

Repository method to suit different investment strategies

Alma Lilia Garcia-Almanza and Edward P.K. Tsang

Abstract—This work is motivated by the interest in finding significant movements in financial stock prices. The detection of such movements is important because these could represent a good opportunity to invest. However, when the number of profitable opportunities is very small the prediction of these cases is very difficult.

In previous works Repository Method (RM) was introduced. The aim of this approach is to classify financial data sets in extreme imbalanced environments. RM offers a range of solutions to suit the risk guidelines of the investor.

The aims of this paper are 1) to show that RM can produce a range of solutions to suit the investor requirements and 2) to analyze the influence of the evolutionary process in the RM performance. Three series of experiments were performed, RM was tested using two artificial data sets whose solutions have different level of complexity. Finally RM was tested in a data set from the London stock market. Experimental results show: 1) RM offers a range of solutions to fit the risk guidelines of the investor and 2) the contribution of the evolutionary process is very valuable in the performance of RM and 3) RM is able to extract predictive rules even from earliest stages of the evolutionary process.

I. INTRODUCTION

This work is motivated by the interest in finding significant movements in financial stock prices. The detection of such movements is important because these could represent a good opportunity to invest.

To detect big movements in financial stock prices our approach is inspired by a previous work called EDDIE [1]. This forecasting tool trains a Genetic Program (GP) [2] using a set of examples. Its objective is to forecast future movements in stock prices. However, in some situations, when the number of profitable investment opportunities is extremely small, for example, in finding arbitrage opportunities [3]. It becomes very difficult to detect such chances.

Machine learning classifiers, like other forecasting techniques, extend the past experiences into the future. However, the imbalance between positive and negative cases poses a serious challenge to machine learning techniques [4],[5],[6],[7]. In imbalanced data sets, classifiers tend to favor the majority class because this prediction has a high chance of being correct. To illustrate this point consider a data set of 100 cases where 98 of them are negative, if the system classifies all of them as negative, the accuracy will be 98%. However, the classifier missed all positive cases, which makes it useless for picking up investment opportunities.

Alma Lilia Garcia-Almanza is with the Department of Computer Science, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK (phone: (44) 01206-873975; email: algarc@essex.ac.uk).

Edward P.K. Tsang is with the Department of Computer Science, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK (phone: (44) 01206-872774; email: edward@essex.ac.uk).

In previous works Repository Method (RM) was introduced [8], [9]. The objective of this approach is to classify financial data sets in extreme imbalanced environments. The procedure of RM is to collect multiple rules to form a more reliable classifier in rare cases. To generate many candidate solutions (decision trees) we use a GP because it naturally produces multiple solutions for a single problem. The decision trees produced by the evolutionary process are analyzed by RM to select and collect predictive rules.

RM offers a range of solutions to suit the risk guidelines of the investor. Thus the user can choose the best trade-off between the costs of miss-classification and false alarms according to his/her requirements.

The aims of this paper are 1) To show that RM is able to produce a range of solutions to suit the investor requirements and 2) to analyze the influence of the evolutionary process in the range of solutions provided by RM .

For analysis purposes three series of experiments were performed. RM was tested using two artificial data sets whose solutions have different level of complexity. Additionally RM was tested in a data set of Barclays stock prices. Experimental results show: 1) RM offers a range of solutions to fit the risk guidelines of the investor 2) the contribution of the evolutionary process is very valuable in the performance of RM and 3) RM is able to extract predictive rules even from earliest stages of the evolutionary process.

The remainder of this paper is organized as follows: section II provides a brief explanation of the problem, while section III gives a description of our approach. Section IV describes the Receiver Operating Characteristic (ROC) curve. Section V describes the experimental procedure and results. Finally, section VI summarizes the conclusions.

II. PROBLEM DESCRIPTION

EDDIE [1], [10], [11], [3] is a financial forecasting tool that trains a GP using a set of examples. Every instance in the data set is composed by a set of *attributes* or *independent variables* and a *signal*. The independent variables are indicators derived from financial technical analysis, these indicators have been used to identify patterns that can suggest future activity [12]. The signal is calculated looking ahead in a future horizon of n units of time, trying to detect an increase or decrease of at least $r\%$. However, when the value of r is very high, which implies an important movement in the stock price, the number of positive cases is extremely small and it becomes very difficult to detect these events. Given that the number of positive examples is scarce it is important to gather all available information about their detection. For that reason we propose to compile several solutions that classify in diverse ways the rare cases. Repository method

TABLE I
DISCRIMINATOR GRAMMAR

G	→	<Root>
<Root>	→	"If-then-else", <Conjunction> <Condition>,"Class","No Class"
<Conditional>	→	<Operation>, <Variable>, <Threshold> <Variable>
<Conjunction>	→	"and" "or",<Conjunction> <Conditional>, <Conjunction> <Conditional>
<Operation>	→	"<",">"
<Variable>	→	Variable ₁ Variable ₂ ... Variable _n
<Threshold>	→	Real number

is a classifier that relies on thresholds, this characteristic lets us plot the ROC curve.

To generate many candidate solutions (decision trees) we use a GP because it naturally produces multiple solutions for a single problem. The decision trees produced by the evolutionary process are analyzed by RM to select and collect solutions capable to identify the rare cases. The selection is based on the *performance* and *novelty* of the solution.

In the context of this work solutions are represented by rules. Let us define a rule $R_i \in T$ as a minimal set of conditions whose intersection satisfies the decision tree T . It means that the tree could contain one or more rules. A decision tree is satisfied when at least one of its rules is satisfied and a rule is satisfied when all its conditions are satisfied. Figure 1 shows a decision tree that hold three rules, as can be seen each of them is able to satisfy the decision tree.

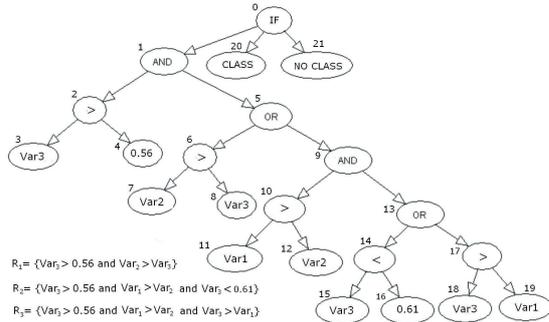


Fig. 1. The figure shows a decision tree generated by DG and its rules.

III. REPOSITORY METHOD DESCRIPTION

The Repository Method is a process which combines multiple decision trees. It picks out and collects rules from a set of decision trees. This section provides a brief description of RM, for a more detailed explanation readers are referred to [8],[9].

The Repository Method (RM) provides a means of generating prediction rule sets to suit different user needs. RM provides the choice to look for the best balance between misclassification and false alarms. Here we give an overview of the main steps of our approach:

- 1) *Creation of different solutions* - in this step a large set of candidate solutions (decision trees) is

- 2) *Rule extraction* - this procedure analyzes each decision created in the previous step in order to identify the embedded rules in the tree. Once a rule is identified, it is evaluated. If the rule achieves a predefined Precision Threshold (PT), this will pass to the next step. Otherwise it will be discarded. This process is the first filter, which selects the rule by its performance.
- 3) *Rule simplification* - the aim of this process is to simplify the rule by removing redundancy and noise. The goal is to prepare the rule for the next step (New Rule detection).
- 4) *New rule detection* - this process compares the rule against the set of previously selected rules. This process is the second filter and its objective is 1) to pick novel rules and 2) to replace old rules with *similar rules* which offer better performance.

A. Creation of different solutions

To generate multiple solutions we use a GP, because it is able to produce different solution for a single problem. However, THE normal procedure is to choose only the best individual of the evolution as the optimal solution of the problem. In contrast we presume that the remaining individuals in the population could contain useful information that is not necessarily included in the best individual of the evolution.

The decision trees in this work meet the syntax of Discriminator Grammar (DG) in Figure I. This grammar¹ helps to simplify the delimitation of rules, producing decision trees that classify or not a single class. Figure 1 illustrates a decision tree that was created using DG.

B. Rule extraction

Once the set of possible solutions (decision trees) has been generated, the next step is the rule extraction. The aim of this procedure is to analyze the decision trees to delimit their rules(see figure 1). Once a rule $R_k \in T$ has been delimited, it is individually evaluated against the training data set. If the precision of R_k achieves a predefined Precision Threshold (PT), where $PT > 0$, then R_k is considered for the next step (rule simplification), otherwise R_k is discarded. It is required that the precision of R_k is bigger than 0 because it implies that R_k has to classify at least one positive case. This constraint is very important because it discards rules composed by contradictory or unresolved conditions, (e.g. $R_1 = \{var_1 > var_1\}$ or $R_2 = \{var_1 > var_2 \text{ and } var_1 < var_2\}$).

C. Rule Simplification

The objective of rule simplification is to remove noisy and redundant conditions. Noisy conditions do not affect the decision of the rule. Redundant conditions are those which

¹The term grammar refers to a representation concerned with the syntactic components and the regulation that specify it [13]

are repeated or report the same event e.g. $R_1 = \{var_1 > 0.5 \text{ and } var_1 > 0.7\}$ the first condition is redundant. The simplification of rules is an important process because it prepares the rules for the next step (New rule detection).

Rule simplification is a hard task, specially for decision trees that were generated in the latest stages of a GP process. Because these tend to grow and accumulate introns [14],[15],[16],[17]. To simplify rules we have defined two types of conditions: *hard conditions* and *flexible conditions*. A hard condition is a comparison of two variables (e.g. $var_1 < var_2$). A flexible condition is the equation between a variable and a threshold (e.g. $var_1 < 0.8$). When two flexible conditions have the same variable and *operation* (see grammar DG) they are defined as *similar conditions* (e.g. $var_1 < 3$ and $var_1 < 2$ are similar conditions). Conditions have been divided, in hard and flexible, because the conditions that compare thresholds could be difficult to differentiate (e.g. $var_1 < 0.8912$ and $var_1 < 0.8910$); however, they can be easily simplified (e.g. $Var_1 < 0.8910$). The following steps describes this process. Let $R_k = \{c_i\}$ be the set of conditions in R_k .

- If $c_1, c_2 \in R_k$ are hard conditions and $c_1 = c_2$ then $R_k = R_k - c_2$
- If $c_1, c_2 \in R_k$ are flexible conditions and c_1 and c_2 are similar conditions then c_1 and c_2 are simplified using the *simplification table* (see [9]).
- If $c_i \in R_k$ and $Performance(R_k) = Performance(R_k - c_i)$ then $R_k = R_k - c_i$

The rule performance is determined by the fitness function that was used in the GP.

D. New rule detection

Once a rule R_k has been simplified, we have to determine the novelty of this by comparing R_k against the rules in the repository. To compare rules effectively, let R_i be a *hard rule* if it is composed exclusively of hard conditions and let R_i be a *flexible rule* if it has at least one flexible condition. R_k and R_i are *similar rules* if they are flexible rules and have the same hard conditions and similar flexible conditions. For instance: $R_1=(var_1 > Var_2, var_3 > 0.30)$ and $R_2=(var_1 > Var_2, var_3 > 0.35)$ are similar rules

When two similar rules are found we pick the rule with the best performance, it helps to tune the thresholds in the conditions. The following procedure determines if R_k is added or not to the rule repository *Rep*.

- If R_k is a hard rule and $\nexists R_i \in Rep$ such as $R_i = R_k$ then $Rep = Rep \cup R_k$
- If R_k is a flexible rule and $\exists R_i \in Rep$ such as R_k and R_i are similar rules and $Performance(R_k) > Performance(R_i)$ then $Rep = (Rep - R_i) \cup R_k$
- If R_k is a flexible rule and $\nexists R_i \in Rep$ such as R_k and R_i are similar rules then $Rep = Rep \cup R_k$

IV. ROC CURVE

Receiver Operating Characteristic (ROC) is a technique that graphs the performance of a classifier [18]. It has been

used to evaluate the performance of diagnosis tests [19], [20], [21],[22]. We have selected ROC as a performance measure because: 1) ROC is able to deal with imbalanced data sets, 2) it is capable to measure the performance of a classifier that rely on thresholds, 3) it enables researches to tune the behavior of the classifier according to the best tradeoff between miss-classification and false alarms costs.

Before to introduce more technical details about ROC, let's introduce a brief explanation about the confusion matrix. A confusion matrix displays the data about actual and predicted classifications done by a classifier [23]. Given an instance and a classifier there are four possible results: The instance is positive and it is classified as positive *true positive*. The instance is positive and it is counted as negative *false positive*. The instance is negative and it is classified as positive *false negative*. The instance is negative and it is predicted as negative *true negative* (TN). Table II shows a confusion matrix for two classes.

TABLE II
CONFUSION MATRIX TO CLASSIFY TWO CLASSES

	Actual Positive	Actual Negative
Positive Prediction	<i>True Positive</i> (TP)	<i>False Positive</i> (FP)
Negative Prediction	<i>False Negative</i> (FN)	<i>True Negative</i> (TN)
	Total Positive	Total Negative

The ROC graph is constructed by plotting the *true positive rate*² (recall) on the Y-axis and the *false positive rate*³ on the X-axis [24],[18]. Figure 2 shows the ROC space. The classifiers whose performance is plotted in the left hand side in the ROC space close to the X-axis, are denominated *conservative* because they make a positive classification just when they have strong evidences. On the other hand classifiers on the upper right hand side of a ROC graph are called *liberal* because they make positive classification with unsubstantial evidence. Finally the diagonal line between (0,0) and (1,1) describes the performance of a random classifier.

The result of a discrete classifier is a confusion matrix which represents a single point in the ROC graph. Some classifiers manage a threshold to tune its precision (this is the case of RM), every threshold value produce a plot in the ROC space, thus the classifier move from the liberal to the conservative part of the ROC area.

The Area Under the ROC Curve (AUC) has been used to indicate the quality of the classifier [20]. It was showed theoretically and empirically that AUC is a better measure than accuracy [25]. When AUC =1 it means the the classifier

²it is the proportion of positive cases that were correctly identified, it is determined by the formula: $true\ positive\ rate = TP / (TP + FN)$

³False positive rate is the proportion of miss-classified positive cases, its equation is: $False\ positive\ rate = FN / (TP + FN)$

is perfectly accurate, When AUC is close to .5 it represents a random classifier performance.

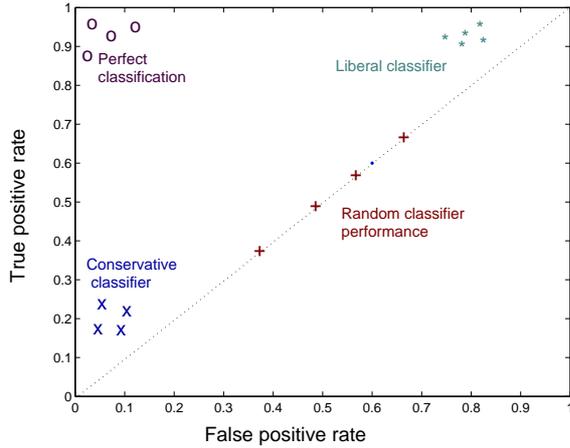


Fig. 2. ROC space

V. EXPERIMENTAL SECTION

The aims of this paper are 1) to show that RM is able to produce a range of solutions capable to suit the investor requirements and 2) to analyze the influence of the evolutionary process in the RM performance. Three series of experiments were performed, RM was tested using two artificial data sets whose solutions have different level of complexity. Additionally RM was tested in a data set from the London stock market.

This section is organized as follows: section V-A describes the GP that was used to generate the candidate solutions (decision trees). Section V-B explains the procedure to generate the artificial data sets while section V-C describes the data set from the London stock market. Finally, sections V-D and V-E describe the experimental procedure and the results respectively.

A. Genetic programming description

The GP used in this work to generate the sets of decision trees was inspired by EDDIE. Table III summarized the GP parameters, notice that the fitness function of the GP is the geometric mean of the product of the precision and recall, which is a common metric to evaluate the performance of classifiers in imbalanced environments.

Let P_k be a population of decision trees that has evolved for k generations. It means that the population P_0 is a set of random decision trees and P_{10} represents a population of decision trees that have been evolved for 10 generations. This symbology is used in section V-D

B. Artificial data sets creation

In order to control the complexity of the data sets in the experiment, two data sets were created artificially as follows:

- 1) A set of 1,200 records was created, every record holds eight independent variables with real values. Every variable was randomly generated in a range of [0-1].

TABLE III
SUMMARY OF PARAMETERS

Parameter	Value
Population size	1,000
Initialization method	Growth
Generations	100
Crossover Rate	0.8
Mutation Rate	0.05
Selection	Tournament (size 2)
Control bloat growing	50% of trees whose largest branch exceed 6 nodes are penalized with 20% in its fitness
Fitness Function	$\sqrt{\text{Recall} \cdot \text{Precision}}$

- 2) Every record was labeled with a class (*positive* or *negative*). The records that meet the requirements in at least one of the rules in S_1 (see figure 3) is labeled as positive. Otherwise the record is classified as negative.
- 3) The data is split in two data sets (training and testing) holding the same number of records (600).
- 4) The second artificial data set was created repeating the steps 1-3, but using S_2 instead of S_1 .

The data sets *Artificial S_1* and *Artificial S_2* have the following characteristics:

- The sets of rules S_1 and S_2 were designed to create imbalanced data sets. The training data set created by S_1 holds 28 (4.6%) positive cases and the testing data set 29 (4.8%) positive cases. The training data set generated by S_2 holds 18 (3.0%) positive cases and the testing data set 17 (2.8%) positive cases.
- For sake of simplicity we measure the complexity of the data set for the number of conditions involved in the solution. When the number of conditions in the solution increases the solution is more complex. It means that the solution of the data sets created by S_1 is easier than the solution in S_2 .
- Since the independent variables were created randomly, it means that variables are not correlated.

C. Barclays data set description

The data set of Barclays stock is composed by the prices from March, 1998 to January, 2005. The attributes of each record are composed by indicators derived from financial technical analysis. Technical analysis has been used in financial markets to analyze the stock price behavior, this is mainly based on historical prices and volume trends [12]. The indicators were calculated on the basis of the daily *closing price*⁴, volume and some financial indices as the FTSE⁵.

⁴The settled price at which a traded instrument is last traded at on a particular trading day.

⁵An index of 100 large capitalization companies stock on the London Stock Exchange, also known as "Footsie".

Fig. 3. Set of rules used to created the artificial data sets

$S_1 = \{$	$R_1 = var_1 > 0.99$
	$R_2 = var_2 < 0.009$
	$R_3 = var_5 < 0.898 \text{ and } var_5 > 0.89$
	$R_4 = var_5 < 0.01$
	$R_5 = var_6 > 0.88 \text{ and } var_6 < 0.89 \}$
$S_2 = \{$	$R_1 = var_1 > 0.5 \text{ and } var_1 < 0.58 \text{ and } var_2 > 0.5 \text{ and } var_3 < 0.7 \text{ and } var_4 < var_3$
	$R_2 = var_3 < 0.45 \text{ and } var_3 > var_2 \text{ and } var_3 > var_4 \text{ and } var_3 > var_5 \text{ and } var_3 > var_6$
	$R_3 = var_8 < 0.898 \text{ and } var_8 > 0.86 \text{ and } var_5 > 0.065 \text{ and } var_5 < 0.35 \text{ and } var_3 > var_7$
	$R_4 = var_1 > 0.5 \text{ and } var_1 < 0.58 \text{ and } var_2 > 0.5 \text{ and } var_3 < 0.7 \text{ and } var_4 < var_3 \text{ and } var_4 < var_6$
	$R_5 = var_6 > 0.56 \text{ and } var_7 > var_6 \text{ and } var_8 > var_6 \text{ and } var_8 < var_1$
	$R_6 = var_1 > var_7 \text{ and } var_1 > var_6 \text{ and } var_6 < 0.23 \text{ and } var_5 < var_6 \}$

TABLE IV
FINANCIAL INDICATORS USED IN THE EXPERIMENT

Indicator name	Short period (Days)	Long period (Days)
Price moving average	12	50
Price Trading breaking rule	5	50
Filter rule	5	63
Price volatility	12	50
Volume moving average	10	60
Momentum	10	60
Momentum 10 days moving average	10	-
Momentum 60 days moving average	60	-
Generalized Momentum indicator	10	60
FOOTSIE moving average	12	50
LIBOR: 3 months moving average	12	50

D. Experiment description

RM was tested using the data sets Artificial S_1 , Artificial S_2 and Barclays. For every data set the following experiments were performed.

1) Experiment 1

RM collects rules from P_0 , a random population of decision trees. The individuals in P_0 were created using the DG grammar. It is expected that the performance of RM will be low, because the decision trees are random. We decided to perform this experiment to test the capacity of RM to select patterns. The results produced by P_0 have the advantage of being created in a really fast way, because there is not evolution at all.

2) Experiment 2

RM gathers rules from P_{10} a population, that has been evolved during 10 generations. It is expected that the results in this experiment outperform the results in experiment 1.

3) Experiment 3

RM collects and accumulated rules from $P_{10}, P_{20}, \dots, P_{100}$ which means that every ten generations, RM collected and accumulated rules. Obviously it is expected that this experiment outperform the results of the previous experiments.

Notice that all the population in this research is composed by 1,000 individuals. We search in the complete population because in a previous work [9] it was demonstrated that it is possible to gather predictive rules even from low fitness individuals. In every series of experiments RM was tested using different precision thresholds $PT = 10\%, 20\%, \dots, 100\%$ to plot the ROC curve. The results given in the next section described average results of 20 runs.

E. Experiment results

Figures 4, 5 and 6 display the ROC curve for Barclays, Artificial S_1 and Artificial S_2 respectively. In order to identify precisely the points plotted by RM, this data is presented in tables VI, VII and V.

A standard GP produces a single prediction for every data set, in contrast RM allows the investor to tune the prediction according to his/her risk guidelines. If the requirement of the investor is to detect as most positive cases as possible, the PT has to decrease in order to move to the liberal part in the ROC space. In contrast if the preference of the user is to decrease the risk, PT has to increase in order to move to the conservative part in the graph.

Barclays data set

The result of the standard GP is: recall =14%, precision=5% and accuracy= 89%. This result is plotted in (0.09, 0.14) in ROC graph. Figure 4 displays the ROC curves plotted by RM in the following experiments:

- *Experiment 1* Using P_0 the AUC = .69, as can be observed from figure 4 the majority of the points are clustered in the conservative part of the ROC curve because these did not classify any positive case. However RM was able to generate an interesting choice for the investor, this is: recall =38%, precision=9% and accuracy= 87%
- *Experiment 2* Using P_{10} the performance of RM increased considerably, the AUC increased from 0.69 to 0.74. In this experiment RM offers two valuable choices when $PT=30\%$ and $PT=20\%$. However one of the choices is in the conservative side and the other in the liberal side of the ROC curve as table X shows.
- *Experiment 3* Using $P_{10}, P_{20}, \dots, P_{100}$ the AUC increased from 0.74 to 0.76. In addition the predictions have been distributed along the ROC curve. This allows the investor to choose in a wide range of options the most suitable prediction to her/his requirements. So it is possible to detect from 26% to 81% of the positive cases with an accuracy bigger than 65% (see table V).

Data set Artificial S_1

First, let's introduce the results of the standard GP: recall

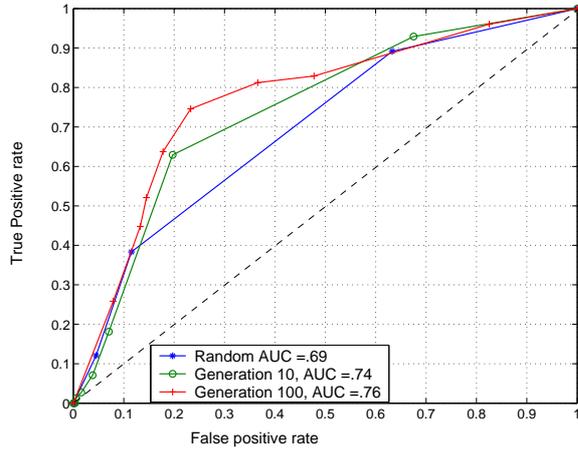


Fig. 4. Barclays ROC curve

=34%, precision=95% and accuracy= 97% and it is plotted in the position (0.001, 0.34) in the ROC graph. Figure 5 displays the ROC curves plotted by RM in the following experiments:

- *Experiment 1* Using P_0 the majority of the classifications are concentrated in the conservative part of the ROC space, it limits the options of the user. However, RM was able to produce two interesting predictions 1) the detection of 35% of the positive cases with a precision of 57% and accuracy of 96% and 2) the detection of 45% of the positive cases with a precision of 33% and accuracy of 93%. Both of them are competitive options with the solution proposed by the standard GP. In this case the advantage of RM is that its solution was produced very fast because RM used a random population.
- *Experiment 2* Using P_{10} the performance of RM increases considerably, and the AUC increases from 0.78 to 0.83. In this experiment RM offers good choices to the investor. He/she can detect from 48% to 60% of the positive cases with an accuracy between 73% and 96% (see table IX).
- *Experiment 3* Using $P_{10}, P_{20}, \dots, P_{100}$ the AUC increased slightly from 0.83 to 0.84. In addition the results are very similar to those in experiment 2. It indicates that the majority of the predictive rules were formed in the 10 first generations or exploration phase. It means that result in experiment 3 did not pay the computational effort of the extra 90 generations.

Data set Artificial S_2

This section tests RM using the data set Artificial S_2 . As in the previous section the results are compared against a standard GP. Let's introduce the results of the standard GP: recall =0.64%, precision=0.36% and accuracy= 0.96% and its point in the ROC graph is (0.03, 0.64). Figure 6 displays the ROC curves plotted by RM in the following experiments:

- *Experiment 1* Using P_0 the majority of the predictions

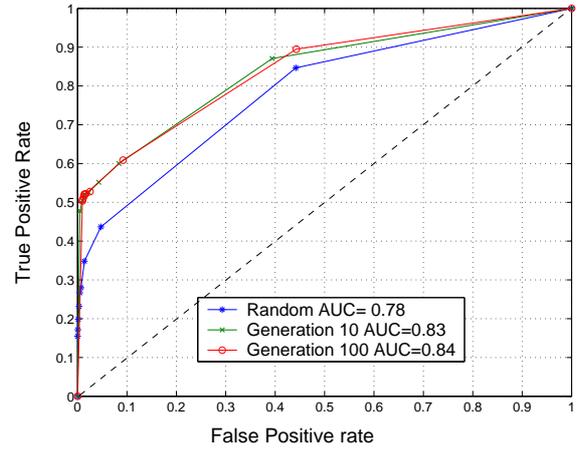


Fig. 5. Artificial S_1 ROC curve

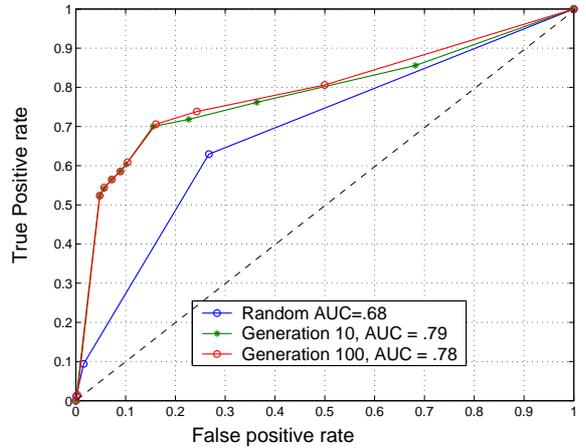


Fig. 6. Artificial S_2 ROC curve

are clustered in the conservative part of the ROC curve. This experiment did not generated any interesting choice for the investor.

- *Experiment 2* Using P_{10} the performance of RM increases considerably, the AUC increased from 0.68 to 0.79. In this experiment RM offers many good options to the investor. He/she can choose detecting from the 52% to 74% of the positive cases with an accuracy bigger than 89% as table X shows.
- *Experiment 3* The performance of RM varied slightly, the AUC decreased from 0.79 to 0.78. However it is possible to choose to detection from the 52% to 76% of the positive cases with an accuracy bigger than 90% as table X shows. This results slightly overcome the results in experiment 1. However this improvement has to be paid with computational effort. However the investor is the best person to determine is that improvement paid the computational effort.

VI. CONCLUSIONS

It has been shown that RM offers a range of solutions to suit the risk guidelines of the investors. Thus the user can

TABLE V

BARCLAYS RESULTS X-AXIS IS THE FALSE POSITIVE RATE AND Y-AXIS IS THE TRUE POSITIVE RATE

PT	Random decision trees (P_0)		Generation 10 (P_{10})		Generation 100 ($P_{10} \dots P_{100}$)	
	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
100	0.00	0.00	0.00	0.00	0.08	0.26
90	0.00	0.00	0.00	0.00	0.08	0.26
80	0.00	0.00	0.00	0.00	0.13	0.45
70	0.00	0.00	0.01	0.01	0.14	0.52
60	0.00	0.00	0.02	0.03	0.18	0.64
50	0.00	0.00	0.04	0.07	0.23	0.75
40	0.05	0.12	0.07	0.18	0.37	0.81
30	0.12	0.38	0.20	0.63	0.48	0.83
20	0.63	0.89	0.68	0.93	0.83	0.96
10	1.00	1.00	1.00	1.00	1.00	1.00

TABLE VI

ARTIFICIAL S_1 RESULTS, X-AXIS IS THE FALSE POSITIVE RATE AND Y-AXIS IS THE TRUE POSITIVE RATE

PT	Random decision trees (P_0)		Generation 10 (P_{10})		Generation 100 ($P_{10} \dots P_{100}$)	
	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
100	0.00	0.00	0.00	0.00	0.00	0.00
90	0.00	0.16	0.00	0.48	0.01	0.50
80	0.00	0.18	0.01	0.51	0.01	0.51
70	0.00	0.21	0.01	0.51	0.01	0.51
60	0.00	0.24	0.01	0.52	0.01	0.52
50	0.00	0.28	0.01	0.52	0.01	0.52
40	0.01	0.29	0.02	0.53	0.03	0.53
30	0.01	0.36	0.04	0.55	0.02	0.52
20	0.05	0.45	0.08	0.60	0.09	0.61
10	0.44	0.88	0.39	0.87	0.44	0.89
0	1.00	1.00	1.00	1.00	1.00	1.00

choose the best balance between miss-classification and false alarms according to his/her requirements.

RM is able to extract predictive rules even from earliest stages of the evolutionary process. However to create a wider range of solutions, it is advisable to evolve the GP at least after the exploration phase. Specially when the solution of the problem is complex.

REFERENCES

- [1] E. P. Tsang, J. Li, and J. Butler, "Eddie beats the bookies," in *International Journal of Software, Practice and Experience*, ser. 10, vol. 28. Wiley, August 1998, pp. 1033–1043.
- [2] J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, Massachusetts: The MIT Press, 1992.
- [3] E. P. Tsang, S. Markose, and H. Er, "Chance discovery in stock index option and future arbitrage," in *New Mathematics and Natural Computation, World Scientific*, ser. 3, vol. 1, 2005, pp. 435–447.
- [4] N. Japkowicz, "The class imbalance problem: Significance and strategies," in *Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000)*, vol. 1, 2000, pp. 111–117. [Online]. Available: citeseer.ist.psu.edu/japkowicz00class.html
- [5] G. M. Weiss, "Mining with rarity: A unifying framework," in *Special issue on learning from imbalanced datasets*, vol. 6, 2004, pp. 7–19.

TABLE VII

ARTIFICIAL S_2 RESULTS, X-AXIS IS THE FALSE POSITIVE RATE AND Y-AXIS IS THE TRUE POSITIVE RATE

PT	Random decision trees (P_0)		Generation 10 (P_{10})		Generation 100 ($P_{10} \dots P_{100}$)	
	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
100	0.00	0.00	0.00	0.00	0.00	0.00
90	0.00	0.00	0.00	0.01	0.05	0.52
80	0.00	0.00	0.05	0.52	0.06	0.54
70	0.00	0.00	0.06	0.54	0.07	0.56
60	0.00	0.00	0.07	0.56	0.09	0.59
50	0.00	0.00	0.09	0.59	0.10	0.60
40	0.00	0.00	0.10	0.61	0.15	0.70
30	0.00	0.01	0.16	0.71	0.23	0.72
20	0.02	0.09	0.24	0.74	0.36	0.76
10	0.27	0.63	0.50	0.81	0.68	0.86
0	1.00	1.00	1.00	1.00	1.00	1.00

TABLE VIII

BARCLAYS RECALL (1), PRECISION (2) AND ACCURACY (3)

PT	Random decision trees (P_0)			Generation 10 (P_{10})			Generation 100 ($P_{10} \dots P_{100}$)		
	%	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)
100	0	–	.97	0	–	.97	0	–	.97
90	0	–	.96	0	–	.97	.26	.09	.90
80	0	–	.96	0	–	.97	.26	.09	.90
70	0	–	.96	0	–	.96	.45	.09	.86
60	0	–	.96	.01	.06	.96	.52	.10	.85
50	0	–	.96	.03	.05	.95	.64	.10	.83
40	0	–	.96	.07	.05	.93	.75	.09	.78
30	.12	.07	.93	.18	.07	.91	.81	.06	.65
20	.38	.09	.87	.63	.09	.81	.83	.05	.54
10	.89	.04	.38	.93	.04	.35	.96	.03	.20

- [6] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, 2004.
- [7] K. McCarthy, B. Zabar, and G. Weiss, "Does cost-sensitive learning beat sampling for classifying rare classes?" in *UBDM '05: Proceedings of the 1st international workshop on Utility-based data mining*. New York, NY, USA: ACM Press, 2005, pp. 69–77.
- [8] A. L. Garcia-Almanza and E. P. Tsang, "The repository method for chance discovery in financial forecasting," in *To appear in KES2006 10th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, Springer-Verlag, Ed., 2006.
- [9] —, "Forecasting stock prices using genetic programming and chance discovery," in *12th International Conference On Computing In Economics And Finance*, 2006.
- [10] J. Li, *A genetic programming based tool for financial forecasting*. Colchester CO4 3SQ, UK: PhD Thesis, University of Essex, 2001.
- [11] E. P. Tsang, P. Yung, and J. Li, "Eddie-automation, a decision support tool for financial forecasting," in *Journal of Decision Support Systems, Special Issue on Data Mining for Financial Decision Making*, ser. 4, vol. 37, 2004.
- [12] W. F. Sharpe, G. J. Alexander, and J. V. Bailey, *Investments*. Upper Saddle River, New Jersey 07458: Prentice-Hall International, Inc, 1995.
- [13] N. Chomsky, *Aspects of the theory of syntax*. Cambridge M.I.T. Press, 1965.
- [14] P. Angeline, "Genetic Programming and Emergent Intelligence," in *Advances in Genetic Programming*, K. E. Kinnear, Jr., Ed. MIT Press, 1994, ch. 4, pp. 75–98.
- [15] P. Nordin, F. Francone, and W. Banzhaf, "Explicitly Defined Introns and Destructive Crossover in Genetic Programming," in *Proceedings of*

TABLE IX
ARTIFICIAL S_1 RECALL (1), PRECISION (2) AND ACCURACY (3)

PT	Random decision trees (P_0)			Generation 10 (P_{10})			Generation 100 ($P_{10} \dots P_{100}$)		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
100	0	–	.95	0	–	.97	0	–	.97
90	.16	.99	.96	.48	.83	.96	.50	.72	.92
80	.18	.94	.96	.51	.72	.92	.51	.72	.91
70	.21	.88	.96	.51	.68	.91	.51	.68	.90
60	.24	.80	.96	.52	.67	.90	.52	.66	.88
50	.28	.75	.96	.52	.64	.88	.52	.64	.87
40	.29	.68	.96	.53	.54	.87	.53	.52	.82
30	.36	.57	.96	.55	.39	.81	.52	.63	.75
20	.45	.33	.93	.60	.27	.73	.61	.25	.61
10	.88	.09	.57	.87	.10	.48	.89	.09	.30

TABLE X
ARTIFICIAL S_2 RECALL (1), PRECISION (2) AND ACCURACY (3)

PT	Random decision trees (P_0)			Generation 10 (P_{10})			Generation 100 ($P_{10} \dots P_{100}$)		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
100	0	–	.95	0	–	.95	0	–	.95
90	0	–	.95	.01	.13	.96	.52	.24	.97
80	0	–	.95	.52	.24	.96	.54	.22	.96
70	0	–	.95	.54	.22	.96	.56	.19	.96
60	0	–	.95	.56	.19	.96	.59	.16	.96
50	0	–	.95	.59	.16	.96	.60	.15	.96
40	0	–	.95	.61	.15	.95	.70	.12	.95
30	.01	.24	.95	.71	.11	.96	.72	.09	.93
20	.09	.15	.92	.74	.08	.89	.76	.06	.90
10	.63	.06	.57	.81	.04	.57	.86	.04	.61

the Workshop on Genetic Programming: From Theory to Real-World Applications, J. P. Rosca, Ed., Tahoe City, California, USA, 9 July 1995, pp. 6–22.

- [16] T. Soule and J. A. Foster, “Code size and depth flows in genetic programming,” in *Proceeding of the Second Annual Conference*, J. R. Koza, K. Deb, M. Dorigo, D. B. Fogel, M. Garzon, H. Iba, and R. R. Riolo, Eds. Morgan Kaufmann, 1997, pp. 313–320.
- [17] W. B. Langdon, “Quadratic bloat in genetic programming,” in *Proceedings of the Genetic and evolutionary Computation Conference (GECCO-2000)*, 2000, pp. 451–458.
- [18] T. Fawcett, “Roc graphs: Notes and practical considerations for researchers,” in *Introductory paper*, 2004.
- [19] S. H. Park, J. M. Goo, and C.-H. Jo, “Receiver operating characteristic (roc) curve: Practical review for radiologist,” *Korean Journal of Radiology*, vol. 5, no. 1, pp. 11–18, 2004.
- [20] G. Vanagas, “Receiver operating characteristic curves and comparison of cardiac surgery risk stratification systems,” *Interact CardioVasc Thorac Surg*, vol. 3, no. 2, pp. 319–322, 2004.
- [21] M. Greiner, D. Pfeiffer, and R. Smith, “Principles and practical application of receiver-operating characteristic analysis for diagnostic tests,” *Prevent Veterinary Med*, vol. 45, pp. 23–41, 2000.
- [22] H. Liu and T. Wu, “Estimating the area under a receiver operating characteristic curve for repeated measures design,” *Journal of Statistical Software*, vol. 8, no. 12, pp. 1–18, 2003. [Online]. Available: <http://www.jstatsoft.org/v08/i12>; <http://www.jstatsoft.org/v08/i12/roc.pdf>
- [23] R. Kohavi and F. Provost, “Glossary of terms,” in *Edited for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process*, vol. 30, February 1998.
- [24] J. A. Hanley and B. J. McNeil, “The meaning and use of the area under a receiver operating characteristic roc curve,” in *Radiology*, vol. 143. W. Madison, 1998, pp. 29–36. [Online]. Available: citeseer.ist.psu.edu/provost97case.html
- [25] J. Huang and C. X. Ling, “Using auc and accuracy in evaluating