Improving a Rule-based Fraud Detection System with Classification Based on Association Rule Mining

Michaela Baumann[1]

Abstract: The detection of fraudulent insurance claims is a great challenge for insurance companies. Although the detection possibilities are getting better and better, fraudsters do not hesitate also using newer and more sophisticated methods. Apart from establishing new fraud detection systems, also the existing systems need to be updated and improved as best as possible. One common detection system is a rule-based expert system that checks predefined rules and gives alerts when certain conditions are met. Usually, the rules are treated separately and correlations within the rules are considered insufficiently. The work at hand describes how the classification based on association rule mining is used for improving such rule-based systems by bringing in relations between pairs of rules. The rule weights are determined through a genetic optimizer.

Keywords: Insurance fraud detection; Association rule mining; Expert system; Genetic optimizer; Classification

1 Introduction

The business model of insurance companies is the financial hedging of certain risks of their customers. The compensation in the collective, i.e., that many customers pay a relatively small amount of money and get a compensation in the event of a claim, makes this business worthwhile [Fa11, Ko13]. However, in all the working areas of an insurer, e.g., when designing products or calculating premiums, the notion of moral risk resp. moral hazard is present [He89]. The availability of an insurance may be disincentiving and delude an insured to take risks he or she would not have taken without the insurance [DB00]. Moral hazard may occur when a certain information asymmetry is present, i.e., when the insured knows more about its own risk-taking than the insurer does [Hö79, Sh92]. Insurance fraud constitutes a certain kind of moral hazard whereby its definition is not fixed per se. A very broad definition includes the excessive, involuntary, or unnecessary use of insurance benefits [Sc04]. A stricter definition of fraud is that it implies a criminal act with a serious misrepresentation, the intention to deceive, and the objective of obtaining an unjustified benefit [DZ02, De02, ED03, VD04]. Apart from the fact that a watertight proof of fraud is rarely possible [VD04], it is clear that insurance fraud leads to serious problems on the insurance market irrespective of the definition or type of fraud. The German insurance association ("Gesamtverband der Deutschen Versicherungswirtschaft", GDV) estimates the

[1] NÜRNBERGER Versicherung, Anwendungsentwicklung BIACC, Ostendstraße 100, 90334 Nürnberg, Germany
michaela.baumann@nuernberger.de

damage caused by insurance fraud at 5 billion ($5 \cdot 10^9$) EUR in 2018 in Germany with about 10 % of all claims assessed as dubious [Fr18]. The damage is thereby not only a problem of the insurance company. Fraudulent claims increase the amount of claimed money which is passed on to all policyholders through an increase of the premiums or a reduction of the insurance coverage. However, dissatisfied or annoyed customers usually only blame the insurance company since the principle of insurance is often not clear to them [Kö15, VD04]. It is therefore necessary for insurance companies to detect and prevent insurance fraud as best as possible to compete on the market and indirectly maintain the (hopefully) good image of the company. However, this is a difficult task and resembles a tightrope walk since too much mistrust on the part of the insurers or even a general suspicion towards the insured has an extremely negative effect on the insurers' image. This is why fraud detection receives a lot of attention and especially during the last years, new methods from the fields of big data, machine learning, and AI have found their way into this field. The work at hand focuses on insurance fraud, more specifically on insurance fraud at claim time, and contributes to the efforts of fraud detection by showing a way of how a so-called expert system can be improved with a relatively little expense using association rule mining. Thereby, association rule mining is used to find correlations between existing rules of the expert system. Considering these correlations and adding them in form of additional rules to the original rule set might allow for a more accurate classification of insurance claims.

The paper is organized as follows: Section 2 gives a non-exhaustive overview of current methods for fraud detection. Section 3 shows the structure and the functioning of an expert system (Section 3.1), how classification with association rule mining works in general (Section 3.2), and how it is applied to the rule-based expert system (Section 3.3). In Section 4, the association rule mining is applied on a real-world expert system and it is shown how this technique can improve its expressiveness. Section 5 concludes the paper.

2 Related Work on Fraud Detection Methods

The ways of fraudulent behavior in the insurance sector as well as, e.g., credit card fraud, telecommunication fraud, or money laundering are manifold, as have to be the means to detect or prevent such behavior [BH02, Ko04, AMZ16]. When focusing on fraud at claim time, there are some general difficulties related to fraud detection. One difficulty is that fraud is not self-revealing [VD04], meaning that fraud has to be actively looked for and it has to be detected as early as possible. Furthermore, and this is especially important when trying to use supervised methods for fraud detection, it is difficult to actually prove fraud. Most target variables in supervised learning techniques flag suspicious claims, but not necessarily fraudulent claims. Methods for unsupervised fraud detection circumvent this problem and are mostly based on outlier detection and profiling [BH02].

One possibility of detecting claim fraud is to use rule-based expert systems. Such systems match every claim with a list of predefined indicators and when a claim shows a certain set of indicators, it is marked as suspicious [De02, VD04]. The construction of such systems usually requires labeled data or reliable expert assessments and the rules themselves are of the form *If (condition) then (consequent)* [BH02]. Rule-based expert systems are, for example, used for detecting superimposed fraud cases in telecommunications networks [Hi09], for detecting computer (network) intrusions [GL91], for detecting anomalies in healthcare insurance claims [SA13], or for alerting fraud in the field of consumer credits [Le95]. An automatical re-engineering of rule-based expert systems may be conducted with cultural algorithms [SR97]. In such a setting, every rule is associated with a certain score, i.e., the rules are of the form *If (condition i) then add p_i to the fraud score* and the sum of the scores is a measure for the fraud likelihood. Apart from rule-based expert systems, there are also supervised machine learning methods for classifying claims into fraudulent and non-fraudulent ones such as logistic regression, k-nearest neighbors, decision trees, Bayesian neural networks, or support vector machines (SVMs) for classification [Vi02]. In particular, there exist neural networks for detecting management fraud [GC97, FCS95], neural and Bayesian networks for detecting fraud in communications networks [Ta98], decision trees and SVMs for detecting credit card fraud [ŞD11], or data mining models for detecting anomalies in healthcare insurance claims [SA13]. Fraud in the automobile insurance market is, e.g., predicted with (nested) multinomial logistic regression models [AAG99]. For training supervised models, not only traditional numeric features may be used but also, e.g., text features extracted from accident descriptions via text processing algorithms such as Latent Dirichlet Allocation [WX18]. When assessing the algorithms' performance, a particular focus lies on the Type I (false positive) and Type II (false negative) error rates [GC97]. As the imbalance of the data, i.e., a disproportionate ratio of observations in each class, is a big problem when applying supervised machine learning methods, there also exists a bundle of unsupervised methods for fraud detection such as an interactive, unsupervised machine learning approach for detecting fraud in the healthcare insurance [KGK15] or the so-called Peer Group Analysis, an unsupervised method for fraud detection where similar objects and their behavior around a target sequence is used for predicting the expected behavior of the target, for analyzing time series data with respect to outliers in financial data [FM06, BH01]. With a similar approach time series data of mobile phone users can be analyzed for fraudulent cases [BST01]. With unsupervised spectral ranking techniques outliers, for instance auto insurance fraudsters, can be ranked by their degree of deviation [Ni16]. Other anomaly detection methods imply a network analysis, e.g., social network analysis for detecting internet auction fraud [Ch11], fraudulent credit card transactions [Va15], automobile insurance fraud [ŠFB11], or money laundering [FCR17]. In addition to systems working with the internal data of an organization, fraud detection can also be efficiently supported with centralized insurance fraud bureaus [De02]. In Germany, e.g., there exists the HIS ("Hinweis- und Informationssystem"), an indication and information system of the German insurance industry, which supports the detection and prevention of insurance fraud [iHG]. Such centralized bureaus allow a certain degree of information pooling, whereby the insurers do not have direct access to the data to preserve honest competition and to fulfill all privacy issues [VD04]. Independent of the specific fraud detection methods, relevant fraud detection components must be chosen and it needs to be decided how they are combined into an organization's fraud detection system [FVB11].

During all efforts for detecting or preventing fraud, the most important issue is to keep the balance between effort and outcome [De02], in particular, the relationship between the insurer's investment for preventing/detecting fraud and the effort costs of the policyholder for different behavior [Ok13].

3 Adjusting the Rule-based Fraud Detection System

As stated in Section 1, we improve the detection of insurance fraud by extending an already existing rule-based expert system with very little effort. For doing this, we first characterize the rule-based system and show the functioning of a classification based on association rule mining (CARM) before applying the CARM on the rule-based system.

3.1 Characterization of Rule-based Expert Systems and Problem Description

The expert system, which is improved hereafter, with a CARM is a rule-based system similar to that one described by Sternberg and Reynolds [SR97]. During the car claims process, relevant information, e.g., the time and place of the accident, is recorded. As soon as the information is stored, the rule-based system goes into action. The claim data is matched independently with a set of rules. Each rule can either be triggered, i.e., its conditions are met, or not. For example, the accident may have happened during the weekend (or not). Furthermore, each rule has a certain weight. For each claim, the weights of the triggered rules are aggregated and when this aggregation meets a specified threshold, further actions are to follow. For example, a person responsible should have a closer look on the claim.

Definition 1 (rule-based system)**.** A rule-based system is a vector $(\mathfrak{r},\mathfrak{w})$, where $\mathfrak{r} = (r_1, r_2, \ldots, r_k)$ is a vector of rules with $1 \leq k \in \mathbb{N}$. Each rule r_i has a weight $w_i > 0$ given in the weight vector $\mathfrak{w} = (w_1, \ldots, w_k)$. A rule is a function which evaluates a claim $c \in \mathcal{C}$ to $r_i(c) = 1$ when the rule is fulfilled, i.e., the condition is true, and to $r_i(c) = 0$ when it is not fulfilled. The whole claim assessment is done via an aggregation function g which depends on the weight vector $\mathfrak{w}$: $g_{\mathfrak{w}}(c) = g(r_1(c) \cdot w_1, \ldots, r_k(c) \cdot w_k) \geq 0$. An alert is signalled when $g_{\mathfrak{w}}(c) \geq l$ for some threshold $l > 0$.

An example for a rule-based system consisting of four rules could be: (a) If there are exactly two cars involved, assign 10 to the fraud score. (b) If the accident happened during the weekend, assign 5 to the fraud score. (c) If there is exactly one witness, assign 5 to the fraud score. (d) If the same insured reported another claim in the six months before, assign 30 to the fraud score. Written in the notation of Definition 1, we would get the rule vector $\mathfrak{r}_{ex} =$ (There are exactly two cars involved, The accident happened during the weekend, There is exactly one witness, The same insured reported another claim in the six months before) with the corresponding weight vector $\mathfrak{w}_{ex} = (10, 5, 5, 30)$. Note that with the notation above, the rule-based system evaluates the claims in two steps. In the first step, it is checked whether the condition of a rule is fulfilled or not leading to a rule conclusion of either 0 or 1, i.e., we get Boolean conditions. In a second step, each conclusion is multiplied with its respective weight. This leads to the same result as if the weight would have been directly part of the rule conclusion, but it allows to change the weighting in re-engineering phases [SR97] without having to change the rules themselves. One possibility for the aggregation function g is the summation [SR97], which we further assume if not otherwise stated. Continuing the example and assuming a certain claim c_{ex}, the rules could evaluate to $\mathfrak{r}_{ex}(c_{ex}) = (1,0,0,1)$. Using the summation for g this would lead to a fraud score of $g_{\mathfrak{w}_{ex}}(c_{ex}) = 10+0+0+30 = 40$.

The weights for the rules as well as the rules themselves try to reflect the experts' valuable knowledge of fraudulent behavior. However, depending on the specific rule formulations and the number of rules involved, the above definition does not ensure that dependencies between the various rule conditions are considered; a claim adjuster would most likely take into account such dependencies during a manual checking [VD04]. In the rule-based system, every rule has a certain fixed weight which is added to the aggregation independently of the fulfillment of the other rules with g being the summation. When looking at real-world claim data, we may encounter three different types of dependencies between every two rule conditions: (i) no dependence, i.e., the two conditions are truly independent, (ii) a reinforcing dependence, i.e., the fulfillment of the two rules at the same time is more severe than their aggregated single weights, or (iii) a weakening dependence, i.e., the two rules each have a certain base weight but their simultaneous occurrence is not problematic (it may even be the normal case).

This observation also implies that weights may not only be positive, strengthening the suspiciousness of a claim, but also negative, i.e., mitigating its suspiciousness. In the four-rule-example, it could be the case that when there are exactly two cars involved and there is exactly one witness, i.e., the two rules are true at the same time, this is a more suspicious situation than the $10+5$ fraud score may suggest. On the other hand, exactly one witness for an accident that happened during the weekend may not be as suspicious as the $5+5$ fraud score proposes. The question arises of how to bring rule dependencies into the model while taking account of a certain cost-efficiency. This problem may be divided into two sub-questions. The first one is: *(1) How can we detect meaningful dependencies among the set of all rules?* The second one tackles the issue of economical efficiency. Any change to the existing system should be avoided; in particular, the surrounding systems should not be affected by the adaptation. *(2) How can we add the additional information to the system without changing its current operating mode?* The answer to Question (2) is more or less simple: As we only have rules and weights in the system, the additional information has to be added as either rules or weights. Notice that adjusting the function g towards a more involved function is not practical, since every time a rule is added or a weight is changed, the function g would otherwise have to be entirely redetermined, see Question (2). Further, the interpretability of the resulting function cannot be guaranteed when using sophisticated functions g, violating Question (1). Multi-level checks like "If the outcome of the first rule is this, then check this in the next round, if the outcome is that, then check something else" would require a change of the system's operating mode, which we want to avoid if possible. As weights are fixed for their respective rules, the information about dependencies between existing rules has to be added in form of new rules. Thereby, it is a straightforward way to establish rule combinations, i.e., two (or more) rules each are combined to new rules with an "and-link"(conjunction). The definition of a binary combination is:

Definition 2 (binary rule combinations)**.** For a given rule vector $\mathfrak{r}$ and claims $c \in \mathcal{C}$ we define a binary rule combination as $\tilde{r}_{i,j} : \mathcal{C} \to \{0,1\}$, $0 < i < j \leq k$, with $\tilde{r}_{i,j}(c) = r_i(c) \cdot r_j(c)$. The extended rule vector that contains all possible binary combinations is $\tilde{\mathfrak{r}} = (r_1, \ldots, r_k, \tilde{r}_{1,2}, \ldots, \tilde{r}_{1,k}, \tilde{r}_{2,3}, \ldots, \ldots, \tilde{r}_{k-1,k})$.

With the definition above it is analogously possible to generate ternary and higher degree rule combinations. However, due to complexity and computational issues, we stick to binary combinations for the moment. When considering a rule vector $\mathfrak{r}$ of k rules, adding every possible binary combination would lead to additional $\binom{k}{2} = \frac{k(k-1)}{2}$ rules, of which many would probably not be meaningful. The identification of the meaningful ones is exactly the problem stated in Question (1). For finding them, we use the CARM explained next.

3.2 Classification Based on Association Rule Mining

Association rule mining is a very common technique in data mining, mainly originating from market basket analysis where predications like "When a customer buys products A and B, it is likely (with a probability of x %) that the customer also buys product C" are to be found [HGN00]. Such rules are derived from past purchases of all customers and then used to propose active customers further products based on their previous purchases or their current shopping cart. The single products are called *items* and the previous purchases are called *transactions*. Two major problems occur when applying association rule mining: first, the number of all possible rules grows exponentially with the number of items, and second, from the set of all mined rules the interesting ones have to be selected [HGN00]. Various algorithms exist to cope with these problems [ZZ02].

In more detail, the formal definition of association rule mining according to [AIS93] is the following: Let $\mathcal{I} = \{i_1, \ldots, i_n\}$ be the set of all items. A subset of $\mathcal{I}$ is called *itemset*. The set of all transactions is $\mathcal{T} = \{t_1, \ldots, t_m\} = \{(tid_1, A_1), \ldots, (tid_m, A_m)\}$ with tid_i being transaction identifiers and $A_i \subseteq \mathcal{I}$ being itemsets, $i = 1, \ldots, m$. A *rule* is an implication $X \Rightarrow Y$ with $X, Y \subseteq \mathcal{I}$. Usually, only rules with $X \cap Y = \emptyset$ are of interest. To limit the number of rules to the interesting ones there exist two basic concepts. The *support* of a rule $X \Rightarrow Y$ is given through

$$\text{supp}(X \Rightarrow Y) := \text{supp}(X \cup Y) := \frac{|\{t \in \mathcal{T} | X \cup Y \subseteq t\}|}{|\mathcal{T}|} =: \frac{t_{X,Y}}{|\mathcal{T}|}$$

where $X \cup Y \subseteq t$ if $t = (tid, A)$ and $X \cup Y \subseteq A$. Interesting rules need to have a minimum support, i.e., their left and right hand sides need to appear together often enough. Rules that do not have enough support are considered irrelevant and not generalizable. The second concept is that of *confidence* and measures the reliability of a rule, i.e., how often the rule holds given the left hand side is true:

$$\mathrm{conf}(X \Rightarrow Y) := \frac{\mathrm{supp}(X \cup Y)}{\mathrm{supp}(X)} = \frac{|\{t \in \mathcal{T} | X \cup Y \subseteq t\}|}{|\{t \in \mathcal{T} | X \subseteq t\}|} =: \frac{t_{X,Y}}{t_X}$$

Interesting rules need to have a minimum confidence, too. There exist further measures for filtering out interesting rules, like interest or lift [Br97, HGH05]. The integration of classification and association rule mining is an approach to transfer the actually unsupervised data mining method of finding arbitrary association rules to a supervised method for finding rules for a classification [LHM98, CLZ05]. In the classification setting, we have labeled data where the class attribute together with its values is a subset of the set of items and in all transactions there is the class attribute together with exactly one of its values. During the rule generation process only a subset of all association rules is selected, namely those rules with only the class attribute on the right-hand side, i.e., the rules are of the form $X \Rightarrow \{\text{label}\}$ where "label" denotes the class attribute together with one of its values (label $\in \mathcal{I}$). This set of rules is referred to as class association rules (CARs) [LHM98]. The support and confidence are considered here as well. The generation of the classifier itself is not relevant for this paper as we are interested solely in the rules.

3.3 Extending the Rule-based Fraud Detection System

For extending the rule-based expert system $(\mathfrak{r}, \mathfrak{w})$ with binary rule combinations according to Definition 2, we define $\mathcal{I} = \mathcal{I}_{rules} \cup \mathcal{I}_{label}$ with $\mathcal{I}_{rules} = \{r_1 = 0, r_1 = 1, \ldots, r_k = 1\}$, $\mathcal{I}_{label} = \{\text{fraud} = 0, \text{fraud} = 1\}$, fix a support level and a confidence level, and take only those rules into account that fulfill both the confidence and the support level. Note that one claim is one transaction t. We further narrow the CARs to those with only "$= 1$" rule items and with at most two rule items, both on the left-hand side of the rules (the latter because of runtime issues). For instance, we have a set containing the following CARs:

- {There are exactly two cars involved = 1, There is exactly one witness = 1} ⇒ {fraud = 1}
- {The same insured reported another claim in the six months before = 1, There is exactly one witness = 1} ⇒ {fraud = 1}
- {The same insured reported another claim in the six months before = 1} ⇒ {fraud = 1}
- {The accident happened during the weekend = 1, There is exactly one witness = 1} ⇒ {fraud = 0}

Additionally, we delete those rules of the binary ones whose left-hand side is a superset of the left hand side of a CAR with the same consequent to further reduce the number of rules and to concentrate on the meaningful ones and prevent overfitting (superset criterion). In the above example, the second association rule would be deleted according to this criterion. The superset criterion is a heuristic approach proposed for pruning rule generation trees [CLZ05]. In our case, we can motivate it as follows: Assume two association rules $X_1 \Rightarrow Y$ and $X_2 \Rightarrow Y$ with $X_1 \subset X_2$, in our case $|X_1| = 1$ and $|X_2| = 2$ when regarding only unary and binary association rules, then it clearly holds that $\mathrm{supp}(X_1 \Rightarrow Y) \geq \mathrm{supp}(X_2 \Rightarrow Y)$, i.e., the more general rule $X_1 \Rightarrow Y$ has a higher support. For the confidence there is no such inequality, however, when doing classification and the rule $X_1 \Rightarrow Y$ fulfills the minimum confidence requirement, i.e., from X_1 we can conclude the target, then the (single) rule on the left hand side is a strong indicator for the respective target value (fraud $= 0$ or fraud $= 1$). In the context of fraud detection this is usually only possible for very specific rules (provided that the minimum confidence is relatively high) with a very small support where a more specialized rule would not be meaningful. This is not a contradiction to the motivation of our approach of finding correlating fraud conditions that provide additional information to better classify the claims.

After having extended the rule vector $\mathfrak{r}$ to $\bar{\mathfrak{r}}$ (with all initial expert rules and only meaningful combinations), we have to extend and adjust the weight vector $\mathfrak{w}$. The new weight vector $\bar{\mathfrak{w}}$ contains the weights for the old rules as well as the weights for the rule combinations. The alert threshold $l > 0$ may be maintained. For the set of all claims, the weights in $\bar{\mathfrak{w}}$ need to be optimal related to a fitness criterion so that only suspicious claims have an aggregated value above l. This is, of course, unlikely to be feasible, but the separation of suspicious and non-suspicious claims should be as good as possible while avoiding overfitting. One way of how to assign optimal weights is the usage of cultural algorithms [SR97], which are a specific kind of evolutionary algorithms. For weighting attributes for a k-nearest neighbors classification a similar, evolutionary optimization approach can be applied [KJD91]. Evolutionary algorithms are especially useful for large-scale optimization problems [EHG05]. The fitness criterion resp. objective function has to make sure that the number of correctly classified claims is maximal, e.g., it can be maximized using accuracy measures like the numbers of true positives and true negatives. Regarding the field of insurance fraud, it is desirable to have no false negatives, i.e., that all fraud attempts are discovered. At the same time, false positives cause huge costs for an insurer as the manual inspection of claims is time-consuming and therefore expensive. Depending on the rule base and its extent, it may be impossible to detect all attempts of fraud in the claim data. Hence, it is of special interest to reduce the false positives when extending an existing rule-based detection system while not worsening the false negatives. Such a focus can also be considered in the objective function.

4 Evaluating the Approach on a Real-World Data Set

For evaluating the improvement approach we consider a real-world data set of motor insurance claims from a German insurer. The data set consists of about 140,000 claims covering several years. An already existing rule-based expert system approximately working as stated in Definition 1 consists of about 110 unary, weighted rules. The data has been manually labeled during the claims' regular processing. However, this label only marks a certain suspiciousness, not a fraud. The data is highly unbalanced, which is very typical for fraud data as most of the claims an insurer gets are justified ones. Less than 5 % of the cases are marked as positive (suspicious). Thus, before conducting the algorithm for a CARM, we have to correct the imbalance in the data.

A first idea was to add further positive cases which are older than the original time horizon. Although this proved very promising, we omitted this possibility because of several problems. Some of the rules are not applicable to the old data, such as the rule "An electric car is involved" and would always evaluate to 0 in the old data leading to biased results. Furthermore, also the enriched data set would still have shown a great imbalance. Instead, we checked for several common sampling methods. When performing case generating algorithms like SMOTE [Ch02] or ROSE [MT14], these algorithms may actually deteriorate the results for extreme imbalances [Kr16]. When we performed such case generating algorithms we detected that technically impossible cases (like one contract is of two different types at the same time) were created. We also tried non-random undersampling methods like the OSS algorithm [KM97], which uses so-called Tomek links [To76] to remove redundant rows from the data. But the different non-random undersampling methods we tried only removed fewer than 1 % of the majority class cases and were, thus, not useful. This is why we decided to apply a combination of random over- and undersampling [LMT14] with a target proportion of 45:55 (minority class:majority class), which proved to be suitable after having tried different proportions in an extensive study, although the data set is likely to get biased [Kr16].

After having adjusted the class sizes, we perform the CARM using the R-package "arulesCBA" [JHG20]. We set the confidence to 90 % and the support to 0.2 % to get enough classification rules for the relatively high confidence value. We explicitly mined rules with one and two attributes on the left-hand side and deleted the binary ones fulfilling the superset criterion. This resulted in a total of 54 new rules, where 35 were positive rules, i.e. rules with a positive consequent (suspicious), and 19 negative rules (not suspicious). Considering the number of rows affected by the additional rules, we see that about 10.99 % of the claims fulfill at least one of the additional rules.

To compare the old and the extended rule base, we had to add weights for the new rules and analyze the performance of the two systems. The weights were determined using a genetic algorithm. To create equal conditions, we also adjusted the weights of the original rule-based system with the same genetic algorithm so that we can state that differences in the accuracy really stem from the new binary rules and not from a non-optimal weight assignment in the original system. For the optimization, we use the R-package "GA" [Sc19] with the original weights as one of the initial populations. For the weights, we allow values between b_{min} and b_{max} with $0 > b_{min} \in \mathbb{Z}_-$ and $0 < b_{max} \in \mathbb{Z}_+$. Since the optimization algorithm works on real values and not on integers, we take the floor of the generated weights [Sc]. As objective function to be maximized we chose $tpr^{w_{tpr}} * tnr^{w_{tnr}}$ where tpr is the rate of the true positives and tnr the rate of the true negatives. In order to place a higher importance on the true negatives resp. false positives, we set the weight for the true negatives to $w_{tnr} = 1 - w_{tpr} = 0.75$. Under this setting, we get the following results, that are summarized in Table 1: For the original rule set, the false positive rate is 12.85 % and the false negative rate is 57.78 %. The extended rule set gives a false positive rate of 10.68 % and a false negative rate of 58.27 %. This is a decrease of the false positive rate of about

16.89 % and an increase of the false negative rate of about 0.85 %. The false negative rate seems to be quite high, but considering how the rules are constructed and on what data they rely on, it was clear from the beginning that the rules are not able to catch all the fraudulent behavior. The main focus was on reducing the false positives (without increasing the false negatives, which we could not exactly achieve in our experiment) to save inspection costs and time and to prevent false suspicions. This is extremely important as the absolute number of false positives is times higher than the absolute number of false negatives. Considering this, the decrease in the false positive rate can be regarded as a success.

rule set	false positive rate	false negative rate
$\mathfrak{r}$	12.85 %	57.78 %
$\bar{\mathfrak{r}}$	10.68 %	58.27 %

Tab. 1: Error rates for the original and the extended rule-based fraud detection system. The false positive error rate decreased by 16.89 % while the false negative rate increased by 0.85 %

As by construction, a genetic algorithm does not necessarily lead to the global optimum, we wanted to run the algorithm with different seeds and average the results, but this was prohibited by runtime issues in our experimental setting. However, it is particularly interesting that 82.86 % of the rules associated with a positive suspicion got on average a positive weight. The remaining 17.14 % got a weight of 0, so perhaps they have not been tested by the algorithm at all. 52.64 % of the rules associated with a negative suspicion got a negative weight, 36.84 % got a weight of 0, and 10.52 % a positive weight, but near 0. These results are summarized in Tab. 2. Thus, the rule weights and the right-hand side of the association rules are consistent for almost all cases, which confirms our approach.

target	positive weight	zero weight	negative weight
fraud = 1	82.86 %	17.14 %	0.00 %
fraud = 0	10.52 %	36.84 %	52.64 %

Tab. 2: Target value and sign of the weights. While none of the "fraud = 1" rule weights are not in line with the implication, there are 10.52 % of the "fraud = 0" rule weights not in line with the implication

5 Conclusions

This paper shows a method of how an existing rule-based expert system for fraud detection may be improved with relatively little construction effort. It uses the concept of association rule mining for classification to add meaningful new rules to the set of already existing rules. These new rules reflect dependencies between the old rules, which allows a more precise scoring of insurance claims. This approach is conducted on a real world example, for which the extended rule-based system yields satisfying results. Above all, the distinction between rules intended to strengthen the suspiciousness and those to weaken the suspiciousness could be observed after assigning new rule weights with help of a genetic optimizer. A faster method for the weight assignment would improve the feasibility of the approach. For its evaluation, the rules resulting from the association rule mining algorithm have been taken over unmodified. Before, however, applying such an extension on a productive system, it is recommended that the fraud expert has a closer look at the automatically generated rules and at the assigned weights to estimate their meaningfulness, explainability, and quality. A particular focus should lie on the interplay of the original rules, the combinations and how the combinations increase or decrease the suspiciousness of the original rules.

Acknowledgement

The opinions expressed in this report are those of the author and not necessarily those of the author's employer. The author's employer does not guarantee the accuracy or reliability of the information provided herein. The author works as a Data Scientist at NÜRNBERGER Lebensversicherung AG, Nuremberg, Germany. The author thanks Michael H. Baumann (Universität Bayreuth) for valuable discussions and the referees for their helpful comments.

Bibliography

[AAG99] Artís, Manuel; Ayuso, Mercedes; Guillén, Montserrat: Modelling different types of automobile insurance fraud behaviour in the Spanish market. Insurance: Mathematics and Economics, 24(1):67–81, 1999.

[AIS93] Agrawal, Rakesh; Imieliundefinedski, Tomasz; Swami, Arun: Mining Association Rules between Sets of Items in Large Databases. SIGMOD Rec., 22(2):207–216, June 1993.

[AMZ16] Abdallah, Aisha; Maarof, Mohd Aizaini; Zainal, Anazida: Fraud detection system: A survey. Journal of Network and Computer Applications, 68:90–113, 2016.

[BH01] Bolton, Richard J; Hand, David J: Unsupervised profiling methods for fraud detection. Credit scoring and credit control VII, pp. 235–255, 2001.

[BH02] Bolton, Richard J.; Hand, David J.: Statistical Fraud Detection: A Review. Statistical Science, 17(3):235–249, 2002.

[Br97] Brin, Sergey; Motwani, Rajeev; Ullman, Jeffrey D.; Tsur, Shalom: Dynamic Itemset Counting and Implication Rules for Market Basket Data. SIGMOD Rec., 26(2):255–264, 1997.

[BST01] Burge, Peter; Shawe-Taylor, John: An Unsupervised Neural Network Approach to Profiling the Behavior of Mobile Phone Users for Use in Fraud Detection. Journal of Parallel and Distributed Computing, 61(7):915–925, 2001.

[Ch02] Chawla, Nitesh V; Bowyer, Kevin W; Hall, Lawrence O; Kegelmeyer, W Philip: SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16:321–357, 2002.

[Ch11] Chiu, Chaochang; Ku, Yungchang; Lie, Ting; Chen, Yuchi: Internet Auction Fraud Detection Using Social Network Analysis and Classification Tree Approaches. International Journal of Electronic Commerce, 15(3):123–147, 2011.

[CLZ05] Coenen, Frans; Leng, Paul; Zhang, Lu: Threshold Tuning for Improved Classification Association Rule Mining. In: Advances in Knowledge Discovery and Data Mining. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 216–225, 2005.

[DB00] Dembe, Allard E.; Boden, Leslie I.: Moral Hazard: A Question of Morality? NEW SOLUTIONS: A Journal of Environmental and Occupational Health Policy, 10(3):257–279, 2000.

[De02] Derrig, Richard A.: Insurance Fraud. The Journal of Risk and Insurance, 69(3):271–287, 2002.

[DZ02] Derrig, Richard A; Zicko, Valerie: Prosecuting insurance fraud – A case study of the massachusetts experience in the 1990s. Risk Management and Insurance Review, 5(2):77–104, 2002.

[ED03] Ericson, Richard V.; Doyle, Aaron: The Moral Risks of Private Justice: The Case of Insurance Fraud. In: Risk and Morality, chapter 12, pp. 317–363. University of Toronto Press, 2003.

[EHG05] Elbeltagi, Emad; Hegazy, Tarek; Grierson, Donald: Comparison among five evolutionary-based optimization algorithms. Advanced engineering informatics, 19(1):43–53, 2005.

[Fa11] Farny, Dieter: Versicherungsbetriebslehre. VVW GmbH, 2011.

[FCR17] Fronzetti Colladon, Andrea; Remondi, Elisa: Using social network analysis to prevent money laundering. Expert Systems with Applications, 67:49–58, 2017.

[FCS95] Fanning, Kurt; Cogger, Kenneth O.; Srivastava, Rajendra: Detection of Management Fraud: A Neural Network Approach. Intelligent Systems in Accounting, Finance and Management, 4(2):113–126, 1995.

[FM06] Ferdousi, Zakia; Maeda, Akira: Unsupervised Outlier Detection in Time Series data. In: 22nd International Conference on Data Engineering Workshops (ICDEW'06). 2006.

[Fr18] Friedrich, Sara: , Versicherungsbetrug – Du lügst! `https://www.gdv.de/de/themen/positionen-magazin/du-luegst--39848`, 2018. Accessed: 2020-04-15.

[FVB11] Furlan, Štefan; Vasilecas, Olegas; Bajec, Marko: Method for selection of motor insurance fraud management system components based on business performance. Technological and Economic Development of Economy, 17(3):535–561, 2011.

[GC97] Green, Brian Patrick; Choi, Jae Hwa: Assessing the risk of management fraud through neural network technology. Auditing, 16:14–28, 1997.

[GL91] Garvey, Thomas D; Lunt, Teresa F: Model-based Intrusion Detection. In: Proceedings of the 14th national computer security conference. volume 10, pp. 372–385, 1991.

[He89] Heimer, Carol A: Reactive Risk and Rational Action: Managing Moral Hazard in Insurance Contracts. University of California Press, 1989.

[HGH05] Hornik, Kurt; Grün, Bettina; Hahsler, Michael: arules – A computational environment for mining association rules and frequent item sets. Journal of Statistical Software, 14(15):1–25, 2005.

[HGN00] Hipp, Jochen; Güntzer, Ulrich; Nakhaeizadeh, Gholamreza: Algorithms for association rule mining—a general survey and comparison. ACM sigkdd explorations newsletter, 2(1):58–64, 2000.

[Hi09] Hilas, Constantinos S.: Designing an expert system for fraud detection in private telecommunications networks. Expert Systems with Applications, 36(9):11559–11569, 2009.

[Hö79] Hölmstrom, Bengt: Moral Hazard and Observability. The Bell Journal of Economics, 10(1):74–91, 1979.

[iHG] informa HIS GmbH: , Hinweis- und Informationssystem der Versicherungswirtschaft. `https://www.informa-his.de/his-online`. Accessed: 2020-05-19.

[JHG20] Johnson, Ian; Hahsler, Michael; Giallanza, Tyler: . arulesCBA: Classification Based on Association Rules, 2020. R package version 1.1.6.

[KGK15] Kose, Ilker; Gokturk, Mehmet; Kilic, Kemal: An interactive machine-learning-based electronic fraud and abuse detection system in healthcare insurance. Applied Soft Computing, 36:283–299, 2015.

[KJD91] Kelly Jr, James D; Davis, Lawrence: A Hybrid Genetic Algorithm for Classification. In: IJCAI. volume 91, pp. 645–650, 1991.

[KM97] Kubat, Miroslav; Matwin, Stan: Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. In: Icml. volume 97. Nashville, USA, pp. 179–186, 1997.

[Ko04] Kou, Yufeng; Lu, Chang-Tien; Sirwongwattana, S; Huang, Yo-Ping: Survey of fraud detection techniques. In: IEEE International Conference on Networking, Sensing and Control, 2004. volume 2, pp. 749–754, 2004.

[Ko13] Koch, Peter: Versicherungswirtschaft: Ein einführender Überblick. VVW GmbH, 2013.

[Kö15] Köneke, Vanessa; Müller-Peters, Horst; Fetchenhauer, Detlef et al.: Versicherungsbetrug verstehen und verhindern. Springer, 2015.

[Kr16] Krawczyk, Bartosz: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence, 5(4):221–232, 2016.

[Le95] Leonard, Kevin J.: The development of a rule based expert system model for fraud alert in consumer credit. European Journal of Operational Research, 80(2):350–356, 1995.

[LHM98] Liu, Bing; Hsu, Wynne; Ma, Yiming: Integrating classification and association rule mining. In: KDD. volume 98, pp. 80–86, 1998.

[LMT14] Lunardon, Nicola; Menardi, Giovanna; Torelli, Nicola: . ROSE: Random Over-Sampling Examples, 2014. R package version 0.0-3.

[MT14] Menardi, Giovanna; Torelli, Nicola: Training and assessing classification rules with imbalanced data. Data Mining and Knowledge Discovery, 28(1):92–122, 2014.

[Ni16] Nian, Ke; Zhang, Haofan; Tayal, Aditya; Coleman, Thomas; Li, Yuying: Auto insurance fraud detection using unsupervised spectral ranking for anomaly. The Journal of Finance and Data Science, 2(1):58–75, 2016.

[Ok13] Okura, Mahito: The relationship between moral hazard and insurance fraud. Journal of Risk Finance, 14:120–128, 2013.

[SA13] Srinivasan, Uma; Arunasalam, Bavani: Leveraging Big Data Analytics to Reduce Healthcare Costs. IT Professional, 15(6):21–28, 2013.

[Sc] Scrucca, Luca: , A quick tour of GA. `https://cran.r-project.org/web/packages/GA/vignettes/GA.html`. Accessed: 2020-06-14.

[Sc04] Schiller, Jörg: Versicherungsbetrug als ökonomisches Problem: Eine vertragstheoretische Analyse. Zeitschrift für die gesamte Versicherungswirtschaft, 93:835–851, 2004.

[Sc19] Scrucca, Luca: . GA: Genetic Algorithms, 2019. R package version 3.2.

[ŞD11] Şahin, Yusuf G; Duman, Ekrem: Detecting credit card fraud by decision trees and support vector machines. In: Proceedings of the International MultiConference of Engineers and Computer Scientists IMECS. volume 1, p. 6, 2011.

[ŠFB11] Šubelj, Lovro; Furlan, Štefan; Bajec, Marko: An expert system for detecting automobile insurance fraud using social network analysis. Expert Systems with Applications, 38(1):1039–1052, 2011.

[Sh92] Shavell, Steven: On Moral Hazard and Insurance. In: Foundations of Insurance Economics: Readings in Economics and Finance. Springer Netherlands, Dordrecht, pp. 280–301, 1992.

[SR97] Sternberg, M.; Reynolds, R. G.: Using cultural algorithms to support re-engineering of rule-based expert systems in dynamic performance environments: a case study in fraud detection. IEEE Transactions on Evolutionary Computation, 1(4):225–243, 1997.

[Ta98] Taniguchi, M.; Haft, M.; Hollmen, J.; Tresp, V.: Fraud detection in communication networks using neural and probabilistic methods. In: Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP ’98 (Cat. No.98CH36181). volume 2, pp. 1241–1244, 1998.

[To76] Tomek, Ivan: Two Modifications of CNN. IEEE Transactions on Systems, Man and Communications, 6:769–772, 1976.

[Va15] Van Vlasselaer, Véronique; Bravo, Cristián; Caelen, Olivier; Eliassi-Rad, Tina; Akoglu, Leman; Snoeck, Monique; Baesens, Bart: APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions. Decision Support Systems, 75:38–48, 2015.

[VD04] Viaene, S.; Dedene, G.: Insurance Fraud: Issues and Challenges. The Geneva Papers on Risk and Insurance, 29(2):313–333, 2004.

[Vi02] Viaene, Stijn; Derrig, Richard A.; Baesens, Bart; Dedene, Guido: A Comparison of State-of-the-Art Classification Techniques for Expert Automobile Insurance Claim Fraud Detection. Journal of Risk and Insurance, 69(3):373–421, 2002.

[WX18] Wang, Yibo; Xu, Wei: Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud. Decision Support Systems, 105:87–95, 2018.

[ZZ02] Zhang, Chengqi; Zhang, Shichao: Association rule mining: models and algorithms. Springer-Verlag, 2002.