

# An explainable AI decision-support-system to automate loan underwriting

Swati Sachan\*, Jian-Bo Yang, Dong-Ling Xu, David Eraso Benavides, Yang Li

Decision and Cognitive Sciences Research Centre, Alliance Manchester Business School, The University of Manchester, Booth Street West, Manchester M15 6PB, United Kingdom

## ARTICLE INFO

### Article history:

Received 26 April 2019

Revised 31 October 2019

Accepted 23 November 2019

Available online 27 November 2019

### Keywords:

Explainable artificial intelligence

Interpretable machine learning

Loan underwriting

Evidential reasoning

Belief-rule-base

Automated decision making

## ABSTRACT

Widespread adoption of automated decision making by artificial intelligence (AI) is witnessed due to specular advances in computation power and improvements in optimization algorithms especially in machine learning (ML). Complex ML models provide good prediction accuracy; however, the opacity of ML models does not provide sufficient assurance for their adoption in the automation of lending decisions. This paper presents an explainable AI decision-support-system to automate the loan underwriting process by belief-rule-base (BRB). This system can accommodate human knowledge and can also learn from historical data by supervised learning. The hierarchical structure of BRB can accommodate factual and heuristic rules. The system can explain the chain of events leading to a decision for a loan application by the importance of an activated rule and the contribution of antecedent attributes in the rule. A business case study on automation of mortgage underwriting is demonstrated to show that the BRB system can provide a good trade-off between accuracy and explainability. The textual explanation produced by the activation of rules could be used as a reason for denial of a loan. The decision-making process for an application can be comprehended by the significance of rules in providing the decision and contribution of its antecedent attributes.

© 2019 Elsevier Ltd. All rights reserved.

## 1. Introduction

Underwriting skill is learnt through several months of training and exchange of knowledge by senior underwriters. This task requires underwriters to be fairly analytical, very organized, and accurate to give informed decision to approve or reject a loan application. Underwriters concurrently analyse a large quantity of information to find affordability, repayment history and collateral. Furthermore, sometimes they are required to change the process due to a shift in regulatory and compliance standards, investor requirements, and customer demands (Krovvidy, 2008).

New technology and strong machine learning (ML) algorithms have opened the doors for a straight-through loan application process. Artificial intelligence (AI) systems can execute rules and process customers' information in a few milliseconds. Financial institutions have recognized the benefits of AI and are using it in a different subset of the underwriting process and are keen to test

and implement newly introduced digital innovation. AI systems are expected to replicate human decision-making skills. However, even today transformation of various algorithmic concepts into training data could be very challenging to solve every instance of the problem for a range of lending products. It may not be able to solve a tiny subset of the problem (Aggour, Bonissone, Cheetham & Messmer, 2006).

### 1.1. Manual underwriting challenges

Manual underwriting task is a very much paper-based process. It is an inconvenient process of circulation of loan application files within different departments of a lending institution. Full attention to details is requisite to give sound judgment on an application. Human underwriter evaluates scenarios by analysing a large amount of dynamic information in a loan application. This could be a source of inconsistency, inaccuracy, and biases (Peterson, 2017).

Manual underwriting is often successful in processing non-standard loans. Many lenders like high street banks and building societies follow strict rules and do not offer personalized underwriting. However, there are some lenders who exercise common sense lending approaches for assessing both standard and non-standard cases such as non-standard properties, non-standard

\* Corresponding author.

E-mail addresses: [swati.sachan@manchester.ac.uk](mailto:swati.sachan@manchester.ac.uk) (S. Sachan), [jian-bo.yang@manchester.ac.uk](mailto:jian-bo.yang@manchester.ac.uk) (J.-B. Yang), [Ling.Xu@manchester.ac.uk](mailto:Ling.Xu@manchester.ac.uk) (D.-L. Xu), [david.erasobenavides@postgrad.manchester.ac.uk](mailto:david.erasobenavides@postgrad.manchester.ac.uk) (D.E. Benavides), [yang.li-33@postgrad.manchester.ac.uk](mailto:yang.li-33@postgrad.manchester.ac.uk) (Y. Li).

income/employment, and less than perfect credit scores. The underwriting process of non-standard cases is very detailed and individualistic. Common senses lending approaches serve underserved people

## 1.2. Advantages and concerns of automated underwriting systems

In 1995 Fannie Mae, a US largest mortgage lending company introduced first Desktop Underwriter that applied both heuristics and statistics to process mortgage loan in less time, cost, and paperwork (Cocheo, 1995). In the same year, another US lending company, Freddie Mac introduced an automated underwriting system called loan prospector (Cocheo, 1995). Most lenders use an automated system which contains coded underwriter guidelines which provide the decision of acceptance or rejection when certain default rules in the rule base are triggered. Early statistical methods were limited to a small sample of data, but now data scientists have access to big data to provide meaningful learning of AI/ML models. Digital technology advancements have allowed the use of big data to find the multitudinous ways by which features influence each other. Linear and non-linear relationships between features are found by advanced machine learning algorithms, such as ensemble methods, neural network, support vector machine, and random forest. These models have superior ability to classify the risk status of loan applications. Further, they are far more accurate for conforming loans than manual underwriting as ML models provides outcome by combining a large number of observations in the dataset (Gates, Perry & Zorn, 2002). The speedy decision by an automated AI system provides satisfactory customer experience (Krovvidy, 2008) (Gates et al., 2002). Though, a manual underwriter may have access to more detailed information for non-conforming loans.

Automated AI systems have huge potential for sure; however, the rapid adoption of AI in decision making also has concerns. The following questions must be addressed before developing AI-based decision-making tool:

### 1.2.1. Are we accessing the right data to map manual underwriting process?

Automation by advanced AI algorithms raises ethical questions. The data used to train these models may only contain instances for a subset of underwriting practices. Learning of AI/ML models by such data can give a correct decision for only a group of borrowers which exist in the historical data and will disregard borrowers with a rare case or less than a standard application. This, in turn, can make the automated AI a biased system as it is the source of unintentional discrimination. It is important to select the right features to capture the underwriting process.

### 1.2.2. Can AI automated underwriting system restore common senses in lending decisions?

Automated AI system consists of codified rules and advanced AI algorithms. Lending institutions have their predefined standard rules which revolve around affordability, defaults, loan-to-value, and loan-to-income. The codification of these rules may not represent the lending decision of all groups of borrows (Gates et al., 2002). Minority and low-income borrowers are underrepresented through these rules. Depending on the type of lending product, some high street banks give either fully or partially automated decision based on some strict rules which leave no room for looking at loan applications at a personal level. It can be argued that human underwriters can make better decisions for exceptional cases (Glassman & Wilkins, 1997)

### 1.2.3. Can AI incorporate changes in underwriting practices due to a shift in some policy?

Lending institutions change their policy to include traits of high risk borrowers that were previously approved and add traits of promising borrowers that were previously rejected. A new policy requires new data records for a set of attributes or features which reflect new underwriting rules to train AI algorithm.

### 1.2.4. How to deal with lack of explainability in the AI model?

The idea to explain a decision made by an AI model and its inner working started in the 1970s to test and engender user trust in expert systems (Stein, Preece & Boicu, 2018). However, specular advances in computation power and improvements in optimization algorithms shifted the focus towards accuracy, while the ability to explain decisions has taken a back seat (Adadi & Berrada, 2018). In the past, FraudWatch an expert system based on KADS (Knowledge Acquisition and Documentation Structuring) was deployed by Barclay Bank to detect fraud in credit card transactions (Killin, 1992). Another system, X-MATE (Expert Mortgage Ar-rears Threat Advisor) was developed by Hewlett Packard's Knowledge System centre and AI application institute of the University of Edinburgh for a large UK building society to detect high-risk loan applicants (Kingston, 1991). This system was not entirely based on KADS; it simply adapted its knowledge acquisition and structuring methodology. It assessed loan application by comparing it with high-risk application in knowledge-base than an ideal application. Both XMATE and FraudWatch had IF-THEN rules obtained from experienced experts; however, parameters like conclusion given by a rule were not trained by historical data and conclusion in each rule was certain, which does not give room to accommodate uncertainty in cases where experts are not sure about the decisions given by the rules.

Machine learning (ML) is an emerging frontier of AI. Some machine learning models like decision tree and logistic regression can explain decisions. High performance ML models such as deep-neural-network and ensemble methods are very complex and consequently, decisions from these models are considered much more difficult to explain to end-users and other non-technical stakeholders. In the past years, many studies have presented various ML models for credit scoring. Among such ML models, some most popular studies are on ensemble classifier (Xu, Zhang, & Feng, 2018) (Abellán & Castellano, 2017) (Xiao, Xiao & Wang, 2016) (Marqués, García & Sánchez, 2012) (Wang, Ma, Huang & Xu, 2012) (Hung & Chen, 2009) (Nanni & Lumini, 2009), support vector machine (Harris, 2015) (Tomczak & Zięba, 2015) (Hens & Tiwari, 2012) (Chen, Ma & Ma, 2009) (Huang, Chen & Wang, 2007) (Li, Shiue & Huang, 2006) (Schebesch & Stecking, 2005), neural network (Luo, Wu & Wu, 2017) (Zhao et al., 2015) (Li et al., 2006) (Khashman, 2010) (Bensic, Sarlija & Zekic-Susac, 2005) (Kim & Sohn, 2004) (West, 2000), genetic programming (Metawa, Hassan & Elhoseny, 2017) (Abdou, 2009) (Ong, Huang & Tzeng, 2005), Bayesian network (Leong, 2016) (Wu, 2011) (Zhu, Beling & Overstreet, 2002) and decision tree (Bijak & Thomas, 2012) (Yap, Ong & Husain, 2011) (Zhang, Zhou, Leung & Zheng, 2010) (Bensic et al., 2005).

The concept of explainability of decisions from a ML model is not discussed in these studies. The term explainability and interpretability are often used interchangeability. Transparency is used as a synonym for interpretability. Interpretability refers to the understanding of working logic of an AI-based decision-making system and explainability relates to the ability of AI system to provide a suitable explanation on how it arrived at a decision (Montavon, Samek & Müller, 2018) (Lipton, 2016). Some ML/AI model such as decision tree and logistic regression fall into the class of interpretable models. However, these models lack accuracies compared to complex black-box models and have nested

non-linear structure which makes them opaque. Deep-neural-net and ensemble models such as random-forest, XGboost and Adaboost are most successful black-box models. The need for explainability of a decision depends on the impact of wrong decisions. The impact varies with the given use case criticality. For example, the impact of the recommendation of wrong movies and advertisements is much less than the lending loan to unpromising customers, and most importantly misdiagnosis of patients could have catastrophic consequences. The trust for an AI system diminishes with an increase in sensitivity of impact of the decision provided by the AI system rather than a human expert.

Explainability can benefit organisations by boosting confidence in executives accountable for unintended consequences of incorrect decision from AI automated systems. It helps them to comply by the regulatory standards for AI such as the General Data Protection Regulation (GDPR) in Europe. It enables data scientists and developers to detect potential weakness and flaws in a model and strengthen the trust of customers in decisions made by automated AI systems. In recent years, several explainability methods and strategies have been proposed to unfold the outputs of black-box ML models. These methods can be classified into two groups: model specific and model agnostic (Adadi & Berrada, 2018).

Model-specific interpretation methods are limited to a specific model. For example, tree interpreter for random forest and deep-taylor-decomposition for the neural network (Montavon, Lapuschkin, Binder, Samek & Müller, 2017). Model-agnostic models can be applied to any type of ML model. It can be local interpretation method such as Local Interpretable Model Agnostic Explanations (LIME) (Ribeiro, Singh & Guestrin, 2016) and Shapley (Lundberg & Lee, 2017) (Casalicchio, Molnar & Bischl, 2018) that can explain a specific decision and can be a global interpretation method such as sensitivity analysis to understand entire model behavior. These methods provide explanation graphically in the form of local feature importance by multi-feature perturbation around a particular decision or by repeating multi-feature perturbation for a set of random decisions to approximate global feature importance. However, these methods can not reveal the true decision-making process and the chain of events leading to a decision. More research is required to develop strategies to explain ML models and develop better ML/AI algorithms to make a shift towards more transparent AI.

In this paper, we aim to address the above problems in the automation of loan underwriting process by belief-rule-base (BRB) expert system. It is an extension of traditional IF-THEN rule-base-systems. These rules can accommodate the knowledge of experts who have extensive experience in the task of assessing loan applications in a lending institution. The rules in BRB can be trained to learn from historical credit data by capturing the non-linear relationship between antecedent attributes and consequence attribute. The rule-base can be given a hierarchical structure which can contain factual and heuristic rules. It is a transparent decision-making system that can reveal the true decision-making process and a chain of events leading to a decision. The main purpose of this paper is to present a detailed methodology to design a belief-rule-base system as an explainable decision-support-system for automation of loan lending process and to present a business case study on automation of mortgage underwriting.

There are two types of credit scoring task (Khashman, 2010) (Li et al., 2006). The first type is to classify applicants into classes such as 'fund' or 'reject' and 'bad' or 'good' which can be done either manually by an underwriter or by an automated underwriting system. In contrast, the second type discovers existing customer repayment patterns to update future lending policy. In this paper, we focus on making an automated decision of funding or rejecting loan applications by mimicking the underwriting procedure of

a company and extracting the non-linear relationship among attributes (features).

The rest of paper is organized as follows. Section 2 describes the methodology to develop BRB based decision-support-system. In Section 2.1, the representation of rules is described. In Section 2.2, the procedure for knowledge acquisition and data collection are explained. In Section 2.3, we discuss a detailed methodology for the transformation of quantitative and qualitative data into a belief distribution. In Section 2.4, the hierarchical structure of knowledge-base is analysed and in Sub-Section 2.5 we explain the inference of BRB expert system by using the evidential-reasoning (ER) approach and by training BRB system parameters. In Section 3, we demonstrate a business case-study on mortgage lending. The paper is concluded in Section 4.

## 2. Automation of loan underwriting by belief rule-based (BRB) method

### 2.1. Knowledge representation using rules

There are two types of knowledge in an area of expertise, facts and heuristics (Feigenbaum, 1980). Facts are widely accepted information in any particular domain. Heuristics knowledge is the information gained through experience. This type of information leads to the formation of judgmental rules and rules of plausible reasoning. These rules are about one's ability to evaluate, judge and guess in a situation. An expert uses heuristic knowledge gathered through years of experience and learning.

Human knowledge is represented in the form of IF-THEN rules (Sun, 1995). A belief rule is the extension of traditional IF-THEN rule where domain expert knowledge can be represented as fact if it is crisp or uncertain if it is heuristic. A belief rule has two parts, Antecedent part (IF) and consequent part (THEN). The antecedent part contains one or more antecedent attributes, and each antecedent attribute takes referential value from a set of referential values. The consequent part contains one consequent attribute and referential values on consequent attribute are given belief degrees.

Suppose, belief-rule-base (BRB) has  $A_i, i = 1, \dots, I$  antecedent attributes and one consequence attribute  $H$ . The attributes in BRB model is represented as:

$$BRB = \langle A_1, \dots, A_i, \dots, A_I, H \rangle$$

Each antecedent attribute has a set of referential values. The set of referential values for an antecedent attribute can be represented as  $A_i = \{A_{v,i}; v = 1, \dots, V_i\}$ . Referential values can be either numerical for quantitative attribute or categorical for qualitative attributes. If there are  $I$  total antecedent attributes, then there are  $Q$  qualitative attributes and  $J$  quantitative attribute, with  $I = Q + J$ . There is only one consequence attribute. The set of referential values for a consequence attribute can be written as  $H = \{h_n; n = 1, \dots, N\}$ . Suppose there are  $K$  ( $k \in 1, \dots, K$ ) number of rules in the rule-base. Then belief degree (BD) for the consequence of each rule can be written as  $BD = \{(h_1, \beta_{1,k}), \dots, (h_n, \beta_{n,k}), \dots, (h_N, \beta_{N,k})\}, n = 1 \dots N$ . A rule is the combination of different referential values of antecedent attributes in BRB model. The structure of  $k^{th}$  rule can be written as:

$$R_k : \begin{cases} IF : (A_1 \text{ is } A_{v,1}^k) \wedge (A_2 \text{ is } A_{v,2}^k) \wedge \dots \wedge (A_I \text{ is } A_{v,I}^k) \\ THEN : \{(h_1, \beta_{1,k}), \dots, (h_n, \beta_{n,k}), \dots, (h_N, \beta_{N,k})\} \\ \text{sum of belief } \beta_{n,k} : \sum_{n=1}^N \beta_{n,k} = 1 \text{ for all } k \{1, \dots, K\} \\ \text{rule weight : } \theta_k \leq 1 \text{ for all } k \{1, \dots, K\} \\ \text{attribute weight : } \delta_i^k \text{ for all } k \{1, \dots, K\}, i \in \{1, \dots, I\} \end{cases} \quad (1)$$

where  $\wedge$  is a logical connective to represent the AND relationship.

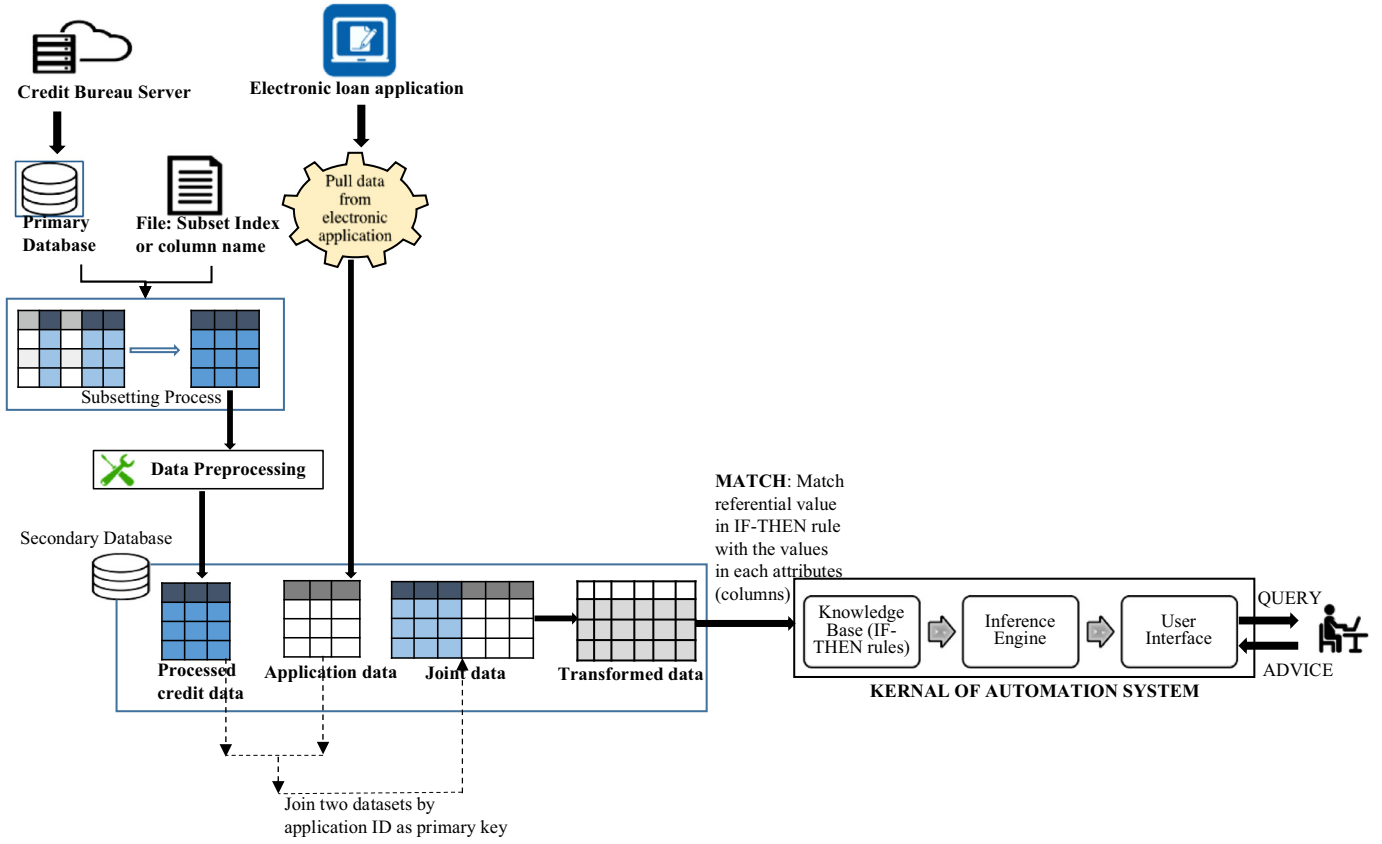


Fig. 1. Knowledge acquisition and data collection.

For example, suppose BRB model has three attributes and one consequence attribute  $BRB = \langle A_1 = \text{Bankruptcy in last 4 years}, A_2 = \text{Credit card arrears in last 18 months}, A_3 = \text{monthly affordability}, H = \text{Decision} \rangle$ . Each attribute has a set of referential values. A rule from these attributes can be written as:

$$R_1 \begin{cases} \text{IF : } (\text{Bankruptcy in last 4 years is } A_{v,1}^1) \\ \quad \wedge (\text{Credit card arrears in last 18 months is } A_{v,2}^1) \\ \quad \wedge (\text{monthly affordability is } A_{v,3}^1) \\ \text{THEN : } \{(\text{Fund}, 0.20), (\text{Reject}, 0.80)\} \end{cases}$$

This rule can be simply rewritten as: **IF**  $A_{v,1}^1 \wedge A_{v,2}^1 \wedge A_{v,3}^1$ ; **THEN**  $\{(\text{Fund}, 0.20), (\text{Reject}, 0.80)\}$ .

## 2.2. Knowledge acquisition and data collection

The task of data extraction demands a good understanding of data layout and underwriting guidelines. Data model or metadata can be used to understand the data. Data extraction is designed by developers. They work in conjunction with knowledge engineers. Knowledge engineers translate the information in the form of IF-THEN rules in the computer. They elicit information from domain experts like experienced underwriters and seniors who design rules based on regulations. Rules vary by type of lending product. Baseline rules are defined in all lending institutions, however, to make manual decision underwriters use their experience which is beyond the defined rules and guidelines. Both extracted data and rules should reflect the practical experience of the underwriters and lending institutions guidelines. Within the field of AI, knowledge acquisition is considered a challenge for the development of an expert system. Knowledge analysis and design structure (KADS) for expert systems (Akinyokun, 2015) and unstructured interview, structured interview, protocol analysis, psychological scal-

ing and card sorting for an expert system in finance described in Wagner, Otto and Chung (2002) can be used for knowledge acquisition. The complexity of domain strongly influences the quality of knowledge (Holsapple, Raj & Wagner, 2008).

Credit data is extracted from a credit bureau server. It is then stored in a primary database of a lending institution. Index or column names are stored in a file. This file is used in future to retrieve the columns which could be used to map the underwriting guidelines and experience. The process of retrieval of parts of a large data file is called subsetting, as shown in Fig. 1. Credit bureaus compile more than 2000 variables or columns for each individual. They have access to multiple sources of information such as banks, telecom, utility, electoral commission, county court (in England and Wales) judgments related to unpaid debt, parking, driving and council tax. Traditionally, lending institutions use credit bureau, loan application, and lenders own loan criteria and existing customer's data. Alternative data sources like transaction data, social network data, and clickstream data are getting a lot of attention to enhance the accuracy and judgement power of machine learning models (Wei, Yildirim, Van den Bulte & Dellarocas, 2015) (Aitken, 2017).

The subset of credit data is preprocessed, which includes cleaning and treatment. The missing values are given label 'unknown'; a missing piece of credit information can be used as evidence. Sufficient credit information of a customer helps in effective decision making. A missing important piece of information or almost no credit history could lead to rejection of a loan application. For data treatment, some of the columns could be aggregated together by average, maximum, minimum, and sum operation. For example, data for the rule 'worst status of fixed-term account in the last 12 months' would be maximum of worst status columns for the current address, previous address, and linking address.



Likewise, data for rule 'Number of bankruptcy in the last 6 years' would be the sum of columns for the number of satisfied and unsatisfied bankruptcy in the last 6 years. Preprocessed data is stored in a secondary database where data from the electronic application is also extracted and stored. The data extracted from the electronic application can be used to calculate the affordability, debt-to-asset ratio, and loan-to-value ratio. It will be explained in detail in the case study. Both processed credit data and electronic application data is merged together by application ID as a primary key.

Merged data set is then transformed in the form of belief structure over the referential values of antecedent attributes. Data transformation is explained in detail in Section 2.3. The referential values of transformed data are matched with the values in IF-THEN rule. In inference engine, rules activated by matching of transformed data and IF-THEN rules are used to calculate the activation weight of each rule at a data point. All activated rules are then aggregated together to generate final inference output. The inference output is utilized by the underwriter (user) to make a decision.

### 2.3. Data transformation

BRB framework can be implemented on precise data and subjective judgment with either uncertainty or certainty. Both certainty and uncertainty are incorporated by transforming data into belief structure over the referential values of antecedent attributes. The belief distribution of each data point is used to calculate the activation weight of a rule. Additionally, the inference engine in BRB is the evidential reasoning approach, which works with data in the format of belief distribution. Input data can have two types of antecedent attributes, quantitative and qualitative. The quantitative attribute contains continuous data and qualitative attribute contains categorical data.

Suppose  $X$  denotes the full dataset. It contains  $x_{m,i}$  data sample where  $m \in \{1, \dots, M\}$  is the number of data points and  $i \in \{1, \dots, I\}$  is the number of attributes. The input dataset is shown in Table 1. It has  $I - 1$  antecedent attributes and one consequence attribute ( $A_I$ ).

The data point  $x_{m,i}$  transformed into a belief distribution can be represented as:

$$S(x_{m,i}) = \left\{ (A_{v,i}, \alpha_{v,i}(x_{m,i})), v \in 1, \dots, V_i, \sum_{v=1}^{V_i} \alpha_{v,i}(x_{m,i}) = 1 \right\} \quad (2)$$

where,  $A_{v,i}$  is the  $v^{th}$  referential value of the  $i^{th}$  antecedent attribute. The set of referential values for antecedent attribute is  $A_i = \{A_{v,i}, v \in 1, \dots, V_i\}$ . Here  $V_i$  is the number of referential values of an antecedent attribute  $A_i$ .  $\alpha_{v,i}(x_{m,i})$  is the matching degree which measures the matching degree of the  $m^{th}$  data point to the  $v^{th}$  referential value of the  $i^{th}$  antecedent attribute. Method to obtain  $\alpha_{v,i}(x_{m,i})$  depends on the data type of antecedent attribute  $A_i$ . Data transformation for quantitative attributes (continuous) and qualitative attributes (categorical) is summarized below.

**Table 1**  
Dataset.

Number of data points	Antecedent attributes					Consequence attribute
	$A_1$	$\dots$	$A_i$	$\dots$	$A_{I-1}$	$A_I$
1	$x_{1,1}$		$x_{1,i}$		$x_{1,I-1}$	$x_{1,I}$
$\vdots$	$\vdots$		$\vdots$		$\vdots$	$\vdots$
$m$	$x_{m,1}$	$\dots$	$x_{m,i}$	$\dots$	$x_{m,I-1}$	$x_{m,I}$
$\vdots$	$\vdots$		$\vdots$		$\vdots$	$\vdots$
$M$	$x_{M,1}$		$x_{M,i}$		$x_{M,I-1}$	$x_{M,I}$

#### 2.3.1. Quantitative attribute

If  $A_i$  is quantitative attribute then  $\alpha_{v,i}$  for  $x_{m,i}$  can be obtained by referential values  $A_{v,i}$  and  $A_{v+1,i}$  if  $x_{m,i}$  lies between these two referential values.

$$\alpha_{v,i}(x_{m,i}) = \frac{A_{v+1,i} - x_{m,i}}{A_{v+1,i} - A_{v,i}} \text{ and } \alpha_{v+1,i}(x_{m,i}) = 1 - \alpha_{v,i}(x_{m,i}) \quad (2a)$$

$$\text{if } A_{v,i} \leq x_{m,i} \leq A_{v+1,i} \text{ with } \alpha_{v,i}(x_{m,i}) + \alpha_{v+1,i}(x_{m,i}) = 1$$

$$\alpha_{v',i}(x_{m,i}) = 0 \text{ for } v' = 1, \dots, V_i \text{ and } v' \neq v, v+1 \quad (2b)$$

For intuitive illustration, suppose a set of referential values for antecedent attribute called credit score is  $A_1 = \{0, 100, 300, 600\}$  and we want to transform the 1st and the 2nd data point in the dataset,  $x_{1,1} = 33$  and  $x_{2,1} = 456$ , respectively. For data transformation we need two successive referential values such that  $x_{1,1}$  and  $x_{2,1}$  lies between them. The 1st data point  $x_{1,1} = 33$  lies between  $A_{1,1} = 0$  and  $A_{2,1} = 100$  so  $\alpha_{1,1}(x_{1,1}) = \frac{100-33}{100-0} = 0.67$  and  $\alpha_{2,1}(x_{1,1}) = 1 - \alpha_{1,1}(x_{1,1}) = 1 - 0.67 = 0.33$ . Similarly, the 2nd data point  $x_{2,1} = 456$  lies between  $A_{3,1} = 300$  and  $A_{4,1} = 600$  then  $\alpha_{3,1}(x_{2,1}) = \frac{600-456}{600-300} = 0.48$  and  $\alpha_{4,1}(x_{2,1}) = 1 - 0.48 = 0.52$ . Table 2 shows the transformed data of a quantitative attribute.

#### 2.3.2. Qualitative attribute

*Transformation into belief structure over the referential values of qualitative attributes:* The referential value of a qualitative antecedent attribute is described by a set of linguistic value. It can also be called categorical levels. The qualitative value of an attribute belongs only to one of the categories in a set of referential values. This implies that the qualitative data point will have 100% membership to one of the referential values. For example, suppose credit score ( $A_1$ ) is a qualitative attribute and its set of referential value is  $A_1 = \{\text{excellent}, \text{good}, \text{average}, \text{poor}\}$ . If the 1st data point in credit score attribute is 'good' and the 2nd is 'average' then,  $\alpha_{\text{good},1}(x_{1,1}) = 1$  and  $\alpha_{\text{average},1}(x_{2,1}) = 1$ . Table 3 shows the transformed data of a qualitative attribute.

*Data fusion – maximum likelihood evidence reasoning:* The BRB framework has shown the advantage of finding non-linear relationships between inputs and output in a probabilistic manner and has the ability to handle inaccurate and ambiguous data by giving weights to rules and attributes in each rule (Yang, Liu, Wang, Sii & Wang, 2006). However, in the BRB framework, the number of rules can increase exponentially with the number of antecedent attributes and the number of referential values in each antecedent attribute. The number of rules in BRB is:

$$K = \prod_{i=1}^I V_i \quad (3)$$

*Credit data has a large volume:* Each applicant can have several hundreds of columns and credit data is combined with other data sources as well, as described in Section 2.2. Several columns can

**Table 2**  
Example of quantitative data transformation.

# data points	$A_1$ : Credit score	Transformed data
1	33	$\{(0,0.67), (100,0.33), (300,0), (600,0)\}$
2	456	$\{(0,0), (100,0), (300,0.48), (600,0.52)\}$

**Table 3**  
Example of qualitative data transformation.

# data points	$A_1$ : Credit score	Transformed data
1	good	$\{(\text{excellent}, 0), (\text{good}, 1), (\text{average}, 0), (\text{poor}, 0)\}$
2	average	$\{(\text{excellent}, 0), (\text{good}, 0), (\text{average}, 1), (\text{poor}, 0)\}$

be selected to map a lending organization's rules and reflect underwriter knowledge. Increase in the number of columns or attributes increase the number of rules; consequently, this increases the number of training parameters in the BRB model. This challenge could be overcome by fusing qualitative attributes (categorical data) by maximum likelihood evidence reasoning (MAKER). The MAKER framework combines evidence to fuse qualitative attributes. It is a data-driven inference modelling approach, which was introduced in [Yang and Xu \(2017\)](#), and extends Bayesian inference. The core of Bayesian inference is Bayes' rule that is used to combine prior evidence with new one. Prior evidence is profiled as a probability distribution over a set of hypotheses, while new evidence is characterized by the likelihood that the evidence is observed for each given hypothesis. Bayes' rule updates knowledge once new evidence is available. Unlike Bayes' rule, the MAKER framework does not depend on prior probability and can handle ambiguous information for inference. In MAKER, prior is treated as an independent piece of evidence and added to evidence set if available; otherwise, it is treated as completely unknown. MAKER is introduced briefly in this section.

Suppose  $H = \{h_1, \dots, h_n, \dots, h_N, n \in \{1, \dots, N\}\}$  is a set of mutually exclusive and collectively exhaustive hypotheses, or  $h_q \cap h_{q'} \neq \emptyset$  for any  $q, q' \in \{1, \dots, N\}$  and  $i \neq j$ , where  $\emptyset$  is the empty set.  $H$  can be referred as frame of discernment ([Shafer, 1976](#)). The power set of  $H$  consists of  $2^N$  subsets of  $H$ , denoted by  $P(H)$ , as follows:

$$P(H) = \{\emptyset, \{h_1\}, \dots, \{h_n\}, \{h_1, h_2\}, \dots, \{h_1, h_N\}, \dots, \{h_1, \dots, h_{N-1}\}, H\} \quad (4)$$

The set of categorical or qualitative attributes can be represented in the following way:

$$A' = \{A'_1, \dots, A'_q, \dots, A'_Q\}, q \in \{1, \dots, Q\} \quad (5)$$

A piece of evidence is denoted by  $e$ . Each categorical attribute has  $V_q$  ( $q \in \{1, \dots, Q\}$ ) linguistic referential values. The referential value of a categorical attribute  $A'_q$  is given by  $A'_{v,q}$  ( $v \in \{1, \dots, V_q\}$ ). A piece of evidence  $e_{v,q}$  is obtained from the  $v^{\text{th}}$  referential value of the  $q^{\text{th}}$  categorical attribute and  $e_{v,q}(h)$  is evidence  $e_{v,q}$  pointing exactly to a proposition  $h$ . In the MAKER framework, pieces of evidence obtained from a set of input variables are combined to generate combined support for a proposition  $h$ . One piece of evidence is combined with another piece of evidence by taking account of four elements – basic probability, the interrelation between two pieces of evidence, the reliability and weight of each piece of evidence. A piece of evidence  $e_{v,q}$  is profiled by a belief distribution or basic probability distribution as follows:

$$e_{v,q} = \left\{ (e_{v,q}(h), p_{h,v,q}), \forall h \in P(H) \text{ and } \sum_{h \in P(H)} \alpha_{h,v,q} = 1 \right\} \quad (6)$$

The basic probability  $\alpha_{h,v,q}$  is obtained from data collected for attribute  $A'_q$ . The evidence from single input variable  $A'_q$  can be transformed to a basic probability distribution for all  $h \in P(H)$ . This is called one-dimensional evidence acquisition process. It can be obtained from [Eq. \(7\)](#). The basic probability is obtained from the normalized likelihood. The likelihood is obtained by generating a contingency table. Let  $C_{h,v,q}$  be the likelihood of observing the  $v^{\text{th}}$  referential value of an attribute  $A'_q$  for given proposition  $h$ . The basic probability is then generated by the following normalised likelihood:

$$p_{h,v,q} = \frac{C_{h,v,q}}{\sum_{g \in P(H)} C_{g,v,q}} \forall h \in P(H) \quad (7)$$

Similarly, joint basic probability can be acquired from joint likelihood function. Suppose, we want to combine two categorical attribute  $A'_q$  and  $A'_{q'}$ , their set of referential values is

$A'_{v,q}$  ( $v \in \{1, \dots, V_q\}$ ) and  $A'_{v',q'}$  ( $v' \in \{1, \dots, V_{q'}\}$ ), respectively. Suppose two pieces of evidence  $e_{v,q}$  and  $e_{v',q'}$  are acquired from attributes  $A'_q$  and  $A'_{q'}$ , respectively. If  $C_{h,v,q,v',q'}$  is the joint likelihood of observing  $A'_{v,q}$  and  $A'_{v',q'}$  for given proposition  $h$ , then the joint basic probability that both evidence  $e_{v,q}$  and  $e_{v',q'}$  point to proposition  $h$  is given by:

$$p_{h,v,q,v',q'} = \frac{C_{h,v,q,v',q'}}{\sum_{g \in P(H)} C_{g,v,q,v',q'}} \forall h \in P(H) \quad (8)$$

**2.3.2.1. Interrelation between two pieces of evidence.** Interrelation between two pieces of evidence is obtained from single and joint basic probabilities. The degree of interdependence between evidence  $e_{v,q}(Z1)$  and evidence  $e_{v',q'}(Z2)$ ,  $Z1 \cap Z2 = h$ ,  $\forall h \in P(H)$  is given by:

$$\psi_{h,v,q,v',q'} = \begin{cases} 0 & \text{if } p_{Z1,v,q} = 0 \text{ or } p_{Z2,v',q'} = 0 \\ \frac{p_{h,v,q,v',q'}}{p_{Z1,v,q} \cdot p_{Z2,v',q'}} & \text{otherwise} \end{cases} \quad (9a)$$

where,  $\psi_{h,v,q,v',q'}$  is interdependence index. It has the following property:

$$\psi_{h,v,q,v',q'} = \begin{cases} 0 & \text{if } e_{v,q}(Z1) \text{ and } e_{v',q'}(Z2) \text{ are disjoint} \\ 1 & \text{if } e_{v,q}(Z1) \text{ and } e_{v',q'}(Z2) \text{ are independent} \end{cases} \quad (9b)$$

**2.3.2.2. Complete and incomplete data.** The MAKER framework can be implemented on complete, incomplete or ambiguous datasets. A complete dataset has records of all input attributes. For example, in [Fig. 2](#) both attribute  $A'_q$  and  $A'_{q'}$  are available, and none of them is missing. It has missing output, labelled as 'unknown', where  $unknown = H = \{h_1, \dots, h_n, \dots, h_N; n \in \{1, \dots, N\}\}$ . This type of dataset is called complete and ambiguous. The data could be unambiguous if all outputs are available. Most conventional methods utilize complete input and unambiguous output data or use imputation methods to make up missing data. However, this kind of approach could lead to loss of information and misleading results. An incomplete and ambiguous dataset has missing input and missing output, respectively. [Fig. 3](#) shows an example of an incomplete and ambiguous dataset. This dataset contains complete records for both attributes, missing records for one of the attributes and missing records for both attributes. This incomplete dataset is partitioned into three sub-tables to utilize all data records for inference. Sub-[Table 1](#) and sub-[Table 2](#) contain separate complete records for attributes  $A'_q$  and  $A'_{q'}$ , respectively. Both tables are used to find the basic probability by [Eq. \(7\)](#). Sub-[Table 3](#) contains complete records

Table: Complete Dataset		
Attributes		Output
$A'_q$	$A'_{q'}$	
$A'_{v,q}$	$A'_{v',q'}$	$h_1$
$A'_{v,q}$	$A'_{v',q'}$	$h_2$
$A'_{v,q}$	$A'_{v',q'}$	unknown
$A'_{v,q}$	$A'_{v',q'}$	$h_3$
$A'_{v,q}$	$A'_{v',q'}$	unknown
$A'_{v,q}$	$A'_{v',q'}$	$h_1$
$A'_{v,q}$	$A'_{v',q'}$	$h_2$
$A'_{v,q}$	$A'_{v',q'}$	unknown
$A'_{v,q}$	$A'_{v',q'}$	$h_1$
$A'_{v,q}$	$A'_{v',q'}$	$h_2$
$A'_{v,q}$	$A'_{v',q'}$	$h_3$
$A'_{v,q}$	$A'_{v',q'}$	$h_1$

Fig. 2. Example of complete data.

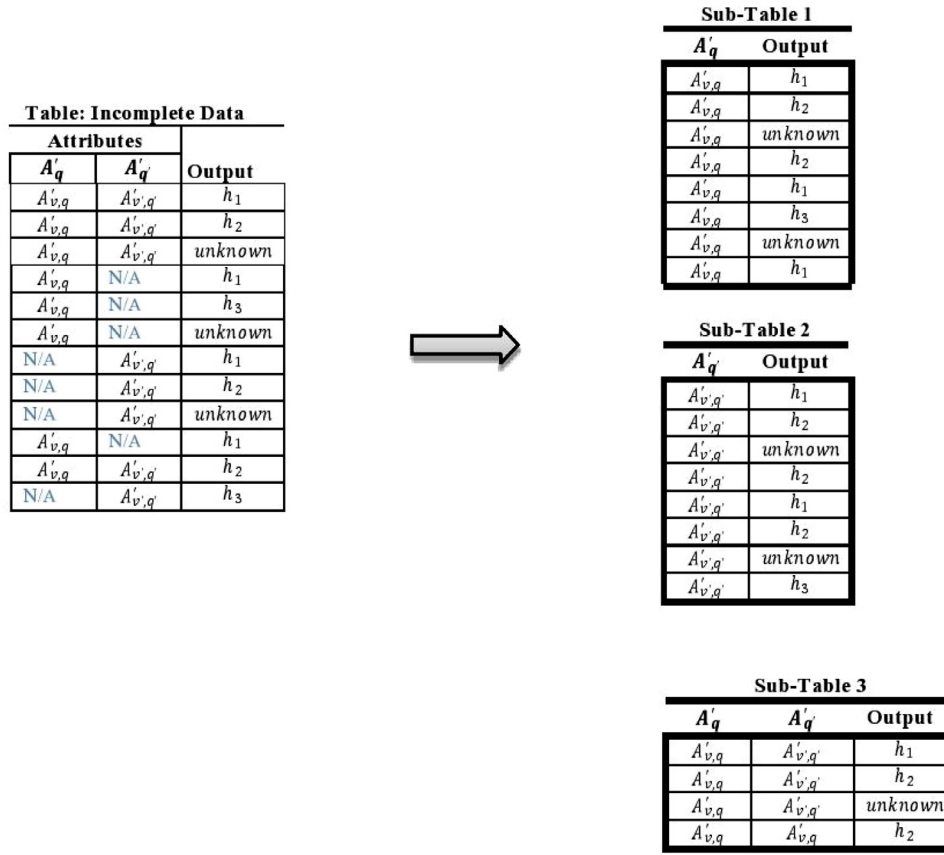


Fig. 3. Example of incomplete data.

for both attributes  $A'_q$  and  $A'_{q'}$  and is used to obtain joint basic probability by Eq. (8). A complete data in Fig. 2 does not require separation into sub-tables. The first column for  $A'_q$  and the second column for  $A'_{q'}$  is used to calculate basic probability and both columns are used to calculate the joint probability. The joint basic probability and the one-dimensional basic probability is used to obtain interrelation by Eq. (9a).

**2.3.2.3. Reliability and weight of evidence.** Joint probability given by Eq. (8) combines independent evidence under the assumption that all evidence is fully reliable. It does not consider the weight and reliability of the evidence and ignore the interrelation between pieces of evidence. Reliability represents the ability of the information source to point to a correct assessment of a given problem (Smarandache, Dezert & Tacnet, 2010). The reliability measures the degree of support for proposition  $h$ , given that evidence points to proposition  $h$ . The importance of information is indicated by the weight of evidence. Let the weight of evidence  $e_{v,q}$  pointing to proposition  $h$  be denoted by  $w_{h,v,q}$ . The probability mass that proposition  $h$  is supported by evidence  $e_{v,q}$  is:

$$m_{h,v,q} = w_{h,v,q} p_{h,v,q} \quad (10)$$

Let the reliability of evidence  $e_{v,q}$  pointing to proposition  $h$  be denoted by  $r_{h,v,q}$ . The reliability of evidence is:

$$r_{v,q} = \sum_{h \in P(H)} r_{h,v,q} p_{h,v,q} \quad (11)$$

Initial values of  $r_{h,v,q}$  and  $w_{h,v,q}$  is assumed to be equal to 1. These values are trained through supervised learning to estimate combined probability as close as possible to expected probability.

**2.3.2.4. Combine two pieces of evidence from two different attributes.** Two pieces of evidence from two different attributes are combined together by probability mass of each evidence, interrelation between evidence, joint support of evidence, weight and reliability of evidence. Suppose two pieces of evidence  $e_{v,q}$  and  $e_{v',q'}$  are from attribute  $A'_q$  and attribute  $A'_{q'}$ , respectively. The combined probability  $\alpha_{h,vq,v'q'}$  that proposition  $h$  is supported by evidence  $e_{v,q}$  and evidence  $e_{v',q'}$  is given by:

$$\alpha = \begin{cases} 0 & h = \emptyset \\ \alpha_{h,vq,v'q'} = \frac{\tilde{m}_{h,vq,v'q'}}{\sum_{h \in P(H)} \tilde{m}_{h,vq,v'q'} + \tilde{m}_{P(H),vq,v'q'}} & \forall h \in P(H), h \neq \emptyset \end{cases} \quad (12a)$$

$$\tilde{m}_{h,vq,v'q'} = [(1 - r_{v',q'}) m_{h,v,q} + (1 - r_{v,q}) m_{h,v',q'}] + \sum_{Z1 \cap Z2 = h} \gamma_{Z1,Z2,vq,v'q'} \psi_{Z1,Z2,vq,v'q'} m_{Z1,v,q} m_{Z2,v',q'} \quad (12b)$$

The residual support, which is earmarked to the power set, is given by

$$\tilde{m}_{P(H),vq,v'q'} = m_{H,v,q} m_{H,v',q'} \quad (12c)$$

The combined probability mass of evidence  $e_{v,q}$  and evidence  $e_{v',q'}$  is given by Eq. (12b). The parameter  $\gamma_{Z1,Z2,vq,v'q'}$  is the ratio of the joint reliability of the two pieces of evidence and the product of their individual reliabilities, referred to as reliability ratio for short. Z1 and Z2 show that both pieces of evidence from two different attributes point to a common proposition  $h$  in power set  $P(H)$ . The values of  $r_{h,v,q}$ ,  $r_{h,v',q'}$ ,  $w_{h,v',q'}$ ,  $w_{h,v',q'}$  and  $\gamma_{Z1,Z2,vq,v'q'}$  are trained from input-output dataset.

**2.3.2.5. Combine multiple pieces of evidence.** This framework can combine evidence from any number of attributes. Two pieces of

evidence pointing to proposition  $h$  are first combined before combining it with the third one and so on. Independent attributes can be combined in any order, while dependent attributes are combined on the basis of their interrelationships. Suppose we want to combine  $Q$  attributes  $A'_1, \dots, A'_q, \dots, A'_Q$ , ( $q \in \{1, \dots, Q\}$ ) each with  $V_q$  pieces of evidence. Single probability, i.e.  $p_{h,v,1}, \dots, p_{h,v,q}, \dots, p_{h,v,Q}$  ( $v \in \{1, \dots, V_q\}$ ,  $\forall h \in P(H)$ ), is obtained using Eq. (7). Similarly, joint probability, i.e.  $p_{h,v,1,v,2}, \dots, p_{h,v,1}, \dots, vq, \dots, p_{h,v,1}, \dots, vQ$ , is obtained using Eq. (8). The interrelation between evidence, i.e.  $\psi_{h,v,1,v,2}, \dots, \psi_{h,v,1}, \dots, vq, \dots, \psi_{h,v,1}, \dots, vQ$ , is obtained from Eq. (9). Single probability mass, i.e.  $m_{h,v,1}, \dots, m_{h,v,q}, \dots, m_{h,v,Q}$  ( $v \in \{1, \dots, V_q\}$ ,  $\forall h \in P(H)$ ), is obtained from Eq. (10). Combined probability mass is calculated iteratively once single probability, joint probability, and interrelation are calculated from the data. The degree of the combined support  $\tilde{m}_{h,v,1,v,2}$  of evidence  $e_{v,1}$  and evidence  $e_{v,2}$  for proposition  $h$  for all  $h \in P(H)$  is:

$$\begin{aligned} \tilde{m}_{h,v1,v2} = & \left[ (1 - r_{v,2}) m_{h,v,1} + (1 - r_{v,1}) m_{h,v,2} \right] \\ & + \sum_{Z1 \cap Z2 = h} \gamma_{Z1,Z2,v1,v2} \psi_{Z1,Z2,v1,v2} m_{Z1,v,1} m_{Z2,v,2} \end{aligned} \quad (13)$$

The degrees of the combined support from evidence  $e_{v1,v2}$  and evidence  $e_{v,3}$  pointing to preposition  $h$  are then given by:

$$\begin{aligned} \hat{m}_{h,v1,v2,v3} &= [(1 - r_{v,3})m_{h,v1,v2} + m_{P(H),v1,v2} m_{h,v,3}] \\ &+ \sum_{Z1 \cap Z2 = h} \gamma_{Z1,Z2,v1,v2,v3} \psi_{Z1,Z2,v1,v2,v3} m_{Z1,v1,v2} m_{Z2,v,3} \end{aligned} \quad (14)$$

The above process is repeated until all pieces of evidence are combined, to generate the degree of the combined support  $\tilde{m}_{h,v_1,\dots,v_Q}$ . It is then finally normalized (Eq. 12(a)) to obtain the combined probability  $\alpha_{h,v_1,\dots,v_Q}$ . The combined probability mass  $\tilde{m}_{h,v_1,\dots,v_Q}$  pointing to proposition  $h$  is given by:

$$\begin{aligned} \hat{m}_{h,v1,\dots,VQ} = & \left[ (1 - r_{v,Q}) m_{h,v1,\dots,VQ-1} + m_{P(H),v1,\dots,vQ-1} m_{h,v,Q} \right] \\ & + \sum_{Z1 \cap Z2 = h} \gamma_{Z1,Z2,v1,\dots,vQ} \psi_{Z1,Z2,v1,\dots,vQ} m_{Z1,v1,\dots,vQ-1} m_{Z2,v,Q} \quad (15a) \end{aligned}$$

The combined probability mass left for the power set  $\tilde{m}_{P(H), v1, \dots, vQ}$  is given by:

$$\tilde{m}_{P(H),v1,\dots,VQ} = m_{P(H),v1,\dots,VQ-1} m_{P(H),v,Q} \quad (15b)$$

The combined probability  $\alpha_{h,v1, ..., vQ}$  after normalization of combined probability mass is:

$$\begin{aligned}\alpha_{h,v1,\dots,VQ} &= \frac{\tilde{m}_{h,v1,\dots,VQ}}{\sum_{g \in P(H)} \tilde{m}_{g,v1,\dots,VQ} + \tilde{m}_{P(H),v1,\dots,VQ}} \quad \forall h \in P(H) \text{ and} \\ \alpha_{P(H),v1,\dots,VQ} &= \frac{\tilde{m}_{P(H),v1,\dots,VQ}}{\sum_{g \in P(H)} \tilde{m}_{g,v1,\dots,VQ} + \tilde{m}_{P(H),v1,\dots,VQ}}\end{aligned}\quad (15c)$$

**2.3.2.6. Training of maker model.** The MAKER framework has three types of parameters – reliability, weight, and reliability ratio. The

parameters are optimized by maximizing the likelihood of the true state.

#### 2.3.2.7. Objective function.

$$\text{Minimize : } f(\text{parameter}) = \frac{1}{2M} \sum_{m=1}^M \sum_{h \in P(H)} (\alpha - \alpha(\text{parameters}))^2 \quad (16)$$

where,  $parameters =$

$$parameters \left( \begin{array}{l} \text{Reliability : } r_{h,v,1}, \dots, r_{h,v,q}, \dots, r_{h,v,Q}, r_{h,v1,v2}, \dots, r_{h,v1,\dots vQ} \\ \text{Weight : } w_{h,v,1}, \dots, w_{h,v,q}, \dots, w_{h,v,Q}, w_{h,v1,v2}, \dots, w_{h,v1,\dots vQ} \\ \text{Reliability ratios : } \gamma_{Z1,Z2,v1,v2}, \dots, \gamma_{Z1,Z2,v1,\dots,vQ} \end{array} \right)$$

#### 2.3.2.8. Constraint.

$$\begin{aligned} 0 &\leq r_{h,v,1} \leq 1, \dots, 0 \leq r_{h,v,q} \leq 1, \dots, 0 \leq r_{h,v,Q} \leq 1, \\ 0 &\leq r_{h,v1,v2} \leq 1, \dots, 0 \leq r_{h,v1,\dots,vQ} \leq 1 \\ 0 &\leq w_{h,v,1} \leq 1, \dots, 0 \leq w_{h,v,q} \leq 1, \dots, 0 \leq w_{h,v,Q} \leq 1, \\ 0 &\leq w_{h,v1,v2} \leq 1, \dots, 0 \leq w_{h,v1,\dots,vQ} \leq 1 \\ 0 &\leq \gamma_{Z1,Z2,v1,v2}, \dots, 0 \leq \gamma_{Z1,Z2,v1,\dots,vQ} \end{aligned}$$

*Example:* Two attributes: debt searches and credit searches are taken from a business case study in this paper to demonstrate the MAKER framework for data transformation and data fusion. There are two sets of quantitative attributes  $A' = \{A'_1 = \text{debt searches}, A'_2 = \text{credit searches}\}$ . The frame of discernment is  $H = \{F, R\}$  where  $F$  is fund and  $R$  is reject. It is a complete data set. In this case, the power set is  $P(H) = \{\varphi, F, R, H\}$ . Note that, for incomplete data, power set would be  $P(H) = \{\varphi, F, R, \text{unknown}\}$ , where  $\text{unknown} = H = \{F, R\}$ . Both debt searches and credit searches have four referential values  $\{A1, B1, C1, D1\}$  and  $\{A2, B2, C2, D2\}$ , respectively. The categories  $A$  to  $B$  range from poor to excellent status.

### STEP 1: Contingency-table

Make contingency table for single attribute debt searches and credit searches and the attributes. Table 4 shows the contingency-table (frequency) of debt searches and credit searches. Table 5 shows the contingency-table of both debt and credit searches.

STEP 2: One-dimensional basic probability and joint basic probability

The likelihood table C is obtained by the division of each value in the contingency table by its row sum. One-dimensional basic probability and joint basic probability are obtained by performing division operation in each column of likelihood by sum of column by using Eq. (7) and (8). Table 6 shows the One-dimensional probability for debt and credit searches and Table 7 shows joint basic probability for both debt and credit searches.

**Table 4**  
Contingency table of an attribute.

Debt Searches	A1	B1	C1	D1	Credit Searches	A2	B2	C2	D2
<i>F</i>	0	38	409	2067	<i>F</i>	7	1206	647	654
<i>R</i>	1	48	251	685	<i>R</i>	9	447	235	294

**Table 5**  
Contingency table of two attributes.

Debt Searches	A1				B1				C1				D1			
Credit Searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
<i>F</i>	0	0	0	0	0	18	8	12	1	206	113	89	6	982	526	553
<i>R</i>	0	1	0	0	1	21	13	13	1	120	59	71	7	305	163	210

\*Some joint evidences such as (A1, A2), (A1, C2) and (A1, D2) does not exist in the data set.



**Table 6**  
One-dimensional probability ( $p_{h,v,1}$  and  $p_{h,v,2}$ ).

Debt Searches	A1	B1	C1	D1	Credit Searches	A2	B2	C2	D2
F	0.0	0.23	0.38	0.54	F	0.23	0.51	0.51	0.46
R	1.0	0.77	0.62	0.46	R	0.77	0.49	0.49	0.54

**Table 7**  
Joint Probability ( $p_{h,v1,v2}$ ).

Debt Searches	A1				B1				C1				D1			
Credit Searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	0.0	0.0	0.0	0.0	0.0	0.25	0.20	0.26	0.28	0.40	0.42	0.32	0.25	0.55	0.55	0.50
R	0.0	1.0	0.0	0.0	1.0	0.75	0.80	0.74	0.71	0.60	0.58	0.67	0.76	0.45	0.45	0.50

\*Joint probability in Table 7, interrelation in Table 8 and combined probability in Table 9 are zero for missing joint evidences in the data set.

**Table 8**  
Interrelation ( $\psi_{h,v1,v2}$ ).

Debt Searches	A1				B1				C1				D1			
Credit Searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	0.0	0.0	0.0	0.0	0.00	2.07	1.58	2.40	3.09	2.00	2.12	1.82	1.98	2.00	1.98	2.01
R	0.0	2.05	0.0	0.0	1.71	2.02	2.19	1.80	1.53	2.01	1.94	2.06	2.13	1.98	2.00	2.01

**Table 9**  
Combined probability ( $\alpha_{h,v1,v2}$ ): Before training.

Debt Searches	A1				B1				C1				D1			
Credit Searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	0.0	0.0	0.0	0.0	0.0	0.25	0.19	0.25	0.28	0.41	0.42	0.32	0.26	0.55	0.55	0.51
R	0.0	1.0	0.0	0.0	1.0	0.75	0.81	0.75	0.71	0.59	0.58	0.67	0.74	0.45	0.45	0.49

**Table 10**  
Combined probability ( $\alpha_{h,v1,v2}$ ): After training.

Debt Searches	A1				B1				C1				D1			
Credit Searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	0.23	0.13	0.59	0.53	0.01	0.40	0.28	0.34	0.36	0.62	0.59	0.52	0.51	0.81	0.78	0.76
R	0.77	0.87	0.41	0.41	0.99	0.60	0.72	0.66	0.64	0.38	0.41	0.48	0.49	0.19	0.22	0.24

### STEP 3: Interrelation

Interrelation between two pieces of evidence is calculated from Eq. (9a and 9b). Table 8 shows the interrelation between two pieces of evidence from debt and credit search attribute.

### STEP 4: Set Initial value of parameters

All initial values of weight, reliability and reliability ratios are assumed to be equal to 1. The table for initial weight and initial reliability of single and joint evidence has the same structure as in Tables 6 and 7 and the only difference is that they are initially filled with 1 (see Appendix B: Table 20 to Table 23). Similarly, the structure of the table for initial reliability ratios is similar to Table 7, and the only difference is that it is filled with 1 before training (see Appendix B: Table 24).

### STEP 5: Initial probability mass

In this case, probability mass is obtained by Eq. (10) as the multiplication of basic probability and weight of evidence. Before optimization, the initial value of probability mass for single evidence, debt searches or credit searches, is the same as the basic probability because the initial value of weight is equal to 1 that is multiplied by basic probability.

### STEP 6: Calculate combined probability before training

The combined probability is obtained from Eq. (12a) to (12c). First, use Eq. (12b) to calculate probability mass and Eq. (12c) to

calculate residual support. Then, calculate the combined probability by Eq. (12a), which normalizes probability mass by the sum of all probability mass of all evidence and residual support. Table 9 shows the combined probability before training.

### STEP 7: Training/optimization of MAKER

Table 10 shows the combined probability after training. The value of trained parameters is shown in Appendix B: Table 25 to Table 29. The dataset has missing joint probability due to missing joint evidence in the dataset, shown in Table 5. All single evidence was available, shown in Table 4. After optimization, the combined probability of supporting each state is predicted. It can be seen that zero joint probabilities in Table 9 are also filled with the combined probabilities after training in Table 10. This is because none of the variables is fully reliable and the joint probability is not fully reliability either in deciding whether the two variables support each state or not.

### STEP 8: Data Transformation

The belief-rule-based system takes input in the form of belief-distribution. The combined probability obtained from the MAKER framework is used to transform data of one or more attributes (or variables) into belief-distribution. Table 11 shows an example of transformation data from two attributes debt and credit searches into a belief-distribution by using trained combined probability in

**Table 11**  
Example of data transformation by MAKER.

# data points	Debt Searches	Credit Searches	Transformed data (Fund, Reject)
1	A1	B2	{(F, 0.13), (R, 0.87)}
2	C1	B2	{(F, 0.62), (R, 0.38)}
3	B1	B2	{(F, 0.40), (R, 0.60)}
4	D1	A2	{(F, 0.51), (R, 0.49)}
⋮	⋮	⋮	⋮

**Table 10.** The belief-distribution add explainability in the BRB system. For example, in Table 11 probability of fund and reject of first loan application (debt searches = A1 and credit searches = B2) is 0.13 and 0.87, respectively. Similarly, probability of fund and reject of second loan application (debt searches = C1 and credit searches = B2) is 0.62 and 0.38, respectively and so on. The data transformed by MAKER is fed into the BRB system. The transformed data activates multiple rules in the BRB system and belief-degree of activated rules are combined by an inference engine.

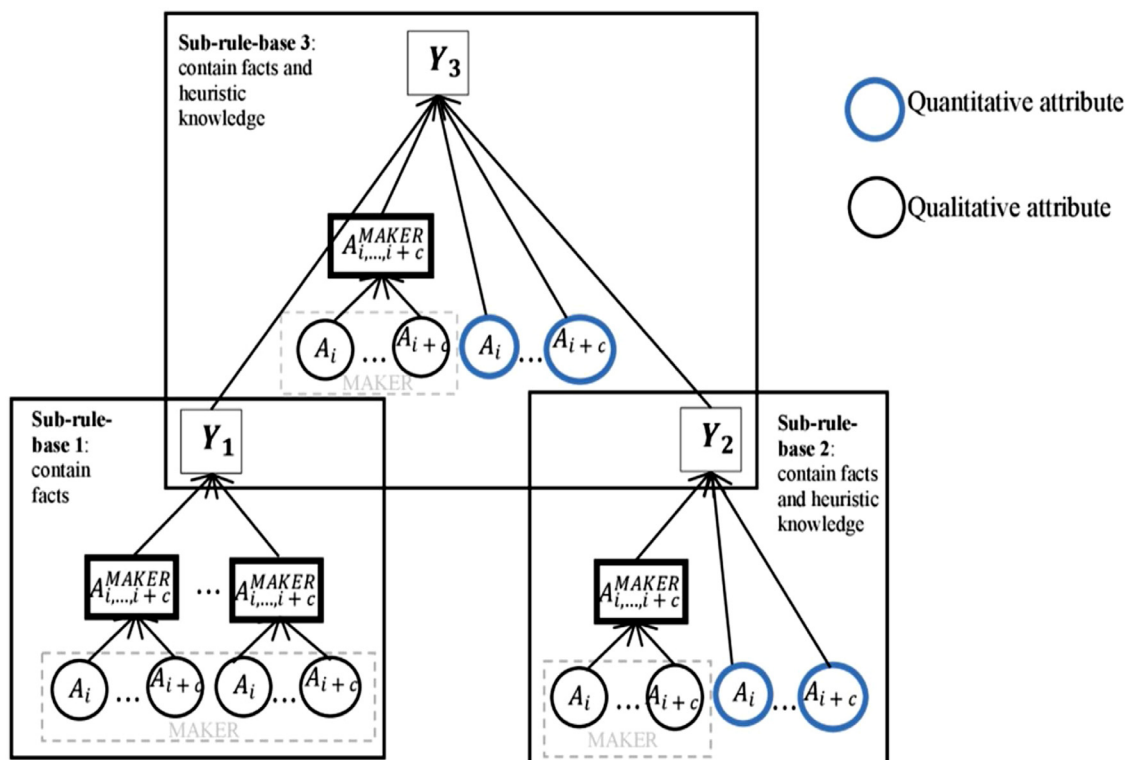
## 2.12. Hierarchical structure of knowledge-base

The underwriting decision-making process is complex and requires a large number of input variables to reflect the underwriter knowledge and guidelines. The number of rules increases exponentially with the numbers of attributes and their referential values, as explained in para *Data fusion – Maximum Likelihood Evidence Reasoning*. Rules in the knowledge base are all possible combinations of referential values of attributes. The number of training parameters increases with the increase in the number of rules. A hierarchical knowledge base can be used to reduce the number of training parameters in the BRB model by focusing on modeling the parts of the relationship between inputs and output, which are revealed by data. It is a bottom-up approach. The pieces of evidence at the lower sub-rule-base are aggregated first and then used as the evidence at a higher sub-rule-base. In each sub-rule base, at-

tributes similar in context or highly interrelated can be fused together by the MAKER framework to make the most of data for likelihood analysis of data and also to reduce the number of attributes in each sub-rule-base.

For underwriting decision making, a lower sub-rule-base can have factual rules. Factual rules are hard knowledge and do not follow any heuristic principles. Facts can be assessed first before moving up to the higher sub-rule-base consisting of heuristic knowledge. It has rules which do not obey strict or hard rules; they depend on approaches by which humans solve the problem (Gomez, Hull, Karr, Hosken & Verhagen, 1992).

Fig. 4 shows an example of a hierarchical knowledge base. It has three sub-rule-bases, with the MAKER framework as the first layer to fuse the attributes that are closely related. Sub-rule-base 1 contains only factual rules; it is at the lower level of the hierarchy. All facts are tested before moving up to sub-rule-base 3. Sub-rule-base 2 contains both facts and heuristic rules. It is independent of sub-rule-base 1. Information is processed separately in both sub-rule-bases 1 and 2 before propagating from the lower level to the higher level. Sub-rule-base 1 does not require training or learning by optimization since it only contains facts or hard knowledge. A factual rule has 100% degree of belief for a value in the set of consequence. For example, suppose there are factual rules “IF number of bankruptcy in the last 3 year  $\geq 2$  and affordability = fail, THEN [(fund,0),(reject,1)]” and “IF number of bankruptcy in the last 3 year  $\geq 2$  and affordability = pass, THEN [(fund,0),(reject,1)]”. If



**Fig. 4.** Example of hierarchical structure of knowledge base.

**Table 12**  
Rules in knowledge base.

Rules Number	Rules				Belief Degree				
1	$A_1$	$\dots$	$A_i$	$\dots$	$A_I$	$\beta_1$	$\dots$	$\beta_n$	$\beta_N$
$\vdots$	$A_{v,1}^1$		$A_{v,i}^1$		$A_{v,I}^1$	0.10	$\dots$	0.90	$\dots$
$\vdots$	$\vdots$		$\vdots$		$\vdots$				
$k$	$A_{v,1}^k$	$\dots$	$A_{v,i}^k$	$\dots$	$A_{v,I}^k$	0.00	$\dots$	1.00	$\dots$
$\vdots$	$\vdots$		$\vdots$		$\vdots$				
$K$	$A_{v,1}^K$		$A_{v,i}^K$		$A_{v,I}^K$	0.80	$\dots$	0.10	$\dots$

any of the conditions are not satisfied, then the degree of belief for rejection is 1 (or 100%). In this example, “number of bankruptcy in the last 3 years  $\geq 2$ ” in both rules does not satisfy the funding decision. Note that the implementation of the MAKER framework to combine attributes is optional, depending upon whether there is sufficient data for likelihood data analysis. If all attributes in factual-rule-base are qualitative attributes, it will activate only one rule. More than one rule is activated if a rule-base contain one or more quantitative attributes.

### 2.13. Inference

The transformed data is stored in a primary database. A data point is a vector  $x_m = \{x_{m1}, \dots, x_{mi}, \dots, x_{mI}\}$  containing values of each attribute, shown in Table 1. A data set has  $M$  data points,  $m \in 1, \dots, M$  for  $I$  attributes,  $i \in 1, \dots, I$ . The set of attributes has  $Q$  qualitative attributes and  $J$  quantitative attribute,  $I = Q + J$ . Each attribute has  $V_i$  referential values, where  $v \in 1, \dots, V_i$ . A transformed data point  $x_m = \{(A_{1,1}, \alpha_{1,1}), \dots, (A_{v,1}, \alpha_{v,1}), \dots, (A_{1,i}, \alpha_{1,i}), \dots, (A_{v,i}, \alpha_{v,i}), \dots, (A_{1,I}, \alpha_{1,I}), \dots, (A_{v,I}, \alpha_{v,I})\}$  contains the matching degree of each referential value of an antecedent attribute. The transformed data is sent to the inference engine where the referential values of each antecedent attribute are matched with the referential values of the IF-THEN rules in the knowledge base, shown in Fig. 1. Rules in knowledge-base are illustrated in Table 12. The degrees of belief for each rule can be given by experts initially. If expert opinion is not available, then the degrees of beliefs are set randomly and trained later. Rules in knowledge-base represent the consequent distributions for the combinations of the set of referential values  $(A_{v,i})$  of each attribute  $A_i$ ,  $i \in 1, \dots, I$ . The number of rules in a knowledge-base is given by Eq. (3).

At least one rule is activated by the transformed vector of input data point  $x_m$ . The activation weight of  $k^{th}$  rule by input data point  $x_m$  is given by

$$W_k(x_m) = \frac{\theta_k \prod_{i=1}^I (\alpha_{v,i})^{\bar{\delta}_i^k}}{\sum_{k=1}^K \left[ \theta_k \prod_{i=1}^I (\alpha_{v,i})^{\bar{\delta}_i^k} \right]} \text{ and} \quad \bar{\delta}_i^k = \frac{\delta_i^k}{\max_{i \in \{1, \dots, I\}} \{\delta_i^k\}} \quad \forall k \in \{1, \dots, K\} \quad (17)$$

where  $\theta_k \in [0, 1]$  ( $k \in \{1, \dots, K\}$ ) is rule weight.  $\delta_i^k \in [0, 1]$  ( $i \in \{1, \dots, I\}$ ) is the weight of antecedent attribute in the  $k^{th}$  rule. The matching degree of each attribute is denoted by  $\alpha_{v,i}$  and  $\prod_{i=1}^I (\alpha_{v,i})^{\bar{\delta}_i^k}$  is the combined matching degree. The final inference output  $o(\hat{y}_m)$  is generated by aggregating all the rules activated by the transformed input data  $x_m$ . The inference output  $o(\hat{y}_m)$  generated by  $x_m$  can be represented in the following way:

$$o(\hat{y}_m) = \{(h_n, \beta_n(x_m)), n \in \{1, \dots, N\}\} \quad (18)$$

Since the rules are generated independently, the analytical evidential reasoning (ER) approach can be used for the inference of output. The aggregated degree of belief by the ER approach is calculated by the following expression:

$$\beta_n(x_m) = \mu \times \left[ \prod_{k=1}^K \left( W_k(x_m) \beta_{n,k} + 1 - W_k(x_m) \sum_{n=1}^N \beta_{n,k} \right) - \prod_{k=1}^K \left( 1 - W_k(x_m) \sum_{n=1}^N \beta_{n,k} \right) \right] \quad (19)$$

Where

$$\mu = \left[ \sum_{n=1}^N \prod_{k=1}^K \left( W_k(x_m) \beta_{n,k} + 1 - W_k(x_m) \sum_{n=1}^N \beta_{n,k} \right) - (N-1) \prod_{k=1}^K \left( 1 - W_k(x_m) \sum_{n=1}^N \beta_{n,k} \right) - \prod_{k=1}^K (1 - W_k(x_m)) \right]^{-1}$$

The derivation of activation weight and aggregated belief degree is shown in Yang et al. (2006).

### 2.14. Explainability of brb

Each data point  $x_m = \{x_{m1}, \dots, x_{mi}, \dots, x_{mI}\}$  indicates a unique entity which contains different values in each attribute. The data point  $x_m$  transformed over different referential values of the attributes contains different matching degree  $\alpha_{v,i}(x_{m,i})$ . Therefore, rules activated by a data point  $x_m$  have different activation weight. The importance of rule can be measured by activation weight of the rule. In other words, the importance of an activated rule is equal to the activation weight of the rule. The importance of an activated rule is given by

$$k(x_m) = W_k(x_m) \quad (20a)$$

$$\sum_{k=1}^K \mathbb{I}_k(x_m) = 1 \quad (20b)$$

where,  $\mathbb{I}_k(x_m)$  is the importance of  $k^{th}$  rule activated by data point  $x_m$ . The sum of importance of all rules activated by a data point  $x_m$  is equal to one, shown in Eq. (20b).

In BRB model, the contribution of an attribute in an activated rule can be found by the weight of the attribute in the rule and their matching degree. The contribution of referential value of antecedent attributes in an activated rule is given by

$$\Delta_{v,i}^k(x_m) = \delta_i^k \times \alpha_{v,i} \quad (21a)$$

$$\hat{\Delta}_{v,i}^k = \frac{\Delta_{v,i}^k(x_m)}{\sum_{i=1}^I \Delta_{v,i}^k(x_m)} \quad (21b)$$

$$\sum_{i=1}^I \hat{\Delta}_{v,i}^k(x_m) = 1 \quad (21c)$$

where,  $\Delta_{v,i}^k(x_m)$  is importance of  $v^{th}$  referential value of the  $i^{th}$  antecedent attribute in the  $k^{th}$  rule activated by a transformed data point ( $x_m$ ). The importance of attributes in the  $k^{th}$  rule are normalized to one (Eq. (21b) and (21c)).

## 2.15. Training of BRB model

There are four types of parameters – rule weight ( $\theta_k$ ), attribute weight ( $\delta_i^k$ ), belief degree ( $\beta_{n,k}$ ), and referential value of quantitative attribute ( $A_{v,j}$ ). The initial belief degrees and referential values of quantitative attributes can be given by experts. The initial rule weights and attribute weights are assumed to 1. If expert opinions are not available, random belief degrees for each rule can be used as the initial start point for optimization. The training process improves the accuracy of the BRB model. The objective function minimizes the distance between the observed value  $y_m$  and the expected value  $\hat{y}_m$ . The expected value  $\hat{y}_m$  is a function of parameters ( $\beta_{n,k}$ ,  $\theta_k$ ,  $\delta_i^k$ ,  $A_{v,j}$ ). Details about training and inference for BRB models can be found in the literature (Chen, Yang, Xu, Zhou & Tang, 2011). The objective function and constraints are shown below.

## 2.16. Objective function

$$\text{Minimize } f(\text{parameter}) = \frac{1}{M} \sum_{m=1}^M (y_m - \hat{y}_m(\text{parameters}))^2 \quad (22)$$

where,  $\text{parameters} = \text{parameters}(\beta_{n,k}, \theta_k, \delta_i^k, A_{v,j})$

Constraints:	Number of constraints
<b>Bound Constraints</b>	
$0 \leq \beta_{n,k} \leq 1$	$N \times K$
$0 \leq \theta_k \leq 1$	$K$
$0 \leq \delta_i^k \leq 1$	$I \times K$
<b>Equality Constraints</b>	
$\sum_{n=1}^N \beta_{n,k} = 1$	$K$
<b>Fixed Parameters</b>	
-first referential value of quantitative attribute $A_{1,j} = \text{lowest\_value}_{1,j}$	$J$
-Last referential value of quantitative attribute $A_{V,j} = \text{highest\_value}_{V,j}$	$J$
<b>Inequality Constraints</b>	
$A_{v,j} < A_{v+1,j}$ for all numerical attributes $j, v = 2, \dots, V_j - 1$	$\sum_{j=1}^J V_j - 2$

## 3. Case study: first charge residential mortgage loans

This case study demonstrates the application of the decision support system to a financial product - first charge residential mortgage loan of Together Financial Services, UK. Together Financial Services specialize in offering a 'common sense lending' approach for assessing both standard and non-standard cases such as non-standard properties, non-standard income/employment, and less than perfect credit score. The underwriting process at Together Financial Services is very detailed and individualistic. It requires underwriters to follow underwriting process guidelines and analyze a large amount of information provided in a loan application and credit history – repayments and defaults of each applicant in a loan application. This research was conducted for Together Finance Services to support their options in future to rely more on technology to improve the quality of decision and decision cycle time.

### 3.1. BRB decision support system for mortgage lending

Together Financial Services rely on external credit reporting agencies to obtain the credit history of each applicant in a loan application. They have two data sources: credit bureau and electronic-application. The electronic-application contains personal information about applicants in a loan application. Credit dataset

had 1800 columns and each row is a unique search for an applicant in a loan application. Each row has a unique ID called 'ApplicantID' as the primary key and each unique loan application is given a unique number called 'Account\_ID' as a foreign key. Together Financial Services receive all online applications and aim to process an application in a couple of days. The data in a loan application related to monthly expenditures, income, and loan information with property details are extracted from the electronic application. At Together Financial Services, manual underwriting task is initiated only when a loan application passes the affordability test as shown in Fig. 5. Affordability calculator assesses income (joint income for more than one applicant in an application) and monthly expenditure against minimum expected value for an applicant. The minimum value for expenditure depends on the number of dependents and minimum acceptable values for monthly expenditures is defined by a financial institution. All expenditures, income (income after retirement in case loan repayment extends into retirement period), and loan amount are used to adjust the Stressed Maximum Affordability Monthly Repayment (MAMR) factor. A negative MAMR and/or one or more expenditure below minimum value result in failed affordability criteria, which causes the automatic decline of an application.

The hierarchical structure of the decision support system is shown in Fig. 5. The lower sub-rule base has factual rules. This rule base is deployed to automate the acceptance or decline of loan application. There are four attributes in factual-rule-base – affordability test, number of bankruptcy  $\geq X$  and/or amount of bankruptcy  $\geq \text{£}X$ , number of individual voluntary arrangements (IVA) and county court judgments (CCJ)  $> X$ , and number of payday loans in past  $Y$  years  $> X$ . The affordability test from affordability calculator provide result in either 'pass' or 'fail' and other three attributes are obtained from credit data either 'NONE' or ' $\geq 1$ ', where 'NONE' means that none of the applicants had these credit default in the past and ' $\geq 1$ ' means that one or more than one applicant in a loan application had these credit default in the past. Together Financial Services accept an application for decision making by their underwriters only when all the applicants satisfy these rules. An application is declined if the affordability test is 'fail' or other three attributes on bankruptcy, CCJ, and payday loan is ' $\geq 1$ '. A loan application is accepted only when the affordability test is 'pass' and other attributes are 'NONE'. The factual-rule-base has 16 rules ( $2^4 = 16$  rules, as each attribute has two categories and there are four attributes), and the rules in this rule-base do not require training (optimization) as they are facts.

The higher sub-rule-base contains heuristic knowledge. The heuristic-rule-base maps the decision-making strategy of underwriters. The underwriter's knowledge acquisition and data collection process are explained in Section 2.4. The knowledge acquisition process in this research requires a thorough understanding of underwriting guidelines to select attributes from credit data as well as recommendations from several experienced underwriters at Together Financial Services. Each underwriter demonstrated a few case studies, which were used to exemplify the underwriting process and decision-making strategy for different cases. The attributes  $A_1, A_2, \dots, A_{12}$  in Fig. 5 are fused and transformed by the MAKER framework described in para *Data fusion – Maximum Likelihood Evidence Reasoning*. The MAKER framework provides six aggregated attributes  $MK_1, MK_2, \dots, MK_6$ . The heuristic-rule-base has 64 rules ( $2^6 = 64$  rules, each attribute has two categories (F, R) and there are six attributes). The rule weights, attribute weights and belief degrees in the heuristic-rule-base are trained by the training dataset. The loan criteria checker (attribute A10) assures that loan fits within the loan plans established by Together Financial Services based on property type, property valuation, and the total number of defaults in the past. The attributes names are not revealed to maintain data and business confidentiality; however, this



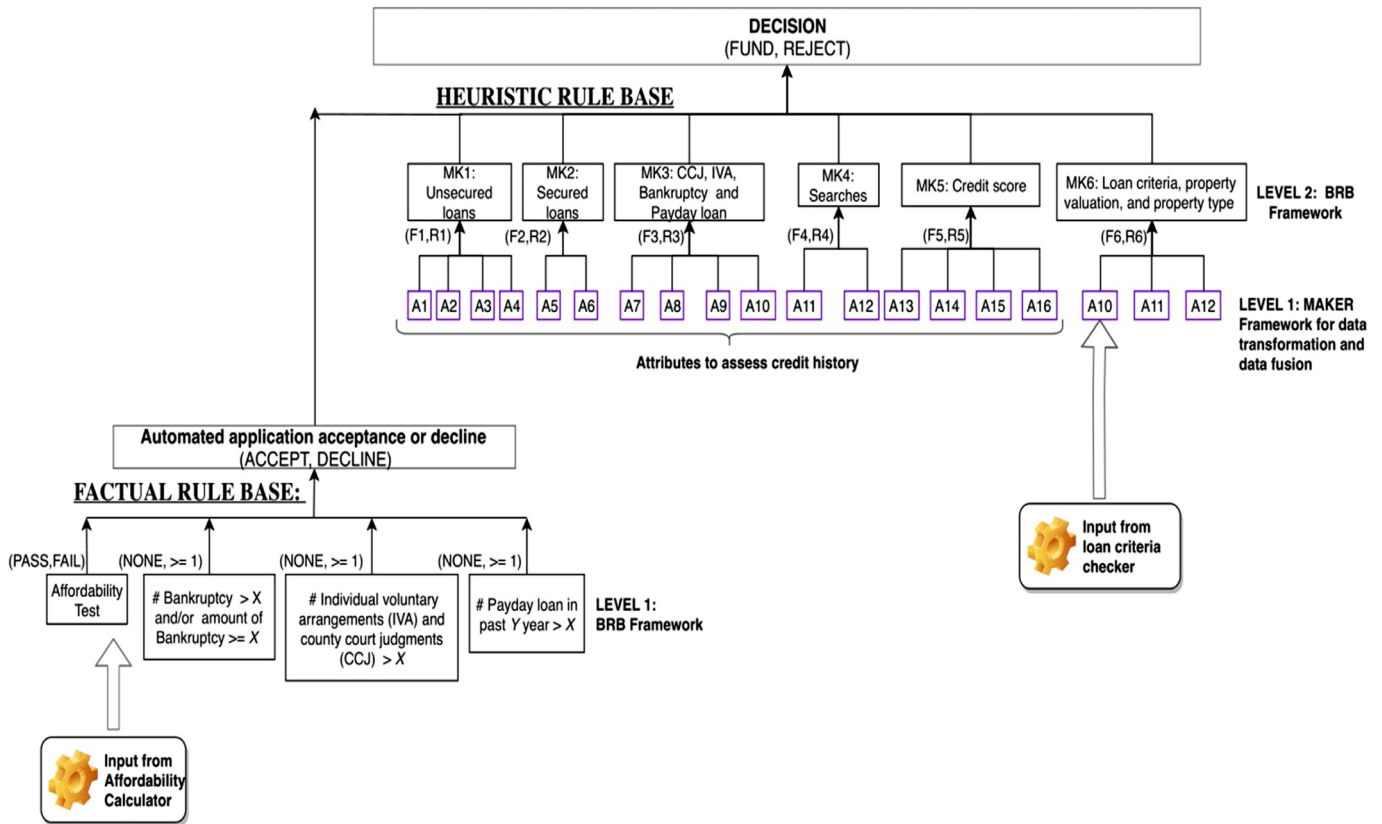


Fig. 5. Structure of decision support system.

case study has revealed the types of attributes used to develop this automated loan lending system for this company. The attributes used in this system are related to affordability, unsecured loans, secured loans, bankruptcy, searches, credit score, property, and loan criteria.

The system evaluates the creditworthiness of an applicant based on her past records on unsecured loans, secured loans, searches (debt and credit), credit score and CCJ, IVA, bankruptcy and payday loans. A loan application is assessed as a whole by securitizing the credit history of each applicant in an application. The pieces of evidence for each evaluation factor can be aggregated to obtain joint evidence for each evaluation factor to assess the entire loan application, as shown in Fig. 6. The aggregation of each creditworthiness evaluation factor is done by the evidential reasoning rule (Yang & Xu, 2013). This approach is also used in BRB to aggregate the belief degrees of activated rules. In this research, applicants are given the same weight. To consider special cases, some applicants can be given higher weights than other applicants. For example, only the main applicant is employed in a loan application, or one of the applicants is retired. Some of the attributes are analyzed by affordability calculator to compute MAMR. These attributes can be accommodated in a rule-base in absence of affordability calculator in similar implementations by other businesses.

### 3.2. Training and validation

The rules in factual-rule-base are shown in Table 17 of Appendix A. These rules are facts and do not require training. In this case study, all attributes in factual-rule-base are qualitative attributes, it will activate one rule for each loan application. More than one rule is activated if rule-base contains one or more quantitative attributes. The rule weight, attribute weight, and degree of belief of rules in the heuristic-rule-base are tuned by training data.

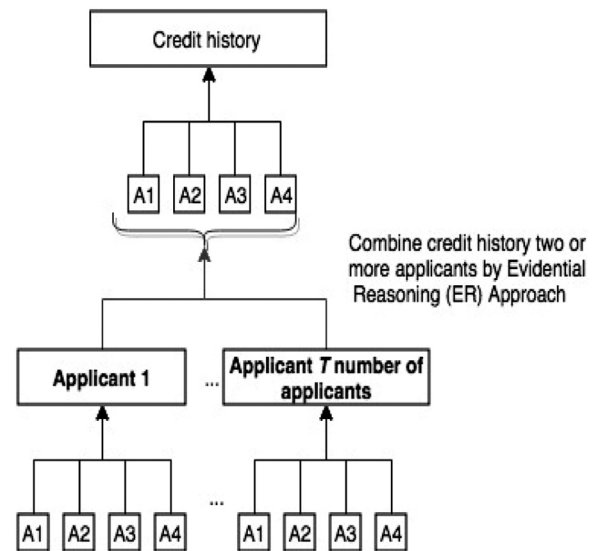


Fig. 6. Combine the credit history of two or more applicants.

It is not mandatory to train all the rules or all parameters of a rule in heuristic-rule-base. The belief degree, attribute weight and rule weight provided by experts in a rule can be kept constant.

Lending institutions change their lending policy from time to time. In finance, AI/ML decision making primarily depends on supervised learning model. However, change in policy and practices impacts the adoption of such models. A change in a policy discards some of the attributes in the data which reflect the practices under the preceding policy. This circumstance could be mitigated by updating training data if a referential value of attributes is turned into a strict policy. For example, rejection of all loan

**Table 13**  
3-Fold cross-validation of heuristic-rule-base.

Fold number	Size of training data in a fold	Size of test data in a fold	Accuracy	Precision	True positive (fund) rate or Recall	True negative (reject) rate	F-Score
1	2110	1055	0.956	0.956	0.9906	0.8282	0.9729
2	2110	1055	0.957	0.9549	0.9906	0.8434	0.9750
3	2110	1055	0.952	0.9600	0.9867	0.8232	0.9705

\*Total number of instances were 3498. The factual-rule-base declined 333 applications. The heuristic-rule-base is trained with the remaining 3165 instances.

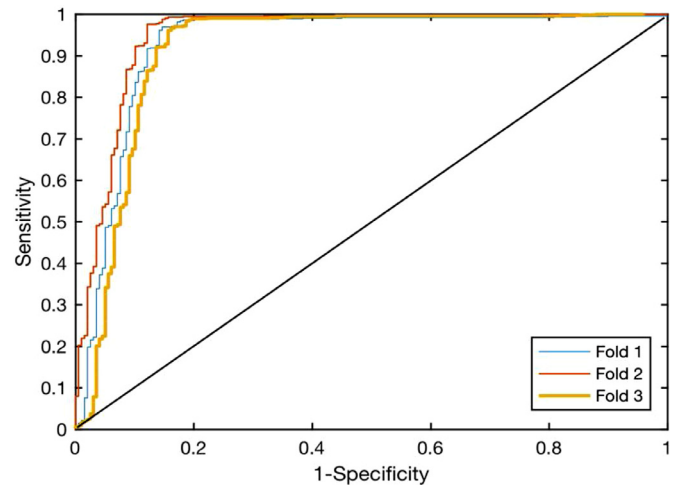
**Table 14**  
Bias and variance of test and training set in each fold of cross-validation set.

Fold number	Training set error	Test set error	Bias	Variance
1	0.040	0.044	Low	Low
2	0.035	0.040	Low	Low
3	0.041	0.045	Low	Low
Average	0.038	0.043		

applications with more than 2 defaults in unsecured loan in past 1 year is a stringent criterion. The training data can be updated by modifying labels of such instances as rejected (R) and then a supervised learning model can be retrained. In BRB referential value of such attribute can be used as factual rules; if not then heuristic-rule-base can be retrained by newly labelled data. The heuristic-rule-base reflects underwriter judgment. If underwriting practices change dramatically over a short period of time, then heuristic-rule-based with expert underwriter opinion can be used for decision-making until adequate data is available for parameter training.

The untrained or initial parameters of the rules in the heuristic-rule-base are shown in Table 16. Table 17 shows the parameters of the trained heuristic-rule-base. The initial degrees of belief in Table 16 are assessed by underwriters, which is a good starting point for the optimization of BRB parameters. The heuristic rules in the BRB system used for the operation are trained by all available data and performance was evaluated by k-fold cross-validation. The method for validation of a supervised learning model is discussed by Cawley and Talbot (2010). The dataset had 3498 instances of past loan application data. The factual-rule-base declined 333 applications and accepted 3165 applications. All parameters in heuristic-rule-base were trained by these 3165 instances. In this case-study, 3-fold of the dataset was used for cross-validation. The data is shuffled and partitioned into three groups. In each fold, 2/3 of data was utilized for training and 1/3 was utilized to evaluate the performance of the test set. The performance of the model was evaluated by bias and variance. Bias is the error in fitting the model by learning algorithm. Variance is the change in error when it is tested with unseen data compared to training data. Ideally, a model should have low bias and low variance, which suggests that the model has captured the underlying pattern of the data and model performance does not vary considerably compared to its performance for training dataset, respectively. Table 13 shows the result of 3-fold cross-validation of heuristic-rule-base. It can be seen that the accuracy 3-folds does not vary a lot. The second fold has good performance, especially for true rejection rate. In each fold, test data had 198 rejection instances, the heuristic-rule-base has 166 (83.83%) true average rejection rate. The combined rejection rate of factual and heuristic rule base is around 88.43%.

Table 14 show that all 3-folds has low bias and low variance as a consequence of low error and low variance of error between test and training data, respectively. The over-all model bias is evaluated by finding the mean error of k-fold test dataset. The average test data error of 0.043 ensures that the model is trained well, and data is precise enough to capture pertinent patterns into the rules. The



**Fig. 7.** ROC curves.

variance of k-fold test error is equal to  $4.67 \times 10^{-6}$ . Low variance between training and test folds and low variance in test error as certain that model performance would not vary when it is trained with all available data set.

The ROC curve of the test data of 3-fold cross-validation is shown in Fig. 7. As discussed before, the system in operation is trained with all available data. The visualization of the confusion matrix is shown in Fig. 8, where the x-axis is the degree of belief at which "Fund" decision is made and y-axis represents the number of decisions made at a particular degree of belief. It shows that 333 applications were automatically declined in the early stage by the factual-rule-base and other applications were processed by the heuristic-rule-base. The frequency of true fund and true rejection has the degree of beliefs ranging from 0.85 to 1 and 0.04 to 0.15, respectively. Some of the rejection reasons that this system cannot predict to reject loan applications are inconsistency in a file, change in customer circumstances, customer integrity concerns, loan no longer customers best interest. The system gives false-fund decisions due to the inability to predict these rejection reasons.

### 3.3. Performance comparison

The performance of the BRB system was compared with five different machine learning models. The performance of the classifiers was tested with 3-fold cross-validation method. The data sample in each fold was the same as the data in each fold used for evaluation of BRB model in Table 13. An efficient architecture or design of each machine learning model was selected by tuning its hyper-parameters. The average performance metrics of 3-fold cross-validation for these machine learning models is shown in Table 15. It can be seen that the accuracy and F-score of BRB model is slightly higher than deep-neural-network (DNN). Precision in DNN is 0.090% higher than BRB; however, recall is 0.66% higher in BRB. Recall measures the cost of rejecting potential loan applications whereas precision measures the cost of funding a poor loan application. F-score is the balance between precision and re-

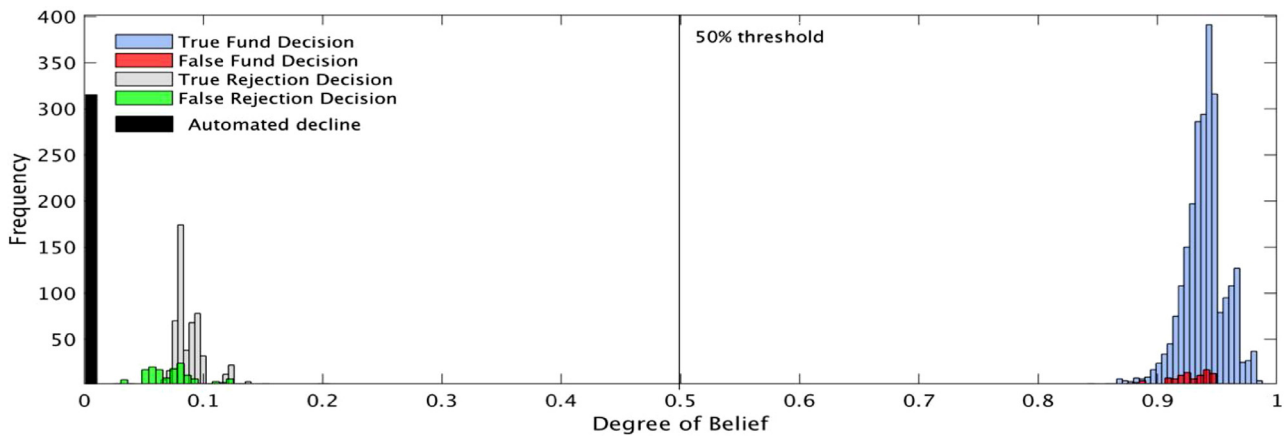


Fig. 8. Visualization of confusion matrix.

**Table 15**  
Performance comparison.

Machine learning Model	Accuracy	Precision	Recall	F-Score
Decision Tree	0.8767	0.9900	0.8747	0.9288
Random Forest	0.9303	0.9883	0.9302	0.9584
XGBoost (Extreme Gradient Boosting)	0.9471	0.9883	0.9487	0.9681
SVM (Support Vector Machine)	0.9460	0.9893	0.9466	0.9675
DNN (Deep Neural Network)	0.9527	0.9902	0.9504	0.9714
BRB (Belief-Rule-Base)	0.9550	0.9893	0.9568	0.9728

**Table 16**

Data transformed of searches attribute by MAKER for application 3 to 6 in Fig. 9.

Application	Debt Searches	Credit Searches	MK4: (Fund, Reject)
3	D1	C2	{{(F, 0.78), (R, 0.22)}}
4	D1	D2	{{(F, 0.76), (R, 0.24)}}
5	D1	B2	{{(F, 0.81), (R, 0.19)}}
6	C1	B2	{{(F, 0.62), (R, 0.38)}}

call. Among all models, it is the highest in BRB and the lowest in the decision tree. The tree-based ensemble methods, random forest and XGBoost performed better than the decision tree but lower than BRB and DNN.

### 3.4. Explainability of BRB for mortgage lending

Fig. 9 illustrates the explanation provided by the BRB system for some of the applications. Figs. 10–17 show the importance of rules and contribution of attributes in the activated rules.

The MAKER framework was used to combine and transform attributes, related to unsecured loans, secured loans, bankruptcy & payday loans, searches, credit score, and property & loan criteria. It transforms attributes to belief-distribution over a set of referential values of consequence attribute. The referential values of consequence attribute are fund (F) and reject (R). Table 10 in para (Data fusion – Maximum Likelihood Evidence Reasoning) shows the probability to fund and reject loan applications based on joint evidence from two attributes, debt search and credit search. This table was used to transform the attribute searches for heuristic-rule-base. Similarly, all other attributes were combined and then transformed by MAKER framework. In Fig. 9, application 1 and application 2 were rejected by factual rules whereas application 3 to 6 passed the strict rules in factual-rule-base. Table 16 shows the transformed data of the attribute searches (MK4) for application 3 to 6. The values in Fig. 9 are matching degrees. The BRB system designed in this case study can be explained, first by the rules activated in the factual-rule-base, second by the data transformed by combined probability obtained from the MAKER framework, and

third by the importance of activated rules in the heuristic-rule-base and the contribution of attributes in activated rules. Different types of statement for textual explanation are added in the system for different outcomes of the MAKER framework and important rules and its attributes in the BRB.

- Application 1 & 2: Application 1 had automated rejection due to failed affordability test and application 2 had automated rejection due to an inadequate number & amount of bankruptcy and an inadequate number of payday loans.
- Application 3:
  - Factual-rules-base: passed all rules
  - Level 1 MAKER in heuristic-rule-base: In Fig. 9, all attributes point towards the high probability of funding this loan application.
  - Level 2 BRB in Heuristic-rule-base: Among all rules, rule 1 had the highest importance in providing the decision, shown in Fig. 10. It has F code (fund) in all the attributes (rules shown in Table 19). In this rule, property & loan criteria (MK6) has the highest contribution of 0.28 followed by unsecured loans (MK1), shown in Fig. 11.
- Application 4:
  - Factual-rules-base: passed all rules
  - Level 1 MAKER in heuristic-rule-base: In Fig. 9, property & loan criteria point towards rejection and all other attributes point towards the funding of this loan application.
  - Level 2 BRB in heuristic-rule-base: Rule 33 has the highest importance, then rules 49, 34 and 37, shown in Fig. 12. All rules have R code in loan criteria & property attributes. In rules 33, 49, 34, and 37 property & loan criteria (MK6) has the highest contribution towards rejection of about 0.219, 0.231, 0.233, and 0.236, respectively, shown in Fig. 13. The R code also exists in credit score (MK5), unsecured loans (MK1) and CCJ, IVA& bankruptcy (MK3) in rules 49, 34, and 37, respectively, but their contribution towards rejection is very small (approximately 0.080).
- Application 5:

Decision Explanation Illustrator													
	Application 1		Application 2		Application 3		Application 4		Application 5		Application 6		
<b><i>Factual Rule Base</i></b>													
Affordability test	FAIL		PASS		PASS		PASS		PASS		PASS		
Number and amount of bankruptcy	NONE		>=1		NONE		NONE		NONE		NONE		
Number of IVA & CCJ	NONE		NONE		NONE		NONE		NONE		NONE		
Number of payday loans	NONE		>=1		NONE		NONE		NONE		NONE		
<b>Decision: Automated Application Acceptance or Decline</b>	DECLINE		DECLINE		ACCEPT		ACCEPT		ACCEPT		ACCEPT		
<b><i>Heuristic Rule Base</i></b>	<i>F</i>	<i>R</i>	<i>F</i>	<i>R</i>	<i>F</i>	<i>R</i>	<i>F</i>	<i>R</i>	<i>F</i>	<i>R</i>	<i>F</i>	<i>R</i>	
Unsecured loans	<i>Application declined in early stage</i>		<i>Application declined in early stage</i>		0.79	0.21	0.64	0.36	0.27	0.73	0.64	0.36	
Secured loans					0.64	0.36	0.74	0.26	0.74	0.26	0.75	0.25	
CCJ, IVA, Bankruptcy & payday loans					0.76	0.24	0.66	0.34	0.76	0.24	0.66	0.34	
Searches					0.78	0.22	0.76	0.24	0.81	0.19	0.62	0.38	
Credit score					0.79	0.21	0.61	0.39	0.62	0.38	0.73	0.27	
Loan criteria, property valuation & property type					0.95	0.045	0.04	0.96	0.04	0.96	0.88	0.12	
<b>Predicted Output</b>					0.96	0.04	0.07	0.93	0.07	0.93	0.89	0.11	
<b>Decision: Fund or Reject</b>	REJECT		REJECT		FUND		REJECT		REJECT		FUND		
<b>Textual explanation for a rejected application</b>	Application has failed affordability test		Applicant have inadequate number and amount of bankruptcy and payday loans				-The property is poor and it has failed mortgage valuation. -The loan application do not fit our product-plan (loan criteria).		-The applicants have bad unsecured loan. -The property is poor and it has failed mortgage valuation. -The loan application do not fit our product-plan (loan criteria).				

Fig. 9. Example of loan applications in decision explanation illustrator.

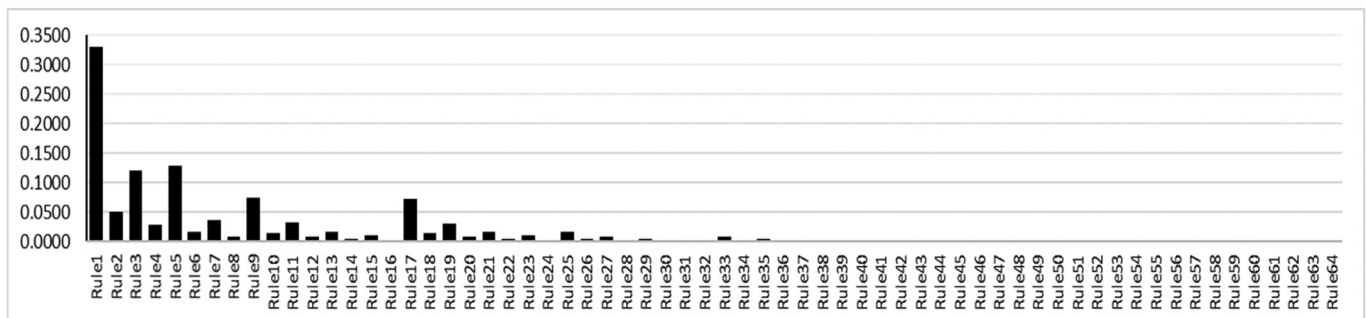


Fig. 10. Importance of rules activated by Application 3.



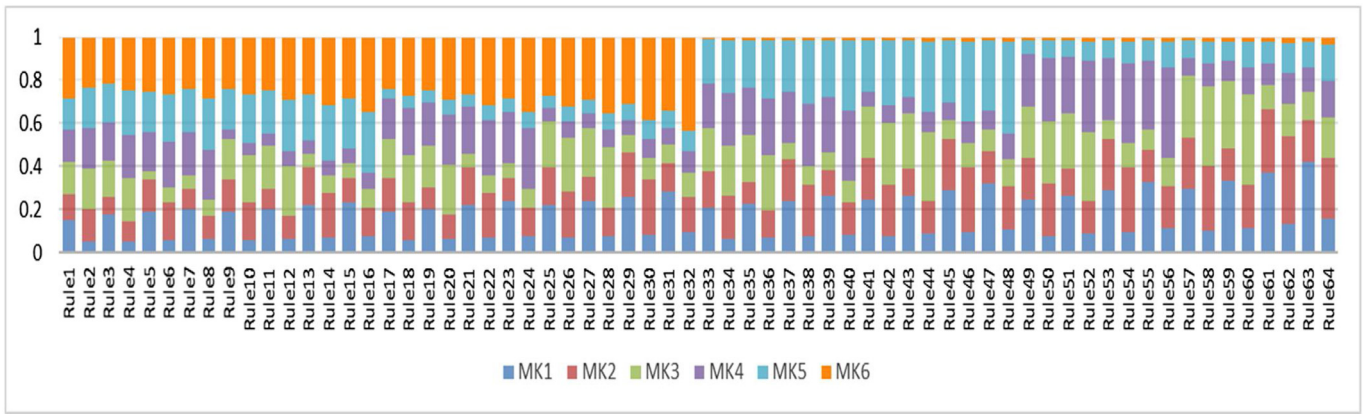


Fig. 11. Importance of attribute in rules activated by Application 3.

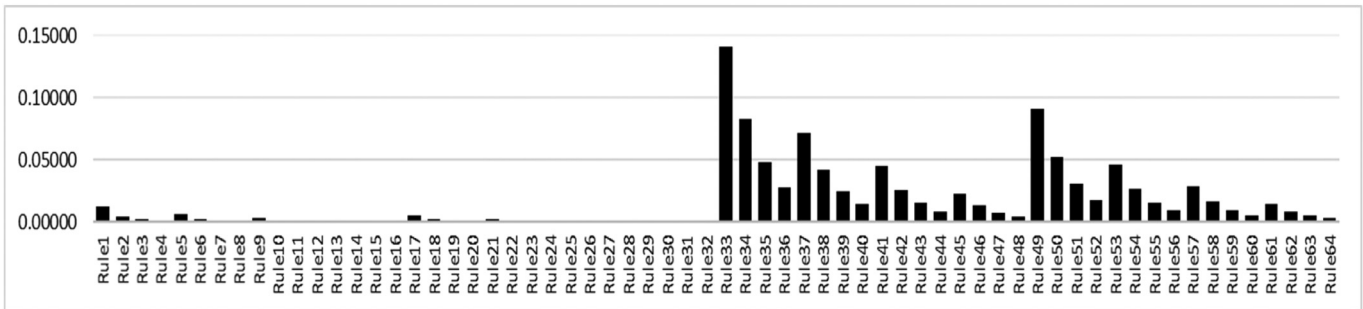


Fig. 12. Importance of rules activated by Application 4.

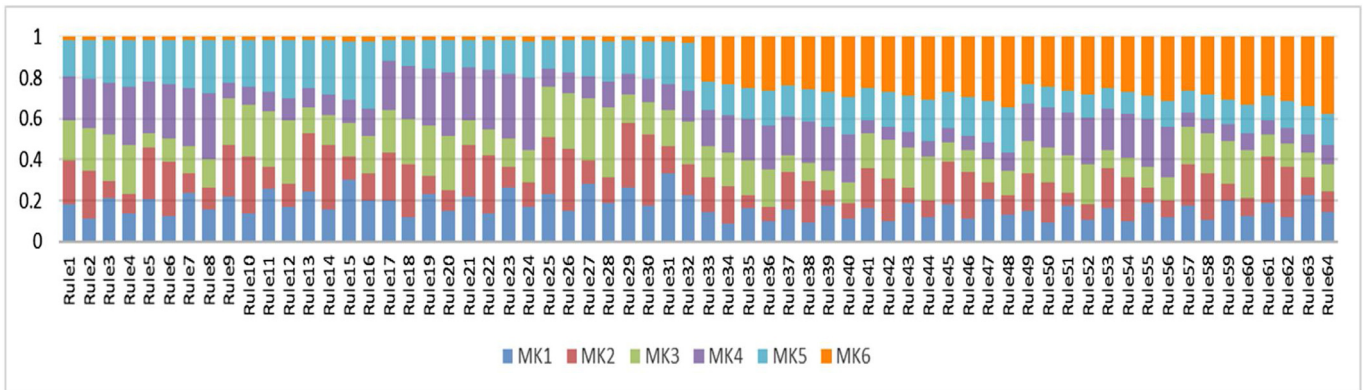


Fig. 13. Importance of attribute in rules activated by Application 4.

- Factual-rules-base: passed all rules
- Level 1 MAKER in heuristic-rule-base: In Fig. 9, two types of attributes unsecured loans and property & loan criteria have a high probability of rejection of loan application and all other attributes point towards the funding of this loan application.
- Level 2 BRB in heuristic-rule-base: For this applicant, rules 34 and 50 were most relevant. Both rules have R code (for reject) in both unsecured loans (MK1) and property & loan criteria (MK6). Additionally, rule 50 has R code in credit score; however, its contribution towards rejection is very small. The contribution of property & loan criteria in rules 34 and 50 towards rejection is 0.208 and 0.210, respectively. It is followed by contribution for unsecured loans 0.156 and 0.164, respectively. The contribution of attributes in rules for application 5 is shown in Fig. 15.

• Application 6:

- Factual-rules-base: passed all rules
- Level 1 MAKER in heuristic-rule-base: In Fig. 9, all attributes have a high probability for funding this loan application.
- Level 2 BRB in Heuristic-rule-base: Among all rules, rule 1 has the highest importance, shown in Fig. 16. It has F code in all the attributes. In this rule, loan criteria & property (MK6) has the highest contribution of 0.28 followed by unsecured loans (MK2), shown in Fig. 17.

This system can provide textual explanations for a rejected application. The examples of textual explanation are shown in the last row of Fig. 9. This type of textual explanation is crafted for different outcomes of MAKER framework and activated rules in a BRB system. Additional domain knowledge can be applied to build more coherent explanations, which could be sent to rejected applicants as a reason for denying their loan applications. The BRB system can explain a single decision and a chain of events

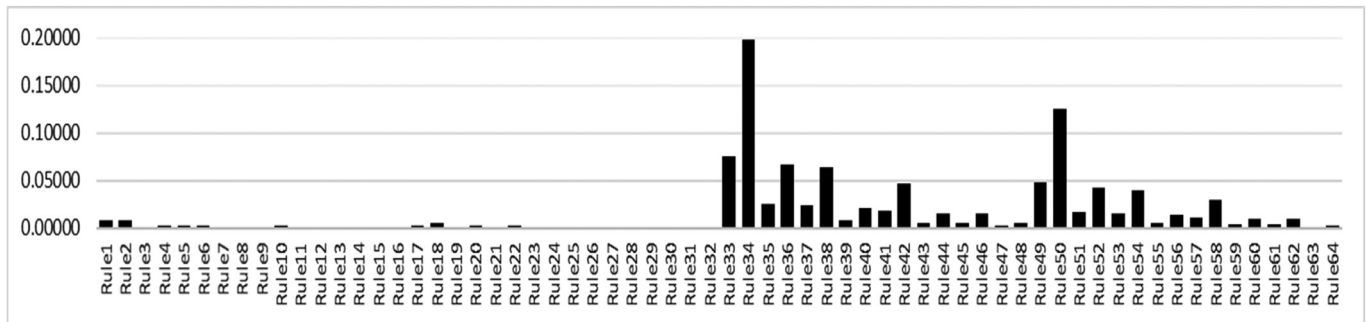


Fig. 14. Importance of rules activated by Application 5.

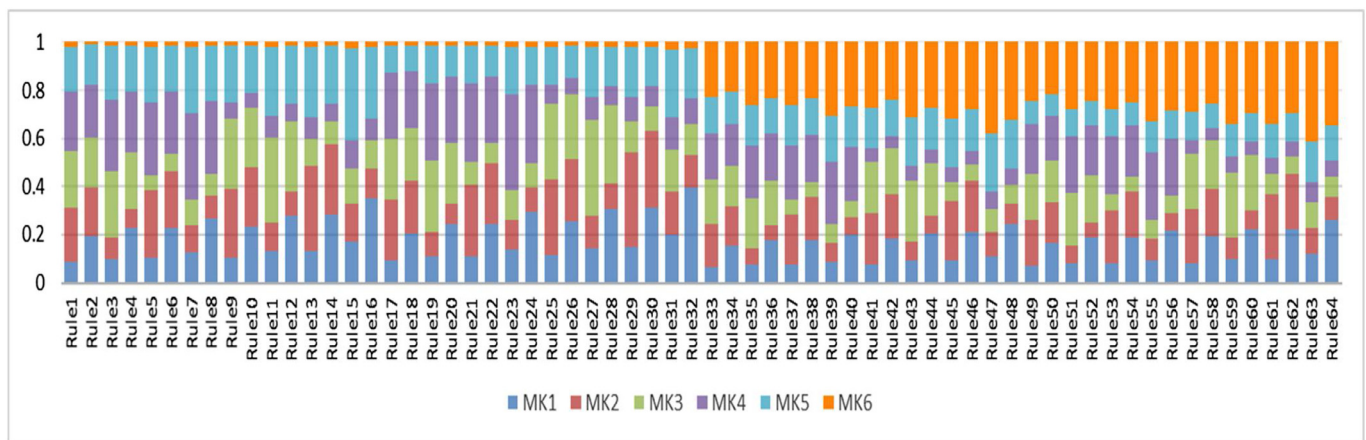


Fig. 15. Importance of attributes in rules activated by Application 5.

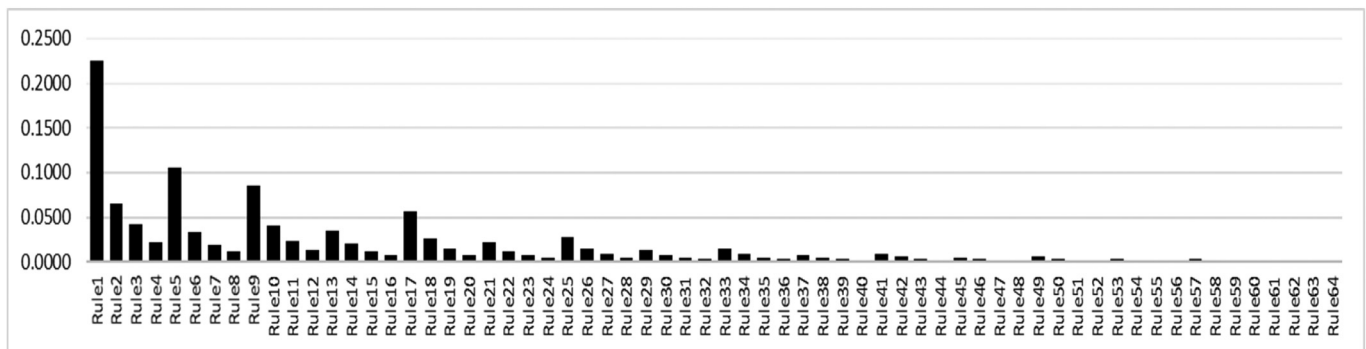


Fig. 16. Importance of rule activated by Application 6.

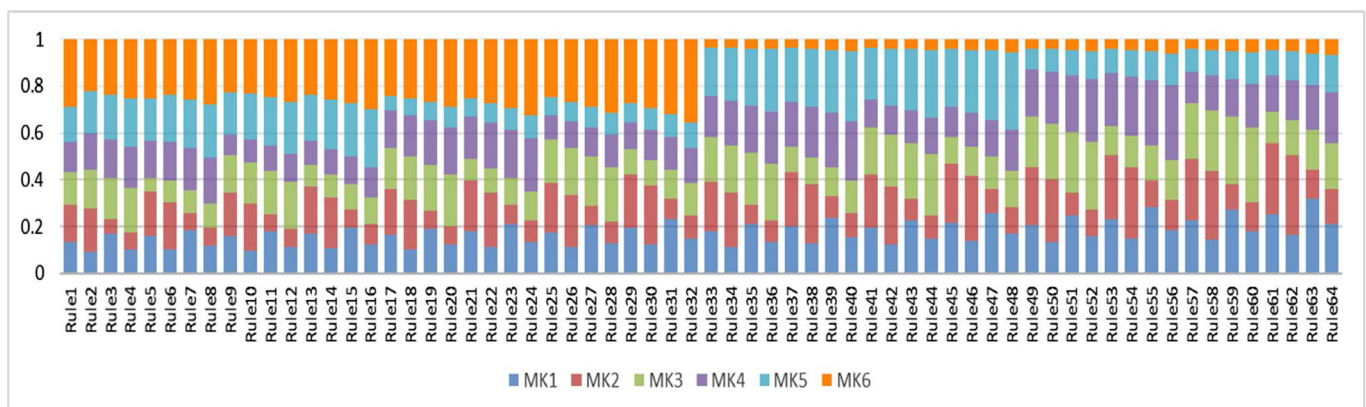


Fig. 17. Importance of attributes in rules activated by Application 6.

leading to the decision. This is enabled by inference for each category of creditworthiness factors (unsecured loans, secured loans, searches, credit score and CCJ, IVA, bankruptcy and payday loans) by the MAKER framework and contribution of attributes in activated rules in BRB. This type of explainability helps underwriters to understand the decision-making process at each level of the system, which enables them to make fast and accurate choices for robust lending decision-making.

The AI-based systems can reap benefits to businesses only when their executives can trust these systems. The transparent decision-making process of BRB system could provide sufficient assurance to executives in lending companies to sanction the deployment of BRB system. This enables lending companies to be in regulatory compliance such as the General Data Protection Regulation (GDPR) in Europe which mandates explanation of outcomes from AI/ML systems.

All rules are activated by the transformed data of an application. The activation weights of many rules are very small and could be close to zero; therefore visually the importance of some least important rules in Figs. 10, 12, 14 and 16 is zero.

### 3.5. Limitation

The proposed system can explain each step of the decision-making process. It combines expert knowledge and fine-tunes the system by training it with data. However, the development of this system is a time-consuming process as it demands deep domain knowledge. It requires expert inputs in finalizing the structure of a belief-rule-base system and appending their certain and uncertain judgment in IF-THEN rules in the form of degree of belief. Further, it requires cleaning and aggregation of large credit data and information provided by applicants in loan applications to map and train lending organization baseline rules and heuristic knowledge of experts. This system was adopted to automate the underwriting process at Together Financial Services; however, it cannot reject loan applications based on inconsistency in loan applications, such as missing or undisclosed information and customer integrity concerns. A loan application must be screened by another system or manually by humans before providing decision by this system.

## 4. Conclusion

In this paper, we presented the methodology to develop the belief-rule-based (BRB) system as an explainable AI decision-support-system to automate the underwriting process of lend

loans. Unlike black-box models, the BRB system can explicitly accommodate expert knowledge and can also learn from data by supervised learning, though the acquisition of expert knowledge can be a time-consuming and labor-intensive task. The decision-making process in this system can be explained by the importance of rules activated by a data point representing a loan application and by the contribution of attributes in activated rules. Through a business case study, we have demonstrated that the proposed AI decision-support-system provides a good trade-off between prediction accuracy and explainability. The importance of activated rules and their attributes in the rules help to understand the reasoning behind the decisions. The textual explanations initiated by the chain of events in the factual-rule-base to the heuristic-rule-base could be sent to rejected applicants as reasons for denying their loan applications. The BRB structure and its decision-making process can easily be narrated and understood by non-technical people. A transparent decision-making process provides sufficient assurance to executives to allow the deployment of this system in their companies.

## Declaration of Competing Interest

None.

## Acknowledgements

We are thankful to Together Financial Services for providing financial data. We are especially indebted to dr. Max Griffiths, Sonia Awan and all the underwriters at Together Financial services for their insight regarding underwriting process. We also wish to express our appreciation to the reviewers for their insightful comments, which have helped us significantly improve this manuscript. This work is funded by the Alliance Strategic Grant of The University of Manchester and by the FinTech innovation lab at Alliance Manchester Business School, UK. The second author is grateful for the support to this and related research by NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization (U1709215). The second and third authors would like to thank European Commission for its support under the grant number 823759.

## Appendix A

Tables 17–19.

**Table 17**  
Rules in Factual-Rule-Base (do not require training contain facts).

Rule Number	Rule1: Affordability	Rule2: # of bankruptcy > X and amount of bankruptcy > X	Rule3: # IVA and CCJ > X	Rule4: #payday loans in past Y years > X	Belief Degree (accept, decline)
1	PASS	NONE	NONE	NONE	(1,0)
2	FAIL	NONE	NONE	NONE	(0,1)
3	PASS	=>1	NONE	NONE	(0,1)
4	FAIL	=>1	NONE	NONE	(0,1)
5	PASS	NONE	=>1	NONE	(0,1)
6	FAIL	NONE	=>1	NONE	(0,1)
7	PASS	=>1	=>1	NONE	(0,1)
8	FAIL	=>1	=>1	NONE	(0,1)
9	PASS	NONE	NONE	=>1	(0,1)
10	FAIL	NONE	NONE	=>1	(0,1)
11	PASS	=>1	NONE	=>1	(0,1)
12	FAIL	=>1	NONE	=>1	(0,1)
13	PASS	NONE	=>1	=>1	(0,1)
14	FAIL	NONE	=>1	=>1	(0,1)
15	PASS	=>1	=>1	=>1	(0,1)
16	FAIL	=>1	=>1	=>1	(0,1)

**Table 18**  
Rules in Heuristic-rule-base  
Before training.

Rule number	Rule weight	MK1: Unsecured loans	MK2: Secured loans	MK3: CCJ, IVA, Bankruptcy and payday	MK4: Debt and credit searches	MK5: Credit score	MK6: Loan criteria, property valuation and property type	Attribute weight	Belief degree
1	1	F1	F2	F3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(1.0,0.0)
2	1	R1	F2	F3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.4,0.6)
3	1	F1	R2	F3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
4	1	R1	R2	F3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.6,0.4)
5	1	F1	F2	R3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(1.0,0.0)
6	1	R1	F2	R3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
7	1	F1	R2	R3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
8	1	R1	R2	R3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.5,0.5)
9	1	F1	F2	F3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.8,0.2)
10	1	R1	F2	F3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
11	1	F1	R2	F3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65,0.35)
12	1	R1	R2	F3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65,0.35)
13	1	F1	F2	R3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
14	1	R1	F2	R3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.3,0.7)
15	1	F1	R2	R3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65,0.35)
16	1	R1	R2	R3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.3,0.7)
17	1	F1	F2	F3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.8,0.2)
18	1	R1	F2	F3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.8,0.2)
19	1	F1	R2	F3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
20	1	R1	R2	F3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65,0.35)
21	1	F1	F2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.6,0.4)
22	1	R1	F2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
23	1	F1	R2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65, 0.35)
24	1	R1	R2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7, 0.3)
25	1	F1	F2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.8, 0.2)
26	1	R1	F2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65,0.35)
27	1	F1	R2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.7,0.3)
28	1	R1	R2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.65,0.35)
29	1	F1	F2	R3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.6,0.4)
30	1	R1	F2	R3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.3,0.7)
31	1	F1	R2	R3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.5,0.5)
32	1	R1	R2	R3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.3, 0.7)
33	1	F1	F2	F3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.5,0.5)
34	1	R1	F2	F3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
35	1	F1	R2	F3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.5,0.5)
36	1	R1	R2	F3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
37	1	F1	F2	R3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
38	1	R1	F2	R3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
39	1	F1	R2	R3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
40	1	R1	R2	R3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
41	1	F1	F2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
42	1	R1	F2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
43	1	F1	R2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
44	1	R1	R2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
45	1	F1	F2	R3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
46	1	R1	F2	R3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
47	1	F1	R2	R3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
48	1	R1	R2	R3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
49	1	F1	F2	F3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
50	1	R1	F2	F3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
51	1	F1	R2	F3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
52	1	R1	R2	F3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
53	1	F1	F2	R3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
54	1	R1	F2	R3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
55	1	F1	R2	R3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
56	1	R1	R2	R3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
57	1	F1	F2	F3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
58	1	R1	F2	F3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
59	1	F1	R2	F3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
60	1	R1	R2	F3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
61	1	F1	F2	R3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
62	1	R1	F2	R3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)
63	1	F1	R2	R3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.2,0.8)
64	1	R1	R2	R3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.1,0.9)



**Table 19**  
Rules in heuristic-rule-base  
After training.

Rule number	Rule weight	MK1: Un-secured loans	MK2: Secured loans	MK3: CCJ, IVA, Bankruptcy and payday	MK4: Debt and credit searches	MK5: Credit score	MK6: Loan criteria, property valuation and property type	Attribute weight	Belief degree
1	0.75	F1	F2	F3	F4	F5	F6	0.64, 0.64, 0.64, 0.63, 0.62, 1.0	(1.0,0.0)
2	0.92	R1	F2	F3	F4	F5	F6	0.99, 1.0, 1.0, 0.99, 1.0, 1.0	(1.0,0.0)
3	0.92	F1	R2	F3	F4	F5	F6	0.98, 0.94, 0.98, 0.98, 0.98, 1.0	(1.0,0.0)
4	0.97	R1	R2	F3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.85,0.15)
5	0.91	F1	F2	R3	F4	F5	F6	0.89, 0.89, 0.58, 0.89, 0.88, 1.0	(1.0,0.0)
6	0.97	R1	F2	R3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.95,0.05)
7	0.97	F1	R2	R3	F4	F5	F6	1.0, 0.99, 0.99, 0.99, 1.0, 1.0	(0.96,0.04)
8	0.99	R1	R2	R3	F4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.59,0.41)
9	0.92	F1	F2	F3	R4	F5	F6	0.95, 0.95, 0.95, 0.83, 0.95, 1.0	(1.0,0.0)
10	0.97	R1	F2	F3	R4	F5	F6	0.99, 1.0, 1.0, 1.0, 1.0, 1.0	(0.95,0.05)
11	0.97	F1	R2	F3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.91,0.09)
12	0.99	R1	R2	F3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.74,0.26)
13	0.97	F1	F2	R3	R4	F5	F6	1.0, 0.99, 0.99, 0.99, 0.99, 1.0	(0.96,0.04)
14	0.99	R1	F2	R3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.39,0.61)
15	0.99	F1	R2	R3	R4	F5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.75,0.25)
16	1.00	R1	R2	R3	R4	F5	F6	1.0, 1.0, 1.0, 1.0, 0.99, 1.0	(0.33,0.67)
17	0.91	F1	F2	F3	F4	R5	F6	0.95, 0.95, 0.95, 0.95, 0.83, 1.0	(1.0,0.0)
18	0.97	R1	F2	F3	F4	R5	F6	0.96, 0.99, 0.99, 0.99, 0.96, 1.0	(0.96,0.04)
19	0.97	F1	R2	F3	F4	R5	F6	1.0, 1.0, 1.0, 1.0, 0.99, 1.0	(0.98,0.02)
20	0.99	R1	R2	F3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.74,0.26)
21	0.97	F1	F2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.87,0.13)
22	0.99	R1	F2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.79,0.21)
23	0.99	F1	R2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.75,0.25)
24	1.00	R1	R2	R3	F4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	0.73,0.22
25	0.97	F1	F2	F3	R4	R5	F6	0.99, 0.98, 0.99, 0.96, 0.96, 1.0	(0.97,0.03)
26	0.99	R1	F2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.74,0.26)
27	0.99	F1	R2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.81,0.19)
28	1.00	R1	R2	F3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.68,0.42)
29	0.99	F1	F2	R3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.70,0.30)
30	1.00	R1	F2	R3	R4	R5	F6	1.0, 0.99, 0.99, 0.99, 1.0, 1.0	(0.33,0.67)
31	1.00	F1	R2	R3	R4	R5	F6	1.0,1.0,1.0,1.0,1.0,1.0	(0.54,0.46)
32	1.00	R1	R2	R3	R4	R5	F6	1.0, 0.99, 0.99, 0.99, 1.0, 0.99	(0.31,0.69)
33	0.93	F1	F2	F3	F4	F5	R6	0.99, 0.99, 1.0, 1.0, 0.99, 1.0	(0.03,0.97)
34	0.97	R1	F2	F3	F4	F5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 1.0	(0.00,1.0)
35	0.97	F1	R2	F3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.31,0.69)
36	0.99	R1	R2	F3	F4	F5	R6	1.0, 0.99, 0.99, 0.99, 0.99, 1.0	(0.10,0.90)
37	0.97	F1	F2	R3	F4	F5	R6	0.99, 0.99, 1.0, 0.99, 0.99, 1.0	(0.02,0.98)
38	0.99	R1	F2	R3	F4	F5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 1.0	(0.02,0.98)
39	0.99	F1	R2	R3	F4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.12,0.88)
40	0.99	R1	R2	R3	F4	F5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 1.0	(0.06,0.94)
41	0.97	F1	F2	F3	R4	F5	R6	0.99, 0.99, 1.0, 1.0, 0.99, 1.0	(0.04,0.96)
42	0.99	R1	F2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.12,0.88)
43	0.99	F1	R2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.13,0.87)
44	0.99	R1	R2	F3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.16,0.84)
45	0.99	F1	F2	R3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.13,0.87)
46	1.00	R1	F2	R3	R4	F5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 1.0	(0.07,0.93)
47	1.00	F1	R2	R3	R4	F5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.17,0.83)
48	1.00	R1	R2	R3	R4	F5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 1.0	(0.08,0.92)
49	0.97	F1	F2	F3	F4	R5	R6	0.99, 0.99, 0.99, 0.97, 0.98, 1.0	(0.00,1.0)
50	0.99	R1	F2	F3	F4	R5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 0.99	(0.08,0.92)
51	0.99	F1	R2	F3	F4	R5	R6	0.99, 1.0, 0.99, 0.99, 1.0, 1.0	(0.09,0.91)
52	0.99	R1	R2	F3	F4	R5	R6	1.0, 0.99, 0.99, 0.99, 1.0, 0.99	(0.14,0.86)
53	0.99	F1	F2	R3	F4	R5	R6	0.99, 0.99, 1.0, 1.0, 1.0, 1.0	(0.10,0.90)
54	0.99	R1	F2	R3	F4	R5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 0.99	(0.05,0.95)
55	0.99	F1	R2	R3	F4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.15,0.85)
56	1.00	R1	R2	R3	F4	R5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 0.93	(0.07,0.93)
57	0.99	F1	F2	F3	R4	R5	R6	0.98, 1.0, 1.0, 1.0, 1.0, 1.0	(0.10,0.90)
58	0.99	R1	F2	F3	R4	R5	R6	1.0, 0.99, 0.99, 0.99, 1.0, 0.99	(0.15,0.85)
59	0.99	F1	R2	F3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.15,0.85)
60	1.00	R1	R2	F3	R4	R5	R6	1.0, 1.0, 0.99, 0.99, 1.0, 0.99	(0.17,0.83)
61	0.99	F1	F2	R3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.16,0.84)
62	1.00	R1	F2	R3	R4	R5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 0.99	(0.08,0.92)
63	1.00	F1	R2	R3	R4	R5	R6	1.0,1.0,1.0,1.0,1.0,1.0	(0.18,0.92)
64	1.00	R1	R2	R3	R4	R5	R6	0.99, 0.99, 0.99, 0.99, 0.99, 0.99	(0.09,0.91)

## Appendix B. Parameter value of MAKER example: Before Training

### Tables 20–29

**Table 20**  
Initial weight of single evidence ( $w_{h,v,1}$  and  $w_{h,v,2}$ ).

Debt searches	A1	B1	C1	D1	Credit searches	A2	B2	C2	D2
F	1	1	1	1	F	1	1	1	1
R	1	1	1	1	R	1	1	1	1

**Table 21**  
Initial weight of joint evidence ( $w_{h,v1,v2}$ ).

Debt searches	A1				B1				C1				D1			
	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

**Table 22**  
Initial reliability of single evidence ( $r_{h,v,1}$  and  $r_{h,v,2}$ ).

Debt searches	A1	B1	C1	D1	Credit searches	A2	B2	C2	D2
F	1	1	1	1	F	1	1	1	1
R	1	1	1	1	R	1	1	1	1

**Table 23**  
Initial reliability of joint evidence ( $r_{h,v1,v2}$ ).

Debt Searches	A1				B1				C1				D1			
	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

**Table 24**  
Initial reliability ratio ( $\gamma_{h,v1,v2}$ ).

Debt searches	A1				B1				C1				D1			
	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

**Table 25**  
Weight of single evidence ( $w_{h,v,1}$  and  $w_{h,v,2}$ )  
Parameter value of MAKER example: After Training.

Debt searches	A1	B1	C1	D1	Credit searches	A2	B2	C2	D2
F	1.00	1.00	1.00	0.94	F	1.00	0.98	1.00	1.00
R	0.99	0.97	0.73	0.46	R	0.99	0.62	0.76	0.75

**Table 26**  
Weight of joint evidence ( $w_{h,v1,v2}$ ).

Debt searches	A1				B1				C1				D1			
	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	1	1	1	1	1	1	1	1	1	1	1	1	1	0.98	1	0.99
R	1	0.98	0.98	1	1	0.97	0.98	0.98	0.99	0.89	1	0.98	0.98	0.99	0.98	0.99

**Table 27**Reliability of single evidence ( $r_{h,v,1}$  and  $r_{h,v,2}$ ).

Debt searches	A1	B1	C1	D1	Credit searches	A2	B2	C2	D2
F	1.00	1.00	1.00	0.91	F	1.00	0.98	1.00	1.00
R	0.99	0.97	0.99	0.93	R	0.99	0.89	0.99	0.86

**Table 28**Reliability of joint evidence ( $r_{h,v1,v2}$ ).

Debt searches	A1				B1				C1				D1			
Credit searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	1	1	0.99	1	1	1	1	1	1	1	1	1	1	1	1	1
R	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.99

**Table 29**Reliability ratio ( $\gamma_{h,v1,v2}$ ).

Debt searches	A1				B1				C1				D1			
Credit searches	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2	A2	B2	C2	D2
F	1	1	1	1	1	1.01	1.05	1.09	1.00	1.11	1.07	1.06	1.02	1.21	1.14	1.15
R	1	1.0	1	1	1.00	0.98	0.99	0.98	0.99	0.88	0.92	0.93	0.99	0.78	0.85	0.84

## References

- Abdou, H. A. (2009). Genetic programming for credit scoring: The case of Egyptian public sector banks. *Expert Systems with Applications*, 36(9), 11402–11417.
- Abellán, J., & Castellano, J. G. (2017). A comparative study on base classifiers in ensemble methods for credit scoring. *Expert Systems with Applications*, 73, 1–10.
- Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access: Practical Innovations, Open Solutions*, 6, 52138–52160.
- Aggour, K. S., Bonissone, P. P., Cheetham, W. E., & Messmer, R. P. (2006). Automating the underwriting of insurance applications. *AI magazine*, 27(3), 36–36.
- Aitken, R. (2017). "All data is credit data": Constituting the unbanked. *Competition & Change*, 21(4), 274–300.
- Akinyokun, O. C. (2015). Fuzzy logic-driven expert system for the diagnosis of heart failure disease. *Artificial Intelligence Research*, 4, 12–21.
- Bensic, M., Sarlija, N., & Zekic-Susac, M. (2005). Modelling small-business credit scoring by using logistic regression, neural networks and decision trees. *Intelligent Systems in Accounting, Finance & Management: International Journal*, 13(3), 133–150.
- Bijak, K., & Thomas, L. C. (2012). Does segmentation always improve model performance in credit scoring. *Expert Systems with Applications*, 39(3), 2433–2442.
- Casalicchio, G., Molnar, C., & Bischl, B. (2018). Visualizing the feature importance for black box models. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 655–670.
- Cawley, G. C., & Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(Jul), 2079–2107.
- Chen, W., Ma, C., & Ma, L. (2009). Mining the customer credit using hybrid support vector machine technique. *Expert Systems with Applications*, 36(4), 7611–7616.
- Chen, Y. W., Yang, J. B., Xu, D. L., Zhou, Z. J., & Tang, D. W. (2011). Inference analysis and adaptive training for belief rule based systems. *Expert Systems with Applications*, 38(10), 12845–12860.
- Cocheo, S. (1995). Automated underwriting: Tower of techno-Babel. *American Bankers Association.ABA Banking Journal*, 87(9), 68.
- Feigenbaum, E. A. (1980). Knowledge engineering: The applied side of artificial intelligence. Stanford University CA Department of Computer Science No. STAN-CS-80-812.
- Gates, S. W., Perry, V. G., & Zorn, P. M. (2002). Automated underwriting in mortgage lending: Good news for the underserved. *Housing Policy Debate*, 13(2), 369–391.
- Glassman, C. A., & Wilkins, H. M. (1997). Credit scoring: Probabilities and pitfalls. *Journal of Retail Banking Services*, 19(2), 53–57.
- Gomez, F., Hull, R., Karr, C., Hosken, B., & Verhagen, W. (1992). Combining factual and heuristic knowledge in knowledge acquisition. *Telematics and Informatics*, 9(3–4), 297–311.
- Harris, T. (2015). Credit scoring using the clustered support vector machine. *Expert Systems with Applications*, 42(2), 741–750.
- Hens, A. B., & Tiwari, M. K. (2012). Computational time reduction for credit scoring: An integrated approach based on support vector machine and stratified sampling method. *Expert Systems with Applications*, 39(8), 6774–6781.
- Holsapple, C. W., Raj, V., & Wagner, W. P. (2008). An experimental investigation of the impact of domain complexity on knowledge acquisition (KA) methods. *Expert Systems with Applications*, 35(3), 1084–1094.
- Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert systems with applications*, 33(4), 847–856.
- Hung, C., & Chen, J. H. (2009). A selective ensemble based on expected probabilities for bankruptcy prediction. *Expert systems with applications*, 36(3), 5297–5303.
- Khashman, A. (2010). Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes. *Expert Systems with Applications*, 37(9), 6233–6239.
- Killin, J. (1992). The management and maintenance of an operational kads system development. In *International conference on knowledge engineering and knowledge management* (p. 6). Springer. -6.
- Kim, Y. S., & Sohn, S. Y. (2004). Managing loan customers using misclassification patterns of credit scoring model. *Expert Systems with Applications*, 26(4), 567–573.
- Kingston, J. (1991). X-MATE: Creating an interpretation model for credit risk assessment. In *In research and development in expert systems VIII: Proceedings of 11th annual technical conference of the BCS specialist group* (pp. 165–174). Cambridge University Press.
- Krovvidy, S. (2008). Custom DU: A web-based business user-driven automated underwriting system. *AI Magazine*, 29(1), 41–41.
- Leong, C. K. (2016). Credit risk scoring with bayesian network models. *Computational Economics*, 47(3), 423–446.
- Li, S. T., Shiue, W., & Huang, M. H. (2006). The evaluation of consumer loans using support vector machines. *Expert Systems with Applications*, 30(4), 772–782.
- Lipton, Z. C. (2016). The mythos of model interpretability. In *Proceedings of the ICML workshop on human interpretability in machine learning*.
- Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In S. M. Lundberg, & S. I. Lee (Eds.), *Advances in neural information processing systems* (pp. 4765–4774). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*.
- Luo, C., Wu, D., & Wu, D. (2017). A deep learning approach for credit scoring using credit default swaps. *Engineering Applications of Artificial Intelligence*, 65, 465–470.
- Marqués, A. I., García, V., & Sánchez, J. S. (2012). Exploring the behaviour of base classifiers in credit scoring ensembles. *Expert Systems with Applications*, 39(11), 10244–10250.
- Metawa, N., Hassan, M. K., & Elhoseny, M. (2017). Genetic algorithm-based model for optimizing bank lending decisions. *Expert Systems with Applications*, 80, 75–82.
- Montavon, G., Lapuschkin, S., Binder, A., Samek, W., & Müller, K. R. (2017). Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65, 211–222.
- Montavon, G., Samek, W., & Müller, K. R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73, 1–15.
- Nanni, L., & Lumini, A. (2009). An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring. *Expert systems with applications*, 36(2), 3028–3033.
- Ong, C. S., Huang, J. J., & Tzeng, G. H. (2005). Building credit scoring models using genetic programming. *Expert Systems with Applications*, 29(1), 41–47.
- Peterson, D. (2017). *Maximize efficiency: How automation can improve your loan origination process*. Moody's Analytics.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Model-agnostic interpretability of machine learning. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*: 1606 (p. 05386). New-York.
- Schebesch, K. B., & Stecking, R. (2005). Support vector machines for classifying and describing credit applicants: Detecting typical and critical regions. *Journal of the Operational Research Society*, 56(9), 1082–1088.
- Shafer, G. (1976). *A mathematical theory of evidence*. Princeton university press.
- Smarandache, F., Dezert, J., & Tacnet, J. M. (2010). Fusion of sources of evidence with different importances and reliabilities. In *Proceedings of the 13th international conference on information fusion* (pp. 1–8). IEEE.

- Stein, F., Preece, A., & Boicu, M. (2018). Artificial intelligence in government and public sector proceedings. In *Proceedings of the AAAI FSS-18* Arlington, Virginia, USA arXiv:1810.06018.
- Sun, R. (1995). Robust reasoning: Integrating rule-based and similarity-based reasoning. *Artificial Intelligence*, 75(2), 241–295.
- Tomczak, J. M., & Zięba, M. (2015). Classification restricted boltzmann machine for comprehensible credit scoring model. *Expert Systems with Applications*, 42(4), 1789–1796.
- Wagner, W. P., Otto, J., & Chung, Q. B. (2002). Knowledge acquisition for expert systems in accounting and financial problem domains. *Knowledge-Based Systems*, 15(8), 439–447.
- Wang, G., Ma, J., Huang, L., & Xu, K. (2012). Two credit scoring models based on dual strategy ensemble trees. *Knowledge-Based Systems*, 26, 61–68.
- Wei, Y., Yildirim, P., Van den Bulte, C., & Dellarocas, C. (2015). Credit scoring with social network data. *Marketing Science*, 35(2), 234–258.
- West, D. (2000). Neural network credit scoring models. *Computers & Operations Research*, 27(11–12), 1131–1152.
- Wu, W. W. (2011). Improving classification accuracy and causal knowledge for better credit decisions. *International Journal of Neural Systems*, 21(04), 297–309.
- Xiao, H., Xiao, Z., & Wang, Y. (2016). Ensemble classification based on supervised clustering for credit scoring. *Applied Soft Computing*, 43, 73–86.
- Xu, D., Zhang, X., & Feng, H. (2018). Generalized fuzzy soft sets theory-based novel hybrid ensemble credit scoring model. *International Journal of Finance & Economics*, 24(2), 903–921.
- Yang, J. B., Liu, J., Wang, J., Sii, H. S., & Wang, H. W. (2006). Belief rule-base inference methodology using the evidential reasoning approach-RIMER. *IEEE Transactions on Systems, Man, and Cybernetics-part A: Systems and Humans*, 36(2), 266–285.
- Yang, J. B., & Xu, D. L. (2013). Evidential reasoning rule for evidence combination. *Artificial Intelligence*, 205, 1–29.
- Yang, J. B., & Xu, D. L. (2017). Inferential modelling and decision making with data. In *Proceedings of the 23rd International conference on automation and computing (ICAC)* (pp. 1–6). IEEE.
- Yap, B. W., Ong, S. H., & Husain, N. H. (2011). Using data mining to improve assessment of credit worthiness via credit scoring models. *Expert Systems with Applications*, 38(10), 13274–13283.
- Zhang, D., Zhou, X., Leung, S. C., & Zheng, J. (2010). Vertical bagging decision trees model for credit scoring. *Expert Systems with Applications*, 37(12), 7838–7843.
- Zhao, Z., Xu, S., Kang, B. H., Kabir, M. M., Liu, Y., & Wasinger, R. (2015). Investigation and improvement of multi-layer perceptron neural networks for credit scoring. *Expert Systems with Applications*, 42(7), 3508–3516.
- Zhu, H., Beling, P. A., & Overstreet, G. A. (2002). A bayesian framework for the combination of classifier outputs. *Journal of the Operational Research Society*, 53(7), 719–727.