Ave. of test accuracy (%)	Std. Dev. of test accuracy
genetic algorithm	77.20	2.48
auto-encoder	80.60	1.59
no genetic algorithm no auto-encoder	76.50	2.26

As pruning redundant and useless hidden can lower generalization error and improve generalization performance [8], auto-encoder is beneficial to obtain high accuracy on test set. In addition, the auto-associative neutral network extracts classification rules and denoises input features for robust data. Since relevant and necessary features are selected and irrelevant and redundant ones are eliminated, genetic algorithms can improve classification accuracy and simplify learning process [2]. As to the reason why the performance of genetic algorithms is not better than that of auto-encoder, genetic algorithms cannot guarantee optimal solution and the quality of any solution. In addition, some useful and relevant features may be eliminated because the original data set used only contains 24 different features. Genetic algorithms may perform better on large data set that contains much more features.

The average of accuracy on the test set in the research paper using a combined feature selection based on decision trees and neural networks method on the same data set fluctuates between 73% and 76% and standard deviation is between 0.35 and 1.65 [14]. Compared to it, the test accuracy of the auto-associative network is higher but more unstable than the accuracy in the research paper. As the improved robustness to noise by feature selection, the test accuracy is stable in the research paper. However, the number of selected attributes used as input is 7, which may lose useful information, so the accuracy on the test set is lower than that of auto-encoder method. Compared two combined feature selection and neural networks methods, the test accuracy of genetic algorithms is slightly higher but more unstable than that of decision trees method. Because genetic algorithms cannot guarantee the quality of any solution, the selected solution may not be optimal. As a consequence, even the test accuracy of genetic algorithms is slightly higher than that of decision trees, its stability is decreased.

4 Conclusion and Future Work

In this paper two approaches are conducted to the issue of classifying loan applicants as good or bad credit risks. One uses an auto-associative network to create a compressed representation of the input patterns. A mini-batch training neural network is modeled to learn the compression representation and original targets and then to predict outputs. Another is a combined genetic algorithm for feature selection and neural networks method. The auto-encoder technique can obtain a slightly higher accuracy than genetic algorithm. Compared the results to a combined feature selection based on decision trees and neural networks method on the same data set, the test accuracy of genetic algorithms is slightly higher but more unstable than accuracy of decision trees method.

Future work includes combining auto-associative networks with two feature selection methods to see whether the test accuracy can be higher and more stable. In addition, since shared weight and bidirectional topology can reduce the number of free parameters and speed up training of input to hidden weights, they are likely to obtain better generalization performance [7]. Therefore, different constraints will be imposed on the auto-associative neutral network weights to obtain better performance in future proposed work. Moreover, a genetic algorithm can be used to find the optimal hyper parameters for neutral networks.

References

1. Dua, D., & Karra Taniskidou, E. (2017). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.
2. Chaikla, N., & Qi, Y. (1999). Genetic algorithms in feature selection. In *Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on* (Vol. 5, pp. 538-540). IEEE.

3. Sola, J., & Sevilla, J. (1997). Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions on nuclear science*, *44*(3), 1464-1468.
4. Yang, J., & Honavar, V. (1998). Feature subset selection using a genetic algorithm. In *Feature extraction, construction and selection* (pp. 117-136). Springer, Boston, MA.
5. Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT press.
6. Gedeon, T. D., & Harris, D. (1992, June). Progressive image compression. In *Neural Networks, 1992. IJCNN., International Joint Conference on* (Vol. 4, pp. 403-407). IEEE.
7. Gedeon, T. D. (1998, October). Stochastic bidirectional training. In S*ystems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on* (Vol. 2, pp. 1968-1971). IEEE.
8. Gedeon, T. D., Catalan, J. A., & Jin, J. Image Compression using Shared Weights and Bidirectional Networks. In *Proceedings 2nd International ICSC Symposium on Soft Computing (SOCO'97)* (pp. 374-381).
9. Sutton, R. S. (1986). Two problems with backpropagation and other steepest-descent learning procedures for networks. In *Proceedings of Eightth Annual Conference of the Cognitive Science Society, 1986*.
10. Qian, N. (1999). On the momentum term in gradient descent learning algorithms. *Neural networks*, *12*(1), 145-151.
11. Li, M., Zhang, T., Chen, Y., & Smola, A. J. (2014, August). Efficient mini-batch training for stochastic optimization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 661-670). ACM.
12. Hoffer, E., Hubara, I., & Soudry, D. (2017). Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In *Advances in Neural Information Processing Systems* (pp. 1729-1739).
13. Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.
14. O'Dea, P., Griffith, J., & O'Riordan, C. (2001). Combining feature selection and neural networks for solving classification problems. *Information Technology Department, National University of Ireland*.