PhUSE US Connect 2018

Paper AB07

Feature Selection for Predictive Modelling - a Needle in a Haystack Problem

Munshi Imran Hossain, Cytel Statistical Software & Services Pvt. Ltd., Pune, India
Sudipta Basu, Cytel Statistical Software & Services Pvt. Ltd., Pune, India

ABSTRACT

Feature extraction from data usually leads to many more features than what one would use in predictive models. With hundreds of features being par for the course, selecting the best features for a model is the search of a combination in a solution space containing more than 10^{30} combinations. This is akin to finding a needle in a haystack!

Genetic algorithms (GA) are a class of algorithms that are well suited for searching in this huge solution space. These algorithms are inspired from ideas of evolutionary biology and have analogous operations such as reproduction, crossover and mutation that can be used to arrive at optimal solutions.

This presentation discusses the application of GA for selecting features to model data obtained from a device trial. The solution will be implemented using R. The optimal combination selected by the GA will also be discussed in terms of performance relative to other solutions.

INTRODUCTION

With the availability of large amount of data, building complex models for prediction of outcomes has become an important part of most businesses. This is true of the banking, financial services and insurance industry, telecoms, manufacturing and healthcare among many others.

The basic idea is that some data is used to fit a model. This model is then used to predict output on observations that it has not encountered. By fitting a model, we really mean to estimate the parameters of the model. The data that is used for this purpose is called training data. A separate set of data, the test data, is then used to compute the accuracy of the model in modelling data that it has not been trained on.

Each such model has some independent variables. These independent variables are also called **features of the model**. The features are usually specific to the domain of a problem. For example, the features used in a model for facial recognition are different from those that would be used for predicting the three dimensional structure of proteins.

THE PROBLEM OF FEATURE SELECTION

Feature selection, as the name suggests, is the process of selecting the features to be used in the predictive model. This can be done by domain experts with their knowledge of the domain of the problem. The other way to do this is by using algorithms for automatic selection, free of manual intervention. In this paper, we are dealing with the problem of automatic feature selection.

In many problems, there can be over a hundred features. Not all these features are well suited for use in the model. The objective is to find the correct subset of these features that can result in a model with optimum performance. Considering that each feature can either be present or absent from a model, the number of possible combinations, for more than a hundred features, can be well over a billion.

One way to find the best model could be to use each of these feature combinations to build a model with the training data and test its performance on test data. This is a brute force method. This will likely take hundreds of hours of computing on modest server machines too. Clearly, this is not a feasible solution.

Fortunately, there are other methods that can find optimal subset(s) of features with lesser computing time. Genetic algorithms (GA) are a kind of algorithm that can be used for feature selection. In this paper, we are going to discuss the use of GA for feature selection and provide some results of our work.

THE GENETIC ALGORITHM

Genetic algorithms are numerical optimization algorithms that are inspired by ideas from natural selection and evolutionary biology. The method is quite generic, which means that it can be used to solve optimization problems of a wide range.

Conventional methods of optimization are based on calculus. However, these have constraints such as getting trapped in local optima. Another shortcoming of such methods is that they are based on the existence of derivatives. This condition is difficult to satisfy for objective functions for many problems. In problems where calculus-based optimization methods are not suitable, GA can be useful for optimization.

The promise of these algorithms can be gauged from the wide variety of problems in which they have been successfully applied. Some of the areas in which GA have been used are -

1. Prediction of three dimensional protein structure
2. Automatic evolution of computer software
3. Training and designing artificial neural networks
4. Image processing
5. Job shop scheduling

Before we understand the algorithm, it is important to know the lexicon of GA. The basic terms used while working with GA are described in the following section.

IMPORTANT TERMS RELATED TO GENETIC ALGORITHMS

Some of the terms that are most commonly used while talking about GA are described in this section.

- **Population –** Genetic algorithms work with a group of guess solutions. This group of solutions is called a population of solutions.

- **Chromosome –** Each guess solution in a population is called a chromosome.

- **Fitness –** This is a metric that measures how good a particular solution (chromosome) is at solving the problem.

- **Selection –** This is the process by which some solutions of a population are chosen to generate new solutions. This is analogous to natural selection in biological systems.

- **Crossover –** This is the process of exchange of information between the population members that are selected for generating new solutions. This is analogous to the exchange of genetic material between biological organisms during sexual reproduction.

- **Mutation –** This is a random change in a solution. It is analogous to the mutations that occur in the genes of a chromosome.

- **Generation -** This refers to each iteration of the algorithm. All chromosomes of a population that compete for being selected based on their fitness are said to belong to one generation. New chromosomes that are generated as a result of selection, crossover and mutation are said to belong to the next generation.

As can be seen from these terms, genetic algorithms largely draw from ideas in genetics and evolutionary biology.

A SCHEMATIC GENETIC ALGORITHM

The following figure is a schematic representation of the flow of a simple genetic algorithm.

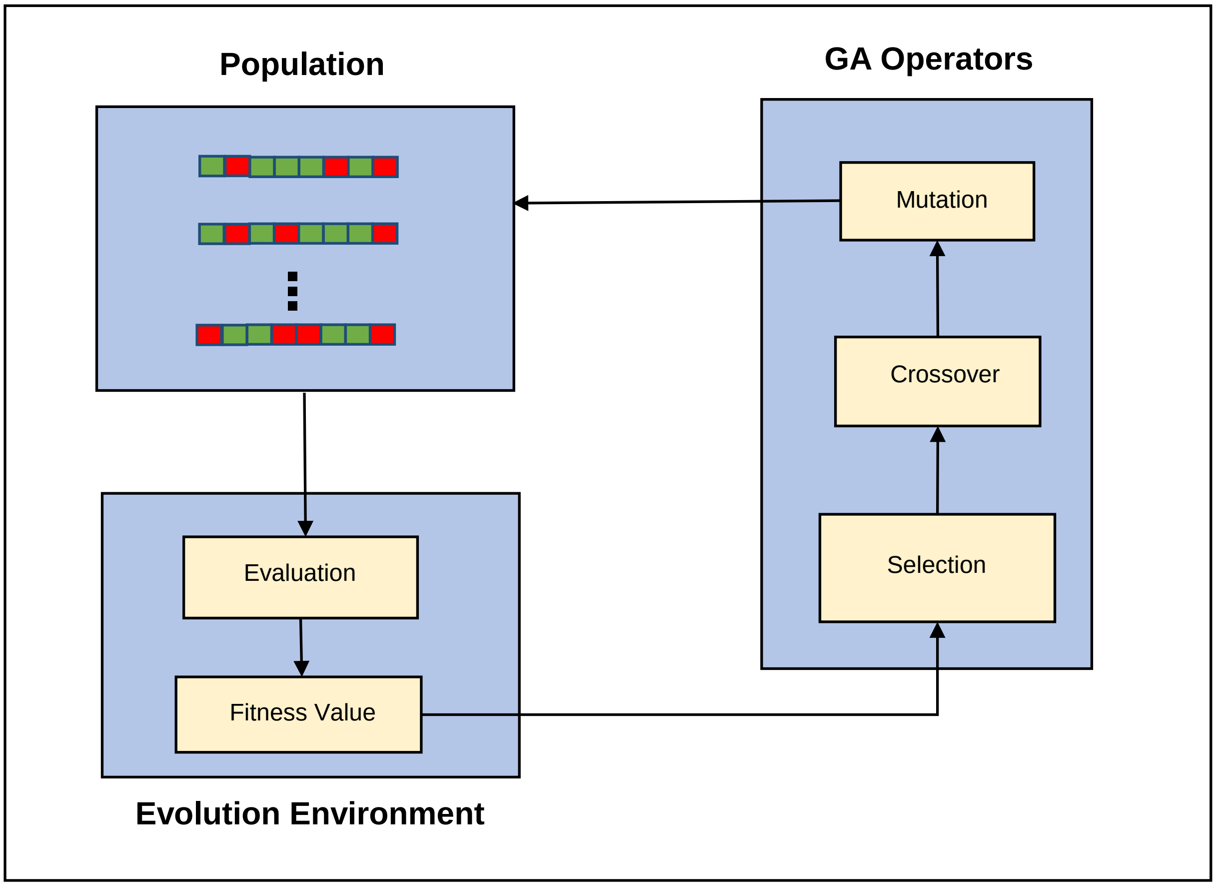

Figure 1: Schematic representation of a Genetic Algorithm

The algorithm begins with a population of solutions, also called chromosomes. The most common representation of solutions is in the form of binary numbers. In the figure, let us assume that the red cells represent *0* and the green cells, *1*.

Each solution is applied on the problem and a fitness value is evaluated for the solution. This fitness is evaluated according to a defined metric. For example, in finding the root of an equation, the fitness may be the magnitude of the difference between the value obtained by evaluating the equation with the guess solution and the actual value. The lesser this difference, the fitter the chromosome.

After the fitness of all chromosomes in a population is evaluated, selection is applied to the chromosomes. This process selects some chromosomes of the population for crossover. The process is stochastic in nature with a higher probability of selection being assigned to fitter chromosomes. There are various methods of selection, one of the most common being Roulette Wheel Sampling. The number of chromosomes to be selected is a tunable parameter that can be selected by the user.

The selected chromosomes are then used for crossover. Crossover is a process in which information is exchanged between two parent chromosomes to generate new chromosomes. In our schematic, chromosomes are strings of binary numbers. Crossover means cleaving of two parent chromosomes at a common location and concatenating pieces of one chromosome with those of another. This results in new strings of binary numbers that represent new solutions. The following figure is a pictorial representation of a single-point crossover.

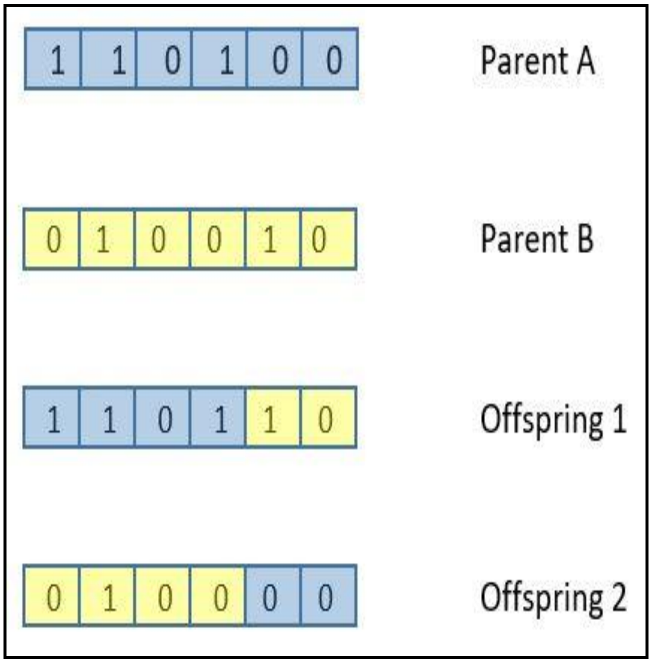

Figure 2: Schematic representation of single-point crossover between positions 4 and 5

New offspring chromosomes are subjected to mutation with a low probability. Mutation is the flipping of a *1* at a particular location in a chromosome to a *0* and vice versa. Mutation is an important process because it helps to maintain genetic diversity. It helps to keep sufficient diversity in the chromosomes of a population for generating new solutions in future generations.

The processes of selection, crossover and mutation result in new chromosomes that belong to a next generation. These chromosomes are again applied to the problem and evaluated for fitness. Selection, crossover and mutation are again applied on these to get chromosomes for another new generation. This process is repeated until we get a solution of optimum fitness or until a pre-specified number of generations have elapsed.

THE USE OF GA FOR FEATURE SELECTION

Now that we have a basic understanding of the idea of genetic algorithms, let us understand how the method can be used to solve our problem of feature selection. The objective is to find the subset of features that will result in a model with optimal discriminatory power on the test data.

The chromosomes for this problem are defined as arrays of *1* s and *0* s of length equal to the total number of independent variables. Each independent variable is assigned a fixed position in the chromosome with a *1* in that position representing the presence of the variable in the model and a *0*, its absence from the model.

A population of chromosomes is, therefore, a collection of subsets of features, such that each subset can be used to build a model. The model parameters are estimated using the training data and the fitness of the model is evaluated by measuring its accuracy in predicting the output of the test data. After each chromosome is assigned a fitness value, the population is subjected to the processes of selection, crossover and mutation to generate a new population of chromosomes.

In figure 2, if parent A represents a feature subset consisting of the first, second and fourth features and parent B represents a feature subset consisting of the second and fifth features, then the crossover has resulted in offspring 1 which is a feature set containing the first, second, fourth and fifth features and offspring 2 containing the second feature. The crossover has therefore resulted in new solutions that can be applied to the problem and evaluated for fitness.

Over multiple generations, the algorithm produces solutions of increasing fitness. The process is stopped when a set number of generations have been completed or when the fitness over many generations does not change significantly. The fittest chromosome(s) of the last generation contain the features that are best suited to model the data.

The data in this case is generated from a device that is used to monitor breathing. The data consists of measurements on more than a hundred health parameters of subjects. The number of subjects is a little over 1500. The objective is to determine whether the breathing action of a subject is normal or labored. This is a classification problem in which we wish to find the set of features that will be used to build a linear discriminant analysis (LDA) model. This model will be used to classify breathing of subjects as being either normal or labored.

RESULTS

The chromosomes of each population were used to build LDA models. The data was split into training and test sets in a ratio of 80:20. 10,000 such splits were made randomly and on each such split, the LDA model was built using those features that were marked as present in the chromosome. The model was then used to predict the outcomes on the test data and area under the receiver operating characteristic curve was computed. The median value of the area under the curve (AUC) over the 10,000 simulations was used as the fitness value of the chromosome.

Each generation of the GA had a population size of 100 chromosomes and the algorithm was run over 100 generations. Selection was done using roulette-wheel sampling and mutation was done with a probability of 1 over the length of a chromosome. The following graph shows the performance of the algorithm over generations.

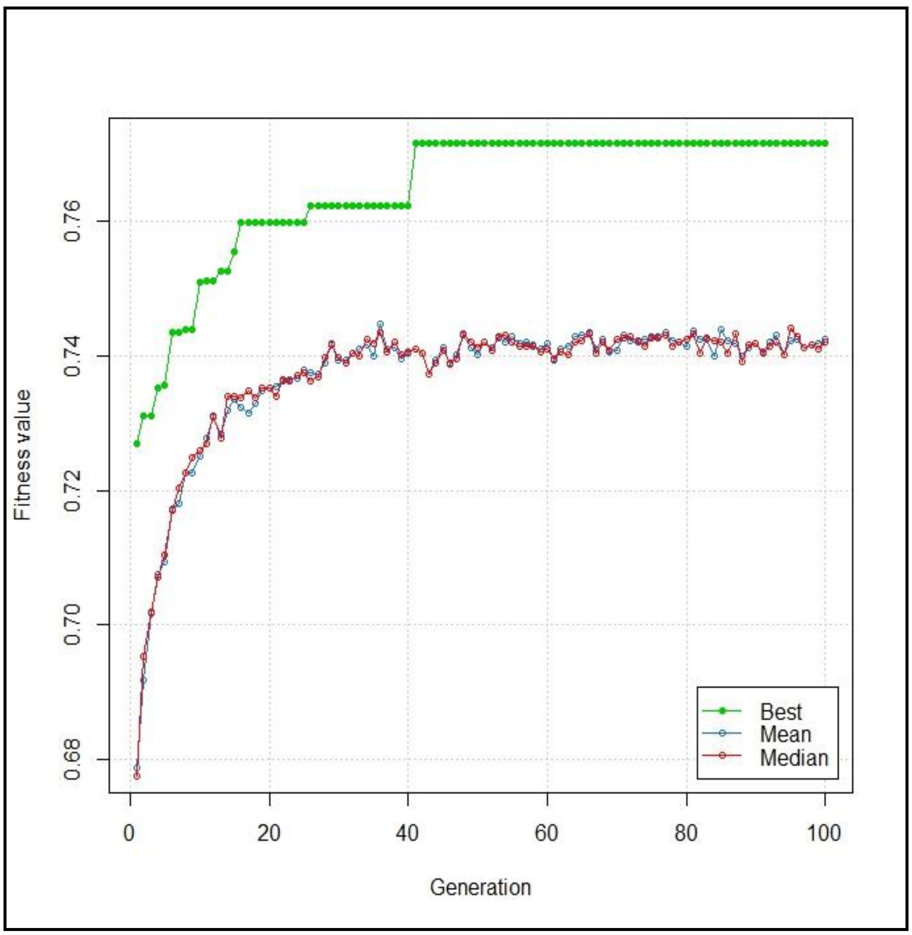

Figure 3: Plot of fitness value of chromosomes across generations

CONCLUSION

In the era of big data, when the search space is too complex, conventional optimization methods are sometimes not agile enough to solve the problem. In such cases, genetic algorithms can prove to be a promising method for solving problems. With higher computing power being easily accessible from cloud computing infrastructures, genetic algorithms can be applied for solving problems in reasonable time.

REFERENCES

1. Goldberg, David E., Genetic Algorithms in search, Optimization and Machine Learning, 1989, Pearson Education Inc.

2. Coley, David A., An Introduction to Genetic Algorithms for Scientists and Engineers, 1999, World Scientific Publishing Co. Pte. Ltd.

3. Hossain, Munshi Imran, (2018, January 15), Addressing the Problem of Feature Selection Using Genetic Algorithms [Blog post]. Retrieved from www.cytel.com/blog/feature-selection-using-genetic-algorithms

ACKNOWLEDGMENTS

We would like to thank our colleagues at Cytel for their support, co-ordination and valuable comments. Special thanks to Mr. Ajay Sathe, Mr. Rajat Mukherjee and Mr. Aniruddha Deshmukh for their help and guidance.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name	Munshi Imran Hossain, Sudipta Basu
Company	Cytel Statistical Software and Services Pvt. Ltd.
Address	S. No. 150, Lohia-Jain IT Park, 6th floor, Paud Road, Kothrud.
City / Postcode	Pune - 411038, Maharashtra, India.
Work Phone:	+91 20 6709 0230, +91 20 6709 0228
Email:	munshiimran.hossain@cytel.com, sudipta.basu@cytel.com
Web:	www.cytel.com