Detection of Misstated Financial Reports Using XBRL Mining and Intelligible MLP

Duarte Trigueiros

ISTAR, University Institute of Lisbon, Portugal
duarte.trigueiros@iscte-iul.pt

Abstract. Considerable effort has been devoted to the development of integrated software to assist in the detection of financial misstatements. Despite this, the use of such tools has been sparse due to the opacity of the resulting output and the complicated task of importing the financial data they require. This article presents a conceptual framework for modelling financial statements that leads to significantly improved performance, allowing a Multilayer Perceptron with a modified learning method to form internal representations that can be easily interpreted by financial analysts. The article discusses the use of XBRL data extraction from the web, showing how a judicious selection of accounts can help solving the cumbersome problem of importing data. The resulting tool makes the detection of financial misstatements both understandable and easy.

Keywords: Financial Misstatement, Web Mining, XBRL, Knowledge Extraction, Multilayer Perceptron, Financial Analysis, Financial Ratio.

1 Introduction

This article describes software that helps to identify misstatements in financial reports of listed companies. The aim is to simplify and make more accessible an extensively studied but scarcely utilised application by integrating web searching with AI tools.

Financial misstatements cost the owners of US companies more than $400 billion a year and continue on a massive scale despite all the deterrent measures put in place [1]. Internal control is proving largely ineffective because most fraud occurs at the top of the organisation, where audit controls cannot reach [2].

AI and statistical methods are also used to identify misstated accounts [3][4][5] but analysts rarely use these tools due to the cumbersome data importing tasks they require and the fact that results are opaque [2]. Since analysts are accountable for their decisions, any tool they use to support decisions must be understandable.

The tool described here is based on a conceptual framework that leads to improved performance and enables a Multilayer Perceptron (MLP) with a modified learning method to internally construct representations such as financial ratios that are easily interpreted by analysts. A judicious use of the XBRL reporting language then solves the tedious problem of extracting data from the web. The resulting tool makes the detection of financial misstatements understandable and easy.

Figure 1. The most basic hierarchy of attributes used in financial analysis.

The article is structured as follows: Section 2 characterises the problem, cites existing research and develops the framework on which the tool is based; Section 3 depicts the methods used; Section 4 describes the data and results.

2 The conceptual framework

There are countless types of deception [6][7] and a framework designed to uncover credit card fraud, for example, may not be efficient at detecting other types of deceit. The Multilayer Perceptron has been used in financial misstatement research [8][9][10] [11][12] with classification accuracy of less than 75 per cent for large, inhomogeneous samples, whereas for small, homogeneous samples, accuracy can increase to 85 per cent [12]. The difference between Type I and Type II errors is at least 10 per cent.

Auditing tools that scan batches of transactions for consistency or suspicious coincidences are common and in use, but a comprehensive review of available resources did not identify any tool to perform misstatement detection in published reports.

Using large, inhomogeneous data and a balanced design, the out-of-sample accuracy of the tool presented here is around 88 per cent with a 5 per cent imbalance. Such a result is achieved regardless of whether an MLP or other algorithms are used. Most publications focus on comparing the performance of algorithms, but here the algorithm is relevant in so far as it performs knowledge discovery. The observed performance gain is due to the framework described in this section.

2.1 AI-based financial analysis

Listed companies are required to report their financial position and gains at the end of each period. To do this, companies prepare a set of money amounts with a meaning: income and expenses for the period, period-end assets, liabilities, and other. These reports are produced through an accounting process that involves recognising, adjusting, and aggregating into a collection of items (set of accounts) all significant transactions that occur during the period. After publication, financial reports are analysed by regulators and other parties interested in making decisions about individual companies and industry groups, in what is called ‘financial analysis’.

Financial analysis aims to identify the financial outlook of a company by evaluating the status of several key attributes (Figure 1). Once identified and evaluated, these

attributes provide an economic portrayal of a company's future potential and can help making important decisions such as lending or buying ownership shares. Financial attributes therefore constitute the body of knowledge on which investing, loaning, dividend paying, and other financial decisions are based.

When examined by seasoned analysts, financial reports can expose even the most sensitive financial attributes. For example, it is possible to predict a company's insolvency more than a year in advance [13]. This efficiency in communicating the state of important attributes is the motive for the falsification of accounts by managers, but fortunately the falsification itself can also be detected [3][14].

The financial analysis of a company is grounded on the comparison of items in the same report. To make such comparisons, analysts use a quotient known as a 'ratio'. For example, comparing a company's income at the end of a given period with the assets required to generate that income provides an indication of profitability in the form of a ratio. Moreover, since the size of a company has the same effect regardless of the account (provided it belongs to the same report) it follows that when a ratio is formed, size is cancelled out and no longer shown. In this way, by forming ratios, analysts can also compare the characteristics of companies of different sizes [15].

AI-based discovery of the knowledge contained in financial reports consists of assigning a set of attributes to each financial report. This assignment is done using models to recognise attributes [12]. When complete, such a process facilitates the analyst's task, allowing them to focus on problematic companies. But if the modelling is unreliable for most attributes, or if the data importing process is cumbersome, such AI-based discovery becomes unfeasible. This is the present condition.

When building AI-based models, software engineers imitate analysts by using ratios. But although ratios are the analyst's main tool, they are inadequate for modelling because of their unusual random characteristics and because the selection of ratios requires knowledge [16]. These two problems are summarised in the following lines.

First, the amounts reported in the sets of accounts, as well as their ratios, follow a multiplicative probability law rather than an additive one. Therefore, the reported amounts are accumulations and have statistical distributions close to lognormal, with long tails, highly influential cases and excessive heteroscedasticity [15].

Second, any ratio that is used requires the previous knowledge by analyst that when two amounts are compared, a financial attribute is demonstrated. As emphasised above, AI-based knowledge extraction tools must present their results in the form of ratios, otherwise they will not be understandable to analysts. Therefore, an unavoidable task that these AI tools must face is the discovery of the appropriate ratios to use.

The following lines show that these two problems are interrelated. A modelling methodology is developed to deal with the problematic randomness of financial data. The same methodology is then shown to be capable of discovering appropriate ratios for a given task.

2.2 Cross-section characterisation of reported numbers

Considering that the amounts from a specific report have in common the effect of the size of the reporting company, and hypothesizing that such amounts are lognormal, then the cross-section variability of x_{ij}, the i^{th} item from the j^{th} report, is expressed as

$$Log\ x_{ij} = a_i + v_j + u_{ij} \tag{1}$$

where the logarithm (hereafter 'log') of x_{ij} is formulated as an a_i, an expectation encompassing the mean and the difference from the mean introduced by item *i*, plus v_j, the effect of the size associated to report *j*, plus the size-free unexplained variability u_{ij} [15]. The u_{ij} are not necessarily independent. Given two items x_{den} and x_{num} from the same set of accounts, the log of the ratio of x_{num} to x_{den} will be

$$Log\frac{x_{num}}{x_{den}} = Log\ x_{num} - Log\ x_{den} = (a_{num} - a_{den}) + (u_{num} - u_{den}) \tag{2}$$

where size is cancelled. If ratios are indeed appropriate as an analytical tool, then the assumptions underlying (1) must be verified. If this were not the case, ratios for large companies would behave in a different way from ratios for small companies. Profitability, as well as the other financial attributes, would be useless.

Negative-valued items are the result of subtracting positive-valued items. If $x = x_A - x_B$ where x_A and x_B are positive-valued items, then the formulation

$$Log\ |x_A - x_B| = Log\ x_A + Log\left|1 - \frac{x_B}{x_A}\right| \tag{3}$$

holds for any *x*. As x_A is positive-valued, (1) applies to $Log\ x_A$; x_B/x_A is size-free as in (2) thus $Log|1 - x_B/x_A|$ is size-free. Using the notation in (1), the expectation of $Log\ |x_A - x_B|$ can be stated as a', being the result of adding a_A to the expectation of $Log|1 - x_B/x_A|$ and, in the same way, the unexplained term is stated as u'. From (3),

$$Log\ |x| = a' + v_j + u' \tag{4}$$

for any *x*, no matter the sign. While a' and u' in (4) do not keep the straightforward meaning of a and u in (1), remarkably v_j in (4) has the same meaning as in (1), which is why ratios remove the effect of size even if one component is negative.

Transformations to be applied to items should be able to preserve the effect of size in negative values, so that a subtraction of two such log items cancel size as in (2). The Log-Modulus [17], for example, transforms *x* into

$$sign\ x\ Log\ |x| \tag{5}$$

Assume a regression formulation with transformed items such as

$$y = a + b_1 Log\ x_1 + b_2 Log\ x_2 + \cdots \tag{6}$$

in which predictors x_1, x_2 ... are items belonging to one specific report. The financial attribute to be modelled is *y*. If x_1, x_2 ... obey (1), then (6) becomes

$$y = A + b_1 u_1 + b_2 u_2 + \cdots + (b_1 + b_2 + \cdots)\ v_j \tag{7}$$

in which $A = a + b_1 a_1 + b_2 a_2 + \cdots$ is constant. Note that the variability made available to explain *y* has two terms, namely a size-free term $b_1 u_1 + b_2 u_2 + \cdots$ and a size-related term $(b_1 + b_2 + \cdots)\ v_j$. In the case of a size-free y, the term $(b_1 + b_2 + \cdots)\ v_j$ must be equal to zero to preclude any size-related variability from explaining the relationship. Therefore, coefficients $b_1, b_2,$ ... must add to zero. When, for example, such

size-free y is predicted by a regression using just 2 log-transformed items, then $b_1 + b_2 = 0$ or $b_2 = -b_1 = b$ and (6) now is

$$y = a + b\, Log \frac{x_2}{x_1} \tag{8}$$

In short, the ratio of x_2 to $x_1$will emerge as an answer to a size-free y. As an example, if the ratio x_2/x_1 is known to predict insolvency well, then, anywhere the logs of its two components are exposed to a regression tool, then the formulation that best explains insolvency will be discovered for $b_2 = -b_1$ because the ratio is formed at that exact locus, releasing its predictive power while the effect of size is cancelled. Likewise, if size-free y is predicted by 3 log-transformed items by means of a regression, then $b_3 = -b_1 - b_2$ and (6) is $y = a + b_1 \log(x_1/x_3) + b_2 \log(x_2/x_3)$. Prompted by size-free y, two ratios emerge from 3 log-items.

When considering a regression with N log items as predictors, the manual pairing of items would be required to form ratios. The following lines show how an MLP can learn to do this pairing, forming N – 1 ratios from the N significant items.

Typically, MLP learning is carried out using cases (instances) representing varied company sizes, while the learned attribute is size-free. Hidden nodes therefore lean towards self-organising themselves into size-free representations that are at the same time capable of explaining the learned attribute. But, as mentioned, if the MLP input are lognormal, self-organisation leads to financial ratios. The MLP learning described here avoids the forming of ratios with more than one numerator or denominator by constraining nodes to have no more than two synaptic weights (connections).

MLP learning proceeds as usual until a minimum is found. Then, a severe weight pruning algorithm [20] is used, resulting in a reduction in the number of connections, input variables (especially industry dummies) and entire nodes. The next stage consists of a crude penalisation of the synaptic weights connecting the inputs that remain, the $\log x_i$ in (6), to hidden nodes: each epoch reduces the absolute value of the weights by a small margin, typically 0.001. This leads to a competition for survival among weights, and it is verified that some weights are resilient in the sense that they recover their values, while others are not resilient and quickly decay to zero and are pruned. At the final stage, all but the two largest input weights are pruned, starting from the node with the highest predictive power. The pruning is repeated in the other nodes, one by one, while the synaptic weights linking the input variables to all the hidden nodes continue to be penalised.

This learning method is, in general, capable of creating financial ratios as internal representations because, according to (7), the sum of the input weights tends to zero, and if nodes are forced to have no more than two weights, these will be symmetrical. Hidden nodes become log ratios and can be interpreted in a similar way (Figure 2).

3 Methodology

The application described here uses web mining of XBRL financial reports to obtain item amounts, then pre-selects input variables from a larger set of items and uses a specific MLP topology and learning method to form representations that are ratios.

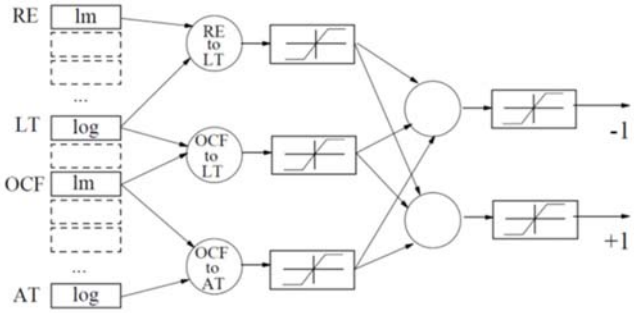

Figure 2. Given the adequately transformed four items RE (Retained Earnings), LT (Total Liabilities), OCF (Operating Cash Flow), and AT (Total Assets), the MLP forms three ratios (RE/LT, OCF/LT, OCF/AT) that are optimal in predicting insolvency.

3.1 Web-mining of XBRL financial reports

Historically, financial reports have been published in disparate formats, namely PDF and Excel, requiring a significant amount of interpretation and manual data manipulation by users to retrieve meaningful data, resulting in unaffordable expenses. From 2010, the US Securities and Exchange Commission, the UK HM Revenue and Customs, and other regulatory bodies, began requiring companies to publish their financial reports using the XML-based eXtensible Business Reporting Language (XBRL). Users of XBRL now include regulators, banks, tax filers, statistical agencies, investors, and financial analysts worldwide [18].

XBRL incorporates the XML syntax and standards, namely XML Schema, XLink, XPath and namespaces to enable unambiguous extraction of financial data. Communication is defined by metadata in taxonomies that describe reported amounts and their relationships. XBRL therefore promotes harmonisation and interpretability and simplifies data import and storage tasks. Effective web search of financial data is now within reach of AI tools, or so it seems.

In practice, because of the idiosyncratic item definitions used by some companies, it is not possible to use revenue items, some expenses, payables, fixed assets, and most items not at the higher level of aggregation. To use XBRL mechanically, it is first necessary to compile a list of items that are unique to all listed companies, and then to select from this list those items that are predictive of financial misstatement.

In the tool described here, users enter the selection criteria to retrieve XBRL content, namely the company's Central Index Key (CIK) and the year of the report required. Then, the search of pre-existing indexes identifies web locations that contain that report. In the US, such locations are the Securities and Exchange Commission´s (SEC) filings repository known as 'EDGAR'.

To access and retrieve SEC filings, this application uses the XBRL package made available as part of the R language [19]. The filings are on a standard format known as 10-K (10-Q for quarterly reports). The package extracts data from a XBRL report file and from the accompanying files that comprise its 'Discoverable Taxonomy Set'. The corresponding documents are saved, and the relevant data is put in place.

3.2 Input pre-selection, MLP topology and learning

The MLP is set to predict the trustworthiness of reports using two classes: fraudulent (manipulated, misstated) vs non-fraudulent [3][5][14]. The two separate samples used for MLP learning and the testing the resulting model performance are drawn from the Accounting and Auditing Enforcement Releases (AAER) ensuing from inquiries conducted by the SEC [3] during the 1983-2013 period. The Compustat database provides the item amounts from fraudulent and non-fraudulent instances.

The input to the MLP is the log, or the log-modulus (5), of 35 of the aggregated accounts in the retrieved 10-K reports from two successive years. Changes in relation to the preceding year are calculated and then added to the input.

When analysing attributes, financial analysts must identify the ratios involved, their location regarding industry standards and which course they are taking. To answer the first of these requirements, MLP learning and topology are, as detailed above, constructed so that ratios emerge as representations in the hidden nodes.

At the inception of MLP learning, the topology consists of 95 input nodes (35+35 items plus 24 dummies for industry groups, plus a bias); one hidden layer with 12 nodes; two output nodes with symmetrical outcomes about zero, and the conforming biases are allotted the fixed amount of 1 and -1. For all the nodes, the transfer functions used are hyperbolic tangents (that is, functions which are symmetrical about zero). Since there are 24 dummies as input, the biases of the hidden nodes, which are allotted the fixed amount of 1, should be superfluous. But as MLP connections are exposed to a severe pruning so that many of the original connections vanish during learning, the bias frequently is the only 'dummy' that subsist. As the result of training, nodes become log ratios but, if the input already is a ratio or a difference about the preceding year, there is only one remaining connection to an input, not two.

Even though the magnitudes of the two remaining connections in each node are, after learning, like each other in absolute terms, they vary through nodes. These variations, plus the relative values of connections linking the hidden nodes to the output nodes, constitutes an approximate assessment of the importance of each node for the MLP performance. In the final stage of learning, the less important of the hidden nodes is checked for pruning. If, after pruning, the observed decrease in the performance of the MLP is non-significant, then the pruning is confirmed. The result of this final stage is a sparing, conservative model with expressive hidden nodes.

Once hidden nodes acquire a ratio meaning, analysts can directly read the deviations from expectation specific to each company. Because the expected $a_{num} - a_{den}$ from (2) are modelled by the node, node output and attributes' groups are balanced, and the industry dummies actually subtract industry log-ratio standards from MLP representations, making them like the $u_{num} - u_{den}$ in (2). Such variation is, in logs, what analysts need to know when they compare a ratio with its expectation.

4 Results

This section reports on the inputs that remain after MLP pruning, the ratios produced in the hidden layers and their comparative significance for the modelling of the out-

come. Out-of-sample classifying precision is also given and contrasted with that of the Logit regression with similar datasets and the MLP remaining inputs as predictors. In this manner, formulations that use the newfound ratios as predictors are evaluated against the more usual logit formulations.

The building of the two misstatement-detecting samples uses 3,403 AAERs containing enforcement releases against 1,297 companies that manipulated financial reports. After removing cases for which no detailed financial data is available in Compustat, the database contains 1,152 releases. Manipulated reports from the same company in different years are not removed from the sample. The Enron company, for instance, was the object of six releases and all of them are included.

Two random samples of nearly 550 different cases each are drawn from the 1,152 releases to be used as learning and testing sets. These random samples are then paired with the same number of reports from companies that are not investigated during the period or declared insolvent in that year. Paired samples have some 1,100 cases each. One sample is devoted to constructing the logit and MLP models and the other sample is employed in assessing model precision. Due to the presence of missing cases, the samples available for constructing and testing models have the following frequencies:

Learning set: not misstated, 335 cases, 46 per cent.
Learning set: misstated, 398 cases, 54 per cent.
Performance testing set: not misstated, 353 cases, 46 per cent.
Performance testing set: misstated, 411 cases, 54 per cent.

When the MLP learning stages are finished, six hidden nodes remain, forming the following representations, which are ordered by the magnitude of the connection to the output node:

1. The log of the ratio of Total Liabilities to Total Assets
2. The log of the ratio of Cash and Short-Term Investments to Revenue
3. The log of the ratio of Long-Term Debt to Common Stock
4. The log of the ratio of Receivables to Common Stock
5. The log of the Change in Total Liabilities
6. The log of the ratio of Revenue to Common Stock

Five of these representations are financial ratios in log space and have two surviving input weights exhibiting similar magnitude and opposite sign. One node has only one surviving weight linked to the Change in Total Liabilities input. None of the industry-specific connections survives learning.

The MLP final topology is 8 inputs, 6 hidden nodes, and 2 symmetrical but otherwise identical outputs. Test-set precision is detailed in Table 1. Both the MLP and the Logit show a good increase in test-set precision, of more than 10 per cent compared to existing studies with big, varied samples. Imbalance in the frequency of class identification is subdued. The type II error, which is the costliest here, is greatly reduced.

When a company is presented to the tool, two sets of results are obtained, corresponding to periods $t-1$ and t. After being adjusted to become 0-1 variables, the MLP output can be broadly viewed as the probability of observing the associated input values, given that the predicted class is misstatement. Combined with the prior (prevalent) probability of misstatement, the MLP output thus becomes the (posterior) probability of misstatement given the observed values of the input variables.

Table 1. Out-of-sample misstatement detection classification results.

Frequencies	MLP	Logit
Non-misstated cases correctly classified	299 (85 per cent)	303 (86 per cent)
Non-misstated cases incorrectly classified (FP)	54 (15 per cent)	50 (14 per cent)
Misstated cases correctly classified (TP)	369 (90 per cent)	371 (90 per cent)
Misstated cases incorrectly classified	41 (10 per cent)	39 (10 per cent)
Precision: TP / (TP + FP)	87 per cent	88 per cent

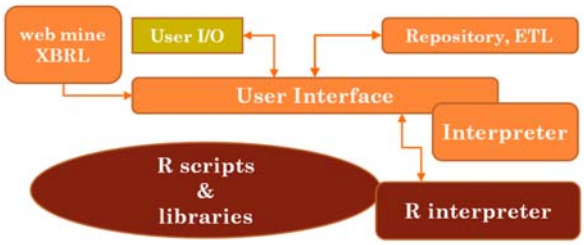

Figure 3. Modular topology of the software tool at its present state.

Once used, the tool delivers to analysts the following output:

1. The probability of the report being misstated given the observed input item amounts, with a sign indicating the direction of the change from $t-1$ to t.
2. The 3 most significant values that the internal representations assume at period t, with a sign showing the direction of change. Values are labelled as the ratio.
3. Names, period, and attributes of three companies chosen from the learning- and test-set, which are the closest to the company being analysed.

Currently, the tool uses different programming languages, libraries, and packages, as shown in Figure 3. The final standalone version is currently being built. Some necessary adaptations to the XBRL retrieval process are not fully discussed here.

5 Conclusion

Despite the wealth of research dedicated to improving the detection of misstatements, the proliferation of financial technology tools on the market has so far failed to produce software that supports web mining of published financial reports and the associated detection of misstatements. This is due to the complexity of importing the required input data and the opacity of the resulting output. The tool presented here aims to solve both problems by simplifying the web mining of the required input data and by producing understandable diagnostics. When used by analysts, the tool delivers an easy-to-understand output that not only draws attention to companies that are likely to have misstated their accounts, but also highlights, rather than obscures, the financial attributes that can support such a diagnosis.

The tool illustrates a case of close alignment between user needs and algorithmic characteristics. The tool is also an example of knowledge discovery, where explanatory variables are discovered among many candidates in order to perform a prediction task with the highest possible accuracy. The choice of the discriminating algorithm,

the MLP, was determined solely by its capability to generate expressive representations internally. Neither precision nor the demonstration of new capabilities was the aim. It is true that the out-of-sample precision achieved is more than 10 per cent better than that reported by other authors for large and diverse samples, but such a gain is achieved by using log-transformed item amounts as input variables rather than predetermined ratios. What has produced a parsimonious, balanced, and robust prediction is a proper understanding of the financial random processes involved, not algorithms.

References

1. Nigrini, M.: Forensic analytics: methods and techniques for forensic accounting. John Wiley and Sons (2011).
2. Albrecht, W., Zimbelman, M.: Fraud examination. Mason, OH South-Western (2009).
3. Dechow, C., Weill, G., Sloan, R.: Predicting material accounting misstatements, Contemporary Accounting Research 28 (1) 17-82 (2011).
4. Ngai, E., Hu, Y., Wong, Y., Chen, X., Sun. H.: The application of data mining techniques in financial fraud detection: a classification framework and an academic review of literature. Decision Support Systems 50 (3) 559-569 (2011).
5. Sharma A., Panigrahi, P.: A review of financial accounting fraud detection based on data mining techniques. International Journal of Computer Applications 39 (1) (2012).
6. Phua, K., Lee, V., Gayler, R.: A comprehensive survey of data mining-based fraud detection research. Clayton School of Information Technology, Monash University (2005).
7. Flegel, U., Vayssire, J., Bitz, G.: A state of the art survey of fraud detection technology. In Insider Threats in Cyber Security, ser. Advances in Information Security, Probst, C., Hunker, J., Gallman, A., Bishop, M. eds. Springer US 49 73-84 (2016).
8. Kirkos, E., Charalambos, S., Manolopoulos, Y.: Data mining techniques for the detection of fraudulent financial statements. Expert Systems with Applications 32 995-1003 (2013).
9. Zhou, W., Kapoor, G.: Detecting evolutionary financial statement fraud. Decision Support Systems 50 570-575 (2011).
10. Ravisankar, V. Ravi, G., Rao, R., Bose, I.: Detection of financial statement fraud and feature selection using data mining. Decision Support Systems, 50 (2) 491-500 (2011).
11. Glancy, F., Yadav, S.: A computational model for financial reporting fraud detection. Decision Support Systems 50 (3) 595-601 (2011).
12. Huang, S., Tsaih, R., Yu, F.: Topological pattern discovery and feature extraction for fraudulent financial reporting. Expert Systems with Applications 41 (9) 4360-4372 (2014).
13. Altman, E.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. The Journal of Finance 23 (4) 589-609 (1968).
14. Beneish, M.: The detection of earnings manipulation. Financial Analysts Journal 55 (5) 24-36 (1999).
15. Trigueiros, D.: Improving the effectiveness of predictors in accounting-based models. Journal of Applied Accounting Research 20 (2) 207-226 (2019).
16. Beaver, W.: Financial ratios as predictors of failure. Journal of Accounting Research, Empirical Research in Accounting: Select Studies 4 71-127 (1966).
17. John, J., Draper, N.: An alternative family of transformations. Journal of the Royal Statistical Society, Series C (Applied Statistics) 29 (2) 190-197 (1980).
18. Dunne, T., Helliar, C., Lymer, A.: Stakeholder engagement in internet financial reporting: The diffusion of XBRL in the UK. The British Accounting Review 45 (3) 167-182 (2013).
19. Bertolusso, R.: Package XBRL. CRAN project (2017).
20. Hassibi, G., Stork, D.: Optimal brain surgeon and general network pruning. In IEEE International Conference on Neural Networks (San Francisco, CA) 1 293-299 (1993).