Bond Rating with πGrammatical Evolution

Anthony Brabazon[1] and Michael O'Neill[1]

Natural Computing Research and Applications Group,
University College Dublin, Ireland.
anthony.brabazon@ucd.ie; m.oneill@ucd.ie

1 Introduction

Most large firms use both share and debt capital to provide long-term finance for their operations. The debt capital may be raised from a bank loan, or may be obtained by selling bonds directly to investors. As an example of the scale of US bond markets, the value of new bonds issued in 2004 totaled $5.48 trillion, and the total value of outstanding marketable bond debt at 31 December 2004 was $23.6 trillion [1]. In comparison, the total global market capitalisation of all companies quoted on the New York Stock Exchange (NYSE) at 31/12/04 was $19.8 trillion [2]. Hence, although company stocks attract most attention in the business press, bond markets are actually substantially larger.

When a company issues traded debt (e.g. bonds), it must obtain a credit rating for the issue from at least one recognised rating agency (Standard and Poor's (S&P), Moody's and Fitches'). The credit rating represents an agency's opinion, at a specific date, of the creditworthiness of a borrower in general (a bond-issuer credit-rating), or in respect of a specific debt issue (a bond credit rating). These ratings impact on the borrowing cost, and the marketability of issued bonds. Although several studies have examined the potential of both statistical and machine-learning methodologies for credit rating prediction [3, 4, 5, 6], many of these studies used relatively small sample sizes, making it difficult to generalise strongly from their findings. This study by contrast, uses a large dataset of 791 firms, and introduces πGE to this domain.

In common with the related corporate failure prediction problem [7], a feature of the bond-rating problem is that there is no clear theoretical framework for guiding the choice of explanatory variables, or model form. Rating agencies assert that their credit rating process involves consideration of both financial and non-financial information about the firm and its industry, but the precise factors, and the related weighting of these factors, are not publicly disclosed. In the absence of an underlying theory, most published work on credit rating prediction employs a data-inductive modelling approach, using firm-specific financial data as explanatory variables, in an attempt to 'recover' the model used by the rating agencies. This produces a high-dimensional combinatorial problem, as the modeller is attempting to uncover a 'good' set of model inputs, and model form, giving rise to particular potential for evolutionary automatic programming methodologies such as GE.

1.1 Structure of Chapter

The next section provides a concise overview the bond rating process, followed by a sections which introduce Grammatical Evolution and its variant πGE. Next, a description of the data set and methodology adopted is provided. The remaining sections provide the results of the experiments followed by a number of conclusions.

2 Background

Several categories of individuals would be interested in a model that could produce accurate estimates of bond ratings. Such a model would be of interest to firms that are considering issuing debt as it would enable them to estimate the likely return investors would require if the debt was issued, thereby providing information for pricing the bonds. The model could also be used to assess the creditworthiness of firms that have not issued debt and hence do not already have a published bond rating. This information would be useful to bankers or other companies that are considering whether they should extend credit to that firm.

2.1 Notation for Credit Ratings

Although the precise notation used to denote the creditworthiness of a bond or issuer varies between rating agencies, the credit status is generally denoted by means of a discrete, mutually exclusive, letter rating. Taking the rating structure of S&P as an example, the ratings are broken down into 10 broad classes. The highest rating is denoted AAA, and the ratings then decrease in the following order, AA, A, BBB, BB, B, CCC, CC, C, D. Ratings between AAA and BBB (inclusive) are deemed to represent *investment grade*, with lower quality ratings deemed to represent debt issues with significant speculative characteristics (also called *junk bonds*). A 'C' grade represents a case where a bankruptcy petition has been filed, and a 'D' rating represents a case where the borrower is currently in default on their financial obligations. As would be expected, the probability of default depends strongly on the initial rating which a bond receives (Table 1). Ratings from AAA to CCC can be modified by the addition of a + or a - to indicate at which end of the rating category the bond rating falls.

Table 1. Rate of default by initial rating category (1987-2002) (from Standard & Poor's, 2002)

Initial Rating	Default Rate (%)
AAA	0.52
AA	1.31
A	2.32
BBB	6.64
BB	19.52
B	35.76
CCC	54.38

2.2 Rating Process

Rating agencies earn fees from bond issuers for evaluating the credit status of new issuers and bonds, and for maintaining credit rating coverage of these firms and bonds. A company obtains a credit rating for a debt issue by contacting a rating agency and requesting that an issue rating be assigned to the new debt to be issued, or that an issuer rating be assigned to the company as a whole. As part of the process of obtaining a rating, the firm submits documentation to the rating agency including recent financial statements, a prospectus for the debt issue, and other non-financial information. Discussions take place between the rating agency and management of the firm and a rating report is then prepared by the analysts examining the firm. This rating report is considered by a rating committee in the rating agency which decides the credit rating to be assigned to the debt issue/issuer.

Rating agencies emphasise that the credit rating process involves consideration of financial as well as non-financial information about the firm, and also considers industry and market-level factors. The precise factors and related weighting of these factors used in determining a bond's rating are not publicly disclosed by the rating agencies. Subsequent to their initial rating, a bond may be re-rated upwards (upgrade) or downwards (downgrade) if company or environmental circumstances change. A re-rating of a bond below investment grade to junk bond status (such bonds are colourfully termed fallen angels) may trigger a significant sell-off as many institutional investors are only allowed, by external or self-imposed regulation, to hold bonds of investment grade.

3 Grammatical Evolution

Grammatical Evolution (GE) is an evolutionary algorithm that can evolve computer programs in any language [8, 9, 10, 11], and can be considered a form of grammar-based genetic programming. Rather than representing the programs as parse trees, as in GP [12], a linear genome representation is used. A genotype-phenotype mapping is employed such that each individual's variable length binary string, contains in its codons (groups of 8 bits) the information to select production rules from a Backus Naur Form (BNF) grammar. The grammar allows the generation of programs in an arbitrary language that are guaranteed to be syntactically correct, and as such it is used as a generative grammar, as opposed to the classical use of grammars in compilers to check syntactic correctness of sentences. The user can tailor the grammar to produce solutions that are purely syntactically constrained, or they may incorporate domain knowledge by biasing the grammar to produce very specific forms of sentences. BNF is a notation that represents a language in the form of production rules. It is comprised of a set of non-terminals that can be mapped to elements of the set of terminals (the primitive symbols that can be used to construct the output program or sentence(s)), according to the production rules. A simple example BNF grammar is given below, where `<expr>` is the start symbol from which all programs are generated. The grammar states that `<expr>` can be replaced with either `<expr><op><expr>` or `<var>`. An `<op>` can become either `+`, `-`, or `*`, and a `<var>` can become either `x`, or `y`.

```
<expr> ::= <expr><op><expr>  (0)
         | <var>             (1)

  <op> ::= +                 (0)
         | -                 (1)
         | *                 (2)

 <var> ::= x                 (0)
         | y                 (1)
```

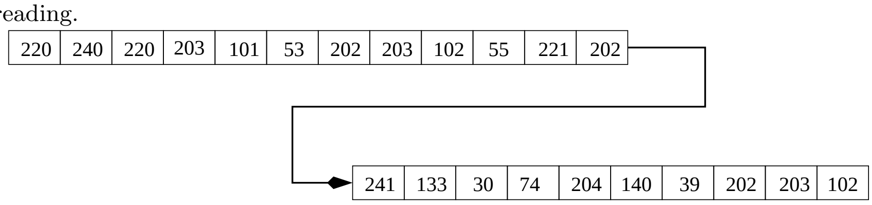
The grammar is used in a developmental process to construct a program by applying production rules, selected by the genome, beginning from the start symbol of the grammar. In order to select a production rule in GE, the next codon value on the genome is read, interpreted, and placed in the following formula:

$$Rule = Codon\ Value\ Mod\ Num.\ Rules$$

where Mod represents the modulus operator. Given the example individual's genome (where each 8-bit codon has been represented as an integer for ease of reading) in Fig.1, the first codon integer value is 220, and given that we have 2 rules to select from for `<expr>` as in the above example, we get $220\ Mod\ 2 = 0$. `<expr>` will therefore be replaced with `<expr><op><expr>`.

Beginning from the left hand side of the genome codon integer values are generated and used to select appropriate rules for the left-most non-terminal in the developing program from the BNF grammar, until one of the following situations arise:

Fig. 1. An example GE individual's genome represented as integers for ease of reading.



- A complete program is generated. This occurs when all the non-terminals in the expression being mapped are transformed into elements from the terminal set of the BNF grammar.
- The end of the genome is reached, in which case the *wrapping* operator is invoked. This results in the return of the genome reading frame to the left hand side of the genome once again. The reading of codons will then continue unless an upper threshold representing the maximum number of wrapping events has occurred during this individual's mapping process.
- In the event that a threshold on the number of wrapping events has occurred and the individual is still incompletely mapped, the mapping process is halted, and the individual assigned the lowest possible fitness value.

Returning to the example individual, the left-most `<expr>` in `<expr><op><expr>` is mapped by reading the next codon integer value 240 and used in $240\ Mod\ 2 = 0$ to become another `<expr><op><expr>`. The developing program now looks like `<expr><op><expr><op><expr>`. Continuing to read subsequent codons and always mapping the left-most non-terminal the individual finally generates the expression `y*x-x-x+x`, leaving a number of unused codons at the end of the individual, which are deemed to be introns and simply ignored. A full description of GE can be found in [8].

4 πGrammatical Evolution

The GE mapping process can be divided into a number of sub-components including the transcription and translation processes as outlined in the previous section. The πGE variant of GE replaces the translation process to allow evolution to specify the order in which production rules are mapped as opposed to the strict depth-first, left to right, mapping of the standard GE algorithm. In πGE we use the genotype to dictate which non-terminal from those present to expand next, before deciding which production rule to apply to the selected non-terminal. The genome of an individual in πGE is different in that there are two components to each codon. That is, each codon corresponds to the pair of values (*nont*, *rule*).

In the first derivation step of the example mapping presented earlier, `<expr>` is replaced with `<expr><op><expr>`. Then in the standard GE genotype-phenotype mapping process, the left-most non-terminal (the first `<expr>`) in the developing program is always expanded first. The πGE mapping process differs in an individual's ability to determine and adapt the order in which non-terminals will be expanded [13]. To this end, a πGE codon corresponds to the pair (*nont*, *rule*), where *nont* and *rule* are represented by N bits each (N=8 in this study), and a chromosome, then, consists of a vector of these pairs. In πGE, we analyse the state of the developing program before each derivation step, counting the number of non-terminals present. If there is more than one non-terminal present in the developing program the next codon's *nont* value is read to pick which non-terminal will be mapped next according to the following mapping function:

$$Non-terminal = Codon\ nont\ Value\ Mod\ Number of non-terminals$$

In the above example, there are 3 non-terminals ($\texttt{<expr>}_0\texttt{<op>}_1\texttt{<expr>}_2$) after application of the first production rule. To decide which non-terminal will be expanded next we use Number of non-terminals = 9 % 3 = 0, i.e., $\texttt{<expr>}_0$ is expanded. The mapping rule for selecting the appropriate rule to apply to the current non-terminal is given in the normal GE fashion:

$$Rule = Codon\ rule\ Value\ Mod\ Number of Rules$$

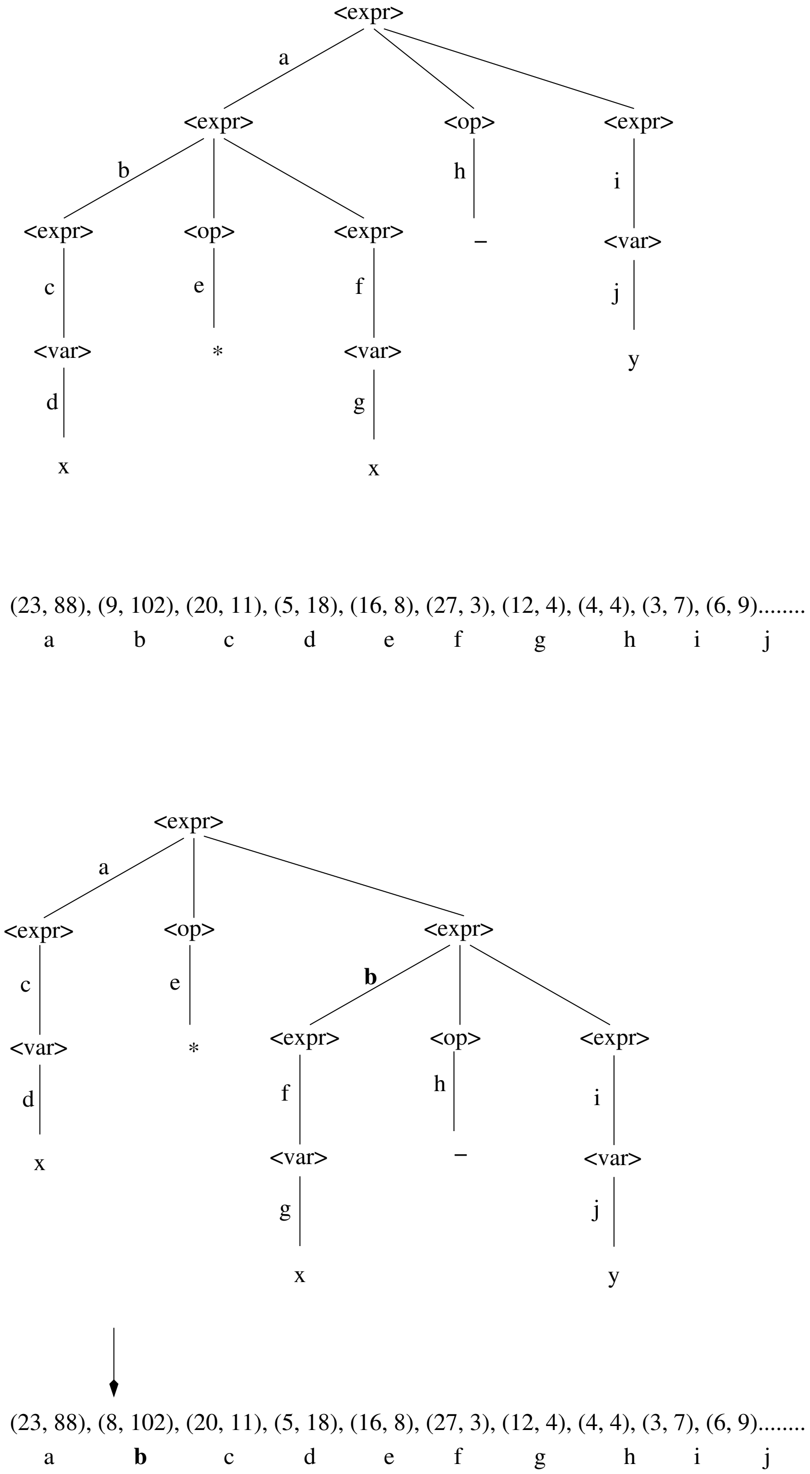
In this approach, evolution can result in a derivation subsequence being moved to a different context as when counting the number of non-terminals present we do not pay attention to the type of non-terminals (e.g. `<expr>` versus `<op>`).

An example of the application of πGE is provided in Fig. 2. In the top derivation tree, 9 Mod 3=0 (this derivation step is labelled **b**), hence the left-most non-terminal is expanded first. In the bottom derivation tree a mutation event transforms the second codon's *nont* value from 9 to 8, giving 8 Mod 3=2 (step **b**), hence the right-most non-terminal is expanded instead. The three subsequent subtrees (derivation steps labelled **c & d**, **e**, and **f & g**) that are produced are redistributed amongst other non-terminals.

In this instance, the single mutation is acting in a similar fashion to a multiple sub-tree exchange or crossover within the individual. The top derivation tree corresponds to the expression (x * x) - y, whereas the bottom tree gives x * (x - y).

We note that πGE could be implemented in more than one way. An alternative approach would be to respect non-terminal types and only allow choices to be made between non-terminals of the same type, thus preserving the semantics of the following derivation subsequence, and simply changing the position in which it appears in the developing program.

Fig. 2. An example of piGE, illustrating a single mutation event in the *nont* position in the second codon.



5 Experimental Approach

The dataset consists of financial data of 791 industrial and service US companies, along with their associated bond-issuer credit-rating, drawn from the S&P Compustat database. Of these companies, 57% have an investment-grade rating (AAA, AA, A, or BBB), and 43% have a junk rating. To allow time for the preparation of year-end financial statements, the filing of these statements with the Securities and Exchange Commission (S.E.C), and the development of a bond rating opinion by Standard and Poor rating agency, the bond rating of the company as at 30 April 2000, is matched with financial information drawn from their financial statements as at 31 December 1999. A subset of 600 firms was randomly sampled from the total of 791 firms, to produce two groups of 300 investment grade and 300 junk rated firms. The 600 firms were randomly allocated to the training set (420) or the hold-out sample (180), ensuring that each set was equally balanced between investment and non-investment grade ratings.

Five groupings of explanatory variables, drawn from financial statements, are given prominence in prior literature as being the prime determinants of bond issue quality and default risk:

i. Liquidity
ii. Debt
iii. Profitability
iv. Activity / Efficiency
v. Size

Liquidity refers to the availability of cash resources to meet short-term cash requirements. Debt measures focus on the relative mix of funding provided by shareholders and lenders. Profitability considers the rate of return generated by a firm, in relation to its size, as measured by sales revenue and/or asset base. Activity measures consider the operational efficiency of the firm in collecting cash, managing stocks and controlling its production or service process. Firm size provides information on both the sales revenue and asset scale of the firm and also provides a proxy metric on firm history. The groupings of potential explanatory variables can be represented by a wide range of individual financial ratios, each with slightly differing information content. The groupings themselves are interconnected, as weak (or strong) financial performance in one area will impact on another. For example, a firm with a high level of debt, may have lower profitability due to high interest costs.

Following the examination of a series of financial ratios under each of these headings, a total of eight financial variables was selected for inclusion in this study. The selection of these variables was guided both by prior literature in bankruptcy prediction [14, 15, 16] and literature on bond rating prediction [17, 18, 19, 20]. These ratios were then further filtered using statistical analysis. The ratios selected were as follows:

i. Current ratio

ii. Retained earnings to total assets
iii. Interest coverage
iv. Debt ratio
v. Net margin
vi. Market to book value
vii. Log (Total assets)
viii. Return on total assets

Table 2. Means of input ratios for investment and junk bond groups of companies

	Investment grade	**Junk grade**
Current ratio	1.354	1.93
Retained earnings/Total assets	0.22	-0.12
Interest coverage	7.08	1.21
Debt ratio	0.32	0.53
Net margin	0.07	-0.44
Market to book value	18.52	4.02
Total assets	10083	1876
Return on total assets	0.10	0.04

The objective in selecting a set of proto-explanatory variables is to choose financial variables that vary between companies in different bond rating classes, and where information overlaps between the variables are minimised (the financial ratios chosen during the selection process are listed at the end of this section). Comparing the means of the chosen ratios (see Table 2) for the two groups of ratings, reveals a statistically significant difference at the 1% level, and as expected, the financial ratios in each case, for the investment ratings are stronger than those for the junk ratings. The only exception is the current ratio, which is stronger for the junk rated companies, possibly indicating a preference for these companies to hoard short-term liquidity, as their access to long-term capital markets is weak. A correlation analysis between the selected ratios indicates that most of the cross-correlations are less than $| 0.20 |$, with the exception of the debt ratio and (Retained Earnings/Total Assets) ratio pairing, which has a correlation of -0.64. The grammar adopted is as follows:

```
<lc> ::= if( <expr> <relop> <expr> )

           class=''Junk'';
         else
           class=''Investment Grade'';

<expr> ::= ( <expr> ) + ( <expr> )

         | <coeff> * <var>

<var> ::= var3[index] | var4[index]

        | var5[index] | var6[index]

        | var7[index] | var8[index]
```

```
        | var9[index] |var10[index]

        | var11[index]

<coeff> ::=  ( <coeff> ) <op> ( <coeff> )

          | <float>

<op> ::= + | - | *

<float> ::= 9 | 8 | 7 | 6 | 5 | 4
          | 3 | 2 | 1 | -1 | .1

<relop> ::= <=
```

where `var3` = Current Ratio, `var4` = Retained Earnings to total assets, `var5` = Interest Coverage, `var6` = Debt Ratio, `var7` = Net Margin, `var8` = Market to book value, `var9` = Total Assets, `var10` = ln (Total Assets), `var11` = Return on total assets.

6 Results

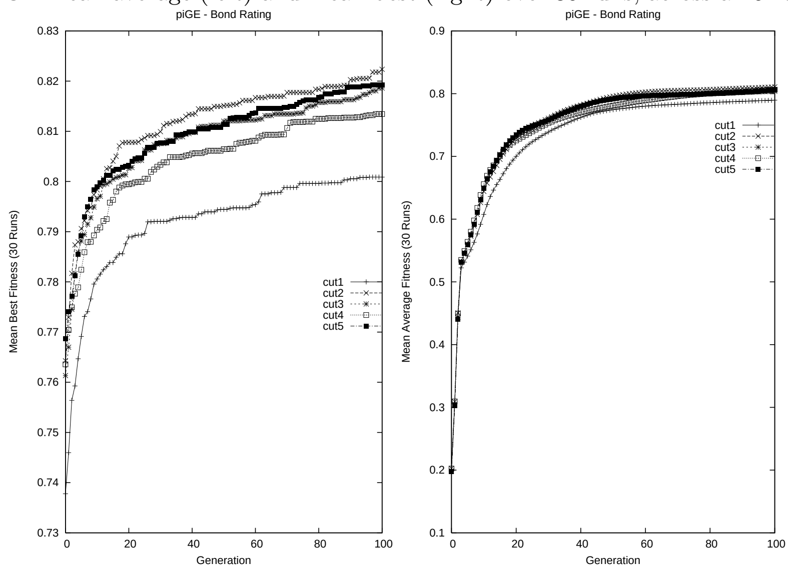
The results from our experiments are now provided. Each of the πGE experiments is run for 100 generations, with variable-length, one-point crossover at a probability of 0.9, one point bit mutation at a probability of 0.01, roulette selection, and steady-state replacement. To assess the stability of the results across different randomisations of the dataset between training and test data, we recut the dataset five times, maintaining an equal balance of investment and non-investment grade ratings in the resulting training and test datasets. In our experiments, fitness is defined as the number of correct classifications obtained by an evolved discriminant rule. The results for the best individual of each cut of the dataset, where 30 independent runs were performed for each cut, averaged over all five randomisations of the dataset, for a population size of 500 is given in Table 3, and Figure 3 displays the evolution of the mean average and mean best results over time.

Table 3. Average performance for the five recuts of the best evolved rules on their in and out-sample datasets.

	Fitness	**TP**	**TN**	**FP**	**FN**
In-sample	0.8450	182.8	172.1	37.9	27.2
Out-sample	0.8500	77.9	75.1	14.9	12.1

To assess the overall hit-ratio of the developed models (out-of-sample), Press's Q statistic [21] was calculated for each model. In all cases, the null hypothesis, that the out-of sample classification accuracies are not significantly better than those that could occur by chance alone, was rejected at the 1% level. A t-test of the hit-ratios also rejected a null hypothesis that the classification accuracies were no better than chance at the 1% level. Across all the data recuts, the best individual achieved classification accuracy of 86% in-sample and 87% out-of-sample.

Fig. 3. Mean average (left) and mean best (right) over 30 runs, across all 5 recuts.



When applying any model induction approach, it is important to reduce the possibility of overfitting. A number of practical steps to reduce the chance of overfitting include the collection of a sufficiently large dataset relative to the number of explanatory variables included in the model, and the testing of the developed model on a sizeable out-of-sample dataset. In this study we have trained the models using 420 data vectors, tested the evolved models using a sizeable out-of-sample dataset (180 data vectors), and have restricted the evolved models to use a maximum of eight explanatory variables. As noted above, the in-sample and out-of-sample classification accuracies are very similar, indicating that overfitting does not seem to have been a problem. Given that the evolved models were restricted to use a maximum of eight explanatory variables we have not implemented a regularisation term in the error function.

Examining the structure of one of the best individuals shows that the evolved discriminant function had the following form:

$$\text{IF } (\ 0 \leq \text{-}2 + \textit{Debt Ratio} - \textit{Total Assets} - 5 * \frac{\textit{Retained Earnings}}{\textit{Total Assets}})$$
$$\text{THEN 'Junk' ELSE 'Investment Grade'}$$

Examining the signs of the coefficients of the evolved rules does not suggest that they conflict with common financial intuition. The rules indicate that low/negative retained earnings, low/negative total assets or high levels of debt finance are symptomatic of a firm that has a junk rating. It is noted that similar risk factors have been identified in predictive models of corporate failure which utilise financial ratios as explanatory inputs [7, 22]. Conversely, low levels of debt, a history of successful profitable trading, and high levels of total assets are symptomatic of firms that have an investment grade rating.

6.1 Comparison of Results

To provide a benchmark for the results obtained by πGE we compare them with the results obtained on the same recuts of the dataset, using a fully-connected, feedforward multi-layer perceptron (MLP), trained using the backpropagation algorithm. The developed networks utilised all the explanatory variables. The optimal number of hidden-layer nodes was found following experimentation on each separate data recut, and varied between two and four nodes. The classification accuracies for the networks, averaged over all five recuts is provided in Table 4.

Table 4. Performance of the MLPs on the training and out-of-sample datasets, averaged over all five recuts of the dataset.

	Fitness	TP	TN	FP	FN
In sample	0.869	181.8	183.2	26.8	28.2
Out-sample	0.850	75.8	77.2	12.8	14.2

The levels of classification accuracy obtained with the MLP are competitive with earlier research, with for example [17] obtaining an out-of-sample classification accuracy of approximately 83.3%, although it is noted that the size of the dataset in their study was small. Comparing the results from the MLP with those of πGE on the initial fitness function (Table 3) suggests that πGE has proven competitive with an MLP methodology, in terms of producing a similar classification accuracy. Benchmark results were also obtained using an LDA methodology. Utilising the same dataset recuts as πGE, LDA produced results (averaged across all five recuts) of 82.74% in-sample, and 85.22% out-of-sample. Again, πGE is competitive against these results in terms of classification accuracy. Comparing the results obtained by the linear classifiers (LDA and πGE) against those of an MLP, suggests that strong non-linearities between the explanatory variables and the dependent variable are not present.

7 Conclusions & Future Work

The objective of this chapter was to introduces a novel classification system based on a variant of Grammatical Evolution, πGE, and to assess the utility of this methodology using information drawn from the financial statements of bond-issuing firms. Despite using data drawn from companies in a variety of industrial sectors, the developed models showed an impressive capability to discriminate between investment and junk rating classifications. The πGE developed models also proved highly competitive with a series of MLP models developed on the same datasets. Several extensions of the methodology in this study are indicated for future work. One route is the inclusion of non-financial company and industry-level information as input variables. A related possibility would be to concentrate on building rating models for individual industrial sectors. Another avenue of research would be to extend the grammar used by πGE in this study to encompass multi-class bond rating predictions.

References

1. Bond Market Association - Research Quarterly (2005), `http://www.bondmarkets.com`
2. NYSE (2005) Market Statistics, `http://www.nyse.com`
3. Ederington, H. (1985) Classification models and bond ratings. Financial Review 20(4):237–262
4. Gentry, J., Whitford, D. and Newbold, P. (1988) Predicting industrial bond ratings with a probit model and funds flow components. Financial Review 23(3):269–286
5. Huang, Z., Chen, H., Hsu, C., Chen, W. and Wu, S. (2004) Credit rating analysis with support vector machines and neural networks: a market comparative study. Decision Support Systems 37(4):543–558
6. Shin, K. and Han, I. (2001) A case-based approach using inductive indexing for corporate bond rating, Decision Support Systems 32:41–52
7. Brabazon, A., O'Neill, M., Matthews, R., and Ryan, C. (2002) Grammatical Evolution and Corporate Failure Prediction. In: Spector, L. et al (eds) Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002). Morgan Kaufmann, New York, pp. 1011–1019
8. O'Neill M., Ryan C. (2003) Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language. Kluwer Academic Publishers, Boston
9. O'Neill, M. (2001) Automatic Programming in an Arbitrary Language: Evolving Programs in Grammatical Evolution. PhD thesis, University of Limerick
10. O'Neill M., Ryan C. (2001) Grammatical Evolution. IEEE Trans. Evolutionary Computation 5(4):349–358
11. Ryan C., Collins J.J., O'Neill M. (1998) Grammatical Evolution: Evolving Programs for an Arbitrary Language. In: Proceedings of the First European Workshop on Genetic Programming, Springer-Verlag, Berlin, pp. 83–95
12. Koza, J. (1992) Genetic Programming. MIT Press, Boston.
13. O'Neill M., Brabazon A., Nicolau M., Mc Garraghy S. and Keenan P. (2004) πGrammatical Evolution. In: Deb, K. et al (eds) Proceedings of GECCO 2004. Springer-Verlag, Berlin, pp 617–629
14. Altman, E. (1993) Corporate Financial Distress and Bankruptcy. John Wiley and Sons Inc, New York
15. Morris, R. (1997) Early Warning Indicators of Corporate Failure: A critical review of previous research and further empirical evidence. Ashgate Publishing Limited, London
16. Altman, E. (1998) The importance and subtlety of credit rating migration. Journal of Banking & Finance 22:1231–1247
17. Dutta, S. and Shekhar, S. (1988) Bond rating: a non-conservative application of neural networks. In: Proceedings of IEEE International Conference on Neural Networks. IEEE Press, Piscataway, New Jersey, pp 443–450
18. Kamstra, M., Kennedy, P. and Suan, T.K. (2001) Combining Bond Rating Forecasts Using Logit. The Financial Review 37:75–96
19. Singleton, J. and Surkan, A. (1991) Modeling the Judgment of Bond Rating Agencies: Artificial Intelligence Applied to Finance. Journal of the Midwest Finance Association 20:72–80
20. Brabazon, A. and O'Neill, M. (2006) Biologically Inspired Algorithms for Financial Modelling. Springer, Berlin

21. Hair, J., Anderson, R., Tatham, R. and Black, W. (1998) Multivariate Data Analysis. Prentice Hall, Upper Saddle River, New Jersey
22. Brabazon, A. and O'Neill, M. (2003) Anticipating Bankruptcy Reorganisation from Raw Financial Data using Grammatical Evolution. In: Raidl, G. et al (eds) Proceedings of EvoIASP 2003. Springer-Verlag, Berlin, pp 368–378

Index