OKÜ Fen Bilimleri Enstitüsü Dergisi
7(2): 834-854, 2024

OKU Journal of The Institute of Science and Technology, 7(2): 834-854, 2024

Osmaniye Korkut Ata Üniversitesi
Fen Bilimleri Enstitüsü
Dergisi

Osmaniye Korkut Ata University
Journal of The Institute of Science
and Technology

Crucial Challenges in Corporate Credit Risk Assessment: A Case Study

Btissam HAJJAOUI[1*]

[1]Kadir Has University, Department of Engineering and Natural Sciences, Industrial Engineering, 34083, Istanbul

[1]https://orcid.org/0000-0003-0851-3428
*Corresponding author: b.hajjaoui@stu.khas.edu.tr

Research Article

Article History:
Received: 10.08.2023
Accepted:11.01.2024
Published online: 11.03.2024

Keywords:
Credit risk assessment
Corporate client
Missing values
Subset creation
Sampling methods
Machine learning

ABSTRACT

This article demonstrates three crucial challenges that can be faced when dealing with credit risk datasets through a case study based on a dataset obtained from one of the leading institutions in the finance sector in Turkey. These datasets have many variables, numerous missing values, and an unbalanced nature. This study shows the step-by-step solutions we used to overcome the three mentioned challenges. Furthermore, predicting whether a borrower has not repaid their loan on time was also part of the study. In this case study, we initially had a large number of variables, which was 401 variables. We reduced this number by identifying the input variables from the others and then studying those inputs to avoid using strongly correlated variables and variables consisting almost entirely of missing or zero values. To solve the issue of missing values, we created seven subsets from our dataset to reflect which group of variables relates to which customer. To overcome the issue of the imbalanced nature of the dataset (96% and 4% non-default and default instances, respectively), we used three sampling techniques to balance the instances in the training sets. Subsequently, we applied six simple classifiers to predict the output variable. As a result of this study, we discovered that most of the variables initially present in the dataset were unnecessary and insignificant. Besides, we found answers to why we had many missing values, which helped us realize that not all variables relate to all customers and helped us deal with missing values effectively. Finally, for the default predictions, we simultaneously achieved sensitivity and specificity above 50%, where the under-sampling technique was the best sampling technique for the minority class, and the synthetic minority oversampling technique and oversampling performed better for the majority class.

Kurumsal Kredi Riski Değerlendirmesinde Önemli Zorluklar: Bir Vaka Çalışması

Araştırma Makalesi

Makale Tarihçesi:
Geliş tarihi: 10.08.2023
Kabul tarihi: 11.01.2024
Online Yayınlanma: 11.03.2024

Anahtar Kelimeler:
Kredi riski değerlendirmesi
Kurumsal müşteri
Kayıp değerler
Alt küme oluşturma
Örnekleme yöntemleri
Makine öğrenimi

ÖZ

Bu makale, Türkiye'de finans sektörünün önde gelen kuruluşlarından birinden elde edilen bir veri setini temel alan bir vaka çalışması aracılığıyla, kredi riski veri setleriyle uğraşırken karşılaşılabilecek üç önemli zorluğu ortaya koymaktadır. Bu veri kümeleri çok sayıda değişkene, çok sayıda eksik değere ve dengesiz bir yapıya sahiptir. Bu çalışma, bahsedilen üç zorluğun üstesinden gelmek için kullandığımız çözümleri adım adım göstermektedir. Ayrıca, borçlunun kredisini zamanında geri ödeyip ödemediğini tahmin etmek de çalışmanın bir parçasıydı. Bu vaka çalışmasında başlangıçta çok sayıda değişkenimiz vardı, bu da 401 değişkendi. Diğerlerinden girdi değişkenlerini belirleyerek ve ardından güçlü korelasyona sahip değişkenleri ve neredeyse tamamen eksik veya sıfır değerlerden oluşan değişkenleri kullanmaktan kaçınmak için bu girdileri inceleyerek bu sayıyı azaltılmaktadır. Eksik değerler sorununu çözmek için, hangi değişken grubunun hangi müşteriyle ilgili olduğunu yansıtacak şekilde veri kümemizden yedi alt küme oluşturulmuştur. Veri setinin dengesiz doğası sorununun (sırasıyla yaklaşık %96 ve %4 varsayılan olmayan ve varsayılan örnekler) üstesinden gelmek için eğitim setlerindeki örnekleri dengelemek amacıyla üç örnekleme tekniği kullanılmış. Daha sonra çıktı değişkenini tahmin etmek için altı basit sınıflandırıcı uygulanmıştır. Bu çalışma sonucunda başlangıçta veri setinde bulunan değişkenlerin çoğunun gereksiz ve önemsiz olduğunu keşfedilmiş. Ayrıca, neden birçok eksik değere sahip olduğumuza dair yanıtlar bulunmuş. Bu, tüm değişkenlerin tüm müşterilerle ilgili olmadığını anlamamıza ve eksik değerlerle etkili bir şekilde başa çıkmamıza yardımcı olmuş. Son olarak, varsayılan tahminler için aynı anda %50'nin üzerinde duyarlılık ve özgüllük elde edilmiş; burada düşük örnekleme tekniği azınlık sınıfı için en iyi örnekleme tekniğiydi ve sentetik azınlık aşırı örnekleme tekniği ve aşırı örnekleme çoğunluk sınıfı için daha iyi performans göstermiştir.

To Cite: Hajjaouı B. Crucial Challenges in Corporate Credit Risk Assessment: A Case Study. Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi 2024; 7(2): 834-854.

1. Introduction

As more and more people apply for different types of loans, including personal, student, business, vehicle, home, etc., banks and financial institutions want to ensure that the loans given to their customers or applicants will be repaid on the due date. Before approving a loan for a credit applicant, the credit risk associated with the applicant should be assessed so that the lender's business is not adversely affected. Credit risk can be defined as the probability of loss due to a borrower's default (Weissova et al., 2015). Credit risk assessment is critical for banks in managing and deciding on credits (Abdou and Pointon, 2011). This evaluation involves many operations, including gathering, studying, and classifying variables and credit-related elements (Abdou and Pointon, 2011).

In this article, we focus exclusively on corporate clients and do not consider non-corporate clients. For corporate clients, the objective is to assess a company's creditworthiness, while for non-corporate clients, the aim is to determine an individual's creditworthiness. Thus, the variables used in credit risk assessment differ from one type of customer to another, although some variables may be used independently of the kind of customer.

The main objective of this work is to build models that can help banks and financial institutions measure credit risk. In this article, we answer some questions about how the credit risk of corporate clients can be assessed and to what extent this risk can be measured. Credit evaluation generally depends on comparing the characteristics of new applicants with those of borrowers who have repaid their loans (Abdou and Pointon, 2011). The loan is refused if the features are similar to those of previously defaulting borrowers (Abdou and Pointon, 2011). Otherwise, if these characteristics are identical to those of borrowers who have not defaulted, the loan should be granted (Abdou and Pointon, 2011).

As we have many variables, 401 variables, we propose to classify the variables before using any machine learning classifier to avoid using variables resulting from the credit application evaluation that can influence the default predictions. We generally classify the variables into input, output, information, and irrelevant, and then further study the 127 input variables, which can be used as features to assess credit risk.

We also distinguish variables with accidentally missing values from those with non-accidentally missing values. This differentiation is essential because most missing values in our dataset do not occur accidentally and indicate a negative or positive situation, depending on the context. Therefore, we suggest separating the customers according to the types of variables available for them by creating seven subsets with no missing values. For variables that occur accidentally, we fill them with a meaningful value whenever possible.

Additionally, we use two simple methods to reduce the number of input variables. The first method is to analyze the variables in terms of missing and zero values and remove those that are nearly empty or mostly filled with zeros. The second method involves studying the correlation between the variables and removing a strongly correlated variable out of two.

We have a binary classification problem, and we work to predict whether the client will default after their loans are approved "1" or not "0". Due to the unbalanced nature of the dataset, we use under-sampling, oversampling, and synthetic minority oversampling techniques to balance the number of instances of the two classes in the training sets.

We will make the default predictions based on six well-known classifiers in the literature: Random Forest, Naïve Bayes, Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbor.

This article demonstrates the challenges encountered in processing and using credit datasets and proposes general methods for overcoming these challenges. Although the variables provided or available to assess credit risk may vary from country to country, this work is still relevant and valid because we treat variables as four groups of data that should be globally accessible: the applicant's data based on the information submitted with the credit application, corporate data, shareholder data, and credit history within the creditor's institution.

This article is composed of five sections. Here is the article's structure: Section 2 is devoted to providing a description of the dataset and explaining in detail all the techniques used to reach the primary purpose of this work; Section 3 shows the results; Section 4 discusses and interprets the results; and finally, Section 5 summarizes the work done in this article and suggests further possible research work on the same type of dataset.

2. Data and Methods

2.1. Dataset description

The dataset used in this article is a corporate credit dataset from a Turkish financial institution. The data was shared with us subject to the Capital Markets Board of Turkey (SPK) regulations and internal institution rules. The data was stored on the institution’s server, and no sensitive data was used in our analysis that could lead to a conflict with the Personal Data Protection Authority (KVKK).

The dataset consists of 401 variables and 244300 rows. Each row refers to a credit application between January 2012 and February 2023, regardless of whether the credit was approved, rejected, or pending. Pending applications refer to credit applications undergoing financial analysis, being evaluated by the underwriter, those canceled by the applicants or the financial institution, applications not processed, etc. The dataset mainly contains four categories of variables: information, input, output, and irrelevant. Variables classified as information (30 variables) are mostly dates or identifiers of applications, customers, dealers, and shareholders. Irrelevant variables, 208 variables, are initially excluded for several reasons, such as these variables being 100% empty or containing a single category or value. On the other hand, the input variables, 127 variables, include the applicant's data, corporate data, shareholder data, and the applicant's credit history within the creditor's institution.

- *Applicant’s data (11 variables):* refers to information relating to corporate clients provided by the applicant when applying for credit, such as the firm’s activity tenure, firm’s type (cooperative, sole proprietorship, partnership, etc.), firm’s sector (automotive, energy, health, education, tourism, construction, etc.), and amount of credit requested.
- *Corporate data (59 variables):* can be obtained from the Central Bank of the Republic of Turkey (TCMB) and is in the form of a table called MEMZUÇ. This table contains variables referring to the risk, limit, and overdue amounts of the company applying for a loan.
- *Shareholder data (34 variables):* can be received from the Credit Registration Office (KKB); for instance, credit card information, scores, customer indebtedness index, loan amounts in different credit products, limits, risks, and delays. It should be mentioned that KKB data is intended for individuals and not companies; in our case, the individuals are the shareholders of the company (corporate client).
- *The applicant’s credit history within the creditor’s institution (23 variables):* refers to the current total loan amount, average loan amount at different time intervals, number of days past due, overdue amounts, etc. If the applicant is not a regular customer of the creditor, these records cannot be available.

Finally, we have output variables that result from evaluating the customer's credit application, such as the final decision of approving or rejecting the loan, approved loan amount, approved interest rate, number of installments, and application score.

In this work, we omit all the information and irrelevant variables. We also omit all the output variables except for the binary target variable (default or non-default) we need. In addition, we will only consider approved credit applications and customers whose default or non-default is known: 78747 out of 244300 applications. Sometimes clients' loans are not yet closed, which means that these clients still have months or years to finish repaying all of their installment loans. In such a situation, if no default had occurred by the client until the date when our dataset was retrieved, the label of the output variable is unknown because that client can still fail to pay their loan as long as the loan is ongoing. To put it differently, customers can default at any time between the loan approval date and the loan closing date. We also remove extreme values from financial variables containing monetary amounts by orders of magnitude. Therefore, we discard 368 observations and base subsequent work on 78379 rows.

2.2. Feature selection

2.2.1. Analysis of missing and zero values in each data group

Our primary issue is that we do not have a group or groups of data for some customers in several cases. However, the absence of at least one data group is significant and justified. Missing values can be due to different reasons, such as the client having just started their business and some information about their firm is unavailable at the time of the credit application. The corporate data variables may be empty for some customers if no record is found for these customers at the TCMB. Additionally, not all customers have a credit history with the financial institution. In other words, if this is a first application, we cannot have the institutional data concerning the current or past loan amount, the arrears of the last month, the last six, or the last 12 months.

We analyze the number of missing values for each data group and eliminate variables with too many missing values. In addition, we eliminate the variables with too many zero values. Indeed, many variables have zero values, and zero as a value is significant in our dataset because it can mean that a customer is not late in paying their loans, which is the case for many customers, or can mean that a customer has no amount of risk, and so on. Thus, some variables may contain tens of thousands of zero values. The main objective of this analysis is to include in our models mainly the variables that work for most customers from each data group. Although other variables with too many missing values may be relevant for some customers, they only concern a minority.

- *Applicant's data group analysis*

The applicant's data group contains 11 input variables. Among these 11 variables, there are 5 variables with no missing values and 6 variables with some missing values. The replacement of missing values was possible with some assumptions for 5 variables, but for the sixth variable, we decided to ignore it because it has a percentage of 93.23% of missing and zero values, which means that it is a poor predictor.

- *Corporate data group analysis*

As stated earlier, the corporate data group has 59 input variables. Figure 1 shows two histograms. The first histogram shows the distribution of percentages of missing values only. The second histogram refers to the distribution of percentages of missing and zero values.

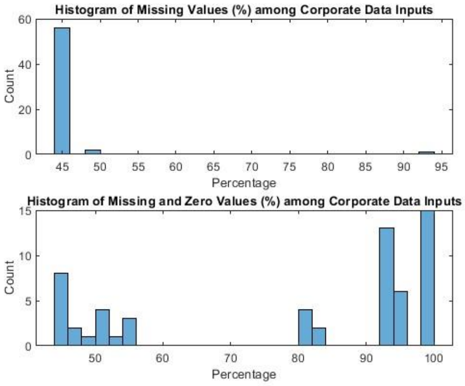

Figure 1. Distribution of the percentages of missing and zero values (corporate data)

From the first plot, we can see that most variables have about 45% missing values; then we exclude any variable with missing values greater than 45%. The second graph shows that many variables are almost filled with zeros or are empty; thus, they cannot be significant. Therefore, we exclude all variables with more than 96% missing or zero values.

- *Shareholder data group analysis*

Shareholder data, as a group, initially consists of 34 input variables. Figure 2 shows the histograms as in Figure 1, but considers only the shareholder data group.

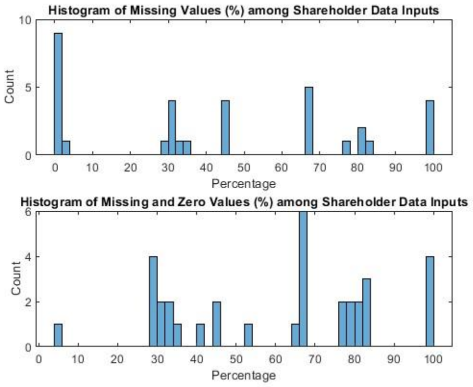

Figure 2. Distribution of the percentages of missing and zero values (shareholder data)

From the first plot in Figure 2, we can see that the variables in this group have different percentages of missing values, which makes it difficult to set a threshold to select or overlook variables. We decided to neglect all variables with more than 40% missing values and those with more than 99% missing and zero values.

- *Credit history data group analysis*

This data group contains 23 variables and has at least 61% missing values for all the variables in this group. And this high percentage was expected because not every credit applicant is a regular customer of the Turkish financial institution. Figure 3 refers to the histograms for this credit history data group in terms of percentages of missing and zero values.

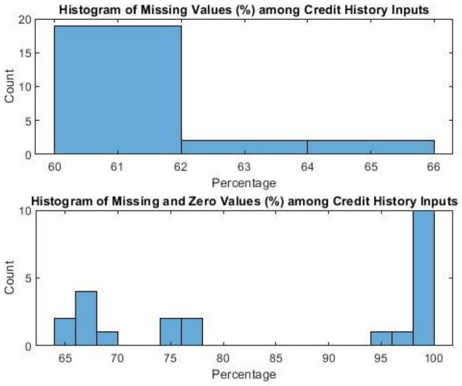

Figure 3. Distribution of the percentages of missing and zero values concerning (credit history)

From Figure 3, the percentages of all missing values for the variables in this group are between 60 and 66%. That means the percentages are very close to each other, so we are not ruling any out. However, when we check the second histogram, we find that some variables are mostly filled with zeros and missing values, so we ignore all variables with more than 97% missing and zero values. In total, we exclude ten variables from this data group.

2.2.2. Correlation analysis

To study the extent to which the variables are correlated, we grouped the input variables (81 remaining after the previous analysis) into 52 subgroups according to their meanings. In other words, we mainly grouped variables that refer to the same thing with a difference in time intervals. We studied the correlation for the subgroups X1 to X19 and ignored the subgroups from X20 to X52 because these subgroups consist of a single variable. Table 1 shows the minimum and maximum correlation coefficients in each subgroup, rounded to three decimal places.

Table 1 shows that the subgrouping was significant since, in most cases, the minimum correlation coefficient within a subgroup was greater than 0.9. As we aim to select the features that can be good predictors, we decided to eliminate certain variables in each subgroup if the correlation coefficient is 0.9 or more between two variables in the same subgroup. We keep the variable or variables with the most recent time interval from the date of application whenever possible.

Table 1. Minimum and maximum correlation coefficients in each subgroup

Data Group	Subgroup	Number of Variables	Subgroup Description	Minimum Correlation Coefficient	Maximum Correlation Coefficient
Applicant's data	X1	2	Collateral	0.637	-
Corporate data	X2	2	Average amount of cash limit at bank	1.000	-
	X3	2	Average amount of cash limit	0.998	-
	X4	3	Average amount of factoring limit	0.964	0.989
	X5	2	Average amount of foreign exchange cash limit	0.978	-
	X6	3	Average amount of leasing limit	0.967	0.996
	X7	2	Average amount of non-cash limit	0.999	-
	X8	2	Average amount of cash risk at bank	1.000	-
	X9	2	Average amount of cash risk	0.999	-
	X10	3	Average amount of factoring risk	0.956	0.988
	X11	2	Average amount of foreign exchange cash risk	0.999	-
	X12	3	Average amount of leasing risk	0.968	0.996
	X13	2	Average amount of non-cash risk	0.999	-
	X14	3	Number of banks the customer works with	0.998	0.999
Credit history	X15	4	Total amount of loan	0.877	0.977
	X16	4	Total amount of arrears	0.732	0.938
	X17	3	Average amount of loan	0.909	0.972
	X18	2	Maximum number of delays in days	0.794	-
Shareholder data	X19	2	Number of credit applications	0.820	-

2.2.3. Summary of feature selection

From the missing/zero values and correlation analysis, we can see that many input variables in the dataset are not significant, and we could reduce the number of features from 127 to 57, which means that 70 variables have been eliminated. In other words, we found that more than 55% of the variables were insignificant or at least could not be good predictors of default among approved credit applications. Table 2 refers to the number of variables (financial and non-financial) after these analyses.

Table 2. Number of input variables (financial and non-financial) in each data group

Data Group	# All Input Variables	# Financial	# Non-Financial
Applicant's data	10	4	6
Corporate data	23	21	2
Shareholder data	17	6	11
Credit history	7	5	2
Total	**57**	**36**	**21**

2.3. Subsets and default

2.3.1. Creating and selecting subsets

Although we reduced the number of variables with too many missing values, we could not avoid having missing values in our dataset because, for some customers, one or more data groups do not concern them. Thus, we propose to use subsets based on the available data group. We first create a subset using only the features from a group or groups of data. Then, we remove rows with at least one missing entry from the subsets. We applied these steps to create 16 subsets with no missing values. Table 3 shows the number of rows cleared of missing values for each of the sixteenth subsets.

Table 3. Number of rows in each subset with no missing values

Subset	Case	Subset	# Rows
1	0000	-	0
2	0001	Applicant's data	78379
3	0010	Credit history	28151
4	0011	Credit history + Applicant's data	28151
5	0100	Shareholder data	45564
6	0101	Shareholder data + Applicant's data	45564
7	0110	Shareholder data + Credit history	7889
8	0111	Shareholder data + Credit history + Applicant's data	7889
9	1000	Corporate data	43188
10	1001	Corporate data + Applicant's data	43188
11	1010	Corporate data + Credit history	8193
12	1011	Corporate data + Credit history + Applicant's data	8193
13	1100	Corporate data + Shareholder data	36595
14	1101	Corporate data + Shareholder data + Applicant's data	36595
15	1110	Corporate data + Personal data + Credit history	6844
16	1111	Corporate data + Shareholder data + Credit history + Applicant's data	6844

From Table 3, we can conclude that we may have 16 subsets, but not all are significant. The applicant's data group does not decrease the number of rows when combined with other data groups. Indeed, we can ignore cases 3, 5, 7, 9, 11, 13, and 15 and maintain cases 4, 6, 8, 10, 12, 14, and 16. The first case is excluded because it does not contain any data groups. In addition, we will not use the second subset because it considers the applicant's data only, which is insufficient to decide about any applicant, meaning we have seven possible combinations of features.

To better understand what was done and the meaning of each subset, we drew the Venn diagram of four circles, referring to the four groups of data we have in our primary dataset (Figure 4). The larger circle in the diagram refers to the applicant's data because each customer has this data group. However, other data groups are available to some customers and unavailable to others. Thus, the three intersecting circles refer to corporate data, shareholder data, and credit history.

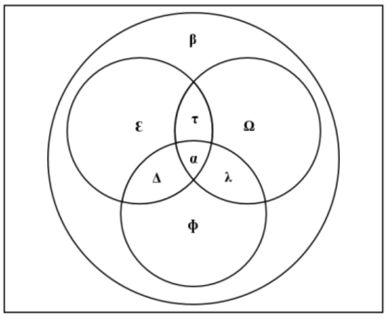

Figure 4. Venn diagram

Table 4 shows each subset in terms of the parameters shown on the Venn diagram.

Table 4. Subsets based on the parameters of the Venn diagram

Subset	Parameters Based on Diagram	Number of Rows
1	Ω + τ + α + λ	43188
2	ϕ + Δ + α + λ	45564
3	ε + τ + α + Δ	28151
4	Δ + α	7889
5	λ + α	36595
6	τ + α	8193
7	α	6844
Entire Dataset	β + Ω + ϕ + ε + τ + α + λ + Δ	78379

2.3.2. Default and non-default in each selected subset

As mentioned in the previous section, we will consider using combinations of features instead of considering all features simultaneously. Table 5 shows the number of rows, non-default and default cases in each subset, and the percentage of default in each subset. This table also demonstrates that each combination results in an imbalanced subset. This type of dataset has some classes, some have few instances, and some have many instances, but the classes with minority cases are often the classes of interest (Zhu et al., 2017). In another definition, an imbalanced dataset is a dataset where the classes are far from available with an equal number of instances (Chawla et al., 2002).

Table 5 also informs us that Subset 3 has the minimum default percentage compared to the other subsets, which is significant because this subset refers to regular customers of the Turkish financial institution. Therefore, if a customer received a loan from the financial institution before their last loan from the same financial institution, then they are less likely to default compared to new customers. However, this does not mean a regular customer will not default after receiving another loan.

Table 5. Non-default and default cases in each subset

No	Subset	# Rows	# Non-default	# Default	Default Percentage
1	Corporate data + Applicant's data	43188	41589	1599	3.70%
2	Shareholder data + Applicant's data	45564	43698	1866	4.09%
3	Credit history + Applicant's data	28151	28072	79	0.28%
4	Shareholder data + Credit history + Applicant's data	7889	7829	60	0.76%
5	Corporate data + Shareholder data + Applicant's data	36595	35270	1325	3.62%
6	Corporate data + Credit history + Applicant's data	8193	8119	74	0.90%
7	Corporate data + Shareholder data + Credit history + Applicant's data	6844	6788	56	0.82%
	Entire Dataset	78379	75515	2864	3.65%

2.4. Credit risk assessment

2.4.1. Sampling methods

Because we have imbalanced subsets, we use under-sampling, oversampling, and synthetic minority oversampling techniques to balance the instances in each subset. We also use random sampling, but only to see the usefulness of other sampling techniques when dealing with an unbalanced dataset.

- *Random sampling:* This simple sampling method was used to randomly divide each subset into a training set (75%) and a test set (25%). Table 6 lists the exact number of default and non-default examples among the training and testing sets for each of the seven subsets when using the random sampling method.

Table 6. Size of training and test sets when using random sampling

Subset	Training set size	Non-default in training set	Default in training set	Test set size	Non-default in test set	Default in test set
1	32392	31192	1200	10796	10397	399
2	34174	32774	1400	11390	10924	466
3	21114	21054	60	7037	7018	19
4	5917	5872	45	1972	1957	15
5	27447	26453	994	9148	8817	331
6	6146	6090	56	2047	2029	18
7	5133	5091	42	1711	1697	14

- *Under-sampling:* is a resampling technique that selects only a few instances from the majority class to obtain a balanced sample of instances (Zhou, 2013). Therefore, under-sampling reduces the number of instances in the majority class (Shelke et al., 2017). This reduction is done randomly, as we use random under-sampling in this work. In this case, the test set (25%) is unchanged; thus, only the training set is balanced using this sampling technique. Table 7 shows the size of the training and test sets when using the under-sampling technique.

Table 7. Size of training and test sets when using under-sampling

Subset	Training set size	Non-default in training set	Default in training set	Test set size	Non-default in test set	Default in test set
1	2400	1200	1200	10796	10397	399
2	2800	1400	1400	11390	10924	466
3	120	60	60	7037	7018	19
4	90	45	45	1972	1957	15
5	1988	994	994	9148	8817	331
6	112	56	56	2047	2029	18
7	84	42	42	1711	1697	14

- *Oversampling:* refers to increasing the number of examples of the minority class that can be obtained by generating new samples or duplicating the original minority class instances (Mohammed et al., 2020). In this work, we randomly duplicate the default instances in the training set to get a balanced dataset, and the test set remains unchanged. Table 8 refers to the number of defaults and non-defaults in the training and test sets when the oversampling method is used.

Table 8. Size of training and test sets when using oversampling

Subset	Training set size	Non-default in training set	Default in training set	Test set size	Non-default in test set	Default in test set
1	62384	31192	31192	10796	10397	399
2	65548	32774	32774	11390	10924	466
3	42108	21054	21054	7037	7018	19
4	11744	5872	5872	1972	1957	15
5	52906	26453	26453	9148	8817	331
6	12180	6090	6090	2047	2029	18
7	10182	5091	5091	1711	1697	14

- *Synthetic Minority Oversampling Technique (SMOTE):* is widely used in the literature to generate additional samples in cases of data shortages in the minority class (Elreedy and Atiya, 2019). The extra examples are generated with the requirement that they belong to one of the nearest neighbors of the dataset (Elreedy and Atiya, 2019). To synthetically generate a data point, a vector must be selected from the nearest neighbors, and then that vector must be multiplied by a random number between 0 and 1; finally, the product should be added to an unmodified data point in the dataset, and the sum will be the new data point (Shelke et al., 2017). We generate synthetic instances of the minority class using the SMOTE function in the R language. We used SMOTE to create the default cases and added or reduced a few non-default cases to balance the number of instances of each class. Table 9 shows the number of default and non-default instances after using this sampling technique.

Table 9. Size of training and test sets when using SMOTE

Subset	Training set size	Non-default in training set	Default in training set	Test set size	Non-default in test set	Default in test set
1	62400	31200	31200	10796	10397	399
2	64540	32340	32200	11390	10924	466
3	42028	21088	20940	7037	7018	19
4	11803	5908	5895	1972	1957	15
5	51688	25844	25844	9148	8817	331
6	12159	6111	6048	2047	2029	18
7	10137	5097	5040	1711	1697	14

2.4.2. Classifiers

In order to obtain the best possible results, we applied six different classifiers: Random Forest, Naïve Bayes, Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbors.

- *Random Forest (RF):* is a tree-based algorithm that gives the average prediction based on the predictions obtained from an ensemble of trees (Schonlau and Zou, 2020). The use of this algorithm for classification purposes works based on the majority of votes of the constructed trees, with each tree voting for a single class (Ishwaran, 2015). For instance, in the case of a binary classification task, if the majority of the trees vote for the label "1" and not "0", then the predicted label for the case will be "1". These trees must contain some variables for their construction, usually the square root of the total number of variables in a dataset for classification purposes (Probst et al., 2019).
- *Naive Bayes (NB):* is a simple and computationally inexpensive algorithm (Kaur and Oberai, 2014). This algorithm depends on Bayes' theorem (Langarizadeh and Moghbeli, 2016). As the NB classifier is based on calculated probabilities, it is a probabilistic method to classify instances of different classes (Yager, 2006). Besides, NB is built on the assumption that all attributes are independent of each other (Langarizadeh and Moghbeli, 2016; Chen et al., 2020).
- *Logistic Regression (LR):* is a powerful tool to study how independent variables affect a dependent variable by quantifying their contribution to the output (Stoltzfus, 2011). LR can only work when the dependent variable has two classes, such as a binary variable (Tripepi et al., 2008).
- *Support Vector Machine (SVM):* is an algorithm that tries to find a hyperplane that can effectively separate one class from another (Mammone et al., 2009). The optimal hyperplane is the one that can separate instances of two classes with the maximum margin (Hearst et al., 1998; Mammone et al., 2009). SVM can solve linear and nonlinear classification problems through its kernel functions (Mammone et al., 2009). The kernel function is a mathematical scheme that allows the SVM algorithm to project data into a higher dimensional space (Noble, 2006).
- *Decision Tree (DT):* is based on a recursive process to establish the classification rules (Quinlan, 1990). The repetitive partitioning process is done according to the importance of the

discriminating features (De Ville, 2013). Two measures are commonly used to assess how impure or inhomogeneous the data is, namely the Gini index and the entropy measure (Kingsford and Salzberg, 2008).

- *K-Nearest Neighbor (KNN):* predicts the outcome based on the proximity of data points (Kurniadi et al., 2018). KNN assumes that if the distance between two objects is short, those objects are more likely to be similar (Meng et al., 2007).

2.4.3. Evaluation metrics

Finally, we use three metrics: sensitivity [TP / (TP + FN)], specificity [TN / (TN + FP)] and accuracy [(TP + TN) / (TP+TN+FP+FN)] to evaluate the performance of each classifier with each sampling method. In our case, the positive class corresponds to instances labeled with 0, while the negative class corresponds to instances labeled with 1. Considering just one of these metrics can be misleading; therefore, we compare the results in the next chapter by considering the three metrics to assess the model's performance. We can have a sensitivity between 90 and 100%, which means that the majority or positive class was well predicted; however, the specificity can be between 0 and 50%, which means that the minority or negative class was not well predicted. For accuracy, it is usually high if the sensitivity is high or low if the sensitivity is low. Thus, accuracy is not a reliable metric for an imbalanced dataset. It is used in this work to check the overall performance of the models, but sensitivity and specificity are used to interpret the results.

3. Results

In this section, we report the best results using each sampling method for each subset, except for random sampling, because no classifier among those used could simultaneously provide a value equal to 50% or more for sensitivity and specificity. However, the other three sampling methods, under-sampling, oversampling, and SMOTE, were able to reduce the bias towards only one of the classes. Although some algorithms led to roughly the same results, we report only one classifier as the best, considering that the two parameters, sensitivity and specificity, are higher together, the better. Table 10 lists the best classifier for each sampling method and the sensitivity, specificity, and accuracy values.

Table 10. Best classifier with each sampling method

Subset	Sampling Method	Best Classifier	Sensitivity	Specificity	Accuracy
1	Under-sampling	RF	71.96%	75.69%	72.10%
	Oversampling	LR	83.44%	52.63%	82.30%
	SMOTE	LR	78.02%	55.64%	77.20%
2	Under-sampling	RF	73.43%	78.33%	73.63%
	Oversampling	SVM	75.01%	69.31%	74.78%
	SMOTE	LR	79.38%	62.45%	78.68%
3	Under-sampling	LR	79.64%	77.78%	79.63%
	Oversampling	DT	87.35%	68.42%	87.30%
	SMOTE	DT	89.54%	63.16%	89.47%
4	Under-sampling	RF	67.14%	66.67%	67.14%
	Oversampling	DT	86.41%	60.00%	86.21%
	SMOTE	SVM	79.56%	66.67%	79.46%
5	Under-sampling	LR	81.44%	66.47%	80.89%
	Oversampling	LR	83.86%	62.84%	83.10%
	SMOTE	LR	81.60%	62.54%	80.91%
6	Under-sampling	RF	68.90%	77.78%	68.98%
	Oversampling	SVM	82.60%	66.67%	82.46%
	SMOTE	SVM	84.48%	72.22%	84.37%
7	Under-sampling	RF	62.05%	78.57%	62.19%
	Oversampling	LR	84.03%	64.29%	83.87%
	SMOTE	LR	87.39%	57.14%	87.14%

4. Discussion

The results reported in Section 3 can be considered insignificant, average, or good, depending on the sampling method and the classifier used. In this section, we compare each sampling technique with the other sampling techniques in terms of their effect on the performance of the six classifiers on the majority class and the minority class.

- *Random sampling*

We can notice from Figure 5 that all the classifiers strongly mispredicted one of the classes, especially the minority class, when using random sampling, with a few exceptions for the NB classifier. In other words, classifiers were biased towards a single class, so none of the results from this sampling technique can be accepted as fair or good.

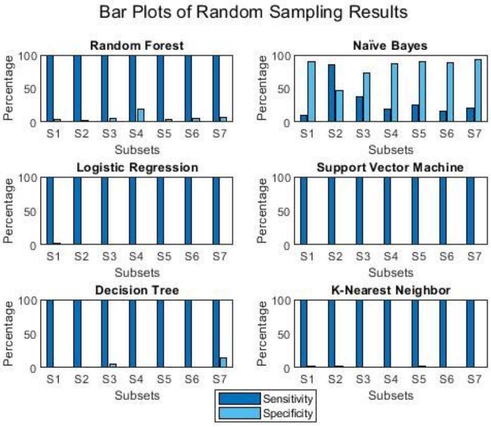

Figure 5. Bar plots of random sampling results with the six classifiers

- *Under-sampling*

The under-sampling method effectively reduced the bias in favor of the minority class compared to random sampling. It gave significantly better results than the previous sampling method (Figure 6). We can also see from the figure that the NB is still biased towards the minority class for most subsets, which is explained by the low sensitivity and medium to high specificity. In contrast, the other classifiers, RF, LR, SVM, DT, and KNN, gave more balanced results when comparing sensitivity to specificity or vice versa.

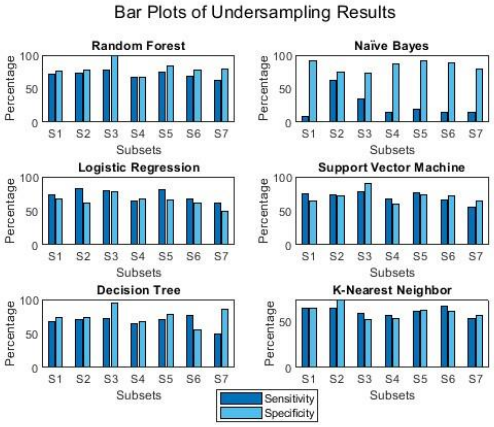

Figure 6. Bar plots of under-sampling results with the six classifiers

- *Oversampling*

The oversampling technique could also increase the classifiers' performance, reduce the bias towards one of the classes, and provide better results than the random sampling technique. However, this fact mainly applies to LR, SVM, and DT classifiers. Random Forest and K-Nearest Neighbor still fail to predict the minority class with the oversampling technique. Moreover, NB still fails to predict most instances of the majority class with this sampling technique, except again for Subset 2. When we compare oversampling to under-sampling, we can say that specificity is often greater than sensitivity when using under-sampling. In contrast, sensitivity was often greater than specificity when using oversampling. In other words, we can say that based on Figures 5 and 6, under-sampling was better at predicting the minority class than oversampling. Still, oversampling was better at predicting the majority class. Nevertheless, it is difficult to decide whether the under-sampling approach is better than the oversampling technique or vice versa, especially since the specificity and sensitivity values are often close to each other, considering the best results obtained from each sampling method and for each of the seven subsets. We must not ignore that the majority class is also important because if we classify a non-default instance as a default instance, the financial institution may lose a customer who could be profitable for its business. On the other hand, if an event of default is mistakenly classified as non-default, it can harm the financial institution.

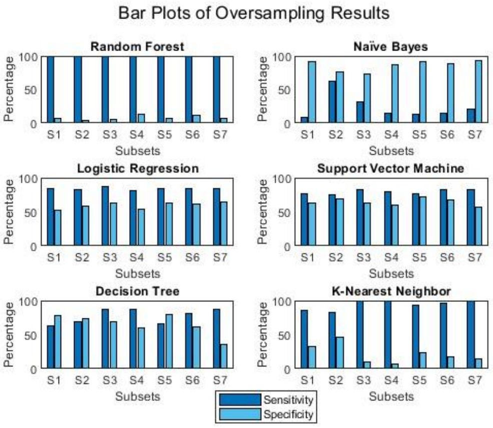

Figure 7. Bar plots of oversampling results with the six classifiers

- *SMOTE*

SMOTE was better at predicting both classes relatively than random sampling; however, this technique that adds synthetic instances of the minority class could not remove the bias towards one class with all classifiers and for all subsets. Figure 8 shows that LR and SVM could simultaneously provide fair sensitivity and specificity for all subsets. Yet, this observation does not apply when checking RF, NB, DT, and KNN subplots. When comparing SMOTE to under-sampling and oversampling, it is again complicated to favor one sampling technique over another because the best results, as reported in Section 3, are very close regarding sensitivity and specificity for each subset.

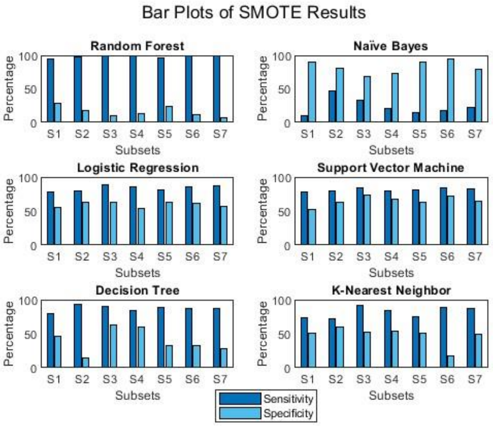

Figure 8. Bar plots of SMOTE results with the six classifiers

Table 11 reports the best combined technique, including the sampling method and classifier, which led to each subset's best sensitivity and specificity. We used Table 10, which includes the best classifier for each sampling method and each subset, to compare the values of these two metrics.

Table 11 shows that oversampling and SMOTE performed better in predicting the majority class, while under-sampling performed better for all subsets in predicting the minority class. This table also tells us that the classifier's performance depends on the sampling method and the subset used. Thus, only the classifier cannot be judged as bad or good without considering the sampling method used and the dataset type, whether unbalanced or balanced.

Table 11. Best combined technique based on sensitivity and specificity

Subset	Best Technique for the Highest Sensitivity	Best Technique for the Highest Specificity
1	Oversampling + LR	Under-sampling + RF
2	SMOTE + LR	Under-sampling + RF
3	SMOTE + DT	Under-sampling + LR
4	Oversampling + DT	Under-sampling + RF and SMOTE + SVM
5	Oversampling + LR	Under-sampling + LR
6	SMOTE + SVM	Under-sampling + RF
7	SMOTE + LR	Under-sampling + RF

Additionally, for most subsets and processes, NB performed well for the minority class and poorly for the majority class. Therefore, this algorithm predicted that most cases were default instances, and it strongly failed to predict the majority class, which means that this classifier is not reliable in our case.

5. Conclusion

In this article, we based our work on the corporate credit dataset of a Turkish financial institution to assess credit risk and demonstrate the three crucial challenges related to credit datasets. First, the variables in the dataset were broadly categorized into input, output, information, and irrelevant variables. Based on this classification, we have identified the variables to be selected as features to evaluate credit risk. These features were further investigated and reduced based on missing/zero value analysis and correlation analysis. Since we had many missing values in many input variables that did not occur accidentally, we divided our dataset into seven subsets based on the data groups available for each customer.

Then we applied four different sampling methods (random sampling, under-sampling, oversampling, and SMOTE); three considered instance balancing. For each subset, we predicted the output variable that refers to the default or non-default using six classifiers: RF, NB, LR, SVM, DT, and KNN.

With the random sampling technique, most classifiers predicted that all or most cases would belong to the majority class, which was wrong. The other sampling techniques were fundamental to overcoming the problem of biased predictions due to the unbalanced nature of the dataset. Under-sampling, oversampling, and SMOTE reduced the bias toward a single class, but these results vary from classifier to classifier. With these sampling methods and some classifiers, we could simultaneously achieve at least greater than 50% sensitivity and specificity for each subset (Table 10). However, the sensitivity did not exceed 79% when classifiers did not ignore negative or positive instances for any of the seven subsets. We can conclude that the default was insufficiently predicted, which may be due to other measurable or non-measurable factors.

Improving sensitivity and specificity, adding variables from companies' financial statements to our models, and adjusting monetary amounts for the effect of inflation on the local currency should be the main goals of further work on the same dataset type.

Acknowledgements

As this article is part of my master's thesis, I would like to express my thanks and gratitude to my thesis supervisor, Prof. Dr. Ayşe Hümeyra Bilge. Without her continued help and guidance, this work would not have been possible. I would also like to thank the Turkish financial institution, which preferred to remain anonymous, for allowing me to use its corporate credit dataset in this article.

Statement of Conflict of Interest

There is no conflict of interest.

Author’s Contributions

The author contributed to this manuscript 100%.

Abbreviations and Symbols

DT	Decision Tree
Ɛ	Applicants s who have credit history and applicant’s data only (Ɛ = 18913)
FN	False negatives
FP	False positives
KKB	Kredi Kayıt Bürosu meaning Credit Registration Office in English
KNN	K-Nearest Neighbors
KVKK	Kişisel Verileri Koruma Kurumu meaning Personal Data Protection Authority in English
LR	Logistic Regression
ML	Machine Learning
NB	Naïve Bayes
ϕ	Applicants who have shareholder data and applicant’s data only (ϕ = 7924)
RF	Random Forest
SMOTE	Synthetic minority oversampling technique
SPK	Sermaye Piyasası Kurulu meaning Capital Markets Board in English
SVM	Support Vector Machine
TCMB	Türkiye Cumhuriyet Merkez Bankası meaning Central Bank of the Republic of Türkiye in English
TN	True negatives
TP	True positives

α Applicants who have all data groups (α = 6844)

β Applicants with only applicant’s data or who have at least one missing entry in each of the remaining data groups (β = 7309)

Δ Applicants with shareholder data, applicant’s data, and credit history but not corporate data (Δ = 1045)

λ Applicants with corporate data, applicant’s data, and shareholder data but not credit history (λ = 29751)

τ Applicants with corporate data, applicant’s data, and credit history but not shareholder data (τ = 1349)

Ω Applicants who have corporate data and applicant’s data only (Ω = 5244)

References

Abdou HA., Pointon J. Credit scoring, statistical techniques and evaluation criteria: a review of the literature. Intelligent Systems in Accounting, Finance and Management 2011; 18(2-3): 59-88.

Chawla NV., Bowyer KW., Hall LO., Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research 2002; 16: 321-357.

Chen S., Webb GI., Liu L., Ma X. A novel selective naïve Bayes algorithm. Knowledge-Based Systems 2020; 192: 105361.

De Ville B. Decision trees. Wiley Interdisciplinary Reviews: Computational Statistics 2013; 5(6): 448-455.

Elreedy D., Atiya AF. A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance. Information Sciences 2019; 505: 32-64.

Ghosh D., Vogt A. Outliers: An evaluation of methodologies. Joint statistical meetings, July 2012, page no: 3455-3460, United States.

Hearst MA., Dumais ST., Osuna E., Platt J., Scholkopf B. Support vector machines. IEEE Intelligent Systems and Their Applications 1998; 13(4): 18-28.

Hodge V., Austin J. A survey of outlier detection methodologies. Artificial Intelligence Review 2004; 22: 85-126.

Ishwaran H. The effect of splitting on random forests. Machine Learning 2015; 99(1): 75-118.

Kaur G., Oberai EN. A review article on Naive Bayes classifier with various smoothing techniques. International Journal of Computer Science and Mobile Computing 2014; 3(10): 864-868.

Kingsford C., Salzberg SL. What are decision trees?. Nature Biotechnology 2008; 26(9): 1011-1013.

Kurniadi D., Abdurachman E., Warnars HLHS., Suparta W. The prediction of scholarship recipients in higher education using k-Nearest neighbor algorithm. IOP conference series: materials science and engineering, 18 April 2018, page no: 012039, Indonesia.

Kwak SK., Kim JH. Statistical data preparation: management of missing values and outliers. Korean

Journal of Anesthesiology 2017; 70(4): 407-411.

Langarizadeh M., Moghbeli F. Applying naive bayesian networks to disease prediction: a systematic review. Acta Informatica Medica 2016; 24(5): 364.

Mammone A., Turchi M., Cristianini N. Support vector machines. Wiley Interdisciplinary Reviews: Computational Statistics 2009; 1(3): 283-289.

Meng Q., Cieszewski CJ., Madden M., Borders BE. K nearest neighbor method for forest inventory using remote sensing data. GIScience & Remote Sensing 2007; 44(2): 149-165.

Mohammed R., Rawashdeh J., Abdullah M. Machine learning with oversampling and undersampling techniques: overview study and experimental results. 11th international conference on information and communication systems (ICICS), 07-09 April 2020, page no: 243-248, Jordan.

Noble WS. What is a support vector machine?. Nature Biotechnology 2006; 24(12): 1565-1567.

Probst P., Wright MN., Boulesteix AL. Hyperparameters and tuning strategies for random forest. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2019; 9(3): e1301.

Quinlan JR. Decision trees and decision-making. IEEE Transactions on Systems, Man, and Cybernetics 1990; 20(2): 339-346.

Schonlau M., Zou RY. The random forest algorithm for statistical learning. The Stata Journal 2020; 20(1): 3-29.

Shelke MS., Deshmukh PR., Shandilya VK. A review on imbalanced data handling using undersampling and oversampling technique. Int. J. Recent Trends Eng. Res 2017; 3(4): 444-449.

Stoltzfus JC. Logistic regression: a brief primer. Academic Emergency Medicine 2011; 18(10): 1099-1104.

Sun S., Huang R. An adaptive k-nearest neighbor algorithm. Seventh international conference on fuzzy systems and knowledge discovery, 10-12 August 2010, page no: 91-94, China.

Tripepi G., Jager KJ., Dekker FW., Zoccali C. Linear and logistic regression analysis. Kidney International 2008; 73(7): 806-810.

Walfish S. A review of statistical outlier methods. Pharmaceutical Technology 2006; 30(11): 82.

Weissova I., Kollar B., Siekelova A. Rating as a useful tool for credit risk measurement. Procedia Economics and Finance 2015; 26: 278-285.

Yager RR. An extension of the naive Bayesian classifier. Information Sciences 2006; 176(5): 577-588.

Zhou L. Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. Knowledge-Based Systems 2013; 41: 16-25.

Zhu T., Lin Y., Liu Y. Synthetic minority oversampling technique for multiclass imbalance problems. Pattern Recognition 2017; 72: 327-340.