American Academic Scientific Research Journal for Engineering, Technology, and Sciences

ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

https://asrjetsjournal.org/index.php/American_Scientific_Journal/index

Detection of Rare Events: The Need to Know the Customer

Ayse Humeyra Bilge[a], Tarkan Ozmen[b], Ayse Tosun[c*]

[a]*Kadir Has University, Istanbul 34083, Turkey*

[a]*Email: ayse.bilge@khas.edu.tr*

[b]*Email: tarkan.ozmen@gmail.com*

[c]*Email: aysetosun2018@gmail.com*

Abstract

The prediction of customer complaints based on a time series of invoices is a two-stage process consisting of determining anomalies in the sequence of invoices and assessing the response of the customers to these anomalies. In the telecommunication sector, the average complaint rate is approximately 10^{-4} hence the prediction of customer complaints falls in the realm of rare event detection. Detecting rare events poses a significant challenge when working with unbalanced datasets. In machine learning applications, oversampling of the minority class and under sampling of the majority class in the training set are well-known preprocessing tools for creating a more balanced set. In previous work, [14] we proposed a cluster based under sampling approach as an alternative to random under sampling of the majority class, based on splitting heterogeneous data into homogeneous subsets, using Principal Component Analysis, to reduce variability within clusters. In the present work we propose a method for assessing the response of the customers to anomalies detected in the time series of invoices.

Keywords: Time series analysis; Unbalanced data; clustering and classification; customer scoring.

1. Introduction

Customer complaints are mostly rare events, but they are a primary cause of customer dissatisfaction and companies try to predict and handle possible causes before the occurrence of a complaint. It is thus desirable to devise efficient and cost-effective strategies for predicting complaints.

Received: 11/29/2024
Accepted: 1/30/2025
Published: 2/12/2025

----

* Corresponding author.

Unless there are clues on possible causes and motives for complaints, predictions are based on the detection of anomalies in a time series and associating these anomalies with the events to be identified, i.e., complaints. The aim of the present work is to framework for dealing with rare customer complaints, within a heterogeneous customer pool.

In the case of a heterogeneous customer pool, a detailed statistical analysis of the customer behavior is indispensable for obtaining satisfactory predictions and for applying efficient corrective measures. It is advisable to remove outliers as customers to be paid special attention and proceed with the remaining group, which is still highly heterogeneous. The first step consisting of the detection of anomalies is a more or less standard classification problem for which elaborate machine learning tools are available. Nevertheless, the detection of extremely rare events is a challenging problem that may further be complicated by heterogeneities in the data set. The second step consisting of associating anomalies to complaints requires a more delicate treatment. Because the features of the majority group that do not file any complaint may display anomalies, possibly because some of these anomalies correspond to conscious actions. Furthermore, different people may react differently to the same anomaly. Thus, whichever classifier is used, the amount of "false positive" cases can be quite large. The key to the solution of this problem is to integrate the response of the customer to the anomaly to the classification problem by a feedback mechanism.

In the present work, we propose a customer scoring system that monitors the reaction of a customer to anomalies of different types and generate a score for the association of anomalies to complaints. Preliminaries and literature survey are presented in Section 2. A detailed statistical analysis is presented in Section 3. Customer scoring is given in Section 4. An illustrative example for scoring is given in Section 5. Concluding remarks are given in Section 6

2. Preliminaries

The classification problem: In dealing with a heterogeneous pool of customers, clustering and classification of the customers with respect to the behavior to be analyzed is a crucial preliminary step. The classification problem is the process of deciding on the group to which a given individual belongs. The population is divided into disjoint sets and each individual is characterized by a set of features. A classification algorithm takes the features as an input and decides on the “class” of that individual. In the case of classifying customers with respect to complaint behavior, roughly speaking, we obtain groups with low mean-low standard deviation invoice series, invoice series with high variability and a group of invoice series that are characterized by extreme values in an otherwise steady time series.

In order to obtain a classification, algorithms are developed by an analysis of the data; split as “training” and “test” sets. Machine learning methods applied to this classification, result in either correct or incorrect categorization into four groups: “True-Positive,” “True-Negative,” “False-Positive,” and “False-Negative”. The performance of the classification algorithms is measured by sensitivity (the ratio of “True-Positive” cases to total “Positive” cases) and specificity (the ratio of “True-Negative” cases to total “Negative” cases).

In the case of rare event detection data is unbalanced, i.e., “positive” cases that are aimed to be detected form the “minority” class, while “negative” cases constitute the “majority class. For a successful training of the model, it is necessary to provide a sufficient number of positive and negative examples for the algorithm's training phase. However, in the case of rare events, the number of positive cases is often too low, and oversampling techniques, such as the Synthetic Minority Oversampling Technique (SMOTE), can be used to artificially increase the number of positive examples

When the ratio of minority to majority class is extremely low, oversampling of the minority class may not be sufficient to balance the dataset and random under-sampling of the majority class is often applied. Random under sampling works well for homogeneous data, however, for heterogeneous data, this approach may fail to yield satisfactory results. In previous work [14] we proposed a preprocessing approach consisting of clustering the data into homogeneous subsets and sample data at based on Principal Component Analysis in order to reflect faithfully the variance in each subset. This method has been applied to detecting customer complaints related to monthly invoices at a major telecommunications company with about 18 million customers and a monthly complaint rate of 1,300 representing an extremely rare event.

Imbalanced data problems have been a significant challenge in various real-world applications, including medical diagnostics, financial risk prediction, and text classification tasks such as spam detection [1]. In particular, imbalanced datasets frequently arise in domains where the minority class is the primary focus, such as in fraud detection, rare disease identification, or anomaly detection [2]. If left unaddressed, this imbalance can lead to predictive models that heavily favor the majority class, often at the cost of ignoring the minority class, resulting in poor overall performance in critical cases. This has become a central issue in machine learning and data mining, where learning algorithms tend to prioritize accuracy on the majority class while disregarding the minority class, leading to suboptimal results in high-stakes scenarios.

As a result, there has been a growing interest in addressing this issue in the data mining and machine learning fields. Approaches to solving the imbalanced data problem are typically categorized into four major groups: algorithm-level techniques, data-level techniques, cost-sensitive learning methods, and ensemble methods [3,4]. Cost-sensitive methods introduce penalties for misclassifying instances from the minority class, encouraging the model to prioritize accuracy for that class. On the other hand, data-level techniques such as over-sampling and under-sampling directly manipulate the training data to create a more balanced class distribution. These preprocessing steps help ensure that learning algorithms are exposed to sufficient examples from both the majority and minority classes, improving their performance on imbalanced datasets [5].

Among data-level techniques, the **Synthetic Minority Over-sampling Technique (SMOTE)**, proposed by Chawla and his colleagues [6], has dominated the over-sampling area. SMOTE generates synthetic instances for the minority class by interpolating between existing instances, helping to reduce the class imbalance. Several studies, including an investigation by Fernández and his colleagues [7], have demonstrated that SMOTE and its variants can significantly enhance the performance of classifiers on imbalanced datasets. However, it is not without limitations. While SMOTE is a powerful tool for enhancing classification performance, it may introduce noise or over-generalization if the minority class contains significant variance.

Similarly, **random under-sampling**—a method where a random subset of the majority class is selected—can help in reducing class imbalance by discarding excessive instances from the dominant class. However, random under-sampling carries the risk of information loss, as important data points may be removed arbitrarily. To address this issue, researchers have developed hybrid approaches. For example, **RUSBoost**, introduced by Seiffert and his colleagues [9], combines random under-sampling with the popular boosting technique, thereby improving classification accuracy by mitigating class imbalance while preserving valuable data. Another hybrid approach, **UnderBagging**, utilizes bagging techniques combined with random under-sampling to create a balanced training dataset and has been shown to improve classifier performance in imbalanced scenarios [9].

In domains such as credit card fraud detection, highly imbalanced datasets are the norm. Randhawa and his colleagues [10] used Adaboost and Majority Voting to evaluate twelve different classifiers in this domain, showing that ensemble methods provide superior results compared to individual classifiers. Similarly, Buda and his colleagues [8] found that combining under-sampling with boosting techniques significantly enhanced classification performance, particularly in highly imbalanced datasets where the minority class constitutes less than 1% of the overall data.

The literature suggests that no single method is universally effective for addressing imbalanced data across all datasets [11]. Typically, the choice of method depends on the specific characteristics of the dataset and the nature of the minority class problem. As such, this work proposes a novel approach that aims to enhance predictive performance by first segmenting the data into clusters with similar characteristics and then applying classification models within each segment. This clustering-based methodology, discussed in the following sections, offers a promising way to handle imbalanced data by tailoring the classification process to the specific structure of the dataset, an approach not widely explored in the current literature.

Description of Data:

In this study, at we worked with monthly invoices of a major telecommunication company that has a customer pool of about 20 million people. At various phases of the analysis, we worked with different samples from this pool. Various statistics that we present aim to display the proportion of complaints and the structure of multiple complaints to a given invoice.

Table 1 below, displays the number of complaints to a sample consisting of over 2.3 million customers over 9 months (07/2021-03/2022). This customer pool has been used for the analysis of detecting retail expenditures[13].

Table 3: Distribution of complaints (NC: No Complaint, C: Complaint, over 2.3 customers, 07/2021-03/2022)

Month	1	2	3	4	5	6	7	8	9
NC	2 375 072	2 375 153	2 375 120	2 375 100	2 375 136	2 375 351	2 375 438	2 375 497	2 375 351
C	1 444	1 363	1 396	1 416	1 380	1 165	1 078	1 019	1 165

A second sample consists of an 11-month billing history (09/2021-06/2022) for 843,637 subscribers has been used for a clustering of customer invoice sets [13].The number of multiple objections are given in Table 2 below. It is recommended that customers who have filed objections for more than 40% of the monitoring period be placed on a separate observation list and excluded from the general analysis. It can be observed that the majority of customers did not file any objections, and multiple objections are rare.

Table 4: Statistics of multiple objections based on Sample 2 (843,637 customers, 09/2021-06/2022)

Number of Objections	0	1	2	3	4	5	6	7	8	9	10	11
Number of people	830499	12619	384	73	28	15	9	1	2	3	1	3

The main result of our previous work is that complaints are extremely rare, they are in general related to anomalies in the invoice series, but extreme anomalies in general correspond to conscious spendings and they don't lead to objections and complaints. Furthermore, although rare, repeated objections indicate the need for special treatment and corrective actions for a group of customers. Finally, even after classifying customers into relatively homogeneous subsets, still the response of the individuals to anomalies displays high variability. As a result, in the present work we investigate to role of customer response to anomalies.

3. Statistical analysis of the customer pool

The following chart displays typical the billing sequences for customers who objected six or more times within the 11-month period. In these graphs, an asterisk (*) on the curve indicates an objection to the corresponding invoice.

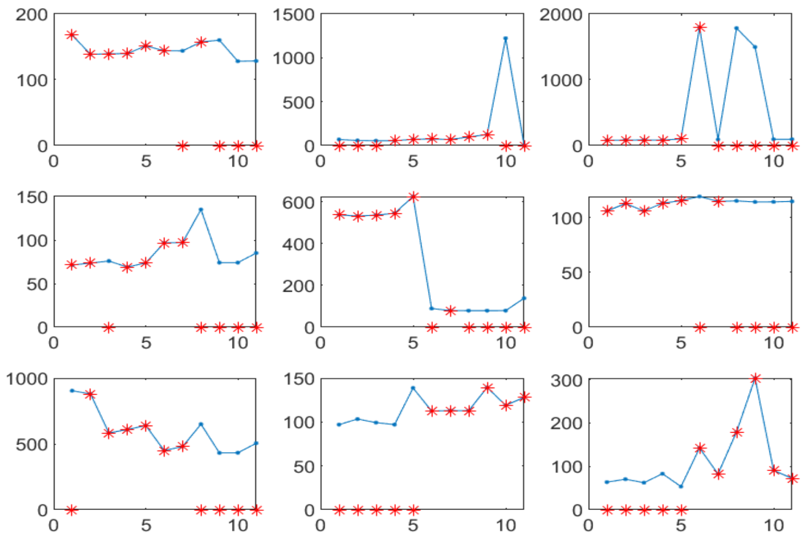

Figure 1: Subscribers with six to seven objections during the 11-month period.

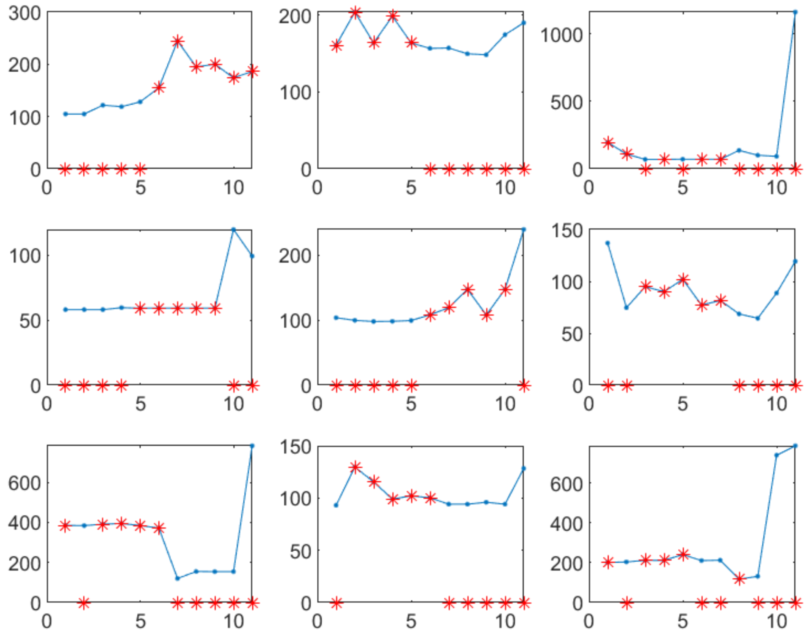

Figure 16

As seen in the graphs, some objections are related to the invoice amounts. However, whether high or low, continuous objections to invoices indicate that these customers have issues that need to be managed specifically. In such cases of persistent objections, special attention should be given to the subscriber, and measures should be taken to prevent the ratio of objections from reaching that level. Therefore, subscribers who have filed objections exceeding 40% during the observation period will be placed on a separate observation list, "Observation List 1" and removed from the general list.

Invoices Containing Extreme Values:

Invoices that exceed a specified threshold value in any given month are considered extreme expenditures. When such an expenditure occurs, the customer should be placed on a separate observation list for assessing the nature and cause of the expenditure. This threshold value may vary from month to month. In the current example, the threshold value has been set at 10,000 TL for all months, and 138 customers who made extreme expenditures in any month have been identified. These customers have filed objections to only six invoices. Three of these objections were made against invoices with extreme values, while the other three were made against relatively low-amount invoices. Typical billing sequences of these subscribers are displayed below.

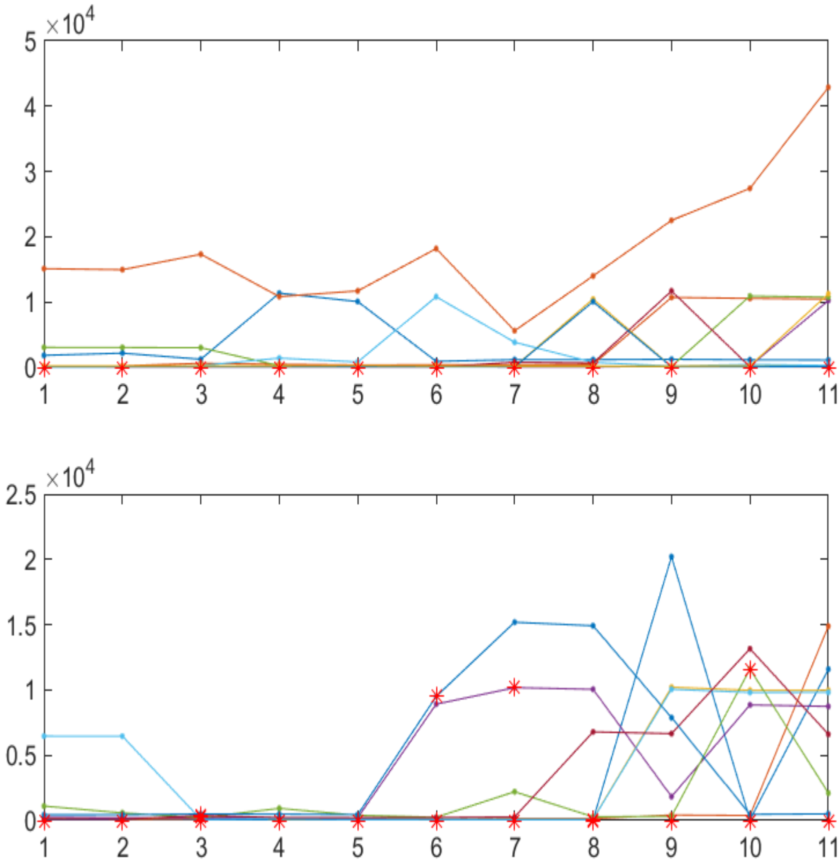

Figure 2: Subscribers with invoices exceeding 10,000 TL in any month.

These invoices are interesting in that they also reflect the retail-related expenditures discussed earlier, and they have been provided in detail to exemplify these types of expenditures. These customers are moved to "Observation List 2". At this stage, the analysis will continue with 843,465 invoice records.

General Structure of Features

The distribution of the average, standard deviation, and maximum-minimum difference attributes is shown below in Figure 3. In this way, invoice records that are concentrated on a straight line are noteworthy. It will be demonstrated that these records consist of invoices containing a small number of extreme values. However, to perform various proportional calculations, invoice records with zero or very low attribute values will be eliminated first.

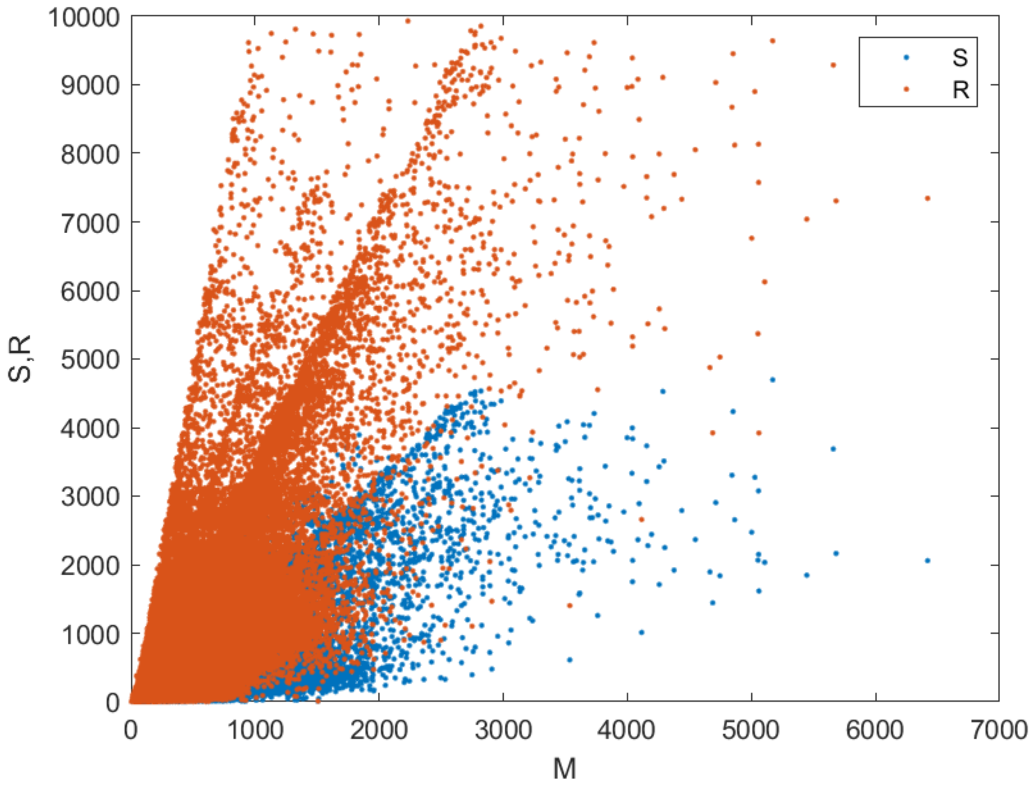

Figure 3: Scatter plots of the standard deviation (S) and the range (R) with respect to the mean (M).

From this data, 10,170 customers with a standard deviation of less than 1 TL, 69 customers with a maximum-minimum difference of less than 1 TL, and 21 customers with an average of less than 10 TL have been identified and transferred to the "Observation List 3". Those customers are unlikely to file any objection, and the general analysis will continue with 833,205 customers. We note that by increasing the lower bounds it is possible place a considerable amount of customers in this no-risk group.

After this stage, clustering will be performed based on spending trends, and separate time series analyses will be conducted for each cluster.

Almost 2-level invoice series

A typical group of invoice series consist of a few number of high bills in a sequence of relatively low mean low variation invoices. Let's assume that in such a sequence of n elements, there are k high-value (b) invoices and n-k low-value (a) invoices. Such sequences take the form of { a, a, a, a, ..., b, ..., a, a, b, ..., a}. For this sequence, the maximum-minimum difference (R), average (M), and standard deviation (S) are calculated as follows.

$$R = b - a,$$

$$\boldsymbol{M} = \boldsymbol{a} + \frac{\boldsymbol{k}}{\boldsymbol{n}}(\boldsymbol{b} - \boldsymbol{a})$$

$$\boldsymbol{S^2 = \frac{1}{n-1}\left[(n-k)\left(\frac{k}{n}(b-a)\right)^2 + k\left(b-a-\frac{k}{n}(b-a)\right)^2\right]}$$

$$\boldsymbol{S = (b-a)\sqrt{\frac{k(n-k)}{n(n-1)}}}$$

From this, it can be immediately observed that the S/R ratio is independent of the values of a and b and depends on k and n only.

Typical values for a=100, b=1000, n=11 and k=1,…,10, M,S,R and the ratio S/R are given as follows.

Table 1: Values of M, S, R, and S/R for a=100a, b=1000, n=11, =11, and k=1,…,10k

K	M	S	R		S/R
1	181.81	271.36	900		0.3015
2	263.63	364.06	900		0.4045
3	345.45	420.38	900		0.4671
4	427.27	454.07	900		0.5045
5	509.09	470.00	900		0.5222
6	590.90	470.00	900		0.5222
7	672.72	454.07	900		0.5045
8	754.54	420.38	900		0.4671
9	836.36	364.06	900		0.4045
10	918.18	271.36	900		0.3015

In Figure 4 below, we display the ratios S/R as a function of M. In this figure, data agglomerated along straight lines thus correspond to nearly binary valued invoices. Those singular high expenditures may relate to retail expenses or exceptional telecommunication expenditure such as usage in foreign countries, which are likely to be made consciously and may not lead to complaints. On the other hand, persistently high-value expenditures significantly above the average may relate to installment payments or package changes, and the likelihood of complaints for this group is also low.

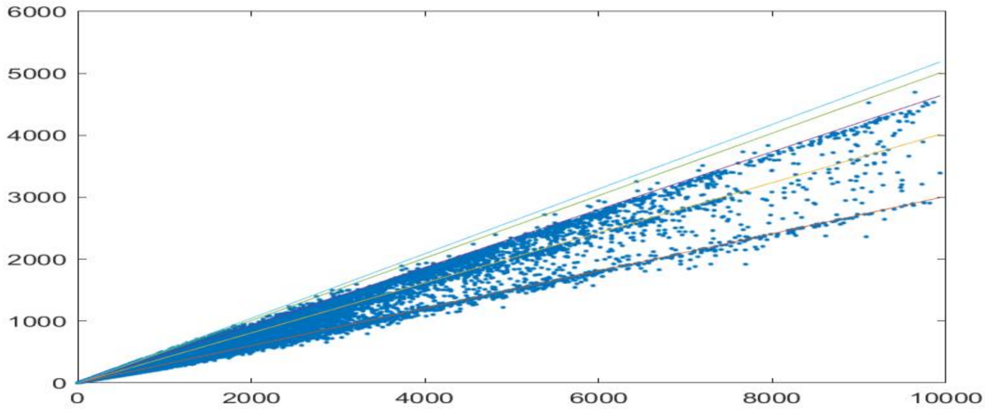

Figure 4: S/R ratios for all invoice series and linear lines with a constant slope for k=1,…,5.

Data segmentation

In the implementation of the algorithm, it is necessary to segment the data into subsets. This segmentation is obtained hierarchically, by segmenting with respect to M, then with respect to S and finally with respect to R, Sorted values of M, S, R and their ratios are given below.

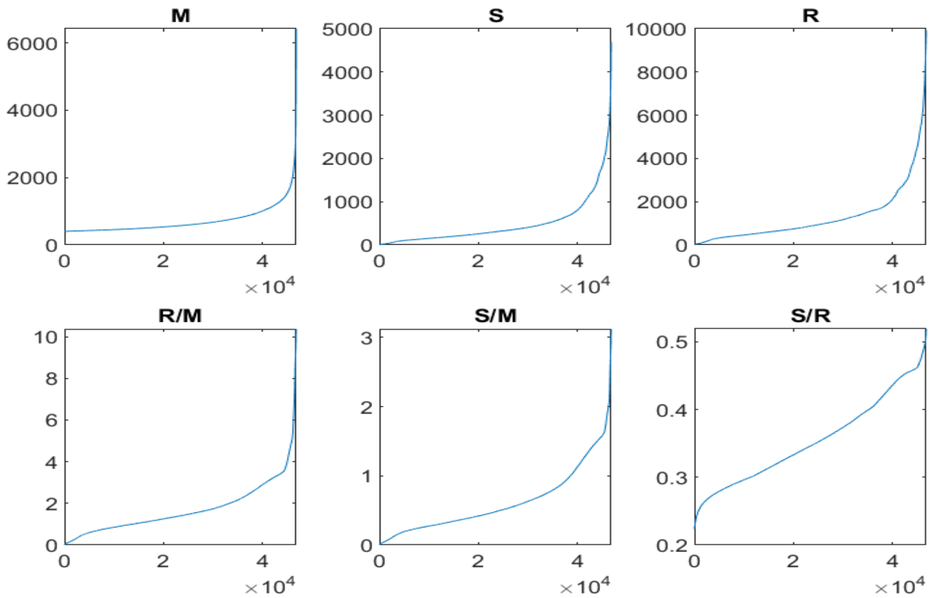

Figure 5: Sorted values of M, S, R and their ratios.

The thresholds for M are determined in terms of logarithms as shown below.

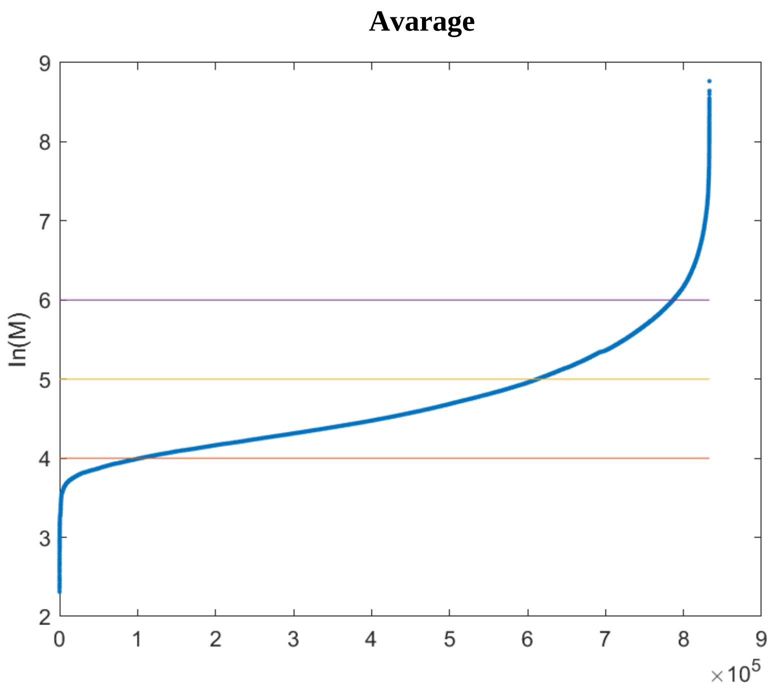

Figure 6: Limits for clustering based on the average.

The scatter plots of R and S with respect to M for each group are shown below.

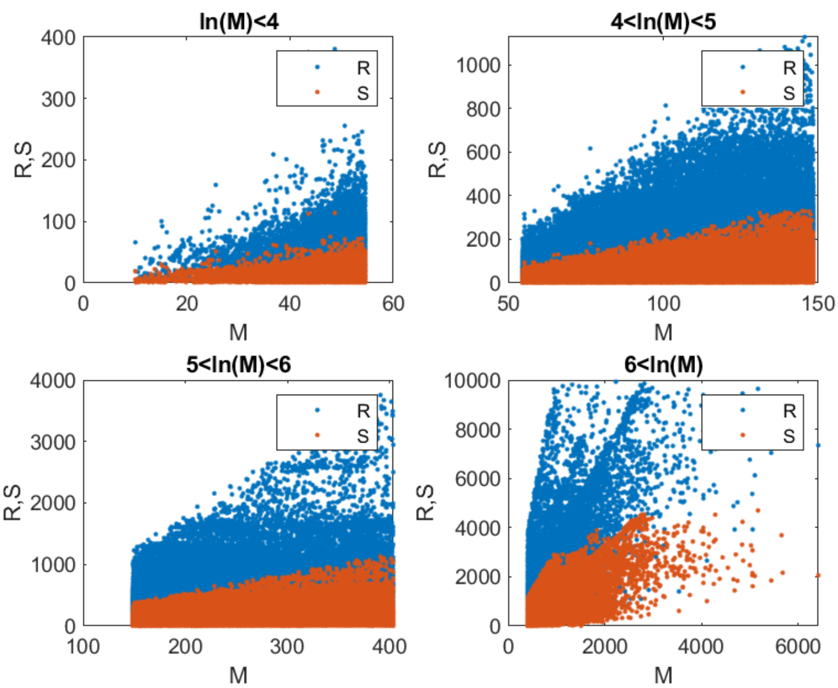

Figure 7: Distribution of standard deviation and maximum-minimum difference according to the average.

In all groups, there is a concentration along a line with a slope of around 0.29-0.30. In other groups, there is also a noticeable concentration along lines with less distinctly increasing slopes.

The monthly complaint rates for these four sections are as follows:

Table 2: Number of Complaints and Rates

	Total	ln(M)<4	4<ln(M)<5	5<ln(M)<6	6<ln(M)
Number	833 205	103 320	509 338	173 826	46 721
Number of Objections	13 816	702	7 965	4 215	934
Objection Rate	0.0015	0.000617	0.0014	0.0022	0.0018

The hierarchical clustering algorithm involves dividing the ordered variables M, S, and R into 4 sections, resulting in the creation of 64 classes (4^3 = 64). The thresholds are determined in terms of logarithmic plots as follows. The linear region in the middle corresponds to exponential growth, while the ends are sharper than exponential change regions.

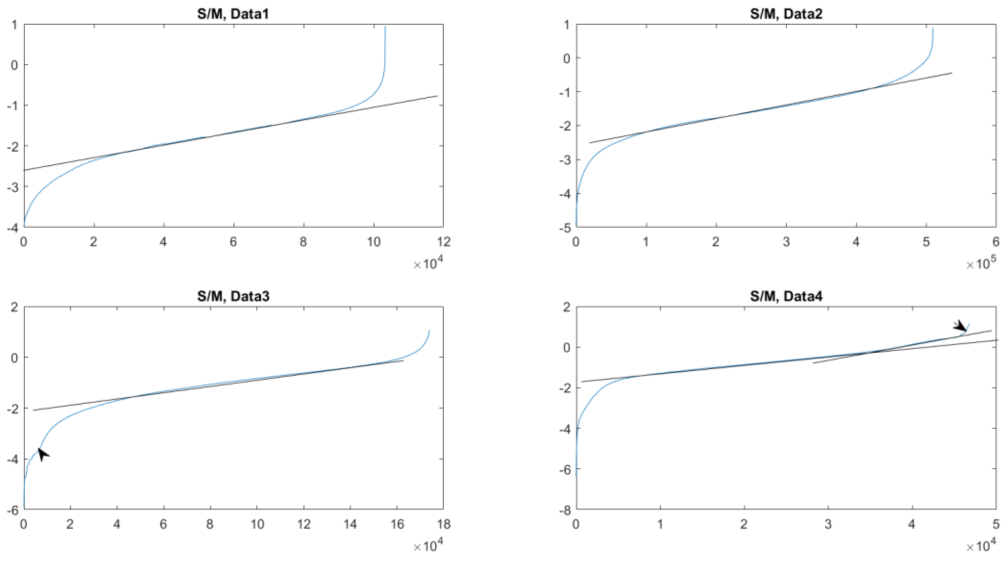

Figure 8: Clustering of data according to ln(S/M)

The linear region in the logarithm corresponds to exponential growth. The need for further subdivision into more regions has emerged for DATA3 and DATA4. The threshold values are shown below.

Table 5

DATA1.1	ln(S/M) < -2.3
DATA1.2	-2.3 < ln(S/M) < -1.2
DATA1.3	-1.2 < ln(S/M)
DATA2.1	ln(S/M) < -2.2
DATA2.2	-2.2 < ln(S/M) < -0.8
DATA2.3	-0.8 < ln(S/M)
DATA3.1	ln(S/M) < -3.6
DATA3.2	-3.6 < ln(S/M) < -1.6
DATA3.3	-1.6 < ln(S/M) < -0.2
DATA3.4	-0.2 < ln(S/M)
DATA4.1	ln(S/M) < -1.5
DATA4.2	-1.5 < ln(S/M) < -0.3
DATA4.3	-0.3 < ln(S/M) < 0.5
DATA4.4	0.5 < ln(S/M) < 0.7
DATA4.5	0.7 < ln(S/M)

The numbers of these groups, the number of objections, and the objection rates are shown below.

Table 6

	Total Number	Number of Objections	Objection Rate
DATA	833 205	13 816	0.0015
DATA 1	103 320	702	0.0006
DATA 1.1	22 252	75	0.0003
DATA 1.2	65 333	400	0.0006
DATA 1.3	15 735	227	0.0013
DATA 2	509 338	7 965	0.0014
DATA 2.1	97 989	802	0.0007
DATA 2.2	340 708	5 333	0.0014
DATA 2.3	70 641	1 830	0.0024
DATA 3	173 826	4 215	0.0022
DATA 3.1	6 682	11	0.0001
DATA 3.2	38 218	646	0.0015
DATA 3.3	109 394	3 086	0.0026
DATA 3.4	19 532	472	0.0022
DATA 4	46 721	934	0.0018
DATA 4.1	6 497	89	0.0012
DATA 4.2	27 057	489	0.0016
DATA 4.3	11 864	316	0.0024
DATA 4.4	794	29	0.0033
DATA 4.5	509	11	0.0020

4. Customer Risk Scoring Algorithm

In this section, we present an algorithm for assigning a risk score to customers by analyzing their invoice series. The data set consists of the invoice series from January 2021 to December 2023, covering a 36-month period. It contains information for 357,957 subscribers, including their subscriber numbers, monthly invoices, and complaint information.

The invoice series will be analyzed through its features: mean (M), standard deviation (S), and range (R) (maximum-minimum difference). The 36-month invoice series clearly reflects price increases during the studied period, as shown in Figure 1

Mean and Standard Deviation

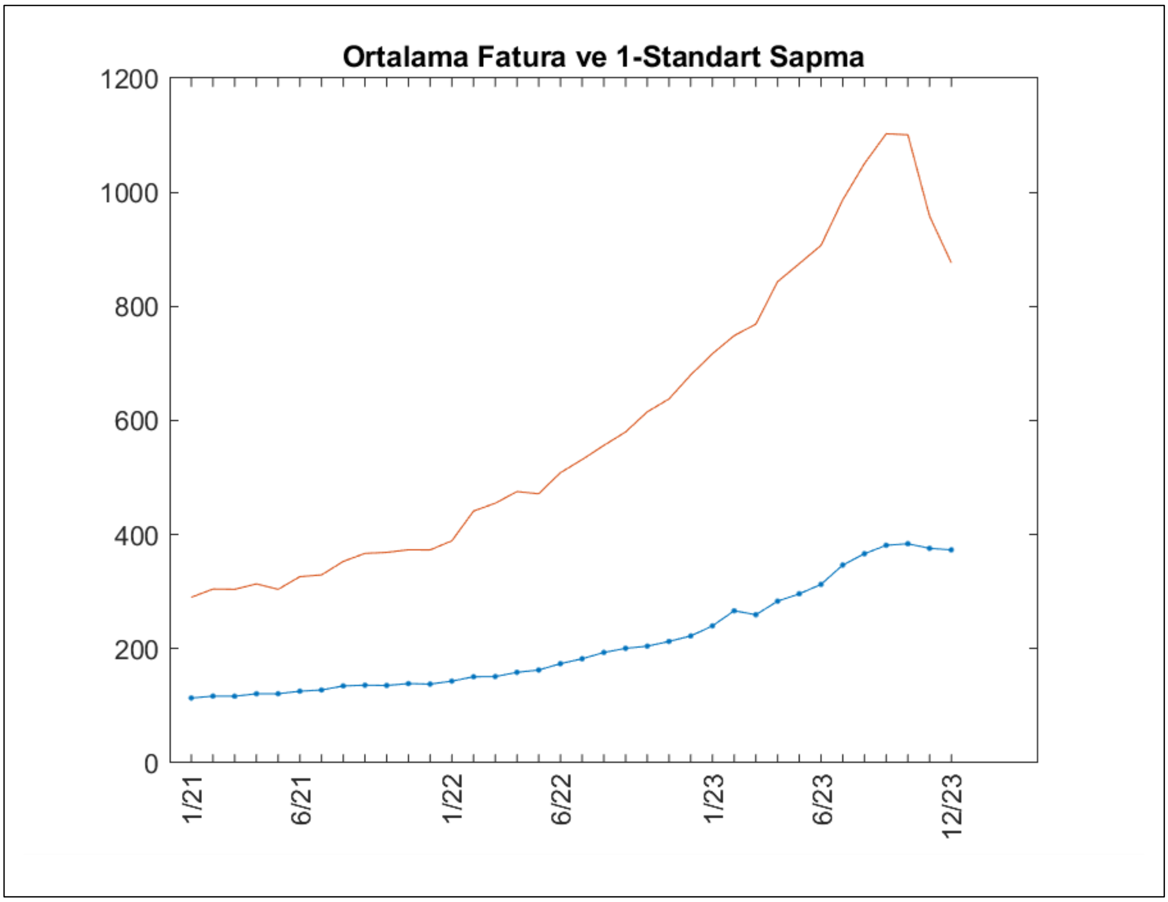

Figure 10: Monthly Average of Customer Invoices and 1-Standard Deviation Boundary.

In the studied sample, the total number of objections made over 36 months is 4702 which corresponds to a monthly objection rate of 0.0003, consistent with the rates observed in samples analyzed in the previous section.

Among the customers, there are 8 customers who have made 6-10 objections within the 36-month period. As discussed above, these situations should be treated separately. We display below their invoice sequences . As can be seen, objections do not always correspond to invoice anomalies.in Figure 11.

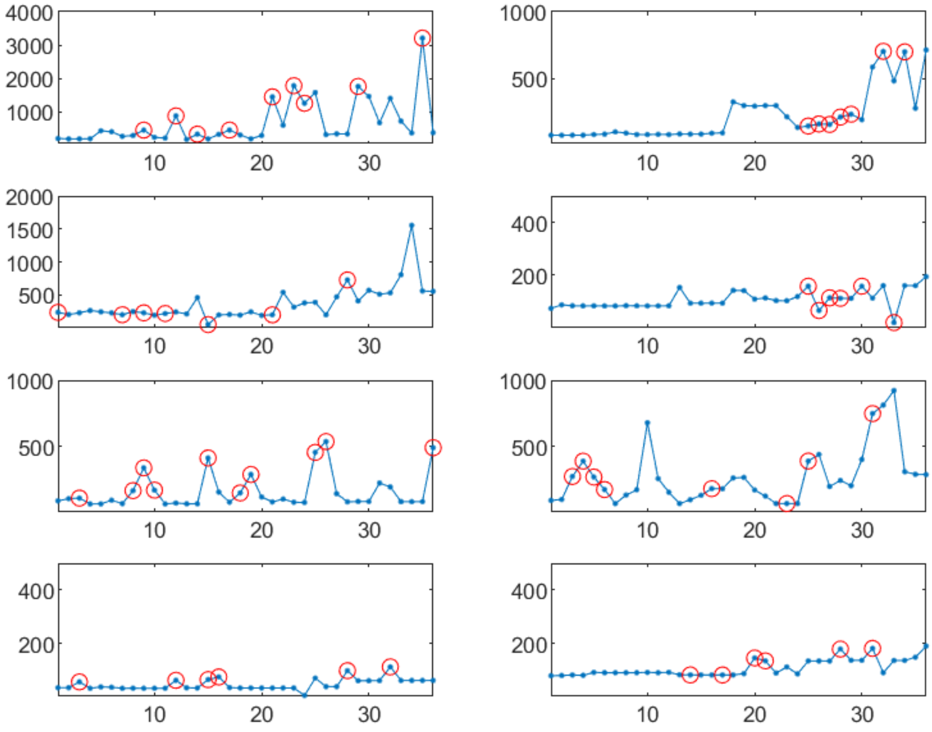

Figure 11: The 36-month invoice patterns and objections of customers who have objected six or more times.

Among the remaining objections, there are 5 customers who have objected 5 times, 12 customers who have objected 4 times, 37 customers who have objected 3 times, 208 customers who have objected 2 times, and 4,043 customers who have objected only once.

After identifying customers with a high number of objections, customers with invoices that can be considered extreme will also be distinguished. This is to characterize the situation of "rare events with significant consequences." When an expenditure or invoice above a certain amount arises and turns into an objection, the potential financial loss can be high, so these situations need to be treated separately. In **Figure 12** below, the average invoices and the invoices of each month are presented in increasing order.

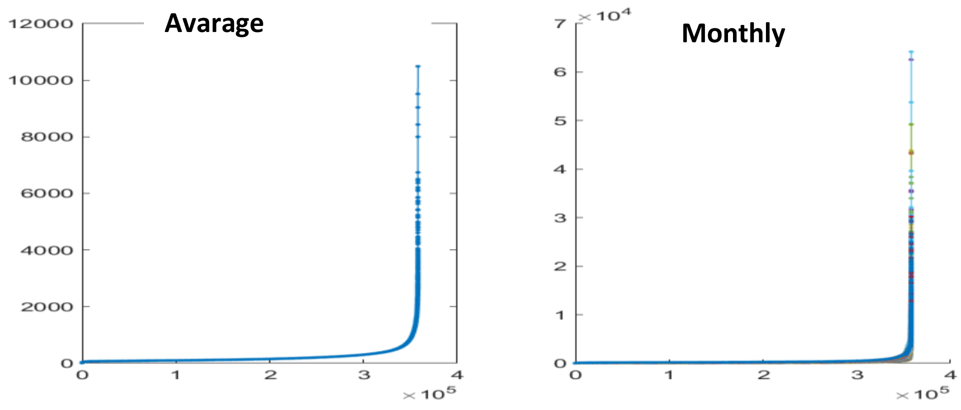

Figure 12: Increasingly ordered average invoices (left graph) and monthly invoices (right graph). It can be observed that highest values of average invoices is around 10,000 TL, while highest monthly invoices are around 50,000-60,000 TL.

Customers with extreme expenditures were distinguished in two stages: those whose invoice in any given month was 100 and 50 times the average invoice of all customers for that month. In the first group, there are 36 customers, and in the second group, 663 customers. As high invoices are removed from the sample, the average will also change, so this process must be repeated. For the second group, three iterations were required.

For the remaining customers, considered part of the general group, the average and standard deviation of the past 6 months' invoice series were calculated. **Figure 13** below shows these curves as an example for a customer who has objected 5 times.

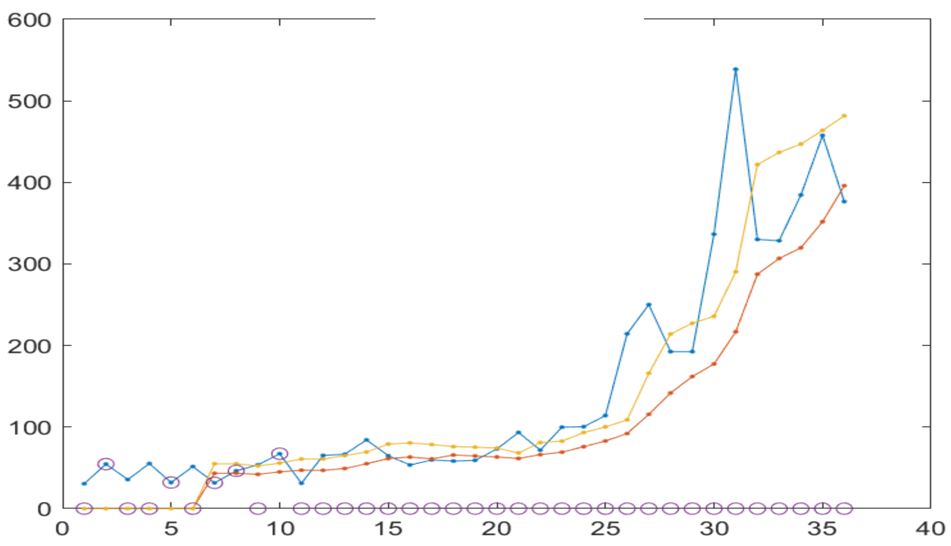

Figure 13: Invoice series (blue), 6-month moving average (red), and 6-month standard deviation (yellow) for a customer who has objected five times.

In **Figure 14**, a similar graph is provided for a customer who has never objected.

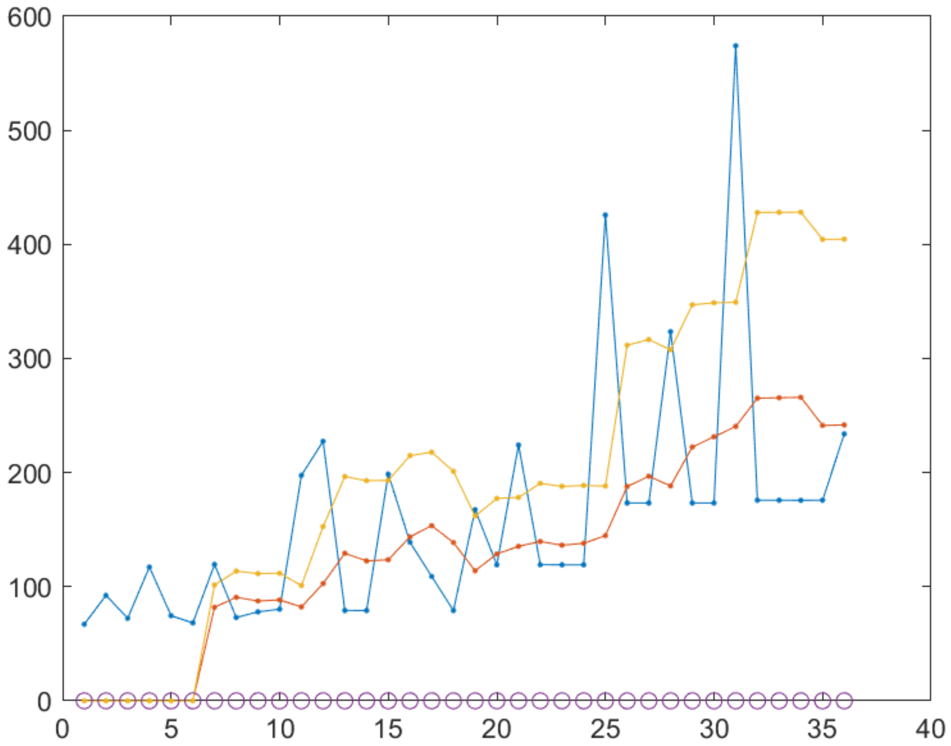

Figure 14: Invoice series (blue), 6-month moving average (red), and 6-month standard deviation (yellow) for a customer who has never objected.

After calculating the 6-month moving average and standard deviation for the entire data set, an "ANOMALY VECTOR" is created for each customer. In this vector, the criteria previously used in the LOGISTIC REGRESSION analysis is adopted: if the difference between the invoice of the (n+1)th month and the invoice of the nth month is *d*, and *d* is 50% above the 6-month average, it is defined as an anomaly. This criterion captures sudden spikes. On the other hand, to separate consecutive high invoices, the standard deviation criterion is required. If the ratio of *d* to the standard deviation is greater than 1, the invoice for that month is defined as an anomaly.

It should be noted that the ratio of standard deviation to the average must be used carefully when constructing these criteria. In sequences with very low standard deviation, this criterion may flag very small invoice fluctuations as anomalies.

The algorithm described below is a simple offline, statistical observation algorithm. After the anomaly vector is created, a RESPONSE vector to the anomaly is constructed. This vector consists of:

0 True Negative: No anomaly, no complaint

1 False Positive Anomaly present, no complaint

2 True Positive Anomaly present, complaint present

3 False Negative No anomaly, complaint present

After obtaining the response vector, a SCORE vector is created, which determines an action for each type of RESPONSE. The SCORE initially starts at a neutral value of 1. In cases of no anomaly or no response to an anomaly, the score is decreased by a certain percentage. On the other hand, if there is a complaint, the score is increased. Specifically, score reduction rules of 0.95 and 0.8 are chosen, and score increase rules of 1 and 2 are applied to derive a score vector for each cus[illegible]his scoring is shown in Figure 6 below

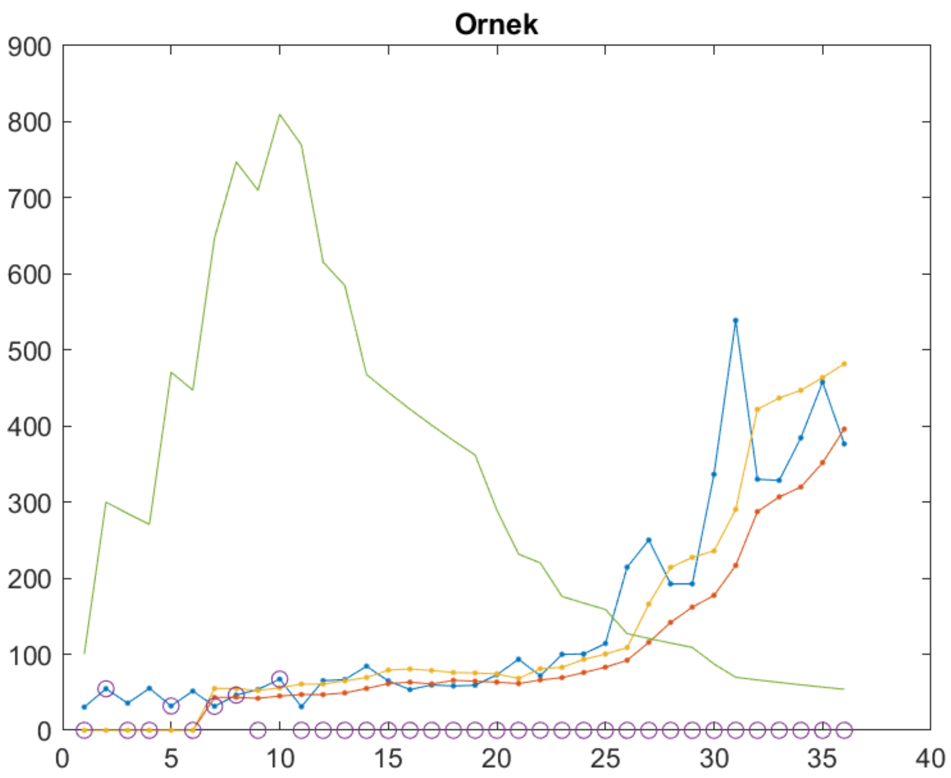

Figure 15: Increase in scores due to invoice complaints (the score value is plotted multiplied by 100).

In the example given above, the consecutive complaints that were initially unrelated to anomalies significantly raised the score, but later, as there were no complaints, the score decreased.

It should be noted that, with the simple algorithm provided above, complaints are predicted correctly at a rate of 50%, while false positives remain around 18%. In our other machine learning algorithms, by using data augmentation techniques, the correct prediction rate for complaints can reach up to 80%. However, the goal of the customer complaint score here is to manage false positives and reduce unnecessary alerts.

The above algorithm assumes that it is known in real-time whether there was a complaint in the month when an anomaly occurred. In actual practice, this is not true, and the code should be modified so that when each type of anomaly is first encountered, the score increases; if there is a complaint, it should be added to the observation list, and if there is no complaint, it should decrease.

Integrating machine learning algorithms into the code has not been done at this stage, but it is necessary for the commercialization phase.

5. Methodology for customer scoring

The methodology that is described in this section aims to provide a tool for associating customer responses to anomalies in a time series. The time series to be analyzed has to be long enough to give statistically meaningful results, but one has to ensure that the series is stationary throughout the observation period. We note that the method involves actually 2 time series. The first one is the sequence of monthly invoices and the second one is the time series for objections. The time series for invoices can be detrended and prices can be normalized as a preprocessing. On the other hand, changes in the nature of the expenditures may lead to non-stationarity in the sequence of invoices. A typical example for this is retail expenditures via telecommunication bills. Detection of retail expenditures has been studied in [13], where it was shown that anomalies reflecting retail expenditures were not likely to be associated with complaints. This study has led to a clustering of invoice series, which also constituted a basis for under sampling of the majority class in [14]. In the present work we assume that the time series of invoices is stationary. Non-stationarity in the time series of customer complaint is less likely, as it would correspond to a change in the personality of the customer. At any rate, in designing a customer scoring system one has to ensure that both time series are stationary, and provided this is satisfied, the longer the observation time, the higher will be the reliability of the score. In the present work for illustration purposes, we consider a time window of 12 months.

An added value of customer scoring system is the possibility of incorporating personal checkpoints into the anomaly detection system. For example, the monthly bill usually increases at times of renewing contracts; these increases appear as an anomaly, but customers are normally aware of these changes and such anomalies, normally, should not be associated with complaints. On the other hand, it has been observed that there is a non-negligible proportion of complaints associated with contract renewals. Reaching and informing all customers every year for contract removal would be a costly solution to this problem. Running a customer scoring over multiple contract renewing periods would distinguish between customers that are conscious about contract renewing procedures and the ones that that have difficulties in handling these changes. In addition, certain services are prone to complaints, hence each time any of these services is used, the corresponding invoice is labeled by rule-based risk types. The scoring system consists of the following steps.

Step 1. Rule based risk assessment of the invoice: A risk score is assigned to each invoice based on certain rules.

P (invoice with no risk) or Q_i (invoice with risk of type i).

Step 2. Anomaly assessment of the invoice: An anomaly detection system is applied to past time series to classify the invoice, as

A (displays no anomaly) or B_i (displays anomaly of type i).

As a result, one obtains a time series of risk and anomaly labels for each customer, as shown below for a typical period of 12 months.

Table 7

Month	1	2	3	4	5	6	7	8	9	10	11	12
Rule-based risk	P	P	P	Q_1	P	P	P	Q_2	P	P	P	P
Anomaly type	A	A	A	A	B_1	A	A	A	B_3	A	A	A

In this example, Q_1 may represent the contract renewal time and Q_2 may represent using a service that leads to complaints.

Step 3. Watch list scoring for rule-based risks: For each rule-based risk Q_i, the risk score SQ_i is computed as follows. SQ_i is initially set to 1 and the customer is in the watch list for the risk Q_i, Each time there is a complaint for that risk the risk score is increased by a factor k_u, and each time the is no complaint for this risk, the risk score is decreased by a factor k_d. If the risk score falls below a predetermined threshold, the customer is removed from the watch list.

In the following, complaints to rule-based risks will not be included in the evaluation of risk score with respect to anomalies.

Step 4. Complaint tracking: Objections are incorporated to the system by appending rows below to the table for invoice characteristics, as below. If there is an objection to a rule base risk, the customer is added to a watch list for that rule based risk.

Table 8

Month	1	2	3	4	5	6	7	8	9	10	11	12
Anomaly type	A	A	A	A	B_1	A	A	A	B_3	A	A	A
Complaint t=0	0	0	1	0	0	0	0	1	1	0	0	0

Here the customer has objected to an invoice with no anomaly at months 3 and 8, and to an anomaly of type B_3

at month 9.,

Step 5. Lagged complaints. It has been observed that customer complaints to an invoice may lag up to 4 months. Lines are added below the table to display lagged complaints, as below.

Table 9

Month	1	2	3	4	5	6	7	8	9	10	11	12
Anomaly type	A	A	A	A	B_1	A	A	A	B_3	A	A	A
Complaint t=0	0	0	1	0	0	0	0	1	1	0	0	0
Lag 1	0	0	0	1	0	0	1	0	0	0	0	-
Lag 2	0	0	0	0	0	0	0	0	1	0	-	-
Lag 3	0	0	0	0	0	0	0	0	0	-	-	-
Total	**0**	**0**	**1**	**1**	**0**	**0**	**1**	**0**	**2**	**-**	**-**	**-**

This matrix shows all complaint statistics, including the information on the invoices to which the customer objected, how many times and with what lags. The statistics of objections are given below.

Table 10

	A	B1	B2	B3
Number of Invoices	7/9	1/9	0	1/9
Objection Rate	3/7	0	0	2/1
Multiple objections	0	0	0	1
Late objections	1/3	0	0	1/1

Customer score for each type of objection is evaluated as below. At the beginning the scores for each anomaly type are set to 1. Each time there is no objection the score is reduced by a factor k and each time there is an objection the score is increased by a factor k. Assuming that there are no objections at the last 3 months, and using the aggregate number of objections, we obtain the following customer scoring for each anomaly type.

Table 11

Month	1	2	3	4	5	6	7	8	9	10	11	12
Anomaly type	A	A	A	A	B_1	A	A	A	B_3	A	A	A
Total	**0**	**0**	**1**	**1**	**0**	**0**	**1**	**0**	**2**	**0**	**0**	**0**
Score for A	1	1/k	1	1*k	1*k	1	1*k	1	1	1/k	$1/k^2$	$1/k^3$
Score for B_1	1	1	1	1	1	1	1	1	1	1	1	1
Score for B_3	1	1	1	1	1	1	1	1	$1*k^2$	$1*k^2$	$1*k^2$	$1*k^2$

6. Concluding Remarks

In this work, we proposed a methodology for preprocessing imbalanced data with the aim of detecting rare events. We recall that in the application of all machine learning algorithms, data is split into training and test sets and one can preprocess the training set in order to improve the performance of the algorithm. With the purpose of detecting rare events, described in terms of minority and majority classes, oversampling of the minority class is a well-known and successful technique. On the other hand, preprocessing of the majority class is usually restricted to random under-sampling.

In this work we propose a decomposition of the majority group into subsets, based on the similarity of the items in each set. The similarity is determined by Principal Component Analysis, the similarity measure being the percentage of the variance explained by the largest eigenvalue of the covariance matrix, denoted as PCA_1.

This methodology is applied to the complaints to a series of invoices in telecommunication sector, where the events to be detected, i.e., the minority class is about 10^{-4} of the total population. The clustering of the training set is based on extracting homogeneous subsets, based on the range, the standard deviation, the mean and other application specific features of the data.

The preprocessing scheme that we propose is random under-sampling from each cluster at different selection rates. For example, fewer samples will be selected from clusters with higher PCA_1 values. As items in these clusters are very similar to each other, selecting too many of these will not add any new information to the models, but it will increase the computational power needed, also increasing the running time of the models. For the clusters with lower PCA_1 values, we propose selecting more observations to represent the variability of the items. Finally, if a cluster does not have enough observations to start with, oversampling algorithms like SMOTE can be used to enhance the number of observations. Application of the method proposed here and the investigation of its performance in terms specificity and sensitivity and run-time improvement is planned for future work.

References

[1] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, “A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches”, IEEE Trans. Syst. Man Cybern. – Part C, 42 (4), 463–484, 2012

[2] C. Beyan and R. Fisher, “Classifying imbalanced data sets using similarity-based hierarchical decomposition,” Pattern Recognition, vol. 48, no. 5, pp. 1653–1672, 2015.

[3] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Trans. Syst. Man Cybern. – Part C 42 (4) (2012) 463–484.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: Synthetic minority over-

sampling technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.

[5] Azhar, N.A., M.S.M. Pozi, A.M. Din, and A. Jatowt. "An Investigation of SMOTE Based Methods for Imbalanced Datasets with Data Complexity Analysis." IEEE Transactions on Knowledge and Data Engineering, Knowledge and Data Engineering, IEEE Transactions on, IEEE Trans. Knowl. Data Eng 35, no. 7 (July 1, 2023): 6651–72. doi:10.1109/TKDE.2022.3179381.

[6] Jason Van Hulse, Taghi M Khoshgoftaar, and Amri Napolitano. "An empirical comparison of repetitive undersampling techniques." In Information Reuse & Integration, 2009. IRI'09. IEEE International Conference

on, pages 29–34. IEEE, 2009.

[7] Hasanin, T., & Khoshgoftaar, T. (2018). "The Effects of Random Undersampling with Simulated Class Imbalance for Big Data." 2018 IEEE International Conference on Information Reuse and Integration (IRI), Information Reuse and Integration (IRI), 2018 IEEE International Conference on, IRI, 70–79. https://icproxy.khas.edu.tr:2071/10.1109/IRI.2018.00018

[8] C. Seiffert, T. Khoshgoftaar, J. Van Hulse, A. Napolitano, "RUSBoost: a hybrid approach to alleviating class imbalance", IEEE Trans. Syst. Man Cybern. – Part A 40 (1), 185–197, 2010

[9] R. Barandela, R.M. Valdovinos, J.S. Sanchez, "New applications of ensembles of classifiers", Pattern Anal. , Appl. 6 ,245–256, 2003.

[10] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, "Credit Card Fraud Detection Using AdaBoost and Majority Voting", IEEE Access, Vol. 6, pp. 14277–14284, 2018.

[11] M. Zareapoor and P. Shamsolmoali, "Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier", Procedia Computer Science, Vol. 48, pp. 679–685, 2015.

[12] J. O. Awoyemi, A. O. Adetunmbi, and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis", In: Proc. of 2017 International Conference on Computing Networking and Informatics (ICCNI), pp. 1–9, 2017.

[13] Bilge, A. H. ., Ogrenci, A. S. ., Carpanali, H. ., Aktunc, E. A. ., Atas, F., Ozmen, T. ., & Kaya, B. E. . (2022). Detection of Expenditure Trends in the Telecommunication Sector. American Scientific Research Journal for Engineering, Technology, and Sciences, 90(1), 340–350.

[14] Bilge, A., Ogrenci, A. S. Carpanali, H., Ozmen, T, Tosun, A ., Cakar, K. (2023) Detection of Rare Events: Cluster Based Preprocessing of the Training Set: The Case on Complaints for Invoice Time Series