Proceedings of the 55th Hawaii International Conference on System Sciences | 2022

Predicting the Outcome of a Debt Collection Process Using Bayesian Networks

Benedikt Köhler
Karlsruhe Institute of Technology
benedikt.koehler@outlook.com

Hansjörg Fromm
Karlsruhe Institute of Technology
hansjoerg.fromm@kit.edu

Abstract

Many companies rely on professional debt-collection agencies to handle their outstanding debts. These agencies conduct a debt collection process consisting of successive, escalating actions with the aim of getting a debtor to settle an overdue claim. The sequence of actions is administered by agents who often have to make decisions on a case-by-case basis. This requires understanding of complex data and making decisions under uncertainty. This decision-making process has hardly been investigated so far. We are proposing Bayesian networks as the analytical basis for a decision support system. Bayesian networks are strong in dealing with uncertainties. They can be used for both predicting the success of a case and making recommendations on actions. The evaluation shows that Bayesian networks have a very good predictive performance which gets even better as the process evolves. With this instrument, the agents can make better-informed decisions in the debt collection process.

1. Introduction

Late payments and nonrecoverable debts are increasingly common phenomena in today's economy. One in five of Europe's SMEs (companies with less than 250 employees) say that late payments are a threat to their business [1]. Collecting outstanding payments is a lengthy and time-consuming process and many companies are overwhelmed with their administration.

Therefore, companies rely on professional debt-collection agencies. These agencies are, in contrast to the creditor, specialized in the debt collection process. They work more efficiently and can also act more effectively towards the debtor.

The core business of the debt collection service is the processing of collection files, a process that has been supported by IT systems for many years. However, decisions in this process are largely made manually by agents. The use of intelligent data analysis methods to increase efficiency and subsequently automate the process is still in an early stage. So et al. [2] call the debt collection process "little researched compared with other operations management activities".

This study contributes to addressing this gap by applying data analytics methods to the collection process in order to support processing with intelligent algorithms. Specifically, we investigate whether Bayesian networks can be used as a decision support system for the agents to deal with the many uncertainties in debt collection.

1.1. The Debt Collection Process

The debt collection process consists of a sequence of successive, escalating actions with the aim of getting the debtor to settle the overdue claim. Among these actions are formal reminders, telephone calls, the involvement of lawyers, the appeal to a court, foreclosure auctions, etc.

The collection process is controlled by agents who often have to make decisions on a case-by-case basis. The agent must assess the future behavior of the debtor and the likelihood of payment in order to be able to make decisions about how to proceed. At any point in time, further investments in time and money must be weighed against the prospects for success.

The wide range of actions that can be carried out by the agent in connection with the different possible (re)actions by the debtor results in a multitude of potential execution paths for a case. Certain process steps are required by the legal dunning procedure. But around this "core process", many "ad-hoc" actions and decisions are made. Figure 1 gives an impression of this. Three cases are shown: (a) a case in which the debtor has fully paid his debt after a few actions triggered by the agent, (b) a case which included an installment plan and which was closed after the remaining debt was low, and (c) a case which is still open and whose progress is uncertain after a number of actions.

Due to the high complexity of the collection process, it is common practice that the agent follows simple and pragmatic rules based on limited information and intuition in order to decide how to proceed. Thus, information may remain unused, and this can lead to suboptimal decisions.

URI: https://hdl.handle.net/10125/79548
978-0-9981331-5-7

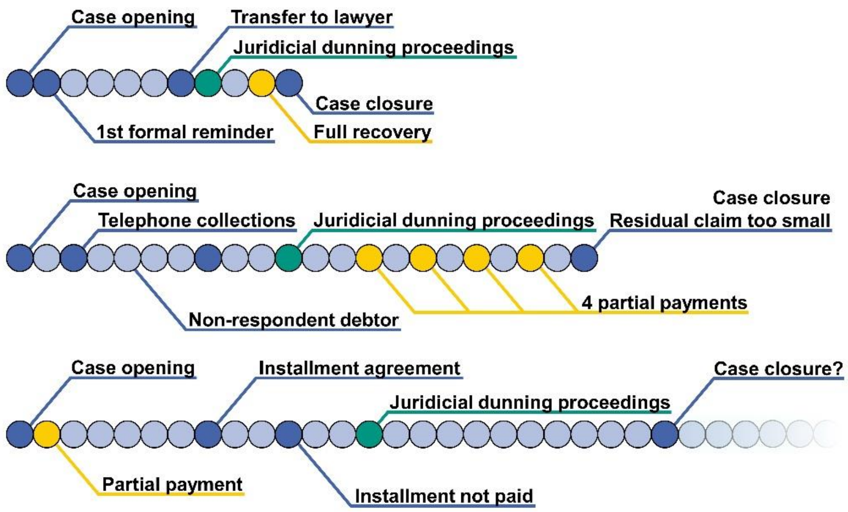

Figure 1. The individual course of a case can vary greatly.

A decision support system (DSS) is desirable that takes all information into account and assists the agent to make rational and objective decisions considering the likelihood of success.

The objective of such a DSS is (a) to make a prediction at any time - from the start of the process - as to whether the case will ultimately be positive, i.e. whether the debtor will pay his outstanding debts in full, and (b) to show which effect follow-up actions and debtor responses have on the likelihood of the case being positive.

2. Related Work

Over the last years, a number of studies appeared that apply analytical methods to the debt collection process. Although the title of the publications may indicate a similarity with our problem - the underlying business situations can be very different. A distinction must be made according to (a) the relationship between the debtor and the creditor, (b) the question of whether a decision is to be made before or during the collection process, (c) the extent of data that is available for the decision-making process. The problem presented here addresses collection agencies that are not in a customer-provider relationship with the debtor (a), supports decisions in the course of the collection process (b) and, due to the lack of a continuous, direct relationship, has little data on the debtor and relies mainly on data that is collected during the course of the process (c).

Related, but also quite different from debt collection is *credit scoring*. Credit scoring is used by companies to decide whether or not a credit should be granted to a customer. It is a one-time decision, and typically, the creditor has plenty of data available on the credit applicant. Credit scoring is a well-researched area and widely used in practice. All known classification methods have been applied to address this problem, e.g. linear regression, decision trees, support vector machines, neural networks, etc. (for an overview, see Louzada et al. [3]). In all characteristics (a), (b), and (c), credit scoring is different from debt collection.

A group of publications addresses a situation in which an unsecured credit has been granted, e.g. in form of a service contract, and the customer is late with a payment. The creditor must decide what actions to take, typically based on a prediction of the likelihood of repayment. In contrast to credit scoring, where only ex-ante information on the debtor is available, the creditor here has information about the customer's payment behavior over a longer period of time. This additional ex-post information is now used together with the regular customer data for the prediction. Examples in the literature are related to telecommunications service providers [4], [5], [6], credit card companies [7], microfinancing services [8], banks [2] or taxation authorities [9]. Possible actions are reminders, transfer to lawyers, transfer to debt collection agencies [10], [11], but also limitation or discontinuation of the service [4]. Since this situation can be viewed as a deferred credit scoring, it is no surprise that similar classification methods have been applied, e.g. k-nearest neighbor, random forest, SVM, neural networks, and naive Bayes classification [10], [8], [5]. Other approaches are rule-based [7] or using fuzzy sets [8].

A contrary situation arises in hospitals that follow a “rescue first, pay after” strategy. For patients who cannot pay their bill, the hospital has almost no backward information and can only make a repayment prediction based on currently available personal data (age, marital status, occupation, etc.). [12] present such a case and use eight different classification methods, among them Bayesian networks, for the repayment prediction.

So et al. [2] is the work closest to our research. They formulate debt collection as a stochastic dynamic programming problem. The objective is to find an optimal solution for which actions should be carried out for how long and in which order. An example with three actions (or phases) is given: (1) communicating with the debtor, (2) using legal procedures, and (3) writing off the debt. The model is applied to a direct provider-customer situation, the provider being a bank. Accordingly, the creditor has knowledge on the debtor's previous repayment behavior.

Hardly any publication is addressing decision making at debt collection agencies. An exception is [13], who present a solution for the following decision problem: should a portfolio of customer insolvency cases offered by a company be bought by a debt collector agency and at which price? The collector has to make an ex-ante prediction on the overall repayment and balance the incurred collection costs with the expected amount of recoverably money.

Our presented problem differs from all others in that decisions can almost only be based on information that is captured in the course of the collection process. Almost every activity of the process changes the probability of repayment. We selected Bayesian networks because of their ability to model conditional and marginal probabilities, their ability to combine expert knowledge with data, their comprehensibility based on a graphic representation, and their predictive performance. Our use of Bayesian networks is dynamic in the process, and it is not comparable with the static application of Bayesian networks as presented in the work of Shi et al. [12].

3. Bayesian Networks

A Bayesian network is a graphical representation of a set of random variables $X_1, X_2, \dots, X_n$ and the relationships that exist between them. The variables are symbolized by nodes and directed arcs between the nodes indicate causal or influential relationships between the individual variables. In most practical problems, the structure is relatively sparse, such that every node has only a few nodes that influence it (parent nodes) and every node has only a few nodes that it influences (children nodes).

In mathematical terms, the Bayesian network reflects a joint probability distribution $P(X_1, X_2, \dots, X_n)$ of the *n* variables. From probability theory, we know that the joint probability can be factorized into:

$$P(X_1, X_2, \dots, X_n) = \\ P(X_1)P(X_2|X_1) \dots P(X_n|X_1, \dots X_{n-1}).$$

Since a node is dependent only on its (few) parent nodes, this reduces to

$$P(X_1, X_2, \dots, X_n) = \prod_i P(X_i|Parents(X_i)) \\ \text{with } Parents(X_i) \subseteq \{X_1, \dots, X_{i-1}\}.$$

This formula allows to calculate the joint probability distribution $P(X_1, X_2, \dots, X_n)$ and the marginal probability distributions $P(X_1)$, $P(X_2)$, $P(X_n)$, given that the conditional probability distributions $P(X_i|Parents(X_i))$ are known. In the next section we will describe how the conditional probability distributions will be determined.

A typical usage scenario of Bayesian networks is *inference* or *causal reasoning*: inference means determining $P(X_i|X_j = x_j, X_k = x_k, X_l = x_l, \dots)$, the probability that X_i occurs after the realization $x_j, x_k, x_l, \dots$ of the variables $X_j, X_k, X_l, \dots$ is known (i.e. could been observed). Often, X_i is called the *variable of interest*, or *query variable* X, whereas the observed variables $X_j, X_k, X_l, \dots$ are called the *evidence variables* $E_j, E_k, E_l, \dots$, so that the conditional probability is written as

$$P(X|E_j = e_j, E_k = e_k, E_l = e_l, \dots).$$

However, it should be pointed out that any variable in the network could be a query or an evidence variable.

The determination of the relevant variables, the network structure, and the conditional probability distributions together are called the *construction* of the Bayesian network.

Construction of the Bayesian Network

Relevant Variables. The careful selection of relevant variables is the first important step in constructing a Bayesian network. The selection should be guided by the objectives of the project and the available information. Domain knowledge is necessary; therefore, experts should be included in the discussion.

Structure Learning. Very often, the domain expert already has an opinion on the direct relationships between certain variables. Thus, a dependency graph could be manually constructed. But depending on the complexity of the problem and the amount of data available, automatic learning from data might be the preferable option.

Structure learning for Bayesian networks is a NP-hard problem [14] and therefore intractable for larger networks. As a consequence, heuristic methods have been developed that use *search and score* approaches. The score is a measure of how well the structure fits to the given data. Frequently used scores are the Akaike information criterion (AIC), the Bayesian information criterion (BIC), and the log-likelihood criterion (LOGLIK) [15]. The search algorithms used in this research are the hill-climbing greedy search (HC) [15], the tabu search (TABU) [16], max-min hill-climbing (MMHC) [17], and the 2-phase restricted maximization HITON parents and children (RSMAX2) [18].

Hill-climbing greedy search (HC) and tabu search (TABU) are widely applicable heuristics which can also be applied to Bayesian networks. HC [15] starts with an arbitrary network structure for which the selected score

(BIC) is calculated. New structures are generated by randomly adding, removing, or reversing edges, and a new structure replaces the old structure if it has a higher score. The algorithm stops if modifications do not result in a further score improvement. It is obvious that hill-climbing can easily lead to local optima. Tabu search [16] is a variation of HC that avoids getting trapped in local optima by keeping a list of already rejected solutions. HC and TABU are known as score-based algorithms.

Other algorithms look deeper into the variable relationships to restrict the possible network structures for the search. They are called hybrid algorithms because they combine the score-based with a constraint-based approach (maximize and restrict). Max-min hill-climbing (MMHC) [17] starts with a skeleton (i.e. a graph with undirected edges) of the Bayesian network, learned by a local discovery algorithm called max-min parents and children (MMPC). Hill-climbing greedy search is then used to orient the edges in the network. A more general implementation of MMHC is RS2MAX (2-phase restricted maximization). It can combine different constraint-based and score-based algorithms. In this hybrid approach, we used HITON parents and children (RSMAX2) [18] to construct the skeleton and hill-climbing greedy search to orient the edges in the network.

Parameter Learning. Once the structure of the Bayesian network has been established, the conditional probability distributions need to be assigned to each node. While structure learning represents the qualitative subtask of the network construction, parameter learning represents the quantitative subtask. The conditional probability distributions are often estimated with the maximum-likelihood approach from the observed frequencies in the dataset associated with the network. In the case of discrete variables, it can simply be calculated by counting how often the value X_i^k of the variable X_i is occurring in conjunction with each possible realization $Parents(X_i)^j$ of the parent variables $Parents(X_i)$, divided by the total occurrence of this realization [20]:

$$P\left(X_i^k \middle| Parents(X_i)^j\right) = \frac{Count(X_i^k, Parents(X_i)^j)}{Count(Parents(X_i)^j)}$$

Maximum likelihood estimation (MLE) is the classical frequentist approach to parameter learning. If uncertainty is to be considered in parameter estimation, then Bayesian estimation (BE) is preferable. BE corrects to formula above by adding the prior probabilities to the terms in the numerator and denominator [20]. The advantage of Bayesian estimates is that they deliver more realistic results for small data sets [20] and that missing observations don't result in zero probabilities in the network [15]. However, as the size of the data set increases, the two estimates converge.

Inference. After structure and parameters of the Bayesian network have been determined, the network is „ready. As already mentioned, the primary usage scenario is *inference*, also called *causal reasoning* or *belief updating*: given that the values of some of the variables are known (evidence variables), what is the probability that a node X (the query node) has a particular value of x?

While this conditional probability can easily be calculated for small network structures, the computational effort increases more than exponentially for larger networks. Therefore, in addition to the exact approach, approximate interference algorithms have been proposed. Two well-known algorithms are simulation-based and called *logic sampling* and *likelihood weighting* [21].

Bayesian Network Classifier

We introduced Bayesian inference as the problem of identifying the conditional probabilities $P\left(X \middle| E_j = e_j, E_k = e_k, E_l = e_l, \ldots\right)$ of a variable X given evidence $E_j = e_j, E_k = e_k, E_l = e_l, \ldots$. If X is a discrete variable that can take on the values x (e.g. $x = 0$ and $x = 1$ for a binary variable), then an assignment of the *evidence* (or now called *predictor variables*) $E_j = e_j, E_k = e_k, E_l = e_l, \ldots$. to

$$\underset{x}{\operatorname{argmax}} P(X = x | E_j = e_j, E_k = e_k, E_l = e_l, \ldots),$$

the value of X with the highest posterior probability, is a *classification* with the *class variable* X. By using a Bayesian network in this way, it becomes a *Bayesian network classifier* [22].

4. Bayesian Network Analysis of the Debt Collection Process

4.1. Data and Variables

The presented research was conducted in collaboration with a German debt collection agency whose clients are companies from all industries. Their debt collection process is fully digitized, i.e. the date of invoice, the amount due, the complete history of actions taken, etc. are all electronically available and accessible by the agents during the entire process.

Due to the different approaches and types of communication, only claims against private individuals were considered in the present study; corporate debtors were excluded. The claims accessible to us originate from a period from March 2016 to June 2017. Cases were declared as *successful* or *positive*, if after one year, more than 95% of the total debt was recovered. They were declared as *unsuccessful* or *negative*, if less than 5% were recovered. The small percentage of cases (2,76%) with partial repayments (between 5% and 95%) was excluded. This resulted in a number of 54.537 files, of which 40.549 (74,36%) were positive and 13.988 (25,64%) were negative.

The debt collection system allows 688 different types of actions. A total of 6.7 million actions were recorded in the entirety of files. Together with the agents, we selected 117 of these action types as relevant for evaluating the collection process (other types were purely technical). These action types can be classified into 19 categories that became the underlying variables of the Bayesian network (Table 1).

4.2. Network Construction

After all data had been prepared and variables been defined, the network construction was carried out using the R package bnlearn [23]. The four algorithms HC, TABU, MMHC and RSMAX2 were used for structure learning. The algorithms created different network structures which were more or less complex.

HC and TABU show the best results for the three indicators AIC, BIC and LOGLIK; the TABU model has a minimal lead (Table 2). This corresponds to the expectation, since the TABU algorithm, in contrast to Hill-Climbing Greedy Search (HC), can overcome local optima. The MMHC and RSMAX2 models are a bit behind because they optimize BIC only in the second step.

Due to the limited space, we present the structures of only two networks exemplarily. Figure 2 and Figure 3 show the Bayesian networks which result from applying the MMHC and HC algorithms.

The network structures were generated solely by learning from data. They have no prior information on how the debt collection process should look like. Different learning algorithms identify different dependencies between the variables. But at their core, both presented models show a sequence of actions that is predetermined by the legal dunning procedure: transfer to lawyer, court order, enforcement notice, enforcement. Even if the nodes of the network are random variables, not process steps, one could say that the learning algorithms have recreated at least parts of the collection process from data. In this respect, Bayesian network construction is comparable to process discovery techniques used in process mining [24].

The BIC-maximizing model HC (same as TABU) has a very confusing structure. The large amount of dependencies may optimize the BIC, but they are difficult to understand and could be caused by overfitting. The most transparent structure is the MMHC model. The agent's actions are reflected in a straightforward escalation sequence which ultimately leads to repayment. Installment agreements and negative features accompany this sequence. The contact to the debtor via the telephone is positioned in a side branch of the network, which is linked to full recovery via the variable verbal commitment to pay. The variable arrest warrant is isolated from the rest of the network, i.e. it is recognized as independent from all other variables. This means that in the MMHC model, the presence of an arrest warrant doesn't change the probability of a repayment.

Table 1. Defined network variables, descriptions and frequencies (in alphabetical order).

Variable	Description	Relative frequency in all files
Announcement of Measures	Follow-up measures are announced to the debtor, e.g., the judicial dunning procedure or a foreclosure.	36,7%
Arrest Warrant	Coercive measure against the debtor to provide a statement of assets.	0,3%
Contact Refusal	The debtor refuses the contact with the debt collection agency, e.g., by refusing to take note of letters or not answering calls.	0,4%
Court Order	Delivery of the payment order to the debtor by the competent dunning court.	38,0%
Enforcement	Execution of foreclosure against the debtor. Valuables or money of the debtor are garnished (usually by a bailiff).	16,1%
Enforcement Notice	Issuance of an enforcement notice.	31,1%
Formal Objection	The debtor files a formal objection, querying all or part of the claim.	2,4%
2nd Formal Reminder	The debt collection agency sends a letter containing the second reminder.	60,2%
3rd Formal Reminder	The debt collection agency sends a letter containing the third reminder.	50,8%
Formal Reminder by Lawyer	Dispatch of a reminder by the lawyer.	50,4%
Full Recovery	Settlement of all outstanding debt. The recovery is considered as full, if the remaining debt (including fees) is less than 5%.	74,4%
Installment Agreement	Agreement of an installment plan between the debt collection agency and the debtor.	16,5%
Installment Reminder	In case of missing installments, the debtor is reminded in writing of his obligation to pay.	11,3%
Negative Communication	The debtor announces not to pay the claim.	0,2%
Neutral Communication	The debtor asks for information and announces to review the process.	1,4%
Ongoing Bankruptcy Proceedings	The debtor has declared personal bankruptcy within the past seven years before the start of the file.	4,8%
Other Negative Indicators	Information on negative indicators in past files such as financial and property information, foreclosure auction, etc.	23,5%
Signaling of Insolvency	The debtor verbally announces payment difficulties, e. g. due to job loss or private bankruptcy.	2,8%
Telephone Collections	Execution of telephone collections.	3,5%
Verbal Commitment to Pay	The debtor verbally announces to settle the claim.	8,2%

Table 2. Average network scores AIC, BIC and LOGLIK.

	HC	TABU	RSMAX2	MMHC
AIC	-226287	-226260	-231749	-229113
BIC	-227361	-227306	-232214	-229638
LOGLIK	-226042	-226022	-231643	-228993

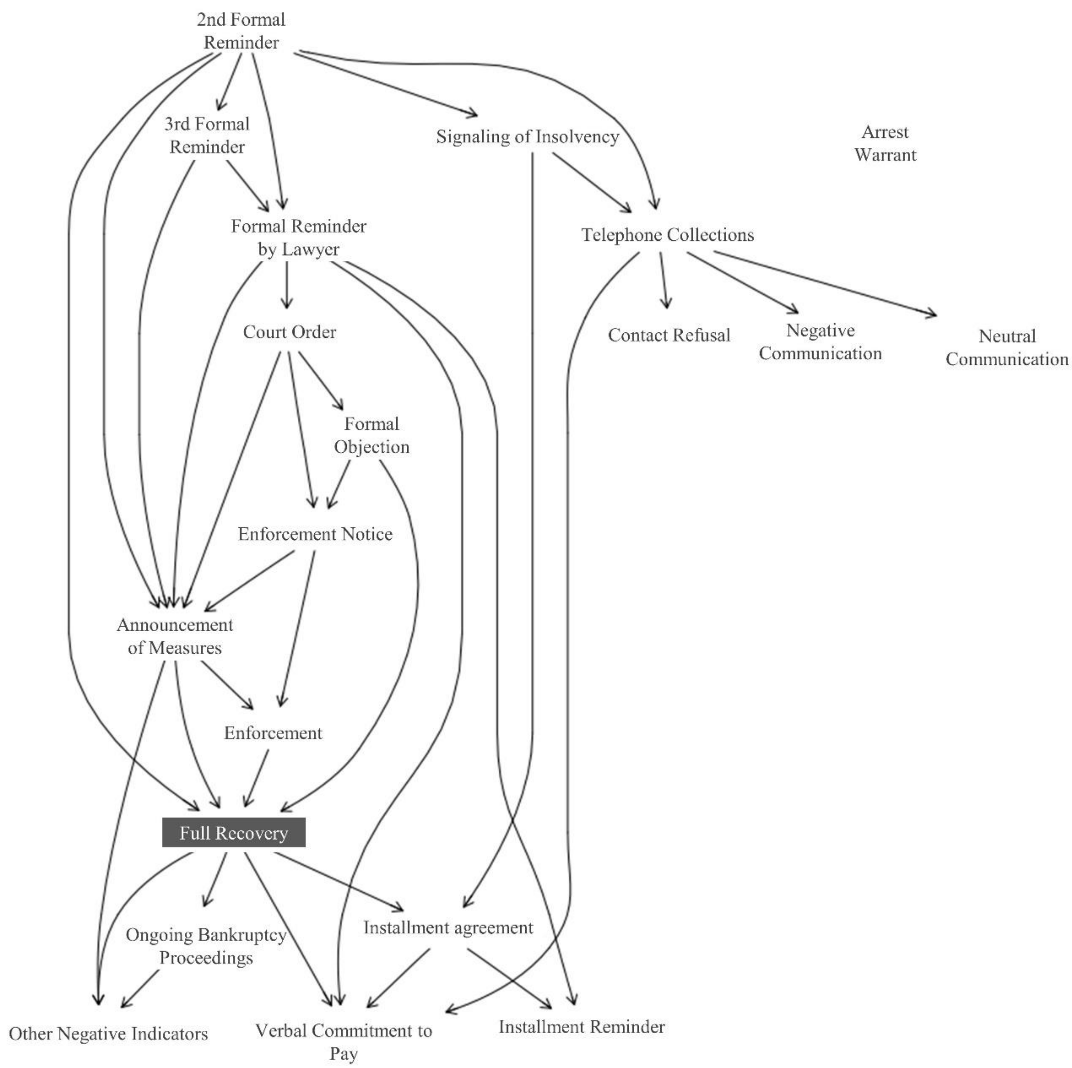

Figure 2. Structure using Max-Min Hill-Climbing (Model MMHC).

After the structure of the networks was established, parameter learning was carried out on the basis of Bayesian estimates (BE). Inference was sampled using the likelihood weighting method.

4.3. Classification

Our first objective was to make a prediction at any time - from the start of the process - as to whether the case will ultimately be positive, i.e. whether the debtor will pay his outstanding debts in full (≥ 95%). This prediction can be achieved by Bayesian network classification with the target variable full recovery. Full recovery receives a value of 1, if the file is positive (successful), and a value of 0, if it is negative (unsuccessful). The evidence variables corresponding to agent's actions (formal reminders, telephone collections, transfer to lawyer) or debtor's responses (contact refusal, formal objection) are set to 1 if the action has taken place.

Classification is a well understood and widely practiced task in machine learning. Standard procedures and techniques for learning and performance evaluation are available. One such technique concerns the handling of the dataset. We used 10-fold cross-validation on the available data to avoid bias and overfitting. The classification results were evaluated using the standard performance measures precision, recall, and F1-score (the harmonic mean of precision and recall). For comparison, we also applied a naive Bayes model for the target variable full recovery, which assumes statistical independence between all influencing variables and could be drawn as a simple, two-layer, tree-structured Bayesian network.

Benchmark for all models is the trivial classifier that simply classifies all files as positive (because they are in

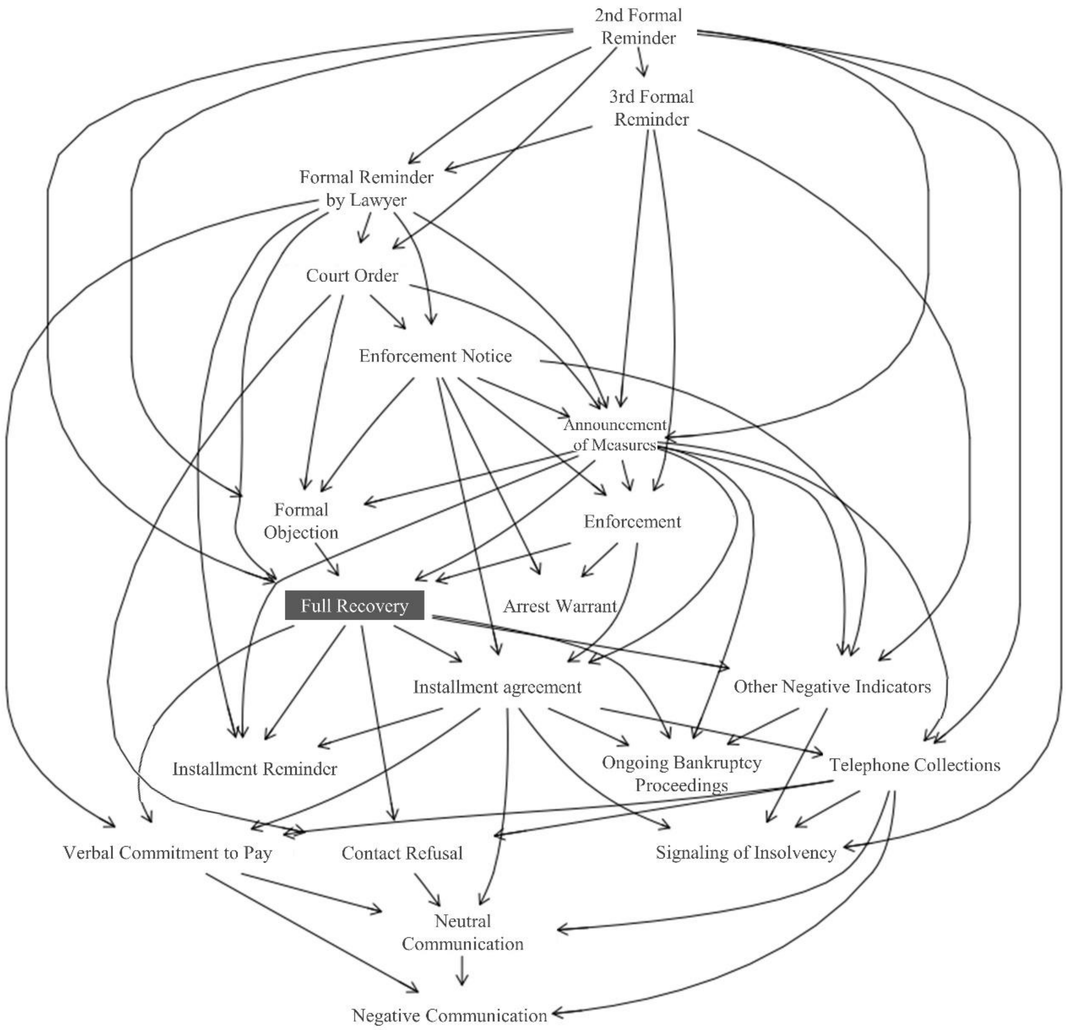

Figure 3. Structure using Hill-Climbing Greedy Search (Model HC).

the majority). According to the share of 74.36% positive and 25.64% negative files, the trivial classifier will achieve a precision of 0.7436, a recall of 1.0, and an F1-score of 0.862.

Table 3 shows the classification performance of the networks that result from applying the different structure learning algorithms.

Table 3. Classification performance of the different networks

	HC	TABU	RSMAX2	MMHC	NAIVE
Precision	88,9%	88,8%	87,4%	88,5%	88,9%
Recall	93,5%	93,5%	94,2%	93,7%	83,8%
F1-score	91,1%	91,1%	90,7%	91,0%	86,3%

The precision results of all models are between 0.86 and 0.89. This means that 86% to 89% of the files that are classified as positive are indeed positive. The RSMAX2 model lags a little behind. A clearer picture emerges for the recall, since the naive model consistently performs around 10% worse than the models obtained through structure learning. Their results are all in a very narrow range between 93% and 94.2%. This means that 93% and 94.2% of the files that were truly positive were identified by the algorithms as positive. The trained models also achieve very good results in identifying positives without losing precision, which is reflected in the F1-score. Due to its poor recall, the naive model achieved an F1-score of only 86.3% on average, which is only slightly above that of the trivial classifier (86.2%).

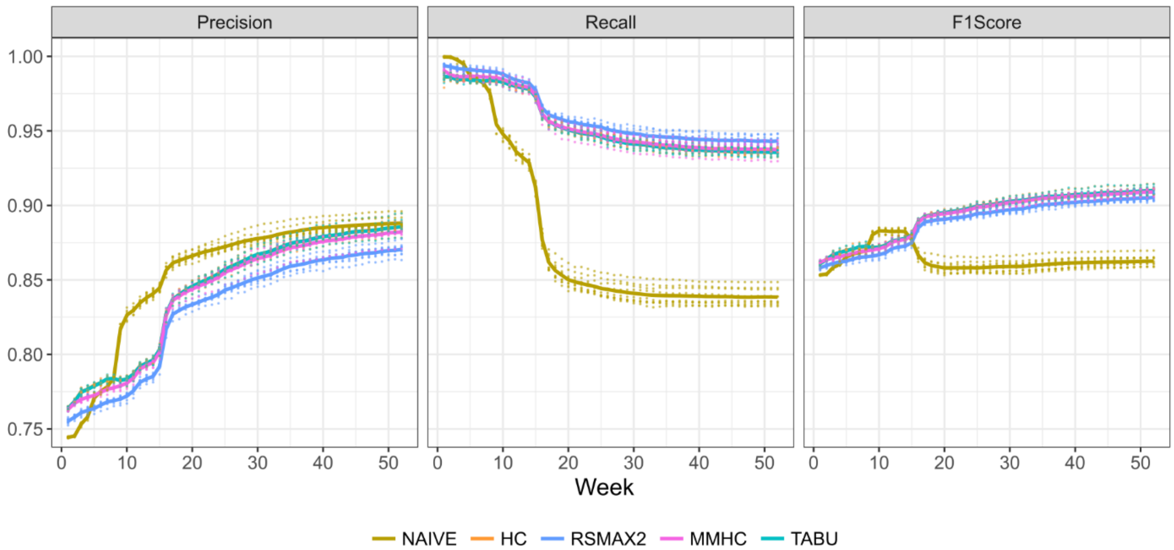

Figure 4. Classification performance of the different networks in weekly progress.

The classification results shown in Table 3 are based on the data available in the file after one year. On this basis, the models predict whether the case will be positive or negative. Essential to supporting case management, however, is a continuous assessment of the outcome as the process continues to evolve. Therefore, the measures precision, recall and F1-score were also evaluated over time. In order to estimate the success of the files after one year, the Bayesian networks were fed with the information available up to processing week w (w = 1, . . . 52). Figure 4 shows the results for all considered networks. For each model, the superimposed points represent the results of the ten folds; solid lines show the weekly arithmetic mean.

The almost identical start values for precision and recall result from the scarcely available information on actions at the beginning of process. There are still no evidences available that would change the conditional probabilities in the network.

As with the static analysis after one year (Table 3), the algorithmically learned models behave very similarly over time; only the naive model deviates from it. The course of the naive model is remarkable; it dominates in the precision, but drops extremely sharply in the recall. The model classifies relatively few files incorrectly as positive, but on the other side, many files that are actually positive are not recognized as such. This results in a very low F1-score from week 20 on, which is in the range of the trivial classifier.

The models HC, TABU, RSMAX2 and MMHC are a little less performant in the first few weeks. From week 15 on, however, they clearly surpass the naive, and especially the trivial classifier. A little behind is RSMAX2, which is slightly better in terms of recall, but clearly worse in terms of precision. The best performers are HC and TABU, which, despite the different structures, can hardly be distinguished in the diagram and achieve slightly better values than the MMHC model.

The models HC and TABU show the best results for BIC and predictive performance. However, their structures are complex and some of the identified dependencies are difficult to understand. The MMHC model is hardly inferior to HC and TABU, but the dependency structure is much more transparent and comprehensible, reflecting the legally determined sequence of actions at their core. The RSMAX2 model is similarly transparent, however, achieves slightly worse performance results. Had a decision to be made, however, the MMHC model would be the favorite because of the good performance results and the transparent structure.

4.4. Sensitivity Analysis

In addition to the analysis of the network structures and their prediction performance, a one-dimensional sensitivity analysis was carried out. Sensitivity analysis answers the question: how sensitive is the target variable to small changes in the evidence values [25]?

The sensitivity analysis (see Figure 5) showed that some of the actions represented by network variables, like formal reminders by the debt collection agency or the execution of telephone collections, had almost no influence on the assessment of the full recovery probability. Only increasing the probability for the handover of the file to the lawyer and the subsequent judicial dunning procedure with the steps court order, enforcement notice and enforcement (left) change the repayment probability significantly. The necessity that an action must be taken makes it less likely that the case will end positively. The same is true for a debtor

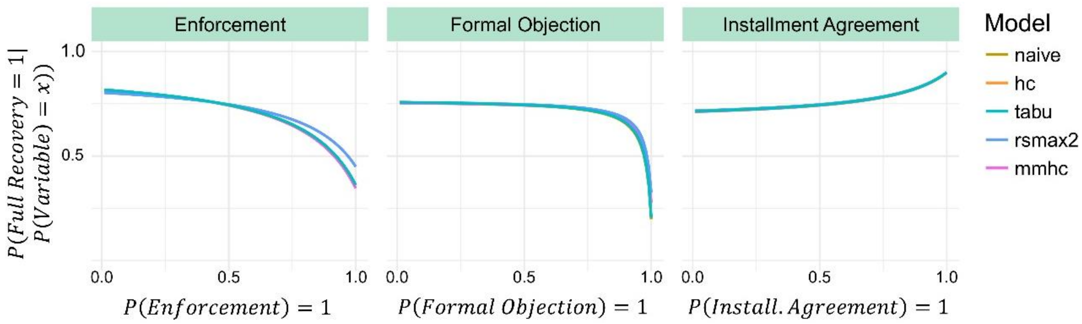

Figure 5. Sensitivity analysis of the variable Full Recovery (Excerpt of the variables Enforcement, Formal Objection and Installment Agreement).

response such as a formal objection (middle). In contrast to that, evidence on installment agreement (right) and verbal commitment to pay have a positive effect on the prospects of success.

Based on this information, the agent in charge of the process can continuously evaluate the prospects of the case to determine whether it makes economic sense to continue the process or whether the file should be closed.

5. Conclusion

The objective of this study was to demonstrate that analytical, data-driven methods can be used as the basis for a decision support system for the debt collection process. It turned out that Bayesian networks best met the requirements of the problem. Unlike credit scoring, debt collection has very little data at hand at the beginning of the process but relies on data generated during the process. Bayesian networks are strong in dealing with uncertainties. They can be used to predict the probability of the debtor's full repayment at the beginning of the process. This prediction can be improved in the course of the process as more and more information on the collector's actions and debtor's responses becomes available. The prediction quality is very good and gets better and better as time goes on. This allows the agent to decide at any time whether the case should be continued or not.

However, the system not only predicts the prospects of the case, but also makes recommendations to the agent as to which actions will have an impact on the probability of repayment. This information is obtained through a sensitivity analysis of the Bayesian network.

Bayesian networks are a valuable aid to support the agents in the debt collection process. They are predictive, but not prescriptive. Agents still have to make their own decisions - but now on a much better information basis.

A possible next step might be the extension of the Bayesian networks to *decision networks* [20]. These distinguish more clearly between nodes which can be influenced by decisions (actions) and nodes which represent uncertainties. The decision nodes in the network can additionally be attached with costs, so that a sequence of decisions could be determined which optimizes the overall utility (expected repayment netted with collection costs). The present work establishes the basis for such a decision network.

6. Literature

[1] https://www.amk.world/one-in-five-of-uaes-smes-say-late-payments-are-a-threat-to-their-business/, last accessed on 14-06-2021.

[2] So, M. C., Mues, C., de Almeida Filho, A. T., & Thomas, L. C. (2019). Debtor level collection operations using Bayesian dynamic programming. *Journal of the Operational Research Society*, *70*(8), 1332-1348.

[3] Louzada, F., Ara, A., & Fernandes, G. B. (2016). Classification methods applied to credit scoring: Systematic review and overall comparison. *Surveys in Operations Research and Management Science*, 21(2), 117-134.

[4] Bahrami, M., Bozkaya, B., & Balcisoy, S. (2020). Using behavioral analytics to predict customer invoice payment. *Big data*, *8*(1), 25-37.

[5] Onar, S. C., Oztaysi, B., Kahraman, C., & Öztürk, E. (2019, July). A decision support system to optimize debt collection assignments. In *International Conference on Intelligent and Fuzzy Systems* (pp. 178-187). Springer, Cham.

[6] Onar, S. C., Oztaysi, B., Kahraman, C., & Ozturk, E. (2020). Evaluation of legal debt collection services by using Hesitant Pythagorean (Intuitionistic Type 2) fuzzy AHP. *Journal of Intelligent & Fuzzy Systems*, *38*(1), 883-894.

[7] Chin, A. G., & Kotak, H. (2006). Improving debt collection processes using rule-based decision engines: A case study of Capital One. *International journal of information management*, *26*(1), 81-88.

[8] Wang, J., Zhang, L., Shen, P., Zhu, G., & Zhang, Y. (2018, October). Preprocessing and feature

extraction methods for microfinance overdue data. In *CCF Conference on Big Data* (pp. 23-43). Springer, Singapore.

[9] Abe, N., Melville, P., Pendus, C., Reddy, C. K., Jensen, D. L., Thomas, V. P., ... & Gardinier, T. (2010, July). Optimizing debt collections using constrained reinforcement learning. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 75-84).

[10] Kim, J., & Kang, P. (2016). Late payment prediction models for fair allocation of customer contact lists to call center agents. *Decision Support Systems*, *85*, 84-101.

[11] van de Geer, R., Wang, Q., & Bhulai, S. (2018). Data-driven consumer debt collection via machine learning and approximate dynamic programming. *Available at SSRN 3250755*.

[12] Shi, D., Zurada, J., & Guan, J. (2016, January). A Bayesian network approach to classifying bad debt in hospitals. In 2016 49th Hawaii International Conference on System Sciences (HICSS) (pp. 3298-3307). IEEE.

[13] Kajdanowicz, T., & Kazienko, P. (2009, November). Prediction of sequential values for debt recovery. In *Iberoamerican Congress on Pattern Recognition* (pp. 337-344). Springer, Berlin, Heidelberg.

[14] Chickering, D., Heckerman, D. & Meek, C. (2004). Large-Sample Learning of Bayesian Networks is NP-Hard. *Journal of Machine Learning Research*, 5, 1287-1330.

[15] Nagarajan, R. et al. (2013). *Bayesian Networks in R: with Applications in Systems Biology* (M. Scutari & S. Levebre, Hrsg.). New York, NY, USA: Springer.

[16] Glover, F. & Laguna, M. (1997). *Tabu search*. Boston: Kluwer Academic.

[18] Aliferis, C., Tsamardinos, I. & Statnikov, A. (2003). HITON: A Novel Markov Blanket Algorithm for Optimal Variable Selection. *AMIA Annual Symposium proceedings*, 21-5.

[17] Tsamardinos, I., Brown, L. E. & Aliferis, C. F. (2006). The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65 (1), 31-78.

[20] Ji, Z., Xia, Q., & Meng, G. (2015, August). A review of parameter learning methods in Bayesian network. In *International Conference on Intelligent Computing* (pp. 3-12). Springer, Cham.

[20] Koller, D. & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. Cambridge, MA, USA: MIT Press.

[21] Shachter, R. D., & Peot, M. A. (1990). Simulation approaches to general probabilistic inference on belief networks. In *Machine Intelligence and Pattern Recognition* (Vol. 10, pp. 221-231). North-Holland.

[22] Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine learning*, 29(2), 131-163.

[23] Scutari, M. (2010). Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, 35 (3), 1{22.

[24] van der Aalst, W. (2016). *Process Mining: Data Science in Action*. Berlin, Heidelberg: Springer.

[25] Castillo, E., Gutiérrez, J. M., & Hadi, A. S. (1997). Sensitivity analysis in discrete Bayesian networks. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *27*(4), 412-423.