Comput Econ (2014) 43:411–431
DOI 10.1007/s10614-013-9390-y

An Abductive-Reasoning Guide for Finance Practitioners

Rua-Haun Tsaih · Hsiou-Wei William Lin · Wen-Chyan Ke

Accepted: 15 June 2013 / Published online: 27 June 2013

Abstract This article proposes a process through which a finance practitioner's knowledge interacts with artificial intelligence (AI) models. AI models are widely applied, but how these models learn or whether they learn the right things is not easily unveiled. Extant studies especially regarding neural networks have attempted to extract reliable rules/features from AI models. However, if these models make mistakes, then the decision maker may establish paradoxical beliefs. Therefore, extracted rules/features should be justified via the prior thoughts, and vice versa. That is, with these extracted rules/features, a practitioner may need either to update his or her belief or to disregard the AI models. This study sets up a finance demonstraion for the proposed process. The proposed guide demonstrates an abductive-reasoning effect.

Keywords Abductive reasoning · Rule extraction · Neural networks · Linear/nonlinear programming

1 Background and Related Research

Suppose a practitioner has the financial domain knowledge to set up propositions and data-mining schemes to develop acceptable artificial intelligence (AI) models and then

Electronic supplementary material The online version of this article (doi:10.1007/s10614-013-9390-y) contains supplementary material, which is available to authorized users.

R.-H. Tsaih
Department of Management Information Systems, National Chengchi University, Taipei, Taiwan

H.-W. W. Lin
Department of International Business, National Taiwan University, Taipei, Taiwan

W.-C. Ke (✉)
Department of Finance and Cooperative Management, National Taipei University, Taipei, Taiwan
e-mail: wenchyan@gm.ntpu.edu.tw; wenchyan@mail.ntpu.edu.tw

to extract reliable rules/features. With the extracted rules/features, the practitioner may update his or her beliefs regarding the hypothetical features. On one hand, he or she may feel convinced of certain features that are however absent in the literature; on the other hand, the practitioner may find certain features stated in the literature unconvincingly hypothetical. This study proposes an abductive-reasoning[1] guide whereby such a practitioner can update his or her beliefs based on rules/features extracted from obtained AI models.[2]

Moreover, this proposed abductive-reasoning guide may add to the literature in the agent-based computational finance (ref. to LeBaron 2000), where the computer simulated markets with individual adaptive agents are used. Furthermore, in the artificial market, using a neural network trained with several inputs including all publicly available price information from earlier periods, agents may be equipped with capacity of price forecast. This has the impact on price dynamics, in which the researchers are usually interested, and learning alone is another interesting topic. However, to our best knowledge, there is lack of studies in exploring agents' knowledge embedded in the networks and further justifying the obtained knowledge. This study addresses this challenge.

In addition to the artificial market, our proposed guide may apply in the real market. According to the financial options history, options are traded in the market before the introduction of pricing models (Black and Scholes 1973). Option traders were eager but unable to extract features or draw conclusions from the observed option prices. Moreover, traders encounter the same problem for several well-traded exotic options. With our guide, option traders may obtain empirical comprehension before the theoretical pricing models are proposed. The popular options pricing simulation calculates the fair price without knowing the closed-form valuation formula. Given the fair price suggested by the simulation results, the investor benefits from a practice that extracts key pricing features from the observed option prices, which allows the investor to update his or her beliefs regarding the hypothetical features that are either unconvinced-but-existing or convinced-but-absent.

For convenience, let y be the explained (dependent) variable and $\mathbf{x}$ be the vector of explanatory (independent) variables. Moreover, eliminating the noise, y may be the function of $\mathbf{x}$ in general cases. Then the practitioner usually uses samples in the target area, a set of $\{\mathbf{x}\}$ space, to train AI models. Based on his or her finance expertise, the practitioner holds a list of hypothetical features; one of the listed features, for instance, is the positive second-order differential relation between the response y and the ith explanatory variable x_i embedded in the (training) sample. Because the AI model is well-known for its ability as a universal approximator, the practitioner comes to Hypothesis (1), in which y' is the output of model that approximates y:

[1] Suppose the hypothesis: *IF A, THEN B*. Peirce (1992, 2011) stated that we abduce a hypothetical explanation *A* from an observed unanticipated circumstance *B*. The reasoning is that *A* may be true because then *B* would be a matter of course.

[2] This task works in concert with the emphasis of Domingos (2007) on the importance of the process of inducing knowledge from both the practitioner and the data-mining process.

$$\begin{aligned} &\textit{If there is an embedded feature} \frac{\partial^2 y}{\partial x_i^2} > 0 \textit{ in the (training) samples of the} \\ &\textit{target area, then the acceptable AI models should present in unanimity the} \\ &\textit{feature} \frac{\partial^2 y'}{\partial x_i^2} > 0 \textit{ in the target area.} \end{aligned} \tag{1}$$

Take neural networks, a popular AI model (e.g. Andreou et al. 2006; Kiani and Kastens 2008; Kiani 2011; Reboredo et al. 2012), as an example to illustrate the abductive-reasoning guide. Existing studies in neural networks demonstrate several ways for extracting understandable rules/features from well-trained neural networks (Andrews et al. 1995; Saito and Nakano 2002; Setiono et al. 2002; Setiono and Liu 1996, 1997; Taha and Ghosh 1999; Tickle et al. 1998; Tsaih et al. 1998). For instance, let f: $\mathbf{X} \rightarrow \mathbf{Y}$ be the function of the trained network and $y' \equiv f(\mathbf{x})$. Relevant neural network studies derive the following (typical) syntax rules:

$$\text{If}(\mathbf{x} \in \text{the } k\text{th } \textbf{region}), \quad \text{then}(y' = f_k(\mathbf{x})). \tag{2}$$

The expression ($\mathbf{x} \in$ the kth **region**) is the premise for applying the rule, and y' is determined via the approximation function $f_k(\mathbf{x})$. An approximation of the activation function is usually required for extracting comprehensible rules from the trained network. That is, the activation function of each hidden node is approximated by either a piecewise linear function (Setiono et al. 2002) or a multivariate polynomial function (Saito and Nakano 2002). This approach is analogous to traditional statistical approaches such as parametric regression, which strips away nonessential details.

Adding to the literature extracting certain rules/features from well-trained neural networks, we propose a guide for implementing abductive reasoning. For instance, the abductive reasoning for the hypothetical feature $\frac{\partial^2 y}{\partial x_i^2} > 0$ is that if $\frac{\partial^2 y'}{\partial x_i^2} > 0$ has high credibility, namely, it is exhibited in all networks, then this belief $\frac{\partial^2 y}{\partial x_i^2} > 0$ is highly plausible. Specifically, given an odd number of well-trained real-valued single-hidden layer feed-forward neural networks (SLFNs) with one output node, we propose an abductive-reasoning guide whereby a finance practitioner may update his or her beliefs regarding unconvinced-but-existing or convinced-but-absent features based on extracted features from obtained SLFNs.

In the proposed guide, two designed procedures follow after obtaining acceptable SLFNs. First, the rule/feature-extracting procedure provides that, for each obtained SLFN, the area-dividing mechanism partitions the $\{\mathbf{x}\}$ space into several disjoined regions and the feature-identifying mechanism (and partitioning mechanism) specifies an exhibited feature at each disjoined region through an examination of the extracted rules. Mathematical programming analysis, rather than data analysis, identifies the exact region of each (obtained) rule premise and justifies the features exhibited at each disjoined region. Through rigorous approximation of the activation function of all hidden nodes, the designed rule/feature-extracting procedure focuses on fidelity, which describes the extent to which the extracted rules mimic the behavior of a network (Andrews et al. 1995).

Second, the concluding procedure rates the plausibility of the practitioner's belief as unconvinced-but-existing or convinced-but-absent features based on the credibility of extracted features. After the rule/feature-extraction procedure, the practitioner may find that (i) for each SLFN, the exhibited feature either conforms to or deviates from the listed feature and (ii) for different SLFNs, the division and the features exhibited in the disjoined regions may be different. A concern for imperfect learning may bother the practitioner. To address this challenge, a reasoning mechanism in the concluding procedure consolidates the exhibited features among the different disjoined regions of the target area and obtained SLFNs to update the practitioner's belief. The concluding procedure requests an accuracy that describes the extent to which the extracted rules may be generalized.

The remaining sections of this paper are organized as follows. The next section provides details of the proposed rule/feature-extracting procedure and concluding procedure. The following discussion presents the proposed abductive-reasoning guide to the pricing of debt securities because the nonlinear bond pricing may be common and more easily understood by most nonfinancial audiences than the option pricing. Finally, the conclusion offers managerial implications and suggestions for future work.

2 Proposed Abductive-Reasoning Guide

This guide is motivated by coping with several unaddressed issues in the literature extracting certain rules/features. First, certain scenarios may be described by the presumed regions that have few or no data observations. For instance, an investigation of risk factors for illiquid early-issued options may produce few observations. Second, in a single network, the exhibited feature in different disjoined regions of the target area may be distinct; however, different networks may present distinctive features in the same region. The practitioner needs a way to consolidate the exhibited features among different disjoined regions of the target area and all obtained networks.

Third, to identify the premise of a single rule, most current studies use either training data or generated data, which the trained network itself yields. Due to the countable number of (training or generated) data instances, such a data analysis covers only a limited number of points in the (presumed) region of the rule premise and thus leads to deficient generalizations.

Finally, in most finance applications such as the option pricing, the best architecture of a network is characteristically difficult to determine, and therefore a local minimum convergence in learning occurs. In addition, noises in the training samples—for example, option prices before the introduction of Black and Scholes (1973) pricing models—prevent perfect fittings within networks.

2.1 Neural Network and Pre-Requirements

The proposed guide can be applied to any problem with continuous variables that has been subjected to real-valued SLFNs. For a real-valued SLFN with the activation function *tanh(t)* in all hidden nodes and the linear activation function used in the output node, let

$$hj \equiv tanh\left(\left(\mathbf{w}_j^H\right)^{\mathrm{T}}\mathbf{x} + w_{j0}^H\right) \quad \text{and} \tag{3}$$

$$y' \equiv w_0^o + \sum_{j=1}^{p} + w_j^o hj, \tag{4}$$

where h_j denotes the activation value of the jth hidden node regarding the input $\mathbf{x} \equiv (x_1, x_2, \ldots, x_m)^{\mathrm{T}}$; m is the number of input variables; and p is the number of hidden nodes. w_{j0}^H is the bias of the jth hidden node; w_{ji}^H is the weight between the jth hidden node and the ith input node; $\mathbf{w}_j^H \equiv (w_{j1}^H, w_{j2}^H, \ldots, w_{jm}^H)^{\mathrm{T}}$ stands for the $m \times 1$ vector of weights between the jth hidden node and the input layer, with j ranged from 1 to p; w_0^o is the bias of the output node; and $\mathbf{w}^o \equiv (w_1^o, w_2^o, \ldots, w_p^o)^{\mathrm{T}}$ denotes the $p \times 1$ vector of weights between the output node and all hidden nodes. Hereafter, characters in bold represent column vectors, superscript T indicates transposition, superscript H indicates quantities related to the hidden layer, and superscript o indicates quantities related to the output layer.

Four operative requirements exist regarding the proposed abductive-reasoning guide:

- Requirements 1: The practitioner has an odd number of SLFNs, each of which is perceived as well-trained. Hereafter, a well-trained SLFN satisfies the practitioner's predetermined benchmark of the forecast performance but does not necessarily provide a globally optimal learning result.
- Requirements 2: Based on the practitioner's expertise, a list of hypothetical features accompanies each associated target area. The practitioner is interested in extracting information regarding unconvinced-but-existing or convinced-but-absent features but not in an unconvinced-and-absent feature from the obtained SLFNs. A convinced-and-existing feature, nevertheless, helps identifying whether the SLFNs are suitable for abductive reasoning.
- Requirements 3: The practitioner determines the order of the adopted piecewise polynomial approximation function according to the order of each of his or her listed feature. That is, with respect to the listed feature that depicts the nth order (partial) differential relation, the practitioner adopts a piecewise polynomial function with a maximal power of $(n + 1)$ to approximate the *tanh* function.
- Requirements 4: A (computer-generated) round-off effect exists in the tanh function and $|tanh(x)| = 1$ when $|x| > \Psi$.[3] Namely, x cannot coexist with $tanh(x) \in (-1, tanh(\Psi)) \cup (tanh(\Psi), 1)$. The constant Ψ may vary with the level of precision of the computer simulation. For instance, in our computer simulation environment with a numerical analysis package, MATLAB (Mathworks, Inc., Natick, MA) on PC, Ψ is (approximately) 19.0615.

Moreover, this study focuses on the application of the proposed guide to update the investor's beliefs regarding features that have an nth order (partial) differential relation with $n \geq 2$, such as five hedge ratios (usually called option Greeks) of the Black and Scholes (1973) formula. This specification distinguishes the study from

[3] In fact, Requirement 4 just describe a truth in the computing.

other classification studies with linear settings (Baesens et al. 2003; Setiono and Liu 1997).

With respect to the listed feature that depicts the nth order (partial) differential relation, the rule/feature-extracting procedure follows Steps 1–3, and Step 4 implements the concluding procedure:

Step 1: For each obtained SLFN, divide the target area into several disjoined regions, at each of which exists a specific multivariate polynomial of degree $(n+1)$ approximating output y'[4];

Step 2: For each obtained SLFN, identify the multivariate polynomial of degree $(n+1)$ that approximates output y' regarding the $\mathbf{x}$ at each disjoined region;

Step 3: Apply the partitioning mechanism to each obtained SLFN to further split the region exhibiting a null feature in Step 2 into two (sub-) regions, at each of which a specific feature definitely is exhibited; and

Step 4: Apply the reasoning mechanism to consolidate the identified features in each disjoined region of all SLFNs and rate the plausibility of the practitioner's belief.

Without losing the generalization and for a full demonstration, next we take the listed feature $\frac{\partial^2 y}{\partial x_i^2} > 0$, which is a positive second-order differential relation between y and the ith explanatory variable x_i, to illustrate the details of the area-dividing mechanism, the feature-identifying mechanism, the partitioning mechanism, and the reasoning mechanism.

2.2 Area-Dividing Mechanism

Because the listed feature is $\frac{\partial^2 y}{\partial x_i^2} > 0$, based on Requirements 3 and 4, we use the following piecewise polynomial function $g^{\tilde{2}}(x)$ to approximate the $tanh(x)$ function:

$$g_{\tilde{2}}(x) = \begin{cases} g_{\tilde{1}}^{\tilde{2}}(x) \equiv 1 & \text{if } x > 19.0615; \\ g_{\tilde{2}}^{\tilde{2}}(x) \equiv 0.9965 + 0.00026x & \text{if } 2.4661 \le x \le 19.0615; \\ g_{\tilde{3}}^{\tilde{2}}(x) \equiv 1.1383x - 0.4464x^2 + 0.0603x^3 & \text{if } 0 \le x \le 2.4661; \\ g_{\tilde{4}}^{\tilde{2}}(x) \equiv 1.1383x + 0.4464x^2 + 0.0603x^3 & \text{if } -2.4661 \le x \le 0; \\ g_{\tilde{5}}^{\tilde{2}}(x) \equiv -0.9965 + 0.00026x & \text{if } -19.0615 \le x \le -2.4661; \\ g_{\tilde{6}}^{\tilde{2}}(x) \equiv -1 & \text{if } x < -19.0615, \end{cases} \quad (5)$$

which is obtained from a (nonlinear) programming problem stated in (S1) of Supplement and the right superscript of $\tilde{2}$ indicates the order of the listed feature.[5]

The activation function of the jth hidden node, $tanh((\mathbf{w}_j^H)^{\mathrm{T}}\mathbf{x} + w_{j0}^H)$, is approximated with $g^{\tilde{2}}((\mathbf{w}_j^H)^T\mathbf{x} + w_{j0}^H)$, which splits the entire $\{\mathbf{x}\}$ space into six disjoined paral-

[4] Following Requirement 3 and Eq. (4), the output y' in each disjoined region is thus approximated with a comprehensible multivariate polynomial representation with a maximal power of $(n+1)$.

[5] Please refer to our supplement. It is available at SSRN: http://ssrn.com/abstract=2283696.

lel regions, as Eq. (5) shows. According to the value of $net_j \equiv (\mathbf{w}_j^H)^T\mathbf{x} + w_{j0}^H$ and Eq. (5), for example, when the value of net_j is greater than 19.0615, $g^{\tilde{\iota}^2}((\mathbf{w}_j^H)^T\mathbf{x} + w_{j0}^H)$ is expressed as $g_1^{\tilde{\iota}^2}(net_j)$ (which is set as 1).

For the jth hidden node, let the $\iota_j^{\tilde{\iota}^2}$ function defined in Eq. (6) be the index function denoting each of these six disjoined regions and $\iota^{\tilde{\iota}^2} \equiv (\iota_1^{\tilde{\iota}^2}, \iota_2^{\tilde{\iota}^2}, \ldots, \iota_p^{\tilde{\iota}^2})$ be the index vector with $\iota_j^{\tilde{\iota}^2} \in \{1, 2, 3, 4, 5, 6\}$ for every j:

$$\iota_j^{\tilde{\iota}^2} \equiv \begin{cases} 1 & \text{if } net_j > 19.0615; \\ 2 & \text{if } 2.4661 < net_j \leq 19.0615; \\ 3 & \text{if } 0 < net_j \leq 2.4661; \\ 4 & \text{if } -2.4661 < net_j \leq 0; \\ 5 & \text{if } -19.0615 \leq net_j \leq -2.4661; \\ 6 & \text{if } net_j < -19.0615. \end{cases} \tag{6}$$

Because p hidden nodes exist, the entire $\{\mathbf{x}\}$ space can be viewed as a union of 6^p regions, in which the $(\iota^{\tilde{\iota}^2})$th region is specified as $\{\mathbf{x}|\mathbf{A}_{\iota^{\tilde{\iota}^2}}^{\tilde{\iota}^2}\mathbf{x} \geq \mathbf{b}_{\iota^{\tilde{\iota}^2}}^{\tilde{\iota}^2}\}$;

$$\mathbf{A}_{\iota^{\tilde{\iota}^2}}^{\tilde{\iota}^2} \equiv \begin{bmatrix} \boldsymbol{\gamma}_{\iota_1^{\tilde{\iota}^2}} \\ \boldsymbol{\gamma}_{\iota_2^{\tilde{\iota}^2}} \\ \vdots \\ \boldsymbol{\gamma}_{\iota_p^{\tilde{\iota}^2}} \end{bmatrix} \quad \text{with } \boldsymbol{\gamma}_{\iota_j^{\tilde{\iota}^2}} \equiv \begin{cases} (\mathbf{w}_j^H)^{\mathrm{T}} & \text{if } \iota_j^{\tilde{\iota}^2} = 1; \\ \begin{bmatrix} (\mathbf{w}_j^H)^{\mathrm{T}} \\ -(\mathbf{w}_j^H)^{\mathrm{T}} \end{bmatrix} & \text{if } \iota_j^{\tilde{\iota}^2} = 2, 3, 4, \text{ or } 5; \text{ and} \\ -(\mathbf{w}_j^H)^{\mathrm{T}} & \text{if } \iota_j^{\tilde{\iota}^2} = 6. \end{cases} \tag{7}$$

$$\mathbf{b}_{\iota^{\tilde{\iota}^2}}^{\tilde{\iota}^2} \equiv \begin{bmatrix} \nu_{\iota_1^{\tilde{\iota}^2}} \\ \nu_{\iota_2^{\tilde{\iota}^2}} \\ \vdots \\ \nu_{\iota_p^{\tilde{\iota}^2}} \end{bmatrix} \quad \text{with } \nu_{\iota_j^{\tilde{\iota}^2}} \equiv \begin{cases} 19.0615 - w_{j0}^H & \text{if } \iota_j^{\tilde{\iota}^2} = 1; \\ \begin{bmatrix} 2.4661 - w_{j0}^H \\ -19.0615 + w_{j0}^H \end{bmatrix} & \text{if } \iota_j^{\tilde{\iota}^2} = 2; \\ \begin{bmatrix} -w_{j0}^H \\ -2.4661 + w_{j0}^H \end{bmatrix} & \text{if } \iota_j^{\tilde{\iota}^2} = 3; \\ \begin{bmatrix} -2.4661 - w_{j0}^H \\ w_{j0}^H \end{bmatrix} & \text{if } \iota_j^{\tilde{\iota}^2} = 4; \\ \begin{bmatrix} -19.0615 - w_{j0}^H \\ 2.4661 + w_{j0}^H \end{bmatrix} & \text{if } \iota_j^{\tilde{\iota}^2} = 5; \\ 19.0615 + w_{j0}^H & \text{if } \iota_j^{\tilde{\iota}^2} = 6. \end{cases} \tag{8}$$

For practitioners, the target area (hereafter **TA**; regarding each listed feature) may not be the entire $\{\mathbf{x}\}$ space and may be specified as $\{\mathbf{x}|\mathbf{d}(\mathbf{x}) = \mathbf{0}, \mathbf{e}(\mathbf{x}) \geq \mathbf{0}, \mathbf{x} \in \boldsymbol{\Omega}\}$, in which $\mathbf{d}(\mathbf{x}) = \mathbf{0}$ and $\mathbf{e}(\mathbf{x}) \geq \mathbf{0}$ are functional constraints, and $\mathbf{x} \in \boldsymbol{\Omega}$ is a set constraint. When the target area is not the entire $\{\mathbf{x}\}$ space, let the $(\iota^{\tilde{\iota}^2})$th (potential) region be $\{\mathbf{x}|\mathbf{A}_{\iota^{\tilde{\iota}^2}}^{\tilde{\iota}^2}\mathbf{x} \geq \mathbf{b}_{\iota^{\tilde{\iota}^2}}^{\tilde{\iota}^2}, \mathbf{x} \in \mathbf{TA}\}$. Only some of 6^p (potential) regions are extant. Based on the (linear or nonlinear) nature of functional constraints, the existence of the $(\iota^{\tilde{\iota}^2})$th

(potential) region can be determined by applying either the simplex method or a nonlinear programming technique (Luenberger 1984) to the following problem:

$$\begin{aligned} &\text{Minimize: } constant \\ &\text{Subject to: } \mathbf{A}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}\mathbf{x} \geq \mathbf{b}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}, \quad \mathbf{x} \in \mathbf{TA}. \end{aligned} \tag{9}$$

Hereafter, a disjoined region is a single (extant) region obtained by solving problem (9).

2.3 Feature-Identifying Mechanism

Given the listed feature $\frac{\partial^2 y}{\partial x_i^2} > 0$, if the $(\boldsymbol{\iota}^{\tilde{2}})$th region is extant, then the output y' in each disjoined region is approximated with a comprehensible multivariate polynomial representation as follows:

$$\text{If } \mathbf{x} \in \{\mathbf{x} | \mathbf{A}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}\mathbf{x} \geq \mathbf{b}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}, \mathbf{x} \in \mathbf{TA}\}, \quad \text{then } y' = f^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}(\mathbf{x}), \tag{10}$$

where $f^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}(\mathbf{x}) \equiv w_0^o + \sum_{j=1}^{p} w_j^o g^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}_j}((\mathbf{w}_j^H)^T\mathbf{x} + w_{j0}^H)$. The right superscript of $\tilde{2}$ and right subscript of $\boldsymbol{\iota}^{\tilde{2}}$ of $f^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}(\mathrm{x})$ indicate that the adopted approximation function is $g^{\tilde{2}}$ in the $(\boldsymbol{\iota}^{\tilde{2}})$th region. In the $(\boldsymbol{\iota}^{\tilde{2}})$th region, one of the following three relevant features can be identified:

- the $\mathrm{F}_1^{\tilde{2}}$ feature with $\left.\frac{\partial^2 y'}{\partial x_i^2}\right|_{\mathbf{x}\in\left\{\mathbf{x}|\mathbf{A}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}\mathbf{x}\geq\mathbf{b}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}},\mathbf{x}\in\mathbf{TA}\right\}} > 0,$
- the $\mathrm{F}_2^{\tilde{2}}$ feature with $\left.\frac{\partial^2 y'}{\partial x_i^2}\right|_{\mathbf{x}\in\left\{\mathbf{x}|\mathbf{A}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}\mathbf{x}\geq\mathbf{b}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}},\mathbf{x}\in\mathbf{TA}\right\}} < 0,$ or
- the $\mathrm{F}_0^{\tilde{2}}$ feature with a null result.

That is, the feature is maximized or minimized as

$$\frac{\partial^2 f^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}}{\partial x_i^2} \equiv \left.\frac{\partial^2 y'}{\partial x_i^2}\right|_{\mathbf{x}\in\left\{\mathbf{x}|\mathbf{A}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}}\mathbf{x}\geq\mathbf{b}^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}},\mathbf{x}\in\mathbf{TA}\right\}} \equiv \sum_{j=1}^{p} w_j^o \frac{\partial^2 g^{\tilde{2}}_{\boldsymbol{\iota}^{\tilde{2}}_j}((\mathbf{w}_j^H)^{\mathrm{T}}\mathbf{x} + w_{j0}^H)}{\partial x_i^2}.$$

Thus, the $\mathrm{F}_1^{\tilde{2}}$ feature is claimed at the $(\boldsymbol{\iota}^{\tilde{2}})$th region if the minimal solution to the optimization problem (11) is positive; the $\mathrm{F}_2^{\tilde{2}}$ feature is claimed if the maximal solution to the problem (12) is negative; and the $\mathrm{F}_0^{\tilde{2}}$ feature is claimed if neither a positive minimal or a negative maximal solution is obtained to problem (11) or (12), respectively.

$$\begin{aligned}
&\text{Minimize: } \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} \\
&\text{Subject to: } \mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \quad \mathbf{x} \in \mathbf{TA} \qquad (11) \\
&\text{Maximize: } \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} \\
&\text{Subject to: } \mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \quad \mathbf{x} \in \mathbf{TA}. \qquad (12)
\end{aligned}$$

2.4 Partitioning Mechanism

For any disjoined region exhibiting a null result, the following partitioning mechanism further partitions it into two (sub-) regions, each of which exhibits a definite feature. For instance, assume the $(\boldsymbol{\iota}^{\tilde{2}})$th region $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}\}$ exhibits the $\mathrm{F}_0^{\tilde{2}}$ feature regarding the listed feature $\frac{\partial^2 y}{\partial x_i^2} > 0$. With Requirement 3, the maximal power of the polynomial representation of $f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}$ is 3. Thus, $\frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2}$ is either a constant or a linear function. A null result is impossible in the $(\boldsymbol{\iota}^{\tilde{2}})$th region when $\frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2}$ is a constant. However, when $\frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2}$ is a linear function, the region $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}\}$ can be treated as a union of the following three disjoined (sub-) regions: $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} < 0\}$, $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} > 0\}$, and $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} = 0\}$. Note that the $\mathrm{F}_1^{\tilde{2}}$ feature appears at $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} > 0\}$ and the $\mathrm{F}_2^{\tilde{2}}$ feature appears at $\{\mathbf{x}|\mathbf{A}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}\mathbf{x} \geq \mathbf{b}_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f_{\boldsymbol{\iota}^{\tilde{2}}}^{\tilde{2}}}{\partial x_i^2} < 0\}$.

2.5 Reasoning Mechanism

Given Requirement 1, the practitioner should have an odd number of well-trained SLFNs serving as a stabilization measure to the conclusion of extracted features. For each obtained SLFN, the exhibited feature $\frac{\partial^2 y'}{\partial x_i^2}$ in each disjoined region either conforms to or deviates from the listed feature $\frac{\partial^2 y}{\partial x_i^2} > 0$. To consolidate the exhibited features $\frac{\partial^2 y'}{\partial x_i^2}$ among all obtained SLFNs, the following three types of areas of each listed feature are first identified:

- The consistent area (CA): the union of areas in which all obtained SLFNs exhibit the conforming feature without exception.
- The accordant inconsistent area (AIA): the union of areas in which all obtained SLFNs exhibit the deviated feature without exception.
- The discordant area (DA): the union of areas in which some SLFNs exhibit the conforming feature and the others exhibit the deviated feature.

After identifying these three different areas, we consolidate the exhibited features among all obtained SLFNs as follows:

1. Strongly conclude that the feature $\frac{\partial^2 y'}{\partial x_i^2} > 0$ appears in each CA;
2. Strongly conclude that the feature $\frac{\partial^2 y'}{\partial x_i^2} < 0$ appears in each AIA; and
3. For each DA, exhaustively count the number of obtained SLFNs with the exhibited features $\tilde{F_1^2}$ and $\tilde{F_2^2}$ and then use the percentage of these two numbers to derive a (credible) feature. Accordingly, the consolidating scenarios are as follows:
   (i) Conclude that the feature $\frac{\partial^2 y'}{\partial x_i^2} > 0$ appears in the DA if its derived (credible) feature is $\tilde{F_1^2}$.
   (ii) Conclude that the feature $\frac{\partial^2 y'}{\partial x_i^2} < 0$ appears in the DA if its derived (credible) feature is $\tilde{F_2^2}$.

Then, we rate the plausibility of the prior belief regarding the listed feature $\frac{\partial^2 y}{\partial x_i^2} > 0$ in its associated entire target areas as follows:

1. If the CA equals the entire target area, the practitioner's prior belief is rated as having the highest plausibility.
2. If the AIA and the DA do not exhibit the derived feature $\tilde{F_2^2}$, the outcome provides weak support of the practitioner's prior belief, which is rated as having high plausibility.
3. If the AIA does not equal the entire target area or the DA contains the derived feature $\tilde{F_2^2}$, the prior belief is rated as having low plausibility.
4. If the AIA equals the entire target area, the practitioner's prior belief is rated as having the lowest plausibility.

Finally, if the convinced-and-existing feature is rated with the highest (or high) plausibility, the practitioner may claim that the adopted approximation function is acceptable. Otherwise, the practitioner may claim either that the adopted approximation function g is unacceptable or that the obtained SLFNs are not useful in the abductive-reasoning exercise.

2.6 Derivation of the First-Order Features

We take the listed feature $\frac{\partial y}{\partial x_i} > 0$ to illustrate. With this listed feature, the same steps in the rule/feature-extracting procedure are applied, except that the adopted

approximation function is $g^{\tilde{1}}$, as the following, instead of $g^{\tilde{2}}$.[6] Furthermore, $F_0^{\tilde{1}}$, $F_1^{\tilde{1}}$, and $F_2^{\tilde{1}}$ denote the counterparts corresponding to $F_0^{\tilde{2}}$, $F_1^{\tilde{2}}$, and $F_2^{\tilde{2}}$, respectively.

$$g^{\tilde{1}}(x) \equiv \begin{cases} g_1^{\tilde{1}}(x) \equiv 1 & \text{if } x > 19.0615; \\ g_2^{\tilde{1}}(x) \equiv 0.9908 + 0.00067x & \text{if } 1.9643 \le x \le 19.0615; \\ g_3^{\tilde{1}}(x) \equiv 1.0098x - 0.2569x^2 & \text{if } 0 \le x \le 1.9643; \\ g_4^{\tilde{1}}(x) \equiv 1.0098x + 0.2569x^2 & \text{if } -1.9643 \le x \le 0; \\ g_5^{\tilde{1}}(x) \equiv -0.9908 + 0.00067x & \text{if } -19.0615 \le x \le -1.9643; \\ g_6^{\tilde{1}}(x) \equiv -1 & \text{if } x < -19.0615, \end{cases} \tag{13}$$

3 Demonstration of the Abductive-Reasoning Guide

In the section, we train the networks with the data set of Treasury bond prices for the demonstration. In Treasury bond market (hereafter, "Treasury" is ignored), the prices are almost dominated by the sum of the discounted cash flows of bonds, which is a well-defined formula. Therefore, we use the simulated bond prices instead of the historical bond prices. That makes us to demonstration the abductive-reasoning effect in a well-control environment, which is almost the same as the real market. To confirm this point, we also examine the performance of the trained network using the historical bond prices from the website of The Wall Street Journal.

3.1 Design of Demonstration

To simulate the set of data that may be observed by a representative practitioner who knows the bond-pricing mechanism well (but less than perfectly), we generate and use garbled training samples of bond price $y_t = p_t + \varepsilon_t$. p_t is the theoretic value of the bond at time t and is derived from Eq. (14), which serves as an example of complete domain knowledge with respect to the bond pricing model, and ε_t is a white error term provided by a normal random number generator of $N(0, (0.2)^2)$. Namely, y_t is perturbed by a white noise.

$$p_t \equiv \sum_{k=1}^{T_0} \frac{C}{(1+r_t)^{k-t}} + \frac{F}{(1+r_t)^{T_0-t}}. \tag{14}$$

According to Eq. (14), p_t is determined by (i) r_t, the market rate of interest at time t; (ii) F, the face value of the bond, which generally equals 100; (iii) T_0, the term to maturity at time $t = 0$; and (iv) C, the periodic coupon payment, which equals $F \times r_c$. Table 1 shows the 18 hypothetical combinations of term to maturity and contractual interest rate that we use to generate a set of price data with $t = 1/80, 2/80, \ldots, 80/80$ through Eq. (14). The rate r_t is derived from a normal random number generator of $N(2\,\%, (0.1\,\%)^2)$. Accordingly, we have 1,440 training samples with input variables

[6] Please refer to our supplement. It is available at SSRN: http://ssrn.com/abstract=2283696.

Table 1 The 18 hypothetical bonds with different combinations of term to maturity and contractual interest rate

Bond no.	Term to maturity (T_0)	Contractual interest rate (r_c) (%)	Bond no.	Term to maturity (T_0)	Contractual interest rate (r_c) (%)	Bond no.	Term to maturity (T_0)	Contractual interest rate (r_c) (%)
1	2	0.0	7	2	1.5	13	2	3.0
2	4	0.0	8	4	1.5	14	4	3.0
3	7	0.0	9	7	1.5	15	7	3.0
4	10	0.0	10	10	1.5	16	10	3.0
5	15	0.0	11	15	1.5	17	15	3.0
6	20	0.0	12	20	1.5	18	20	3.0

The contractual coupon payments are assumed to be made annually

T_t, r_c, and r_t, and the desired output variable y_t, where $T_t \equiv (T_0 - t)$ is the term to maturity at time t. The range of the sample training data is

$$\{\mathbf{x} \equiv (T_t, r_c, r_t)^T | (1 \le T_t \le 20), (0 \le r_c \le 0.030), \text{ and } (0.017 \le r_t \le 0.024)\}.$$

To examine the generalization of trained networks, we also adopt 1,440 test samples by similar means except that T_0, t, r_c, and r_t are randomly and independently generated from {1, 2, …, 20} with a probability of 5 % for each, {1/80, 2/80, …, 80/80} with a probability of 1.25 % for each, [0.0, 3.0 %] with a probability density function $f(r_c) = 1/0.03$, and $N(2\,\%, (0.1\,\%)^2)$, respectively. This setting results in varying instances among the test samples.

We, as the representative practitioner, adopt the back-propagation learning algorithm (Rumelhart et al. 1986) to train 1,000 SLFNs, each of which has four hidden nodes and random initial weights and biases. Among the 1,000 SLFNs, we pick the three with the smallest mean square error (MSE) for the test samples. Table 2 shows the (final) weights and biases of these three SLFNs, hereafter named Networks I, II and III. The corresponding MSEs for the training samples are 0.414, 0.404, and 0.451, respectively; and the corresponding MSEs for the test samples are 0.429, 0.432, and 0.445, respectively. The average absolute deviation is approximately 0.6, which deviates from the specified error term standard deviation of 0.2. The pricing error is unrelated to theoretic prices p_t but related to observed prices y_t.

We obtain the U.S. Treasury Quotes from the website of The Wall Street Journal for the first trading day of each month in 2010. There are 2,462 Treasury bond quotes and we further select the 141 bond quotes, which are similar to the aforementioned settings. Then, we obtain the estimated prices from Network II. We plot the real prices and estimated prices in Fig. 1a. It shows that the Network II works well for the real historical bond prices. In fact, in Fig. 1b, Network II also displays a positive relationship between real and estimated prices for all 2,462 observations, including the ones not meeting the simulated settings.

Assume that we are interested in shorter-termed premium bonds with ($1 \le T_t \le 10$) and ($r_c > r_t$), and with the following five (listed) features: (i) $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$; (ii)

Table 2 Final weights and biases of Networks I, II and III

Network	Weights and biases						
	w_0^o	j	w_{j0}^H	w_j^o	w_{j1}^H	w_{j2}^H	w_{j3}^H
I	100.4744	1	−0.1689	15.1206	−0.0347	−32.7223	18.8396
		2	−1.3535	−34.3660	0.0544	−36.8286	28.9551
		3	−2.1615	5.6589	0.0988	43.3354	16.8267
		4	1.1698	−21.9999	−0.0643	−36.4188	53.6646
II	93.6583	1	0.4510	−23.3874	−0.0571	−33.4648	71.8090
		2	0.8413	36.9871	−0.0467	32.9078	−10.5926
		3	−1.1572	−10.4621	0.0699	45.5792	43.2855
		4	1.2874	−9.2684	−0.0458	17.5685	−87.3147
III	104.8248	1	0.7832	−14.0352	−0.0519	−27.1299	49.3836
		2	1.3108	−16.7297	−0.0571	−27.3748	14.5874
		3	−1.5287	−30.1819	0.0631	−37.1108	33.3026
		4	−0.6010	13.1504	−0.0524	−34.9042	36.9149

w_0^o denotes the bias of the output node; w_{j0}^H denotes the bias of the jth hidden node; w_j^o denotes the weight between the output node and the jth hidden node; and w_{ji}^H denotes the weight between the jth hidden node and the ith input node

$\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$; (iii) $\frac{\partial^2 y_t}{\partial r_t^2} > 0$; (iv) $\frac{\partial y_t}{\partial T_t} > 0$; and (v) $\frac{\partial y_t}{\partial r_t} < 0$. Their associated target areas are the same:

$$\mathbf{TA} = \{\mathbf{x} \equiv (T_t, r_c, r_t)^T | (1 \le T_t \le 10), (r_c > r_t), (0 \le r_c \le 0.030), \text{ and } (0.017 \le r_t \le 0.024)\}. \tag{15}$$

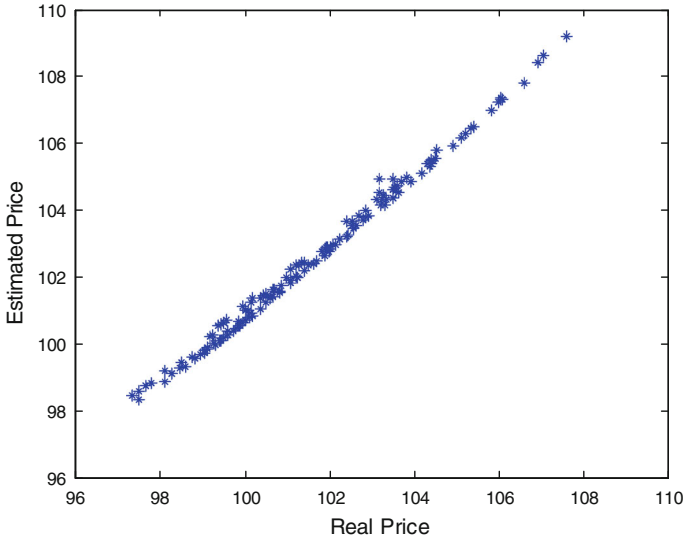
Note that this target area only covers 320 of the 1,440 training samples, which means that some of disjoined regions obtained from the rule-extracted mechanism may have few or no observations.

To illustrate the complexity of extracting features from the obtained SLFNs, we adopt the preimage analysis proposed by Tsaih and Wan (2009) for Network II and its preimage-related properties.[7] The preimage $f^{-1}(y') = \{\mathbf{x} | f(\mathbf{x}) = y'\}$ is the set in the $\{\mathbf{x}\}$ space for a given y'. As shown in Fig. 2, the preimage $f^{-1}(y')$ consists of one (or several) complex two-manifold segments in $\Re^3$.[8]

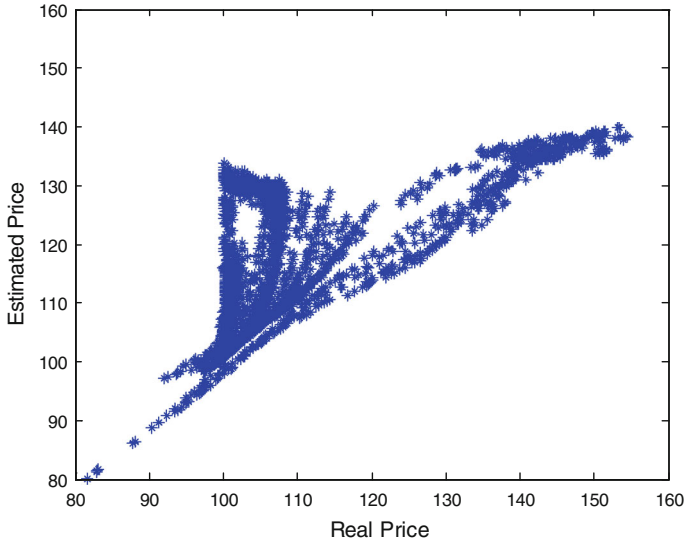
Before extracting rules/features, we also examine the effect of the approximation functions in the Fig. 3. It appears that the approximation function has the same performance as the SLFN for the bond pricing.

[7] Please refer to our supplement. It is available at SSRN: http://ssrn.com/abstract=2283696.

[8] A p-manifold is a Hausdorff space $\mathbf{X}$ with a countable basis such that each point x of $\mathbf{X}$ has a neighborhood that is homomorphic with an open subset of $\Re^p$ (Munkres 1975). A one-manifold is often called a curve, and a two-manifold is called a surface.



(a)



(b)

Fig. 1 The performance of Network II for the history bond prices from the website of *The Wall Street Journal*. The 144 in sample observations are in (**a**) and the all 2,462 bond quotes are in (**b**)

3.2 Illustrating the Rule/Feature-Extracting Procedure

Without losing the generalization and for a full demonstration, we take the listed feature $\frac{\partial^2 y_t}{\partial r_c \partial r_t} > 0$ and Network II to demonstrate the rule/feature-extracting procedure.

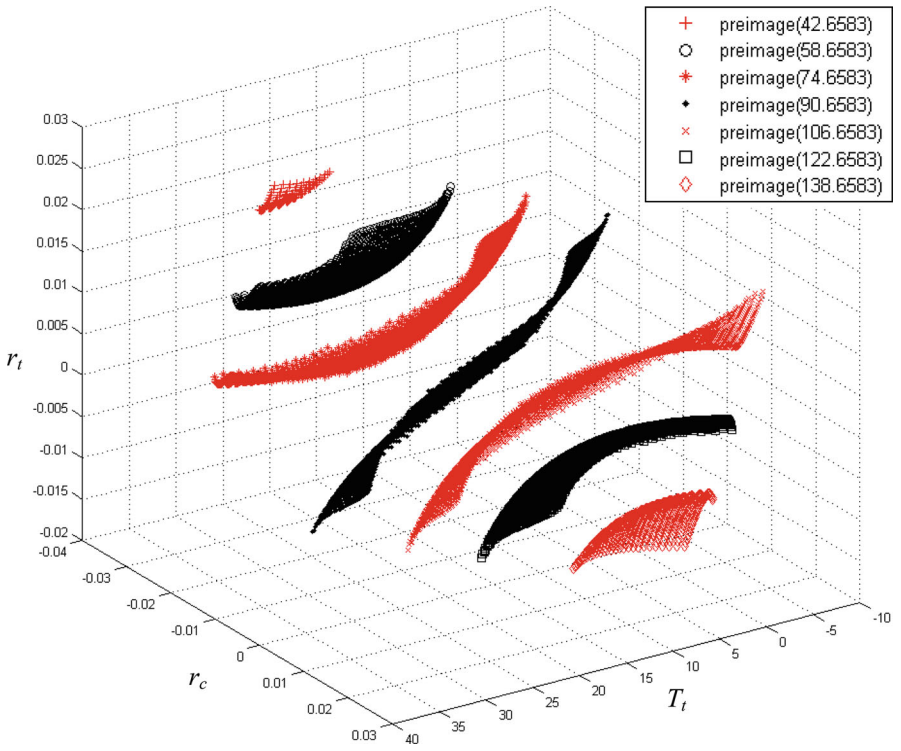


Fig. 2 The preimage graph of Network II. The legend shows the values of y' in parentheses

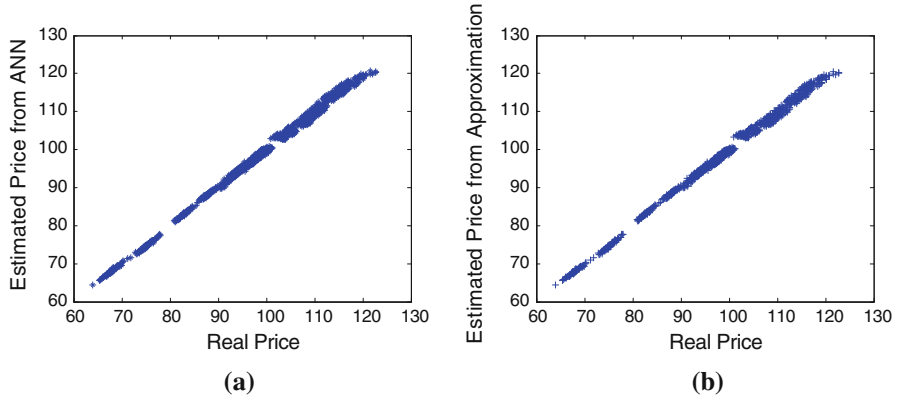


Fig. 3 The performance of the approximation function $g^{\tilde{2}}$. **a** is the scatter of SLFN estimates and the simulated prices; **b** is the scatter of approximated estimates and the prices. Both display similar performances

To do this, we solve optimization problem (16) to examine whether the (ι^2)th (potential) region regarding Network II is extant.

$$\begin{aligned} &\text{Minimize: } constant \\ &\text{Subject to: } \mathbf{A}^{\tilde{2}}_{\iota^{\tilde{2}}}\mathbf{x} \geq \mathbf{b}^{\tilde{2}}_{\iota^{\tilde{2}}}, \mathbf{x} \equiv \mathbf{TA}, \end{aligned} \tag{16}$$

in which, according to Eqs. (10) and (11), $\mathbf{A}^{\tilde{2}}_{\iota^{\tilde{2}}}$ and $\mathbf{b}^{\tilde{2}}_{\iota^{\tilde{2}}}$ can be derived from Table 2. For instance, regarding the (3, 3, 3, 4) region of the Network II,

$$\mathbf{A}^{\tilde{2}}_{(3,3,3,4)} = \begin{bmatrix} -0.0571 & -33.4648 & 71.8090 \\ 0.0571 & 33.4648 & -71.8090 \\ -0.0467 & 32.9078 & -10.5926 \\ 0.0467 & -32.9078 & 10.5926 \\ 0.0699 & 45.5792 & 43.2855 \\ -0.0699 & -45.5792 & -43.2855 \\ -0.0458 & 17.5685 & -87.3147 \\ 0.0458 & -17.5685 & 87.3147 \end{bmatrix} \text{ and } \mathbf{b}^{\tilde{2}}_{(3,3,3,4)} = \begin{bmatrix} -0.4510 \\ -2.0152 \\ -0.8413 \\ -1.6248 \\ 1.1572 \\ -3.6234 \\ -3.7536 \\ 1.2874 \end{bmatrix}$$

The output y' associated with the $(\iota^{\tilde{2}})$th region regarding Network II is approximated with $f^{\tilde{2}}_{\iota^{\tilde{2}}}(\mathbf{x}) \equiv 93.6583 - 23.3874\, g^{\tilde{2}}_{\iota^{\tilde{2}}_1}(net_1) + 36.9871 g^{\tilde{2}}_{\iota^{\tilde{2}}_2}(net_2) - 10.4621 g^{\tilde{2}}_{\iota^{\tilde{2}}_3}(net_3) - 9.2684\, g^{\tilde{2}}_{\iota^{\tilde{2}}_4}(net_4)$. Then, we solve problems (13) and (14) to verify the exhibited feature at the $(\iota^{\tilde{2}})$th disjoined region:

$$\begin{aligned} &\text{Minimize: } \frac{\partial^2 f^{\tilde{2}}_{\iota^{\tilde{2}}}}{\partial r_c \partial r_t} \\ &\text{Subject to: } \mathbf{A}^{\tilde{2}}_{\iota^{\tilde{2}}}\mathbf{x} \geq \mathbf{b}^{\tilde{2}}_{\iota^{\tilde{2}}}, \mathbf{x} \in \mathbf{TA}; \end{aligned} \tag{17}$$

$$\begin{aligned} &\text{Maximize: } \frac{\partial^2 f^{\tilde{2}}_{\iota^{\tilde{2}}}}{\partial r_c \partial r_t} \\ &\text{Subject to: } \mathbf{A}^{\tilde{2}}_{\iota^{\tilde{2}}}\mathbf{x} \geq \mathbf{b}^{\tilde{2}}_{\iota^{\tilde{2}}}, \mathbf{x} \in \mathbf{TA}. \end{aligned} \tag{18}$$

At this stage in the process, this rule/feature-extracting procedure concludes for a listed feature if no null results appear. If one or more null results occur, we then continue to apply the partitioning mechanism to these results. For instance, numerically, a null result appears in the (3, 3, 3, 4) region of the Network II. Accordingly, we further partition the (3, 3, 3, 4) region into the following two sub-regions that exhibit $\mathrm{F}^{\tilde{2}}_1$ and $\mathrm{F}^{\tilde{2}}_2$ features, respectively: $\{\mathbf{x} | \mathbf{A}^{\tilde{2}}_{(3,3,3,4)}\mathbf{x} \geq \mathbf{b}^{\tilde{2}}_{(3,3,3,4)}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f^{\tilde{2}}_{(3,3,3,4)}}{\partial r_c \partial r_t} > 0\}$ and $\{\mathbf{x} | \mathbf{A}^{\tilde{2}}_{(3,3,3,4)}\mathbf{x} \in \mathbf{b}^{\tilde{2}}_{(3,3,3,4)}, \mathbf{x} \in \mathbf{TA}, \frac{\partial^2 f^{\tilde{2}}_{(3,3,3,4)}}{\partial r_c \partial r_t} < 0\}$.

Table 3 shows the results of applying the rule/feature-extracting produce to listed features of $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$, $\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$, and $\frac{\partial^2 y_t}{\partial r_t^2} < 0$ in terms of Networks I, II, and III. In Table 3 and hereafter, $\mathrm{R}^n_{i,\iota^{\tilde{2}}}$ denotes the $(\iota^{\tilde{2}})$th region regarding Network i, in which $i \in \{\mathrm{I}, \mathrm{II}, \mathrm{III}\}$, with the right superscript n indicating the order of the listed feature. We divide the target region of the listed feature $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$ in Network I into two disjoined regions, each of which exhibits the same feature $\frac{\partial^2 y'}{\partial T_t \partial r_c} > 0$. We also divide the target region of the listed feature $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$ in Network II

Table 3 The rule/feature-extracting results

Network	$\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$		$\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$		$\frac{\partial^2 y_t}{\partial r_t^2} < 0$	
	Region	Exhibited feature	Region	Exhibited feature	Region	Exhibited feature
I	$R^{\tilde{2}}_{I,(4,4,3,3)}$	$F^{\tilde{2}}_1$	$R^{\{\tilde{2}\}}_{I,(4,4,3,3)}$	$F^{\tilde{2}}_2$	$R^{\tilde{2}}_{I,(4,4,3,3)}$	$F^{\tilde{2}}_2$
	$R^{\tilde{2}}_{I,(4,4,4,3)}$	$F^{\tilde{2}}_1$	$R^{\tilde{2}}_{I,(4,4,4,3)}$	$F^{\tilde{2}}_2$	$R^{\tilde{2}}_{I,(4,4,4,3)}$	$F^{\tilde{2}}_2$
II	$R^{\tilde{2}}_{II,(3,3,3,3)}$	$F^{\tilde{2}}_1$	$R^{\tilde{2}}_{II,(3,3,3,3)}$	$F^{\tilde{2}}_2$	$R^{\tilde{2}}_{II,(3,3,3,3)}$	$F^{\tilde{2}}_2$
	$R^{\tilde{2}}_{II,(3,3,3,4)}$	$F^{\tilde{2}}_1$	$R^{\tilde{2}}_{II,(3,3,3,4)}\Big\vert_{\frac{\partial^2 y'}{\partial r_c \partial r_t}>0}$	$F^{\tilde{2}}_1$	$R^{\tilde{2}}_{II,(3,3,3,4)}$	$F^{\tilde{2}}_2$
			$R^{\tilde{2}}_{II,(3,3,3,4)}\Big\vert_{\frac{\partial^2 y'}{\partial r_c \partial r_t}<0}$	$F^{\tilde{2}}_2$		
III	$R^{\tilde{2}}_{III,(3,3,4,4)}$	$F^{\tilde{2}}_1$	$R^{\tilde{2}}_{III,(3,3,4,4)}$	$F^{\tilde{2}}_2$	$R^{\tilde{2}}_{III,(3,3,4,4)}$	$F^{\tilde{2}}_2$

This table provides the results of applying the rule/feature-extracting procedure to listed features of $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$, $\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$, and $\frac{\partial^2 y_t}{\partial r_t^2} < 0$. The right superscript of $\tilde{2}$ corresponds to the second-order differential relation depicted in the listed feature. For region, $R^{\tilde{2}}_{i,\iota^{\tilde{2}}}$ denotes the $(\iota^{\tilde{2}})$th sub-region regarding network I, in which $i \in \{\text{I, II, III}\}$. $R^{\tilde{2}}_{i,\iota^{\tilde{2}}}|_{\text{a given feature}}$ denotes the sub-region of $R^{\tilde{2}}_{i,\iota^{\tilde{2}}}$ that exhibits the given feature. For the exhibited feature, $F^{\tilde{2}}_1$ and $F^{\tilde{2}}_2$ denote the exhibited feature in the region. Right subscripts 1 and 2 indicate a positive or a negative differential relation, respectively

into two disjoined regions, each of which exhibits the same feature $\frac{\partial^2 y'}{\partial T_t \partial r_c} > 0$. The target region of the listed feature $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$ in Network III has only one (disjoined) region that exhibits the feature $\frac{\partial^2 y'}{\partial T_t \partial r_c} > 0$. Similar summarization can be applied to the target region of the listed feature $\frac{\partial^2 y_t}{\partial r_t^2} < 0$ for Networks I, II, and III.

For the listed feature $\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$, we divide the target region in Network I into two disjoined regions, each of which exhibits the same feature $\frac{\partial^2 y'}{\partial r_t \partial r_c} < 0$. We divide the target region in Network II into three disjoint regions, in which $R^{\tilde{2}}_{II,(3,3,3,4)}\Big\vert_{\frac{\partial^2 y'}{\partial r_c \partial r_t}>0}$ and $R^{\tilde{2}}_{II,(3,3,3,4)}\Big\vert_{\frac{\partial^2 y'}{\partial r_c \partial r_t}<0}$ denotes the two sub-regions of $R^{\tilde{2}}_{II,(3,3,3,4)}$, exhibiting $\frac{\partial^2 y'}{\partial r_t \partial r_c} > 0$ and $\frac{\partial^2 y'}{\partial r_t \partial r_c} < 0$, respectively. The target region in Network III has only one (disjoined) region that exhibits the feature $\frac{\partial^2 y'}{\partial r_t \partial r_c} < 0$.

Table 4 shows the results of applying the rule/feature-extracting produce to the listed features $\frac{\partial y_t}{\partial T_t} > 0$ and $\frac{\partial y_t}{\partial r_t} < 0$ in terms of Networks I, II, and III. We divide the target region of the listed feature $\frac{\partial y_t}{\partial T_t} > 0$ in Network I into three disjoined regions, two of which exhibit the feature $\frac{\partial y'}{\partial T_t} > 0$. We divide the target region of the listed feature $\frac{\partial y_t}{\partial T_t} > 0$ in Network II into four disjoined regions, two of which exhibits the same feature $\frac{\partial y'}{\partial T_t} > 0$. Finally, we divide the target region of the listed feature

Table 4 The rule/feature-extracting results to listed features of $\frac{\partial y_t}{\partial T_t} > 0$ and $\frac{\partial y_t}{\partial r_t} < 0$

Network	$\frac{\partial y_t}{\partial T_t} > 0$		$\frac{\partial y_t}{\partial r_t} < 0$	
	Region	Exhibited feature	Region	Exhibited feature
I	$R^{\tilde{I}}_{I,(4,4,3,3)}$	$F^{\tilde{I}}_1$	$R^{\tilde{I}}_{I,(4,4,3,3)}$	$F^{\tilde{I}}_2$
	$R^{\tilde{I}}_{I,(4,4,4,3)}\Big\vert_{\frac{\partial y'}{\partial T_t}>0}$	$F^{\tilde{I}}_1$	$R^{\tilde{I}}_{I,(4,4,4,3)}$	$F^{\tilde{I}}_2$
	$R^{\tilde{I}}_{I,(4,4,4,3)}\Big\vert_{\frac{\partial y'}{\partial T_t}<0}$	$F^{\tilde{I}}_2$		
II	$R^{\tilde{I}}_{II,(3,3,3,3)}\Big\vert_{\frac{\partial y'}{\partial T_t}>0}$	$F^{\tilde{I}}_1$	$R^{\tilde{I}}_{II,(3,3,3,3)}$	$F^{\tilde{I}}_2$
	$R^{\tilde{I}}_{II,(3,3,3,3)}\Big\vert_{\frac{\partial y'}{\partial T_t}<0}$	$F^{\tilde{I}}_2$		
	$R^{\tilde{I}}_{II,(3,3,3,4)}\Big\vert_{\frac{\partial y'}{\partial T_t}>0}$	$F^{\tilde{I}}_1$	$R^{\tilde{I}}_{II,(3,3,3,4)}$	$F^{\tilde{I}}_2$
	$R^{\tilde{I}}_{II,(3,3,3,4)}\Big\vert_{\frac{\partial y'}{\partial T_t}<0}$	$F^{\tilde{I}}_2$		
III	$R^{\tilde{I}}_{III,(3,3,4,4)}\Big\vert_{\frac{\partial y'}{\partial T_t}>0}$	$F^{\tilde{I}}_1$	$R^{\tilde{I}}_{III,(3,3,4,4)}$	$F^{\tilde{I}}_2$
	$R^{\tilde{I}}_{III,(3,3,4,4)}\Big\vert_{\frac{\partial y'}{\partial T_t}<0}$	$F^{\tilde{I}}_2$		
	$R^{\tilde{I}}_{III,(3,3,5,4)}$	$F^{\tilde{I}}_1$	$R^{\tilde{I}}_{III,(3,3,5,4)}$	$F^{\tilde{I}}_2$

This table provides the results of applying the rule/feature-extracting procedure to listed features of $\frac{\partial y_t}{\partial T_t} > 0$ and $\frac{\partial y_t}{\partial r_t} < 0$. The right superscript of 1 corresponds to the first-order differential relation depicted in the listed feature. For region,$R^{\tilde{I}}_{i,\iota^{\tilde{I}}}$ denotes the $(\iota^{\tilde{I}})$th region with respect to Network I, in which $i \in \{I, II, III\}$. $R^{\tilde{I}}_{i,\iota^{\tilde{I}}}\Big\vert_{\text{a given feature}}$ denotes the sub-region of $R^{\tilde{I}}_{i,\iota^{\tilde{I}}}$ that exhibits the given feature. For exhibited feature, $F^{\tilde{I}}_1$ and $F^{\tilde{I}}_2$ denote the exhibited feature in the region. Right subscripts 1 and 2 indicate a positive or a negative differential relation, respectively

$\frac{\partial y_t}{\partial T_t} > 0$ in Network III into three disjoined regions, two of which exhibit the feature $\frac{\partial y'}{\partial T_t} > 0$. For the listed feature $\frac{\partial y_t}{\partial r_t} < 0$, each of the target regions in Networks I, II, and III is divided into two disjoined regions, all of which exhibit the feature $\frac{\partial y'}{\partial r_t} < 0$.

3.3 Illustrating the Reasoning Mechanism

With well-established first and second-order features $\frac{\partial y_t}{\partial r_t} < 0$ and $\frac{\partial^2 y_t}{\partial r_t^2} > 0$, the practitioner can examine the approximation functions $g^{\tilde{I}}$ and $g^{\tilde{2}}$, separately. According to Tables 3 and 4, the CA for each of the listed features $\frac{\partial y_t}{\partial r_t} < 0$ and $\frac{\partial^2 y_t}{\partial r_t^2} > 0$ equals the entire target area, and thus they both rate as the highest plausibility. Accordingly, the practitioner may conclude that the obtained SLFNs and the adopted $g^{\tilde{I}}$ and $g^{\tilde{2}}$ approximation functions are suitable for abductive reasoning.

Table 3 shows that the CA regarding the listed feature $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$ equals the entire target area. Accordingly, these three networks strongly support the practitioner's prior belief of $\frac{\partial^2 y_t}{\partial T_t \partial r_c} > 0$. Namely, the experiment leads to the highest plausibility rating for the practitioner's belief that longer termed premium bonds have a higher premium.

However, for the listed feature $\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$, the CA equals $\mathrm{R}^{\tilde{2}}_{\mathrm{II},(3,3,3,3)} \cup \mathrm{R}^{\tilde{2}}_{\mathrm{II},(3,3,3,4)}\big|_{\frac{\partial^2 y'}{\partial r_c \partial r_t}<0}$ and the DA equals $\mathrm{R}^{\tilde{2}}_{\mathrm{II},(3,3,3,4)}\big|_{\frac{\partial^2 y'}{\partial r_c \partial r_t}>0}$, in which two-thirds of the votes support the high credibility of feature $\frac{\partial^2 y'}{\partial r_t \partial r_c} < 0$, suggesting that the prior belief $\frac{\partial^2 y_t}{\partial r_t \partial r_c} < 0$ is highly plausible in this DA. Namely, the practitioner's belief that a bond price decrease that is accompanied by an increase in the market yield will be greater if the coupon rate is higher rates as high plausibility in its associated target area.

An interesting observation in Table 4 is that the AIA of the listed feature $\frac{\partial y_t}{\partial T_t} > 0$ equals $\mathrm{R}^{\tilde{\mathrm{I}}}_{\mathrm{I},(4,4,4,3)}\Big|_{\frac{\partial y'}{\partial T_t}<0} \cap \left(\mathrm{R}^{\tilde{\mathrm{I}}}_{\mathrm{II},(3,3,3,3)}\Big|_{\frac{\partial y'}{\partial T_t}<0} \cup \mathrm{R}^{\tilde{\mathrm{I}}}_{\mathrm{II},(3,3,3,4)}\Big|_{\frac{\partial y'}{\partial T_t}<0} \right) \cap \mathrm{R}^{\tilde{\mathrm{I}}}_{\mathrm{III},(3,3,4,4)}\Big|_{\frac{\partial y'}{\partial T_t}<0}$. Such an AIA rates with low plausibility the prior belief $\frac{\partial y_t}{\partial T_t}\Big|_{\mathbf{x}\in\{\mathbf{x}|r_c>r_t\}} > 0$, which is derived from the perspective of the literature on bonds. $\frac{\partial y_t}{\partial T_t}\Big|_{\mathbf{x}\in\{\mathbf{x}|r_c>r_t\}} > 0$ states that if bond yield remains constant over its life, then the magnitude of the premium will decrease as its term to maturity becomes shorter. The low plausibility for the prior belief of $\frac{\partial y_t}{\partial T_t}\Big|_{\mathbf{x}\in\{\mathbf{x}|r_c>r_t\}} > 0$ leads to an examination of this listed feature and the obtained SLFNs. We find that the theoretical bond-pricing function is a segment function with a boundary $\{\mathbf{x}|r_c = r_t\}$. Specifically, $\frac{\partial y_t}{\partial T_t}$ is positive for premium bonds $(r_c > r_t)$ and negative for discount bonds $(r_c < r_t)$. Also, using a single SLFN to learn simultaneously and successfully two different functions across the boundary line is prohibitively difficult. With further numerical examinations, we find that, regarding these three obtained SLFNs, the AIA of $\frac{\partial y'}{\partial T_t}\Big|_{\mathbf{x}\in\{\mathbf{x}|r_c-r_t>0.00175\}} > 0$ is null.

4 Conclusion and Managerial Implications

This study adds to the literature by introducing an abductive-reasoning guide for finance applications with SLFNs. With the proposed procedure, the practitioner may gain insights from obtained SLFNs for any unexplored data without knowing the most suitable tools for analysis because the SLFN is known for its ability as a universal approximator. The proposed concluding procedure helps the practitioner to consolidate the exhibited features among different disjoined regions of the target area and all obtained SLFNs as well as rate the plausibility of prior beliefs regarding unconvinced-but-existing or convinced-but-absent features. That is, after observing the extracted feature A' from obtained SLFNs, we abduce the hypothetical feature A from the circumstance A' and further to surmise that A may be true because then A' would be a matter of course. To abduce A robustly from A' involves determining that A is sufficient (or nearly sufficient) but not necessary for A'. To develop robust abductive-reasoning skills and experiences is left for future research.

Table 5 The number of training samples contained in each disjoined region of Networks I, II and III, respectively, when the adopted approximation function is $g^{\tilde{I}}$ shown in (S3) of Supplement

Region	$R^{\tilde{I}}_{I,(4,4,3,3)}$	$R^{\tilde{I}}_{I,(4,4,4,3)}$	$R^{\tilde{I}}_{II,(3,3,3,3)}$	$R^{\tilde{I}}_{II,(3,3,3,4)}$	$R^{\tilde{I}}_{III,(3,3,4,4)}$	$R^{\tilde{I}}_{III,(3,3,5,4)}$
Number of contained training samples	160	160	46	274	319	1

Only 320 of the 1,440 training samples are covered in the target area

We also demonstrate the appropriateness of adopting mathematical programming analysis in place of the data analysis (i) to identify the premise of each multivariate polynomial rule and (ii) to verify the features exhibited in each disjoined region. With mathematical programming analysis, both rules and features can be verified, for instance, in the $R^{\tilde{I}}_{III,(3,3,5,4)}$ shown in Table 5, a region with few or no (training) observations.

Note that the practitioner will usually accord his or her prior belief high plausibility when a consistent outcome is obtained from the proposed abductive-reasoning guide. Conversely, an inconsistent outcome will normally trigger a reexamination of the practice instead of an immediate revision of the prior belief. The practitioner may

(i) examine whether some of the obtained well-trained SLFNs are not globally optimal to the extent that they are unsuitable for the purpose of revising the prior beliefs, or
(ii) investigate whether certain factors or noises may cause the design of the SLFN to be inadequate.

Only after considering and eliminating these alternative explanations will the practitioner update his or her prior belief. Otherwise, he or she should use the outcome from the proposed abductive-reasoning guide conservatively.

The proposed guide can be applied to any listed feature with a higher order differential relation. The extracted feature can describe the differential relation with social scientific application concerns. Thus, a possible future avenue of further inquiry may be the exploration of a social scientific application using the proposed process set out here.

Another interesting avenue for further study is the application of the proposed abductive-reasoning guide for different fields; for instance, to price future exotic options, researchers may adopt a Morte Carlo simulation. Using the simulated present values of options and the abductive-reasoning process, the practitioner may enhance his or her understanding or derive a useful pricing model. Another avenue of study may be the elimination of redundant constraints from the premise of a rule or the integration of extracted rules. Moreover, if the practitioner is only interested in finding out the highest or the lowest credible feature (i.e., checking whether the entire target area is CA or AIA for a listed feature), the skills of piecewise linear programming (ref. Cavichia and Arenales 2000; Rozvany 1971) may improve the performance of the abductive-reasoning guide.

References

Andreou, P. C., Charalambous, C., & Martzoukos, S. H. (2006). Robust artificial neural networks for pricing of European options. *Computational Economics*, *27*(2), 329–351.

Andrews, R., Diederich, J., & Tickle, A. B. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems*, *8*(6), 373–389.

Baesens, B., Setiono, R., Mues, C., & Vanthienen, J. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science*, *49*(3), 312–329.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, *81*(3), 637–654.

Cavichia, M. C., & Arenales, M. N. (2000). Piecewise linear programming via interior points. *Computer and Operation Research*, *27*(3), 1303–1324.

Domingos, P. (2007). Toward knowledge-rich data mining. *Data Mining and Knowledge Discovery*, *15*(1), 21–28.

Kiani, K. M., & Kastens, T. L. (2008). Testing forecast accuracy of foreign exchange rates: Predictions from feed forward and various recurrent neural network architectures. *Computational Economics*, *32*(4), 383–406.

Kiani, K. M. (2011). Fluctuations in economic and activity and stabilization policies in the CIS. *Computational Economics*, *37*(2), 193–220.

LeBaron, B. D. (2000). Agent-based computational finance: Suggested readings and early research. *Journal of Economic Dynamics and Control*, *24*, 679–702.

Luenberger, D. (1984). *Linear and nonlinear programming*. Reading: Addison-Wesley.

Munkres, J. (1975). *Topology: A first course*. Englewood Cliffs: Prentice-Hall.

Peirce, C. S. (1992). In K. L. Ketner (Ed.), *Reasoning and the logic of things*. Cambridge: Harvard University Press.

Peirce, C. S. (2011). Collected Papers of Charles Sanders Peirce v. 5, 1903, paragraphs 188–189. http://www.textlog.de/7664-2.html. Accessed 12 Sept 2011.

Reboredo, J. C., Matías, J. M., & Garcia-Rubio, R. (2012). Nonlinearity in forecasting of high-frequency stock returns. *Computational Economics*, *40*(3), 245–264.

Rozvany, G. I. N. (1971). Concave programming and piece-wise linear programming. *International Journal for Numerical Methods in Engineering*, *3*(1), 131–144.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representation by error propagation. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition* (Vol. 1, pp. 318–362). Foundation Cambridge: MIT Press.

Saito, K., & Nakano, R. (2002). Extracting regression rules from neural networks. *Neural Network*, *15*(10), 1297–1288.

Setiono, R., & Liu, H. (1996). Symbolic representation of neural networks. *Computer*, *29*(3), 71–77.

Setiono, R., & Liu, H. (1997). NeuroLinear: From neural networks to oblique decision rules. *Neurocomputing*, *17*(1), 1–24.

Setiono, R., Leow, W. K., & Zurada, J. M. (2002). Extraction of rules from artificial neural networks for nonlinear regression. *IEEE Transactions on Neural Networks*, *13*(3), 564–577.

Taha, I. A., & Ghosh, J. (1999). Symbolic interpretation of artificial neural networks. *IEEE Transactions on Knowledge and Data Engineering*, *11*(3), 448–463.

Tickle, A. B., Andrews, R., Golea, M., & Diederich, J. (1998). The truth will come to light: Directions and challenges extracting the knowledge embedded within trained artificial neural networks. *IEEE Transactions on Neural Networks*, *9*(6), 1057–1068.

Tsaih, R., Hsu, Y., & Lai, C. (1998). Forecasting S&P 500 stock index futures with the hybrid AI system. *Decision Support Systems*, *23*(2), 161–174.

Tsaih, R., & Wan, Y. W. (2009). A guide for the upper bound on the number of continuous-valued hidden nodes of a feed-forward network. *Lecture Notes in Computer Science*, *5768*, 658–667.