# Applying machine learning to anomaly detection in car insurance sales

Michał Piesio[1], Maria Ganzha[1][0000−0001−7714−4844], and Marcin Paprzycki[2][0000−0002−8069−2152]

[1] Warsaw University of Technology, Warsaw, Poland
M.Ganzha@mini.pw.edu.pl
[2] Systems Research Institute Polish Academy of Sciences, Warsaw, Poland
marcin.paprzycki@ibspan.waw.pl

**Abstract.** Financial revenue, in the insurance sector, is systematically rising. This growth is, primarily, related to an increasing number of sold policies. While there exists a substantial body of work focused on discovering insurance fraud, e.g. related to car accidents, an open question remains, is it possible to capture incorrect data in the sales systems. Such erroneous data can result in financial losses. It may be caused by mistakes made by the sales person(s), but may be also a result of a fraud. In this work, research is focused on detecting anomalies in car insurance contracts. It is based on a dataset obtained from an actual insurance company, based in Poland. This dataset is thoroughly analysed, including preprocessing and feature selection. Next, a number of anomaly detection algorithms are applied to it, and their performance is compared. Specifically, clustering algorithms, dynamic classifier selection, and gradient boosted decision trees, are experimented with. Furthermore, the scenario where the size of the dataset is increasing is considered. It is shown that use of, broadly understood, machine learning has a realistic potential to facilitate anomaly detection, during insurance policy sales.

**Keywords:** anomaly detection · fraud detection · insurance sector · clustering algorithms · ensemble methods · gradient boosted decision trees.

## 1 Introduction

According to the 2018 data, available form the Polish Central Statistical Office, in the sector of compulsory motor vehicle insurance, the gross premium amounted to PLN 14.779 billion [1]. Therefore, for the enormous volume of transactions, even small errors in premium calculation can lead to substantial losses. It should be relatively obvious that, broadly understood, machine learning methods may be used to minimise or even eliminate problems, and at least to some extent protect against sales of "dangerous" policies.

In this work we consider possibility of applying *anomaly detection*, during the process of concluding a car insurance policy. In this context, we apply machine learning techniques to an actual dataset from an insurance company located in

Poland. This dataset has been hand tagged to indicate, which policies contain anomalies and which are correct.

In what follows we focus our attention on two classes of anomalies, and both involve values of parameters actually used to calculate the premium of the insurance. Any possible errors in their values could severely affect the financial gain from concluding such a policy and even lead to potential losses.

It should be relatively obvious that the considered anomaly detection problem is actually a *classification problem*, which can be stated as follows. Assuming that a new policy contract is introduced to the computer system (of an insurance company), should it be classified as "safe", or as "dangerous". In a perfect situation, if the considered methods were accurate, the "system" would autonomously conclude all "safe" contracts, while contracts "flagged as potentially anomalous" would be sent to "someone" to be double checked.

One more interesting issue is as follows. Let us assume that an insurance company has a certain amount of data that is used to train "classifier(s)". Over time, subsequent contracts will be concluded (or, rejected). Thus, how to deal with a systematically increasing volume of data. It would be possible to "every Saturday, re-train the classifier(s) with complete data set". Should all available data be actually used, or a "sliding window-type" approach applied? We will present initial results of our attempt at investigating this issue.

In this context, we proceed as follows. We start (in Section 2) with a more detailed conceptualization of the problem at hand. Next, in Section 3, we summarize state-of-the-art found in related literature. We follow with the description of the dataset and the preprocessing that was applied to it (in Section 5). These sections provide the background that is used in the following Section 6, where results of application of clustering, ensemble and decision tree based methods, to untagged data, are presented and analysed. This section should be seen as the core contribution of this work. Lastly, the problem of anomaly detection, considered from the perspective of a system where data is being systematically accumulated is considered, in Section 7.

## 2 Conceptualization of the anomaly detection problem

The presence of abnormal data, can be caused by many factors, and can lead to a significant financial loss, damage to the reputation, and/or customer trust. In the real-world practices of insurance companies, the main reasons for appearance of incorrect data are as follows.

- Human error – insurance agents deal with a steady stream of customers; with large number of tasks, and the time pressure, mistakes can happen;
- deliberate falsification of data – customers may try to deceive the company, in order to reduce the price of the insurance;
- incorrect data deliberately provided by the agent – agents can, dishonestly, increase their revenue, by falsifying customer data, to reduce the premium, to encourage customers to purchase the insurance; this, in turn, can provide the agent with a financial benefit, in the form of the margin on sales;

- errors in the internal computer system – unforeseen errors, in the sales system, may lead to an incorrect calculation of the premium;
- external system errors – connectivity with external systems that are (often) asked about the data directly related to issuing the policy, may be lost and this may result in errors and inaccuracies in the sales system.

Therefore, it would be extremely beneficial to have an effective method of verification of the entered data. The optimal solution would a "module", in the software supporting sales, that would analyse the data and "flag" transactions deemed as "suspicious". Next, it could inform both the agent and her/his supervisor about potential anomaly. If such module could not work in "real-time", it could examine recently sold policies. This would allow the discovery of problematic contracts that could be examined by human assessors. This could result in annulment of the most "problematic" ones, and could still be beneficial to the company. Moreover, it is important to capture changes that may occur over time to the sales process. Such changes may lead to an "evolution of anomalies". For example, agents can discover new methods of cheating, to increase sales.

Let us also stress that, with the amount of available data, and the complexity of the sales process, a thorough human oversight is impossible. Therefore, the only reasonable solution is to try to apply, broadly understood, data analytics approaches to facilitate detection of potential anomalies. The aim of this work is to investigate if such approach can actually work.

## 3 State-of-the-art

Let us now summarize anomaly detection methods that are potentially pertinent to the insurance sales case. Discussed are also methods that, although used in the insurance industry, were not a good choice for the issues covered here.

The first group of methods are the clustering algorithms. They can be used alone, or combined with more complex algorithms, to improve their results. It has been shown that, in datasets with a fairly low number of anomalies, in relation to the "normal data elements", algorithms based on the choice of the leader [2] may be used. Such methods are quite fast, even for large datasets, but do not work well when anomalies occur with high frequency. Therefore, this family of clustering algorithms was omitted.

Work reported in [3] uses clustering to audit claims payments. Proposed algorithm is based on the *k-means method* [4]. Analysis of obtained clusters can support the work of an expert, in the analysis of problematic cases. However, there is still the issue of automating operation of such algorithm, in order to check the policy before its conclusion. Cited work does not provide a solution.

A promising approach, to anomaly detection, is the use of *team methods*, which combine multiple estimators. This allows focusing on data locality, rather than on the case-by-case examination, taking into account the complete dataset [5]. It uses simple base estimators, $kNN$ (the k nearest neighbors), and $LOF$ (local anomaly coefficient). Both approaches are derived from clustering methods and, therefore, can be naturally compared with those described above.

The insurance industry, in [6], has explored use of *logistic model* and *naive Bayesian classifier* as base estimators. Moreover, *neural networks*, *decision trees* and *support vector machines* have been tried (see, [7]). However, this work dealt with (car insurance) claims reporting, not policy purchase. Another slightly problematic aspect was the use of base estimators, with relatively long training time.

Another group of algorithms, used in the insurance industry, are methods based on the support vector machines (SVM) [8]. SVM-based algorithms try to find decision boundaries, by mapping input data into a multidimensional kernel space. Meanwhile, the [9] demonstrates the use of *spectral analysis*, as an approximation of an SVM, to point to fraudulent insurance claims. SVMs, however, require accurate selection of hyperparameters, which can be problematic, given the dynamics of the sales system. It can be difficult to adapt the algorithms to work effectively during the sales process, which discourages their use.

Labeled training dataset allows use of "supervised learning". Here, the leading methods are *random forests* [10] and *gradient boosted decision trees* [11] (belonging to the category of decision tree based methods). Particularly promising results, in the context of detecting damage-related financial fraud, were reported in [11]. Additionally, the authors note that decision trees may successfully detect anomalies in newly arriving data, which is very useful in insurance industry.

In the context of detecting fraudulent claims, studies have been carried, reaching beyond insurances and claims. The focus of [12] is exploring claimants data, searching for suspicious relationship cycles that indicate the possibility of deals between agents and customers. However, this method was not investigated, due to lack of needed data, and difficulty of collecting it, to facilitate research.

In summary, various methods have been applied to detect anomalies in insurance industry. However, to the best of our knowledge, while detection of anomalies in reported claims is often investigated, detecting fraud in policy sales has not been discussed so far.

## 4 Selected algorithms

Ultimately, it was decided to investigate use of three approaches to anomaly detection: clustering, ensemble methods, and gradient boosted decision trees. Let us now discuss them in more detail, and state reasons for their selection.

### 4.1 Clustering algorithms

Clustering algorithms have been selected because of their simplicity and ease of visualization of results. Here, work reported in [3] applies the centroid algorithm, an extension of the k-means algorithm, based on working with mini-batches [13], which reduces the computation time. However, while being the most popular clustering method, it has problems when working with large datasets. Therefore, it was decided to try also different clustering algorithms, which may improve the results and/or reduce the computation time.

The first of the selected algorithms is clustering applying *Gaussian mixture model clustering* [14]. It is based on the assumption that the dataset has a Gaussian distribution of observation parameters. Moreover, while the centroid algorithm makes a "hard classification" (assigning each point to exactly one cluster), the mixed Gaussian method expresses cluster assignment as a probability, allowing to capture the uncertainty of classification. For the problem at hand, the mixed Gaussian model expresses how likely a given policy will end up in a cluster of "dangerous contracts". This can be advantageous, as it is often difficult to decide with certainty if a policy is problematic or not.

Second, clustering algorithms belonging to the family of hierarchical algorithms, were selected. The first is *BIRCH algorithm* (Balanced Iterative Reducing And Clustering Using Hierarchies; [15]), which was chosen for its efficiency for very large datasets. The last clustering algorithm is the (also hierarchical) Ward method [16]. It usually works well for data with a balanced number of data elements across clusters, as it is based on the sum of squares criterion.

Research reported in [3] operated on the full available dataset, which allowed for greater precision, but involved large computational costs. The selected algorithms require calculating the distance between each pair of points, which was too costly due to the size of the set considered here. In the case of the k-means algorithm, even the use of the optimized (mini-batch) version is associated with the time complexity of $\mathcal{O}(knmt)$, where $k$ is the number of clusters, $n$ is the number of records in the dataset, $m$ number of features of a single record, and $t$ number of iterations of the algorithm. Therefore, a decision was made to reduce the data to two or to three dimensions. This, in addition to the cost reduction, offers easier visualization and interpretation of results. For this purpose, we chose the t-distributed Stochastic Neighbor Embedding *t-SNE* algorithm [17]. This algorithm builds a probability distribution over pairs of points, so that similar points are close to each other, while points with large differences are spaced far apart in the graph. Considering the relatively high percentage of anomalies in the dataset, this approach was expected to allow for easier division of the set into fragments with high, or low, anomaly count.

It is also crucial to note that the information whether fraud/error occurred is not available during the clustering process. It is only considered after the set is clustered, to check how the anomalies are distributed within individual clusters. Overall, the algorithms try to divide the entire dataset in such a way that some clusters contain mostly policies that are fraudulent, while the remaining clusters consist mostly of correct ones. Note that the point is not to split the dataset into two clusters (bad and good), but into groups of clusters with mostly anomalous data and mostly correct data. Achieving such clustering could allow assessment whether a policy is "safe" (or not). Specifically, upon the arrival of a new policy, it is mapped to a single (two- or three-dimensional) point, and assign to the corresponding cluster. If anomalies dominate in this cluster, the new policy becomes "suspicious". Here, note that the mapping of new records, which can use the t-SNE algorithm (see, [18]), has not been tested as focus of this work was on clustering data into groups dominated by correct and bad policies.

### 4.2 Dynamic combination of classifiers

The ensemble-based approach often addresses the problem of incorrectly detecting large numbers of false positives, or false negatives. It has been tried in the insurance industry (see, [6, 7]). Therefore, it was decided to use Dynamic Combination of Detector Scores for Outlier Ensembles (DSO, [5]), an innovative framework that uses base estimators that, so far, have not been explored in the insurance sector.

### 4.3 Gradient boosted decision trees

A common problem in the detection of anomalies is the manual tagging of suspicious elements, which would allow for supervised learning. In the insurance industry, however, there are approaches that take advantage of labeled datasets for high precision anomaly detection [11]. Here, with very good results, the gradient boosted decision trees were used, to detect fraud in insurance claims. It should be also noted that decision trees are characterized by high speed and optimal use of resources.

## 5 Dataset

The dataset examined in this study was obtained from an insurance companies operating in Poland. It contains actual data on automobile insurances for individual clients. Here, data was collected during three years (June 2016-19). Obviously, data was anonymized and business sensitive information was removed. While a much wider set of parameters is collected during insurance policy sales, based on the literature and experience of the lead author, who worked in the insurance company, the following parameters have been selected:

- age of the car owner,
- age of the car co-owner,
- driving permit years,
- number of claims in the last year,
- years since the last insurance claim,
- total number of years of policy ownership,
- bonus-malus,
- car age,
- car mileage,
- difference (in days) between the policy conclusion date and the insurance protection beginning date,
- insurance premium price.

During preprocessing, records with null values, in any of the columns, have been deleted. The only exception was *age of the co-owner*, which was set to 0, when it was not defined. Only a small proportion (7.3%) of policies contain co-owner data, but the experience has shown that this information is valuable

in the fraud detection (so the data was kept). The dataset included policies that were not purchased, and they were removed. However, in the future they could be considered. Originally, the collection consisted of 39805 policies. After preprocessing, 35350 were left. Finally, the entire dataset has been normalized.

Experience of the lead author indicates that two popular types of anomalies, which directly affect the insurance price, should be considered. (1) Incorrect values of the parameter: *total number of years of policy ownership*. Here, when a value higher than the real one is entered in the system, it can lead to a reduction in the premium, and a financial loss to the company. Normally, the value provided by the customer should be verified by an external system, the Insurance Guarantee Fund [19], but this may not always be successful. Here,the service may experience interruptions in availability, or the personal data may be incorrect. If the verification fails, the value entered by the agent is used, which may lead to the abuses described above. (2) Incorrect values of the parameter *years since the last insurance claim*. Here, again, entering a value higher than the actual one may lead to an unwarranted reduction of the premium. Here, the verification by the Insurance Guarantee Fund may fail, again, for the same reasons.

The status of each element in the dataset was tagged by querying the Insurance Guarantee Fund, and setting two binary flags, if data is correct, it is a first, or second, type anomaly. The number of type one anomalies was 12568 (35.6%), and the number of type two anomalies was 9479 (26.8%). Moreover, 2985 (8.4%) of data elements belong simultaneously to both subsets of anomalies. However, this fact was not further explored; though it may be an interesting research direction to be undertaken. To represent the occurrence of anomalies, the t-SNE algorithm was used, resulting in a "flattened" representation depicted in Figure 1.

(a) Anomalies of the total number of years of policy ownership

(b) Anomalies of the years since the last insurance claim

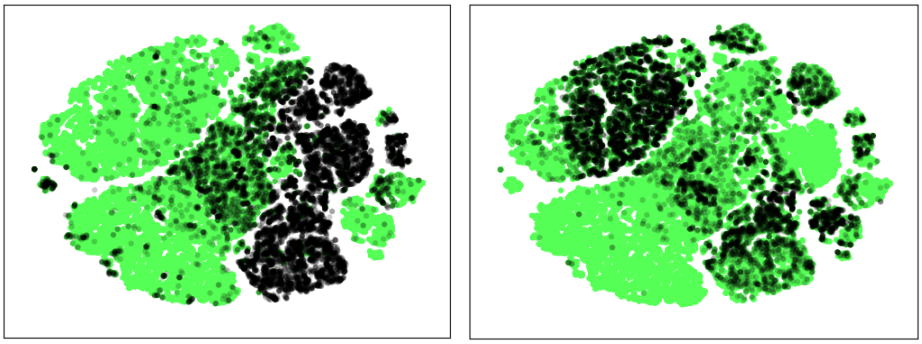

Fig. 1: Distribution of anomalies (black) in the dataset, reduced to two dimensions with the t-SNE algorithm.

As can be seen, policies that contain incorrect data constitute a large part of all contracts. In many places they are "very close to each other" (appear in large clusters), i.e. they have similar values of other parameters. However, in other areas, problematic policies are scattered across clusters of valid contracts.

## 6 Analysis of results

Let us now analyze the results delivered by the selected, and above described, anomaly detection algorithms.

### 6.1 Clustering algorithms

One of important questions, when applying clustering algorithms is the optimal number of clusters. Here, it was experimentally established that the optimal number is between 8 and 20 clusters. Too few clusters did not deliver an appropriate separation of (good/bad) policies. Too many clusters, on the other hand, caused an undesired combination of normal and abnormal data elements within clusters and resulted in unnecessary division of relatively homogeneous policies into separate clusters. Based on multiple experiments, it was decided to report, in this Section, results obtained for 12 clusters. This number brought about satisfactory clustering of policies and understanding of cluster content.

Moreover, operation of algorithms on data, reduced to three dimensions, was verified in order to check whether further flattening of data to two dimensions does not cause a large loss of information, and non-optimal clustering.

Finally, it should be remembered that the clustering algorithms work with untagged data. Hence, information about anomalies was imposed on the results of clustering, to understand and analyse the them. Specifically, each policy, where fraud or an error was committed, was plotted as a point in a different color. As an example, in Figure 2, results of applying BIRCH algorithm to the data with type one anomaly are presented.

It can be seen that BIRCH tends to cluster groups of data elements that are "far away", compared to neighboring groups (separated, green cluster circled with a blue ellipse), while this phenomenon is rare in other algorithms. Other, uniform areas were similarly clustered in the remaining three clustering methods.

All methods adequately separated the entire clusters into the area of the main occurrence of anomalies (marked with a green ellipse). This means that the anomalies are relatively similar, in terms of parameter values. There are as many as 8626 policies in the "black cluster", while there are only 1022 "correct contracts". Such clustering of anomalies, should facilitate their effective detection. If a new policy would be marked as belonging to this cluster it is likely to be "problematic".

In Figure 3, results of applying k-means algorithm to the data with type two anomaly are presented.

As can be seen, data with most anomalies (marked with the green ellipse) materializes in several clusters. Recall, that the goal is to split the data in such

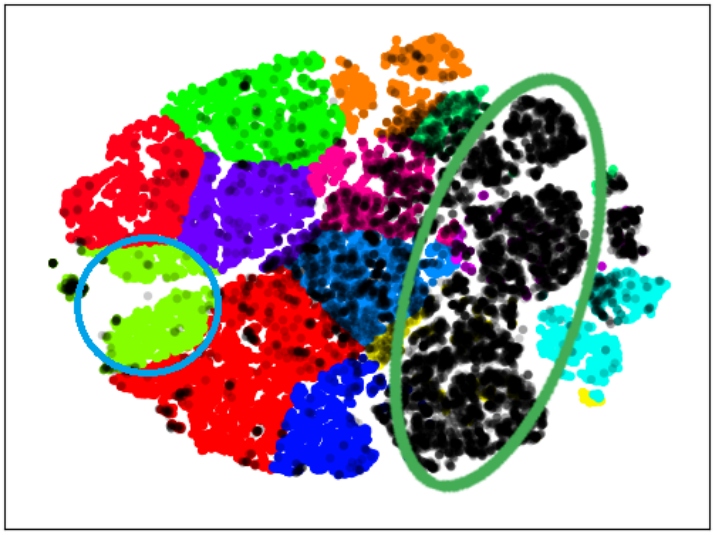

Fig. 2: Clustering results of BIRCH algorithm; 12 clusters; black points are anomalies; colored fragments represent separate clusters; type one anomaly; data reduced to/mapped into two dimensions.

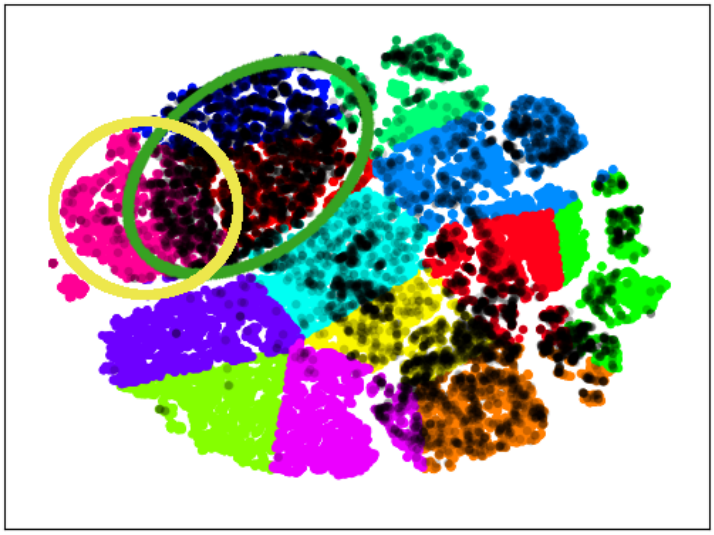

Fig. 3: Clustering with k-means algorithm; 12 clusters;, black points are anomalies; colored fragments represent separate clusters; type two anomaly; data reduced to/mapped into two dimensions.

a way that normal and abnormal data elements dominate separate clusters. Therefore, it is not a problem that the largest number of anomalies dominates multiple (smaller) clusters. In total, there are 4,577 anomalies in the circled area. However, one can notice that part of this "abnormal area" involves clusters (fragments) with very low number of anomalies (marked with a yellow circle). Therefore, if a new policy that would be assigned to the area covered by the yellow ellipse, it is not obvious if it is problematic or not.

Let us now present results of clustering when data is mapped into three dimensions. This approach preserves more information, but makes it more difficult to interpret the visualization. We start, in Figure 4, with type one anomaly, and BIRCH algorithm.

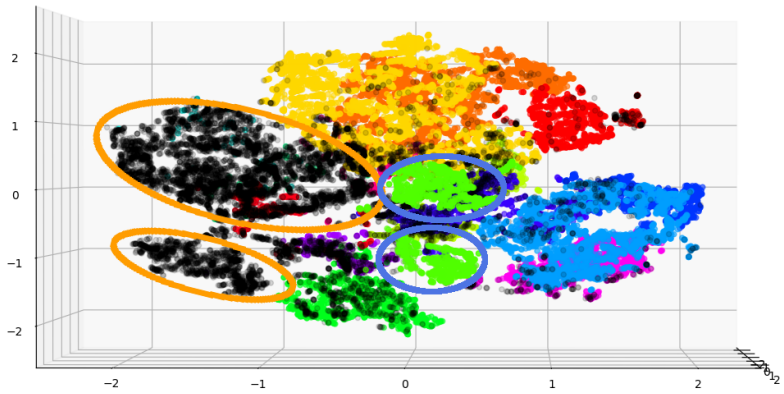


Fig. 4: Clustering results of BIRCH algorithm; 12 clusters; black points are anomalies; colored fragments represent separate clusters; type one anomaly; data reduced to/mapped into three dimensions.

As can be seen, mapping data into three dimensions allowed for better detection of clusters with a very low (maximum of 4.3%) anomaly count (marked with blue ellipses). In addition, the algorithms still managed to find clusters with a large number of abnormal policies (marked with orange ellipses). The separation of anomalies is at a satisfactory level, as 98.1% of dangerous policies have been located. Hence it is a good candidate for actual use in the insurance industry.

Next, three dimensional clustering of data with the type two anomaly, using k-means algorithm, has been depicted in Figure 5.

As can be seen, type two anomalies are more sparsely distributed across the dataset than type one anomalies. This suggests that policies with type two anomalies differ more from each other (as they are not closely grouped). This is also evidenced by the statistics on the "density of anomalies" in the clusters. The maximum density of anomalies in a single cluster was lowest for the k-mean's method, and it was 71.3% of data elements of the particular cluster. However,

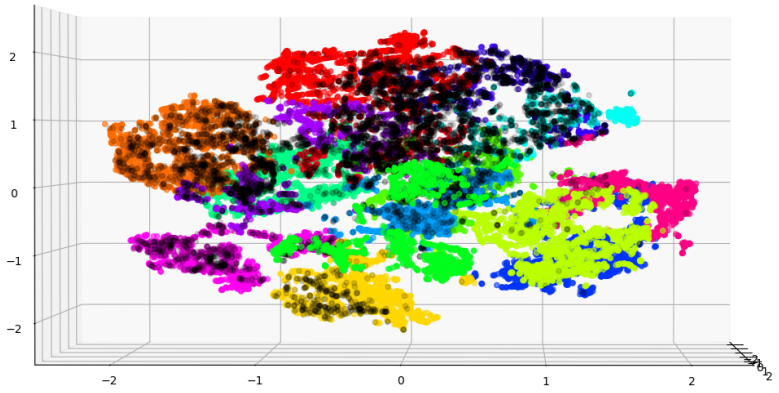


Fig. 5: Clustering results of k-means algorithm, 12 clusters, black points are anomalies, colored fragments represent separate clusters; first type anomaly; data reduced to/mapped into three dimensions.

this method recorded the most significant improvement when data was reduced only into three, and not into two, dimensions.

### 6.2 Dynamic combination of classifiers

In the case of a dynamic combination of classifiers, two yet unexplored in the industry base estimators were selected – k-Nearest Neighbours and Local Outlier Factor.

In order to achieve a clearer visualization of the results, the dataset was reduced to two dimensions, using the t-SNE method. The algorithm, on the other hand, operates on the full available dataset – in case of this method, the dataset reduction method was not used to optimize the computation time, as it was the case for the clustering algorithms. Such procedure was not necessary, because the algorithm used in this approach works with the time complexity $\mathcal{O}(nm + n \log n + s)$, where $n$ number of records in the dataset, $m$ number of features of a single record, and $s$ is the number of combined detectors. As unsupervised detectors are used, the algorithm does not use information as to whether fraud or an error has occurred in the policies tested.

We start with the presentation of results obtained using $k$NN base detectors for the first anomaly type (in Figure 6).

Here, recall that the $k$NN base estimators mark data elements as anomalous if they are further away from their closest neighbors than other points in the "nearby area". Hence, majority of data elements marked as abnormal, have been found on the edges of the clusters, and in the lower density groups (marked with a pink ellipse, in Figure 6). In the larger and more clustered areas "on the left hand side of the figure" (marked with green ellipses), the method was less likely to find anomalies, but still managed to point to a few "suspicious ones".

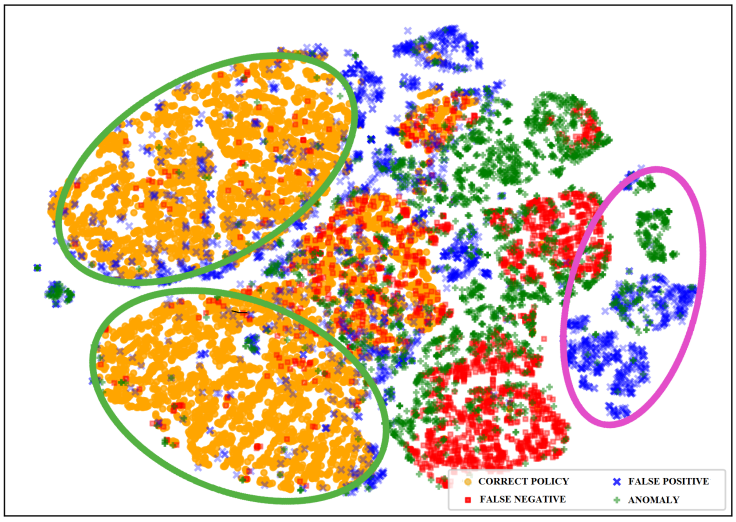


Fig. 6: Prediction of type one anomaly using $k$NN base detectors. Correctly identified anomalies are denoted as green, false negatives – red, false positives – blue, correct policies – yellow.

Nevertheless, numbers of policies incorrectly marked as anomalies (3,492) and undetected anomalies (4,833) are high. This makes it doubtful if this approach can be successfully used in practice.

Next, results of application of the $k$NN base estimators for type two anomaly are presented (in Figure7).

The majority of type two anomalies appear in a large, relatively dense area, in the upper left corner of Figure 7 (marked with a green ellipse). However, the algorithm only occasionally marks the policies from this part of the dataset as anomalous. Very low precision can also be observed in other areas. There, the main problem is incorrect marking of normal data as erroneous (cyan ellipses). Despite attempts to diversify the parameterization of the algorithm, it was not possible to obtain more satisfactory results. Moreover, only 19.4% of actual anomalies were correctly detected, while 23.3% of correct policies were marked as frauds. Likely, this indicates that the $k$NN base detectors should not be applied to this problem in practice.

Interestingly, the results based on the LOF method differ from these based on the $k$NN. This can be seen when comparing Figures 6 and 8. Here, a much higher number of points inside larger, more dense clusters is labeled as abnormal. However, the effectiveness of this approach is also low, which can be seen in large number of false positives. The fragment marked by the cyan ellipse, which has a lower density of points, very rarely has any policies marked as anomalies. Only 18.6% of data elements in this area have been flagged as anomalies when, in fact, this is precisely where the anomalies are located.

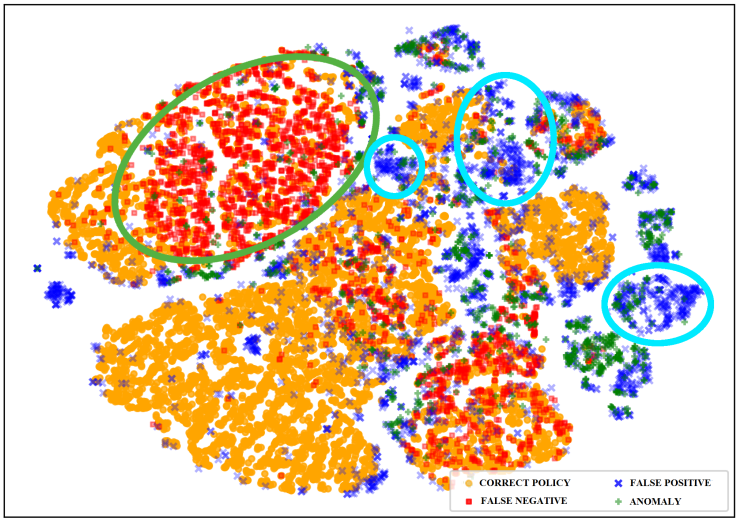


Fig. 7: Prediction of type two anomaly using $k$NN base detectors. Correctly identified anomalies are denoted as green, false negatives – red, false positives – blue, correct policies – yellow.

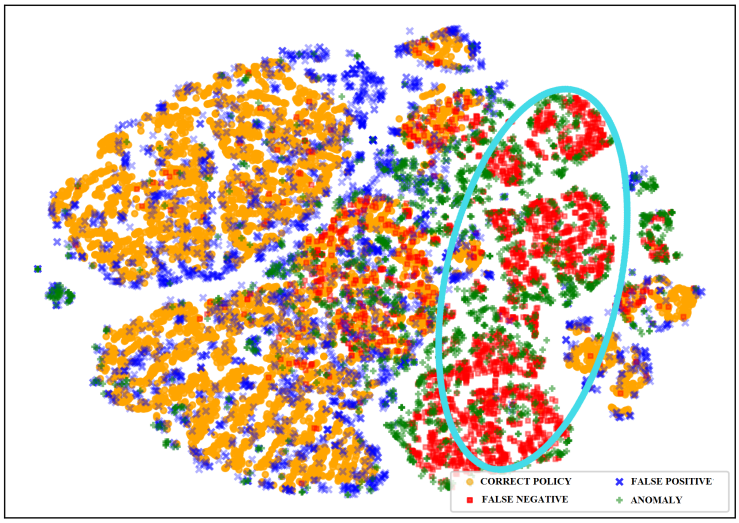


Fig. 8: Anomaly prediction for type one anomaly using LOF base detectors.

The LOF approach worked better for type two anomaly, since it detected at least some problematic policies (in area marked with a cyan ellipse, in Figure 9). Nevertheless, the efficiency of the algorithm remained low. Only 17.5% of anomalies were detected and false positives were also common (9.3% of correct policies were marked as anomalies).

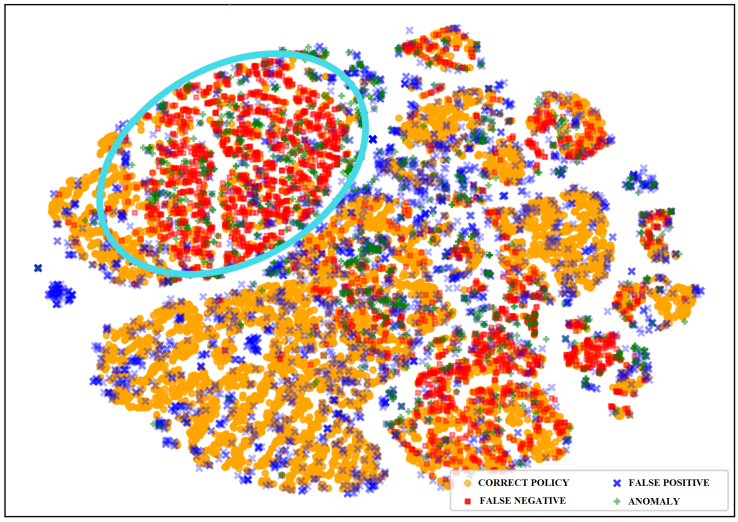


Fig. 9: Anomaly prediction of the parameter of years since the last insurance claim using LOF base detectors.

### 6.3 Gradient boosted decision trees

To complete the overview of experiments clustering data in two dimensions, experiments with gradient boosted decision trees are reported. We start with the type one anomaly (in Figure 10). Here, the XGBOOST library was used.

Analysing the Figure 10, one can observe that decision trees rarely misclassify anomalies when they are clustered in one area (for example, in the area marked by the red ellipse), Here, false positives appear only occasionally (99.8% of data was marked correctly). The most problematic area is the "center of data" (marked with a cyan ellipse). Here, normal and abnormal policies are mixed. However, still, 79.3% of policies have been correctly classified. Note that, in this case, the number of false positives and false negatives is similar (difference of only 4%).

The algorithm also coped relatively well (97.1% precision) with predicting the type two anomaly. However, in the main cluster of anomalies (indicated by the pink ellipse, in Figure 11), 6.4% of anomalies were not detected. Nevertheless, one can also observe areas practically free from mislabeled data elements (marked

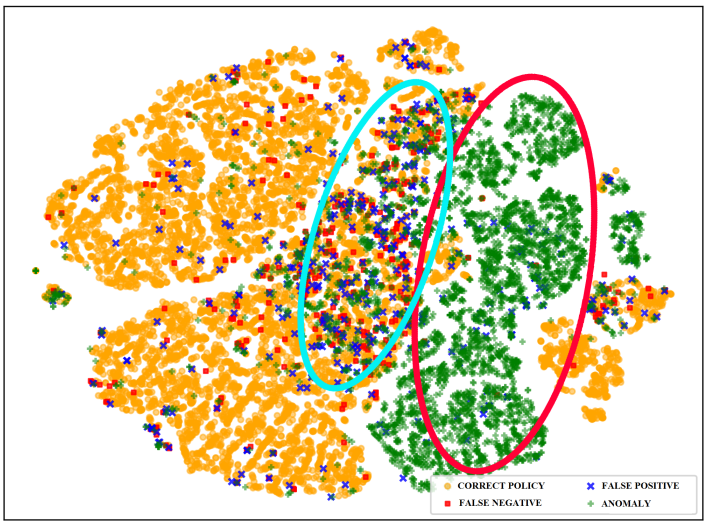


Fig. 10: Prediction of the type one anomaly using gradient boosted decision trees.

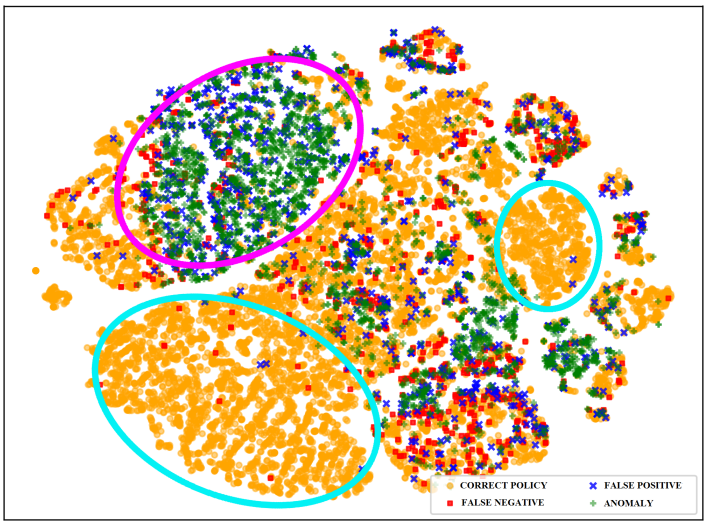


Fig. 11: Predicting anomaly of type two using gradient boosted decision trees.

with cyan ellipses). This is difficult to achieve even for supervised algorithms. In these areas, erroneous predictions constituted only 0.002% of policies.

### 6.4 Comparing effectiveness of different classifiers

Let us now present the combined results achieved by all approaches. Here, the accuracy was measured with the Area Under Curve (AUC) of the Receiver Operating Characteristic [19]. Specifically, the AUC was calculated as follows. Each of the clusters was assigned a numerical value equal to $\frac{ra}{ra+n}$, where $r$ is the ratio of the number of normal to abnormal policies in the entire set, $a$ is the number of anomalies in a given cluster, and $n$ the number of normal data elements in a given cluster. When preparing the ROC plot, all elements belonging to the clusters were marked as anomalies if the value of the cut-off point was lower than the value assigned to the cluster.

Let us start form type one anomaly. As represented by the results in Table 1 clustering data reduced to two dimensions had achieved worse precision than clustering data reduced to three dimensions, albeit the difference was minimal for some clustering algorithms. Dividing the dataset into more clusters tended to produce better results. Overall, best result achieved by the gradient boosted decision trees (AUC=0.9907). However, all three dimensional approaches, also achieve accuracy, which would be satisfying for majority of insurance companies.

Table 1: Type one anomaly; accuracy measured with AUC; best results obtained with the indicated parameters.

| Algorithm | Parameters | AUC |
|---|---|---|
| clustering BIRCH 2D | 14 clusters | 0.9052 |
| clustering Ward 2D | 16 clusters | 0.9273 |
| clustering Gauss 2D | 18 clusters | 0.9334 |
| clustering k-means 2D | 18 clusters | 0.9224 |
| clustering BIRCH 3D | 20 clusters | 0.9381 |
| clustering Ward 3D | 18 clusters | 0.9284 |
| clustering Gauss 3D | 20 clusters | 0.9312 |
| clustering k-means 3D | 16 clusters | 0.9313 |
| ensemble kNN | maximum of averages pseudo-truth | 0.6884 |
| ensemble LOF | average of maximums pseudo-truth | 0.5766 |
| decision trees | tree depth 8 | 0.9907 |

Precision of the algorithms, for the type two anomaly, was significantly lower across all tested methods (see, Table 2). The smallest decrease of precision was observed for the gradient boosted decision trees (decline of only 0.03). The decline for the remaining clustering approaches was of order of 0.1. Finally, accuracy of the ensemble methods, was barely better, or even worse, than random selection (i.e. one could as well flip a coin to decide is a policy was an anomaly or not).

Table 2: Type two anomaly; accuracy measured with AUC; best results obtained with the indicated parameters.

| Algorithm | Parameters | AUC |
|---|---|---|
| clustering BIRCH 2D | 14 clusters | 0.7954 |
| clustering Ward 2D | 20 clusters | 0.8304 |
| clustering Gauss 2D | 20 clusters | 0.8230 |
| clustering k-means 2D | 18 clusters | 0.8060 |
| clustering BIRCH 3D | 18 clusters | 0.8113 |
| clustering Ward 3D | 20 clusters | 0.8163 |
| clustering Gauss 3D | 18 clusters | 0.8332 |
| clustering k-means 3D | 20 clusters | 0.8234 |
| ensemble kNN | maximum of averages pseudo-truth | 0.5164 |
| ensemble LOF | average of maximums pseudo-truth | 0.4743 |
| decision trees | tree depth 8 | 0.9607 |

## 7 Accumulating data in insurance company system

Let us now consider a typical situation when data, concerning sold policies, is being accumulated in the computer systems(s) of the insurance company. As noted above, it is possible to use some (possibly very simple) method to compare each new policy to the existing data model and establish "where it belongs". As a result the policy can be judged to be normal or potentially problematic. Next, every N days (e.g. every weekend) the complete existing dataset can be used to create a new data model. However, in this approach, an important question needs to be answered, in order to achieve the highest possible anomaly detection precision. How to deal with "older" data. Should all data be kept and used to train the model or, maybe, older data should be removed, as it may have adverse effect on detecting anomalies in the dynamically changing world.

There is one more aspect of this problem. While the dataset examined in this study is relatively large, this is not going to be the case for companies that are just entering the insurance industry. Here, they will have only a small initial dataset available as a source for model training. Note that it is not important how large and how small are the respective datasets. The key point is that one of them is considerably smaller than the other. Therefore, it was decided that by limiting the size of the (currently available) dataset it will be possible to simulate a situation where a relatively new insurance company tries to create a business tool to be used with a small (initial) dataset.

Therefore, as an extra contribution, it was decided to perform some preliminary investigations, to lay foundation of understanding of what is happening in the considered scenario. Thus, while thus far we have ignored the chronology of data elements, this information was available in the dataset. Therefore, the data was sorted chronologically, and only the "initial part" was allocated to the training set. First, the dependence of the precision of anomaly detection, on the size of the initial training set was tested. Models built on the initial training sets consisting of 6, 12, 18, 24, and 30 thousand data elements were prepared and

their effectiveness was compared on the corresponding fragments of the test set. Each fragment of the set, for which the effectiveness was checked, consisted of 2000 policies. Results for type one anomaly have been depicted in Figure 12.

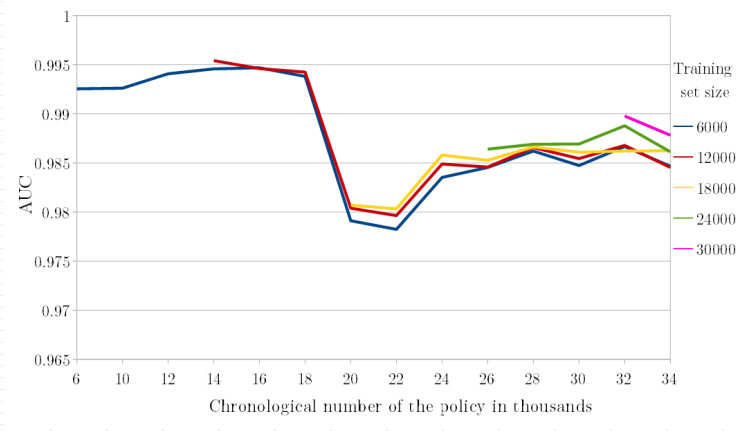


Fig. 12: Precision of prediction depending on the training set size; type one anomaly; AUC; gradient boosted decision trees.

The first observation is obvious, the lowest accuracy occurs for the smallest training set. However, even then, accuracy remains acceptable for a real-life insurance company. Sudden changes to the precision around the 18 thousandth policy can be historically tracked to the changes to the tariff used to determine the final insurance premium sum, which changed the structure of the dataset. They could have also resulted in agents finding different methods of cheating.

The second considered scenario was, what would happen if one used a "sliding window" and kept only certain volume of most current data, and discarded the older ones, in order to capture the latest trends. This scenario was investigated as follows – results from the previous experiment were compared to the precision obtained with the training set where the older data was omitted. Sliding window of 12,000 data elements was used. Results have been depicted in Figure 13.

As can be seen, across the board, omitting old data did not improve the results delivered by the trained models. Nevertheless, in sporadic cases, there were areas where the models trained with the reduced training sets obtained better results. However, the differences were not significant.

In summary, while it is obvious that a number of open questions remains, and is worthy investigation, it was established that: (1) for the size of the dataset that was available for this study, it seems that the best approach it to keep all the available data; and (2) even for relatively small datasets, gradient boosted decision trees provide reasonable accuracy of prediction of anomalies of both types.

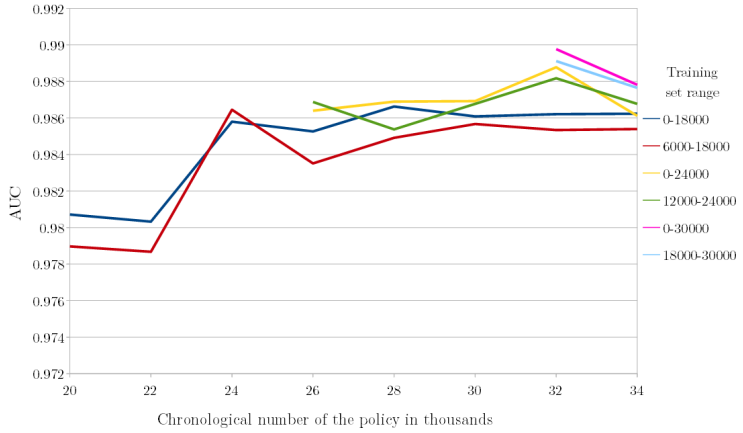


Fig. 13: Precision of prediction depending on the training set range; type one anomaly; AUC; gradient boosted decision trees.

## 8 Concluding remarks

In this work we have considered a novel problem of detecting anomalies in policy sales. Experimental work was based on a car insurance dataset, obtained from an actual insurance company. To establish anomalies in an unsupervised learning scenario, three classes of algorithms have been applied: clustering algorithms, dynamic classifier selection, and gradient boosted decision trees.

The gradient boosted decision trees algorithm dominated all the tested methods in terms of the anomaly prediction accuracy. The favorable characteristics of this approach is also that it allows for easy and quick analysis of new policies, already during their conclusion, allowing the company to be adequately secured against financial losses. Further analysis of the algorithm also showed that it performs well when the size of the dataset, available for training, is small. This may be a significant advantage for start-ups, wanting to use machine learning to guarantee optimal protection against potential frauds and mistakes. The study of the deviation of the effectiveness of the algorithm considering the influx of new data showed that this method is quite resistant to potential changes in the nature of anomalies. However, especially in this area, a number of open questions remain. Nevertheless, to address them properly, a much larger (especially from the time of collection perspective) dataset should be used.

## References

[1] Polish Central Statistical Office: Polish Insurance Market in 2018, https://stat.gov.pl/en/topics/economic-activities-finances/financial-results/polish-insurance-market-in-2018,2,8.html (2019)

[2] Priyanga Dilini Talagala, Rob J. Hyndman, Kate Smith-Miles: Anomaly Detection in High Dimensional Data, Journal of Computational and Graphical Statistics (2020)
[3] Sutapat Thiprungsri, Miklos A. Vasarhelyi: Cluster Analysis for Anomaly Detection in Accounting Data: An Accounting Approach, The International Journal of Digital Accounting Research (2011)
[4] J. MacQueen: Some methods for classification and analysis of multivariate observations, Berkeley Symposium on Mathematical Statistics and Probability (1967)
[5] Yue Zhao, Maciej K. Hryniewicki: DCSO: Dynamic Combination of Detector Scores for Outlier Ensembles, ACM KDD Workshop on Outlier Detection De-constructed (ODD v5.0) (2018)
[6] S. Viaene, R. A. Derrig, B. Baesens, and G. Dedene: A Comparison of State - of - the - Art Classification Techniques for Expert Automobile Insurance Claim Fraud Detection, Journal of Risk & Insurance (2002)
[7] Amira Kamil Ibrahim Hassan, Ajith Abraham: Modeling Insurance Fraud Detection Using Ensemble Combining Classification, International Journal of Computer Information Systems and Industrial Management Applications (2016)
[8] Dave DeBarr, Harry Wechsler: Fraud Detection Using Reputation Features, SVMs, and Random Forests, Proceedings of the International Conference on Data Science (2013)
[9] Ke Niana, Haofan Zhanga, Aditya Tayal, Thomas Coleman, Yuying Li: Auto insurance fraud detection using unsupervised spectral ranking for anomaly, The Journal of Finance and Data Science (2016)
[10] Simon D. Duque Anton, Sapna Sinha, Hans Dieter Schotten: Anomaly-based Intrusion Detection in Industrial Data with SVM and Random Forests, International Conference on Software, Telecommunications and Computer Networks (2019)
[11] Najmeddine Dhieb, Hakim Ghazzai, Hichem Besbes, Yehia Massoud: Extreme Gradient Boosting Machine Learning Algorithm For Safe Auto Insurance Operations, IEEE International Conference of Vehicular Electronics and Safety (2019)
[12] Arezo Bodaghi, Babak Teimourpour: Automobile Insurance Fraud Detection Using Social Network Analysis, Applications of Data Management and Analysis (2018)
[13] Javier Béjar: K-means vs Mini Batch K-means: A comparison, KEMLG - Grup d'Enginyeria del Coneixement i Aprenentatge Automàtic - Reports de recerca (2013)
[14] Geoffrey J. McLachlan, Kaye E. Basford: Mixture models. Inference and applications to clustering (1988)
[15] Tian Zhang, Raghu Ramakrishnan, Miron Livny: BIRCH: an efficient data clustering method for very large databases (1996)
[16] Joe H. Ward Jr.: Hierarchical Grouping to Optimize an Objective Function (1963)
[17] Laurens van der Maaten, Geoffrey Hinton: Visualizing data using t-SNE, The Journal of Machine Learning Research (2008)
[18] Laurens van der Maaten: Learning a Parametric Embedding by Preserving Local Structure, Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics (2009)
[19] Tom Fawcett: An introduction to ROC analysis, Pattern Recognition Letters (2006)
[19] Insurance Guarantee Fund, https://www.ufg.pl/infoportal/faces/pages_homepage