Developing a Scorecard using a Simple Artificial Immune System (SAIS) Algorithm and a Real-World Unbalanced Dataset

K. Leung* and F. Cheong and C. Cheong

School of Business Information Technology, RMIT University,
Melbourne, Victoria, Australia
**E-mail: kevin.leung@rmit.edu.au*
www.rmit.edu.au

S. O'Farrell and R. Tissington

Credit Analytics Department, ANZ Bank
Melbourne, Victoria, Australia

A simple artificial immune system (SAIS), which was previously developed, can predict class outcomes accurately and therefore has good classification accuracy, which is the percentage of correctly classified data. Classification accuracy works well on balanced datasets; however, since in this study, a large unbalanced dataset was obtained, classification accuracy cannot be used as a measure of performance. Instead, the Gini coefficient, which is the main performance measure used in industry for generating scorecard and which is insensitive to changes in class distribution, will be used. SAIS was modified to generate a Gini coefficient and an investigation of its suitability for scorecard development was made. We found that further modifications are needed in order for it to perform as well as logistic regression, which is the main technique used in practice for developing scorecard.

Keywords: Credit scoring; Unbalanced dataset; Artificial immune system

1. Introduction

Credit scoring, being one of the earliest financial risk management tools developed, has become one of the most successful applications in banking and finance [1]. Due to its rapid growth since the 1960s, there has been a lot of research on the use of different methods, ranging from conventional statistical to artificial intelligence techniques, to improve credit scoring models or scorecards.

In our previous studies [2,3], we developed a scorecard using an artificial immune system (AIS) algorithm, which we called Simple AIS (SAIS). The scorecard was tested on three benchmarks datasets and was found to be very competitive. While the classification accuracy, which is the percentage of correctly classified 'good' (those who are likely to repay their financial obligations) and 'bad' (those who are likely to default) cases and which is suitable for balanced datasets (equal number of 'good' and 'bad' cases), was used as one of the main performance measures, the same cannot be used in this study. This is because a real unbalanced credit scoring dataset containing 93% of 'good' and 7% 'bad' classes is used for scorecard development.

The Gini (G) coefficient is the main performance measure used in practice and is more suitable for this study since it is unaffected by the presence of unbalanced classes in the dataset. However, the G coefficient is generated from a probability or a score which represents the degree of confidence that an applicant will be categorised into a particular class. Since SAIS is a discrete classifier that can only produce a class decision (i.e. 'good' or 'bad'), there is a need to modify the algorithm so that it can generate a score (representing a degree of confidence) in order to calculate the G coefficient. This study, thus, discusses the modifications made to SAIS and investigates its suitability for scorecard development using a real unbalanced credit scoring dataset.

2. A Review of SAIS

The SAIS algorithm is based on the natural immune system of the human body in that it adopts the concept of affinity maturation which deals with stimulation, cloning and mutation of cells. It is able to generate a compact classifier using a predefined number of exemplars per class.

The SAIS algorithm is shown below while key concepts are explained in Table 1.

```
Load antigen population (training data)
Current B-cell ← randomly initialized B-cell
repeat
  Evolve the B-cell by cloning and mutation
  Evaluate mutated B-cells by calculating their classification performance
  New B-cell ← mutated B-cell with best performance
  if performance of new B-cell > current B-cell
    then Current B-cell ← new B-cell
  endif
until maxIteration
Classifier ← current B-cell
```

Table 1. Key Terms of SAIS

Name	Description
clonalRate	A value used to determine the number of mutated clones an exemplar is allowed to produce
Exemplar	Part of a B-cell
hyperMutationRate	A value used to determine the number of mutated clones generated into the cell population
maxIteration	Maximum number of iterations (use to stop the training process)
probMutation	Probability that a given clone will mutate

Readers are recommended to read [4] for a detailed discussion of the SAIS algorithm; however, a simple description of the operation of the SAIS algorithm is as follows:

(1) A set of training data (referred to as 'antigen' in the algorithm) is loaded and an initial classifier is created as a single B-cell containing a predefined number of exemplars initialized from random values. This B-cell represents the complete classifier.
(2) An evolution process is then performed and iterated until the best possible classifier is obtained. The current B-cell is cloned and the number of clones that can be produced is determined by the clonal and hypermutation rates. Mutants are then generated by using the hypermutation process found in natural immune systems. More specifically, this is achieved by randomly mutating the attributes of each clone created.
(3) The classifier is then evaluated by using the classification performance. The classification performance is a measure of the percentage of correctly classified data. If the classification performance of the best mutant is better than that of the current B-cell, then the best mutant is taken as the current B-cell.
(4) The current B-cell represents the classifier.

Two advantages of the SAIS algorithm are: 1) optimizations are performed at a global level, i.e. they are based on the classification performance of the whole classifier as compared to other AIS classifiers where optimizations are performed locally, and 2) no population control mechanism is required as the classifier is made up of only one exemplar per class. The algorithm therefore generates a very compact classifier.

SAIS has been thoroughly tested using both credit scoring datasets [2,3] and other types of datasets [4] and was found to be very competitive. The 'minimum distance' classification method of SAIS, which is adapted from instance-based learning, was found to perform best and will consequently be used in this study. However, since this method can only produce class decisions, SAIS was modified to produce a score which represents the degree of confidence to which an instance belongs to a member of a class.

3. A Modified SAIS Algorithm

The first modification was done on the 'minimum distance' method, which was found to be, by far, the most consistent method compared to discriminant analysis and polynomial techniques [4]. The 'minimum distance' method makes use of the heterogeneous Euclidean-overlap metric (HEOM) distance function. However, Johnson and Wichern [5] advocate that Euclidean distance is unsatisfactory for most statistical purposes since each variable contributes equally to the calculation of the distance. Instead, they propose to use a statistical distance measure that accounts for differences in variation.

One way of doing this is to divide each variable by the sample standard deviation, thereby standardizing the variables making them on an equal footing with one another. The statistical distance is defined as follows:

$$D_{total}(x_1, x_2) = \sqrt{\sum_{i=1}^{n} \left(\frac{D_{ed}(x_{1,i}, x_{2,i})^2}{sd(x_{2,i})^2} \right)} \tag{1}$$

where D_{total} is the total distance, x_1 is an exemplar, x_2 is an antigen (training data), n is the number of attributes, sd is the standard deviation and d_{ed} is the Euclidean distance.

The second modification was made in order to generate the degree of confidence (score) an instance is a member of a particular class. By using the two distances obtained from both exemplars *good* and *bad* (note that this study makes use of one exemplar per class and since there are two classes, two exemplars are generated by the model), the degree of confidence or score can be calculated as follows:

$$S_{Good} = \left(\frac{D_{Bad}}{D_{Good} + D_{Bad}} \right) \tag{2}$$

$$S_{Bad} = \left(\frac{D_{Good}}{D_{Good} + D_{Bad}} \right) \tag{3}$$

The degree of confidence or score is then used to calculate the G coefficient.

4. Experiments

The credit scoring dataset was obtained from a major Australian bank and consists of many different types of variables (138 in total) with 37,766 records. It is also highly unbalanced having 93% of 'good' and 7% 'bad' cases. Before using the dataset, it was first cleaned at both record and attribute levels. The clean dataset is made up of 50 variables with 15,576 records. A stepwise regression technique was used to obtain the most relevant variables and 20 attributes were selected for scorecard development. Through the use of a stratified sampling method, the dataset was divided into an 80% training set and a 20% testing set. The training set is used to develop the SAIS scorecard while the testing set is used to test its performance. The system parameters of SAIS used are shown in Table 2.

Table 2. System Parameters of SAIS

Name	Description
clonalRate	Default value = 10
hyperMutationRate	Default value = 10
	Number of clones that can be mutated = 100 (10 × 10)
maxIterations	Maximum number of iteration = 600
probMutation	Probability of mutation = 0.7

To compare the performance of our model, two other techniques were used:

(1) Logistic regression (LR) is the main technique used in practice for generating a scorecard. It is designed to predict the probability of an event (for example, granting credit) happening. It assumes that the log likelihood ratio (odds) is linear and takes the form of:

$$\log \left(\frac{y}{1-y} \right) = c + \sum_{i=1}^{n} w_i x_i \tag{4}$$

where y is the probability of classification outcome, c is a constant, w is the weight of each attribute and x is the independent attribute.

Based on (4), the value of y can be generated; hence, the G coefficient can be directly calculated.

(2) Artificial immune recognition system (AIRS) has now become a benchmark model in the field of AIS. Based on the principle of resource-limited AIS and developed by Watkins et al. [6], AIRS has proved to be a very powerful classification tool having been ranked among the top five to eight classifiers when compared to the 30 best classifiers on publicly available classification problem sets. However, very much like SAIS, its outcome is a class decision. Therefore, we have taken the average distance, which is used in the class decision, as a score to calculate the G coefficient. That was done with the value of 'k nearest neighbour' (refer to system parameters of AIRS in [6]) equal to 1 and 7. The best result was obtained with 'k' equal to 7.

The results of the three models are shown in Table 3.

Table 3. Gini Coefficient

Model	Training	Testing
SAIS	0.420	0.360
AIRS	0.134	0.230
LR	0.544	0.580

A typical scorecard would have a G coefficient ranging from 40%-70%. The results clearly indicate that LR performs very well with G coefficient above 50%. This is probably why most financial institutions are still using this method for scorecard development.

LR also seems to outperform both AIS models. One possible reason as to why the AIS models did not perform well can be due to the fact that they are not designed to generate a score which represents the degree of confidence, to discriminate between 'good' and 'bad' applicants. As a result, the G coefficients generated are not as high as for LR.

5. Conclusion

While many artificial intelligence models, including SAIS, are able to accurately predict class decisions, most of them do not make use of the G coefficient as a performance measure, probably because few studies make use of unbalanced datasets or do not consider the inaccuracy that may arise when using the percentage of correctly classified cases as measure of performance. In this study, we have modified our SAIS algorithm so that it is able to produce a score, which represent the degree of confidence to which an instance will belong to a class membership and which is used to calculate the G coefficient. We investigated the suitability of SAIS for developing a scorecard using a real unbalanced dataset obtained from a leading Australian bank.

It was found that SAIS did not perform as well as LR. AIRS, which is another well-known AIS technique, did not do that well too. This is probably because they are designed to generate class decisions, not a score (degree of confidence) and consequently, G coefficient.

The fact that LR outperforms SAIS suggests perhaps that there is a need to modify our algorithm again. In the current SAIS algorithm, the B-cells are evolved, through cloning and mutation, with the aim of improving the classification accuracy. We could perhaps change the algorithm so that the B-cells are evolved based on the probability of default and this could be a potential area for future research.

Another possible way to improve the G coefficient for SAIS is to change the way the distance measure and consequently the way the scores are generated. The statistical distance used in this study assumes that the variables are independent to each other. If that assumption does not hold, Johnson and Wichern [5] suggest rotating the original coordinate of the system through an angle. Again, there is a need to further investigate this feature as part of a future work.

References

1. L. C. Thomas, D. B. Edelman and J. N. Crook, *Credit Scoring and Its Applications* (Elsevier Science Publishers, North-Holland, Amsterdam, 2002).
2. K. Leung, F. Cheong and C. Cheong, Consumer credit scoring using an artificial immune system algorithm, in *Proc. CEC2007: IEEE Congress on Evolutionary Computation*, (Singapore, 2007).
3. K. Leung, F. Cheong and C. Cheong, A comparison of conventional and simple artificial immune system (SAIS) techniques in consumer credit scoring, Under review, (2008).
4. K. Leung, F. Cheong and C. Cheong, *IEEE Transactions on Systems, Man and Cybernetics, Part B* **37**, 1344 (2007).
5. R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis* (Prentice Hall, Upper Saddle River, N.J., 2002).
6. A. Watkins, J. Timmis and L. Boggess, *Genetic Programming and Evolvable Machines* **5**, 291 (2004).