Reduction of High Dimension Dataset

Yutao Ge
College of Engineering and Computer Science
u6283016@anu.edu.au

Abstract. This report describes the process of implementing and improving a simple 3-layer neural network based on real radar data. This report has tried to do feature selection using genetic algorithm to improve the predict power of classifier or regressor, and the result shows that the accuracy can be improved through feature selection.

Keywords: Feature selection, neural network, genetic algorithm.

1 Introduction

1.1 Database selection

Ionosphere is a dataset collected by a system in Goose Bay, Labrador, which contains 34 dimensions (features) and 2 classes (Sigillito, 1989). This dataset was used in minimal distance neural methods (Dich, Grudzinski, & Diercksen, 1998) and feature selection for unsupervised learning (Dy & Brodley, 2004).

Ionosphere dataset has various advantages. Firstly, it contains hundreds of instances, thus it can provide relative enough training data in network training. Secondly, the dataset has completed values and 34 different dimensions (features) which are relative enough to figure out the power of feature selection. Thirdly, the dataset is collected for finding the evidence that some types of structure exist in ionosphere, and feature selection can improve the accuracy of analysis. Finally, as this dataset has been widely used, the comparison with other resources can be simply achieved.

1.2 Problem and Modeling

One of the main focuses of this report is to find out whether genetic algorithm can help improve the predict power of classifier in a classification network.

This report has implemented a simple three-layer network with 10 hidden neurons, 2 output neurons and a dynamic dimension of input layer decided by the dimension of real input data.

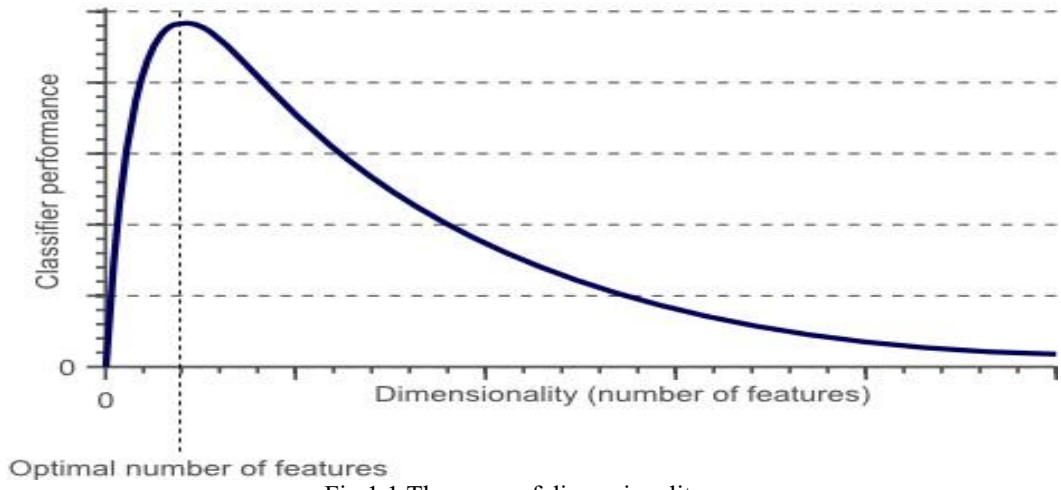

Fig 1.1 The curse of dimensionality

1.3 Feature selection

In machine learning area, feature selection, also known as attributes selection, is the process of selecting a subset of given features. Feature selection techniques are used in this report to avoid the curse of dimensionality.
As shown above (Fig 1.1), with the rise of dimensionality, the performance of classifier increases at first but then decreases. The occurrence of this phenomenon is due to the existence of some features that are either redundant or irrelevant. Feature selection is used to remove both redundant and irrelevant features.

1.4 Terminology

Here are some terminologies has been used in this report.

- Population: The evolution of species is based on group which is called population.
- Individual: Every member in a population is called individual.
- Gene: A gene encodes a sequence of DNA or RNA.
- Chromosome: A chromosome contains a group of genes. Each individual has its special chromosome.
- Crossover & mutation: Crossover is a process that an exchange of genetic material between chromosomes when generate offspring. During this process, mutation results from some errors which lead to the change of genes.
- Fitness: Fitness is an individual's ability to propagate its genes.

2 Method

2.1 Data processing

The first step is to process the data. In this dataset, all the data are divided into two classes, including "good" and "bad". We replaced all the "good" with value 1 and "bad" with value 0. Then, the data was loaded and wrapped by tensors and variables.

2.2 Implementation: simple three-layer network

The network has three layers, including dynamic number of input neurons, 10 hidden neurons and 2 output neurons. The optimizer and loss function used in this network are Stochastic Gradient Descent (SGD) and cross-entropy loss function. For each training process, we set 500 as the epoch number and 0.03 as the learning rate.

2.3 Feature selection

The flowchart below illustrates the basic framework of feature selection (Fig 2.1). Fitness evaluation happens twice in the process. The first one is used to evaluate each chromosome in the original population, while the second one is in the loop, which aims to evaluate the fitness value of every new chromosome in the population.

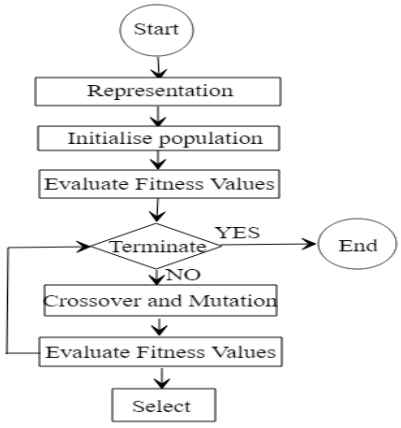

Fig 2.1 Flowchart of feature selection

2.3.1 Representation

In this case, we defined each chromosome has 34 genes, and each gene controls the representation of one feature. Here the chromosomes are formed in the sequence of binary values as shown in figure 2.2, in which a chromosome represents a feature subset, and only the bit with value 1 indicates the feature is in the subset.

Fig 2.2 example of chromosome

2.3.2 Initialise population & Crossover & Mutation

First, we generated the initial population composed of a series of random individuals, and then put a special individual with a chromosome representing all the features (with a sequence of 1) into the population.
Crossover is considered in the chromosome generation of siblings, and we mainly focused on the uniform crossover. At each crossover stage, a mask is randomly generated and the chromosomes of two individuals in the population are selected randomly as the parental chromosomes (Figure 2.3).
The result of Uniform crossover is the generation of two siblings. A sibling inherits the genes from parent 1 where the corresponding positions in mask have the value 1, while the genes are gained from parent 2 where the corresponding positions in mask have the value 2. On the contrary, the other sibling inherits the genes from parent 2 where the corresponding positions in mask have the value 1, while the genes are gained from parent 1 where the corresponding positions in mask have the value 2.

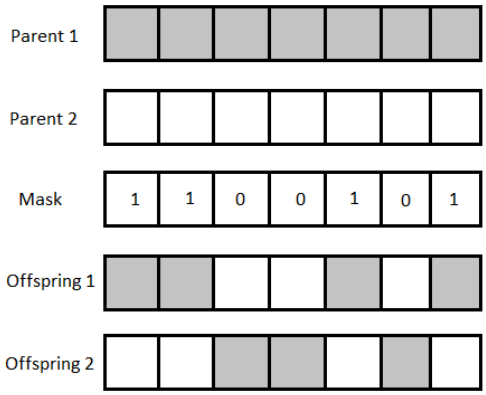

Fig 2.3 example of uniform crossover

The purpose of mutation is introducing diversity to prevent chromosomes from becoming too similar to each other, which can slow or even stop evolution of the population.
As the mutation happens according to a preset mutation probability, in this report, we implemented the mutation operator by generating a random variable for each crossover operation. It can be seen from figure 2.4, the fourth bit value should be 0, but it flips to 1.

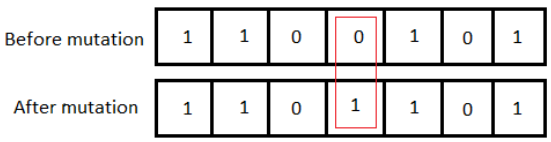

Fig 2.4 mutation

2.3.3 Evaluation & Selection

Evaluation function is used to evaluate fitness value for each chromosome containing information of a subset of features. In this report, the fitness value represents how well the subset of features can train neural network, therefore, it can be measured by the accuracy of chromosome classification after training.

Selection is used to select chromosomes that will proceed to next generation. Here are some common strategies: 1) remove the oldest individuals from the population; 2) remove those individuals with chromosomes having the minimum fitness in the population; 3) stochastic acceptance, which means choose some individuals randomly. This report used the second strategy by implementing a priority queue, which stores the fitness value of each chromosome sorted by descending fitness values. For each loop, after removing the first 10% individuals from the population, the rest individuals involved in the next reproduction.

2.3.4 Terminate

The process of feature selection can be terminated by setting a max number for population. This report chooses 1000 because this number can get the result quickly and is relatively enough to find a better chromosome then the original one.

3 Results and Discussion

This report has implemented a simple three-layer neural network and a feature selection genetic algorithm. The outcome of this implementation has been visualized (figure 3.1). According to figure 3.1, the maximum fitness at the 0^{th} generation is only 84%, while after six-generation iterations, we can find a better subset of features and accuracy increases to about 92%, so the existence of some irrelevant or redundant features that influence the accuracy can be confirmed.

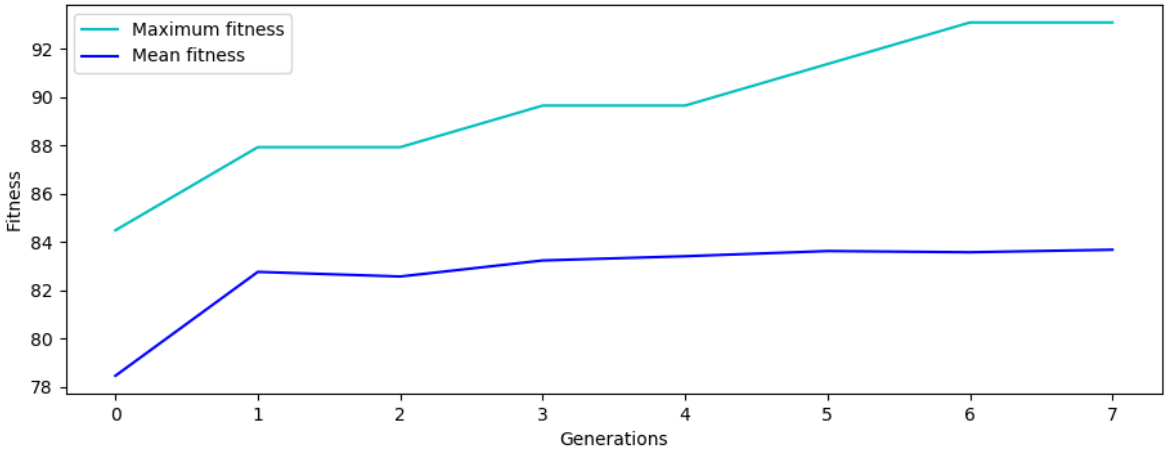

Fig 3.1 outcome

To evaluate the work better, some comparison with other published paper has been made. Kim and Park (2004) used the same dataset to examine the data reduction in support vector machines and achieved the test accuracy of 95.20%. There are several potential reasons for the difference of the two results. First reason might be the number of training epoch used in this report, the number is 500 which is relatively too small to get a higher accuracy. Second, in this report, we only iterated for 7 generations and more iteration is needed in order to generate a better individual. The most important reason might be Kim and Park implemented a kernelized ionic interaction model, which is obviously complex compared to our three-layer model.

4 Conclusion and Future Work

The report used the genetic algorithm to enhance the predict power of classifier in a neural network. Using the structure of the simple three-layer back-propagation network, we have found that feature selection can remove redundant and irrelevant features in a dataset to increase the classification accuracy.

For future work, it is necessary to learn more neural network structure and model, for example, the kernelized ionic interaction model (Kim & Park, 2004) and the comparison between different models could be developed. By summarize of advantages of different models, more accurate neural network model can be implemented.

References

Dich, W., Grudzinski, K., & Diercksen, G. H. (1998). Minimal distance neural methods. *Neural Networks Proceedings, 2*, 1299-1304. doi:10.1109/ijcnn.1998.685962

Dy, J. G., & Brodley, C. E. (2004). Feature Selection for Unsupervised Learning. *The Journal of Machine Learning Research, 5*, 845-889.

KIM, H., & PARK, H. (2004). Data Reduction in Support Vector Machines by a Kernelized Ionic Interaction Model. *Proceedings Of The 2004 SIAM International Conference On Data Mining*, 507-511. doi:10.1137/1.9781611972740.56

Sigillito, V. (1989, 1 1). *Ionosphere Data Set*. Retrieved from UCI Machine Learning Repository: https://archive.ics.uci.edu/ml/datasets/Ionosphere