Computer Science and Information Systems 23(1):513–534 https://doi.org/10.2298/CSIS250226013B

Using Genetic Programming as a Feature Selector and Classifier to Implement Bankruptcy Prediction Models*

Ángel Beade[1], José Santos[2], and Manuel Rodríguez[1]

[1] Business Department, University of A Coruña,
Campus de Elviña, s/n 15071 A Coruña, Spain
{a.beade, manuel.rodriguez.lopez}@udc.es
[2] CITIC (Centre for Information and Communications Technology Research), Department of Computer Science and Information Technologies, University of A Coruña, Campus de Elviña, s/n 15071 A Coruña, Spain
jose.santos@udc.es

Abstract. Genetic Programming (GP) was used as a feature selector and classifier to implement bankruptcy prediction models for medium-sized companies. Two sets of input variables were used for the prediction models: one using a large number of exclusively financial variables and the other incorporating variables from the economic environment, which allows analyzing the capability of the latter to improve performance. Two strategies were defined for GP as a feature selector, based on the statistical relevance of the selected features in the GP process, with a novel proposal based on a progressive reduction of the set of selected variables and with the aim of minimizing the risk of eliminating relevant features. An analysis is performed of the improvement obtained with feature selection with both GP-based methods in comparison with the use of complete sets of variables and using GP as a classifier. With the selected variables, we also compared GP as a classifier with respect to other standard classifiers, using automatic parameter adjustment with AutoWeka for these classifiers. The best results are obtained with the synergy of using GP as a feature selector and as a classifier, with the advantage of the direct interpretability that GP provides in the application.

Keywords: Genetic programming, feature selection, bankruptcy prediction models.

1. Introduction

The prediction of insolvency or bankruptcy of a company was one of the traditional research topics in financial economics. In the last two decades, Machine Learning (ML) models have emerged as a better alternative to traditional prediction models based on statistical models [1, 50].

The bankruptcy prediction problem is treated, with the information available in the financial statements of companies, as a classification problem. In the ML field, there are a large number of classification techniques applied to bankruptcy prediction, including ensembles of classifiers [32, 35, 46] and deep learning methods [29, 30, 47]. However, our previous work [6–8] has focused on the use of Genetic Programming (GP) [25, 39].

*This paper is an extension of INISTA 2024 Conference.

The main reason is that GP, by evolving "programs" (usually using a tree-based representation), allows for straightforward and easier interpretability (at least readability) compared to classification techniques that are considered black boxes. This is very important in this field from the point of view of the end user of the prediction models, as well as of the economic and political institutions that are beginning to demand interpretability and explainability of the prediction results [14–16]. Moreover, GP does not impose restrictions on the explanatory variables of the prediction models (contrary to classical statistical methods). Finally, the complexity of the evolved programs can be easily regulated. For example, the maximum depth of the evolved classification trees, the maximum number of nodes in the evolved trees and the set of functions that can be used by the tree nodes are decisions aimed at facilitating interpretability while maintaining high prediction performance. Moreover, as shown in previous work [6], GP can obtain bankruptcy prediction models (BPMs) with high performance and stable predictive power over time, even considering a long post-training period that includes a changing economic environment (with a sharp recession and a strong recovery) [6].

Furthermore, the selection of explanatory variables for the BPMs is, after modeling techniques, a major issue in business failure research [48] and studies show that BPMs can be more effective with proper variable selection [46]. But variable selection is not an issue that can be addressed in isolation, as has been repeatedly pointed out since early work on business failure prediction, as variable selection and the classification method are intrinsically linked [1, 17, 51], and there is no consensus on which variables are relevant or how to perform variable selection [1, 3, 4, 19, 21, 26, 32]. One of the common ways to select independent variables in a model is to start from an initial set of available variables of high dimensionality and reduce this dimensionality by means of Feature Selection (FS) methods, FS that is also expected to improve model performance.

In our previous study [7], we analyzed the suitability of a method based on GP to address dimensionality reduction through feature selection. The FS method is based on the relative frequency of occurrence, in GP evolved solutions, of the input variables and their statistical significance. In this aforementioned study [7], FS by means of GP was compared with other widely used FS methods. The comparison was also performed using a set of 15 classification methods, also including GP as a classifier.

Previous works [34, 42] have also considered the frequency of occurrence of variables in the GP process as a basis for performing FS. Nevertheless, Neshatian and Zhang [34] only considered the best final evolved solutions, which is a very small subset where the optimized trees can be very similar. But, as Liu et al. [28] point out, even considering the best-evolved solutions with GP, "the features generated in the output individual may still contain redundant features". Liu et al. [28] addressed this problem by considering the best individuals generated during GP evolution instead of a single best individual, also in combination with a feature ranking. However, the bloat problem may appear in many individuals in the population, since a part of the tree that is irrelevant to the fitness of a solution may be maintained in several solutions in the population.

To address the problem and ensure a correct FS process, decreasing the possibility of selection of irrelevant variables due to the GP bloat problem, our previous proposal [7] considers a large number of independent GP runs to compute the frequency of variable occurrence in the GP process (logically with the disadvantage of increased

computational time). Moreover, unlike the previous works discussed [28, 34, 42], the FS process defined in [7] also considered the statistical relevance of variable occurrence, with the aim of ensuring a correct selection of the most significant variables. This FS process based on GP presented better results considering efficiency (i.e., classification improves when using FS compared to using all features) and also performance (i.e., for a particular classifier, the classification results obtained with the different FS methods are compared) with respect to other feature selection methods [7].

This work proposes an alternative to the one presented in [7], with a new proposal that aims to minimize the risk of information loss due to discarded variables. This second proposed methodology is a stepwise process that addresses a progressive reduction in the size of the set of explanatory variables by eliminating, at each stage, those variables clearly showing irrelevance, thus minimizing the risk of possible elimination of relevant variables. Since the work in [7] performed a detailed comparison of the GP-based FS with other widely used FS methods, the present work focuses on the comparison of the new alternative with the previous one.

In addition, to make the conclusions of the comparison more robust, two different initial sets of variables were used as possible explanatory variables of the GP-evolved BPMs (different BPMs with different prediction time horizons). The first set is defined with only financial variables of the company and the second set is an extended set incorporating, among others, variables of the economic environment. This use of two sets of variables also allows us to situate the results of GP-evolved BPMs in relation to the existing debate in the field between those authors who use and defend the sufficiency of using only explanatory variables constructed exclusively from accounting data [10, 12, 45], and those authors who incorporate non-financial variables [3, 4, 20, 40]. In this regard, our previous work presented at the INISTA 2024 conference [8], analyzed the comparison of performance prediction with both sets of explanatory variables. Thus, the present work extends that comparison when different feature selection methods are used.

Furthermore, apart from the selection of relevant explanatory variables provided by GP-based FS methods, to evaluate GP as a classification method, in this work a comprehensive comparison with other classification methods has been carried out. For this purpose, the Weka package called AutoWeka [24] was used, which provides automatic adjustment of the parameters of the other ML classifiers.

Therefore, the main objective of this paper is to analyze the GP-based FS methods, with emphasis on the novel proposal oriented to minimize the risk of eliminating possible relevant variables in its iterative FS procedure. For the analysis of the FS methods, different BPMs with different prediction horizons are evolved, also considering the two sets of possible explanatory variables mentioned above, where one includes non-financial variables. In addition, an analysis is performed comparing GP as a classifier to implement BPMs with respect to other ML classification methods, where the AutoWeka environment automates the parameterization of the classifiers, an analysis oriented to emphasize the link between the selection of variables and the classification methods.

The remainder of this article is structured as follows. Section 2 details the different methods used for the implementation of the BPMs, such as the explanation of the input variables in the two sets of variables considered, the dataset used to evolve the GP-based BPMs, as well as the FS methods considered, with emphasis on the new proposal of this work. Section 3 details the results in the comparison with the two FS strategies

considered, as well as the performance results when using both sets of possible input variables in the BPMs. Section 3 also details the performance comparison between GP as a classifier with respect to other ML classifiers. Finally, Section 4 summarizes the main conclusions of the paper and possible avenues for further research.

2. Methods

2.1. Input Variables for the Prediction Models

Two sets of input variables were used as inputs for the prediction models. The first set includes only financial variables of the companies (set A). This set includes 97 explanatory variables that refer to a wide range of aspects that are considered, a priori, relevant to business failure. Table 1 shows the categories (standard categorization in financial analysis) of these variables.

The explanatory variables of the BPMs included in set A were preferentially chosen based on two criteria: i) relevance shown in the bibliography and their presence in prediction models contrasted in previous work. Most of the variables correspond to financial ratios (understood as the quotient of two financial variables), as they are the traditional basis for financial analysis. ii) other variables collected from the annual accounts of the companies and which refer to different aspects that are rarely used (e.g., variations in ratios or variations in magnitudes) or novel in this field (e.g., degree of the balance sheet decomposition and variables referring to fraud and productivity)

It is important to emphasize the large number of explanatory variables considered, which is much higher than the number usually used in the literature dealing with insolvency prediction [3, 44]. According to systematic reviews, the trend in contemporary models is to use between 15 and 30 variables in the initial stage, especially when employing advanced machine learning and big data techniques. However, automatic variable selection processes tend to reduce the final model to 5-8 predominant ratios. This pattern appears in international contexts (US, Europe, India, etc.) and in various business sectors [1–3, 11, 27, 41].

The second set extends set A with four qualitative non-financial variables with company data and 11 quantitative variables with information on the economic environment (set B). The 4 qualitative variables correspond to the age of the company (3 categories considering the age in ranges 0-3 years, 4-9 years and more than 9 years), the auditor's qualification in the financial year (negative, not available or positive) and two variables related to financial stress (true/false/not available). The 11 variables that attempt to reflect the economic environment are defined considering aspects (in each financial year and referring to Spain) such as the gross operating surplus and gross mixed income, the number of bankrupt companies, the gross domestic product, the market prices, the employment level and interest rates. Note that, with the objective (not addressed here) of counteracting the prediction of failure and thus minimizing the risk of bankruptcy, the firm can act on all the variables in set A, while it cannot act on the variables of the economic environment and on most of the firm's qualitative variables, which are incorporated in set B.

Table 1. Sets A and B of input variables

Category	Number	Set
Changes in financial ratios	2	A, B
Contribution	2	A, B
Decomposition degree	3	A, B
Efficiency	11	A, B
Financial structure	14	A, B
Fraud	11	A, B
Growth	1	A, B
Interest expenses	5	A, B
Liquidity and solvency	18	A, B
Productivity	4	A, B
Profitability	12	A, B
Size	4	A, B
Turnover	7	A, B
Variations in magnitudes	3	A, B
Qualitative variables of the company	4	B
Economic environment variables	11	B

In the case of the quantitative variables in set A, since some of the company's observations (financial information of the company in a financial year) present an outlier in one of the numerous explanatory variables considered, a limiting of extreme values has been carried out. To this end, the variables of a company in a financial year with values above the 97.5% percentile or below the 2.5% percentile, corresponding to the distribution of a specific variable, are replaced with the respective reference percentile value. The objective is to eliminate values that are clearly outside the usual range of values and, thus, the difficulties that they entail for learning. This delimitation of values is common in ML [36, 38, 43] and was carried out in order not to eliminate information from companies, with the aim of including them in the training and test sets (next subsection).

In addition, as a last step, with the values of the variables bounded (of set A), a transformation is performed for each variable (according to a logistic distribution and considering the values in the training period detailed below), with the aim of improving the generalization capacity of the prediction models.

2.2. Dataset and Prediction Models

This paper considers the prediction of corporate insolvency in medium-sized Spanish companies. A population of 11,158 firms is considered (10,091 classified as non-failed, 1,067 classified as failed), with their accounting information from 2005 to 2019.

Both the accounting data (balance sheet and income statement) and other relevant data (date of incorporation, audit, etc.) have been obtained from the Iberian Balance Sheet Analysis System (SABI) database [52]. The concept of bankruptcy followed in this work refers to the legal declaration of suspension of payments, which is the definition most commonly considered in the field [27, 50].

A large set of different BMPs were considered to draw solid conclusions with the GP processes: first, 9 BPMs labeled M1, M2, … M9, named as "annual BPMs" as these predict the failure of a company 1, 2, … and 9 years in advance. Secondly, five "multiperiod BPMs" [9], which predict failure over a range of future years (instead of failure at a specific future point in time). For example, BPM M1-3 predicts the failure of a company in the interval between 1 and 3 years in the future.

The use of multi-period BPMs, although less common in the field of financial economics, is an interesting approach since, for example, a lender is surely more interested in estimating bankruptcy during the loan period (rather than at a specific point in the future), as discussed in [9]. Consequently, we have also incorporated this alternative, in addition to the classic annual BPMS, into the analysis. Therefore, this set of annual/multiperiod BPMs covers a wide range of prediction years corresponding to short-, medium- and long-term prediction horizons.

The training and test sets of all these prediction models include observations of companies in the period 2005-2007. An "observation" is defined as the financial data of a company in a particular financial year. Thus, for example with the observations of 2005, M9 predicts failure in 2014 and with the observations of 2007 it predicts failure in 2016. Table 2 shows the number of observations in the training and test sets for the different prediction models. Note that the data are highly imbalanced in this application (between failed and healthy companies). However, balanced training sets are used (Table 2), where the available number of failed observations is randomly divided between the training and test set observations. In the case of non-failed observations, those in the training set are selected from the much larger total number of non-failed observations, while the remainder are inserted into the test set. The use of balanced training sets avoid biased predictions in the machine learning process.

Various authors provide the sizes of the training sets used in different BPM studies for different techniques and different time horizons [1, 3, 5, 20, 22]. In particular, the work of du Jardin [20] analyzes studies referring to annual models with a time horizon of up to 5 years, work in which the number of companies used in the training set and the wide variety of sizes in that set can be observed. Therefore, although there is no standard for determining the optimal size of a BPM training set, the range used in this study for annual models fits perfectly, for example, with the sizes in the studies reviewed by du Jardin and Séverin [22]. They also fit with the size of class 1 (failures) in the study by Altman et al. [3], which ranges from 38 to 180 failed companies.

The three consecutive years (2005-2007) are considered for the training and test sets, with the number of observations specified in Table 2. This allows for a long post-learning period to check, for example, the stability of predictions over that period (as performed in [6] on the performance of genetic programming-evolved BPMs over that period, which is not the aim here). It should be noted that there is no definitive consensus on the number of exercises to be used in training a model, as performance depends both on the quality of the model fit and the dataset used. In machine learning studies, it is common to use between 3 and 5 years of financial data for the training period [13].

Table 2. BMPs considered and their training/test sets

	BPM	Number of observations Training set Failures | non-failures	 Test set Failures | non-failures
Annual BPMs	M1	41 | 41	41 | 22,289
	M2	89 | 89	88 | 22,241
	M3	113 | 113	112 | 22,217
	M4	123 | 123	124 | 22,207
	M5	141 | 141	141 | 22,189
	M6	170 | 170	170 | 22,160
	M7	160 | 160	161 | 22,170
	M8	116 | 116	115 | 22,214
	M9	61 | 61	61 | 22,269
Multiperiod BPMs	M1-3	242 | 242	242 | 22,088
	M1-6	677 | 677	676 | 21,653
	M1-9	1,013 | 1,013	1,014 | 21,317
	M4-6	434 | 434	435 | 21,896
	M7-9	337 | 337	337 | 21,993

2.3. Genetic Programming FS Methods

This paper addresses dimensionality reduction by means of FS using GP. The underlying idea is that, in the evolutionary process of GP, variables that are relevant will be maintained, over generations, in the population, as opposed to irrelevant variables that will be progressively eliminated by selection pressure. Therefore, GP has an advantage in feature selection over other ML FS techniques, as it is an intrinsic selection process. This is because the evolution of the final solutions involves an automatic selection of which are the relevant input variables in the optimized solutions.

With this idea, two feature selection methods were considered to automatically obtain the most relevant variables for BPMs. The first is detailed in [7, 8], which is summarized below, while the second incorporates a new proposal as a variant of the first.

First variant of Genetic Programming Feature Selection ($GPFS_1$). For FS, a set of GP independent runs is considered. The relative frequency of occurrence of each explanatory variable in a subset of these independent GP runs is used. The most frequent inputs are expected to be the most relevant variables for the prediction models.

For FS, a null hypothesis is considered in which the frequency of occurrence of each explanatory variable is due to chance, leading to a normal distribution of selection for

each feature. A statistical test (Kolmogorov-Smirnov test), considering the real distribution obtained for each variable, allows to reject (or not) the aforementioned null hypothesis, and the p-value obtained from it will also allow establishing a relevance ranking of each variable. The steps of the GPFS1 process can be summarized as:

i. For each of the BPMs (M1, M2, ...), a large experiment (1,000 independent runs of GP) is available with the totality of the input variables. The high number of runs (1,000) guarantees that, in independent experiments with that number of runs, a very high degree of agreement in the selected variables is obtained (data not detailed here).
ii. Based on the results of the experiment with the independent runs, a subset of GP runs (5%) is established, those that provide solutions considered to be the best for the application: those solutions that provide a classification (evaluated on the test set) with the highest AUC (Area Under the ROC Curve) value. In addition, these solutions are selected with a filtering, choosing among the 5% of solutions with the best sum of true positive rate (TPR) + true negative rate (TNR). Therefore, solutions that focus their performance on the extremes of the ROC curve are avoided, which is especially relevant given that the test sets are unbalanced.
iii. For each explanatory variable, its relative frequency and its p-value are calculated (the null hypothesis being that the relative frequency is due to chance), both referring to the solutions evolved in the subset of selected GP runs (in the whole evolutionary process). Finally, those variables with p-value<0.05 will be selected, i.e., those with statistically relevant results and not due to chance. That is, the number of selected features (p-value<0.05) is determined automatically.

Finally, with the variables determined as most relevant in each prediction model, a new experiment is performed (1,000 new GP runs), but now with the variables selected with $GPFS_1$, which will provide the best final prediction model (selecting the one that provides the best AUC over the test set, filtering again from those with the best TPR + TNR). That is, GP is first used as an FS method and, in a second independent step, it is used as a classification method to provide the final evolved BPMs.

Second variant of Genetic Programming Feature Selection ($GPFS_2$). One of the risks of FS is losing information about relevant variables. Feature selection is a non-monotonic problem, since it is difficult for the best subset of p features to include the best subset of q features, with $q<p$. With $GPFS_1$, dimensionality reduction is addressed in terms of the statistical significance of the relative frequency of each of the variables in the input set (p-value<0.05). However, another approach that tries to avoid the loss of possible relevant variables can be analyzed.

Thus, the second method proposed here is a stepwise process that addresses a progressive decrease in the size of the set of explanatory variables, removing, in each of the steps, those variables that show clear irrelevance (p-value>=0.3333). Therefore, this second method follows a similar philosophy to that applied in Guyon et al. [18] with its recursive feature elimination algorithm. The main difference is that $GPFS_2$ takes into account the statistical significance of the presence of features.

Therefore, the aim is to progressively decrease the number of input variables, with the consequent minimization of the risk of elimination of relevant variables. To this end, it is established that the variables selected should have a p-value<0.3333 (which should

minimize the risk of eliminating relevant variables). A p-value<0.05 or p-value<0.01 are usual values for determining that a variable is statistically relevant. A p-value<0.3333 implies that variables with a probability of being relevant >= 66.67% will be taken, a very lax criterion, compared to the standard 95-99%.

Three variants were considered with this alternative. These three variants differ in the number of steps considered in the selection of variables, where each step selects a subset of the previously selected features:

1. $GPFS_{21}$, in which variables are selected according to the p-value of the relative frequency of each variable in the subset of solutions of the independent GP runs (p-value<0.3333, i.e., same as $GPFS_1$, changing the threshold p-value).
2. $GPFS_{22}$, repeating the process in a second iterative process. That is, using as inputs the subset of variables selected with $GPFS_{21}$, a subsequent experiment is performed with 1,000 independent GP runs, calculating again the relative frequency of occurrence of each variable. Choosing again those with p-value<0.3333 will provide a more limited set of $GPFS_{22}$ selected variables.
3. $GPFS_{23}$, repeating the process in a third iterative process, providing a set of $GPFS_{23}$ selected variables.
   In other words, a repetitive cycle of the successive process of elimination of non-relevant variables with 3 iterations.

Some considerations regarding $GPFS_1$ and $GPFS_2$. The FS methods are traditionally classified as filter, wrapper and embedded methods [23]. Filter methods attempt to detect variables whose values differ sufficiently between different classes, but not within instances of the same class, and are therefore independent of the classification method. In wrapper methods, features are evaluated based on their performance in a specific modeling algorithm, making them more computationally expensive. Finally, in embedded methods, feature selection is performed as the learning method is applied, since selection is integrated into the learning method. Consequently, the first consideration is that $GPFS_1$ and $GPFS_2$ can be classified as embedded FS methods, since the FS process is embedded in the GP learning process.

Secondly, these GPFS methods are context-sensitive, since the relevance of a feature is measured in the presence of other variables. Finally, $GPFS_1$ and $GPFS_2$ base their approaches on the statistical significance of variable appearance (contrary to the works in [34] and [42]), as indicated in the Introduction section. Moreover, the consideration of a large set of evolved trees in $GPFS_1$ and $GPFS_2$ (to calculate the frequency of variable occurrence) diminishes the possible effect of bloat, as parts of a tree that do not affect fitness (bloat) may appear in an evolved tree, but are unlikely to appear in all evolved trees and in independent GP runs.

2.4. GP Environment

All GP processes were implemented with the HeuristicLab (HL) environment [49, 53]. Standard tree-based representations were considered, modeled in HL as a problem of symbolic classification.

HL allows to divide the training set into a fitness subset and a validation subset. The fitness subset drives the evolutionary process, since the fitness of solutions (MSE, Table 3) is determined by this subset. At the end of the evolutionary process, HL provides two final solutions: the best training solution, which corresponds to the solution with the

best fitness in the total training set, and the best validation solution, which corresponds to the best program in the validation subset. The objective of the validation subset is to discover solutions that may perform well when generalized to the test set, that is, to prevent over-fitting. Note that these 2 solutions are the ones taken into account in each GP run to finally select the one with the best AUC (on the test set) (Section 2.3).

The GP parameter tuning was performed with a simple sweep of the most relevant parameters, while others were set to the standard values considered in HL (e.g., Model Creator and Solution Creator). Table 3 summarizes the GP parameters (HL nomenclature).

Table 3. GP parameters

GP Parameter	Value/option
Fitness function	MSE (Mean Squared Error, between real and predicted values)
Solution Creator	Probabilistic Tree Creator
Tree Grammar	Arithmetic functions (+, -, *, /)
Maximum Depth	10 (maximum depth of the tree/program)
Maximum Length	100 (maximum number of tree nodes)
Generations	100
Population Size	1,500
Selector	Tournament - Window size: 8 (used in mutation and crossover)
Elites	Preserves the best solution between generations
Crossover	Subtree Swapping at the crossing point
Mutation	Multi Symbolic Expression Tree Manipulator (allows different types of mutations)
Mut. Probability	15%
Fitness and validation sets	Fitness subset: 100 % of the training set, Validation subset: 30 % of the training set
Model Creator	Accuracy Maximizing Thresholds (the solutions with the classification threshold that maximizes the percentage success in the training set are returned)

The parameter values were selected as those that provide solutions with the highest classification performances (AUC, considering the filtered solutions) when the BPMs are evaluated on their test sets. To obtain high-performance BPMs, basic arithmetic functions were sufficient, which also facilitates the interpretability (at least readability) of the prediction models. The fitness/validation split shown in Table 3 (the subsets of fitness and validation need not be disjoint) was also chosen experimentally with the same AUC evaluation and after considering other alternatives with disjoint subsets. This experimental tuning was performed considering the total set of BPMs, so the GP parameter set is common to all BPMs.

Table 4. AUC comparison with and without dimensionality reduction with the best FS method in each BPM and using financial variables from set A. The highest values in terms of maximum, minimum and average AUC of the 5 best solutions are highlighted in bold for each model

	With dimensionality reduction				Without dimensionality reduction		
BPM	Selected variables	Maximum AUC (test)	Minimum AUC (test)	Average AUC (test)	Maximum AUC (test)	Minimum AUC (test)	Average AUC (test)
M1	$GPFS_1$	**94.29%**	**93.62%**	**93.86%**	93.91%	93.19%	93.64%
M2	$GPFS_{22}$	**91.53%**	**90.88%**	**91.11%**	90.65%	90.22%	90.44%
M3	$GPFS_{23}$	85.60%	**85.49%**	**85.54%**	**85.64%**	84.79%	85.10%
M4	$GPFS_{23}$	**85.78%**	**85.51%**	**85.64%**	85.61%	84.89%	85.23%
M5	$GPFS_{23}$	**80.84%**	**80.31%**	**80.54%**	80.45%	79.56%	79.83%
M6	$GPFS_1$	**79.02%**	**78.60%**	**78.78%**	78.98%	78.21%	78.50%
M7	$GPFS_{21}$	**76.08%**	**74.26%**	**75.12%**	74.00%	73.71%	73.81%
M8	$GPFS_1$	**67.39%**	**66.77%**	**67.05%**	66.25%	65.18%	65.81%
M9	$GPFS_1$	**70.21%**	**68.27%**	**68.91%**	67.93%	67.37%	67.60%
M1-3	$GPFS_1$	**89.46%**	**89.21%**	**89.34%**	89.10%	88.67%	88.91%
M1-6	$GPFS_1$	**83.93%**	**83.74%**	**83.79%**	83.56%	83.29%	83.45%
M1-9	$GPFS_1$	**80.12%**	**79.90%**	**80.04%**	79.37%	79.23%	79.28%
M4-6	$GPFS_{23}$	**81.70%**	**81.45%**	**81.56%**	81.63%	81.34%	81.47%
M7-9	$GPFS_{23}$	**73.51%**	**72.73%**	**73.03%**	73.12%	72.44%	72.62%

3. Results

3.1. Results with Set A Using GP as a Classifier

First, we compare the different GP-based FS methods explained above. The comparison can be performed, for example, considering the best AUC (over the test set and with the filtered solutions) obtained with the evolved solutions. That is, the best-evolved solutions of the independent GP runs when GP is used as a classifier, after the previous step of using GP as an FS method. However, we considered the average AUC of the 5 best solutions, in order to determine if an FS method can obtain recurrently good solutions in the different independent GP runs considered in the FS process.

Table 4 shows the best FS methods in each of the BPMs (i.e., the method with the best average AUC) and the maximum/minimum/average AUC of the 5 best solutions of the best FS method. Table 4 also includes the same data before dimensionality reduction, i.e., the evolved BPMs can use the full set of input variables.

The highest values in terms of maximum, minimum and average AUC of the 5 best solutions are highlighted in bold for each model. As can be seen and as might be expected, dimensionality reduction improves the results with respect to the use of the total number of input variables in all aspects (maximum, minimum and average AUC), with the only exception of the maximum AUC of M3. Moreover, $GPFS_{1,}$ with its

dimensionality reduction based on p-value<0.05, shows better results than other alternatives ($GPFS_{21}$, $GPFS_{22}$ or $GPFS_{23}$), but not in all BPMs (7 out of 14 models), followed by $GPFS_{23}$.

Table 5. AUC comparison with and without dimensionality reduction with the best FS method in each BPM and using variables from set B. The highest values in terms of maximum, minimum and average AUC of the 5 best solutions are highlighted in bold for each model

	With dimensionality reduction				Without dimensionality reduction		
BPM	Selected variables	Maximum AUC (test)	Minimum AUC (test)	Average AUC (test)	Maximum AUC (test)	Minimum AUC (test)	Average AUC (test)
M1	$GPFS_{23}$	**95.77%**	**95.19%**	**95.44%**	95.25%	94.40%	94.69%
M2	$GPFS_{1}$	**91.21%**	**90.76%**	**90.94%**	90.64%	89.66%	90.11%
M3	$GPFS_{1}$	**85.38%**	**84.35%**	**84.86%**	84.20%	83.83%	83.97%
M4	$GPFS_{23}$	**86.17%**	**85.40%**	**85.72%**	85.85%	84.85%	85.24%
M5	$GPFS_{1}$	**80.54%**	**80.14%**	**80.29%**	79.67%	79.11%	79.42%
M6	$GPFS_{97+15}$	**78.59%**	**78.11%**	**78.33%**	78.32%	77.97%	78.21%
M7	$GPFS_{23}$	**74.64%**	**73.40%**	**73.85%**	72.79%	72.22%	72.55%
M8	$GPFS_{22}$	**73.37%**	**72.82%**	**72.96%**	70.99%	70.51%	70.73%
M9	$GPFS_{1}$	**72.78%**	**71.83%**	**72.12%**	71.28%	70.33%	70.71%
M1-3	$GPFS_{97+15}$	89.39%	**89.10%**	**89.21%**	**89.44%**	88.57%	88.86%
M1-6	$GPFS_{23}$	**84.04%**	**83.83%**	**83.91%**	84.03%	83.72%	83.85%
M1-9	$GPFS_{1}$	**80.11%**	**80.01%**	**80.07%**	79.70%	79.42%	79.51%
M4-6	$GPFS_{1}$	**82.03%**	**81.95%**	**81.97%**	81.55%	81.20%	81.30%
M7-9	$GPFS_{1}$	**75.70%**	**74.85%**	**75.16%**	74.42%	73.96%	74.12%

3.2. Results with Set B Using GP as a Classifier

Table 5 shows the same information as in the previous table when the explanatory variables correspond to set B. Now, the different GPFS alternatives can select among 112 explanatory variables (97 from set A plus 15 additional variables defining set B). An additional possibility was included in the analysis: on top of the previous dimensionality reduction (with set A), the feature set corresponding to the best reduction method applied to set A (FS method specified in Table 4), increased by the 15 variables that differentiate sets A and B, possibility labeled "$GPFS_{97+15}$".

Several comments can be made from the results shown in Table 5. First, dimensionality reduction again improves the results with respect to the use of the total number of variables, as can be seen with the best values obtained in the maximum, minimum and average AUC (highlighted in bold for each model). Only in M1-3 the maximum AUC is slightly lower with dimensionality reduction than with the total variables. Secondly, for set B, again dimensionality reduction based on p-value<0.05 ($GPFS_1$) shows better results, but not in a generalized way (7 of 14 BPMs).

Another aspect to comment on is that, in only BPMs M6 and M1-3 the best method is $GPFS_{97+15}$ (adding the 15 extra variables to the best set of variables selected from set A). That is, the extra 15 variables are not enough to obtain better results than those provided by the FS from set A (see maximum values in Table 4 for M6 and M1-3). On the other hand, the differences in AUC are not evident except in M8, with clearly higher AUC results using set B as the initial variables to be selected. The comparative results between Tables 4 and 5 are summarized in Table 6. Each cell in Table 6 specifies whether the best AUC values (over the test set, with respect to the maximum, minimum and average values) are obtained using set A or set B with the corresponding best evolved BPM (AUC values shown in Tables 4 and 5).

It is observed (Table 6) that the situation is one of total equilibrium when deciding on the best initial set of explanatory variables. For example, in the case of taking into account the BPMs evolved after dimensionalty reduction, considering the maximum and minimum AUC, in 7 of the 14 models, the dimensionality reduction applied on the set of financial variables of the company (set A) presents better results than when the reduction is applied on the total of variables of the company and of the economic environment (set B). Considering the average AUC, the use of set B of possible explanatory variables to be selected also presents better results (8 out of 14 cases).

The same analysis, considering the best-performing BPMs without dimensionality reduction, provides similar conclusions. For example, considering the maximum and average AUC values, in 8 out of 14 cases (maximum AUC) and 7 out of 14 cases (average AUC), set B offers better results in terms of AUC. Consequently, the dilemma about the use, in the initial set of input variables, of variables other than financial variables is not clearly resolved, as previously analyzed in [8].

Table 6. Comparison of results between using set A or B on the evolved BPMs. Each cell corresponds to the set that provides the best AUC results on the best-evolved BPM (values shown in Tables 4 and 5)

	With dimensionality reduction			Without dimensionality reduction		
BPM	Maximum AUC (test)	Minimum AUC (test)	Average AUC (test)	Maximum AUC (test)	Minimum AUC (test)	Average AUC (test)
M1	B	B	B	B	B	B
M2	A	A	A	A	A	A
M3	A	A	A	A	A	A
M4	B	A	B	B	A	B
M5	A	A	A	A	A	A
M6	A	A	A	A	A	A
M7	A	A	A	A	A	A
M8	B	B	B	B	B	B
M9	B	B	B	B	B	B
M1-3	A	A	A	B	A	A
M1-6	B	B	B	B	B	B
M1-9	A	B	B	B	B	B
M4-6	B	B	B	A	A	A
M7-9	B	B	B	B	B	B

3.3. Comparison of GP as a Classifier with other Classifiers Adjusted with AutoWeka

To check the capability of GP as a classifier, its results (with the best GPFS method), are compared with different classification methods. Our previous study in [7] performed a comparison of GP as a classifier, using $GPFS_1$ as the FS method (and other widely used FS methods), with a large set of standard classification methods (including ensembles of classifiers), with the standard values in the defining parameters of the classifiers and using the variables selected by $GPFS_1$. However, in order to make the conclusions from the analysis of GP as a classifier more robust, a more thorough comparison can be made with other ML classifiers where there is now an automatic tuning process for their defining parameters. For this purpose, the Weka package called AutoWeka [24] was used, which provides an automatic adjustment of the parameters in each ML method for correct comparison. Automated Machine Learning (AutoML) is the comprehensive automation of machine learning model building tasks, with AutoWeka being one of the leading software packages for its implementation. AutoML is widely used for classification and regression in sectors such as healthcare, finance, marketing and natural language processing. AutoML has also been used in predicting business failure [31, 33, 37].

AutoWeka is a free and open-source software package that automates algorithm selection and hyperparameter settings for each classification method in the learning process. Its main functions are: i) Algorithm self-selection: AutoWeka automatically evaluates a wide range of ML algorithms and selects the most suitable one for a specific data set; ii) Hyperparameter optimization: AutoWeka searches for the optimal hyperparameter settings of the selected algorithm to improve model performance (including FS); iii) Fully automated approach: No user intervention is required for algorithm selection or hyperparameter settings. AutoWeka configuration parameters are minimal, the most relevant being the time limit to perform the process.

To perform the comparison, AutoWeka is supplied with the same training/test sets and the same sets (A and B) of explanatory variables used with GP. The annual BPMs were considered in the comparison. For each of the annual BPMs, AutoWeka was run with different time limits (15, 30, 60, 120, 180 and 240 minutes, in a platform with 24 GB of RAM and Intel i7-7700 processor, with 240 minutes being the heuristic time limit to obtain the best results in a generalized way). For each temporal prediction horizon (1, 2, ...9 years prior to failure) the best result of the different AutoWeka runs (15, 30, ..., 240 minutes) is chosen. The metric used for comparison is again the AUC calculated on the test set. Note that AutoWeka also performs an FS process to determine the most relevant variables to be used in each classifier.

In the case of GP, HeuristicLab allows an automatic simplification and manual pruning of the evolved solutions (oriented to improve AUC over the test set, a process that can also eliminate bloat in the evolved trees). The results of the best-evolved BPMs before such processes are shown in Tables 4 and 5, corresponding to the maximum values of the 5 best GP-evolved solutions. Table 7 shows the AUC values after applying the HL simplification and pruning to those best annual BPMs obtained after dimensionality reduction. As an example, Figure 1 includes the evolved trees of M5 (medium-long term prediction), with the evolved tree (with the best AUC) before and after simplification and pruning. The evolved trees are shown in a hierarchical

representation provided by HL, corresponding to a mathematical expression that uses the basic arithmetic operations (specified in Table 3) on the nodes of the tree to process the input information of the selected explanatory variables. The input variables are labeled in HL as *log_X*, where *log* refers to the logistic transformation discussed in Section 2.1 and *X* refers to the index of the input variable (out of the 97 in set A where the best M5 model is obtained). Table 8 includes a brief definition of these input variables used in this prediction model. As can be seen, the BPM M5 uses a considerable number of explanatory variables. At this stage, it is not possible to know the relevance of each of them in the value estimated by the model, but it is possible to note (in a very general way) two main focuses of interest: the profitability (variables 13, 37, 43, 66, 82) and the indebtedness (variables 24, 27, 59, 69, 77), which are "a priori" consistent with the prediction horizon of the model.

Table 7 also shows, for each set of explanatory variables, the AUC corresponding to the best solution found by AutoWeka for each BPM (the best considering the different time limits). The meaning of the AutoWeka classifier acronyms is as follows [24]: MLP (Multilayer Perceptron), LWL (Locally Weigththted Learning), LMT (Logistic Model Tree), Ibk (Instance-based k-neighbors), SGD (Stochastic Gradient Descent) and SMO (Sequential Minimal Optimization - support vector classification).

Table 7. Best AUC values, evaluated on the test set of annual BPMs obtained with GP (after simplification and pruning) and those obtained with AutoWeka. GP (as a classifier) uses the best feature subset selected with the GPFS methods. The values in bold correspond to the best AUC values in each of the sets of possible explanatory variables (A and B). The values in filled gray correspond to the best AUC values in the comparison between sets A and B

	AUC (test set) - set A			AUC (test set) - set B		
BPM	GP (after simplification and pruning)	AutoWeka	AutoWeka Classifier	GP (after simplification and pruning)	AutoWeka	AutoWeka Classifier
M1	**95.02%**	91.80%	MLP	**95.94%**	91.30%	lazy-LWL
M2	**91.78%**	90.20%	lazy-LWL	**92.40%**	89.10%	RandomForest
M3	**86.60%**	84.80%	lazy-LWL	**85.64%**	83.50%	Bagging
M4	**86.44%**	83.50%	LMT	**87.30%**	83.60%	SimpleLogistic
M5	**82.10%**	80.10%	lazy-Ibk	81.30%	**81.60%**	RandomForest
M6	**80.02%**	75.40%	SGD	**78.95%**	76.60%	Bagging
M7	**76.18%**	72.80%	Bagging	**75.78%**	72.00%	NaiveBayes
M8	**71.75%**	63.30%	lazy-LWL	**74.91%**	63.20%	SMO
M9	**71.41%**	65.70%	SGD	**73.89%**	68.90%	MLP

It is observed that the models obtained with GP offer better performance than those obtained with other ML classifiers. Whether only financial variables (set A) or financial and non-financial variables are used (set B), for all the temporal horizons of the prediction models, the AUC evaluated on the test set of the models obtained with GP is superior to that obtained with AutoWeka, with considerable increases in AUC in some cases, as in M8 (8.45% AUC increase with set A and 11.71% with set B). The only exception of M5 with set B.

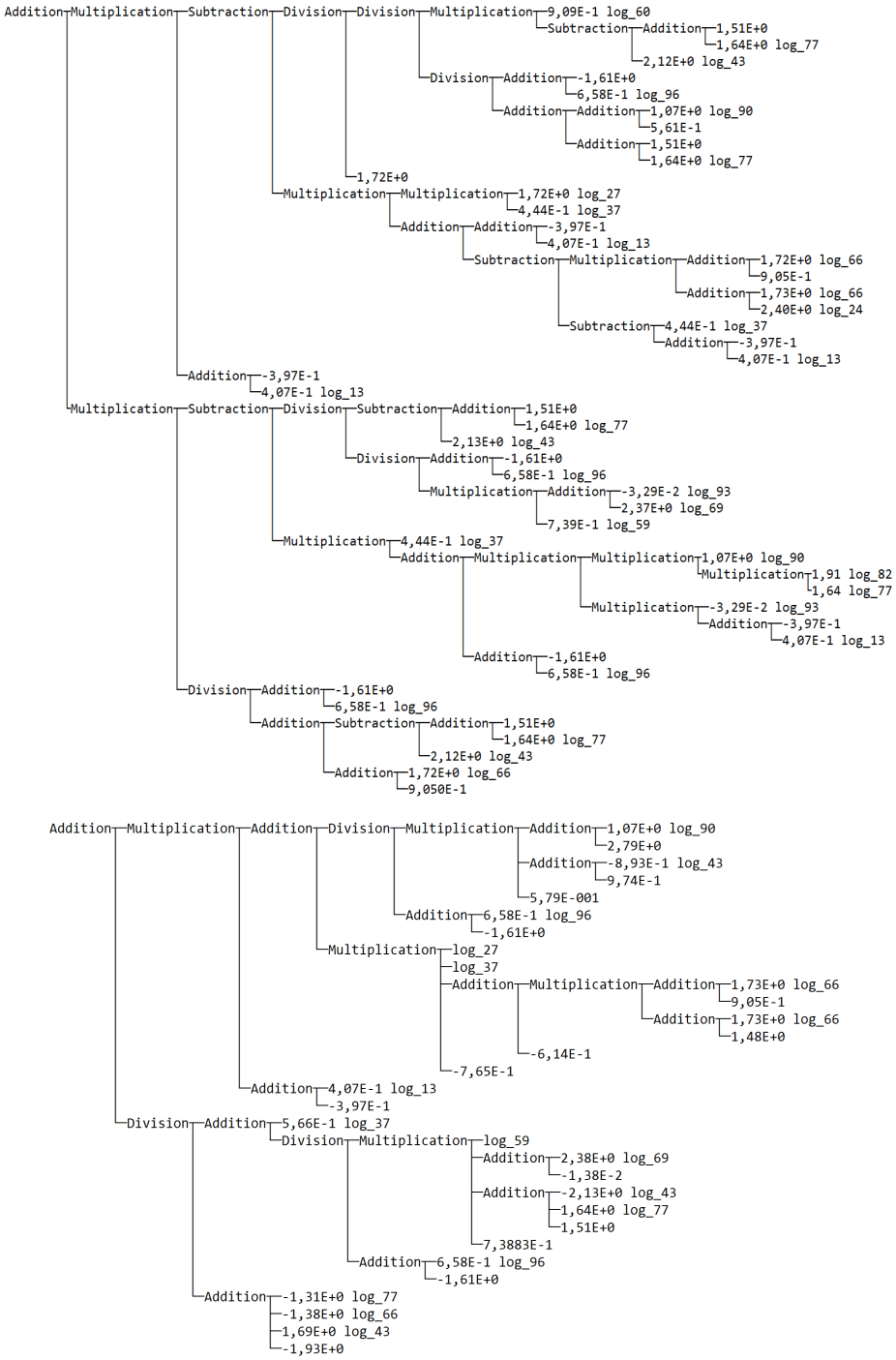

Fig. 1. Best evolved BPM M5, before (upper figure) and after (bottom figure) the simplification and pruning process

The comparison of computational time should also be taken into account. In this comparison, as a reference, each independent run of GP implies an average computation time of 76 seconds, using a model with an average training set (282 observations) and evaluating 150,000 solutions per independent run (population size=1,500, Generations=100, Table 3). That is, the 1,000 independent runs involve 1,267 minutes.

In the case of AutoWeka, as stated, 240 minutes was the heuristic time limit to obtain the best AUC values in a generalized way. For example, doubling this time to 480 minutes (representing an average of 7,500 configurations tested by AutoWeka), when using set A, the improvement in the percentage of misclassified instances in the training set is generalized in the nine BPMs (5 of them present an error of 0 in the classification of the training set). However, only one of the nine AUCs evaluated on the test set, with 480 minutes, improves (by less than 2%) on those shown in Table 7.

It is noteworthy that, even with the BPMs evolved before simplification and pruning, the AUC values (shown in Tables 4 and 5, maximum AUC) in practically all BPMs are better compared to those obtained by the classifiers selected by AutoWeka (the only exception is again M5 with set B). Consequently, the combination of the best FS method based on GP together with GP as a classifier improves the results of the other ML methods, even with the automatic adjustment provided by AutoWeka in those other ML methods.

Table 8. Variables of evolved BPM M5

Variable	Definition
13	EBITDA / Interest expenses
24	Total debt / Total assets
27	(Current liabilities - Cash) / Total assets
37	Net income / Shareholder funds
43	Operating ROA
59	Accounts payable / Total sales
60	Inventory / Total sales
66	Interest expense percent of EBITDA
69	Interest expense / Total debt
77	Change in financial debts
82	Variation *t* over *t-1* in purchases / Variation t over *t-1* in accounts payable
90	LN (Total assets) squared
93	Degree of assets decomposition
96	Number of employees

4. Conclusions and Future Work

In this paper, a novel feature selection approach based on genetic programming was proposed and defined, with an iterative process of eliminating irrelevant variables, as opposed to a previous approach that directly selects relevant variables based on the statistical significance of their appearance in the evolved GP solutions. The novel proposal maintains the statistical significance to select the variables, but with a more lax

criterion and with its repetition in an iterative process to reduce the possibility of removing possible relevant variables that could be discarded with a strict selection of variables.

The comparison of the results has been performed considering a large number of BPMs (annual and multiperiod) with different prediction horizons, as well as considering two sets (A and B) of explanatory variables in the BPMs. Several conclusions can be drawn from the experimentation with both FS strategies and both sets of explanatory variables:

i. First, the results show the usefulness of the proposed FS methods, since their use improves BPM performance in practically all cases with respect to the use of the whole set of input variables. It also shows that the proposed FS methods manage to automatically select the most relevant variables for each particular BPM, highlighting that the proposed FS methods automatically determine the number of explanatory variables to be taken into account. Moreover, this selection is performed in a context-sensitive manner (i.e., considering the relationship with the other selected variables).

ii. Secondly, the comparison of results, with both FS strategies and using GP as classifier, shows the robustness of the approach that directly selects the relevant variables based on their statistical significance ($GPFS_1$, variables selected with p-value<0.05), since it is the FS method with better results in more evolved BPMs. However, in several cases, $GPFS_1$ was outperformed by the alternative iterative FS method, which means that some relevant variables can be discarded with strict selection (p-value<0.05). The FS methods can be further developed, possibly by including, in addition to the p-value, some other indicator (such as mean or median) of relative frequency and/or by modifying the established p-value limits.

iii. Third, the comparison of the results between the use of sets A and B of explanatory variables (the latter incorporating variables from the economic environment) does not provide conclusive results on the dilemma of whether the use of variables other than financial variables is necessary to improve the performance of predictive models. It can be hypothesized that, in our case, the standardization of variables with the logistic transformation implies that, indirectly, the effect of the economic environment is incorporated in the variables considered in set A (only financial variables), so that set B does not provide additional information to improve the classification performance.

Furthermore, the comparison between GP-based FS approaches was performed using GP as a classifier. A comparison between GP as a classifier and other ML classifiers was also performed, with the automatic tuning (including FS) provided by AutoWeka. Another conclusion can be drawn from this comparison is:

iv. Even with AutoWeka's automatic search for the best classifier (and its parameter tuning), GP provides better performance results. As discussed in the discussion of the results of this comparison, it is noteworthy that the performance (AUC on the test set) of the BPMs that were evolved before the simplification and subsequent pruning processes (aimed at improving AUC performance) is better (in virtually all BPMs) compared to the performance of the AutoWeka classifiers. This demonstrates the synergy obtained by using GP both as a feature selector and as a classifier.

In future work, since GP-defining parameters were experimentally adjusted and are common to all the BPMs considered, an automatic adjustment of the parameters could be performed, also particularized for each BPM and in order to improve the AUC results in each BPM. In addition, an analysis of the types of variables selected (groups of financial variables), not analyzed here for each of the temporal horizons, should also be carried out.

Acknowledgements. This study was funded by the Xunta de Galicia and the European Union (European Regional Development Fund - Galicia 2021-2027 FEDER Program), with grants CITIC (ED431G 2023/01), GPC ED431B 2022/33 and GRC ED431C 2025/49, as well as by the Spanish Ministry of Science, Innovation and Universities (MICIU/AEI/10.13039/501100011033, project PID2023-148531NB-I00).

References

1. Alaka, H. A., Oyedele, L. O., Owolabi, H. A., Kumar, V., Ajayi, S.O., Akinade, O. O., Bilal, M.: Systematic Review of Bankruptcy Prediction Models: Towards a Framework for Tool Selection. Expert Systems with Applications Vol. 94, 164-184. (2018)
2. Altman, E. I.: Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy. The Journal of Finance, Vol. 23, 589-609. (1968)
3. Altman. E. I., Iwanicz-Drozdowska, M., Laitinen, E., Suvas, A.: Financial and Non-Financial Variables as Long-Horizon Predictors of Bankruptcy. SSRN Electronic Journal. (2015)
4. Altman, E. I., Iwanicz-Drozdowska, M., Laitinen, E. K., Suvas, A.: A Race for Long Horizon Bankruptcy Prediction. Applied Economics, Vol. 52, 4092-4111. (2020)
5. Aziz, M. A., Dar, H. A.: Predicting Corporate Bankruptcy: Where We Stand? Corporate Governance: The International Journal of Business in Society, Vol. 6, 18-33. (2006)
6. Beade, Á., Rodríguez, M., Santos, J.: Business Failure Prediction Models with High and Stable Predictive Power Over Time Using Genetic Programming. Operational Research, Vol. 24, Article 52. (2024)
7. Beade, Á., Rodríguez, M., Santos, J.: Variable Selection in The Prediction of Business Failure Using Genetic Programming. Knowledge-Based Systems, Vol. 289, 111529. (2024)
8. Beade, Á., Rodríguez, M., Santos, J.: Genetic Programming for Feature Selection in Business Failure Prediction. Comparison of the Use of Financial Variables and Economic Environment Variables. In Proceedings of 2024 International Conference on INnovations in Intelligent SysTems and Applications (INISTA). 1-6. (2024)
9. Beade, Á., Rodríguez, M., Santos, J.: Multiperiod Bankruptcy Prediction Models with Interpretable Single Models. Comput Econ, Vol. 64, 1357-1390. (2024)
10. Beaver, W. H., McNichols, M. F., Rhie, J-W.: Have Financial Statements Become Less Informative? Evidence from the Ability of Financial Ratios to Predict Bankruptcy. Rev Acc Stud, Vol. 10, 93-122. (2005)
11. Bellovary, J. L., Giacomino, D. E., Akers, M. D.: A Review of Bankruptcy Prediction Studies: 1930 to Present. Journal of Financial Education, Vol. 33, 1-42. (2007)
12. Das, S. R., Hanouna, P., Sarin, A.: Accounting-Based Versus Market-Based Cross-Sectional Models of CDS Spreads. Journal of Banking & Finance, Vol. 33, 719-730. (2009)
13. Dasilas, A., Rigani, A.: Machine Learning Techniques in Bankruptcy Prediction: A Systematic Literature Review. Expert Systems with Applications, Vol. 255, 124761. (2024)
14. European Banking Authority: Discussion Paper on Machine Learning for IRB Models - EBA/DP/2021/04. (2021)

15. European Commission: Proposal for a Regulation Laying Down Harmonised Rules on Artificial Intelligence Shaping Europe's Digital Future (2021). Available: https://digital-strategy.ec.europa.eu/en/library/proposal-regulation-laying-down-harmonised-rules-artificial-intelligence.
16. European Parliament: Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance) - EU Law in Force - Publications Office of the EU. In: EU Law in Force (2016). Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng
17. García, V., Marqués, A. I., Sánchez, J. S.: Exploring the Synergetic Effects of Sample Types on the Performance of Ensembles for Credit Risk and Corporate Bankruptcy Prediction. Information Fusion, Vol. 47, 88-101. (2019)
18. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene Selection for Cancer Classification Using Support Vector Machines. Machine Learning, Vol. 46, 389-422. (2002)
19. Hosaka, T.: Bankruptcy Prediction Using Imaged Financial Ratios and Convolutional Neural Networks. Expert Syst Appl, Vol. 117, 287-299. (2019)
20. du Jardin, P.: Dynamics of Firm Financial Evolution and Bankruptcy Prediction. Expert Systems with Applications, Vol. 75, 25-43. (2017)
21. du Jardin, P.: Dynamic Self-Organizing Feature Map-Based Models Applied to Bankruptcy Prediction. Decision Support Systems, Vol. 147, 113576. (2021)
22. du Jardin, P., Séverin, E.: Forecasting Financial Failure Using a Kohonen Map: A Comparative Study to Improve Model Stability Over Time. European Journal of Operational Research, Vol. 221, 378-396. (2012)
23. Jović, A., Brkić, K., Bogunović, N.: A Review of Feature Selection Methods with Applications. In Proceedings of 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). 1200-1205. (2015)
24. Kotthoff, L., Thornton, C., Hoos, H. H., Hutter, F., Leyton-Brown, K.: Auto-Weka: Automatic Model Selection and Hyperparameter Optimization in Weka. In Hutter, F., Kotthoff. L., Vanschoren. J. (eds.): Automated Machine Learning: Methods, Systems, Challenges. Springer International Publishing, Cham, 81-95. (2019)
25. Koza, J. R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. The MIT Press, Cambridge, MA, USA. (1992)
26. Laitinen, E. K.: Financial Ratios and Different Failure Processes. Journal of Business Finance & Accounting, Vol. 18, 649-673. (1991)
27. Laitinen, E. K., Camacho-Miñano, M-M., Muñoz-Izquierdo, N.: A Review of the Limitations of Financial Failure Prediction Research. Revista de Contabilidad - Spanish Accounting Review, Vol. 26, 255-273. (2023)
28. Liu, G., Ma, J., Hu, T., Gao, X.: A Feature Selection Method with Feature Ranking Using Genetic Programming. Connection Science, Vol. 34, 1146-1168. (2022)
29. Lombardo, G., Bertogalli, A., Consoli, S., Reforgiato Recupero, D.: Natural Language Processing and Deep Learning for Bankruptcy Prediction: An End-to-End Architecture. IEEE Access, Vol. 12, 151075-151091. (2024)
30. Mai, F., Tian, S., Lee, C., Ma, L.: Deep Learning Models for Bankruptcy Prediction Using Textual Disclosures. Eur J Oper Res, Vol. 274, 743-758. (2019)
31. Mohan, B., Badra, J.: A Novel Automated Superlearner Using a Genetic Algorithm-Based Hyperparameter Optimization. Advances in Engineering Software, Vol. 175, 103358. (2023)
32. Muslim, M. A., Dasril, Y.: Company Bankruptcy Prediction Framework Based on the Most Influential Features Using XGBoost and Stacking Ensemble Learning. International Journal of Electrical and Computer Engineering (IJECE), Vol. 11, 5549-5557. (2021)
33. Nazareth, N., Ramana Reddy, Y. V.: Financial Applications of Machine Learning: A Literature Review. Expert Systems with Applications, Vol. 219, 119640. (2023)

34. Neshatian, K., Zhang, M.: Using Genetic Programming for Context-Sensitive Feature Scoring in Classification Problems. Connection Science, Vol. 23, 183-207. (2011)
35. Nguyen, H. H., Viviani, J-L., Ben Jabeur, S.: Bankruptcy Prediction Using Machine Learning and Shapley Additive Explanations. Rev Quant Finan Acc., Vol. 65, 107-148 (2023)
36. Nyitrai, T., Virág, M.: The Effects of Handling Outliers on the Performance of Bankruptcy Prediction Models. Socio-Economic Planning Sciences, Vol. 67, 34-42. (2019)
37. Papík, M., Papíková, L.: The Possibilities of Using AutoML in Bankruptcy Prediction: Case of Slovakia. Technological Forecasting and Social Change, Vol. 215, 124098. (2025)
38. Pawełek, B., Pociecha, J., Kostrzewska, J., Baryła, M., Lipieta, A.: Problem of Outliers in Corporate Bankruptcy Prediction. Repository KITopen (2017) Available: https://publikationen.bibliothek.kit.edu/1000069937
39. Poli, R., Langdon, W. B. (William B), McPhee, N. F., Koza, J. R.: A Field Guide to Genetic Programming. Lulu Press. Available: lulu.com (2008)
40. Ratajczak, P., Szutowski, D., Szulczewska-Remi, A.: Long-Term Bankruptcy Prediction. Systematic Literature Review. Research in Corporate Finance (2022) Available: http://dx.doi.org/10.2139/ssrn.4054665.
41. Sathyanarayana, N., Narayanan, R.: A Systematic Review of Models for the Prediction of Corporate Insolvency. Salud, Ciencia y Tecnología - Serie de Conferencias, Vol. 3, 952-952. (2024)
42. Suárez, R. R., Valencia-Ramírez, J. M., Graff, M.: Genetic Programming as a Feature Selection Algorithm. In Proceedings of 2014 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC). 1-5. (2014)
43. Szántó, T. K.: Handling Outliers in Bankruptcy Prediction Models Based on Logistic Regression. Public Finance Quarterly, Vol. 69, No. 3. (2023)
44. Tian, S., Yu, Y.: Financial Ratios and Bankruptcy Predictions: An International Evidence. International Review of Economics & Finance, Vol. 51, 510-526. (2017)
45. Tian, S., Yu, Y., Guo, H.: Variable Selection and Corporate Bankruptcy Forecasts. Journal of Banking & Finance, Vol. 52, 89-100. (2015)
46. Tsai, C-F., Sue, K-L., Hu, Y-H., Chiu, A.: Combining Feature Selection, Instance Selection, and Ensemble Classification Techniques for Improved Financial Distress Prediction. Journal of Business Research, Vol. 130, 200-209. (2021)
47. Vellamcheti, S., Singh, P.: Class Imbalance Deep Learning for Bankruptcy Prediction. In Proceedings of First International Conference on Power, Control and Computing Technologies (ICPC2T). 421-425. (2020)
48. Volkov, A., Benoit, D. F., Van den Poel, D.: Incorporating Sequential Information in Bankruptcy Prediction with Predictors Based on Markov for Discrimination. Decision Support Systems, Vol. 98, 59-68. (2017)
49. Wagner, S., Kronberger, G., Beham, A., Kommenda, M., Scheibenpflug, A., Pitzer, E., Vonolfen, S., Kofler, M., Winkler, S., Dorfer, V., Affenzeller, M.: Architecture and Design of the HeuristicLab Optimization Environment. In: Klempous, R., Nikodem, J., Jacak, W., Chaczko, Z. (eds.) Advanced Methods and Applications in Computational Intelligence. Springer, 197-261. (2014)
50. Zhao, J., Ouenniche, J., De Smedt, J.: Survey, Classification and Critical Analysis of the Literature on Corporate Bankruptcy and Financial Distress Prediction. Machine Learning with Applications, Vol. 15, 100527. (2024)
51. Zoričák, M., Gnip, P., Drotár, P., Gazda, V.: Bankruptcy Prediction for Small- and Medium-Sized Companies Using Severely Imbalanced Datasets. Economic Modelling, Vol. 84, 165-176. (2020)
52. SABI database. Available: https://www.einforma.com/soluciones-y-herramientas/sabi
53. HeuristicLab. Available: https://dev.heuristiclab.com/trac.fcgi/

Ángel Antonio Beade Torreiro defended his PhD thesis "Predicting business failure using Genetic Programming" in 2024 at the University of A Coruña, Spain.

José Santos Reyes is Professor in the Department of Computer Science and Information Technologies at the University of A Coruña, CITIC (Centre for Information and Communications Technology Research), Spain. His research interests include artificial life, neural computation, evolutionary computation, autonomous robotics and computational biology.

Manuel López is a Chair Director at Collections and Payments Manager at ABANCA, and adjunct professor at the University of A Coruña, Spain.

Received: February 25, 2015; Accepted: December 01, 2025.