Fraud Detection in Health Insurance Using Ensemble Learning Methods

Rimantė Kunickaitė[a], Monika Zdanavičiūtė[a,b] and Tomas Krilavičius[a,c]

[a]Baltic Institute of Advanced Technology, Vilnius, Lithuania
[b]Department of Mathematics and Statistics, Vytautas Magnus University Kaunas, Lithuania
[c]Department of Applied Informatics, Vytautas Magnus University Kaunas, Lithuania

Abstract
Insurance fraud is one of the most expensive economic financial crimes. Most risk management solutions use rules to detect potential abuse, but as the patterns of abuse change, those solutions become ineffective. In this paper we apply machine learning (Decision Trees, Bagging, Random Forests and Boosting) for fraud detection in health insurance. Performance of the model is evaluated using accuracy, error rate, sensitivity and specificity. The best results were achieved using Bagging technique. In further research it would be useful to analyze applicability of deep learning models and anomaly detection methods.

Keywords
insurance, fraud, ensemble learning, classification.

1. Introduction

Insurance fraud is one of the most expensive economic financial crimes [1]. In order to conquer a larger share of the insurance market, insurance companies offer beneficial insurance terms and, as a result, in such a way providing new opportunities for fraud as well. The number of crimes in the insurance sector is increasing every year. As a result, service prices (premiums) increase because insurance is based on the principle of solidarity, hence the loss is distributed to all participants in the insurance relationship and, as the loss increases, the contribution of each participant to cover them increases.

Identification of insurance fraud is rather complicated. Such type of fraud can be practiced by very diverse people, i.e. independent of education and professions. Most risk management solutions use rules to detect potential abuse, and some solutions seem to learn from the examples, but as the patterns of abuse change, those solutions stop work for novel types of fraud. Four models are provided for identifying potential health insurance abuse using AI, mostly, looking for an anomalous behavior. Which are not filtered by VMD or radar tracker methods respectively.

IVUS 2020: Information Society and University Studies, 23 April 2020, KTU Santaka Valley, Kaunas, Lithuania
rimante.kunickaite@bpti.lt (R. Kunickaitė);
monika.zdanaviciute@bpti.lt (M. Zdanavičiūtė);
tomas.krilavicius@bpti.eu (T. Krilavičius)

2. Literature Review

The health insurance fraud claims are broadly classified into several classes. In [2] a solution for duplicated claims fraud is proposed, namely, detection of cases, when people are submitting just slightly different bills repeatedly, changing some small portion like the date, in order to charge insurance company twice for the same service rendered. Example: An exact copy of the original claim is not filed for the second time, but rather some portion like date is changed to get the benefit twice the original. In this approach, first, the insurance claims are clustered according to the disease type using Evolving Clustering Method and then they are classified to detect duplicate claims using Support Vector Machine.

Detection of fraudulent health insurance claims by identifying correlation or association between some of the attributes on the claim documents is analyzed in [3]. Unsupervised learning based clustering were used to group health insurance claims, and then unsupervised association to identify the correlation between attributes, and afterwards classifiers to identify fraudulent claims.

Anomaly detection is studied in [4], where statistical decision rules and k-means clustering were applied for historical claim data, using outliers detection and association rule-based mining with Gaussian distribution. Such outliers often correspond to fraud insurance claims in the data.

Multilayer perceptron neural network (MLP) model and fraud diamond theory (FDT)'s fraud elements as fraud indicators, were proposed in [5], where a fraud

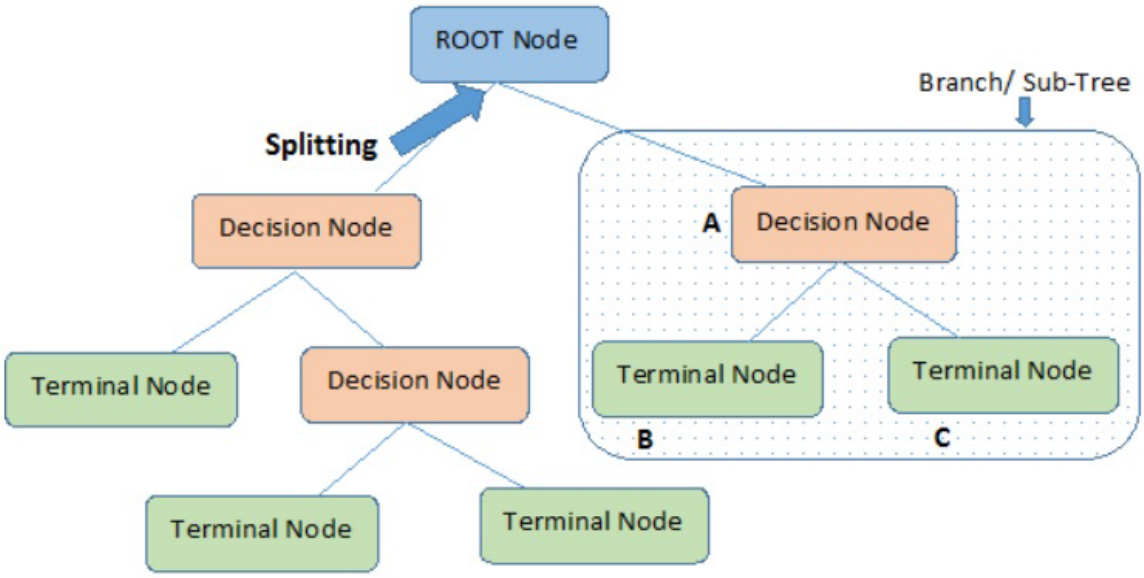

Figure 1: Decision Tree structure [6]

prediction model was developed to check whether a claim presented by a customer is fraudulent or non-fraudulent.

3. Methods

3.1. Decision Tree

Decision tree is a classifier, which is based on the idea of identifying data set division points (branching points) [6]. The structure of the decision tree is illustrated in Fig 1.

Concepts describing the decision tree [6]:

1. **Root Node** is a starting point i a decision tree.
2. **Splitting** is a process of dividing a node into two or more sub-nodes.
3. **Decision Node** is a sub-node which splits into further sub-nodes.
4. **Terminal Node** or a **Leaf** is a node that does not split and specifies the output result.
5. **Pruning** is a process of removing sub-nodes from a decision node.The opposite of pruning is splitting.
6. **Branch** is a sub-section of the entire decision tree.
7. **Parent Node** of the sub-nodes is a node, which is divided into sub-nodes.

Recursive binary splitting is applied to grow a classification tree. **Gini Index** can be used as a criterion for making the binary splits. For a feature space of size p, a subset of $\mathbb{R}^p$, the space is divided into M regions R_m, if a region R_m includes data that is mostly from a single class c then the **Gini Index** value will be small:

$$G = \sum_{c=1}^{C} \widehat{\pi}_{mc} \left(1 - \widehat{\pi}_{mc}\right) , \tag{1}$$

where $\widehat{\pi}_{mc}$ represents the fraction of training data in region R_m that belong to class c.

3.2. Ensemble Learning

Ensemble learning methods are based on the hypothesis that combining multiple models together can often produce a much more powerful model [7, 8, 9]. Then, the idea of ensemble methods is to try reducing bias and/or variance of such weak learners by combining several of them together in order to create a strong learner (or ensemble model) that achieves better performances.

3.2.1. Bagging

One of the most popular parallel methods is **Bagging** (Fig. 2) that goal at producing an ensemble model that is more powerful than individual models composing it [8].

Bootstrapping is statistical approach consists in generating sample of size B called bootstrap sample from

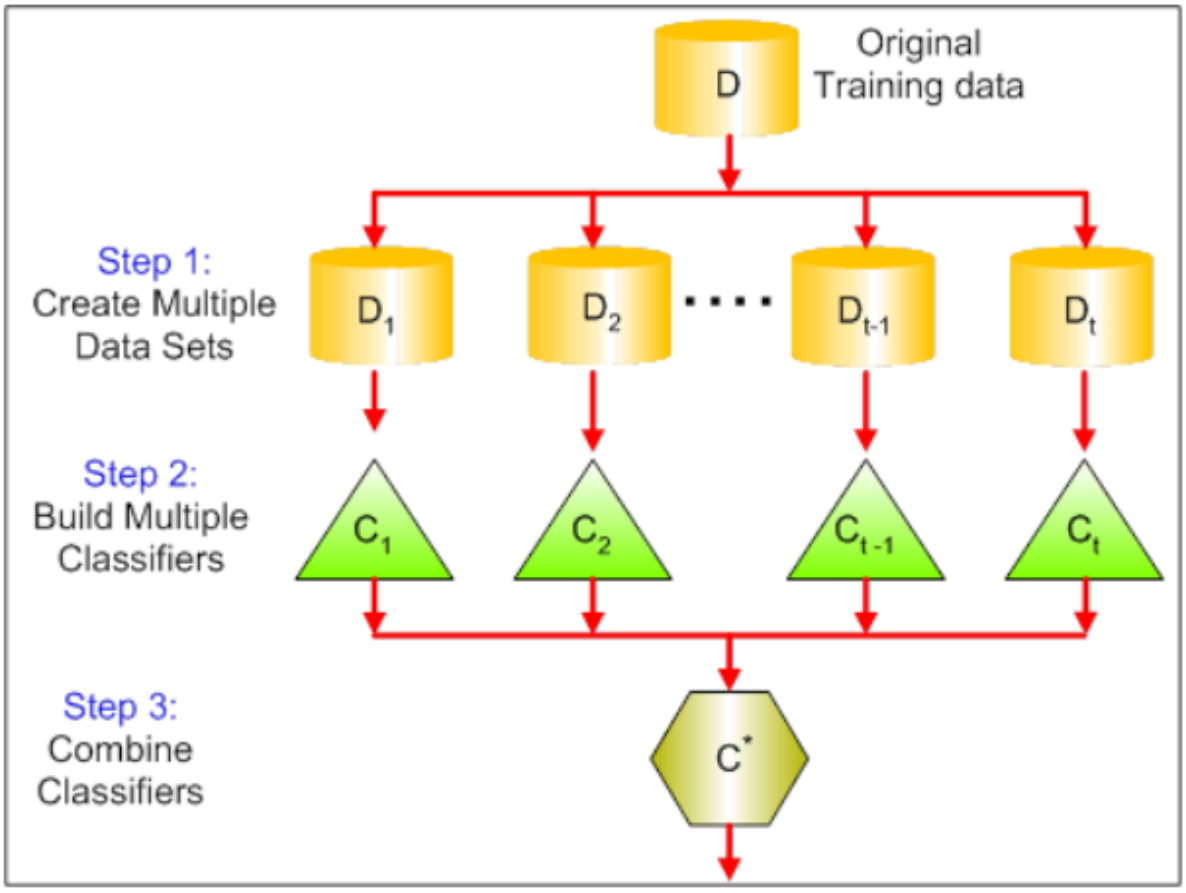

Figure 2: Steps of Bagging [6]

initial data set of size N by randomly taking with replacement B observations [8]. Bootstrap samples can be considered as representative and independent samples of the true data distribution [8].

Assuming that we have L bootstrap samples of size B denoted [8]:

$$\{z_1^1, z_2^1, \dots, z_B^1\},\ \{z_1^2, z_2^2, \dots, z_B^2\}, \dots \\ \dots, \{z_1^L, z_2^L, \dots, z_B^L\}, \tag{2}$$

where z_b^l is $b-th$ observation of the $l-th$ bootstrap sample, we can fit L almost independent weak learners [8]::

$$w_1(\cdot), w_2(\cdot), \dots, w_L(\cdot) \tag{3}$$

and then combine them into averaging process in order to get an ensemble model with a lower variance. Simple majority vote for classification problem [8]:

$$s_L(\cdot) = \arg\max_k \left[card(l | w_l(\cdot) = k) \right]. \tag{4}$$

3.2.2. Random Forest

Random Forest is a machine learning algorithm that creates groups of decision trees during the learning process [6]. The basic idea of random forest is that the classifier is formed by combining many binary decision trees constructed using different subsets of data from the original data set and randomly selected subsets of attributes. This is the main difference between random forests and bagging. The structure of the random forest is illustrated in Fig 3.

3.2.3. Boosting

Boosting is sequential method that based on fitting sequentially multiple weak learners in an adaptive manner: each model in the sequence is fitted giving more

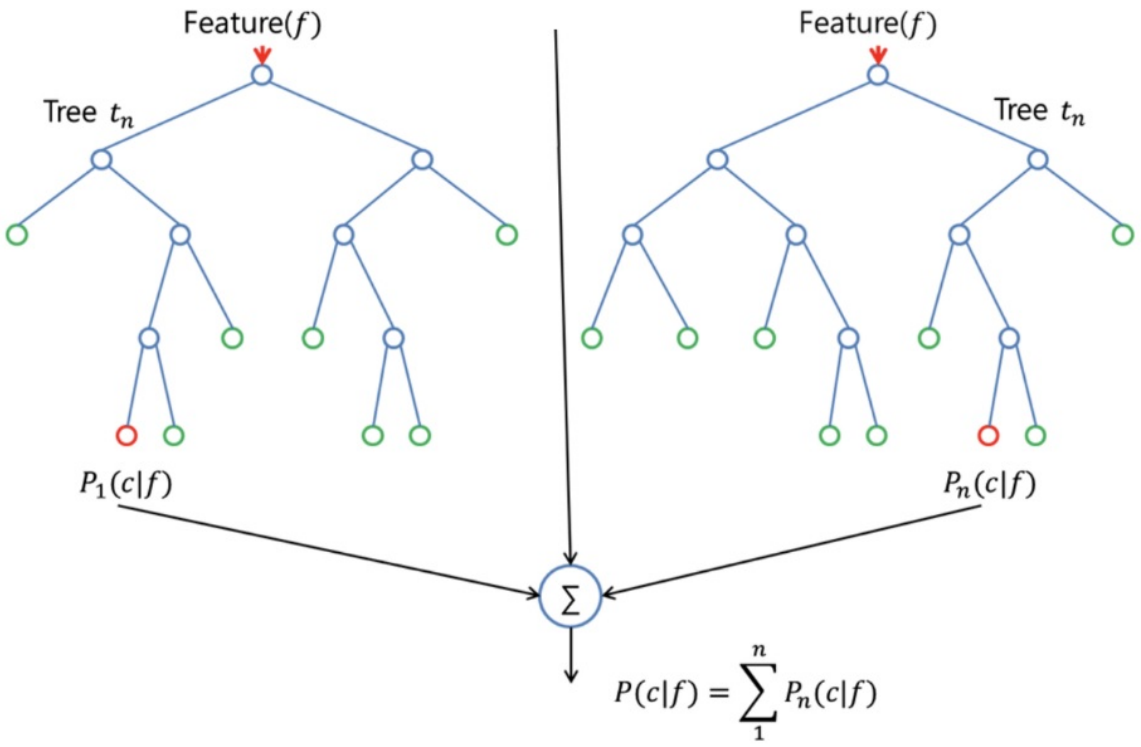

Figure 3: Random Forest structure [6]

effect to observations in the data set that were poorly handled by the previous models in the sequence [8]. Adaptive boosting updates the weights attached to each of the training data set observations.

Ensemble model is defined as a weighted sum of L weak learners [8]:

$$s_L(\cdot) = \sum_{l=1}^{L} c_l \times w_l(\cdot)\,, \tag{5}$$

where c_l's are coefficients and w_l's are weak learners.

Solving optimisation problem we define recurrently the (s_l)'s [8]:

$$s_l(\cdot) = s_{l-1}(\cdot) + c_l \times w_l(\cdot)\,, \tag{6}$$

where c_l and w_l are chosen that s_l is the model that fits the best the training data and that is the best possible improvement over $s_{(l-1)}$. Then we denote [8]:

$$\begin{aligned}(c_l,\ w_l(\cdot)) &= \arg\min_{c,w(\cdot)} E(s_{l-1}(\cdot) + c \times w(\cdot)) = \\ &= \arg\min_{c,w(\cdot)} \sum_{n=1}^{N} e(y_n,\ s_{l-1}(x_n) + c \times w(x_n))\,,\end{aligned} \tag{7}$$

where $E(\cdot)$ is the fitting error of the given model and $e(\cdot\,,\,\cdot)$ is the loss/error function. Thus, instead of optimising "globally" over all the L models, we approximate the optimum by optimising "locally" building and adding the weak learners to the strong model one by one.

4. Dataset

The data contains three part: the insurance policy, the claim application and the details of the risk assessment. Whether the claim for reimbursement is satisfied according to the terms and conditions of the insurance contract is specified by the specific binary variable. Payment Claim Period 01/01/2018 - 23/12/2019, total number of entries is **2662308**.

It was decided not to analyze the medical records in this stage of research, which eliminated all cases where the document type was an invoice for the medication purchased. Records that did not specify a risk type or service were also removed.

Following the expert assessment, 27 the most important variables were selected from the data set and

were used for further data analysis:

1. *SERV_INST_CITY* - service institution city,
2. *INSURER_TYPE* – insurer status (individual/juridical),
3. *SELL_UNIT_CODE* – product code,
4. *INS_PERS_GNDR* - insured person gender,
5. *INS_PERS_CNTR* - insured person center,
6. *REC_TYPE* – receiver status (individual/juridical),
7. *REC_CNTR* - receiver country,
8. *TYPE* – type of medical document,
9. *AUCH_CH_ID* – insurer type ID,
10. *AUCH_CH_CODE* – insurer type code,
11. *DOC_CODE* – type of document (original/scanned),
12. *ODE_IMP_REC* – online data exchange feature,
13. *SERV_INST_PRICE* – price from service institution price list,
14. *MED_INST_PRICE* – price from medical institution price list,
15. *REAL_SUM* – amount of paid money,
16. *DISCOUNT* – amount of discount,
17. *RCPT_SUM* – amount of money stated in the document,
18. *EV_COUNT* – quantity of services,
19. *RISK_LIMIT* – maximum sum insured,
20. *GRP_LIMIT* – maximum sum of insurance risk group,
21. *SUPER_GRP_LIMIT* – maximum sum of insurance super group,
22. *INS_BIRTH* - insured person birth year,
23. *APPLIED_SUM* – amount of money presented for payment,
24. *DIFF_POL_IND* – difference in days between policy start date and indemnity submission date,
25. *SERVICE* – probability that service is non-insurance,
26. *RISK* – risk group,
27. *STATUS* – result of risk assessment (insurance/pretension).

A variable *Status* is data label, if *Status=Pretension* data records is considered fraud event. If *Status=Insurance* data record is normal. After data cleaning 1997076 records left in data set, and 75069 of these records have status *Pretension*. In this case unintentional human mistake is also considered fraud event. It means that *Pretension* label reason can be "Not insured for risk", "Invalid indemnity detail", "Person not insured", "Amount is exceeded".

5. Experimental setup

Fraud detection in health insurance data is formulated as classification task. The aim of the model is to learn classifying data in to Insurance and Pretension records.

5.1. Evaluation Metrics

The following evaluation metrics were used to evaluate the performance of potential fraud detection models: accuracy, error rate, sensitivity, specificity [10].

These measures can be calculated based on the confusion matrix, which is a table with two rows and two columns that reports the number of false positives, false negatives, true positives, and true negatives (Fig. 5).

The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing [10]:

1. **TP** and **TN** indicate that the fraud and insured events are correctly classified (predicted).
2. **FP** means that the insured event was misclassified as a fraud event.
3. **FN** indicates that the fraud event was misclassified as an insured event.

Accuracy is the proportion of true results (both true positives and true negatives) and total number of cases [10, 11]:

$$A = \frac{TP + TN}{TP + TN + FP + FN}. \tag{8}$$

Error rate is the proportion of false results (both false positives and false negatives) and the total number of cases [10, 12]:

$$ER = \frac{FP + FN}{TP + TN + FP + FN}. \tag{9}$$

Sensitivity of a classifier is the ratio between correctly identified positives and actual positives [10]:

$$SE = \frac{TP}{TP + FN}. \tag{10}$$

Specificity of a classifier is the ratio between correctly classified negatives and actual negatives [10]:

$$SP = \frac{TN}{TN + FP}. \tag{11}$$

5.2. Data Preprocessing

Data preprocessing was performed with the following steps:

SERV_INST_CIT	INSURER_TYPE	SELL_UNIT_CODE	INS_PERS_GNDR	INS_PERS_CNTR	REC_TYPE
Riga	Juridical	VESR	Woman	Latvia	Individual
Riga	Juridical	VESR	Man	Latvia	Individual

REC_CNTR	TYPE	AUTH_CH_ID	AUTH_CH_CODE	DOC_TYPE	ODE_IMP_REC	SERV_INST_PRICE
Latvia	Application	89387877	WAI1001999	Scanned	NA	40
Latvia	Application	89386688	WAI1001999	Scanned	NA	3.5

MED_INST_PRICE	REAL_SUM	DISCOUNT	RCPT_SUM	EV_COUNT	RISK_LIMIT	GRP_LIMIT
40	459	100	459	1	3600	3600
3.5	388	100	388	1	250	250

SUPER_GRP_LIMIT	INS_BIRTH	APPLIED_SUM	DIFF_POL_IND	SERVICE	RISK	STATUS
3600	1968	912	118	0.36	Dental services	Pretension
250	1989	188	273	0.43	Rehabilitation services	Pretension

Figure 4: Examples of *Pretension* type record.

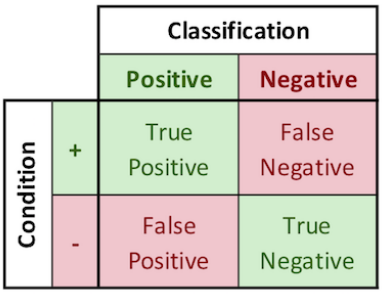

Figure 5: Confusion matrix [10].

1. The original data set was imbalance. For dividing into training and testing data sets all existing 75069 *Pretension* and 100000 *Insurance* records were selected with the widest possible combination of variable values. 70 % of the 175069 records were assigned to the train data set and 30 % to the test data set:
   - $Training$ 122548 records: 70000 *Insurance* and 52548 *Pretension*.
   - $Testing_1$ 52521 records: 30000 *Insurance* and 22521 *Pretension*.
2. The remaining 1822007 records of the original data set were assigned to $Testing_2$ data set. This data set was used to verify the effectiveness of the the best created model.
3. Categorical variables were expressed in numerical binary expression using one hot encoding technique.
4. Min-max normalization was used for numeric variables.

5.3. Results

The confusion matrices of the results obtained by the models, where the positive class is Insurance and the negative class is Pretension, are shown in figures 6, 7, 8, 9. Results show that decision trees outperform other methods, but does not work well with *Pretensions*, where Boosting is performs best.

Experiments results are provided in table 1. Bagging method is the most accurate (93.87%). Sensitivity shows how well positive are predicted (in this case – normal insurance event), most accurately 99.46 % predicted by decision tree. Specificity shows how well negatives are predicted (in this case – potential fraud), the best performing method (92.91%) is boosting. According to the error rate, the best performing method is bagging (6.13 %).

6. Conclusions

Experiments with the health insurance data set show that:

1. Signature based identification methods, constructed as classifiers for labeled insurance events

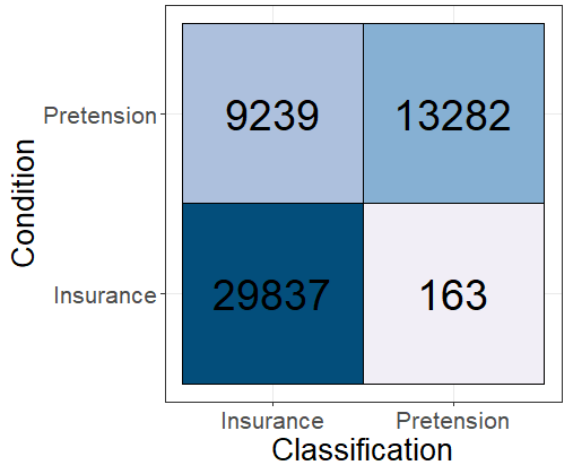

Figure 6: Decision tree confusion matrix.

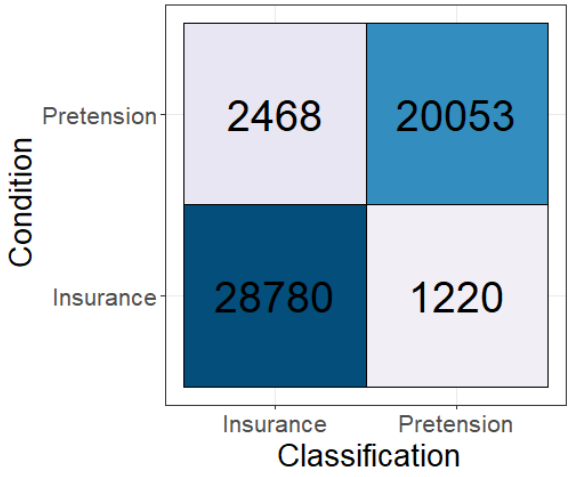

Figure 7: Random forest confusion matrix.

Table 1
Evaluation metrics

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	Error rate (%)
Decision tree	*82.1*	**99.46**	*58.98*	*17.9*
Random forest	92.98	95.93	89.04	7.02
Bagging	**93.87**	95.93	91.13	**6.13**
Boosting	87.68	*83.76*	**92.91**	12.32

data are effective for potential fraud (including erroneous claims) detection.

2. According to the accuracy, the best performing method is bagging (93.87 %).
3. Decision trees outperform other methods by sensitivity (99.46 %), but does not work well with *Pretensions* (specificity is 58.98 %), where Boosting is performs best (specificity is 92.91 %).

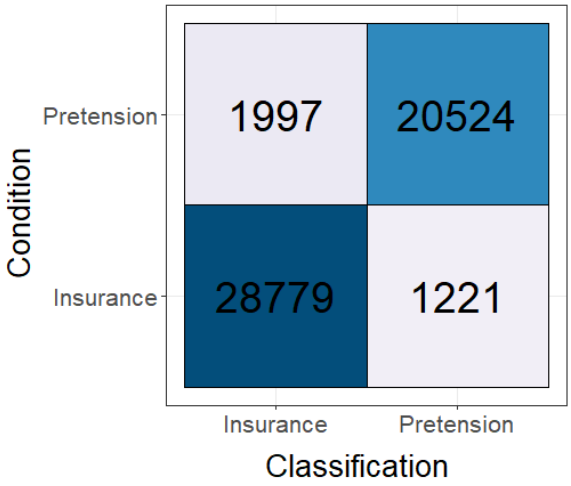

Figure 8: Bagging confusion matrix.

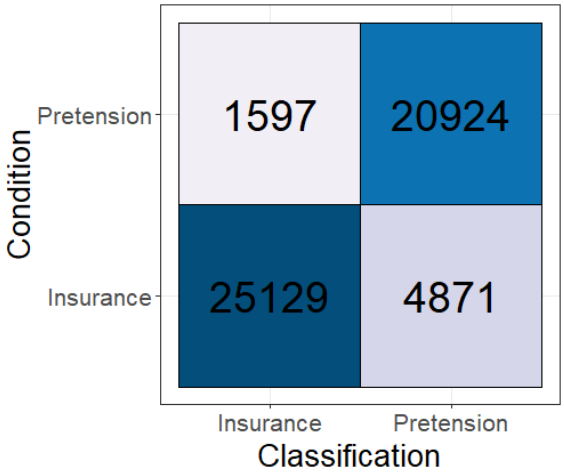

Figure 9: Boosting confusion matrix.

In further research it would be useful to analyze methods variety of Artificial Neural Networks and others anomaly detection models.

7. Acknowledgments

We thank Virginijus Jakštys and UAB Data house [1] for cooperation and useful insights. Research was partially funded by Lithuanian Business Support Agency (J05-LVPA-K-02-0013).

[1](www.data-house.lt)

References

[1] M. S. Anbarasi, S. Dhivya, Fraud detection using outlier predictor in health insurance data, in: 2017 International Conference on Information Communication and Embedded Systems (ICICES), 2017, pp. 1–6. doi:10.1109/ICICES.2017.8070750.

[2] V. Rawte, A. Srinivas, Fraud detection in health insurance using data mining techniques, 2015, pp. 1–5. doi:10.1109/ICCICT.2015.7045689.

[3] S. Kareem, R. Ahmad, A. Sarlan, Framework for the identification of fraudulent health insurance claims using association rule mining, volume 2018-January, 2018, pp. 99–104. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85047435296&doi=10.1109%2fICBDAA.2017.8284114&partnerID=40&md5=32f134dd6a775decec634b9c90eb8b70. doi:10.1109/ICBDAA.2017.8284114.

[4] A. Verma, A. Taneja, A. Arora, Fraud detection and frequent pattern matching in insurance claims using data mining techniques, in: 2017 Tenth International Conference on Contemporary Computing (IC3), IEEE Computer Society, Los Alamitos, CA, USA, 2017, pp. 1–7. URL: https://doi.ieeecomputersociety.org/10.1109/IC3.2017.8284299. doi:10.1109/IC3.2017.8284299.

[5] E. Larnyo, B. Dai, T. Udimal, W. Chen, Detecting and combating fraudulent health insurance claims using ann 56 (2018) 1–9. doi:10.7176/JHMN/2018-348.

[6] J. Le, Decision trees in r, DataCamp (2018). https://www.datacamp.com/community/tutorials/decision-trees-R.

[7] G. Lo Sciuto, S. Russo, C. Napoli, A cloud-based flexible solution for psychometric tests validation, administration and evaluation, in: CEUR Workshop Proceedings, volume 2468, 2019, pp. 16–21.

[8] J. Rocca, Ensemble methods: bagging, boosting and stacking, Towards Data Sci. (2019). https://towardsdatascience.com/ensemble-methods-\-bagging-boosting-and-stacking-c9214a10a205.

[9] F. Bonanno, G. Capizzi, G. Sciuto, C. Napoli, Wavelet recurrent neural network with semi-parametric input data preprocessing for micro-wind power forecasting in integrated generation systems, 2015, pp. 602–609.

[10] A. Tharwat, Classification assessment methods, Applied Computing and Informatics (2018). URL: http://www.sciencedirect.com/science/article/pii/S2210832718301546. doi:https://doi.org/10.1016/j.aci.2018.08.003.

[11] F. Beritelli, G. Capizzi, G. Lo Sciuto, C. Napoli, F. Scaglione, Rainfall estimation based on the intensity of the received signal in a lte/4g mobile terminal by using a probabilistic neural network, IEEE Access 6 (2018) 30865–30873.

[12] F. Beritelli, G. Capizzi, G. Lo Sciuto, C. Napoli, M. Woźniak, A novel training method to preserve generalization of rbpnn classifiers applied to ecg signals diagnosis, Neural Networks 108 (2018) 331–338.