UNIVERSIDADE FEDERAL DO CEARÁ

CENTRO DE CIÊNCIAS

DEPARTAMENTO DE COMPUTAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

LEOPOLDO SOARES DE MELO JUNIOR

IMPROVING DYNAMIC SELECTION PREDICTION IN IMBALANCED CREDIT SCORING PROBLEMS

FORTALEZA

2020

LEOPOLDO SOARES DE MELO JUNIOR

IMPROVING DYNAMIC SELECTION PREDICTION IN IMBALANCED CREDIT SCORING PROBLEMS

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de doutor em Ciência da computação. Área de Concentração: Aprendizado de máquina

Orientador: Prof. Dr. José Antônio Fernandes de Macêdo

FORTALEZA

2020

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

M485i Melo Junior, Leopoldo Soares de.
Improving dynamic selection prediction in imbalanced credit scoring problems / Leopoldo Soares de Melo Junior. – 2020.
105 f. : il. color.

Tese (doutorado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação , Fortaleza, 2020.
Orientação: Prof. Dr. José Antônio Fernandes de Macêdo.

1. Credit scoring. 2. Imbalanced learning. 3. Dynamic Selection Classification. I. Título.
CDD 005

LEOPOLDO SOARES DE MELO JUNIOR

IMPROVING DYNAMIC SELECTION PREDICTION IN IMBALANCED CREDIT SCORING PROBLEMS

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de doutor em Ciência da computação. Área de Concentração: Aprendizado de máquina

Aprovada em: 15 de Junho de 2020

BANCA EXAMINADORA

Prof. Dr. José Antônio Fernandes de Macêdo (Orientador)
Universidade Federal do Ceará (UFC)

Prof[a]. Dr[a]. Chiara Renso
Consiglio Nazionale delle Ricerche (CNR-Italy)

Prof. Dr. Franco Maria Nardini
Consiglio Nazionale delle Ricerche (CNR-Italy)

Prof. Dr. César Lincoln Cavalcante Mattos
Universidade Federal do Ceará - UFC

Prof. Dr. Regis Pires Magalhães
Universidade Federal do Ceará - UFC

Prof[a]. Dr[a]. Ticiana Linhares Coelho da Silva
Universidade Federal do Ceará - UFC

To my family, for their ability to believe and invest in me.

ACKNOWLEDGEMENTS

I thank my parents for my existence and for being able to learn every day.

I also thank the constant support and encouragement of my wife Renata, my mother Alcida, and my uncle Thomé, who have always supported me and taught me the importance of education at all times in my life. Thanks also to my dear sisters, Elisabeth, Natália (in memoriam), and Cristiane.

I am very grateful to my wonderful son Leonardo for understanding all times that we could not be together due to research activities developed during this work.

To Prof. José Macedo for being my advisor in this Doctoral degree even without knowing me properly. Thank you also for all the support received throughout this research.

To Chiara, Franco, César Lincoln, Regis, and Ticiana, the members of this Thesis defense committee, for the readings, comments, and contributions for the improvement of this work.

To all friends from the HPC Lab in ISTI-CNR-Pisa, especially Chiara Renso, Franco Maria Nardini, Raffaele Perego, and Roberto Trani for the wonderful reception that I had during my stay in Italy in 2018 and for always being very attentive and accurate in their observations, which were essential for the accomplishment of this work. Fortunately, we were able to chat and use many remote meet sessions in HCP Lab and through countless other moments.

To Vinícius Monteiro, my good Brazilian friend in Italy. The year I lived in Pisa was much more enjoyable because of him. He treated me like a real brother. I am also grateful to all friends and other people who helped me at CNR-Pisa: Salvatori Trani, Ida Mele, Emanuele Carlini, Matteo Catena, Cristina Muntean, Patrizio Dazzi, Massimo Coppola, Salvatore Orlando, Nicola Tonellotto, Fabrizio Fabbrini, ...

To the great friends and partners of Ph.D. degree, research, and study in the Insight Lab: Regis Pires, Lívia Almada, Luis Cesar Vasconcelos, and Nicksson Arrais. You inspire me every day. Thanks for everything.

To the great friends of the Insight Lab, who contributed along this long journey in moments of study, work, and discussions: David Araújo, Igo Brilhante, Gustavo Coutinho, Emanuel Oliveira, Tércio Jorge, Lívio Freire, Luís César, Ricardo Ávila, Guilherme, Nicksson, João Holanda, Victor, Darley, Victories, Felipes, Hinessa, Abelardo, among others. Thank you for the excellent moments of interaction and learning that we could share along with this challenging and, at the same time, a pleasant walk because of your presence.

To all friends of the Banco do Nordeste do Brasil S/A, who helped me to understand the real life requirements of a credit scoring application: Stelio Gama, Rosa Cristina, Lauro Ramos, Lúcia de Fátima, Jarbas Sousa, Egidio Gomes, Manoel Neto, José Valente, and Jesuino José, among others. Thank you for the excellent moments of interaction and learning that we could share along this journey.

To all support received by the Banco do Nordeste do Brasil S/A (BNB) for allowing me to work integrally on my Ph.D. course for 2 years and 11 months.

I thank all my teachers and mentors for providing me with traditional knowledge and the need to improve my skills daily and understand the relevance of education to transform this world into a better place. Thank you for helping me learn and grow.

To all who were not mentioned up to this line but who also contributed directly or indirectly to accomplishing this work.

Work hard in silence, let your success be your noise.

(Frank Ocean)

RESUMO

Os credores, como bancos e empresas de cartão de crédito, usam modelos de credit scoring para avaliar o risco potencial representado pelo empréstimo de dinheiro aos consumidores e, portanto, para mitigar perdas devido a inadimplência. Assim, a rentabilidade dos bancos depende muito dos modelos utilizados para decidir sobre os empréstimos dos clientes. Modelos de credit scoring de última geração usam aprendizado de máquina e métodos estatísticos. Um dos principais problemas desse campo é que os credores geralmente lidam com conjuntos de dados desequilibrados que geralmente contêm muitos empréstimos pagos, mas muito poucos empréstimos não pagos (chamados *defaults*). Recentemente, métodos de seleção dinâmica combinados com técnicas de pré-processamento têm sido avaliados para melhorar os modelos de classificação em dados desequilibrados apresentando vantagens sobre os métodos de aprendizado de máquina estáticos. Em uma técnica de seleção dinâmica, amostras conhecidas na vizinhança de uma amostra desconhecida são usadas para calcular a competência local dos classificadores base. Então, essas técnicas selecionam apenas classificadores localmente competentes na vizinhança da amostra desconhecida. A maioria das técnicas de seleção dinâmica usa o algoritmo k-NN para definir o conceito de região local. Nesta tese, modificamos técnicas de seleção dinâmica para melhorar o desempenho de previsão em conjuntos de dados de credit scoring desequilibrados. Primeiramente, avaliamos o desempenho de técnicas estáticas quando submetidas a vários níveis de desequilíbrio. A seguir, aplicamos técnicas de seleção dinâmica nos melhores ensembles do experimento anterior com uma nova definição da região local, a Reduced Minority k-NN (RMkNN). A intuição por trás do RMkNN é superar o comportamento tendencioso do k-NN na definição das regiões locais em conjuntos de dados desequilibrados, principalmente selecionando amostras da classe majoritária. Depois, exploramos as melhorias modificando a métrica de desempenho usada para calcular a competência local dos classificadores básicos. A intuição é substituir a acurácia por uma medida mais adequada para conjuntos de dados desequilibrados. Esta métrica é FA^2, a combinação da Fmeasure com o quadrado da acurácia. Descobrimos que essas modificações melhoram o desempenho de previsão em dados de credit scoring desequilibrados. Finalmente, combinamos as técnicas RMkNN e FA^2 para avaliar a melhoria total da previsão no problema de credit scoring. Conduzimos uma avaliação abrangente da técnica proposta contra concorrentes de última geração em seis conjuntos de dados públicos do mundo real e um privado. Experimentos mostram que RMkNN e FA^2 melhoram o desempenho de classificação dos dados avaliados em até 18 % em relação a sete medidas de desempenho.

Keywords: Credit scoring. Aprendizagem desequilibrada. Seleção dinâmica de Classificadores.

ABSTRACT

Lenders, such as banks and credit card companies use credit scoring models to evaluate the potential risk posed by lending money to consumers and, therefore, to mitigate losses due to bad credit. Thus, the profitability of the banks highly depends on the models used to decide on the customer's loans. State-of-the-art credit scoring models use machine learning and statistical methods. One of the major problems of this field is that lenders often deal with imbalanced datasets that usually contain many paid loans but very few not paid ones (called *defaults*). Recently, dynamic selection methods combined with preprocessing techniques have been evaluated to improve classification models in imbalanced datasets presenting advantages over the static machine learning methods. In a dynamic selection technique, samples in the neighborhood of each query sample are used to compute the base classifiers' local competence. Then, these techniques select only locally competent classifiers according to each query sample. Most dynamic selection techniques use the k-NN algorithm to define the concept of the local region. In this thesis, we modify dynamic selection techniques to improve the prediction performance in imbalanced credit scoring datasets. First, we evaluate the performance of static techniques when submitted to several imbalanced levels. Next, we apply dynamic selection techniques in the best ensembles of the previous experiment with a new definition of the local region, the Reduced Minority k-Nearest Neighbors (RMkNN). The intuition behind RMkNN is to overcome the biased behavior of kNN in defining the local regions in imbalanced datasets, mainly selecting samples of the majority class. After, we explore improvements by modifying the performance measure used to compute the local competence of base classifiers. The intuition is to replace accuracy with a measure better suited to imbalanced datasets. This metric is FA^2, the combination of F-measure with the square of accuracy. We find out that these modifications improve the prediction performance in imbalanced credit scoring datasets. Finally, we combine RMkNN and FA^2 techniques to evaluate the total prediction improvement on the credit scoring problem. We conduct a comprehensive evaluation of the proposed technique against state-of-art competitors on six real-world public datasets and one private one. Experiments show that RMkNN and FA^2 improve the classification performance of the evaluated datasets up to 18% regarding seven performance measures.

Keywords: Credit scoring. Imbalanced learning. Dynamic Selection Classification.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AdaBoost	Adaptive Boosting
ANN	Artificial Neural Networks
AUC	Area Under the receiver-operating-characteristics Curve
BBAG	Balanced Bagging (Bagging + RUS)
BBLR	Balanced Bagging of Logistic Regression
BBTR	Balanced Bagging of Decision Trees
BGSM	Bagging SMOTE (Bagging + SMOTE)
BRND	Balanced Random Forest (Random Forest + RUS)
BROT	Balanced Rotation Forest (Rotation Forest + RUS)
CART	Classification and Regression Trees
CMEC	Calibrated Adaboost with Minimum Expected Cost
DA	Discriminant Analysis
DCS	Dynamic Classifier Selection
DES	Dynamic Ensemble Selection
DES-MI	DES Multiclass Imbalance
DS	Dynamic Selection
DSC	Dynamic Selection Classification
DSEL	Dynamic Selection Dataset
EAD	Exposure at Default
EASY	Easy ensemble (Bagging of AdaBoost + RUS)
ELM	Extreme Learning Machines
F-1	Fisher's Discriminant Ratio
FUZZY	Fuzzy Pattern Tree
GB	Gradient Boosting
GENET	genetic programming symbolic classifier
GNB	Gaussian naive Bayes
GNG	Gabriel Neighborhood Graph editing
IR	Imbalanced Ratio
KNE	K-Nearest Oracles-Eliminate
KNIU	K-Nearest Oracles-Imbalanced Union

kNN	k-Nearest Neighbors
KNORA	K-Nearest Oracles
KNORA-IU	KNORA-Imbalanced Union
KNU	K-Nearest Oracles-Union
LCA	Local Class Accuracy
LDA	Linear Discriminant Analysis
LDP	low default portfolios
LGD	Loss Given Default
LOGR	Logistic Regression
LSVM	Linear Support Vector Machine
MARS	Multivariate Adaptive Regression Splines
N2	Ratio of Average Intra/Inter class Nearest Neighbor distance
N3	Error Rate for 1 Nearest Neighbor Classifier
NN	Nearest Neighbor
PD	Probability of Default
QDA	Quadratic Discriminant Analysis
RAMO	Ranked Minority Oversampling
RFSM	Random Forest SMOTE (Random Forest + SMOTE)
RMkNN	Reduced Minority k-Nearest Neighbors
RNDF	Random Forest
RNK	Modified Classifier Rank
RUS	*Random Undersampling*
RUSB	RUS Boost (AdaBoost + RUS)
SLFN	single-hidden layer feed-forward neural network
SMOTE	*Synthetic Minority Over-sampling Technique*
SMTB	SMOTE Boost (AdaBoost + SMOTE)
SS	Static Selection
SVM	Support Vector Machine
TREES	Decision Trees
XGB	eXtreme Gradient Boosting

CONTENTS

1 INTRODUCTION

Credit offer is a crucial activity for banks that aim at improving their profitability and competitiveness. Minor improvements in the default prediction imply significant profits to financial institutions (HAND; HENLEY, 1997). However, the decision to grant a loan to a customer is complex and risky because it requires an accurate default prediction to protect banks from financial losses, especially during financial crises. Thomas *et al.* (2017) pointed out several aspects affecting the default rate over time, such as the cost of the money (interest rate), the supply and demand for credit, the state of the economy, and the cyclical variations of credit over time. Besides these aspects, data availability, accuracy, and reliability make the default prediction much harder than other domain-specific classification problems. Therefore, new methods and techniques, called credit scoring models, are required to cope with these problems while guaranteeing a low percentage of defaults.

Available historical loan data creates an excellent opportunity to take advantage of trending machine learning methods for detecting defaulters, people that do not pay back the loan. However, real credit scoring datasets are usually high imbalanced. They are called low default portfolios (LDP) since they are highly skewed and with a low default rate. Furthermore, these prediction models must follow constraints imposed by international accords, called Basel accords (PENIKAS, 2015). The Basel Accords are three sequential banking regulation agreements (Basel I, II, and III) set by the Basel Committee on Bank Supervision (BCBS). Basel Capital Accord II (ATIK, 2010), for instance, defines the validation and verification of three estimates: the Probability of Default (PD), Loss Given Default (LGD), and Exposure at Default (EAD) (THOMAS *et al.*, 2005).

Recent credit scoring papers (GARCÍA *et al.*, 2019; HE *et al.*, 2018; SUN *et al.*, 2018; XIA *et al.*, 2018; ABELLÁN; CASTELLANO, 2017) evaluate improvements in defaulter's prediction by using ensembles, a classification approach that combines the predictions of a set of base classifiers instead of only one. These papers usually use a set of available credit scoring data to evaluate their approaches. We observe that most of the datasets used in these papers are low imbalanced, when the IR, the ratio between the number of samples of the classes, is under 3. However, in the real world, credit scoring datasets are moderate or high imbalanced, $IR \geq 3$; and skewed data is a challenge for machine learning methods since classifiers tend to predict only the majority class.

Another remarkable aspect of the credit scoring research field is the regulation

issues about using Dynamic Selection Classification (DSC), a kind of ensemble that performs predictions using a subset of base classifiers according to the sample. Basel accords (PENIKAS, 2015) require that the same credit scoring system evaluate all customers. It means that DSC approaches have regulatory issues to be adopted in the real world.

This thesis aims at improving the use of DSC in imbalanced credit scoring problems. To this end, we perform a benchmark on static classification approaches to find the most competitive static ensembles in the credit scoring field. After, we measure the improvements of combining DSC and imbalanced approaches in credit data. Then, we found an equivalency between a dynamic selection approach and a static one to address credit scoring regulatory constraints. After, we investigate two different imbalanced approaches to attenuate the impact of skewed data on DSC techniques. Finally, we evaluate the combination of these two approaches.

1.1 Issues affecting existing solutions

We performed some preliminary studies on several distinct existing Dynamic Selection (DS) algorithms to identify issues on state-of-the-art dynamic classification selection solutions and contribute to this field. These studies allowed us to identify some of these solutions' most critical issues when applied to imbalanced datasets.

We started evaluating the implementation elements of dynamic selection approaches. Most of these approaches use the k-Nearest Neighbors (kNN) algorithm to choose a list of known samples in the neighborhood of a query sample. These known samples define the query sample local region. The dynamic selection techniques use these local regions to compute the local competence of the ensemble's base classifiers. The second important element we observe is that most DS implementations use accuracy to determine the local competence of the base classifiers.

We noticed that, for moderate and high imbalanced datasets, when the number of samples of the majority class is more than three times higher than the samples of the minority class, kNN does not choose an appropriate number of samples of the minority class to define the local region of a query sample. The lack of minority class samples in the local region creates a bias analysis in the local competence evaluation of the base classifiers. Roy *et al.* (2018) tried to overcome this issue by applying oversampling approaches, but we empirically observed that the noise produced by oversampling techniques reduced the classification performance on credit scoring datasets.

We also noticed that most DS techniques use accuracy as the performance measure

to compute the local competence of base classifiers. Again, as kNN, accuracy does not produce appropriate results in imbalanced datasets. DS techniques use accuracy to compute the local competence (the ability to predict samples in a region of the feature space correctly) of the base classifiers. When we use imbalanced data, accuracy can not differentiate the base classifier that always predicts the majority class from the base classifier that can predict both classes correctly in a specific neighborhood.

Besides these issues regarding imbalanced datasets in DS approaches, we noticed that most credit scoring papers evaluate mainly low imbalanced datasets ($IR < 3$). Recent papers that propose classification approaches to credit scoring (GARCÍA *et al.*, 2019; FENG *et al.*, 2018; HE *et al.*, 2018; SUN *et al.*, 2018; XIA *et al.*, 2018) evaluate their approaches using mainly low imbalanced datasets. Considering the experiments performed in Chapter 3, we empirically conclude that the challenge of achieving good results in low imbalanced datasets is different from achieving good results on moderate and high imbalanced ones.

These problems helped us understand the importance of investigating improvements on DS techniques applied to moderate and high imbalanced credit scoring problems. Once we noticed that these datasets have different requirements from the low imbalanced datasets, they require specific techniques.

The primary motivation of this thesis came from the combination of two previous conclusions. First, Thomas *et al.* (2017) reported that credit data are complex and noisy. Next, the conclusion of Britto Jr *et al.* (2014) that dynamic selection techniques are appropriate for complex datasets. We saw an opportunity to improve the defaulters' recognition using dynamic selection classification.

Then, to propose a dynamic selection classification for credit scoring problem, we must ensure that the complexity of credit data is appropriate to dynamic selection classification. However, we did not find any study related to complexity level comparison in the credit scoring field. This non-existence motivates us to explore this complexity level comparison.

Another obstacle we need to handle about dynamic selection is the regulation. As highlighted by Lessmann *et al.* (2015), dynamic selection classification techniques might violate regulatory requirements of Basel Accords (ATIK, 2010) for credit scoring because they use different scorecards for different customers. The motivation for this regulation constraint is to avoid customer discrimination. Again, the lack of previous regulation-compliant works proposing dynamic selection classification for the credit scoring field motivates us to work on this topic.

1.2 Problem statement

Loan default prediction is essential because minor improvements in the prediction performance can considerably increase the profit of a financial institution. Loan prediction faces the challenge of having imbalanced data to build prediction models (SUN *et al.*, 2018; BROWN; MUES, 2012; MARQUÉS *et al.*, 2013). Additionally, credit data are complex and noisy (THOMAS *et al.*, 2017), increasing the challenge of defaulters recognition models. Another gap in credit scoring related works is that they usually evaluate more low imbalanced data than high imbalanced ones (GARCÍA *et al.*, 2019; FENG *et al.*, 2018; HE *et al.*, 2018; SUN *et al.*, 2018; XIA *et al.*, 2018). To complete the context of the loan prediction problem, Basel accords (PENIKAS, 2015) regulate the financial institutions, imposing constraints on the prediction models. One of these constraints is that the prediction model must be the same for all customers to avoid discrimination.

The difference between the profit of a successful loan and the loss of a defaulted loan drives the loan grant problem. In most cases, the loss caused by a non-paid loan is much higher than the profit of a regular one (WEST, 2000; ALTMAN *et al.*, 1977). This difference increases the challenge of this classification problem because the credit model must handle the described prediction scenario and maximize the financial institution's profit.

1.3 Objectives

This thesis's main target is to find improvements to imbalanced credit scoring classification using dynamic selection techniques. To achieve this objective, we achieved these :

- To find out the best static ensembles for imbalanced credit scoring problems.
- To determine whether dynamic selection classification is suitable for credit scoring problem.
- To find out alternatives to overcome the data skewed in dynamic selection classification.
- To find out alternatives to measure the local competence of base classifiers on the model's selection in imbalanced datasets.

1.4 Contributions

This research aims to analyze the suitability of dynamic selection classification for the probability of default estimation problem and investigate improvements of this class

of classifiers. We identify that dynamic selection is appropriate to the credit scoring problem, describe an equivalence of dynamic selection classification to the static selection, and conduct experiments to measure the improvement of dynamic selection classification over baseline credit scoring models. Next, we describe the contributions achieved during this work.

– **A credit scoring benchmark for imbalanced datasets**. Even with some benchmarks developed up to 10 years ago in this field, we decided to compare recent and traditional credit scoring approaches for imbalanced credit scoring datasets. This comparison includes evaluating the performance of the classification approaches when submitted to several different levels of imbalance ratio among good and bad customers. This benchmark confirms that the ensembles Random Forest and Gradient Boosting produce good results, regardless of the imbalance level of the data.

– **The suitability of dynamic selection techniques to credit scoring problem.** From the study carried out by Britto Jr *et al.* (2014), which concluded that DSC is suitable for complex datasets, we use some complexity measures proposed by Ho e Basu (2002) to evaluate credit data. Comparing credit data with datasets of other fields showed that the credit scoring problem is more complex, on average.

– **A novel procedure to define the local region for imbalanced dynamic selection classification.** Since DSC is suitable for credit scoring data, but it does not handle imbalanced datasets properly (ROY *et al.*, 2018), we started investigating the alternatives to improve DSC performance on imbalanced data. Considering that DSC uses kNN to define a local region in the feature space to identify the most competent base classifiers, we proposed modifying kNN called RMkNN. This modification of kNN reduces the distance of the minority class samples based on the dataset's IR. This approach tries to balance the samples in the kNN algorithm, mainly in overlapping areas of the feature space. We observe empirically that our proposed approach overcomes traditional classifiers in a real-life credit scoring experiment, detailed in subsection 4.5.2.3.

– **A novel procedure to compute the local competence of the base classifiers.** Parallel to the contribution presented above, we decided to investigate the effect of changing the performance measure used to compute the local competence of base classifiers. Once this local

competence computation uses few samples, usually 7 (BRITTO JR *et al.*, 2014), we define FA^2. The measure combines the F1-score, when samples of both classes are available, with the square of accuracy, when only samples of one class are available. This combination aims to reduce the weakness of accuracy performance measure in imbalanced distributions.

– **A novel method that combines RMkNN with FA^2.** We combine the two techniques described above, RMkNN and FA^2, in one dynamic selection approach and evaluate it.

– **The equivalence between a dynamic selection technique and a static classification approach.** To address the compliance requirements imposed by Basel Accords, which requires that the same decision model is used to evaluate all the customers, we found equivalence between a dynamic selection approach and a static approach.

1.5 Thesis organization

The organization of this thesis is as follows. Chapter 2 describes preliminary concepts and related works of this thesis. Chapter 3 compares the performance of several classification techniques when submitted to different dataset imbalance levels, which is the first contribution of this thesis. Chapter 4 presents the main contribution of this thesis that is the Reduced Minority kNN. Chapter 5 presents a new dynamic selection technique for imbalanced datasets based on a novel performance measure to compute the local competence of the base classifiers. Chapter 6 presents the results of the previous techniques combined. Finally, chapter 7 describes the conclusions of this thesis.

2 PRELIMINARY CONCEPTS AND RELATED WORKS

This study involves four main elements: credit scoring, imbalanced learning, pool generators, and dynamic selection classification. Next, we present the background of credit scoring, pool generators, imbalanced learning, and dynamic selection classification. After, we describe the credit scoring related works.

2.1 Preliminary concepts

2.1.1 Credit scoring

As defined by Thomas *et al.* (2017), credit scoring is a set of decision models that aid lenders in granting consumer credit. Financial institutions use these techniques to decide who will get credit, how much they should get, what price they should get it at, and what operational strategies will enhance the profitability of the borrowers to the lenders.

These techniques assess the risk of lending individually to each consumer (THOMAS *et al.*, 2017). This assessment by a borrower's lender reflects the circumstances of both and the lender's view of the likely future economic scenarios. Thus some lenders will assess an individual as creditworthy, and others will not.

A credit scoring dataset contains two groups of data — the first one corresponds to the regular customer information, such as age or level of scholarship. The last group corresponds to the credit behavior of this customer in previous loans. Sometimes, only the first group is available to evaluate the customer.

The vital point in a credit scoring system is that there is a large sample of previous customers with their application details and subsequent credit history available (THOMAS *et al.*, 2017). All the credit scoring techniques use samples to identify the connection between the characteristics of the consumers and how "good" or "bad" theirs subsequent story is, where bad usually means defaulting, not paid ones, in a given period, and good means not defaulting. Next, we discuss imbalanced learning approaches.

2.1.2 Imbalanced learning approaches

As mentioned in the Introduction, the prediction task in credit scoring datasets suffers from the lack of sufficient samples of the minority class, the defaulters. Haixiang *et al.*

(2017) defined four categories of techniques for handling class imbalance: (1) modify the data distribution, *preprocessing solutions*; (2) apply different costs to misclassification of positive and negative samples, the *cost-sensitive solutions*. (1) and (2) are “basic strategies" for imbalanced learning. (3) and (4) are “classification algorithms": (3) adapts a classifier to deal with the class imbalance, the *algorithm level solutions*; and (4) *ensemble-based solutions*, combines the previous solutions using an ensemble. We describe the two most common imbalanced approaches briefly in the following paragraphs, preprocessing and ensemble-based.

Preprocessing comes before the learning phase. Resampling, the most common preprocessing technique, balances the sample space for an imbalanced dataset to reduce the skewed class distribution in the learning process. There are three possible methods to do it over-sampling, Undersampling, and hybrid. The first one is over-sampling, which consists of creating new minority class samples synthetically. The widely used method is *Synthetic Minority Over-sampling Technique* (SMOTE) (CHAWLA *et al.*, 2002). The second one is under-sampling, which consists of removing samples from the majority class. The most used method is *Random Undersampling* (RUS) (BARANDELA *et al.*, 2003). Another oversampling method is Ranked Minority Oversampling (RAMO) (CHEN *et al.*, 2010). Finally, The hybrid methods combine oversampling and undersampling methods.

The other common imbalanced approach is ensemble-based methods. Ensemble methods are learning algorithms that construct a set of classifiers and then classify new data points by taking a vote, weighted or not, of their predictions (DIETTERICH, 2000). Ensemble approaches to imbalanced learning consist of combining preprocessing, cost-sensitive, and classifier algorithm modifications. They combine the power of an ensemble with the ability of other imbalanced techniques to overcome the imbalance issue.

Besides these methods, there are also “cost-sensitive solutions” and “classification algorithms”. Cost-sensitive solutions consist of assuming higher costs for the misclassification of minority class samples. On the other hand, the classification algorithms consist of changing the kernel or the activation function to improve the classification performance for imbalanced data.

The classification algorithms approach, on the other hand, consist of changing the kernel or the activation function to improve the classification performance for imbalanced data.

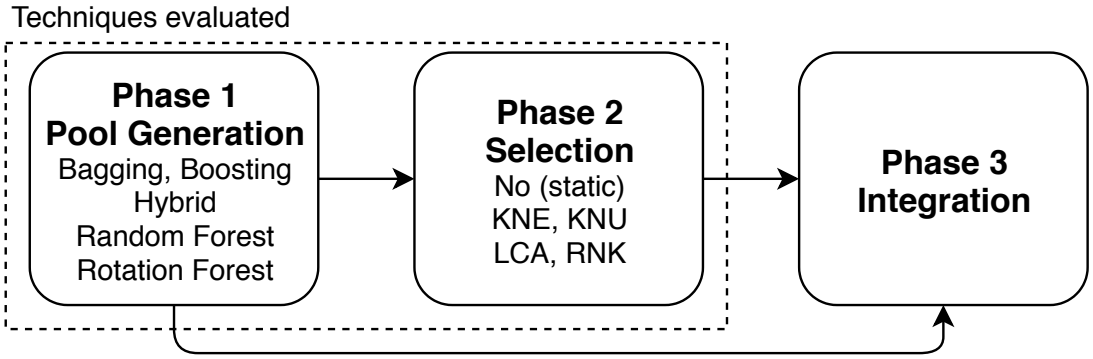


Figure 1 – The three MCS phases and the techniques evaluated in this work.

2.1.3 Ensembles

As shown in Figure 1, a typical ensemble has the following phases: pool generation, selection, and integration. The following subsections present the preliminary concepts of each phase.

2.1.3.1 Pool generators

The main challenge of the pool generation phase is to generate a pool of accurate and diverse classifiers (HANSEN; SALAMON, 1990). Homogeneous or heterogeneous base classifiers can achieve this diversification. Regarding the homogeneous pools, the diversity comes from different subsets of training data (Bagging, Boosting, or Hybrid), or using different features subspaces (Random Subspace Selection), or based on feature extraction (Rotation Forest).

2.1.3.2 Selection

The second phase of an ensemble is the base classifiers' selection to the prediction procedure, as shown in Figure 1. The main concepts of this phase are related to the type of selection and the notion of classifier competence (ability to predict a new sample correctly). The type of selection may be static (DIETTERICH, 2000), where the decision about the competence of the base learners occurs at the fitting time, or dynamic (GIACINTO; ROLI, 1999) when the decision occurs at prediction time.

The intuition behind the preference for dynamic over static selection is to select the most locally accurate classifiers to predict each unknown sample. A dynamic selection approach uses a set of samples in the neighborhood of the query sample, a competence measure, and a procedure to select the best local estimators. The dynamic selection approach selects samples in

the unknown sample neighborhood and computes the local competence of the base classifiers. Finally, according to the selection strategy, only the most competent base classifiers are used to predict the unknown sample (BRITTO JR *et al.*, 2014).

The dynamic selection approaches are classified by the selection methodology (KO *et al.*, 2008). According to this classification, there are two kinds of strategies: Dynamic Classifier Selection (DCS) and Dynamic Ensemble Selection (DES). The difference between them is the number of classifiers selected to predict each sample. DCS selects only the most competent base classifier, and DES selects a set of competent local classifiers.

Roy *et al.* (2018) is the most recent work we found that evaluated dynamic selection techniques to solve imbalance classification problems. As previous papers (XIAO *et al.*, 2012) that evaluated DS in the context of imbalanced learning, they test DS strategies based on different notions of competence measure. For example, Local Class Accuracy (LCA) considers the local class accuracy separately. The Modified Classifier Rank (RNK) ranks the classifiers. These two techniques are DCS. They also test two versions of K-Nearest Oracles (KNORA), which are DES techniques. Next, we briefly describe the four DS strategies obtained from Cruz *et al.* (2020) and adopted in this thesis.

- The Local Class Accuracy (LCA) (WOODS *et al.*, 1997; BRITTO JR *et al.*, 2014) gets the prediction of the test sample of each base classifier and, according to the predicted class, compute the class accuracy regarding only the predicted class. The LCA chooses the classifier with the higher class accuracy to predict the test sample.
- The Modified Classifier Rank (RNK) (SABOURIN *et al.*, 1993; BRITTO JR *et al.*, 2014) method ranks the accuracy of the base classifiers in the neighborhood of each test instance. The classifier with the highest accuracy is used to predict the test instance.
- The K-Nearest Oracles (KNORA) (KO *et al.*, 2008) techniques are inspired by the Oracle (KUNCHEVA, 2002) concept. The most promising are K-Nearest Oracles-Eliminate (KNE) and K-Nearest Oracles-Union (KNU). The KNE selects only the base classifiers with the perfect accuracy in the neighborhood of the test instance. On the other hand, in the KNU technique, the level of competence of a base classifier is measured by the number of correctly classified instances in the defined local region. In this case, every classifier that correctly classified at least one instance can vote for the final prediction.

This thesis proposes improving KNU to imbalanced datasets, the K-Nearest Oracles-Imbalanced Union (KNIU) (MELO JR *et al.*, 2019b). KNIU extends KNU replacing the

accuracy measure to compute the local competence of the base classifiers. Instead, it uses FA^2, a combination of F-measure and the square of accuracy.

The dynamic selection approaches require a Dynamic Selection Dataset (DSEL) to define the local regions of the feature space. This data is used to measure the competence of the base classifiers on each part of the feature space. The main challenge in the DSEL generation is to use a good part of the training data to obtain a good performance of the DS approach and keep the other part for the training the base classifiers (CRUZ *et al.*, 2015). The separation between the training data and the DSEL is essential to avoid overfitting. This task is even more difficult in an imbalanced dataset due to the lack of samples in the minority class (ROY *et al.*, 2018).

The integration is the last step of an ensemble, and it consists of applying the selected classifiers to recognize a given testing sample. In cases where all classifiers are used (without selection) or when a subset is selected, a fusion strategy is necessary. Majority voting is the most common fusion approach used by ensembles. Next, we present related works about ensemble classification approaches for credit scoring.

2.2 Related Works

This section starts with a review of the last five years of credit scoring ensemble approaches. Then, we present previous works that evaluate dynamic selection classification with imbalanced datasets, which one of them evaluated in the credit scoring problem. We compare our proposal with these two DES strategies adapted to imbalanced learning described next.

2.2.1 Credit scoring related works

We discuss in this section the papers of the last five years about load default prediction. Several works have been published in recent years using ensembles focusing on default loan prediction. Most of them do not use dynamic selection (GARCÍA *et al.*, 2019; HE *et al.*, 2018; SUN *et al.*, 2018; XIA *et al.*, 2018; ABELLÁN; CASTELLANO, 2017; XIA *et al.*, 2017) because of the Basel Accords regulatory constraints. Others (FENG *et al.*, 2018; HE *et al.*, 2018; ALA'RAJ; ABBOD, 2016a; ALA'RAJ; ABBOD, 2016b; XIAO *et al.*, 2016) evaluate dynamic selection classification.

Another important aspect we evaluate in these related works is the imbalance level of the datasets used in the experiments. The metric we use to measure it is the IR (ORRIOLS-PUIG;

BERNADÓ-MANSILLA, 2009), which is the cardinality of the majority class divided by the cardinality of the minority class. As Fernández *et al.* (2008), we consider a dataset as a low imbalanced dataset if it has an $IR < 3$. We evaluate it because the classification complexity of a low imbalanced is dataset is different from moderate imbalanced, $3 \leq IR < 9$, and high imbalanced, $IR \geq 9$, datasets (BROWN; MUES, 2012; MARQUÉS *et al.*, 2013).

García *et al.* (2019) explored the potential effects between sample types and the performance of classifier ensemble for credit risk and corporate bankruptcy prediction. The experiments on 14 real-life financial databases show that the ensembles' performance depends on the prevalent type of positive samples. However, half of the evaluated datasets, 7, were considered low imbalanced.

Feng *et al.* (2018) presented a dynamic ensemble model based on soft probability where the classifier selection was based on accuracy, precision, and different costs of type I error (the number of customers with bad credit classified as having good credit) and type II error. Experimental results showed that the proposed model outperforms bagging ensemble and random forest on several imbalanced credit data sets. However, six from ten datasets used to evaluate the techniques were low imbalanced datasets.

He *et al.* (2018) introduced a cascade model that resamples the credit scoring data sets according to their imbalance ratio and a threshold. Each adjusted data set is used for training several random forests and extreme gradient boosting as base classifiers. As the previous papers of this section, half of the datasets used to evaluate the proposed approach were low imbalanced.

Sun *et al.* (2018) proposed an ensemble for imbalanced credit evaluation based on the SMOTE and the bagging technique with different sampling rates. Sun *et al.* (2018) evaluated only one low imbalanced dataset.

Xia *et al.* (2018) designed a heterogeneous ensemble credit scoring model by integrating the bagging algorithm with the stacking method; despite the model introduced did not focus on class imbalance problems, it showed a good performance on moderately imbalanced datasets.

Abellán e Castellano (2017) showed that an ensemble built with the credal decision tree performs better than others based on more complex base learners trained on balanced and imbalanced data sets. Two-thirds, four of six, of the datasets used were low imbalanced.

Xia *et al.* (2017) introduced a sequential extreme gradient boosting model that incorporates a preprocessing step to scale the data and handle missing values. In addition, the

authors used a feature selection system to remove redundant variables. Four datasets of five evaluated in this paper were low imbalanced.

Ala'raj e Abbod (2016a) introduced a new combination approach based on classifier consensus that creates a ranking group as a fusion of individual classifiers. Experimental results showed that the consensus model achieves better performance in terms of the H-measure. Four of the five datasets of this paper were low imbalanced.

Xiao *et al.* (2016) propose an ensemble classification approach based on supervised clustering for credit scoring. Supervised clustering is employed to partition the data samples of each class into several clusters. Clusters from different classes are then pairwise combined to form some training subsets. The results showed that, compared to other ensemble classification methods, the proposed approach could generate base classifiers with higher diversity and local accuracy and improve the accuracy of credit scoring. Four of the five datasets used in this paper are low imbalanced.

Ala'raj e Abbod (2016b) used two preprocessing techniques, Gabriel Neighborhood Graph editing (GNG) and Multivariate Adaptive Regression Splines (MARS), to reduce the size of the data set by filtering samples and choosing the most relevant features. Both algorithms were combined with a consensus ranking approach. Five of the seven datasets used in this paper are low imbalanced.

Lessmann *et al.* (2015) updated the study of Baesens *et al.* (2003) and compared several novel classification algorithms to the state-of-the-art in credit scoring. This paper concluded that ensemble methods perform better than single artificial intelligence and statistical methods.

Table 1 summarizes studies in the literature on classifier ensembles used for credit scoring from 2015 to 2019. The comparison contains the number of datasets used, the percentage of datasets with IR under 3, sampling approaches used, such as RUS, SMOTE, and RAMO, column *Sampling*, whether the developed classifier ensembles are homogeneous or heterogeneous, column *Kind*, the type of selection, Static Selection (SS), or DS, and the pool generators adopted. As can be seen, only our previous paper (MELO JR *et al.*, 2019c) evaluated mainly moderated and high imbalanced datasets.

The central gap in previous papers is the evaluation of DS techniques in moderate or high imbalanced datasets. As these techniques suffer from skewed datasets, once they almost always use kNN to define the local region of the feature space, and most of real credit scoring

Table 1 – Approaches tracking credit scoring in literature

Ref.	Year	# datasets	% datasets w/ $IR \leq 3$	Sampling	Ensemble	
					Kind	Selection
(MELO JR *et al.*, 2019c)	2019	4	25	RUS, SMOTE, RAMO	Homog.	SS, DS
(GARCÍA *et al.*, 2019)	2019	14	50	-	Homog.	-
(FENG *et al.*, 2018)	2018	10	60	-	Heterog.	DS
(HE *et al.*, 2018)	2018	6	50	Based on RUS	Heterog.	SS
(SUN *et al.*, 2018)	2018	1	100	SMOTE	Homog	SS
(XIA *et al.*, 2018)	2018	4	75	-	Heterog.	-
(ABELLÁN; CASTELLANO, 2017)	2017	6	67	-	Homog.	-
(XIA *et al.*, 2017)	2017	5	80	-	Homog.	-
(ALA'RAJ; ABBOD, 2016b)	2016	7	71	-	Heterog.	DS
(XIAO *et al.*, 2016)	2016	2	100	-	Heterog.	DS
(ALA'RAJ; ABBOD, 2016a)	2016	5	80	-	Heterog.	DS
(LESSMANN *et al.*, 2015)	2015	8	75	-	Both	SS, DS

data are imbalanced, previous DS works may fail on real credit problems. Then, we look for strategies to improve DS performance in imbalanced credit data. Another gap in previous works is the discussion about regulatory aspects of the use of DS. Related to this topic, we propose an equivalence between DS and static ones.

2.2.2 Dynamic selection for imbalanced datasets related works

We present previous works that evaluate dynamic selection classification for imbalanced datasets of diverse fields, not only credit data.

2.2.2.1 KNORA-Union combined with SMOTE

As shown in subsection 2.1.3.2, KNORA-Union (BRITTO JR *et al.*, 2014) technique selects all classifiers that can correctly classify at least one sample in the local region of the query sample. The predictions of the selected classifiers are combined using a majority voting scheme, which considers that a base classifier can vote more than once when it correctly classifies more than one instance in a local region (KO *et al.*, 2008). For instance, if a given base classifier predicts the correct label for three samples belonging to some local region, it gains three votes for the majority voting scheme. The votes collected by all base classifiers are aggregated to obtain the ensemble decision.

It is important to note that KNU does not differentiate between the majority or minority classes. It means that, considering an imbalanced DSEL neighborhood of seven samples, with six examples of the majority class and only one from the minority class, a naive learner that always predicts the majority class will obtain a local competence of $6/7$.

That is why Roy et al. Roy *et al.* (2018) proposed over-sampling approaches to

balance the DSEL. This strategy includes diversity in DSEL concerning training data and solves the bias local competence computation described above once both minority and majority classes have the same amount of samples in the DSEL.

Besides the DSEL generation, Roy et al. Roy *et al.* (2018) also included an over-sampling step in the bagging iteration. Thus, the bagging ensemble train occurs with balanced data. In this thesis, we compare our proposal with the approach of this previous paper.

The main issue of this approach is the use of oversampling techniques to generate the DSEL. This approach can include noise in the DSEL. It means that the selection procedure to find the best approaches can suffer from this noise. Next, we describe another dynamic selection approach to imbalanced datasets.

2.2.2.2 DES-Multiclass Imbalance (DES-MI)

García *et al.* (2018) proposed a dynamic selection technique designed for multi-class imbalanced problems. Although this approach is defined for multi-class problems, this intuition also works for binary problems. It consists of two key components: the generation of balanced training data and selecting appropriate classifiers. We briefly describe these steps in the following paragraphs.

The first component proposed in García *et al.* (2018) is a hybrid sampling approach to balance the dataset used to train the classifiers pool. The main characteristic of this approach is the random size of the dataset provided for each base classifier. The method consists of: (i) define the new amount of the majority class randomly, reducing it using undersampling; (ii) use over-sampling to increase the number of samples of the minority class until the new size of the majority class.

The other component of DES-MI is a voting approach that weights the influence of each example in the local region according to the proportion of examples with the same class in the neighborhood of the query sample. The candidate learners who correctly classify more minority class examples belonging to the query sample local region are associated with a higher competence. Thus, the method uses a weighted accuracy to define the competence of each base learner. After computing the competence of all base learners, the method selects a percentage of most competent learners to perform the prediction.

Although García *et al.* (2018)'s approach is originally designed for multi-class classification, it can also be used for credit scoring, a classical binary classification problem.

It is important to mention that, different from this thesis, García *et al.* (2018) paper does not propose any performance measure, and it does not modify the samples used to compute the local competence of base classifiers.

To the best of our knowledge, these are the only efforts related to dynamic classification systems applied to imbalanced datasets. Motivated by the interest of developing novel approaches to the credit scoring problem, we develop novel strategies to help dynamic selection techniques handle imbalanced datasets. Next, we present the datasets and the machine learning algorithms used in this Thesis.

2.3 Datasets, algorithms, and performance measures used

This section shows the datasets and the algorithms used in Chapters 3, 4, 5, and 6.

2.3.1 Datasets

We use eleven real-world credit data in our experiments. Chapter 3 uses three of these eleven data. Chapters 4 uses eight datasets. Chapter 5 uses five real credit data. Moreover, finally, Chapter 6 uses seven datasets. The following paragraphs comment on these datasets, and Table 2 shows the details of these datasets.

The empirical evaluation of Chapter 3 includes three real-world credit scoring datasets. The *Australian Credit* (A), *German Credit* (G), and *Japanese Credit* (J) come from the UCI Machine Learning Repository[1]. They have widely used datasets on credit scoring papers, and they are publicly available.

From these three datasets, the following chapters use only the German one. Once the central aspect evaluated is the imbalanced effect on the classification performance, the Australian and Japanese datasets do not fit the requirements.

In Chapter 4, we perform the comparison by exploiting eight real-world credit scoring datasets. Two datasets, German and Default, are provided by the UCI machine learning repository[2]. PPDai dataset comes from a Chinese internet finance enterprise named PaiPaiDai[3]. The Iranian dataset comes from paper Sabzevari *et al.* (2007). The private dataset comes from

[1] UCI repository available at https://archive.ics.uci.edu/ml/index.php
[2] https://archive.ics.uci.edu
[3] https://www.ppdai.com

a financial institution in Brazil[4]. GiveMe[5] comes from a Kaggle competition. LC2015Q123 LC2015Q4 are the last ones and contain loans of 36 months and a low-interest rate of the three first quarters and the fourth quarter in 2015, respectively, from Lending Club[6].

Chapter 5 uses five credit data. Four of them are also on Chapter 4 list: Default, PPDai, GiveMe, and Iran. Additionally, this chapter also uses another dataset from Lending Club, the first quarter of 2017.

Finally, chapter 6 uses all the datasets from Chapter 4, except for the LC2015Q4, the last quarter of 2015 from Landing Club.

Table 2 shows the details of these datasets. We use the Imbalance Ratio (IR) measure, the cardinality of the majority class divided by the minority class's cardinality, to sort the datasets from the less imbalanced to the most imbalanced. In the first one, Australian, the number of samples of the majority class is 1.24 times higher than the number of samples of the minority class. In the last one, LC2015Q4, the majority class has 93.78 times more samples than the minority class. Readers may notice that we only use balanced datasets, with an imbalanced ratio under 2, such as Australian and Japanese, in Chapter 3, where the imbalanced level of the data is produced by undersampling the original data.

Table 2 – Datasets description

Chapters	**Dataset**	**#Samples**	**#Features**	**# Categ. Features**	**Imbalanced Ratio (IR)**
3	Australian	690	14	7	1.24
3	Japanese	690	15	9	1.24
3, 4, 6	German	1,000	20	13	2.33
4, 5, 6	Default	29,892	24	3	3.52
4, 5, 6	PPDai	55,596	29	7	6.74
4, 6	Private	4,976	56	18	9.05
4, 5, 6	GiveMe	150,000	10	0	13.96
4, 5, 6	Iran	997	0	27	19.77
4, 5	LC2017Q1	95,633	72	6	77.45
6	LC2015Q123	23,677	76	11	80.92
4	LC2015Q4	11,847	76	11	93.78

[4] The citations, observations, analyzes, and conclusions related to any references to this Brazilian financial institution contained in this academic work, and their eventual implications, are the sole responsibility of the first author and do not necessarily represent the thinking or agreement of the institution or its administrators.

[5] https://www.kaggle.com/c/GiveMeSomeCredit

[6] https://www.lendingclub.com

2.3.2 Credit Scoring approaches

This thesis investigates a wide range of credit scoring classification approaches. We define our approaches based on previous credit scoring works (BROWN; MUES, 2012; MARQUÉS *et al.*, 2012) and imbalanced dynamic selection papers (ROY *et al.*, 2018). Next, we describe each classification approach we used in this thesis.

2.3.2.1 Base Classifiers

Our credit scoring benchmarks list starts with Logistic Regression (LOGR). This binary classifier is a trendy statistical model in commercial credit scoring. It models the relationship between independent variables and the response variable using a logistic function.

The Support Vector Machine (SVM) method constructs a hyperplane to split the two classes of borrowers. This thesis uses the linear version of SVM, the Linear Support Vector Machine (LSVM), and the non-linear kernels of SVM, such as $poly$ and rbf.

The next classification approach is the multilayer perceptron Artificial Neural Networks (ANN). It employs sigmoidal functions to determine the model parameters by minimizing some loss-function. We consider ANNs with logistic activation function in the hidden and output layers.

We also test the kNN (COVER; HART, 1967) classifier. This classifier uses the spatial location of known samples to predict an unknown sample. It chooses, among the nearest neighbors, the most frequent class.

The next classification approach evaluated is Decision Trees (TREES) (QUINLAN, 1986). It uses a decision tree to go from observations about an item to conclusions about an unknown sample.

The following classification approach evaluated is Extreme Learning Machines (ELM) (HUANG *et al.*, 2006). It is a learning algorithm with a single-hidden layer feed-forward neural network (SLFN) that randomly chooses hidden nodes and analytically determines the output weights of the SLFN. Bequé e Lessmann (2017) evaluated this approach for customer credit risk management.

Next, we test a Fuzzy Pattern Tree (FUZZY) classifier (HUANG *et al.*, 2008). A fuzzy pattern tree is a hierarchical, tree-like structure whose inner nodes are marked with generalized (fuzzy) logical and arithmetic operators and whose leaf nodes are associated with

fuzzy predicates on input attributes.

We also test Discriminant Analysis (DA) classifier (MCLACHLAN, 2004). It is a simple classifier that defines a decision surface to perform predictions. We test the two types of DA, Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA).

Next, we evaluate a genetic programming symbolic classifier (GENET). We evaluate *gplearn*, an estimator that begins by building a population of naive random formulas to represent a relationship. The formulas are represented as tree-like structures with mathematical functions being recursively applied to variables and constants. Each successive generation of programs is evolved from the one that came before int by selecting the fittest individuals from the population to undergo genetic operations (STEPHENS, 2016).

The last classification approach evaluated is Gaussian naive Bayes (GNB). It is an extension of naive Bayes that assumes that the features have a Gaussian distribution. Next, we show the ensemble list used in this thesis.

2.3.2.2 Regular Ensembles

Our ensemble list starts with the ensembles used in Brown e Mues (2012), Random Forest (RNDF) (LIAW *et al.*, 2002), and Gradient Boosting (GB). RNDF is a modified decision tree-based bagging ensemble that uses a random subset of the features, and GB is a decision tree boosting ensemble. We keep RNDF, but we replace GB with eXtreme Gradient Boosting (XGB) (CHEN; GUESTRIN, 2016), an improvement of GB. Besides, we test Adaptive Boosting (AdaBoost), a decision tree-based boosting ensemble. Table 5 shows the parameter list of the ensembles and their tested values.

2.3.2.3 Imbalanced Ensembles

An imbalanced ensemble is an ensemble designed that uses some sampling technique to balance the data before the base learners training step. We use the imbalanced ensembles available on Lemaître *et al.* (2017) and implement others.

We start with the bagging ensemble (BREIMAN, 1996). Bagging, also known as Bootstrap aggregating, constructs bootstrap samples from the training data to produce T base models. Bagging uses a majority voting to fusion the T predictions. The first imbalanced ensemble we use is Balanced Bagging (Bagging + RUS) (BBAG). It includes an additional step to balance the training set using RUS. We also use Bagging SMOTE (Bagging + SMOTE)

(BGSM), the bagging ensemble, with a SMOTE step to balance the training set.

We also use two imbalanced ensembles using Random Forest. We use Balanced Random Forest (Random Forest + RUS) (BRND) (CHEN *et al.*, 2004), the combination of this ensemble with RUS, and the Random Forest SMOTE (Random Forest + SMOTE) (RFSM), the combination of RNDF with SMOTE over-sampling technique.

Next, we test an imbalanced ensemble that uses the rotation forest. This ensemble applies principal component analysis on bootstrap samples to rotate the training data. Based on BRND, we develop a Balanced Rotation Forest (Rotation Forest + RUS) (BROT), the combination of rotation forest and RUS, to execute the experiments of this thesis.

The next four imbalanced ensembles evaluated are derived from adaptive boosting, also known as AdaBoost. Two ensembles are the combination of AdaBoost and the RUS and SMOTE preprocessing techniques. They are RUS Boost (AdaBoost + RUS) (RUSB), and SMOTE Boost (AdaBoost + SMOTE) (SMTB), respectively. They use preprocessing techniques to balance the data in each step of the boosting algorithm. The last imbalance ensemble, Easy ensemble (Bagging of AdaBoost + RUS) (EASY) (LIU *et al.*, 2009), is a bagging ensemble that uses AdaBoost ensemble as base classifiers. It also uses RUS to balance the data before the training step. The last item of the imbalanced learning strategies is a cost-sensitive version of AdaBoost, called Calibrated Adaboost with Minimum Expected Cost (CMEC) (NIKOLAOU *et al.*, 2016).

2.3.3 Performance measures

A correct selection of evaluation measures is critical to avoid biased results. For instance, the percentage of correctly classified measure is widely used in classification but is not appropriate to an imbalanced dataset since a naive classifier always predicting the majority class achieves a high score.

We evaluate six metrics to measure the predictive accuracy of the classifiers: Area under the ROC curve (AUC), H-measure, balanced accuracy (BAcc), G-mean, F-measure, and True Positive Rate (TPR). As in other work about imbalanced classification, we consider the minority class, namely the bad credit, as the positive class to avoid bias results in F-measure. In the next paragraphs, we present some essential measure elements and comment briefly on each performance measure evaluated in this chapter.

Based on the elements of the confusion matrix, true positive (TP), false negative (FN),

true negative (TN), and false-positive (FP), we can define the precision, $Precision = \frac{TP}{TP+FP}$, the recall, or sensitivity or true positive rate (TPR), $Recall = \frac{TP}{TP+FN}$, the specificity or true negative rate (TNR), $Specificity = \frac{TN}{TN+FP}$, and the false positive rate (FPR), $FPR = 1 - TNR = \frac{FP}{TN+FP}$.

We now describe the performance metrics used in this chapter. AUC is an extensively used evaluation measure obtained from the area under the ROC curve. The x-axis of the ROC curve represents the FPR, and the y-axis represents TPR (sensitivity). The balanced accuracy (BAcc) is the arithmetic mean of the positive and negative class accuracy, as shown in Eq. (2.1). The F-measure is the weighted harmonic mean between precision and recall, as shown in Eq. (2.2). The β in the F-measure formula is a hyper-parameter for weighting differently the precision and recall. In this chapter, we evaluate three values for β: $[1, 5, 35]$. The first, 1, gives equal importance to precision and recall. The second, 5, is based on the misclassification cost difference evaluated in West (2000). The last, 35, is based on Altman *et al.* (1977). Finally, Eq. (2.3) shows the G-mean, the geometric mean of sensitivity and specificity.

$$\text{Balanced Accuracy} = \frac{\frac{TP}{TP+FN} + \frac{TN}{TN+FP}}{2} \tag{2.1}$$

$$\text{F-measure} = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{2.2}$$

$$\text{G-mean} = \sqrt{Sensitivity \times Specificity} \tag{2.3}$$

H-measure is a threshold-varying evaluation metric proposed by Hand (2009). This measure overcomes the AUC deficiency in the use of different misclassification costs distributions for different classifiers. H-measure gives a normalized classifier assessment based on expected minimum misclassification loss, ranging from zero to one for a random and perfect classifier.

Next, we present an imbalanced credit scoring benchmark to evaluate the performance of novel approaches on the imbalanced credit scoring problem.

3 IMBALANCED CREDIT SCORING BENCHMARK

This chapter presents the first contribution of the thesis that is a benchmark of classification approaches for imbalanced credit scoring problem published in Melo Jr *et al.* (2019a). Some works in the literature (BROWN; MUES, 2012; MARQUÉS *et al.*, 2013) evaluate the imbalanced credit scoring problem, but, since new imbalanced learning techniques have been proposed recently (HAIXIANG *et al.*, 2017), we are encouraged to investigate how these recent techniques in machine learning may improve the credit scoring prediction performance compared to the well-known state of art approaches.

In this way, we perform an empirical evaluation with a broad range of hyperparameter tests of 19 novels and establish prediction methods. Besides, we seek to answer the following research questions: **RQ3.1) "How do the more recent techniques in machine learning improve the credit scoring prediction performance compare to well-known state-of-the-art approaches?"**; **RQ3.2) "Is there any better approach for each specific level of imbalanced data?"**.

Our experiments use three real credit scoring datasets to test hundreds of parameter combinations of each classification approach. We evaluate 11 base classifiers, such as logistic regression (WRIGHT, 1995), decision trees (BREIMAN *et al.*, 1984), neural networks (MELO JR *et al.*, 2019c), linear and quadratic discriminant analysis (LACHENBRUCH; GOLDSTEIN, 1979), nearest neighbors (DUDA *et al.*, 1973), fuzzy methods (SENGE, 2014), genetic algorithms (WHITLEY, 1994), support vector machines (VAPNIK, 2013), extreme learning machine (HUANG *et al.*, 2004) and Naive Bayes (RISH *et al.*, 2001). We also evaluate ensembles and imbalanced learning approaches, such as resampling and cost-sensitive solutions. We measure the performance of the classifiers using Area Under the receiver-operating-characteristics Curve (AUC) and rank them using Friedman's average rank. We also use statistical significance tests among all classifiers. As a result, we identify two approaches that presented excellent results in all imbalanced ratios tested.

The main contributions of this chapter are two-fold. First, we evaluate the performance of eleven base classifiers, three ensembles, and five imbalanced approaches in twelve imbalanced versions of three real credit scoring datasets. Second, we define, for each classification approach, several parameter combinations to evaluate the imbalanced scenarios. Next, we show the experiments' details, comment on the results, and present the conclusions.

Table 3 – List of imbalanced ratios datasets tested

Paper	Dataset versions						
(BROWN; MUES, 2012)	IR (% defaults)	2.3 (30%)	5.7 (15%)	9 (10%)	19 (5%)	39 (2.5%)	99 (1%)
(MARQUÉS *et al.*, 2013)		4 (20%)	6 (14.3%)	8 (11.1%)	10 (9.1%)	12 (7.7%)	14 (6.7%)

3.1 Experimental setup

We use a Microsoft Cloud Infrastructure with 16 cores and 64 GB of RAM to run all experiments. Next, we present the methodology used, the performance measure adopted, the classifier's approaches evaluated, and the statistical comparison performed.

3.1.1 Methodology

This experiment we performed has three steps: data preparation, datasets versions generation, and classifier evaluation. Next, we discuss each one of them briefly.

Although it is a critical phase of the machine learning process, we decide to reduce the data preparation steps as much as possible. This decision aims to reduce the influence of preprocessing on the results. We apply z-score standardization for numeric features. Additionally, we use binary encoding to transform categorical features into binary ones.

In the next step, we produce several different skewed versions of the three datasets by randomly removing the minority class samples. These imbalanced datasets versions are generated by randomly removing samples of the minority class. We decide to use the same imbalance levels used in Brown e Mues (2012) and Marqués *et al.* (2013) to compare with previous works. Table 3 shows the IRs tested in this paper and the percentage of defaulters for each dataset.

The last step is the model training and evaluation. For each classifier's hyperparameter combination, we execute *k-fold*, with $k = 3$, to find the best hyperparameter setup of each classification approach. We decide to use a low k-fold because of the reduced amount of minority class samples in the high IRs versions of the datasets. Although we use stratified k-fold, a higher k-fold, it would not be possible to split the modified dataset into train and test parts and keep at least one sample of the minority class on each fold. Finally, we compute the average AUC of the three folds for each classification approach.

3.1.2 Performance measure

We use the AUC measure, which considers the area under the Receiver Operating Characteristic curve, as the performance indicator. This performance measure illustrates the trade-off between the true-positive rate and the false-positive rate. We decide to use this measure because scientific papers and financial institutions widely use it. The highest benefit of this measure is the capability to perform gradual risk exposure, varying the threshold of accepted/rejected loans.

3.1.3 Classifier approaches and hyperparameter tunning

We evaluate eleven base classifiers, three cost-sensitive ensembles, and five imbalanced ensembles techniques. For each strategy, we test several configurations to find the best parameter setup. For each classifier, we test several parameter combinations. Two other guide us in the grid search parameter definition: i) we attempt to include a cost-sensitive imbalance solution to test the ability of the classifier to handle the imbalanced datasets, and ii) we use a regularization constraint to avoid overfitting. In the following paragraphs, we explain how we choose estimators. For each approach, we present the list of parameters evaluated and, for each parameter, the list of values tested in Tables 4, 5, and 6.

Table 4 – List of base classifiers parameters

Estim	Parameters	Values	Estim	Parameters	Values
kNN	n neighbors weights p algorithm leaf size	[1, 5, 9, ... 45, 49] [distance, uniform] [1, 2] [ball tree,kd tree,auto] [3, 5, 10, 15]	ELM	hidden layer(n hidden, activation function)	MLP([10, 20,...100], [sin square, triangular, hyperb tan, hard limit]) RBF([20, 40,...200], rbf width = 0.1)
SVM	C class weight tol max iterations	[0.01, 0.03, 0.1, 0.3, 1] [balanced , none] [0.0001, 0.001] [1000, 2000]	LOGR	C class weight solver tol	[0.01, 0.03, 0.1, 0.3, 1] [balanced , none] [liblinear, saga] [0.0001, 0.001]
TREES	max depth class weight splitter min samples split min samples leaf max features min imp decrease	[none, 5, 10, 15, 30] [none, balanced] [best, random] [2, 4, 6] [1, 2, 3] [sqrt, log2, None] [0., 0.1, 0.3]	ANN	solver hidden layer sizes tol activation alpha max iterations learning rate max iterations	[lbfgs, sgd, adam] [(20), (20, 20)] [0.0001, 0.001] [logistic, tanh, relu] [0.001, 0.0001] [100, 500] [constant,invscaling,adaptive] [100, 200, 400]
FUZZY	max depth num candidates num slaves	[3, 5, 7, 9] [1, 2, 3] [1, 2, 3]	QDA	priors tol reg param	[none, [0.5, 0.5]] [0.0001, 0.001] [0., 0.1, 0.3]
GENET	population size generations tournament size stopping criteria p crossover init method	[100, 400, 1600] [10, 100] [10, 20, 40] [0.5, 1.0, 2.0] [0.5, 0.7, 0.9] [grow, full, half half]	LDA	solver priors shrinkage tol	[lsqr, eigen] [none, [0.5, 0.5]] [none, auto] [0.0001, 0.001]
			GNB	priors	[none, [0.5, 0.5]]

Table 5 – List of ensemble parameters

Ensemble	Parameters	Values
RNDF	n estimators	[100, 400, 1600]
	max depth	[1, 3, 6, 9]
	class weight	[balanced, balanced subsample, none]
	bootstrap	[True, False]
	min samples split	[2, 3, 4]
	max features	[auto, log2]
	min impurity decrease	[0., 0.1, 0.3]
XGB	min child weight	[1, 4, 7, 10]
	max depth	[1, 3, 6, 9]
	colsample bytree	[0.5, 0.7, 0.9]
	subsample	[0.5, 0.7, 0.9]
	gamma	[0.0, 0.2, 0.4]
	learning rate	[0.5, 1, 2]
	scale pos weight	[1, default ratio/(1-default ratio), sqrt(default ratio/(1-default ratio))]
	n estimators	[100, 400, 1600]
ADAB	base estimator	TREE(md=[1,3,6], msl=[2,5], cw=[None,balanced], mid=[0,0.2])
	n estimators	[100, 400, 1600]
	learning rate	[0.5, 1, 2]

md: max depth; msl: min samples leaf, mid: minimal impurity decrease, cw: class weights

Table 4 shows the base classifiers list. To define this list, we first collect all the base classifiers used in Brown e Mues (2012), except one of them. Instead of C4.5 decision trees, we use an optimized version of classification and regression trees (CART) implementation of decision trees available on scikit-learn[1]. Besides this, the list includes logistic regression (LOGR), a trendy statistical model in commercial credit scoring systems; artificial neural networks (ANN), the multilayer perceptron implementation of this classifier inspired in biological neural networks; linear discriminant analysis (LDA), and quadratic discriminant analysis (QDA), two statistical classifiers also used for dimensionality reduction; linear support vector machines (SVM), a fast classifier that uses a hyperplane that differentiate the two classes; and k-nearest neighborhood (kNN), a classifier that uses the neighbors of the query sample to choose its class.

Based on the list of classifiers evaluated by Louzada *et al.* (2016), we also include the Gaussian naive Bayes (GNB), a classifier that considers the features have a Gaussian distribution; fuzzy logic (SENGE, 2014) (FUZZ[2]), a classifier that groups elements in fuzzy sets; and GENET (GNET[3]), a meta-heuristic inspired by the process of natural selection. We finished the list of base classifiers with extreme learning machines (ELM[4]), a recently proposed type of artificial neural network for customer credit risk management evaluated in Bequé e Lessmann (2017). For all classifiers, we use public implementations available on scikit-learn or other sites on the Internet.

[1] https://scikit-learn.org/stable/

[2] FuzzyPatternTree implementation obtained from https://github.com/sorend/fylearn

[3] GENET implementation obtained from http://gplearn.readthedocs.io/en/stable/index.html

[4] ELM implementation obtained from https://github.com/dc9000lambert/Python-ELM

Table 6 – List of imbalanced learning strategies

Approach	Parameters	Values
SMTB	base estimator	TREE(md=[1, 3], msl=[1, 3])
	n samples	[the amount necessary to balance]
	n estimators	[50, 100]
RUSB	base estimator	TREE(md=[1,3,6], msl=[2,5], cw=[none,balanced], mid=[0,0.2])
	n estimators	[100, 400, 1600]
	with replacement	[true, false]
	n samples	[the amount necessary to balance]
	learning rate	[0.5, 1, 2]
BBTR	base estimator	TREE(md=[1,3,6], msl=[2,5], cw=[none,balanced], mid=[0,0.2])
	n estimators	[100, 400, 1600]
	ratio	[auto]
	with replacement	[true, false]
BBLR	base estimator	LOGR(C=[.01,.03,.1,.3,1], solver=[liblinear, saga], cw=[None,balanced])
	with replacement	[true, false]
	n estimators	[100, 400, 1600]
	ratio	[auto]
CMEC	base learner	TREE(md=[1,3,6], msl=[2,5], cw=[none,balanced], mid=[0,0.2])
	ensemble size	[100, 400, 1600]
	FP cost	[1, 1 - default rate/(1 - default rate)]
	FN cost	[1, default rate/(1 - default rate)]

md: max depth; msl: min samples leaf, mid: minimal impurity decrease, cw:class weights

The imbalanced learning strategies list is composed of public and available implementations of approaches to imbalanced learning. We start with the iterative ensembles: the combination of AdaBoost with two well-known resampling techniques, SMOTE and RUS, SMOTEBoost (SMTB) (CHAWLA *et al.*, 2003) and RUSBoost (RUSB) (SEIFFERT *et al.*, 2008). We also test a parallel approach, Balanced Bagging[5] (BB) with undersampling. We test Balanced Bagging of Decision Trees (BBTR) and Balanced Bagging of Logistic Regression (BBLR). The last item of the imbalanced learning strategies is a cost-sensitive version of AdaBoost, called CMEC (NIKOLAOU *et al.*, 2016). Table 6 shows the list of parameters of each approach. We test all combinations of all parameters presented in Tables 4, 5, and 6.

3.1.4 Statistical comparison of classifiers

Similar to several previous works (BROWN; MUES, 2012; MARQUÉS *et al.*, 2012; ABELLÁN; CASTELLANO, 2017), we use Friedman's test (FRIEDMAN, 1940) to compare the performance of the different classifiers. The Friedman test statistics uses the average rank (AR) performances of the classification techniques on each dataset. Besides the average ranked performances, this test also considers the number of classifiers (K) and the number of datasets used (D). This test is distributed according to the Chi-square distribution with K-1 degrees of freedom and a determined probability of error, p. When the value of the Friedman test statistic is

[5] BB implementation obtained from http://contrib.scikit-learn.org/imbalanced-learn/stable/index.html.

higher than the threshold, we can reject the null hypothesis that there is no difference between the techniques.

We also use the post hoc Nemenyi test (NEMENYI, 1962), which is applied to report any significant differences between individual classifiers. This test states that the performances of two or more classifiers are significantly different if their average ranks differ by at least the critical difference defined by *Critical Values for The Studentized Range Distribution* table (KOKOSKA; NEVISON, 1989), hereafter called only Studentized table. Finally, the results from Friedman's statistics and the Nemenyi post hoc tests are displayed in the same way of previous works (LESSMANN *et al.*, 2008). These diagrams show the ranked performances of the classification techniques, along with the critical difference, to clearly show the techniques that report the average rank (AR) value significantly different from other classifiers.

3.2 Results

Tables 7 and 8 reports the AUCs of all 19 classifiers on the three credit scoring datasets together with the results of previous works. Table 7 compares our results with the results obtained by Brown e Mues (2012). Since this previous work does not evaluate the Japanese dataset, we present the comparison of the Australian and German datasets only. On the other hand, we can compare the three datasets with Marqués *et al.* (2013) results in Table 8. For the following two tables, we paint the cells with shades of gray to highlight when our result is better than the previous one, indicating the difference. The darkest shade of gray represents that our results are at least 0.2 greater than the previous one. The second darkest means that our AUC results are still better, but the difference is between 0.1 and 0.2. The lightest shade of gray means that our result is still better, but the difference is under 0.1. The white cells indicate that the previous work gets a better result than ours.

Table 7 has two parts. The top part shows our and Brown's work results per classifier. The bottom part presents the results of the classifiers tested only by our work, compared with the best results of the previous work. To illustrate the shades of gray, we can observe SVM results. For $IR = 99$ and $IR = 39$, our SVM results are higher than 0.2 of the previous one. However, for $IR = 19$, only the German dataset has a difference above 0.2. The Australian difference is under 0.1. In the versions with $IR = 9$ and $IR = 5.7$, the improvement of our results is under 0.1. Finally, our SVM results are worse than the previous work for the version of the datasets with $IR = 2.3$.

Table 7 – Australian (A) and German (G) dataset AUC results of our approach (OA rows) compared with Brown e Mues (2012) (BR rows) in each IR. The upper part compared per classifier and the bottom part against the best results of the previous work.

Imbal. Ratio		99		39		19		9		5.7		2.3	
Appr	WRK	A	G	A	G	A	G	A	G	A	G	A	G
LDA	OA	0.908	0.763	0.904	0.750	0.929	0.783	0.934	0.791	0.932	0.789	0.929	0.798
	BR	0.868	0.583	0.818	0.626	0.935	0.738	0.945	0.742	0.938	0.76	0.944	0.791
LOGR	OA	0.885	0.709	0.899	0.772	0.931	0.769	0.925	0.777	0.924	0.782	0.928	0.798
	BR	0.500	0.500	0.500	0.551	0.500	0.757	0.500	0.766	0.918	0.74	0.906	0.767
ANN	OA	0.922	0.720	0.913	0.770	0.929	0.764	0.920	0.770	0.925	0.777	0.931	0.794
	BR	0.867	0.542	0.700	0.592	0.894	0.683	0.897	0.724	0.921	0.701	0.921	0.727
QDA	OA	0.804	0.642	0.848	0.661	0.857	0.684	0.900	0.727	0.909	0.734	0.915	0.759
	BR	0.52	0.500	0.516	0.500	0.597	0.500	0.849	0.528	0.654	0.597	0.855	0.718
RNDF	OA	0.924	0.812	0.919	0.780	0.936	0.792	0.939	0.788	0.935	0.788	0.935	0.802
	BR	0.901	0.671	0.879	0.691	0.932	0.752	0.932	0.772	0.941	0.769	0.937	0.800
SVM	OA	0.881	0.768	0.892	0.738	0.923	0.772	0.926	0.784	0.928	0.783	0.927	0.794
	BR	0.500	0.500	0.652	0.500	0.878	0.500	0.906	0.768	0.910	0.750	0.951	0.819
TREE	OA	0.830	0.696	0.836	0.687	0.875	0.718	0.888	0.713	0.886	0.716	0.904	0.752
	BR	0.500	0.642	0.587	0.614	0.754	0.565	0.919	0.641	0.916	0.652	0.918	0.712
XGB	OA	0.916	0.774	0.921	0.795	0.934	0.775	0.933	0.774	0.928	0.766	0.933	0.795
GB	BR	0.745	0.594	0.883	0.741	0.931	0.766	0.938	0.753	0.948	0.750	0.949	0.772
kNN	OA	0.811	0.645	0.847	0.717	0.898	0.748	0.926	0.759	0.925	0.752	0.925	0.777
	BR	0.900	0.636	0.878	0.693	0.923	0.758	0.923	0.785	0.926	0.781	0.930	0.793
ADAB	OA	0.860	0.767	0.879	0.736	0.907	0.723	0.921	0.754	0.914	0.753	0.926	0.786
BBLR			0.766	0.869	0.724	0.932	0.773	0.927	0.775	0.925	0.787	0.924	0.799
BBTR					0.745	0.937	0.764	0.922	0.786	0.938	0.789	0.933	0.806
CMEC				0.932	0.763	0.929	0.730	0.916	0.736	0.926	0.747	0.917	0.762
ELM		0.613	0.605	0.626	0.578	0.738	0.558	0.818	0.594	0.870	0.605	0.903	0.669
FUZZ		0.768	0.535	0.871	0.557	0.862	0.689	0.902	0.728	0.895	0.724	0.910	0.761
GNET		0.792	0.732	0.778	0.662	0.707	0.618	0.713	0.599	0.706	0.583	0.690	0.580
GNB		0.587	0.535	0.769	0.629	0.884	0.707	0.896	0.716	0.893	0.730	0.895	0.746
RUSB		0.888	0.825	0.903	0.764	0.914	0.752	0.926	0.765	0.919	0.764	0.931	0.784
SMTB		0.792	0.669	0.860	0.710	0.902	0.707	0.901	0.717	0.900	0.716	0.907	0.754

We observe from the evaluation of Table 7 that an exhaustive hyperparameter search on the training phase produces different results in different imbalanced levels. We achieve better results on higher imbalanced levels, while the results are almost the same on less imbalanced levels.

Table 8 compares the results of this work with the best Marqués *et al.* (2013) results, marking the best result in bold. Unlike Brown e Mues (2012), we do not test the resampling methods evaluated by Marqués *et al.* (2013). We decided it because Brown's results are better than Marqués ones in similar dataset imbalanced versions, e.g., at the imbalanced ratios of 8, 9, and 10. The comparison results with Marqués et al. Marqués *et al.* (2013) show that this decision is correct. As we can see in 8, our results are better than the previous work in almost all approaches tested. We discuss in the following sections how our approach answers the three research questions presented previously.

RQ3.1) *The performance of more recent techniques in machine learning*

This section addresses our first research question related to the gains of recent techniques in machine learning for credit scoring prediction. To answer this question, we analyze

Table 8 – AUC results of this paper (all rows except first) compared with the best results of Marqués *et al.* (2013) (first row) in each IR. (Australian(A), German(G) and Japanese (J) datasets)

IR	14			12			10			8			6			4		
Appr/DS	A	G	J	A	G	J	A	G	J	A	G	J	A	G	J	A	G	J
Marqués et al.	0.87	0.70	0.82	0.88	0.70	0.88	0.87	0.72	0.89	0.89	0.75	0.88	0.87	0.72	0.88	0.89	0.74	0.89
LDA	0.92	0.77	0.90	0.93	0.78	0.89	0.93	0.78	0.90	0.93	0.79	0.89	0.92	0.79	0.90	0.93	0.79	0.90
LOGR	0.92	0.78	0.89	0.92	0.78	0.89	0.93	0.77	0.89	0.92	**0.79**	0.89	0.93	**0.80**	0.90	0.93	**0.80**	0.90
ANN	0.92	0.77	0.87	0.92	0.78	0.88	0.93	0.77	0.88	0.92	0.79	0.88	0.93	0.80	0.90	0.93	0.79	0.89
QDA	0.87	0.66	0.85	0.89	0.72	0.77	0.90	0.71	0.81	0.89	0.72	0.83	0.91	0.74	0.86	0.91	0.75	0.84
RNDF	0.92	0.77	0.92	**0.94**	0.79	0.90	**0.94**	**0.79**	0.91	0.93	0.79	0.90	0.93	0.79	**0.91**	**0.94**	0.79	0.91
SVM	0.92	0.76	0.89	0.92	0.77	0.88	0.92	0.77	0.89	0.92	0.78	0.89	0.92	0.79	0.90	0.93	0.78	0.90
TREE	0.86	0.69	0.87	0.87	0.71	0.86	0.90	0.71	0.87	0.88	0.71	0.85	0.89	0.72	0.87	0.90	0.72	0.87
XGB	**0.93**	0.78	**0.92**	0.93	0.78	**0.91**	0.94	0.76	0.91	0.92	0.77	**0.91**	**0.94**	0.79	0.91	0.94	0.79	**0.91**
kNN	0.90	0.72	0.89	0.92	0.75	0.88	0.92	0.75	0.89	0.92	0.75	0.89	0.92	0.76	0.90	0.92	0.77	0.88
ADAB	0.93	0.72	0.87	0.90	0.77	0.88	0.90	0.75	0.87	0.93	0.76	0.88	0.92	0.77	0.87	0.92	0.77	0.88
BBLR	0.91	0.76	0.89	0.93	**0.80**	0.90	0.92	0.78	**0.92**	0.91	0.79	0.90	0.93	0.79	0.90	0.93	0.80	0.90
BBTR	0.91	**0.78**	0.91	0.93	0.79	0.90	0.93	0.79	0.90	0.93	0.78	0.90	0.93	0.79	0.90	0.94	0.79	0.90
CMEC	0.90	0.75	0.88	0.91	0.74	0.87	0.92	0.74	0.87	0.92	0.74	0.87	0.92	0.74	0.88	0.92	0.75	0.89
ELM	0.75	0.59	0.72	0.77	0.59	0.72	0.83	0.59	0.74	0.83	0.60	0.78	0.85	0.61	0.80	0.89	0.63	0.84
FUZZ	0.88	0.72	0.87	0.89	0.72	0.89	0.91	0.74	0.89	0.90	0.72	0.87	0.90	0.72	0.89	0.91	0.74	0.88
GNET	0.72	0.60	0.73	0.68	0.61	0.74	0.72	0.59	0.76	0.70	0.60	0.74	0.69	0.59	0.71	0.68	0.58	0.73
GNB	0.87	0.70	0.76	0.89	0.72	0.76	0.88	0.71	0.79	0.89	0.74	0.80	0.89	0.75	0.84	0.90	0.75	0.84
RUSB	0.93	0.74	0.90	0.91	0.77	0.91	0.91	0.76	0.90	**0.93**	0.77	0.90	0.93	0.78	0.90	0.93	0.77	0.90
SMTB	0.89	0.72	0.85	0.90	0.72	0.86	0.90	0.71	0.85	0.89	0.72	0.87	0.90	0.73	0.86	0.91	0.74	0.87

the performance of the tested approaches with the Friedman test statistic. Using $p = 0.01$ and 18 degrees of freedom, the number of classifiers tested minus 1, the chi-square critical value for the Friedman test is 34.08. We present in the first line of Table 9 the Friedman test statistic for all dataset versions. Table 9 also shows the Friedman score (average rank (AR)) of each classification approach in each imbalanced level. Random Forest gets the highest AR on ten of twelve different imbalance ratios. XGB gets the other two highest AR.

We can conclude this section by confirming the results of Brown e Mues (2012) about the best approaches for imbalanced credit scoring datasets. The previous work found that RNDF and GB perform very well in a credit scoring context and can cope comparatively well with significant class imbalances in these datasets. We find that only RNDF and XGB, an improvement of GB, have terrific performances in all class imbalances tested. The next step aims to investigate whether there is any better approach for each specific level of imbalanced ratio.

RQ3.2) *Better approach for each specific level of imbalanced data*

This section addresses the research question related to the most suitable approach for each level of class imbalance. To answer this question, we use the Nemenyi post hoc test to identify significant differences between the average rank (AR) of classifiers. With three datasets and 19 classifiers, the *Studentized* table indicates that the significant difference of one classifier

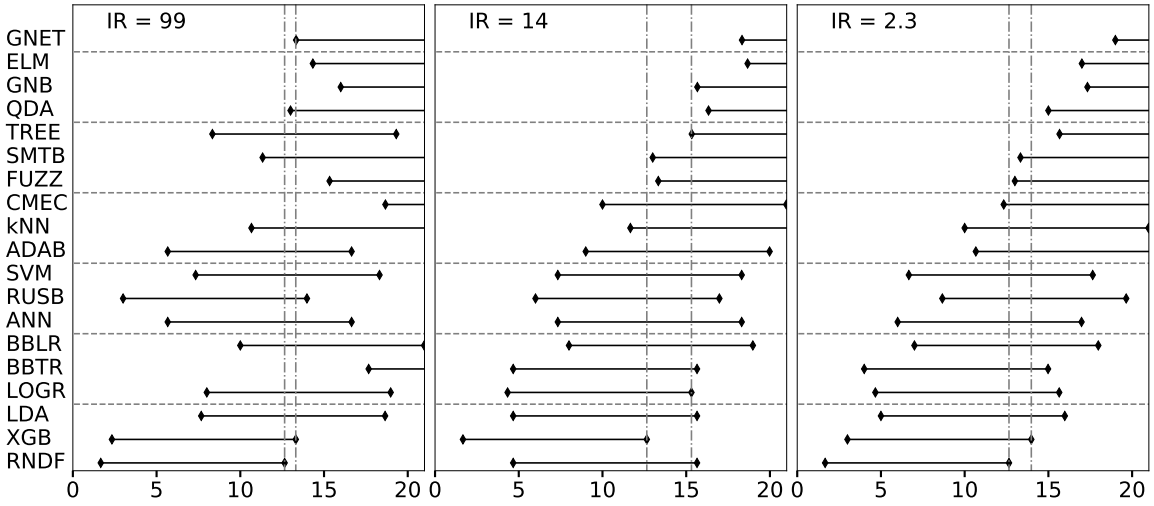
Table 9 – Friedman's Test Statistic and Classifiers rank by imbalance level

IR	99	39	19	14	12	10	9	8	6	5.7	4	2.3
Friedman Test Statistic (p = 0.01)												
>34.8	44.7	38.8	48.8	46.4	47.9	49.8	49.3	46.7	47.7	48.7	48.8	48.5
Classifiers Average Rank												
RNDF	**1.7**	**2.0**	**2.0**	4.7	**2.7**	**1.3**	**2.0**	**2.7**	3.3	**2.7**	**3.0**	**1.7**
XGB	2.3	2.0	2.3	**1.7**	4.0	4.3	3.7	5.0	**2.7**	4.3	3.3	3.0
LDA	7.7	6.7	6.0	4.7	4.7	4.0	3.0	3.7	5.0	4.3	4.7	5.0
LOGR	8.0	7.0	5.7	4.3	6.3	6.0	6.0	6.0	4.3	7.0	4.3	4.7
BBTR	17.7	14.7	3.7	4.7	4.0	3.3	4.7	4.3	6.3	1.3	4.0	4.0
BBLR	10.0	8.7	5.3	8.0	2.3	4.0	6.0	6.3	4.7	5.7	5.7	7.0
NN	5.7	6.0	6.7	7.3	7.3	7.7	9.7	8.0	5.0	7.0	5.3	6.0
RUSB	3.0	4.3	7.7	6.0	7.7	9.0	6.0	5.0	7.3	10.0	8.0	8.7
SVM	7.3	8.7	8.3	7.3	8.7	7.0	6.3	7.7	7.0	6.3	6.7	6.7
ADAB	5.7	8.0	12.3	9.0	10.7	13.7	11.0	8.7	10.7	12.0	11.0	10.7
kNN	10.7	10.7	11.7	11.7	10.3	9.7	9.3	9.0	11.0	10.0	11.3	10.0
CMEC	18.7	8.3	8.0	10.0	11.7	11.0	10.3	11.7	12.0	8.3	11.0	12.3
FUZZ	15.3	14.3	15.7	13.3	11.3	11.3	13.0	14.0	14.0	13.7	14.3	13.0
SMTB	11.3	12.3	12.0	13.0	14.3	14.3	14.7	14.7	14.7	14.3	14.3	13.3
TREE	8.3	12.0	13.7	15.3	16.3	15.0	16.0	16.3	16.0	16.3	16.3	15.7
QDA	13.0	14.0	17.0	16.3	15.0	14.7	15.3	14.7	14.0	14.3	14.7	15.0
GNB	16.0	16.7	15.7	15.7	15.7	16.7	16.0	15.3	15.0	15.3	15.7	17.3
ELM	14.3	17.0	18.7	18.7	18.7	18.3	18.3	18.0	18.0	18.0	17.3	17.0
GNET	13.3	16.7	17.7	18.3	18.3	18.7	18.7	19.0	19.0	19.0	19.0	19.0

AR be statistically better than another is 10.98 at the 5% critical difference level ($\alpha = 0.05$). Figure 2 shows this distance for three imbalanced ratios, 99, 14, and 2.3. For example, with IR=99, RNDF is statistically better than FUZZ, GNB, ELM, CMEC, BBLR, and BBTR. Another remarkable result in the average rank of IR=99 is the performance of RUSB. This imbalanced approach reached the third-best AR in this imbalanced level. However, although the AUC score of RUSB increased as the IR level decreases, the AR of this classifier lost several positions in the lower skewed dataset versions. This result shows that RUS has a lower influence in less imbalanced datasets, but it produces better results on more imbalanced ones. With IR=14, the best average classifier is XGB, which is statistically better than FUZZ, SMTB, TREE, QDA, GNB, ELM, and GNET. Still evaluating the datasets with IR=14, we can see that the second-best average rank classifier, LOGR, is statistically better than QDA, GNB, ELM, and GNET, but they are not better than FUZZ, SMTB, and TREE. We also present in Figure 2 the graphical rank of the classifiers with the version of the dataset with IR=2.3. In this version, RNDF is again the lowest average rank classifier.

About RQ3.2, analyzing all the imbalanced ratios results, and not only the three

Figure 2 – Average rank comparison at imbalanced ratio (IR) equals to 99, 14 and 2.3



diagrams presented here, but we also conclude that the RNDF and XGB are statistically better than other classifiers in all imbalanced ratios (IR). We also highlight that this work does not found any specific better approach for a particular IR. RNDF and XGB work well in all IR versions.

3.3 Conclusions

This chapter evaluated several credit scoring techniques and studied their performance over various imbalanced versions of three real-life credit data. The classification power of these techniques is measured based on the AUC. Friedman's test and Nemenyi's post hoc tests are applied to determine whether the differences between the average ranked AUC performances are statistically significant or not. Finally, significance diagrams show some of these significant results.

The results of these experiments show that random forest (RNDF), extreme gradient boosting (XGB), and RUSBoost (RUSB) performed well in very skewed datasets versions. However, only RNDF and XGB keep the first positions in Friedman score rank for all imbalanced ratios (IR). Simple base classifiers like LDA and LOGR also have a good performance, but mainly on the not skewed dataset versions. Finally, we observed that there is no specific better approach for each imbalance ratio level.

4 REDUCED MINORITY KNN

The second contribution of this thesis is Reduced Minority kNN, a novel kNN algorithm that redefines the local region for the dynamic selection classification of imbalanced credit scoring datasets. To redefine the local region, we develop a new kNN that uses the label of the neighbors and the imbalance ratio of the dataset to choose the list of neighbors of a query sample.

In a dynamic selection technique, the dynamic selection dataset (DSEL) is used to compute the classifier's competency level in each part of the feature space. These parts are called the *local regions*. The neighbors of a query sample define a local region, and kNN finds them. The dynamic selection techniques use these samples to evaluate the competence of each base classifier of the ensemble. Finally, the prediction procedure uses only the most competent classifiers.

This approach works fine in a balanced DSEL. However, the use of kNN in an imbalanced DSEL returns almost always the samples of the majority class (LIU; CHAWLA, 2011). This behavior is not desirable because the measure of competence of the base classifiers considers mainly their ability to predict samples of the majority class in this scenario.

Nevertheless, instead of using sampling techniques to generate a balanced DSEL, this chapter evaluates a modification in the k-NN procedure to balance the set of neighbors used to measure the competence of the base classifiers. The main idea is to reduce the distance of the minority samples from the predicted instance, keeping the distance of majority class samples unchanged.

This study aims to evaluate the performance of a novel dynamic selection approach for imbalanced credit scoring datasets over a wide range of classification techniques. Additionally, this chapter aims to evaluate the suitability of dynamic selection techniques to the credit scoring problem. More specifically, we aim to answer the following research questions:

- **RQ4.1)** Are dynamic selection techniques appropriate for imbalanced credit scoring problems?
- **RQ4.2)** Is there an equivalence between a dynamic selection technique and a static one?
- **RQ4.3)** Does the RMkNN improve the prediction performance of kNN?
- **RQ4.4)** Does the use of the RMkNN technique - which defines a novel local competence region of dynamic selection techniques - improve the classification performance of imbalanced credit scoring datasets?

To evaluate the performance of RMkNN, we extend our previous comparison (MELO JR *et al.*, 2019c), including other combinations of pool generators and preprocessing techniques and testing them on seven datasets. We evaluate static selection classification and several combinations of dynamic selection techniques, sampling approaches, and pool generators to assess our proposal's effectiveness.

To ensure the suitability of dynamic selection techniques for credit scoring problem, we evaluate the complexity of credit scoring datasets. This investigation considers a previous result presented in Britto Jr *et al.* (2014) that dynamic selection is suitable for complex datasets.

To deal with regulation constraints imposed by the credit scoring field, we find equivalence between a dynamic and a static selection. Basel Accords (PENIKAS, 2015) require that the same credit scoring model evaluates all customers. In dynamic selection, it does not happen once the set of classifiers used to predict each sample is chosen dynamically.

The remaining chapter presents an overview of classification techniques used to evaluate RMkNN. Next, we evaluate the suitability of dynamic selection. After, we present the RMkNN algorithm and describe the experimental setup used to compare the proposed technique with the existing ones. Finally, we present the results and conclusions.

4.1 Classification techniques evaluated

For this study, two sampling approaches, four credit scoring benchmarks, and eight imbalanced ensembles have been selected based on previous credit scoring papers (BROWN; MUES, 2012; MELO JR *et al.*, 2019a).

To evaluate RMkNN, we test four dynamic selection techniques presented on Chapter 2. Two of them are dynamic classifier selection techniques, LCA and RNK, where only the most competent base classifier predicts each query sample. The other two are dynamic ensemble selection techniques, KNE, KNU, where a subset of competent base classifiers predicts each query sample.

We also combine RMkNN with eight imbalanced ensembles. We test two versions of bagging, once combined with SMOTE and other with RUS, two versions of random forest, combined with SMOTE and RUS, an imbalanced rotation forest, and three imbalanced ensembles that use Adaboost. This ensembles are on Imbalanced Ensembles section of Table 12.

To compare the effectiveness of RMkNN, we compare its results against six credit scoring benchmarks. They are logistic regression, eXtreme Gradient Boosting, Artificial Neural

Table 10 – Techniques evaluated.

Label	Type	Acronym	Method
(I)	Reduced Minority kNN	RMkNN	Modified kNN that reduce the distance of the minority class samples
(II)	Imbalance Preprocessing	SMTE	Synthetic Minority Over-sampling Technique
		RUS	Random under-sampling
(III)	Imbalanced Ensembles (Pool generator + sampling)	BBAG	Balanced Bagging (Bagging + RUS)
		BGSM	Bagging SMOTE (Bagging + SMOTE)
		BRND	Balanced Random Forest (Random Forest + RUS)
		RFSM	Random Forest SMOTE (Random Forest + SMOTE)
		BROT	Balanced Rotation Forest (Rotation Forest + RUS)
		RUSB	RUS Boost (AdaBoost + RUS)
		SMTB	SMOTE Boost (AdaBoost + SMOTE)
		EASY	Easy ensemble (Bagging of AdaBoost + RUS)
(IV)	Dynamic Selection	KNE	k-Nearest Oracles-Eliminate
		KNU	k-Nearest Oracles-Union
		LCA	Local Class Accuracy
		RNK	Modified Classifier Rank
(V)	Credit Scoring Benchmarks	LOGR	Logistic Regression
		XGB	eXtreme Gradient Boosting
		ANN	Airtificial Neural Networks
		LSVM	Linear Support Vector Machine
		SVM	Support Vector Machine
		RNDF	Random Forest

Networks, linear and non-linear support vector machine, and a static random forest ensemble.

Table 10 shows the list of evaluated combinations: **(I)** contains our proposal of modification of kNN to select balanced samples of the DSEL; **(II)** contains preprocessing techniques (SMOTE, and RUS) to balance the DSEL; **(III)** contains the imbalanced ensembles strategies; **(IV)** lists the dynamic selection techniques evaluated, including our proposed KNIU; and **(V)** lists the credit scoring benchmarks evaluated.

We combine the pool generators and preprocessing approaches as Roy *et al.* (2018). In all imbalanced ensembles that use RUS, each base classifier receives a subset of the dataset with the same number of samples in each class. In the boosting ensembles combined with SMOTE, we double the number of samples of the minority class in each boost iteration. For Bagging and Random Forest ensembles combined with SMOTE, we make the equal size of both classes for each iteration, as done in Roy *et al.* (2018).

Next, we discuss the suitability of dynamic selection classification for the credit scoring problem.

4.2 Suitability of dynamic selection for credit scoring

This section presents an evaluation of the suitability of dynamic selection classification for credit scoring problems. We decide to perform it to avoid inappropriate use of the technique and to evaluate the regulatory constraints of Basel Accords (PENIKAS, 2015).

4.2.1 Dynamic selection for Imbalanced credit scoring datasets

Before evaluating the improvements of dynamic selection techniques to credit scoring datasets, we analyze whether the dynamic selection classification is appropriate to credit scoring datasets. As pointed out by Britto Jr *et al.* (2014), the performance of dynamic selection techniques is related to the classification complexity of the datasets. Considering this, we decide to evaluate the complexity of credit scoring datasets.

To perform this study, we evaluate the twelve complexity measures presented by Ho (1995). However, some complexity measures to binary classification have bias results for imbalanced datasets. For instance, the measure of Error Rate for 1NN Classifier (Error Rate for 1 Nearest Neighbor Classifier (N3)) tends to be low in high imbalanced datasets. Finally, we choose two less influenced by the imbalanced ratio of the dataset, Maximum Fisher's Discriminant Ratio (Fisher's Discriminant Ratio (F-1)), and Ratio of average intra/inter-class NN distance (Ratio of Average Intra/Inter class Nearest Neighbor distance (N2)). Next, we briefly describe the F-1 and N2 measures.

1. *Fisher's Discriminant Ratio (F-1)*: This is a class overlapping measure computed over every single feature as denoted in Eq. 4.1. In this Equation, f_a is the Fisher's Discriminant Ratio of feature a, and μ_{a1}, μ_{a2}, σ^2_{a1}, σ^2_{a2} are the means and the variances of the two classes, respectively. For a multidimensional problem, not necessarily all features have to contribute to class discrimination. As long as there exists one discriminating feature, the problem is easy. Therefore, $F1 = max(f_a), \forall a \in \{features\}$.

$$f_a = \frac{(\mu_{a1} - \mu_{a2})^2}{\sigma^2_{a1} + \sigma^2_{a2}} \tag{4.1}$$

2. *Ratio of Average Intra/Inter class NN distance (N2)*: This is a nonparametric separability of classes measure. It compares the intraclass dispersion with the interclass separability, as denoted in Eq. 4.2. In this equation, let $n_1^{intra}(s_i)$ and $n_1^{inter}(s_i)$ denote the intra and inter-class nearest neighbors of the sample s_i, while δ represents the Euclidian distance.

$N2$ calculates the ratio between the intra and inter-class dispersions. A small $N2$ value suggests high separability, and consequently, an easier classification problem.

$$N2 = \frac{\sum_i^N \delta(n_1^{intra}(s_i), s_i)}{\sum_i^N \delta(n_1^{inter}(s_i), s_i)} \tag{4.2}$$

To answer **RQ4.1) "Are dynamic selection techniques appropriate for imbalanced credit scoring problems?"**, we investigate two classification complexity measures presented by Ho e Basu (2002).

We evaluate the classification complexity based on the conclusion of Britto Jr *et al.* (2014) that dynamic selection techniques are appropriate to complex datasets. We apply *Fisher's Discriminant Ratio (F-1)* and *Ratio of Average Intra/Inter class NN distance (N2)* as classification complexity measures to the datasets described in Table 2. The results are in the top part of Figure 3. Each axis of the graph is one of the complexity measures. Each red cross represents a credit scoring dataset.

The first conclusion we find is that, regarding N2, the datasets Iran (IR), GiveMe (GM), and LC2015Q123 (LC) have a lower ratio of Intra/Inter class NN distance. It means that these datasets have a higher separability of classes. This higher separability explains the different behavior of these datasets in the experiment performed in subsection 4.4.5. As these three datasets have a higher separability of the classes, the impact of reducing the Euclidian distance of the minority class in kNN is lower than in datasets with a more significant overlapping area once the overlapping areas are less frequent. Fortunately, for four of seven credit scoring datasets evaluated, the N2 measure is higher, suggesting that most datasets are complex.

The second finding is related to Fisher's Discriminant Ratio (F-1). Regarding this measure, all seven credit scoring datasets evaluated have $F-1 < 0.31$. This small $F-1$ means that all credit scoring datasets are complex regarding F-1.

We now compare the F-1, and N2 measures with the datasets evaluated by Britto Jr *et al.* (2014) in the bottom part of Figure 3. In this figure, the datasets evaluated by Britto Jr *et al.* (2014) are indicated by the green triangles, while red crosses indicate the credit scoring datasets used in this thesis. Only one among all datasets evaluated by Britto Jr *et al.* (2014) is more complex than the credit scoring datasets of Table 2 regarding F-1. Regarding N2, the easiest credit scoring datasets, Iran (IR) and GiveMe (GM), are among the three harder datasets evaluated by Britto Jr *et al.* (2014).

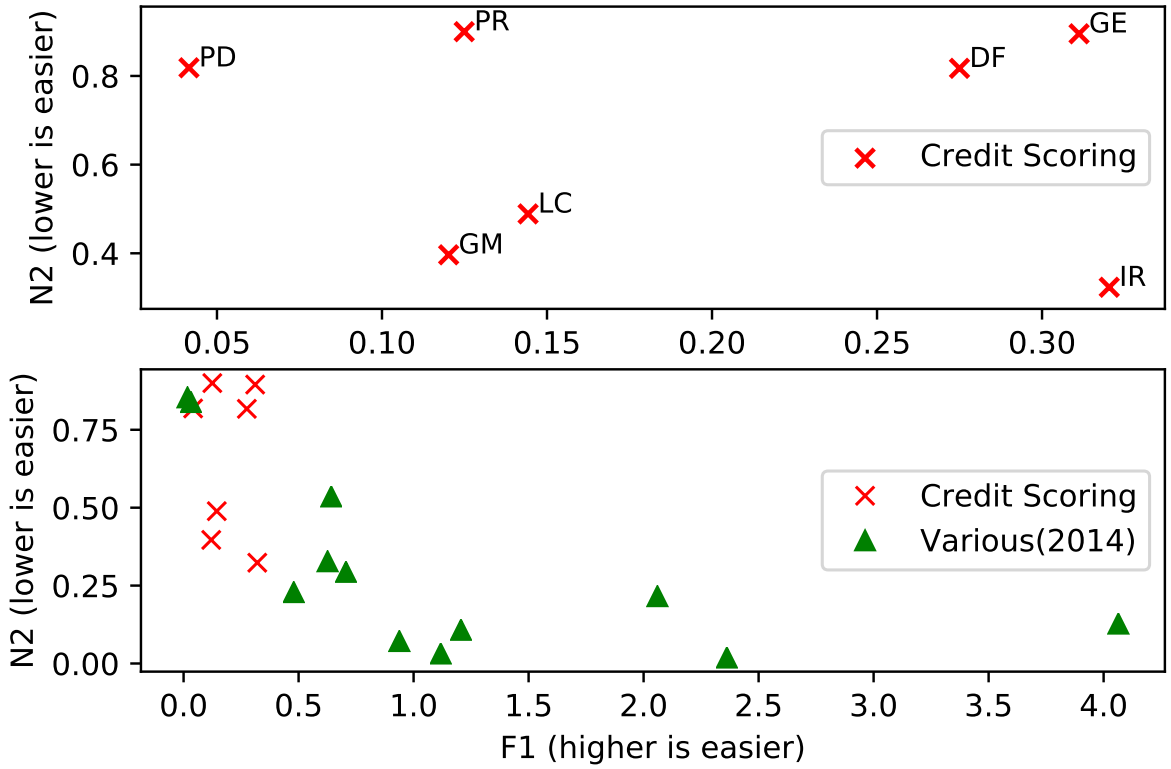


Figure 3 – Credit scoring classification complexity measures (up). Credit scoring classification complexity measures compared with other datasets (bottom).
Dataset abbreviations: DF: Default, GE: German, GM: GiveMe, IR: Iran, LC: LC2015Q123, PD: PPDai, PR: private.

Finally, based on the result of Britto Jr *et al.* (2014) stating that dynamic selection techniques are more appropriate to complex classification problems, and our experiment that demonstrates empirically that credit scoring datasets are complex, we can conclude that dynamic selection techniques are suitable for credit scoring problem. We use Fisher's Discriminant Ratio (F-1) and Ratio of Average Intra/Inter class Nearest Neighbor (NN) distance (N2) to measure and compare with datasets of other fields.

4.2.2 Equivalence of dynamic and static selection techniques

To handle the regulatory compliance of Basel accords (PENIKAS, 2015), which requires the use of the same prediction model to all costumers, we find a static classifier equivalence to a dynamic selection technique of Cruz *et al.* (2020). To answer the research question **RQ4.2) Is there any equivalence between dynamic and static selection techniques?**, we evaluate the implementation of KNU dynamic selection technique. We begin this subsection by describing the structure of KNU. After, we describe the static equivalence of KNU.

KNU implementation starts by collecting all base classifiers' predictions of all DSEL samples. If all classifiers predict the same class for one query sample, there is no selection to be done. Otherwise, this prediction information is used to compute the local accuracy of each base classifier. The local accuracy defines the weight of each base classifier in the final prediction. To illustrate this behavior with an example, consider that the accuracy in some local region of classifiers A, B, and C is 1, 0.7, and 0, respectively, the weights of the classifiers A and B in the final prediction are 1 and 0.7, while the classifier C does not influence the final prediction.

We observe that all information needed to compute the base models' accuracy in each part of the feature space is available right after the fit time. With the base models and the DSEL samples, we can define statically in all local regions of the feature space which base classifiers participate and what is their contribution weight in the final prediction.

To illustrate the concept described previously, Figure 4 shows a simple example of the local regions' definition in a bi-dimensional feature space. We use a bi-dimensional feature space here, but it can be used for any number of dimensions once the space with the same set of nearest neighbors defines a local region. Figure 4 (up) shows a bi-dimensional feature space with nine DSEL samples marked in green. Using only two neighbors to define a local region, Figure 4 (bottom) shows the local regions defined by these 9 DSEL samples in different colors. In each local region, different colors of Figure 4 (bottom), the local competence of the base classifiers is the same. It means that the influence of the base classifiers on the final prediction is the same in all parts of a local region.

As these local regions define the influence of each base classifier on the final prediction, we can define a static tree where the root node has one child for each different local region, in the example of Figure 4 (bottom), 13 child nodes. In each node of this tree, we can have a static ensemble with the weights defined by the competence of the base models in the DSEL samples that define the local region.

Figure 5 shows the static tree equivalent to the dynamic approach. In Figure 5, the SE's represent the 13 different sub-ensembles. As in each local region of Figure 4 the local competence of the base classifiers are the same, each local region define a static sub ensemble of the original ensemble.

In this subsection, we observe that a dynamic classification technique has an equivalent static approach. This equivalence is essential to the credit scoring field because dynamic selection classification uses different base classifiers to evaluate different customers, which can

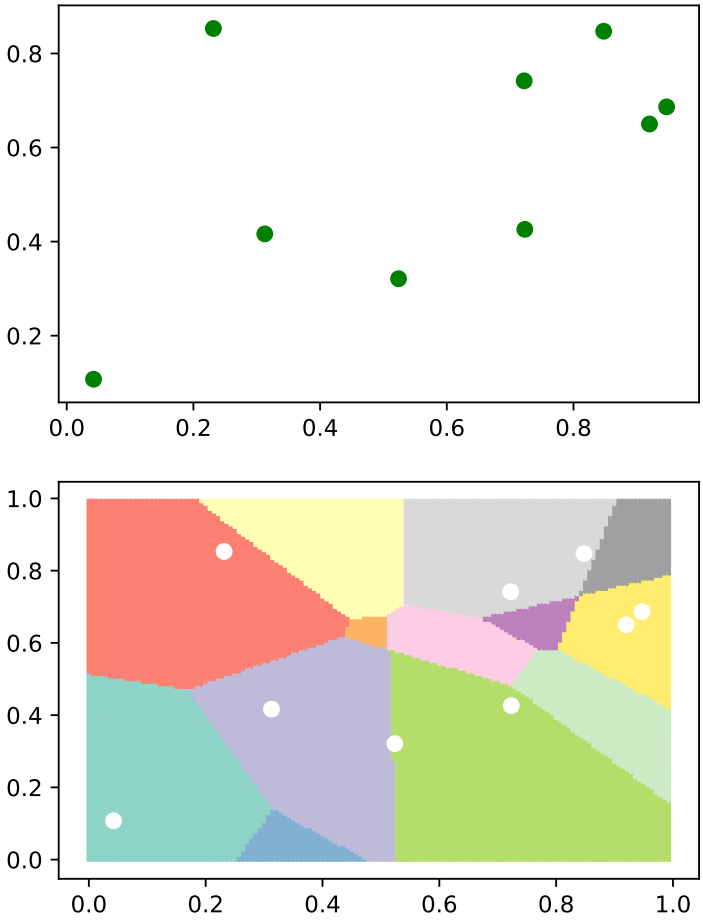


Figure 4 – A bi-dimensional feature space with nine DSEL random samples in green (up). The 13 local regions defined by these nine samples. Each local region is defined by two DSEL nearest neighbors (bottom).

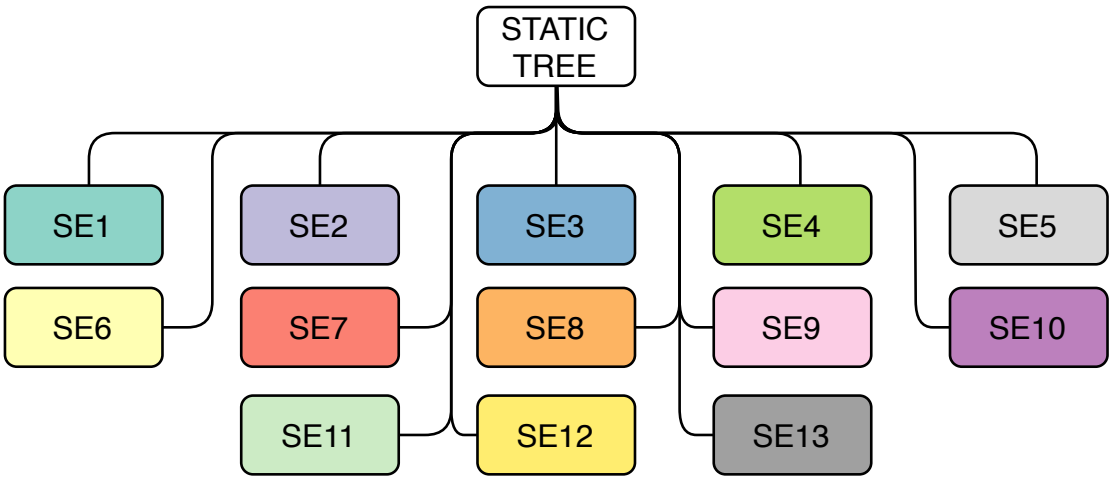


Figure 5 – Static tree equivalent to a dynamic selection classification.

characterize discrimination, and discrimination is not allowed according to Basel accords. Once we find a static equivalent model, we can state that the same static combination of base classifiers evaluates all customers. This find can be a starting point to allow dynamic selection classification in the credit scoring field.

4.3 The Reduced Minority k-NN algorithm

To evaluate the dynamic selection approaches with imbalanced datasets using DSELs without sampling techniques, we develop a modification in the k-NN algorithm shown in Algorithm 1. The intuition is to reduce the distance of the minority class samples from the predicted sample in the k-NN computation. The first step of Algorithm 1 is to separate the samples of each class, lines 2 and 3. After, we compute the imbalance ratio of the dataset, line 4, and compute the k nearest neighbors and their distances from sample query, s_q, for each class, lines 5 and 6. After, we reduce the distances of the minority class samples using the *distance_reduction_function*. The next step is to concatenate the indexes and distances of minority and majority samples, lines 8 and 9. On line 10, we compute the indexes of the k shortest distances of D and return on line 11 the distances and the indexes of the nearest neighbors.

Algoritmo 1: Reduced Minority K Nearest Neighbour

Require: dataset: X, labels: y, sample query: s_q, # neighbors: k, function: $distance\_reduction\_function$

1: $majority\_class, minority\_class \leftarrow get\_classes(y)$
2: $X_M \leftarrow X[y == majority\_class]$
3: $X_m \leftarrow X[y == minority\_class]$
4: $IR \leftarrow$ imbalance ratio of $[y]$
5: $D_m, N_m \leftarrow$ k nearest neighbors of s_q using X_m
6: $D_M, N_M \leftarrow$ k nearest neighbors of s_q using X_M
7: $D_m \leftarrow distance\_reduction\_function(D_m, IR)$
8: $N \leftarrow concatenate(N_m, N_M)$
9: $D \leftarrow concatenate(D_m, D_M)$
10: $I_k \leftarrow$ index of k smallest distances of D
11: **return** $D[I_k], N[I_k]$

We now present a simple example of the modified k-NN in Figure 6. This figure shows a DSEL where the majority samples are numbered circles, and the minority are numbered squares. kNN algorithm finds the k nearest neighbors of the predicted sample, the question mark diamond, to compute the confidence level of the base classifiers in the diamond's local region. Figure 6(A) shows that the normal k-NN, using $n_{neighbors} = 7$, selects only one sample of the minority class, one square, and six samples of the majority class, circles. Figure 6(B) shows the use of Reduced Minority k-NN (RMkNN) for the same scenario. Unlike normal k-NN, before selecting the nearest neighbors, the algorithm reduces the distance between the query

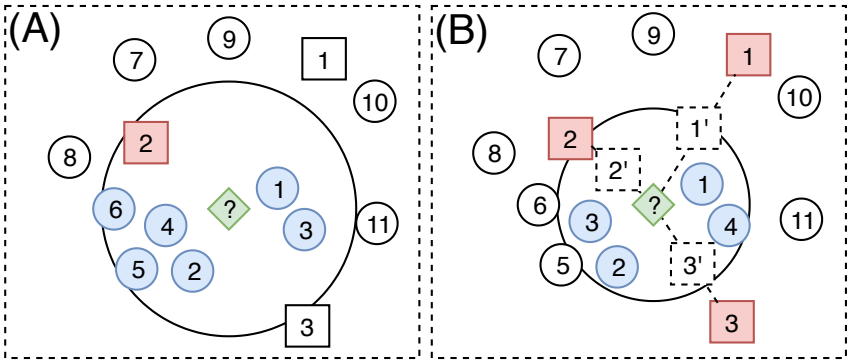


Figure 6 – The original k-NN (left) and the Modified k-NN (right).

sample and the minority class neighbors. Figure 6(B) shows the reduced distance of the minority class samples as the dotted squares. Because of this reduced distance, samples 1 and 3 become nearest neighbors of the query sample. The presented approach introduces a balanced local region definition, in terms of the number of samples, in an imbalanced dataset.

The use of RMkNN permits a fair evaluation of base classifiers without using sampling approaches to balance the DSEL. This avoidance of sampling approaches reduces the noise produced by over-sampling approaches. Also, it enables the use of the entire DSEL, impossible when undersampling approaches balance the DSEL. We believe that this brings a more efficient use of the DSEL to identify the most competent classifiers in the selection step.

However, the intensity of the reduction should be applied carefully. With a significant reduction, only minority class samples will define the local competence of the base classifiers. On the other hand, a slight reduction does not balance the samples used in the local regions. We define the reduction function for our problem based on the datasets evaluated in Section 4.4. Next, we explain the intuition behind RMkNN.

4.3.1 Why does RMkNN work?

This subsection explains why RMkNN should improve the performance of dynamic selection techniques in imbalanced data sets. We analyze the role of kNN on a dynamic selection technique and the benefits of RMkNN for an imbalanced dataset.

In a dynamic selection technique, kNN is used to define the local region of a query sample. This local region is defined by selecting the nearest neighbors, usually 7, of a query sample. Once defined, a dynamic selection procedure computes the competence of the base classifiers in this local region and performs some selection based on the base classifiers' local

competence.

In an imbalanced dataset, it is common to find local regions composed only by majority class samples. This phenomenon is good when the local region contains only majority class samples, but this is not desirable in overlap regions, regions that can contain samples of both classes. The reduced number of minority class samples available can lead to kNN only find majority class samples in the neighborhood of a query sample. An overlapping local region composed only by samples of the majority class may reduce the influence of base classifiers that correctly recognize minority class samples in that local region.

The intuition behind Reduced Minority k-NN (RMkNN) increases the probability of selecting samples of both classes in local regions. RMkNN does it by reducing the distance between the DSEL minority class samples and the query sample. This distance reduction should be enough to include minority samples in overlapping local regions, but it should not include minority samples in majority non-overlapping regions. The intuition of correlation between the imbalanced ratio and the distance reduction is that there are much more samples of the majority class in a high imbalanced dataset than the distance reduction between the query sample and the minority class should be higher. That is the reason why RMkNN uses the imbalanced ratio to define distance reduction.

Next, we describe the other possible kNN modifications evaluated to enhance the prediction performance of dynamic selection techniques.

4.3.2 Other possible kNN approaches

Beyond the RMkNN, we also evaluate two other possible k-NN modifications to handle the DSEL imbalance problem: Weighted k-NN, and using a fixed amount of samples of the same class in the feature space region definition. We comment in the following paragraphs on the reasons why we do not use them.

Weighted k-NN consists of add weights to samples in k-NN computation. For instance, in the example of Figure 6(A), we can define that the weight for each square is 0.9, and the weight for each circle is 0.1. However, for all regions with only majority class samples, we can not evaluate the ability of base learners to classify the minority class. That is why we do not use weighted k-NN to define the competence region in the dynamic selection techniques evaluated.

Another possible approach evaluated in this work is to select a fixed amount of

samples of each class in the k-NN procedure. For instance, in a dynamic selection technique that uses seven samples of DSEL to define the region of a query sample in the feature space, this approach consists of selecting four nearest neighbors of the majority class and three nearest neighbors of the minority one. However, this approach is not desirable in a region of the feature space that contains only majority-class samples. In this kind of local region, the dynamic selection procedure aims to identify the classifiers that correctly recognize the majority class. That is why we decided to evaluate an approach based on reducing the distance of the minority class.

Unlike the approaches presented in this section, the novel RMkNN can define a balanced local region without considering far samples. Additionally, with this approach, the regions containing only one class sample will be evaluated only by this class. The following section describes the methodology used in our experiments.

4.4 Experimental setup

This chapter evaluates a novel approach to define the local region used to compute the competence of base classifiers in imbalanced datasets. We now present the experimental setup used to evaluate our proposal.

4.4.1 Data preprocessing

We perform the following data preprocessing steps. First, we use one-hot encoding to transform each categorical feature with N values in N binary features. We also fill the missing values with the mean/mode for numeric/nominal features. These are the base procedures to train any machine learning model.

Additionally, we apply z-score standardization for numeric features. For instance, considering that a feature of the dataset contains the values $[40, 18, 18, 18]$, after removing the mean and scale, we get the values $[1.732, -0.577, -0.577, -0.577]$. This z-score standardization is vital because our solution uses the kNN algorithm, and different features lie within different ranges. Without feature standardization, large-scale features perform a more significant influence than small-scale ones. Next, we evaluate the approaches we use to measure the gains of our proposal, Reduced Minority k-NN (RMkNN).

4.4.2 Hyper-parameter optimization and experiment framework

We use a grid search to find the best hyper-parameters of each ensemble using F-measure to choose the best model. We test three pool sizes for all ensembles: $[60, 100, 200]$. For Bagging and Random Forest-based ensembles, we test two values for the maximum number of samples: $[0.8, 1]$. For Adaboost based ensembles, SMOTEBoost, RUSBoost, and Easy Ensemble, we test two values for learning rate: $[0.1, 1]$. For BRND, we test three values for the maximum number of features: $[\sqrt{\#features/2}, \sqrt{\#features}, \sqrt{2 \times \#features}]$. From Balanced Rotation Forest (BRTF), we test two possibilities for the size of the feature group. $[3, 9]$. These are the most common values adopted on the credit scoring papers of Table 1.

The SMOTE preprocessing technique also has parameters. We use the number of nearest neighbors equal to 5. Finally, we use seven nearest neighbors to define the region of competence for all the dynamic selection methods. We get these hyper-parameters from Roy *et al.* (2018).

We also test different hyper-parameters for the credit scoring benchmark approaches. For Logistic Regression (LOGR), we test five values for the regularization parameter C: [0.01, 0.03, 0.1, 0.3, 1]. We test two different class weights: [balanced, None], two solvers: [liblinear, saga], and two levels of tolerance for stopping criteria: [0.0001, 0.001]. For linear support vector machine (LSVM), we test five values for the regularization parameter C: [0.01, 0.03, 0.1, 0.3, 1]. We test two different class weights: [balanced, None], to levels of tolerance for stopping criteria: [0.0001, 0.001], and two maximum number of iterations: [1000, 2000]. For non-linear support vector machine (SVM), we test four values for the regularization parameter C: $[0.01, 0.1, 0.5, 1]$. We test two different class weights: [balanced, None], to levels of tolerance for stopping criteria: [0.0001, 0.001], two maximum number of iterations: $[1000, 2000]$, and two different kernels: $[rbf, poly]$. For eXtreme Gradient Boosting (XGB), as for the other ensembles, we test three ensemble sizes: [60, 100, 200]. We also test two values for control the balance of positive and negative class weights: $[1, <$imbalance ratio of the dataset$>]$. We also test to different learning rates: $[0.01, 0.2]$, two max tree depth: $[3, 6]$, three minimal child weight: $[1, 3, 5]$, two values for gamma: $[0.1, 0.3]$, and two values for L1 and L2 regularization weights: $[1e-5, 1e-2]$. We also test the Random Forest ensemble without any resampling step to balance the data (RNDF). For RNDF, we test all the hyperparameters combinations of imbalanced Random Forest ensembles described above, and we also test five different values for maximum tree depth: $[5, 8, 15, 25, 30]$, five values for the minimal samples split: $[2, 5, 10, 15, 100]$, four values for the minimal samples

leaf:$[1, 2, 5, 10]$, and two values for class weight:$[None, balanced]$. Finally, for artificial neural networks, we test two hidden layer sizes:[20, 40].

Figure 7 shows the experimental framework of this work. We perform 5-fold cross-validation to get each method's mean and standard deviation to evaluate each classification approach. For each training fold of the 5-fold, we perform 3-fold grid search cross-validation to find the best hyper-parameters of each static classifier (steps *III* and *V* of boxes C and D of Figure 7). For boxes C, and D, we use the best static ensemble to predict the test part of the 5-fold cross-validation. For boxes A and B of Figure 7, we use the 3-fold training data as DSEL, box A, or to generate the DSEL using a preprocessing approach, box B. Then, we use the DSELs, and the dynamic selection approaches on the imbalanced ensembles to find the best dynamic selection model, boxes A and B of Figure 7.

4.4.3 Dynamic selection setup

As we mentioned above, the DSEL and the training dataset must be different to avoid overfitting. However, when a dataset has few samples of one class, splitting it increases the training phase's complexity. The learning algorithm has to identify the patterns with even fewer samples of one class. Therefore, as adopted by Roy *et al.* (2018), we do not split the data used to train each approach in training and DSEL. Roy et al. used the diversity of the new samples generated by the over-sampling preprocessing techniques to ensure the difference between the training data and the DSEL. However, instead of applying over-sampling techniques, we propose a new approach to define the local region. This approach provides a balanced local region to evaluate the competence of base models.

4.4.4 Evaluation measures

A correct selection of evaluation measures is critical to avoid biased results. For instance, the percentage of correctly classified measure is widely used in classification but is not appropriate to an imbalanced dataset since a naive classifier always predicting the majority class achieves a high score.

To evaluate RMkNN, we evaluate six metrics defined in Subsection 2.3.3: Area under the ROC curve (AUC), H-measure, balanced accuracy (BAcc), G-mean, F-measure, and True Positive Rate (TPR).

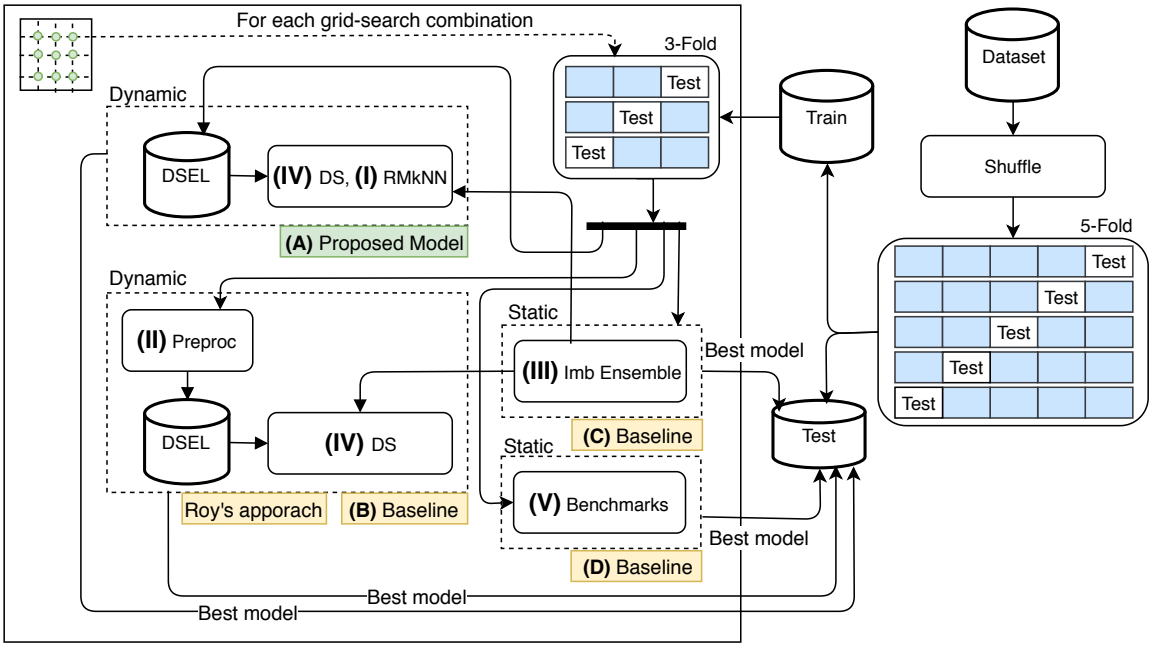


Figure 7 – The proposed approach and the baselines (adapted from Roy *et al.* (2018)).

4.4.5 The reduction function deduction

This section defines the distance function for RMkNN based on the credit scoring datasets. Considering that minority class samples are harder to find in high imbalanced datasets, our intuition is that the distance reduction in the most imbalanced datasets should be more significant than in the less imbalanced ones. This intuition makes us include the dataset's Imbalanced Ratio (IR) in the reduction function. Analyzing the imbalanced ratio (IR) of real credit scoring datasets evaluated in the papers of Table 1, we see that the IRs vary from 1 to 80. For this reason, we analyze the reduction in this range, and we use the datasets we selected to perform our experiments.

To find the distance reduction function boundaries, we use Reduced Minority k-NN to compute the percentage of minority samples selected when we get the seven nearest neighbors of each sample of a dataset with different functions. This experiment aims to find a reduction function that produces a balanced percentage of minority and majority class samples selected. The Equation 4.3 shows the formula used to compute this percentage: S is a dataset, e is a sample of S, e_{mn} means the set of minority samples among the nearest neighbors of e using Reduced Minority k-NN, and k is the number of neighbors, in this experiment, as in previous papers (ROY *et al.*, 2018; CRUZ *et al.*, 2015), seven.

$$Percentage\ of\ minority\ samples\ selected = \frac{\sum_{e \in S} |e_{mn}|}{k \times |S|} \tag{4.3}$$

We perform this experiment with three functions: $f(D_m) = D_m$, $f(D_m) = 2D_m/3$, and $f(D_m) = D_m/2$, where D_m is the distance of some minority sample from the query sample. Figure 8 shows the percentage of minority samples for each dataset we test and for each boundary function. Since the ideal percentage is 50%, half of the minority and majority samples, we consider the desired percentage of the range between 30% and 70%. Indeed, this percentage guarantees minimal samples of the minority and the majority class in the local region definition. We observe empirically that, on average, the reduction function should reduce the distance of the minority class samples by a factor between 1 and $2/3$.

With these parameters in mind, we propose the function in the Eq. 4.4. First, we evaluate the behavior of a linear function. However, the distance reduction of the linear function is too low, mainly in less imbalanced datasets. Then, we decide to use a natural logarithmic function. Finally, we use factor 10 to adjust the result of the reduction function to generate a result near the range of 1 and $2/3$. This range varies from 1 (no distance reduction) to $2/3$, the reduction rate we found empirically, which makes the modified kNN function returns mainly minority class samples. Figure 8 also shows the percentage of minority samples selected with this proposed function. As we can see, the proposed function produces a percentage of minority samples between 30% and 70% for four of seven datasets. Only the datasets GiveMe, Iran, and LC2015 presented a percentage of minority neighbors selected below our established threshold. However, all of them presented about 10% of minority class samples.

$$f(D_m, IR) = \frac{D_m}{(1 + \frac{log(IR)}{10})} \tag{4.4}$$

4.4.6 Statistical significance tests

As recommended by Demšar (2006) and followed by other credit scoring papers (XIA *et al.*, 2018; LESSMANN *et al.*, 2015), we employed nonparametric tests instead of parametric ones because the assumptions of parametric tests tend to be violated when comparing classification models. We employ the Friedman test (FRIEDMAN, 1940), which is a rank-based nonparametric test, to compare different models. Eq. 4.5 formalizes the statistic of the Friedman test.

$$X_F^2 = \frac{12D}{K(K+1)} \left[\sum_{j=1}^{K} AR_j^2 - \frac{K(K+1)^2}{4} \right], where\ AR_j = \frac{1}{D} \sum_{i=1}^{D} r_i^j \cdot \tag{4.5}$$

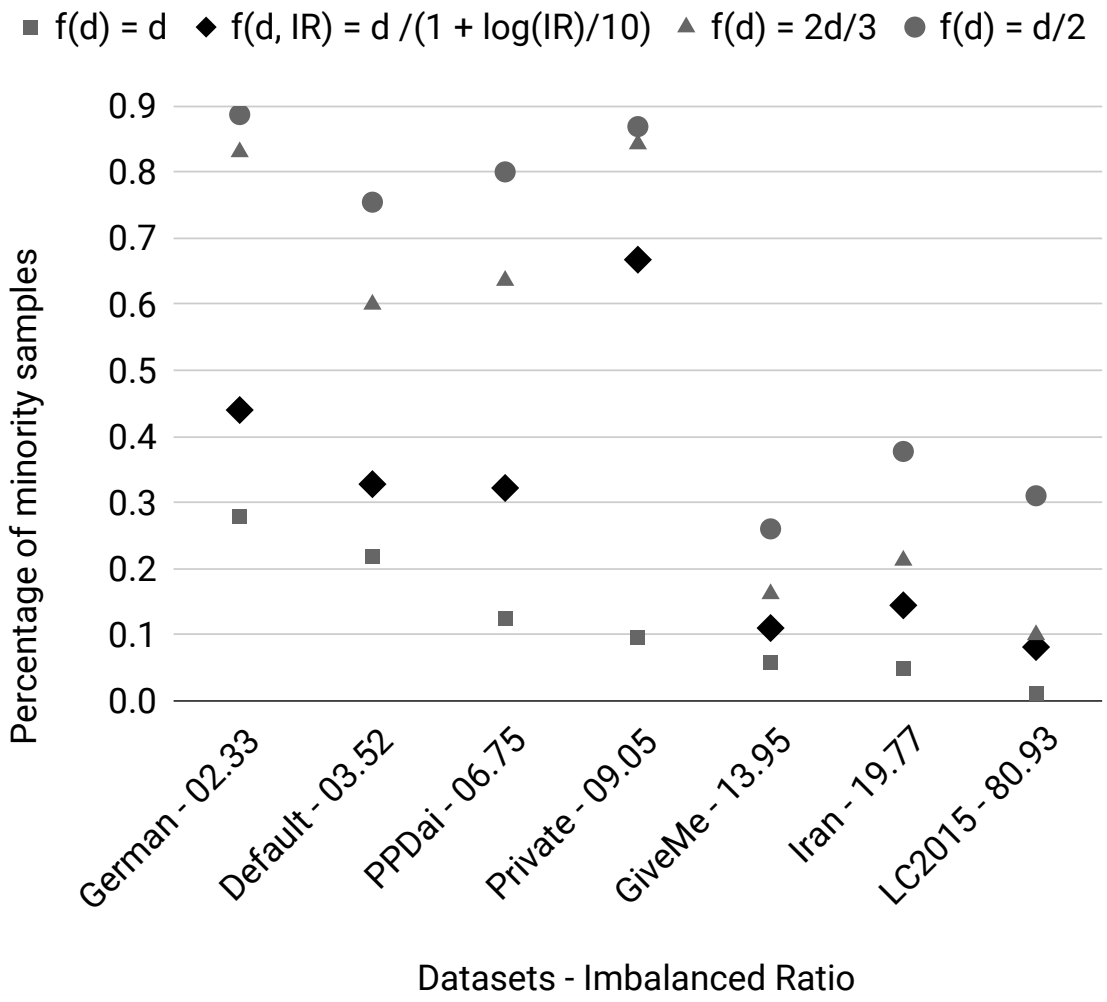


Figure 8 – The percentage of minority samples selected when different reduction functions are used in seven datasets.

In Eq (4.5), D denotes the number of datasets used in the study, K is the total number of classifiers and r_i^j is the rank of classifier j on dataset i. X_F^2 is distributed according to the Chi-square $(\tilde{\chi}^2)$ distribution with $K-1$ degrees of freedom. If the value of X_F^2 is large enough, then the null hypothesis that there is no difference between the techniques can be rejected. The Friedman statistic is well suited for this data analysis as it is less susceptible to outliers.

The post hoc Nemenyi test (NEMENYI, 1962) is applied to report any significant differences between individual classifiers. The Nemenyi post hoc test states that the performances of two or more classifiers are significantly different if their average ranks differ by at least the critical difference (CD), given by

$$CD = q_{\alpha,\infty,K}\sqrt{\frac{K(K+1)}{12D}}. \tag{4.6}$$

In this formula, the value $q_{\alpha,\infty,K}$ is based on the Studentized range statistic (NEMENYI, 1962). Finally, the results from Friedman's statistic and the Nemenyi post hoc tests are displayed using a modified version of significance diagrams (DEMŠAR, 2006; LESSMANN *et al.*, 2008). These diagrams display the ranked performances of the classification techniques and the critical difference to clearly show any techniques that are significantly different from the best-performing classifiers. Next, we discuss the results achieved in these tests.

4.5 Experimental results

We now present the results by answering each research question:

1. we analyze if the dynamic selection techniques are appropriate to credit scoring datasets.
2. We analyze the differences between performance measures.
3. we compare the proposed approach with dynamic ensemble approaches that use DSEL generated by preprocessing techniques and static ensembles.

As in previous works (LESSMANN *et al.*, 2015; ABELLÁN; CASTELLANO, 2017), we use the average rank of the selected performance measures. For the F-measure, we adopted two values for β: $[1,5]$. $\beta = 1$ means to give the same weight for precision and recall in the Equation 2.2. The other F-measure is five times more important to positive class misclassification than to negative class error. Henceforth, we refer to F-measures as F1, and F5, when β is $[1,5]$. Next, subsections answer the research questions.

4.5.1 RMkNN and kNN comparison

To assess **RQ4.3) "Does the RMkNN improve the prediction performance of kNN?"**, we use these techniques as classifiers and perform the static experiment flow of Figure 7 to compare them. Table 11 shows the results of two classifiers over the seven evaluated datasets. To simplify the results evaluation task, we sort the datasets by the imbalance level. About the performance measures, we start with the threshold-free measures, AUC, and H-measure. After, we include balanced accuracy and geometric mean (G-mean). We also include the F-measures measures at an increasing level of True Positive Rate (TPR) influence, F1, F5, and F35. Finally, we include TPR alone. For all threshold-dependent measures, we consider 0.5 as the threshold.

First, we observe that RMkNN outperforms kNN regarding G-mean, F1-score, F5-score, F35-score, and TPR. Evaluating G-mean, we notice that kNN outperforms RMkNN only

Table 11 – kNN and RMkNN comparison (each column contains the average and standard deviation of 5-fold execution)

Dataset	Classifier	Performance measures						
		AUC	H	BAcc	G-mean	F1	F5	TPR
German	kNN	0.74 (0.05)	0.09 (0.05)	0.6 (0.04)	0.5 (0.06)	0.38 (0.08)	0.29 (0.07)	0.28 (0.06)
	RMkNN	**0.75 (0.02)**	**0.16 (0.04)**	**0.69 (0.03)**	**0.69 (0.03)**	**0.57 (0.03)**	**0.71 (0.04)**	**0.72 (0.04)**
Default	kNN	**0.73 (0.02)**	**0.16 (0.02)**	0.64 (0.01)	0.56 (0.02)	0.43 (0.02)	0.34 (0.02)	0.34 (0.02)
	RMkNN	0.72 (0.02)	0.16 (0.03)	**0.67 (0.02)**	**0.65 (0.02)**	**0.48 (0.02)**	**0.52 (0.03)**	**0.52 (0.03)**
PPDai	kNN	0.57 (0.02)	0.01 (0.0)	0.51 (0.0)	0.14 (0.04)	0.04 (0.02)	0.02 (0.01)	0.02 (0.01)
	RMkNN	**0.57 (0.01)**	**0.01 (0.0)**	**0.55 (0.01)**	**0.52 (0.02)**	**0.23 (0.01)**	**0.35 (0.06)**	**0.37 (0.07)**
Private	kNN	**0.57 (0.05)**	**0.01 (0.01)**	**0.51 (0.01)**	0.14 (0.04)	0.04 (0.02)	0.02 (0.01)	0.02 (0.01)
	RMkNN	0.51 (0.05)	0.0 (0.0)	0.5 (0.01)	**0.22 (0.04)**	**0.18 (0.0)**	**0.71 (0.02)**	**0.94 (0.03)**
GiveMe	kNN	0.72 (0.0)	0.07 (0.01)	0.55 (0.0)	0.32 (0.01)	0.17 (0.01)	0.11 (0.01)	0.1 (0.01)
	RMkNN	**0.77 (0.01)**	**0.22 (0.01)**	**0.65 (0.0)**	**0.58 (0.01)**	**0.37 (0.01)**	**0.35 (0.01)**	**0.35 (0.01)**
Iran	kNN	**0.77 (0.04)**	**0.1 (0.14)**	0.5 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
	RMkNN	0.75 (0.06)	0.06 (0.07)	**0.56 (0.08)**	**0.38 (0.24)**	**0.13 (0.1)**	**0.2 (0.16)**	**0.21 (0.17)**
LC2015	kNN	0.6 (0.01)	0.0 (0.0)	0.51 (0.0)	0.17 (0.02)	0.05 (0.01)	0.03 (0.01)	0.03 (0.01)
	RMkNN	**0.61 (0.01)**	**0.02 (0.01)**	**0.57 (0.01)**	**0.55 (0.01)**	**0.28 (0.01)**	**0.42 (0.02)**	**0.44 (0.02)**

(a) **BAcc** stands for balanced accuracy.

on the private dataset. However, the performance difference between RMkNN and kNN is only $0.01 = (0.51 - 0.5)$. Regarding the threshold-free measures, AUC and H-measure, we observe that RMkNN outperforms kNN in 4 of 7 datasets. Additionally, we observe that in the 3 cases where kNN achieves superior performance, RMkNN achieves similar results.

We can conclude the superiority of RMkNN in imbalanced credit scoring problems considering the following arguments. First, we remember that AUC and H-measure give the same weight for the misclassification error of both classes, and F5, F35, and TPR give a higher weight to the positive class misclassification. When we observe split results in AUC and H-measure and RMkNN superiority in F5, F35, and TRP, we notice that RMkNN is more appropriate than kNN to handle classification problems when the positive class misclassification is higher.

4.5.2 Reduced Minority k-NN on dynamic selection techniques

To answer **RQ4.4) Does the use of the RMkNN technique - that defines a novel local competence region of dynamic selection techniques - improve the classification performance of imbalanced credit scoring datasets?**, we perform three experiments. First, we compute the overall average ranking of 110 classification approaches and compare the best estimator of the previous test with the credit scoring benchmarks. Finally, we simulate a real scenario of a credit scoring problem. The following subsections describe each experiment.

Table 12 – Average ranking of all 110 techniques

Appr.	Selection	Performance Measures							Avg
		AUC	H	BAcc	G-mean	F1	F5	TPR	
BRND	KNU+RMkNN	12.8 (12.1)	15.4 (15.1)	**10.5 (14.1)**	11.6 (15.5)	18.7 (17.0)	15.6 (16.9)	19.2 (16.6)	15.3
BROT	KNU+RMkNN	**12.7 (8.5)**	14.8 (18.4)	11.9 (13.9)	**11.4 (13.3)**	**16.8 (14.1)**	17.1 (15.1)	20.7 (12.9)	15.7
BROT	KNU+SMTE	14.7 (8.4)	19.0 (14.9)	13.9 (10.3)	13.6 (11.4)	21.5 (12.7)	16.1 (12.7)	18.7 (12.6)	17.0
BRND	KNU+SMTE	14.4 (7.9)	20.5 (14.6)	13.2 (12.8)	13.7 (14.9)	23.8 (16.9)	15.8 (15.5)	18.5 (15.7)	17.2
BRND	STATIC	15.2 (16.6)	24.6 (19.8)	13.8 (15.4)	14.5 (16.4)	29.6 (25.1)	**14.6 (15.2)**	**15.0 (16.0)**	17.8
BROT	STATIC	14.4 (12.6)	23.6 (16.7)	14.8 (12.3)	14.7 (13.8)	28.9 (21.1)	15.7 (12.8)	16.1 (13.6)	18.1
BROT	KNU+RUS	15.8 (11.8)	23.0 (16.8)	14.3 (11.6)	14.1 (12.6)	27.8 (19.9)	16.1 (13.4)	17.7 (13.4)	18.3
BRND	KNU+RUS	15.5 (14.6)	24.9 (19.1)	15.4 (16.4)	15.5 (17.0)	30.0 (24.8)	16.2 (15.7)	17.4 (16.1)	19.0
EASY	KNU+RMkNN	29.6 (21.6)	21.5 (23.0)	22.2 (29.2)	22.9 (30.9)	31.1 (26.1)	21.3 (22.4)	22.2 (21.8)	24.1
EASY	KNU+SMTE	30.3 (21.1)	23.2 (23.0)	23.8 (28.8)	24.5 (31.5)	33.5 (26.7)	22.4 (22.9)	23.5 (22.2)	25.6
EASY	STATIC	24.4 (21.4)	26.7 (24.4)	26.0 (29.3)	26.0 (31.0)	37.4 (29.6)	22.6 (22.6)	21.6 (23.3)	25.9
EASY	KNU+RUS	32.1 (22.0)	26.1 (25.3)	25.1 (29.0)	25.5 (31.4)	36.7 (30.0)	22.3 (22.6)	22.0 (22.9)	26.5
EASY	KNE+RMkNN	28.2 (20.2)	25.3 (21.1)	31.4 (26.9)	32.6 (28.7)	28.5 (24.5)	32.7 (23.3)	35.1 (22.5)	31.0
BBAG	KNU+RMkNN	26.4 (25.4)	**14.7 (21.0)**	26.6 (26.4)	28.8 (21.9)	20.2 (27.9)	41.9 (20.5)	48.1 (17.4)	31.7
XGB	STATIC	18.5 (20.9)	25.4 (26.4)	26.1 (29.4)	28.6 (32.5)	23.9 (25.4)	41.6 (30.2)	45.2 (29.2)	31.8

4.5.2.1 Overall average ranking

In this experiment, we compare the combinations of pool generators, preprocessing approaches, and dynamic selection techniques of Table 10 with the static application of the imbalanced ensemble and with credit scoring benchmarks. We evaluate the average rank of all 110 combinations (8 imbalanced ensembles $\times$ 4 selection approaches $\times$ 3 strategies to handle DSEL + eight static imbalanced ensembles + 6 credit scoring benchmarks) to start the investigation of the best approaches to imbalanced credit scoring datasets.

To get a first observation of the best results among the 110 approaches evaluated, we compute the average ranking of the eight performance measures evaluated, AUC, H-measure, balanced accuracy, geometric mean, F1-score, F5-score, F35-score, and recall (TPR). After, we compute the average of these averages to find a unique global rank. Table 12 shows these ranks and the overall average of the average ranks. In this table, the gray cells indicate the lowest average rank of each performance measure. As we can see, the balanced versions of Random Forest (BRND) and Rotation Forest (BROT) achieve the best global average rankings.

Table 12 shows that the three imbalanced ensembles achieve the lowest average ranks of all performance measures evaluated. The 14 first places in the ranking are composed only by BRND, Balanced Rotation Forest (BROT), Easy Ensemble (EASY), and Balanced Bagging (BBAG). Extreme Gradient Boosting achieves only 15th place in this rank.

Another important observation extracted from Table 12 is that the lowest average ranking of each imbalanced ensemble uses KNORA-Union and Reduced Minority kNN. We highlight in green the lines of Table 12 these combinations.

4.5.2.2 Comparison of the best average ranking with the credit scoring benchmarks

After this preliminary evaluation, we decide to evaluate the actual results of the balanced random forest combined with the dynamic selection technique KNORA-Union and Reduced Minority kNN (RMkNN), the lowest rank of Table 12, and the benchmark approaches for credit scoring: Logistic Regression (LOGR), Artificial Neural Networks (ANN), Linear Support Vector Machine (LSVM), Non-linear Support Vector Machine (SVM), Random Forest (RNDF) and eXtreme Gradient Boosting (XGB). We also include the static version of the balanced random forest to evaluate the improvement of the dynamic selection technique by each dataset.

Table 13 shows these results. For each dataset evaluated, Table 13 shows the average and the standard deviation of 5-fold execution explained in Figure 7. Here, we also highlight the best result of each dataset and each performance measure in gray.

The investigation of German, Private, Iran and LC2015123 is relatively straightforward. BRND+KNU+RMkNN achieves the best results in at least 4 of 8 performance measures. In the Private dataset, BRND+KNU+RMkNN achieves the best result in 6 of 8 performance measures.

It is a significant result, once we have one low imbalanced dataset, German, one moderate imbalanced one, Private, and two high imbalanced, Iran and LC2015Q123.

On the other hand, BRND+KNU+RMkNN, our proposed combination, does not achieve any best result in any performance measure on the Default, PPDai, and GiveMe datasets. However, if we evaluate the difference between the best approaches of these datasets carefully, we see that the differences between BRND+KNU+RMkNN and the highest scores are under 0.03. For instance, according to AUC and H-measure, measures that give the same importance to the misclassification cost of both classes, the highest differences between BRND+KNU+RMkNN and the best results are 0.023 (H-measure difference in Default dataset) and 0.021 (AUC difference in PPDai dataset). Additionally, evaluating the three different f-measures, the gap of BRND+KNU+RMkNN to the best results is under 0.002, an acceptable result.

4.5.2.3 Real credit scoring scenario

Our last experiment to measure the ability of RMkNN to improve the prediction performance of credit scoring datasets is a practical application of credit scoring. We use the

Table 13 – Balanced Random Forest combined with KNORA-U and RMkNN compared with state-of-the-art classifiers in credit scoring problem

Dataset	Classif.	Selection	Performance Measures						
			AUC	H	BAcc	G-mean	F1	F5	TPR
German	XGB	STATIC	0.788 (0.02)	0.232 (0.04)	0.720 (0.02)	0.717 (0.02)	0.607 (0.02)	0.668 (0.04)	0.673 (0.04)
	LOGR	STATIC	0.795 (0.03)	**0.264 (0.05)**	**0.738 (0.03)**	**0.736 (0.03)**	**0.628 (0.03)**	0.718 (0.08)	0.727 (0.08)
	ANN	STATIC	0.767 (0.02)	0.180 (0.04)	0.675 (0.02)	0.651 (0.03)	0.540 (0.03)	0.501 (0.05)	0.498 (0.05)
	LSVM	STATIC	0.721 (0.02)	0.232 (0.03)	0.721 (0.02)	0.717 (0.03)	0.606 (0.03)	0.685 (0.09)	0.693 (0.10)
	SVM	STATIC	0.796 (0.03)	0.188 (0.07)	0.672 (0.03)	0.637 (0.04)	0.532 (0.05)	0.465 (0.04)	0.460 (0.04)
	RNDF	STATIC	0.791 (0.03)	0.225 (0.04)	0.715 (0.02)	0.711 (0.02)	0.600 (0.03)	0.658 (0.07)	0.663 (0.07)
	BRND	STATIC	0.800 (0.03)	0.240 (0.07)	0.729 (0.03)	0.727 (0.03)	0.617 (0.04)	0.750 (0.04)	0.764 (0.04)
	BRND	KNU+RMk	**0.802 (0.03)**	0.256 (0.06)	0.737 (0.03)	0.736 (0.03)	0.627 (0.04)	**0.757 (0.04)**	**0.770 (0.04)**
Default	XGB	STATIC	0.783 (0.02)	0.229 (0.04)	0.714 (0.02)	**0.709 (0.02)**	0.540 (0.03)	0.617 (0.03)	0.624 (0.03)
	LOGR	STATIC	0.722 (0.02)	0.137 (0.03)	0.672 (0.02)	0.671 (0.02)	0.478 (0.02)	0.623 (0.03)	0.639 (0.03)
	ANN	STATIC	0.773 (0.02)	0.205 (0.04)	0.659 (0.02)	0.592 (0.03)	0.475 (0.04)	0.377 (0.03)	0.371 (0.03)
	LSVM	STATIC	0.673 (0.02)	0.140 (0.03)	0.673 (0.02)	0.671 (0.02)	0.479 (0.02)	0.619 (0.03)	0.635 (0.03)
	SVM	STATIC	0.633 (0.02)	0.001 (0.00)	0.501 (0.00)	0.022 (0.03)	0.003 (0.00)	0.001 (0.00)	0.001 (0.00)
	RNDF	STATIC	**0.784 (0.02)**	**0.239 (0.04)**	**0.714 (0.02)**	0.704 (0.02)	**0.546 (0.03)**	0.591 (0.03)	0.596 (0.03)
	BRND	STATIC	0.780 (0.02)	0.211 (0.04)	0.709 (0.02)	0.706 (0.02)	0.529 (0.02)	**0.631 (0.03)**	**0.641 (0.03)**
	BRND	KNU+RMk	0.779 (0.02)	0.216 (0.04)	0.711 (0.02)	0.707 (0.02)	0.533 (0.03)	0.627 (0.03)	0.637 (0.03)
PPDai	XGB	STATIC	**0.632 (0.05)**	**0.024 (0.02)**	**0.564 (0.04)**	**0.455 (0.26)**	**0.211 (0.12)**	0.357 (0.21)	0.380 (0.22)
	LOGR	STATIC	0.629 (0.03)	0.019 (0.04)	0.521 (0.04)	0.146 (0.20)	0.066 (0.13)	0.058 (0.12)	0.058 (0.12)
	ANN	STATIC	0.627 (0.03)	0.009 (0.00)	0.509 (0.01)	0.132 (0.08)	0.041 (0.03)	0.023 (0.02)	0.022 (0.02)
	LSVM	STATIC	0.519 (0.04)	0.018 (0.04)	0.519 (0.04)	0.115 (0.21)	0.061 (0.13)	0.055 (0.12)	0.054 (0.12)
	SVM	STATIC	0.477 (0.05)	0.000 (0.00)	0.500 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)
	RNDF	STATIC	0.631 (0.04)	0.022 (0.02)	0.560 (0.04)	0.437 (0.25)	0.204 (0.12)	0.406 (0.27)	0.449 (0.31)
	BRND	STATIC	0.615 (0.05)	0.018 (0.01)	0.554 (0.03)	0.428 (0.23)	0.200 (0.11)	**0.459 (0.30)**	**0.521 (0.35)**
	BRND	KNU+RMk	0.611 (0.04)	0.018 (0.01)	0.554 (0.03)	0.432 (0.23)	0.200 (0.11)	0.452 (0.30)	0.511 (0.35)
Private	XGB	STATIC	0.682 (0.04)	0.067 (0.05)	0.603 (0.06)	0.541 (0.13)	0.242 (0.07)	0.369 (0.17)	0.388 (0.19)
	LOGR	STATIC	0.668 (0.05)	0.060 (0.03)	0.620 (0.03)	0.618 (0.03)	0.245 (0.02)	0.550 (0.06)	0.613 (0.08)
	ANN	STATIC	0.561 (0.07)	0.019 (0.01)	0.513 (0.06)	0.414 (0.05)	0.160 (0.04)	0.212 (0.05)	0.221 (0.05)
	LSVM	STATIC	0.546 (0.04)	0.017 (0.02)	0.546 (0.04)	0.403 (0.23)	0.144 (0.09)	0.308 (0.26)	0.347 (0.30)
	SVM	STATIC	0.644 (0.05)	0.015 (0.01)	0.539 (0.03)	0.355 (0.21)	0.149 (0.09)	0.184 (0.12)	0.188 (0.13)
	RNDF	STATIC	0.719 (0.02)	0.106 (0.06)	0.618 (0.05)	0.540 (0.12)	**0.283 (0.07)**	0.343 (0.14)	0.352 (0.15)
	BRND	STATIC	**0.719 (0.03)**	0.105 (0.02)	0.662 (0.02)	0.661 (0.02)	0.279 (0.01)	0.603 (0.05)	0.668 (0.06)
	BRND	KNU+RMk	0.718 (0.03)	**0.111 (0.03)**	**0.667 (0.03)**	**0.666 (0.02)**	0.281 (0.01)	**0.617 (0.06)**	**0.685 (0.07)**
GiveMe	XGB	STATIC	**0.865 (0.00)**	0.343 (0.01)	**0.786 (0.00)**	**0.786 (0.00)**	0.338 (0.00)	0.703 (0.01)	0.772 (0.01)
	LOGR	STATIC	0.806 (0.01)	0.252 (0.01)	0.732 (0.00)	0.726 (0.00)	0.311 (0.01)	0.593 (0.01)	0.642 (0.01)
	ANN	STATIC	0.833 (0.01)	0.125 (0.01)	0.582 (0.00)	0.414 (0.01)	0.266 (0.01)	0.178 (0.01)	0.173 (0.01)
	LSVM	STATIC	0.651 (0.00)	0.089 (0.00)	0.651 (0.00)	0.651 (0.00)	0.203 (0.00)	0.545 (0.01)	0.635 (0.01)
	SVM	STATIC	0.484 (0.03)	0.000 (0.00)	0.500 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)
	RNDF	STATIC	0.862 (0.00)	**0.349 (0.02)**	0.779 (0.00)	0.777 (0.01)	**0.356 (0.02)**	0.674 (0.02)	0.729 (0.03)
	BRND	STATIC	0.862 (0.00)	0.337 (0.01)	0.785 (0.00)	0.785 (0.00)	0.331 (0.00)	**0.708 (0.00)**	**0.782 (0.00)**
	BRND	KNU+RMk	0.861 (0.00)	0.344 (0.01)	0.786 (0.00)	0.785 (0.00)	0.340 (0.00)	0.701 (0.00)	0.769 (0.01)
Iran	XGB	STATIC	0.760 (0.06)	0.146 (0.06)	0.608 (0.03)	0.489 (0.06)	**0.267 (0.06)**	0.251 (0.06)	0.251 (0.06)
	LOGR	STATIC	0.777 (0.06)	0.000 (0.00)	0.499 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)
	ANN	STATIC	0.769 (0.03)	0.010 (0.02)	0.501 (0.01)	0.032 (0.07)	0.018 (0.04)	0.010 (0.02)	0.010 (0.02)
	LSVM	STATIC	0.567 (0.16)	0.093 (0.21)	0.567 (0.16)	0.169 (0.38)	0.062 (0.14)	0.157 (0.35)	0.180 (0.40)
	SVM	STATIC	0.760 (0.09)	0.000 (0.00)	0.500 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)
	RNDF	STATIC	0.792 (0.04)	0.107 (0.04)	0.569 (0.02)	0.374 (0.06)	0.227 (0.06)	0.148 (0.05)	0.144 (0.05)
	BRND	STATIC	0.770 (0.05)	0.176 (0.08)	0.708 (0.05)	0.707 (0.05)	0.192 (0.04)	**0.585 (0.07)**	**0.709 (0.08)**
	BRND	KNU+RMk	**0.810 (0.07)**	**0.279 (0.12)**	**0.735 (0.07)**	**0.722 (0.08)**	0.266 (0.06)	0.566 (0.12)	0.627 (0.15)
LC2015	XGB	STATIC	**0.712 (0.04)**	0.081 (0.02)	0.636 (0.02)	0.622 (0.04)	**0.049 (0.00)**	0.295 (0.03)	0.515 (0.08)
	LOGR	STATIC	0.693 (0.02)	0.028 (0.04)	0.555 (0.08)	0.253 (0.35)	0.017 (0.02)	0.120 (0.16)	0.234 (0.32)
	ANN	STATIC	0.519 (0.01)	0.004 (0.00)	0.503 (0.01)	0.049 (0.08)	0.007 (0.01)	0.013 (0.03)	0.016 (0.03)
	LSVM	STATIC	0.502 (0.00)	0.000 (0.00)	0.502 (0.00)	0.057 (0.13)	0.005 (0.01)	0.015 (0.03)	0.017 (0.04)
	SVM	STATIC	0.550 (0.02)	0.000 (0.00)	0.500 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)	0.000 (0.00)
	RNDF	STATIC	0.709 (0.03)	0.030 (0.05)	0.539 (0.06)	0.202 (0.29)	0.024 (0.03)	0.090 (0.13)	0.121 (0.19)
	BRND	STATIC	0.703 (0.03)	0.083 (0.01)	0.654 (0.01)	0.653 (0.01)	0.043 (0.00)	0.317 (0.01)	**0.676 (0.03)**
	BRND	KNU+RMk	0.702 (0.03)	**0.092 (0.03)**	**0.656 (0.02)**	**0.655 (0.02)**	0.047 (0.00)	**0.319 (0.02)**	0.626 (0.04)

entire LC2015Q123 dataset to train the models using the experimental setup defined in Figure 7, and we evaluate the performance of all 110 models in the last quarter, LC2015Q4.

After collecting the performance measure of all classifier combinations, we compute the average rank of all performance measures to find, by each ensemble, the best combination. The best combinations and the credit scoring benchmarks results are in Table 14.

The first exciting outcome from Table 14 is the amount of best ensemble combinations with RMkNN. Four of the eight best ensemble combinations use RMkNN. They are BRND, Random Forest SMOTE (RFSM), Bagging SMOTE (BGSM), and Easy ensemble (EASY). This result shows the superiority of RMkNN over the other imbalanced dynamic selection strategies evaluated.

Another exciting result of this experiment is the performance of BRND+KNU+RMkNN. As in the results shown in Table 13, BRND+KNU+RMkNN does not achieve the best results on F35 and TPR. However, the performance of this combination on these measures is not far from the best ones.

With these experiments, we infer that RMkNN combined with dynamic selection approaches improves the prediction performance of imbalanced ensembles. We also note that Balanced Random Forest combined with KNORA-Union and RMkNN outperforms classical credit scoring classifiers, such as eXtreme Gradient Boosting, Support Vector Machine, Artificial Neural Networks, and Logistic Regression.

Finally, we test RMkNN in an actual credit scoring scenario, where we train the model with the available data on time to predict future loans. Again, the dynamic selection approaches combined with RMkNN outperform credit scoring benchmarks.

Table 14 – Classification results of the 8 best ensemble combinations and the 4 credit scoring benchmark approaches over the last quarter of 2015 of LC2015 dataset.

Dataset	Appr.	Selection	Performance Measures						
			AUC	H	BAcc	G-mean	F1	F5	TPR
LC2015Q4	BRND	KNU+RMkNN	0.679	**0.065**	**0.631**	**0.629**	0.037	**0.272**	0.576
	BBAG	KNU+RUS	0.67	0.039	0.584	0.534	0.038	0.215	0.348
	RFSM	RNK+RMkNN	0.505	0.01	0.519	0.25	0.036	0.06	0.064
	BGSM	RNK+RMkNN	0.518	0.012	0.518	0.234	**0.038**	0.054	0.056
	RUSB	RNK+RUS	0.572	0.012	0.558	0.558	0.026	0.218	0.576
	EASY	KNU+RMkNN	0.65	0.051	0.62	0.619	0.034	0.262	**0.592**
	BROT	STATIC	0.662	0.04	0.605	0.604	0.032	0.25	0.572
	SMTB	KNU+RUS	0.589	0.012	0.54	0.434	0.031	0.149	0.22
	LOGR	STATIC	0.649	0.037	0.599	0.595	0.033	0.243	0.528
	ANN	STATIC	0.502	0.003	0.502	0.045	0.008	0.004	0.004
	LSVM	STATIC	0.5	0.0	0.5	0.0	0.0	0.0	0.0
	XGB	STATIC	**0.683**	0.04	0.597	0.581	0.035	0.237	0.46
	RNDF	STATIC	0.664	0.021	0.546	0.275	0.019	0.112	0.188
	SVM	STATIC	0.5	0.0	0.5	0.0	0.0	0.0	0.0

4.5.3 Discussion

We now investigate the best combination strategy among all evaluated. To achieve it, we compute a new average rank of the best results of each ensemble combination and the credit scoring benchmarks. Applying the Friedman test on the average ranking of these twelve classifiers, we get a Friedman test statistic $= 90.41$, and a $p - value < 0.005$. As the Friedman test result is significant ($p < 0.005$), we can apply the post hoc Nemenyi test to the distribution.

Figure 9 shows the average ranks of these best combinations and the Critical Distance of the Nemenyi test. This figure shows that balanced random forest (BRND) and balanced Rotation Forest (BROT) combined with KNORA Union (KNU) and using RMkNN to generate the DSEL are the lowest average ranks. These approaches are statistically better than Artificial Neural Networks and Support Vector Machine, as indicated by the critical distance bar.

We also observe that RMkNN is present on four best combinations of eight ensembles. They are highlighted in green on Figure 9, and they are BRND, Balanced Rotation Forest(BROT), Easy Ensemble (EASY), and Balanced Bagging (BBAG). The next three best ranking positions are combinations that use Random Undersampling (RUS) to generate the dynamic selection dataset (DSEL). They are SMOTEBoost (SMTB), RUSBoost (RUSB), and Random Forest SMOTE (RFSM). Only the last position, Bagging SMOTE (BGSM), uses SMOTE to generate the DSEL. Figure 9 highlights these last four combinations in yellow.

Another critical finding is the performance of KNORA-Union (KNU). This dynamic ensemble selection technique is in the six best ranking combinations, BRND, BROT, EASY, BBAG, RUSB, and RFSM. The last two ranking positions use KNORA-Elimination (KNE) and Local Class Accuracy (LCA). This result demonstrates that KNU is an excellent technique to combine with imbalanced pool generators to address imbalanced datasets.

With these experiments, we observe that RMkNN improves the local region definition in a dynamic selection technique. We also observe that KNORA-Union (KNU) is an excellent dynamic selection technique to combine imbalanced ensembles. After, we observe that BRND is the best pool generator to combine with KNU.

4.5.4 Limitations of the study

This study presents RMkNN, a new kNN algorithm used by dynamic selection techniques for imbalanced credit scoring datasets. The first apparent issue in this work is the

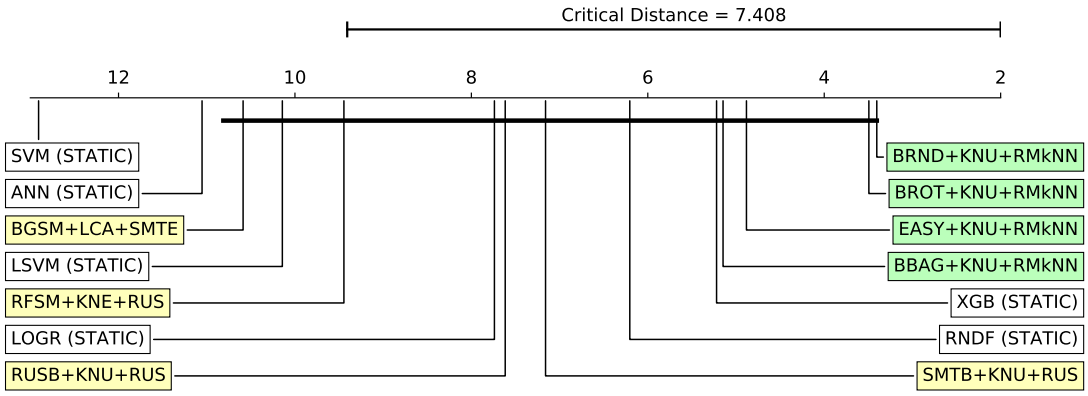


Figure 9 – The average rank of the best combinations.

running performance of RMkNN. The proposed version of kNN runs two kNN internally, one for majority class samples and another for minority class ones. It is necessary to reduce the distance between the query sample and the minority class samples of the DSEL. It is slower than the original kNN algorithm.

Another possible limitation is the reduction function proposed in Eq. 4.4. This reduction function uses only the imbalance ratio. Maybe a better result can be achieved by including other variables, such as a complexity measure of the dataset. A complexity measure can describe, for instance, the degree of separability among the samples of each class. In a less complex dataset, the distance reduction can be shorter than in a more complex one.

Another limitation is that we evaluate the proposed solution only in credit scoring datasets, a binary problem with complex datasets. The superiority of RMkNN probably does not occur in more straightforward datasets, once the overlapping areas are less frequent. Moreover, it is not trivial to extend RMkNN to a non-binary classification problem.

4.6 Conclusions

In this chapter, we present a study of the imbalanced credit scoring problem. We assess the combination of Dynamic Selection (DS) methods, data preprocessing, and pool generation ensembles to deal with the imbalanced nature of the credit scoring data sets using a novel approach to define the local regions of a dynamic selection technique.

We propose RMkNN to perform a balanced selection of DSEL samples in a dynamic selection technique. To assess our technique's performance, we compare our proposal with two preprocessing techniques, SMOTE and RUS.

Experiments conducted on seven datasets shown that combining RMkNN with DS

techniques enhances the prediction performance according to 7 performance measures. We also reduce a DS technique to a static selection approach. After, we empirically conclude that the KNORA-Union (KNU) is the best DS technique to use in these combinations. Finally, we evaluate our proposed technique in a real-life credit scoring problem to assess that RMkNN outperforms other techniques and classical credit scoring benchmark classifiers.

5 KNORA-IU: ENHANCED DYNAMIC SELECTION FOR IMBALANCED CREDIT SCORING PROBLEMS

In this chapter, we describe the third result of this thesis that is the use of a novel performance measure to compute base classifiers' local competence, called KNORA-IU (MELO JR *et al.*, 2019b). We combine this performance measure with KNORA-Union (KNU), a well-established dynamic selection technique.

KNU is a dynamic ensemble selection technique, and it selects all classifiers that correctly classified at least one sample belonging to the region of competence of the query sample. Each selected classifier has the number of votes equals the number of samples in the region of competence that predicts the correct label. The final ensemble decision aggregates all the votes obtained by all base classifiers.

KNU uses accuracy to define the local competency and to determine the contribution weight of each base classifier in the final prediction. As mentioned before, accuracy in an imbalanced dataset reports good results even for a naive learner that predicts only the majority class. This fact motivates us to consider using a different performance measure to compute the competence of each base learner in KNORA-Union.

To evaluate the performance of KNORA-Imbalanced Union, we compare it with benchmarks competitors. More specifically, we aim to answer the following research questions related to the credit scoring problem:

- **RQ5.1)** Does the use of the KNORA-Imbalanced Union - that uses FA^2 as the performance measure to compute the local competence of base classifiers - improve the classification performance of imbalanced credit scoring datasets?
- **RQ5.2)** How does KNORA-IU improve the classification performance of imbalanced credit scoring datasets?

5.1 Description of KNORA-Imbalanced Union (KNIU)

To avoid the noise produced by the over-sampling approaches, we decided to investigate the replacement of accuracy as the measure used to compute the local competence of the base learners. Our intuition is to use a measure that reflects the minority class errors properly. We choose *f-measure*, the harmonic mean of precision and recall because it is a measure that focuses attention on the positive minority class.

However, we can not use f-measure alone. This measure is not defined in a dataset

composed only of samples of one class, and the competence of the base learners is evaluated with few examples of the neighborhood of the query sample. We, therefore, need to choose another competence metric.

To overcome this issue in KNORA-IU, we try to combine f-measure with *accuracy*. We base this decision on the fact that the accuracy is enough to assess the classifiers in a neighborhood with only one class sample. However, even in this scenario, we discover empirically that the accuracy measure does not penalize the base learners appropriately with few prediction errors. We observe that the influence in the final prediction of classifiers with few errors is still strong. To attenuate this "good performance", we decided to consider the square of accuracy. Thus, our proposed performance measure to compute the competence of each base learner can be written as Eq. (5.1).

$$FA^2(yt, yp) = \begin{cases} f1\text{-}score(yt, yp), & \text{if } yt \supset \{-1, 1\} \\ (accuracy(yt, yp))^2, & \text{otherwise} \end{cases} \tag{5.1}$$

In Eq. (5.1), FA^2 is our proposed measure, and yt and yp are, respectively, the true labels and the predicted lablels. We also use the *f1-score*, which is the *f-measure* when $\beta = 1$.

The performance measure defined above fits perfectly with KNORA-Union. As mentioned in **??**, KNORA-Union considers all the base learners that have at least one sample in the DSEL predicted correctly. The KNORA-Union combined with the measure described in Eq. (5.1) increases the influence of good imbalance performance base learners in the final prediction.

To illustrate this previous statement, we now present a simple example that shows the benefits of our approach. Figure 10 shows a dynamic ensemble with three base learners, $C1$, $C2$, and $C3$, to predict two samples indicated with the diamonds 1 and 2. The circles represent the DSEL samples used to compute the base learner's competence. The red diamond and circles represent the minority class and the blue ones, the majority ones. To compute the competence of each base learner, we need to find in the DSEL the nearest neighbors of each query sample. In this example, we use seven nearest neighbors indicated by the small circles in Figure 10. For simplicity, without losing generality, we consider that all classifiers use a linear approach to predict each sample. For example, classifier $C2$ predicts as red every sample under the $C2$ line and blue every sample over the $C2$ line.

Table 15 shows the computation of the prediction of the two samples using KNORA-Union and KNORA-Imbalanced Union. For each query sample, 1 and 2, this table shows the

Table 15 – Classification example results

S[a]	Learner	Pred	KNORA-U		KNORA-IU	
			Acc[b]	weight	FA^2	weight
1	C1	1	1	1	1	1
	C2	-1	0,57	-0,57	0,40	-0,40
	C3	-1	0,57	-0,57	0,40	-0,40
	DS prediction		-0,07 (-1)		0,11 (1)	
2	C1	-1	1	-1	1	-1
	C2	1	0,57	0,57	0,33	0,33
	C3	1	0,71	0,71	0,51	0,51
	DS prediction		0,13 (1)		-0,09 (-1)	

[a]**S** means the query sample evaluated.
[b]**Acc** means the accuracy of the learner in the neighborhood of the query sample.

individual prediction of each base learner, $C1$, $C2$, and $C3$. We consider here that 1 represents the positive class prediction and -1 the negative one. For each of these classifiers, this table shows the performance measures used by KNORA-U and KNORA-IU, accuracy, and FA^2, respectively. This table also shows the weight of each base classifier in the computation of the prediction.

Analyzing the predictions of sample query 1 in Table 15, we see that only $C1$ predicts the sample correctly, and it is enough to KNORA-IU predict the sample 1 correctly. On the other hand, traditional KNORA-U cannot predict this sample correctly. This misclassification happens because KNORA-U uses accuracy to determine the weight of the contribution of each base learner. On the other hand, as KNORA-IU uses F1-score to compute the weight of the contribution of each classifier in the final prediction, the influence of incorrect base learners $C2$ and $C3$ are not enough to contaminate the fusion strategy (the computation of the final DS

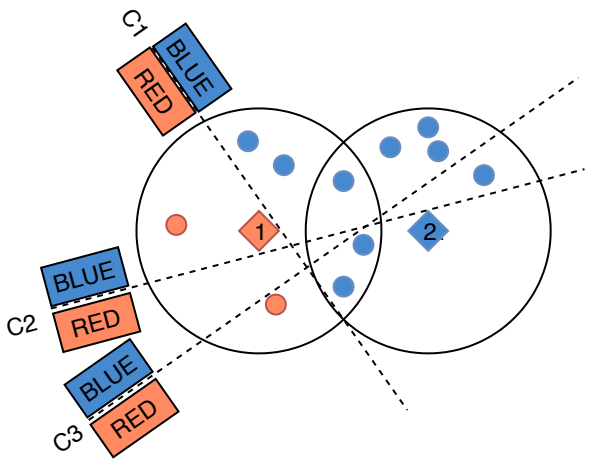


Figure 10 – Example of the proposed approach for classifying two objects with three classifiers

prediction) of the dynamic selection technique. This example illustrates that accuracy is not appropriate to compute the base learner's competence in an imbalanced dataset.

Regarding sample 2, all the neighbors available are of the same class, the majority one. Again, only the learner $C1$ can predict the example correctly. Repeatedly, the KNORA-U fusion strategy performs an incorrect prediction because of the incorrect predictions of classifiers $C2$ and $C3$. On the other hand, KNORA-IU can correctly predict sample 2 because the influence of imperfect learners $C2$ and $C3$ is reduced by the computation of the square of accuracy.

This small example illustrates the two scenarios that KNORA-IU handles differently from KNORA-U the fusion strategy. However, it is easy to see that this strategy can spoil the fusion strategy of the dynamic selection technique. To assess the usefulness of the KNORA-IU, we perform an empirical experiment comparing our approach with KNORA-U and DES-MI.

5.2 Experimental setup

In this section, we provide a complete description of our experiments. We intend to assess if KNORA-IU outperforms previously proposed dynamic selection techniques to imbalanced credit datasets. To achieve it, we compare our approach with KNORA-U combined with SMOTE (ROY *et al.*, 2018) and with DES-MI (GARCÍA *et al.*, 2018). Next, we present the datasets used, the competitors, the ensemble method, the base learners, the experimental setting, and the evaluation measures.

5.2.1 Real credit data and data preparation

As described in 2.3.1, we perform our experiments by exploiting five real-world credit scoring datasets. *Default* is provided by UCI machine learning repository[1]. *PPDai* comes from a Chinese internet finance enterprise named PaiPaiDai[2]. *Iran* comes from Sabzevari *et al.* (2007). *GiveMe*[3] comes from Kaggle competition. The last one, *LC2017Q1*, contains loan data of the first quarter in 2017 from Lending Club[4]. We use the Imbalance Ratio (IR), the cardinality of the majority class divided by the cardinality of the minority class, to sort the datasets from the less imbalanced to the most imbalanced. In the first one, *Default*, the number of samples of the majority class is 3.52 times higher than the number of samples of the minority one. In

[1] https://archive.ics.uci.edu
[2] https://www.ppdai.com
[3] https://www.kaggle.com/c/GiveMeSomeCredit
[4] https://www.lendingclub.com

LC2017Q1, the majority class has 77.45 times more samples than the minority one.

Readers may notice that three frequently used credit scoring datasets, *Australian*, *German*, and *Japanese*, are not on our list. We decide to exclude these datasets because they are almost balanced. Australian and Japanese have an $IR = 1,24$. German has an $IR = 2.33$. As our approach is designed for imbalanced datasets, it is not effective for almost balanced ones. Next, we present our data preparation steps.

We perform the following data preprocessing steps. First, we use one-hot encoding to transform each categorical feature with N values in N binary features. We also filled the missing values with the mean/mode for numeric/nominal features. Numeric features were standardized by removing the mean and scale the data to unit variance.

5.2.2 Competitors

To evaluate the performance of the proposed approach, we choose, beyond the static approach, two state-of-the-art competitors. Roy *et al.* (2018) proposed the first one. This approach consists of training a bagging pool generator in which each base learner receives balanced training data produced by an over-sampling method. Besides that, they also use an over-sampling method to generate the DSEL using the entire training data.

We also compare our technique with the approach proposed by García *et al.* (2018) to multi-class imbalanced datasets. This approach consists of two components: the generation of balanced training datasets and a weighted mechanism to highlight the competence of base learners that are more powerful in classifying examples in the region of underrepresented competence.

5.2.3 Ensemble method and base learners

We use bagging implementation available in sklearn[5]. To handle the imbalance level of the datasets, as Roy *et al.* (2018), we modify it to include an additional step to balance the training set of each bagging iteration using SMOTE.

We use the *bootstrap* feature of bagging to guarantee the diversity between the data available to the base learners and the DSEL. *Bootstrap* means that the samples are drawn with replacement, and each base learner is built with $(1 - \frac{1}{e}) \times \#samples$. It contributes to the diversity of the base learners, important to the performance of the ensemble, and also prevents overfitting

[5] https://scikit-learn.org/stable/

in the dynamic selection technique once no individual base learner uses all DSEL data in the training step.

We use the same base learner, Classification and Regression Trees (CART), for all evaluated approaches. This classifier is one of the best data mining algorithms, and it was employed as a base learner of ensembles in many previous papers (GARCÍA *et al.*, 2018). We also: (i) use 100 base learners (GARCÍA *et al.*, 2018), (ii) define that the misclassification cost of a false negative, a delinquent loan recognized as a good one, is equivalent to five false positives, good loans recognized as bad ones (WEST, 2000).

We combine the learners in a bagging approach to build the static model. To find the best model for each dataset, we perform a grid search to evaluate four different minimum numbers of samples to split a node in the decision tree. This hyper-parameter is used as a regularization factor in the decision tree training. The evaluated values are $min\_samples\_split \in [2^{-4}, 2^{-5}, 2^{-6}, 2^{-7}]$, and they represent the fraction of the number of samples of the dataset. SMOTE, KNORA-Union, and DES-MI also have hyper-parameters. As in Roy *et al.* (2018), we use $k_{neighbors} = 5$ in SMOTE. As previous papers (ROY *et al.*, 2018; CRUZ *et al.*, 2018), the size of the region of competence (neighborhood size) K is 7 for all KNORA-U, KNORA-IU, and DES-MI. For the remaining DES-MI hyper-parameters, we use the recommended by García *et al.* (2018).

5.2.4 Experimental setting

Figure 11 shows the experiment framework. First, we split the available data in training, 80%, and test, 20%, the top part of Figure 11. Then, we use a bagging ensemble with an additional SMOTE step to balance the data to perform a three-fold hyper-parameter grid search in the training data to find the best static model, "Static Model" box in Figure 11. Then, we use the best model of a grid search cross-validation in the test data to evaluate the ensemble. Next, we use this model combined with the dynamic selection techniques.

We apply two different setups with dynamic selection techniques. First, with KNORA-IU and DES-MI, we use the entire available training data to compute the competence of each base learner, the dynamic selection dataset, DSEL. With KNORA-U, we apply SMOTE to balance the training data before using it as DSEL.

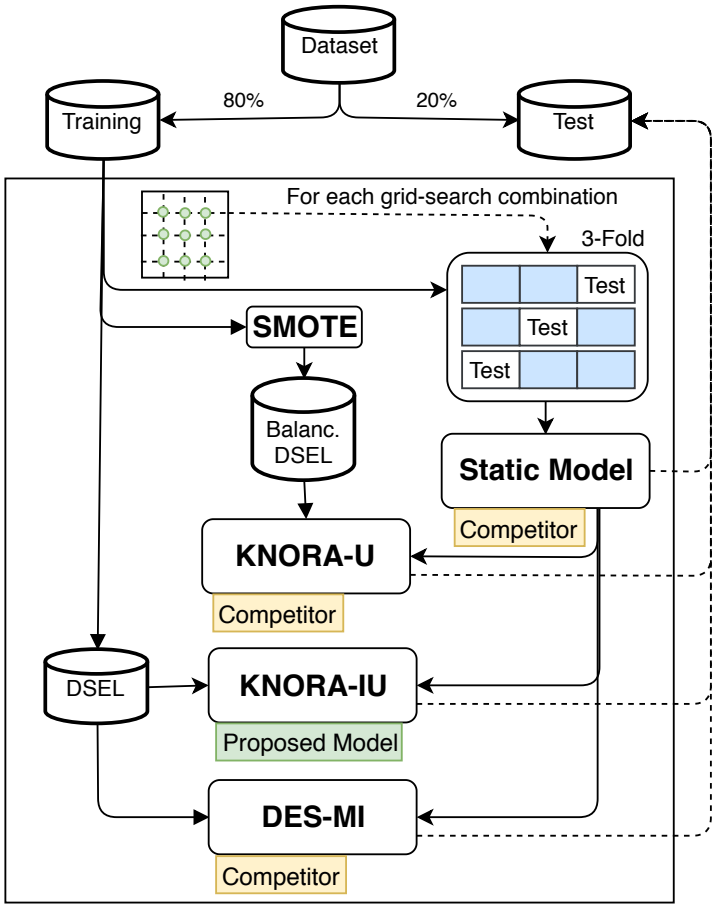


Figure 11 – The experimental framework for KNORA-IU

5.2.5 Evaluation measures and statistical test

The selection of evaluation measures plays a vital role in the final evaluation result. As cited before, the accuracy measure is widely used in classification but is not appropriate to an imbalanced dataset since a naive classifier always predicting the majority class achieves a high score.

In this chapter, we evaluate four metrics defined in Subsection 2.3.3 to measure the classifiers' predictive ability: Area under the ROC curve (AUC), H-measure, F-measure, and G-mean.

To assess the statistical significance and superiority of KNORA-IU over the competitors, we employ McNemar's test (DIETTERICH, 1998). McNemar's test is a simple parametric test. This test uses chi-square (χ^2) statistics, computed from two error matrices and given as $\chi^2 = \frac{(f_{12}-f_{21})^2}{(f_{12}+f_{21})}$, where f_{12} denotes the number of cases that are wrongly classified by classifier one but correctly classified by classifier two, and f_{21} indicates the number of cases that are correctly classified by classifier one but incorrectly classified by classifier two. Next, we present the results of the experiments described in this section.

5.3 Results and analysis

We present the results in this section. The first part shows the results of the predictive performance of KNORA-IU and the three competitors. The second part evaluates how this improvement occurs.

5.3.1 Predictive performance and statistical test

To answer **RQ5.1) "Does the use of the KNORA-Imbalanced Union - that uses FA^2 as the performance measure to compute the local competence of base classifiers - improve the classification performance of imbalanced credit scoring datasets?"**, we carried out the experimental analysis to compare our proposed KNORA-IU with the representative competitor approaches, i.e., Static, KNORA-U combined with SMOTE and DES-MI.

Table 16 shows the experimental results of the four dynamic selection techniques evaluated regarding the four performance measures and the five datasets. We group the results by performance measure, one over others. Table 16 also shows the imbalanced ratio (IR) of each dataset in the datasets' header.

Observing the results presented in Table 16, where the best result in each dataset for each measure is highlighted in bold-face, and the second-best result is underlined, we can quickly note that the proposed KNORA-UI outperforms the other compared methods in the four performance measures for the three first datasets, *Default* (DD), *PPDai* (PP), and *GiveMe* (GM). These datasets have a moderate imbalance ratio, between 3 and 14. KNORA-IU achieves a higher score in all performance measures for these datasets.

On the other hand, DES-MI achieves better results in high imbalanced datasets, *Iran (IR)* and *LC2017Q1 (LC)*, with an IR over 19. In the high-imbalanced datasets, KNORA-IU achieves the best score only in the F1-score of the *LC2017Q1* dataset.

Associating these results with the proposed criteria to define the competence of the base learners, we observe that the reduced performance is related to the higher use of the $accuracy(yt, yp)^2$ measure in Eq. (5.1). In a high imbalanced dataset, the probability of the k nearest neighbors of a query sample are all of the same class is higher. One possible workaround to this issue is increasing the number of samples that define the local region of the dynamic selection approach. This modification will increase the chance to use the F1-score as the metric to evaluate the competence of the classifiers. However, this increment can interfere with the

Table 16 – Classification results of each technique regarding each performance measure and dataset

Measures	**Techniques**[a]	**Datasets**[b] **(IR)**				
		DD (3.5)	**PP** (6.7)	**GM** (13.9)	**IR** (19.8)	**LC** (77.5)
H	KNORA-IU	**0.058**	**0.023**	**0.166**	0.136	0.044
	DES-MI	0.041	0.020	0.152	**0.157**	**0.054**
	KNORA-U/S	0.055	0.022	0.162	0.115	0.047
	Static	0.040	0.019	0.156	0.105	0.049
AUC	KNORA-IU	**0.598**	**0.557**	**0.719**	0.596	0.613
	DES-MI	0.570	0.544	0.709	**0.602**	**0.628**
	KNORA-U/S	0.591	0.552	0.716	0.591	0.618
	Static	0.569	0.544	0.712	0.588	0.622
G-mean	KNORA-IU	**0.494**	**0.409**	**0.689**	0.458	0.613
	DES-MI	0.406	0.335	0.673	**0.460**	**0.619**
	KNORA-U/S	0.469	0.375	0.684	0.455	0.615
	Static	0.408	0.335	0.678	0.454	0.617
F1-score	KNORA-IU	**0.400**	**0.251**	**0.220**	0.286	**0.037**
	DES-MI	0.385	0.246	0.212	**0.316**	0.036
	KNORA-U/S	0.397	0.249	0.218	0.261	0.036
	Static	0.385	0.245	0.215	0.250	0.036

[a] Dynamic selection techniques evaluated.
[b] DD: Default, PP: PPDai, GM: GiveMe, IR: Iran, and LC: LC2017Q1.

dynamic selection performance. Previous research in dynamic selection (BRITTO JR *et al.*, 2014) observed that seven neighbors are an optimal number of samples to define the local region.

We also observe that the performance difference between the static approach and the dynamic selection techniques is more negligible in high imbalanced datasets. The static method achieves the second-best results regarding all measures of the most imbalanced credit dataset, the *LC2017Q1*.

Table 17 shows McNemar's statistical test results when comparing KNORA-IU against competitors. Evaluating them, we observe that, except for *Iran* dataset, the test indicates that a statistical difference between the classification approaches. KNORA-IU is statistically better in the less imbalanced ones, *Default* (DD), *PPDai* (PP), and *GiveMe* (GM). However, all the competitors are statistically better than KNORA-IU in the most imbalanced dataset, *LC2017Q1* (LC).

We can conclude that KNORA-IU outperforms the competitors in moderate imbalanced datasets. Our approach achieved better results in credit scoring datasets with an imbalanced ratio between 3 and 14.

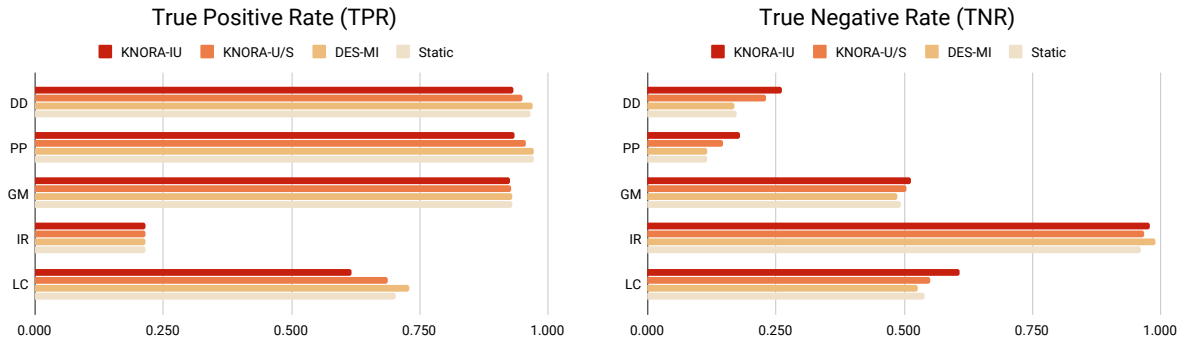


Figure 12 – True positive rate (left) and true negative rate (right) of the evaluated approaches for each dataset.

5.3.2 Improvement evaluation

In this section, we aim to answer the question **RQ5.2) "How does KNORA-IU improve the classification performance of imbalanced credit scoring datasets?"**. To answer it, we evaluate the true positive rate (TPR) and the true negative rate (TNR) of each classification approach in each dataset.

Figure 12 shows the TPR, the rate of correctly recognized defaulters, and TNR, the rate of correctly recognized good payers, of the evaluated methods for all datasets. Analyzing the TNR in Figure 12, we observe that, except for *Iran (IR)* dataset, KNORA-IU achieves the highest TNR for all datasets evaluated. On the other hand, analyzing the TPR, we note that KNORA-IU produces slightly worse results in the moderate imbalanced datasets, *Default*, *PPDai*, and *GiveMe*. We can conclude that KNORA-IU can recognize more good payers, negative class, with almost the same competitors' performance in identifying bad payers, positive type.

Table 17 – Statistically significant differences regarding KNORA-IU[a]

Competitors	***p*-value per Dataset[b]**				
	DD[(c)]	**PP**[(c)]	**GM**[(c)]	**IR**[(c)]	**LC**[(c)]
DES-MI	0.00 (D)	0.00 (D)	0.00 (D)	0.50 (N)	0.00 (D)
KNORA-U/S	0.00 (D)	0.00 (D)	0.00 (D)	0.50 (N)	0.00 (D)
Static	0.00 (D)	0.00 (D)	0.00 (D)	0.25 (N)	0.00 (D)

[a] Statistical significance established with McNemars' test, $p = 0.01$.
[b] DD: Default, PP: PPDai, GM: GiveMe, IR: Iran, and LC: LC2017Q1.
[c] Regarding KNORA-IU: D: Statistically different, N: No difference.

5.4 Conclusion

This chapter has addressed the imbalanced credit scoring problem with dynamic selection classification. We propose an extension of KNORA-Union, the KNORA-Imbalance Union (KNORA-IU). This approach replaces the accuracy, the performance metric used to compute the competence of each base classifier, by a combination of F-measure and accuracy. The intuition is to use a more appropriate metric to evaluate imbalanced datasets. To this end, we use five real-world credit scoring datasets with an imbalance ratio varying from 3 to 77 to evaluate our approach. To assess the performance of KNORA-IU, we compare it with KNORA-U combined with SMOTE, DES-MI, and a static ensemble.

The results demonstrate that KNORA-IU is convenient for moderate imbalanced credit datasets. It outperforms the competitors regarding four measures for datasets with $IR < 14$. We also observe that KNORA-IU improves the true negative rate (TNR) with a slight loss in the true positive rate (TPR).

Though the results of the KNORA-IU are satisfactory, it has some limitations. First, it is unclear if the proposed method is helpful to solve other classification problems where the misclassification costs of the classes are not sharply different. One research direction is evaluating the proposed technique in datasets of different domains. Second, the proposed method does not perform well in high imbalanced datasets. This low performance can be related to the use of accuracy to define the local competence of the base classifiers. Another avenue for future research is evaluating an increment in the number of samples representing the local region in the dynamic selection technique. This increment increases the use of F1-score in high imbalanced datasets without reducing the performance of the dynamic selection technique.

6 REDUCED MINORITY K-NN COMBINED WITH KNORA-IMBALANCED UNION

The previous two chapters present different strategies to enhance the dynamic selection classification. First, RMkNN alters the local region definition to balance the number of class samples in overlapping regions. After, we use a novel performance measure, FA^2, to compute the local competence of base classifiers and propose the KNORA-Imbalanced Union (KNIU). This chapter evaluates the combination of these two strategies.

6.1 Preliminary and hypothesis

RMkNN and KNIU use different strategies to attenuate the imbalanced problem of dynamic ensemble selection techniques. In the following paragraphs, we describe the approach adopted by each one and the intuition used to combine them.

We observe in Chapter 4 that modifying kNN to select more minority class samples in overlapping areas of imbalanced credit scoring datasets improves the dynamic selection technique prediction. The intuition behind this improvement is the inclusion of more minority class samples in the local region definition of overlapping areas.

Latter, we observe in Chapter 5 that modifying the performance measure used to compute the local competence of the base classifiers also improves the prediction performance of dynamic selection classification techniques. We replace accuracy with a combination of F-measure with the square of accuracy. This modification increases the influence of the most locally competent classifiers of KNORA-Union dynamic selection.

Now, we investigate the combination of these two techniques. We believe that these two modifications can cooperate and improve even more the prediction performance of dynamic selection classification techniques in imbalanced credit scoring datasets.

To identify the improvement of these combinations, we repeat the experiment performed in Chapter 4, including KNORA-IU, as a dynamic selection technique. As this experiment evaluates eight ensembles and three DSEL generators (RMkNN and two sampling approaches), we increase 24 techniques' combinations to the 110 previous combinations of Chapter 4. The following section presents the results.

To evaluate this combination performance, we aim to answer the research question **RQ6.1) Does the use of the RMkNN combined with KNORA-IU improve the classification performance of imbalanced credit scoring datasets?**. The following two sections describe

how these two techniques work together and the experimental results of the combination.

6.2 How do RMkNN and KNIU work together?

Before answering RQ6.1, we evaluate how do RMkNN and KNIU work together. RMkNN alters the local region definition in the procedure to include more minority samples in the overlapping areas. This local region modification occurs by reducing the distance between the query sample and the minority class samples. On the other hand, KNIU uses a novel performance measure used to compute the local competence of the base classifiers. Instead of accuracy, we propose a combination of F-measure and the square of accuracy to define the local competence of the base classifiers.

To illustrate how this combination works, we define a small ensemble example with three linear base classifiers in a bi-dimensional data space. We illustrate a local region definition and the prediction fusion of these two techniques alone and their combination.

Figure 13 shows the neighborhood of a query sample represented by the red diamond 1. The left side of Figure 13 shows the local region definition using regular kNN. The seven nearest neighbors of the query sample have six samples of the majority class, blue circles, and one sample of the minority class, red circle. We draw a circle to facilitate the visualization of the seven nearest neighbors. The right side of Figure 13 shows the local region definition using the RMkNN. The new seven nearest neighbors of the query sample have five samples of the majority class and two samples of the minority class. In this part of this figure, we highlight the distance reduction between the query sample and the minority class samples. This distance reduction of RMkNN replaced one sample of the majority class and included a sample of the minority class in the nearest neighbors list. We also highlight the new set of nearest neighbors with a smaller circle.

Figure 13 also shows three linear binary classifiers $C1$, $C2$, e $C3$. We use these base classifiers to compute and compare the KNIU, KNU+RMkNN, and KNIU+RMkNN predictions. To compute this experiment, we consider that the positive class, the red circles, is represented by 1 and the negative class, the blue circles, is represented by -1.

Table 18 shows the computation of the predictions of the query sample 1 using the three compared approaches. The columns “Learner” and “Pred” show, respectively, the base classifiers and their predictions for the query sample 1. The next three columns contain the computation of the fusion procedure of the three dynamic selection classification approaches.

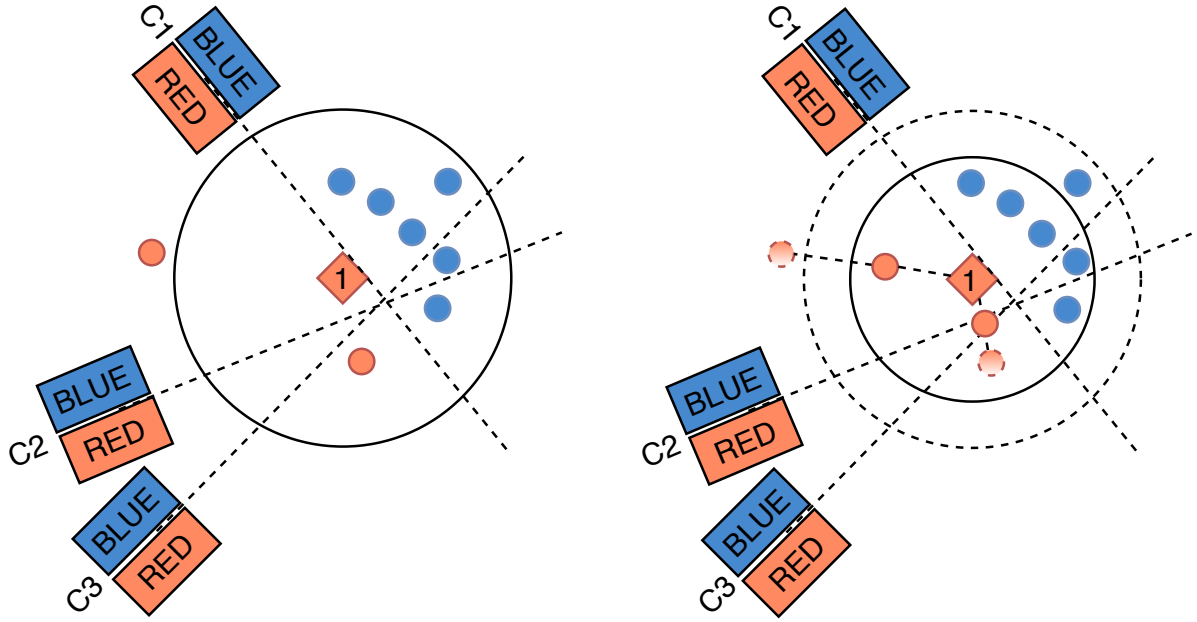


Figure 13 – Example of the combination of RMkNN and KNIU. The left side of the figure shows the local region definition without RMkNN. The right side shows the local region definition with RMkNN, reducing the distance between the query sample and the minority class samples.

Table 18 – Classification example results of KNIU, KNU+RMkNN, and KNIU+RMkNN

S[a]	**Learner**	**Pred**	**KNIU**		**KNU+RMkNN**		**KNIU+RMkNN**	
			FA^2	weight	Acc[b]	weight	FA^2	weight
1	C1	1	1	1	1	1	1	1
	C2	-1	0.67	-0.67	0.71	-0.71	0.50	-0.50
	C3	-1	0.40	-0.40	0.57	-0.57	0.40	-0.40
	DS prediction		-0.02 (-1)		-0.10 (-1)		0.03 (1)	

[a]**S** means the query sample evaluated.
[b]**Acc** means the accuracy of the learner in the neighborhood of the query sample.

The column "KNIU" contains the FA^2 measure for each base classifier and the corresponding weight of the base classifier in the final prediction. Next, the column "KNU+RMkNN" contains the accuracy of each base classifier in the local region of the query sample and the contribution of each base classifier. Finally, the column KNIU+RMkNN contains the FA^2 measure for each base classifier. The three columns called "weight" indicate each base classifier's contribution in the final prediction of the dynamic selection approaches.

Analyzing Table 18, we see that both KNIU and KNU+RMkNN can not predict correctly sample 1. However, KNIU+RMkNN combined predicts correctly the sample 1. The reasons why the combination of these techniques can predict correctly are: (i) FA^2 increases the

weight of the most competent learners in the predictions' fusion step; and (ii) RMkNN reduces the imbalance of the local area around the sample 1. Next, we repeat the experiments performed in Chapter 4 including the KNIU+RMkNN approach to measuring its performance in real credit scoring datasets.

6.3 Results and analysis

To answer **RQ6.1**, as Chapter 4, we perform two experiments. First, we compute the overall average ranking of 134 classification approaches. After, we compare the best estimator of the previous test with the credit scoring benchmarks. The following subsections describe each experiment.

6.3.1 Overall average ranking

In this experiment, we repeat the comparison of the combinations of pool generators, preprocessing approaches, and dynamic selection techniques of Table 10 including KNORA-IU (KNIU) as a dynamic selection technique. We evaluate the average rank of all 134 combinations (8 imbalanced ensembles $\times$ 5 selection approaches $\times$ 3 strategies to handle the DSEL $+$ 8 static imbalanced ensembles $+$ 6 credit scoring benchmarks) to reevaluate the best approaches to imbalanced credit scoring datasets.

As in Chapter 4, we start with the average rank of all 134 classification combinations. We compute the average rank of seven performance measures evaluated, AUC, H-measure, balanced accuracy, geometric mean, F1-score, F5-score, and recall (TPR). After, we compute the average of these averages to find a unique global rank. Table 19 shows the first 15 best combinations of this global rank. In this table, the gray calls indicate the lowest average rank of each performance measure. In green, we also highlight the combinations that use RMkNN and KNIU, in blue the combinations that use only RMkNN, and in yellow the combinations that use only KNIU.

As we can see, nine of the fifteen best combinations use at least one of the two proposed techniques, KNIU or RMkNN. Additionally, three of them use both techniques. Evaluating the pool generators of Table 19, we observe only three ensembles, BRND, BROT, and EASY. The best combination of the three pool generators is KMkNN with KNIU.

Analyzing the four first lines of Table 19, we see RNDF and ROTF combined with

RMkNN, KNIU, and KNU. Comparing KNIU and KNU, we observe that KNIU combinations achieve better rankings in measures that emphasize positive class misclassification, such as F5 and TPR. It means that RMkNN and KNIU reduce the number of default loans. On the other hand, KNU+RMkNN combinations achieve the lowest average rankings in the remaining performance measures, AUC, H-measure, BAcc, G-mean, and F1. It means that KNU+RMkNN grants more good loans than KNIU+RMkNN. Despite that, the global ranking of the KNIU combination still achieves a lower rank, indicating that the default loan reduction of KNIU+RMkNN is more significant than the good loan improvement of KNU+RMkNN.

6.3.2 Comparison of the best average ranking with the credit scoring benchmarks

After this preliminary evaluation, we compare the actual prediction results of Balanced Random Forest (BRND), the lowest rank of Table 19 with the best credit scoring benchmarks observed in Chapter 4, XGboost (XGB), Logistic Regression (LOGR), Random Forest (RNDF). We aim to identify the differences between these approaches.

For each dataset evaluated, Table 20 shows the average and the standard deviation of 5-fold execution explained in Figure 7. Here, we highlight the best result of each dataset and each performance measure in bold and dark gray. The second-best result is also highlighted in light gray. For each approach and each dataset, Table 20 shows seven performance measures, AUC, H-measure, BAcc, G-mean, F1-score, F5-score, and True Positive Rate (TPR).

We begin the analysis comparing the BRND+KNIU+RMkNN and BRND+KNU+RMkNN. We observe BRND+KNIU+RMkNN achieves the best result 24 times in the 49 possible (seven dataset and seven performance measures). On the other hand, BRND+KNU+RMkNN achieves the best score only once. These results lead us to conclude that KNIU improves the performance

Table 19 – Average ranking of all 134 techniques

Appr.	Selection	**Performance Measures** [average ranking (standard deviation)]							Avg
		AUC	H	BAcc	G-mean	F1	F5	TPR	
BRND	KNIU+RMkNN	21.7 (17.4)	22.5 (19.5)	14.5 (17.2)	15.4 (19.1)	26.1 (22.1)	**19.9 (18.7)**	25.5 (17.3)	21.2
BRND	KNU+RMkNN	17.1 (17.1)	20.4 (17.8)	**14.1 (16.7)**	15.5 (18.6)	23.9 (20.4)	23.9 (20.3)	29.1 (20.2)	21.5
BROT	KNIU+RMkNN	22.2 (14.5)	23.1 (23.0)	16.9 (13.7)	15.7 (13.1)	25.3 (15.6)	21.2 (14.6)	26.0 (13.1)	21.9
BROT	KNU+RMkNN	**16.5 (11.4)**	**19.8 (22.4)**	16.0 (16.5)	**15.3 (15.6)**	**21.9 (16.5)**	25.9 (19.0)	31.4 (14.9)	22.1
BROT	KNU+SMTE	19.2 (10.5)	25.9 (17.9)	19.6 (12.2)	18.9 (13.1)	28.1 (14.8)	25.4 (15.6)	29.6 (15.2)	24.4
BRND	KNU+SMTE	19.0 (11.2)	27.7 (17.3)	18.4 (15.8)	18.8 (17.6)	31.1 (19.6)	24.7 (18.3)	28.8 (18.2)	24.6
BRND	STATIC	20.1 (22.9)	32.7 (24.0)	18.8 (19.1)	19.4 (19.7)	38.3 (29.8)	23.1 (18.3)	23.9 (18.9)	25.1
BRND	KNIU+SMTE	24.9 (13.4)	31.7 (20.0)	19.4 (16.2)	19.9 (18.0)	33.4 (22.4)	22.5 (19.0)	26.3 (18.7)	25.5
BROT	STATIC	19.3 (17.7)	31.5 (20.3)	20.3 (15.5)	20.0 (16.4)	37.5 (24.8)	24.7 (16.1)	25.4 (17.0)	25.6
BROT	KNU+RUS	20.8 (16.4)	30.9 (20.5)	19.9 (14.5)	19.3 (14.9)	35.9 (23.6)	25.5 (16.5)	27.8 (16.6)	25.9
BROT	KNIU+SMTE	27.0 (13.1)	32.6 (21.0)	23.4 (12.2)	22.1 (13.4)	34.0 (16.0)	21.9 (13.5)	25.3 (13.9)	26.4
BRND	KNU+RUS	20.3 (20.5)	33.3 (22.9)	20.9 (20.1)	21.3 (20.4)	38.6 (29.2)	25.4 (18.6)	27.9 (19.2)	26.8
BROT	KNIU+RUS	28.6 (18.0)	36.6 (23.0)	22.7 (15.3)	21.7 (15.7)	41.2 (26.1)	23.1 (15.5)	23.9 (16.3)	27.7
BRND	KNIU+RUS	31.7 (21.9)	37.2 (24.3)	22.4 (21.2)	22.1 (20.9)	42.5 (31.7)	22.2 (18.6)	**21.9 (18.6)**	27.8
EASY	KNIU+RMkNN	29.2 (24.3)	30.7 (26.4)	30.2 (34.0)	29.3 (33.8)	40.0 (27.9)	27.1 (25.4)	29.1 (24.5)	30.6

Table 20 – Balanced Random Forest combined with KNORA-U and RMkNN compared with state-of-the-art classifiers in credit scoring problem

Dataset	Classif.	Selection	Performance Measures						
			AUC	H	BAcc	G-mean	F1	F5	TPR
German	XGB	STATIC	0.79 (0.02)	0.23 (0.04)	0.72 (0.02)	0.72 (0.02)	0.61 (0.02)	0.67 (0.04)	0.67 (0.04)
	LOGR	STATIC	0.80 (0.03)	0.26 (0.05)	0.74 (0.03)	0.74 (0.03)	0.63 (0.03)	0.72 (0.08)	0.73 (0.08)
	RNDF	STATIC	0.79 (0.03)	0.23 (0.04)	0.71 (0.02)	0.71 (0.02)	0.60 (0.03)	0.66 (0.07)	0.66 (0.07)
	BRND	STATIC	0.80 (0.03)	0.24 (0.07)	0.73 (0.03)	0.73 (0.03)	0.62 (0.04)	0.75 (0.04)	0.76 (0.04)
	BRND	KNU+RMk	**0.80 (0.03)**	0.26 (0.06)	0.74 (0.03)	0.74 (0.03)	0.63 (0.04)	0.76 (0.04)	0.77 (0.04)
	BRND	KNIU+RMk	0.80 (0.03)	**0.27 (0.07)**	**0.74 (0.03)**	**0.74 (0.03)**	**0.63 (0.04)**	**0.78 (0.03)**	**0.79 (0.03)**
Default	XGB	STATIC	0.78 (0.02)	0.23 (0.04)	0.71 (0.02)	**0.71 (0.02)**	0.54 (0.03)	0.62 (0.03)	0.62 (0.03)
	LOGR	STATIC	0.72 (0.02)	0.14 (0.03)	0.67 (0.02)	0.67 (0.02)	0.48 (0.02)	0.62 (0.03)	0.64 (0.03)
	RNDF	STATIC	**0.78 (0.02)**	**0.24 (0.04)**	**0.71 (0.02)**	0.70 (0.02)	**0.55 (0.03)**	0.59 (0.03)	0.60 (0.03)
	BRND	STATIC	0.78 (0.02)	0.21 (0.04)	0.71 (0.02)	0.71 (0.02)	0.53 (0.02)	0.63 (0.03)	0.64 (0.03)
	BRND	KNU+RMk	0.78 (0.02)	0.22 (0.04)	0.71 (0.02)	0.71 (0.02)	0.53 (0.03)	0.63 (0.03)	0.64 (0.03)
	BRND	KNIU+RMk	0.78 (0.02)	0.21 (0.04)	0.71 (0.02)	0.71 (0.02)	0.53 (0.02)	**0.63 (0.03)**	**0.64 (0.03)**
PPDai	XGB	STATIC	**0.63 (0.05)**	**0.02 (0.02)**	**0.56 (0.04)**	**0.46 (0.26)**	**0.21 (0.12)**	0.36 (0.21)	0.38 (0.22)
	LOGR	STATIC	0.63 (0.03)	0.02 (0.04)	0.52 (0.04)	0.15 (0.20)	0.07 (0.13)	0.06 (0.12)	0.06 (0.12)
	RNDF	STATIC	0.63 (0.04)	0.02 (0.02)	0.56 (0.04)	0.44 (0.25)	0.20 (0.12)	0.41 (0.27)	0.45 (0.31)
	BRND	STATIC	0.61 (0.05)	0.02 (0.01)	0.55 (0.03)	0.43 (0.23)	0.20 (0.11)	**0.46 (0.30)**	**0.52 (0.35)**
	BRND	KNU+RMk	0.61 (0.04)	0.02 (0.01)	0.55 (0.03)	0.43 (0.23)	0.20 (0.11)	0.45 (0.30)	0.51 (0.35)
	BRND	KNIU+RMk	0.60 (0.05)	0.02 (0.01)	0.55 (0.03)	0.43 (0.23)	0.20 (0.11)	0.46 (0.30)	0.52 (0.35)
Private	XGB	STATIC	0.68 (0.04)	0.07 (0.05)	0.60 (0.06)	0.54 (0.13)	0.24 (0.07)	0.37 (0.17)	0.39 (0.19)
	LOGR	STATIC	0.67 (0.05)	0.06 (0.03)	0.62 (0.03)	0.62 (0.03)	0.24 (0.02)	0.55 (0.06)	0.61 (0.08)
	RNDF	STATIC	0.72 (0.02)	0.11 (0.06)	0.62 (0.05)	0.54 (0.12)	0.28 (0.07)	0.34 (0.14)	0.35 (0.15)
	BRND	STATIC	0.72 (0.03)	0.10 (0.02)	0.66 (0.02)	0.66 (0.02)	0.28 (0.01)	0.60 (0.05)	0.67 (0.06)
	BRND	KNU+RMk	0.72 (0.03)	0.11 (0.03)	0.67 (0.03)	0.67 (0.02)	0.28 (0.01)	0.62 (0.06)	**0.69 (0.07)**
	BRND	KNIU+RMk	**0.72 (0.03)**	**0.11 (0.03)**	**0.67 (0.02)**	**0.67 (0.02)**	**0.28 (0.01)**	**0.62 (0.06)**	**0.69 (0.07)**
GiveMe	XGB	STATIC	**0.86 (0.00)**	0.34 (0.01)	**0.79 (0.00)**	**0.79 (0.00)**	0.34 (0.00)	0.70 (0.01)	0.77 (0.01)
	LOGR	STATIC	0.81 (0.01)	0.25 (0.01)	0.73 (0.00)	0.73 (0.00)	0.31 (0.01)	0.59 (0.01)	0.64 (0.01)
	RNDF	STATIC	0.86 (0.00)	**0.35 (0.02)**	0.78 (0.00)	0.78 (0.01)	**0.36 (0.02)**	0.67 (0.02)	0.73 (0.03)
	BRND	STATIC	0.86 (0.00)	0.34 (0.01)	0.79 (0.00)	0.79 (0.00)	0.33 (0.00)	**0.71 (0.00)**	**0.78 (0.00)**
	BRND	KNU+RMk	0.86 (0.00)	0.34 (0.01)	0.79 (0.00)	0.79 (0.00)	0.34 (0.00)	0.70 (0.00)	0.77 (0.01)
	BRND	KNIU+RMk	0.86 (0.00)	0.34 (0.01)	0.78 (0.00)	0.78 (0.00)	0.34 (0.00)	0.70 (0.00)	0.77 (0.00)
Iran	XGB	STATIC	0.76 (0.06)	0.15 (0.06)	0.61 (0.03)	0.49 (0.06)	**0.27 (0.06)**	0.25 (0.06)	0.25 (0.06)
	LOGR	STATIC	0.78 (0.06)	0.00 (0.00)	0.50 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
	RNDF	STATIC	0.79 (0.04)	0.11 (0.04)	0.57 (0.02)	0.37 (0.06)	0.23 (0.06)	0.15 (0.05)	0.14 (0.05)
	BRND	STATIC	0.77 (0.05)	0.18 (0.08)	0.71 (0.05)	0.71 (0.05)	0.19 (0.04)	0.58 (0.07)	**0.71 (0.08)**
	BRND	KNU+RMk	0.81 (0.07)	0.28 (0.12)	0.73 (0.07)	0.72 (0.08)	0.27 (0.06)	0.57 (0.12)	0.63 (0.15)
	BRND	KNIU+RMk	**0.82 (0.06)**	**0.29 (0.14)**	**0.74 (0.08)**	**0.73 (0.09)**	0.26 (0.07)	**0.59 (0.14)**	0.67 (0.16)
LC2015	XGB	STATIC	**0.71 (0.04)**	0.08 (0.02)	0.64 (0.02)	0.62 (0.04)	**0.05 (0.00)**	0.29 (0.03)	0.52 (0.08)
	LOGR	STATIC	0.69 (0.02)	0.03 (0.04)	0.56 (0.08)	0.25 (0.35)	0.02 (0.02)	0.12 (0.16)	0.23 (0.32)
	RNDF	STATIC	0.71 (0.03)	0.03 (0.05)	0.54 (0.06)	0.20 (0.29)	0.02 (0.03)	0.09 (0.13)	0.12 (0.19)
	BRND	STATIC	0.70 (0.03)	0.08 (0.01)	0.65 (0.01)	0.65 (0.01)	0.04 (0.00)	0.32 (0.01)	**0.68 (0.03)**
	BRND	KNU+RMk	0.70 (0.03)	0.09 (0.03)	0.66 (0.02)	0.66 (0.02)	0.05 (0.00)	0.32 (0.02)	0.63 (0.04)
	BRND	KNIU+RMk	0.69 (0.03)	**0.10 (0.03)**	**0.66 (0.03)**	**0.66 (0.03)**	0.05 (0.00)	**0.32 (0.03)**	0.62 (0.05)

of BRND combined with RMkNN regarding the use of regular KNU.

Next, we observe the superiority of BRND+KNIU+RMkNN in the performance measures that give more importance to the positive class misclassification, F5-score, and Recall (TPR). Regarding these two measures, BRND+KNIU+RMkNN achieves the best result or the second-best result in 78.5% (11 times in 14 results). It means that the proposed combination grants fewer default loans.

Additionally, the superiority of BRND+KNIU+RMkNN also occurs among the measures that gives the same weight to misclassification of both classes, AUC, H-measure, BAcc, G-mean, and F1-score. The proposed classification approach achieves the best result in 45,7% (16 of 35 results).

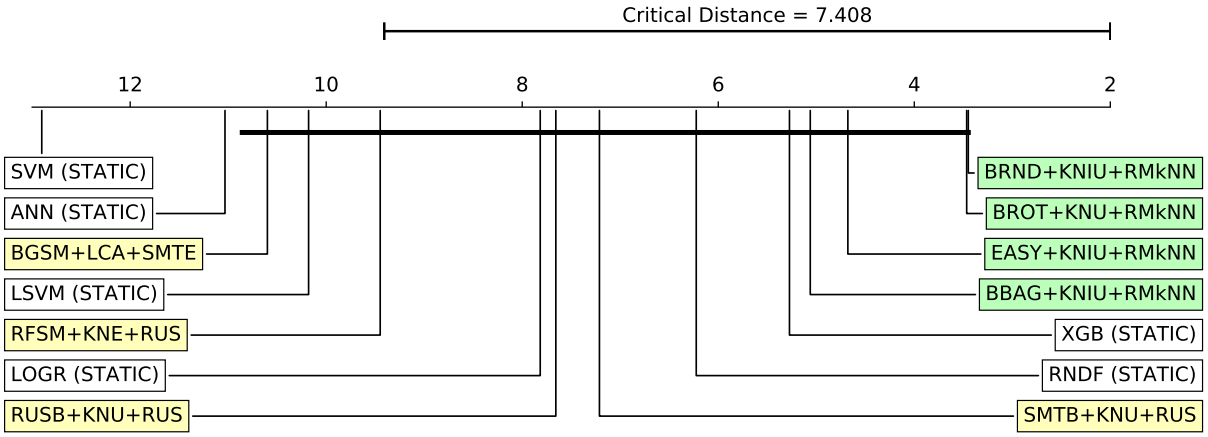


Figure 14 – The average rank of the best combinations including RMkNN and KNORA-IU.

6.4 Discussion

As in Chapter 4, we now investigate the best combination strategy among all evaluated. To achieve it, we compute a new average rank of the best results of each ensemble combination and the credit scoring benchmarks. Applying the Friedman test on the average ranking of these fourteen classifiers, we get a Friedman test statistic $= 90.51$, and a $p-value < 0.005$. As the Friedman test result is significant ($p < 0.005$), we can apply the post hoc Nemenyi test to the distribution.

Figure 14 shows the average ranks of these best combinations and the Critical Distance of the Nemenyi test. This figure shows that balanced random forest (BRND) combined with KNORA Imbalanced Union (KNIU) and using RMkNN to generate the DSEL is the best approach, the lowest average rank. This approach is statistically better than Artificial Neural Networks and Support Vector Machine, as indicated by the critical distance bar.

We also observe that RMkNN and is present on four best combinations of eight ensembles. They are highlighted in green on Figure 14, and they are Balanced Random Forest (BRND), Balanced Rotation Forest(BROT), Easy Ensemble (EASY), and Balanced Bagging (BBAG). The following three best ranking positions are combinations that use Random Undersampling (RUS) to generate the dynamic selection dataset (DSEL). They are: SMOTEBoost (SMTB), RUSBoost (RUSB), and Random Forest SMOTE (RFSM). Only the last position, Bagging SMOTE (BGSM), uses SMOTE to generate the DSEL. Figure 14 highlights these last four combinations in yellow.

With these experiments, we observe that KNIU combined with RMkNN improves the use of RMkNN combined with KNU. We also observe that KNORA-Imbalanced Union

(KNIU) is an excellent dynamic selection technique to combine with imbalanced ensembles. After, we observe that BRND is the best pool generator to combine with KNIU.

6.5 Conclusion

In this chapter, we evaluated the combination of the two techniques presented in Chapters 4 and 5, Reduced Minority kNN, and KNORA-Imbalanced Union, respectively. First, we offer a hypothesis about the use of these two techniques together. After, we demonstrate by an example of how RMkNN and KNIU work together. Next, we compute the average ranking of the imbalanced ensembles, imbalanced preprocessing, dynamic selection techniques, and credit scoring benchmarks of Table 10.

We conclude that the combination of RMkNN and KNIU improves the prediction performance of three imbalanced ensembles regarding the use of RMkNN alone. We also observe that RMkNN and KNIU improve the performance regarding measures that give more weight to positive class misclassification, such as F5 and TPR.

7 CONCLUSIONS AND FUTURE WORKS

7.1 Conlcusions

Credit scoring has become an efficient tool for financial institutions to discriminate against potential default borrowers and manage credit risk. Any slight improvement in the default discrimination can produce a high profit. This benefit motivates several researchers to develop credit scoring works in recent years.

However, because of regulatory constraints (Basel Accords), few works evaluated the use of dynamic selection classification to credit scoring problem. The main issue of these regulatory constraints and dynamic selection is using different models for different borrowers.

In this thesis, we presented several contributions; to allow the use of dynamic selection classification techniques in credit scoring models; to improve the local region definition of dynamic selection techniques when applied to imbalanced credit scoring problem; to enhance the local competence definition of base classifiers of dynamic selection techniques.

First, we presented an imbalanced credit scoring benchmark. We did it by answering the research questions **RQ3.1) "How do the more recent techniques in machine learning improve the credit scoring prediction performance compare to well-known state-of-the-art approaches?"**; and **RQ3.2) "Is there any better approach for each specific level of imbalanced data?"** described on Chapter 3. We concluded that Random Forest and Gradient Boosting have terrific performance regardless of the dataset's imbalance level.

Next, we presented a study about the suitability of dynamic selection techniques to imbalanced credit scoring problem. We did it answering the research question **RQ4.1) Are dynamic selection techniques appropriate for imbalanced credit scoring problems?** described in Chapter 4. We evaluated complexity measures of seven credit scoring datasets and compared the results with the complexity measures of 12 datasets of other domains. We found that the credit scoring datasets are, on average, more complex than datasets of other fields. As dynamic selection techniques are appropriate for complex datasets, we concluded that dynamic selection techniques are suitable for credit scoring.

Next, we investigate the regulatory constraints of Basel Accords related to dynamic selection by answering the research question **RQ4.2) Is there an equivalence between a dynamic selection technique and a static one?** also described in Chapter 4. We found a static ensemble equivalent to the KNORA-Union dynamic selection technique. This finding can be a

starting point for the use of dynamic selection techniques in credit scoring problems.

Later, we described Reduced Minority kNN (RMkNN), a modification on regular kNN to redefine the local region of a dynamic selection technique. We presented the intuition behind RMkNN that considers including more minority class samples on the local region definition without a mandatory inclusion of minority samples. This work relates to the research question **RQ4.3) Does the RMkNN improve the prediction performance of kNN?**. We test RMkNN and kNN on seven credit scoring datasets, and we concluded that RMkNN improves the prediction performance of kNN on imbalanced credit data.

Also related to RMkNN, we investigated the research question **RQ4.4) "Does the use of the RMkNN technique - that defines a novel local-competence region of dynamic selection techniques - improve the classification performance of imbalanced credit scoring datasets?"**. broad set of experiments with seven imbalanced credit scoring datasets, eight imbalanced ensembles, four dynamic selection techniques, and two baseline procedures to generate the dynamic selection dataset (DSEL), and compared the results with six state-of-art credit scoring classifiers to empirically conclude that RMkNN improves the prediction performance of imbalanced credit scoring problems.

After, we moved from the local region definition point of view to the competence evaluation of base classifiers in a dynamic selection technique. We described KNORA-Imbalanced Union (KNIU), the KNORA-Union technique with a novel performance measure to compute the local competence of base classifiers, the FA^2. Our aim in proposing this novel performance measure is to replace accuracy that does not achieve good results in imbalanced datasets. FA^2 is a combination of F-measure and the square of accuracy. To answer the research question **RQ5.1) Does the use of the KNORA-Imbalanced Union - that uses FA^2 as a performance measure to compute the local competence of base classifiers - improve the classification performance of imbalanced credit scoring datasets?**, we performed an experimental evaluation with five credit scoring datasets, and two baselines to empirically conclude that KNIU improves the prediction performance regarding H-measure, AUC, G-mean, and F1-score of moderate imbalanced datasets.

Also related to KNIU, we investigated the research question **RQ5.2) "How does KNORA-IU improve the classification performance of imbalanced credit scoring datasets?"**. We performed experiments to conclude that KNIU improves the true negative rate without decreasing the true positive rate in moderate imbalanced credit data.

Lastly, we evaluated the performance of RMkNN combined with KNORA-IU. To answer the research question **RQ6.1) "Does the use of the RMkNN combined with KNORA-IU improve the classification performance of imbalanced credit scoring datasets?"**, we repeated the experiments performed on Chapter 4 to compare the performance of RMkNN combined KNIU with RMkNN combined with KNU and with state-of-the-art credit scoring classification approaches. We empirically concluded that RMkNN combined with KNORA-IU outperforms RMkNN combined with KNORA-U and other state-of-the-art credit scoring classification approaches.

7.2 Future Works

We see six future works to contribute to the imbalanced credit scoring prediction field.

7.2.1 Investigate the equivalence between static and dynamic selection techniques

Nowadays, the practical use of DS techniques is not allowed in the credit scoring field. The regulation agreements of this field, Basel Accords (ATIK, 2010), require that the use of the same model for all customers. As DS techniques select dynamically the base models that predict each sample, they do not meet the standards.

However, as we see on Subsection 4.2.2, a DS technique can be reduced to a static one. An interesting future direction is the exploration of this equivalence to remove the restriction of the practical use of DS in the credit scoring field.

7.2.2 Improve the performance of RMkNN

The computational cost of using RMkNN in DS techniques is greater than the use of regular kNN. The reason is that, as we need to reduce only the distance between the query sample and the minority class samples of DSEL, we perform kNN twice, one for each class. One future work is to optimize the computational cost of RMkNN.

7.2.3 Include other parameters in the reduction function of RMkNN

An interesting research direction is modifying Eq. 4.4 of Chapter 4 to consider other measures of the dataset. For instance, we can combine the imbalanced ratio (IR) with a

complexity measure presented in Section 4.2. Maybe combining these measures of the dataset can produce a better local region definition to dynamic selection techniques.

7.2.4 Evaluation of performance measures

The use of a performance measure aligned with profit's objectives can improve the gains. We proposed KNORA-Imbalanced Union (KNORA-IU), the KNORA-Union that uses a new performance measure to compute the local competence of the base models. This performance measure is the combination of F-measure and the square of accuracy. The intuition behind this measure is to reduce the poor performance of accuracy in imbalanced datasets. However, a profit-based measure, such as Verbraken *et al.* (2014), can enhance the finance institutions' profit. An exciting and profitable future direction is to find a performance measure aligned to lenders' profit.

Regarding KNORA-IU, we consider F-measure with a $\beta = 1$. This value of β gives the same weight for precision and recall. Another future direction is testing KNORA-IU with different values for β.

7.2.5 Credit scoring and Ethics

An important future direction in the credit scoring field is the ethical aspect. Complex ensembles models are hard to explain and interpret (DOŠILOVIĆ *et al.*, 2018). Explanations for the model decisions are crucial to guarantee ethical decisions.

7.2.6 Compute the effective gain in term of money

Another future direction is the evaluation of the adequate profit produced by the adoption of a prediction system. Serrano-Cinca e Gutiérrez-Nieto (2016) proposed a profit scoring system instead of the regular credit scoring ones. Instead of predicting the probability of default, the authors focus on predicting the expected profitability. We believe the approaches proposed in this thesis can be evaluated as a profit scoring system.

REFERENCES

ABELLÁN, J.; CASTELLANO, J. G. A comparative study on base classifiers in ensemble methods for credit scoring. **Expert Systems with Applications**, [S. l.], v. 73, p. 1–10, 2017. ISSN 0957-4174. Disponível em: https://www.sciencedirect.com/science/article/pii/S0957417416306947. Acesso em: 09 jun. 2020.

ALA'RAJ, M.; ABBOD, M. F. Classifiers consensus system approach for credit scoring. **Knowledge-Based Systems**, [*S. l.*], v. 104, p. 89–105, 2016. ISSN 0950-7051. Disponível em: https://www.sciencedirect.com/science/article/pii/S0950705116300569. Acesso em: 09 jun. 2020.

ALA'RAJ, M.; ABBOD, M. F. A new hybrid ensemble credit scoring model based on classifiers consensus system approach. **Expert Systems with Applications**, [*S. l.*], v. 64, p. 36–55, 2016. ISSN 0957-4174. Disponível em: https://www.sciencedirect.com/science/article/pii/S0957417416303621. Acesso em: 09 jun. 2020.

ALTMAN, E. I.; HALDEMAN, R. G.; NARAYANAN, P. Zetatm analysis a new model to identify bankruptcy risk of corporations. **Journal of Banking & Finance**, [*S. l.*], v. 1, n. 1, p. 29–54, 1977. ISSN 0378-4266. Disponível em: https://www.sciencedirect.com/science/article/pii/0378426677900176. Acesso em: 09 jun. 2020.

ATIK, J. Basel ii: A post-crisis post mortem. **Transnat'l L. & Contemp. Probs.**, [*S. l.*], v. 19, p. 731, 2010. Disponível em: https://ssrn.com/abstract=1725004. Acesso em: 09 jun. 2020.

BAESENS, B.; GESTEL, T. V.; VIAENE, S.; STEPANOVA, M.; SUYKENS, J.; VANTHIENEN, J. Benchmarking state-of-the-art classification algorithms for credit scoring. **Journal of the operational research society**, [*S. l.*], v. 54, n. 6, p. 627–635, 2003. Disponível em: https://doi.org/10.1057/palgrave.jors.2601545. Acesso em: 09 jun. 2020.

BARANDELA, R.; VALDOVINOS, R. M.; SÁNCHEZ, J. S. New applications of ensembles of classifiers. **Pattern Analysis & Applications**, [*S. l.*], v. 6, n. 3, p. 245–256, 2003. Disponível em: https://link.springer.com/article/10.1007/s10044-003-0192-z. Acesso em: 09 jun. 2020.

BEQUÉ, A.; LESSMANN, S. Extreme learning machines for credit scoring: An empirical evaluation. **Expert Systems with Applications**, [*S. l.*], v. 86, p. 42–53, 2017. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0957417417303718. Acesso em: 09 jun. 2020.

BREIMAN, L. Bagging predictors. **Machine learning**, [*S. l.*], v. 24, n. 2, p. 123–140, 1996. Disponível em: https://link.springer.com/article/10.1007/BF00058655. Acesso em: 09 jun. 2020.

BREIMAN, L.; FRIEDMAN, J.; STONE, C. J.; OLSHEN, R. A. **Classification and regression trees**. [*S. l.*]: CRC press, 1984.

BRITTO JR, A. S.; SABOURIN, R.; OLIVEIRA, L. E. Dynamic selection of classifiers—a comprehensive review. **Pattern Recognition**, [*S. l.*], v. 47, n. 11, p. 3665–3680, 2014. Disponível em: https://www.sciencedirect.com/science/article/pii/S0031320314001885. Acesso em: 09 jun. 2020.

BROWN, I.; MUES, C. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. **Expert Systems with Applications**, [*S. l.*], v. 39, n. 3, p. 3446–3453,

2012. Disponível em: https://www.sciencedirect.com/science/article/pii/S095741741101342X. Acesso em: 09 jun. 2020.

CHAWLA, N. V.; BOWYER, K. W.; HALL, L. O.; KEGELMEYER, W. P. Smote: synthetic minority over-sampling technique. **Journal of artificial intelligence research**, [*S. l.*], v. 16, p. 321–357, 2002. Disponível em: https://www.jair.org/index.php/jair/article/view/10302. Acesso em: 09 jun. 2020.

CHAWLA, N. V.; LAZAREVIC, A.; HALL, L. O.; BOWYER, K. W. Smoteboost: Improving prediction of the minority class in boosting. In: SPRINGER. **European Conference on Principles of Data Mining and Knowledge Discovery**. 2003. p. 107–119. Disponível em: https://link.springer.com/chapter/10.1007/978-3-540-39804-2_12. Acesso em: 09 jun. 2020.

CHEN, C.; LIAW, A.; BREIMAN, L. Using random forest to learn imbalanced data. **University of California**, Berkeley, v. 110, p. 1–12, 2004.

CHEN, S.; HE, H.; GARCIA, E. A. Ramoboost: ranked minority oversampling in boosting. **IEEE Transactions on Neural Networks**, [*S. l.*], v. 21, n. 10, p. 1624–1642, 2010. Disponível em: https://www.ele.uri.edu/faculty/he/PDFfiles/ramoboost.pdf. Acesso em: 09 jun. 2020.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: ACM. **Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining**. 2016. p. 785–794. Disponível em: https://dl.acm.org/doi/abs/10.1145/2939672.2939785. Acesso em: 09 jun. 2020.

COVER, T.; HART, P. Nearest neighbor pattern classification. **IEEE transactions on information theory**, [*S. l.*], v. 13, n. 1, p. 21–27, 1967. Disponível em: https://ieeexplore.ieee.org/document/1053964. Acesso em: 09 jun. 2020.

CRUZ, R. M.; HAFEMANN, L. G.; SABOURIN, R.; CAVALCANTI, G. D. Deslib: A dynamic ensemble selection library in python. **Journal of Machine Learning Research**, [*S. l.*], v. 21, n. 8, p. 1–5, 2020. Disponível em: https://www.jmlr.org/papers/volume21/18-144/18-144.pdf. Acesso em: 09 jun. 2020.

CRUZ, R. M.; SABOURIN, R.; CAVALCANTI, G. D. Dynamic classifier selection: Recent advances and perspectives. **Information Fusion**, [*S. l.*], v. 41, p. 195–216, 2018. Disponível em: https://doi.org/10.1016/j.inffus.2017.09.010. Acesso em: 09 jun. 2020.

CRUZ, R. M.; SABOURIN, R.; CAVALCANTI, G. D.; REN, T. I. Meta-des: A dynamic ensemble selection framework using meta-learning. **Pattern recognition**, [*S. l.*], v. 48, n. 5, p. 1925–1935, 2015. Disponível em: https://doi.org/10.1016/j.patcog.2014.12.003. Acesso em: 09 jun. 2020.

DEMŠAR, J. Statistical comparisons of classifiers over multiple data sets. **Journal of Machine learning research**, [*S. l.*], v. 7, n. Jan, p. 1–30, 2006. Disponível em: https://www.jmlr.org/papers/volume7/demsar06a/demsar06a.pdf. Acesso em: 09 jun. 2020.

DIETTERICH, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. **Neural computation**, [*S. l.*], v. 10, n. 7, p. 1895–1923, 1998. Disponível em: https://doi.org/10.1162/089976698300017197. Acesso em: 09 jun. 2020.

DIETTERICH, T. G. Ensemble methods in machine learning. In: SPRINGER. **International workshop on multiple classifier systems**. 2000. p. 1–15. Disponível em: https://link.springer.com/chapter/10.1007/3-540-45014-9_1. Acesso em: 09 jun. 2020.

DOŠILOVIĆ, F. K.; BRČIĆ, M.; HLUPIĆ, N. Explainable artificial intelligence: A survey. In: IEEE. **2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)**. 2018. p. 0210–0215. Disponível em: https://ieeexplore.ieee.org/abstract/document/8400040. Acesso em: 09 jun. 2020.

DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern classification and scene analysis**. [*S. l.*]: Wiley New York, 1973. v. 3.

FENG, X.; XIAO, Z.; ZHONG, B.; QIU, J.; DONG, Y. Dynamic ensemble classification for credit scoring using soft probability. **Applied Soft Computing**, [*S. l.*], v. 65, n. C, p. 139–151, 2018. Disponível em: https://doi.org/10.1016/j.asoc.2018.01.021. Acesso em: 09 jun. 2020.

FERNÁNDEZ, A.; GARCÍA, S.; JESUS, M. J. del; HERRERA, F. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. **Fuzzy Sets and Systems**, North-Holland, [*S. l.*], v. 159, n. 18, p. 2378–2398, 2008. Disponível em: https://doi.org/10.1016/j.fss.2007.12.023. Acesso em: 09 jun. 2020.

FRIEDMAN, M. A comparison of alternative tests of significance for the problem of m rankings. **The Annals of Mathematical Statistics**, [*S. l.*], v. 11, n. 1, p. 86–92, 1940. Disponível em: https://www.jstor.org/stable/2235971. Acesso em: 09 jun. 2020.

GARCÍA, S.; ZHANG, Z.-L.; ALTALHI, A.; ALSHOMRANI, S.; HERRERA, F. Dynamic ensemble selection for multi-class imbalanced datasets. **Information Sciences**, [*S. l.*], v. 445, p. 22–37, 2018. Disponível em: https://doi.org/10.1016/j.ins.2018.03.002. Acesso em: 09 jun. 2020.

GARCÍA, V.; MARQUÉS, A. I.; SÁNCHEZ, J. S. Exploring the synergetic effects of sample types on the performance of ensembles for credit risk and corporate bankruptcy prediction. **Information Fusion**, [*S. l.*], v. 47, p. 88–101, 2019. Disponível em: https://doi.org/10.1016/j.inffus.2018.07.004. Acesso em: 09 jun. 2020.

GIACINTO, G.; ROLI, F. Methods for dynamic classifier selection. In: IEEE. **Proceedings 10th International Conference on Image Analysis and Processing**. 1999. p. 659–664. Disponível em: https://doi.org/10.1109/ICIAP.1999.797670. Acesso em: 09 jun. 2020.

HAIXIANG, G.; YIJING, L.; SHANG, J.; MINGYUN, G.; YUANYUE, H.; BING, G. Learning from class-imbalanced data: Review of methods and applications. **Expert Systems with Applications**, [*S. l.*], v. 73, p. 220–239, 2017. Disponível em: https://doi.org/10.1016/j.eswa.2016.12.035. Acesso em: 09 jun. 2020.

HAND, D. J. Measuring classifier performance: a coherent alternative to the area under the roc curve. **Machine learning**, [*S. l.*], v. 77, n. 1, p. 103–123, 2009. Disponível em: https://doi.org/10.1007/s10994-009-5119-5. Acesso em: 09 jun. 2020.

HAND, D. J.; HENLEY, W. E. Statistical classification methods in consumer credit scoring: a review. **Journal of the Royal Statistical Society: Series A (Statistics in Society)**, [*S. l.*], v. 160, n. 3, 1997. Disponível em: https://doi.org/10.1111/j.1467-985X.1997.00078.x. Acesso em: 09 jun. 2020.

HANSEN, L. K.; SALAMON, P. Neural network ensembles. **IEEE transactions on pattern analysis and machine intelligence**, [*S. l.*], v. 12, n. 10, p. 993–1001, 1990. Disponível em: https://doi.org/10.1109/34.58871. Acesso em: 09 jun. 2020.

HE, H.; ZHANG, W.; ZHANG, S. A novel ensemble method for credit scoring: Adaption of different imbalance ratios. **Expert Systems with Applications**, [*S. l.*], v. 98, p. 105–117, 2018. Disponível em: https://doi.org/10.1016/j.eswa.2018.01.012. Acesso em: 09 jun. 2020.

HO, T. K. Random decision forests. In: **Proceedings of 3rd International Conference on Document Analysis and Recognition**. [*S. n.*], 1995. v. 1, p. 278–282 vol.1. Disponível em: https://doi.org/10.1109/ICDAR.1995.598994. Acesso em: 09 jun. 2020.

HO, T. K.; BASU, M. Complexity measures of supervised classification problems. **IEEE Transactions on Pattern Analysis & Machine Intelligence**, [*S. l.*], v. 24, n. 3, p. 289–300, 2002. Disponível em: https://doi.org/10.1109/34.990132. Acesso em: 09 jun. 2020.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: a new learning scheme of feedforward neural networks. In: IEEE. **2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541)**. 2004. v. 2, p. 985–990. Disponível em: https://doi.org/10.1109/IJCNN.2004.1380068. Acesso em: 09 jun. 2020.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. **Neurocomputing**, [*S. l.*], v. 70, n. 1-3, p. 489–501, 2006. Disponível em: https://doi.org/10.1016/j.neucom.2005.12.126. Acesso em: 09 jun. 2020.

HUANG, Z.; GEDEON, T. D.; NIKRAVESH, M. Pattern trees induction: A new machine learning method. **IEEE Transactions on Fuzzy Systems**, [*S. l.*], v. 16, n. 4, p. 958–970, 2008. Disponível em: https://doi.org/10.1109/TFUZZ.2008.924348. Acesso em: 09 jun. 2020.

KO, A. H.; SABOURIN, R.; BRITTO JR, A. S. From dynamic classifier selection to dynamic ensemble selection. **Pattern Recognition**, [*S. l.*], v. 41, n. 5, p. 1718–1731, 2008. Disponível em: https://doi.org/10.1016/j.patcog.2007.10.015. Acesso em: 09 jun. 2020.

KOKOSKA, S.; NEVISON, C. Critical values for the studentized range distribution. In: **Statistical tables and formulae**. [*S. l.*: *s. n.*], 1989. p. 64–66.

KUNCHEVA, L. I. A theoretical study on six classifier fusion strategies. **IEEE Transactions on pattern analysis and machine intelligence**, [*S. l.*], v. 24, n. 2, p. 281–286, 2002. Disponível em: https://doi.org/10.1109/34.982906. Acesso em: 09 jun. 2020.

LACHENBRUCH, P. A.; GOLDSTEIN, M. Discriminant analysis. **Biometrics**, [*S. l.*], p. 69–85, 1979. Disponível em: https://doi.org/10.2307/2529937. Acesso em: 09 jun. 2020.

LEMAîTRE, G.; NOGUEIRA, F.; ARIDAS, C. K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. **Journal of Machine Learning Research**, [*S. l.*], v. 18, n. 17, p. 1–5, 2017. Disponível em: https://jmlr.org/papers/v18/16-365.html. Acesso em: 09 jun. 2020.

LESSMANN, S.; BAESENS, B.; MUES, C.; PIETSCH, S. Benchmarking classification models for software defect prediction: A proposed framework and novel findings. **IEEE Transactions on Software Engineering**, [*S. l.*], v. 34, n. 4, p. 485–496, 2008. Disponível em: https://doi.org/10.1109/TSE.2008.35. Acesso em: 09 jun. 2020.

LESSMANN, S.; BAESENS, B.; SEOW, H.-V.; THOMAS, L. C. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. **European Journal of Operational Research**, v. 247, n. 1, p. 124–136, 2015. Disponível em: https://doi.org/10.1016/j.ejor.2015.05.030. Acesso em: 09 jun. 2020.

LIAW, A.; WIENER, M. *et al.* Classification and regression by randomforest. **R news**, [*S. l.*], v. 2, n. 3, p. 18–22, 2002.

LIU, W.; CHAWLA, S. Class confidence weighted knn algorithms for imbalanced data sets. In: SPRINGER. **Pacific-Asia conference on knowledge discovery and data mining**. 2011. p. 345–356. Disponível em: https://link.springer.com/chapter/10.1007/978-3-642-20847-8_29. Acesso em: 09 jun. 2020.

LIU, X.-Y.; WU, J.; ZHOU, Z.-H. Exploratory undersampling for class-imbalance learning. **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, [*S. l.*], v. 39, n. 2, 2009. Disponível em: https://doi.org/10.1109/TSMCB.2008.2007853. Acesso em: 09 jun. 2020.

LOUZADA, F.; ARA, A.; FERNANDES, G. B. Classification methods applied to credit scoring: Systematic review and overall comparison. **Surveys in Operations Research and Management Science**, [*S. l.*], v. 21, n. 2, p. 117–134, 2016. Disponível em: https://doi.org/10.1016/j.sorms.2016.10.001. Acesso em: 09 jun. 2020.

MARQUÉS, A.; GARCÍA, V.; SÁNCHEZ, J. S. Exploring the behaviour of base classifiers in credit scoring ensembles. **Expert Systems with Applications**, [*S. l.*], v. 39, n. 11, p. 10244–10250, 2012. Disponível em: https://doi.org/10.1016/j.eswa.2012.02.092. Acesso em: 09 jun. 2020.

MARQUÉS, A. I.; GARCÍA, V.; SÁNCHEZ, J. S. On the suitability of resampling techniques for the class imbalance problem in credit scoring. **Journal of the Operational Research Society**, [*S. l.*], v. 64, n. 7, p. 1060–1070, 2013. Disponível em: https://link.springer.com/article/10.1057/jors.2012.120. Acesso em: 09 jun. 2020.

MCLACHLAN, G. J. **Discriminant analysis and statistical pattern recognition**. [*S. l.*]: John Wiley & Sons, 2004. v. 544.

MELO JR, L.; MACEDO, J. F.; NARDINI, F. M.; RENSO, C. An empirical comparison of classification algorithms for imbalanced credit scoring datasets. In: IEEE. **2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA)**. 2019. p. 747–754. Disponível em: https://doi.org/10.1109/ICMLA.2019.00133. Acesso em: 09 jun. 2020.

MELO JR, L.; NARDINI, F. M.; RENSO, C.; MACEDO, J. A. Knora-iu: Improving the dynamic selection prediction in imbalanced credit scoring problems. In: IEEE. **2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)**. 2019. p. 424–431. Disponível em: https://doi.org/10.1109/ICTAI.2019.00066. Acesso em: 09 jun. 2020.

MELO JR, L.; NARDINI, F. M.; RENSO, C.; MACEDO, J. A. On combining dynamic selection, sampling, and pool generators for credit scoring. In: IBAI-PUBLISHING. **International Conference on Machine Learning and Data Mining in Pattern Recognition**. [*S. l.*], 2019.

MELO JR, L.; NARDINI, F. M.; RENSO, C.; TRANI, R.; MACEDO, J. A. A novel approach to define the local region of dynamic selection techniques in imbalanced credit scoring

problems. **Expert Systems with Applications**, [*S. l.*], v. 152, n. 113351, 2020. Disponível em: https://doi.org/10.1016/j.eswa.2020.113351. Acesso em: 09 jun. 2020.

NEMENYI, P. Distribution-free multiple comparisons. In: INTERNATIONAL BIOMETRIC SOC 1441 I ST, NW, SUITE 700, WASHINGTON, DC 20005-2210. **Biometrics**. [*S. l.*], 1962. v. 18, p. 263.

NIKOLAOU, N.; EDAKUNNI, N.; KULL, M.; FLACH, P.; BROWN, G. Cost-sensitive boosting algorithms: Do we really need them? **Machine Learning**, [*S. l.*], v. 104, n. 2-3, p. 359–384, 2016. Disponível em: https://link.springer.com/article/10.1007/s10994-016-5572-x. Acesso em: 09 jun. 2020.

ORRIOLS-PUIG, A.; BERNADÓ-MANSILLA, E. Evolutionary rule-based systems for imbalanced data sets. **Soft Computing**, [*S. l.*], v. 13, n. 3, p. 213–225, 2009. Disponível em: https://link.springer.com/article/10.1007/s00500-008-0319-7. Acesso em: 09 jun. 2020.

PENIKAS, H. History of banking regulation as developed by the basel committee on banking supervision in 1974-2014 (brief overview). **Financial Stability Journal of the Bank of Spain**, [*S. l.*], v. 28, p. 9–48, 2015. Disponível em: https://repositorio.bde.es/handle/123456789/11433. Acesso em: 09 jun. 2020.

QUINLAN, J. R. Induction of decision trees. **Machine learning**, [*S. l.*], v. 1, n. 1, p. 81–106, 1986. Disponível em: https://link.springer.com/article/10.1007/BF00116251. Acesso em: 09 jun. 2020.

RISH, I. *et al.* An empirical study of the naive bayes classifier. In: **IJCAI 2001 workshop on empirical methods in artificial intelligence**. [*S. n.*], 2001. v. 3, n. 22, p. 41–46. Disponível em: https://www.cc.gatech.edu/~isbell/reading/papers/Rish.pdf. Acesso em: 09 jun. 2020.

ROY, A.; CRUZ, R. M.; SABOURIN, R.; CAVALCANTI, G. D. A study on combining dynamic selection and data preprocessing for imbalance learning. **Neurocomputing**, [*S. l.*], v. 286, p. 179–192, 2018. Disponível em: https://doi.org/10.1016/j.neucom.2018.01.060. Acesso em: 09 jun. 2020.

SABOURIN, M.; MITICHE, A.; THOMAS, D.; NAGY, G. Classifier combination for hand-printed digit recognition. In: IEEE. **Document Analysis and Recognition, 1993., Proceedings of the Second International Conference on**. 1993. p. 163–166. Disponível em: https://doi.org/10.1109/ICDAR.1993.395758. Acesso em: 09 jun. 2020.

SABZEVARI, H.; SOLEYMANI, M.; NOORBAKHSH, E. A comparison between statistical and data mining methods for credit scoring in case of limited available data. In: CITESEER. **Proceedings of the 3rd CRC Credit Scoring Conference**. 2007. p. 1–5. Disponível em: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.459.9556&rep=rep1&type=pdf. Acesso em: 09 jun. 2020.

SEIFFERT, C.; KHOSHGOFTAAR, T. M.; HULSE, J. V.; NAPOLITANO, A. Rusboost: Improving classification performance when training data is skewed. In: IEEE. **Pattern Recognition, 2008. ICPR 2008. 19th International Conference on**. 2008. p. 1–4. Disponível em: https://doi.org/10.1109/ICPR.2008.4761297. Acesso em: 09 jun. 2020.

SENGE, R. Machine learning methods for fuzzy pattern tree induction. Philipps-Universität Marburg Fachbereich Mathematik und Informatik, 2014.

SERRANO-CINCA, C.; GUTIÉRREZ-NIETO, B. The use of profit scoring as an alternative to credit scoring systems in peer-to-peer (p2p) lending. **Decision Support Systems**, [*S. l.*], v. 89, p. 113–122, 2016. Disponível em: https://doi.org/10.1016/j.dss.2016.06.014. Acesso em: 09 jun. 2020.

STEPHENS, T. **gplearn Welcome to gplearn's documentation!** 2016. https://gplearn.readthedocs.io/en/stable/index.html.

SUN, J.; LANG, J.; FUJITA, H.; LI, H. Imbalanced enterprise credit evaluation with dte-sbd: Decision tree ensemble based on smote and bagging with differentiated sampling rates. **Information Sciences**, [*S. l.*], v. 425, p. 76–91, 2018. Disponível em: https://doi.org/10.1016/j.ins.2017.10.017. Acesso em: 09 jun. 2020.

THOMAS, L.; CROOK, J.; EDELMAN, D. **Credit scoring and its applications**. [*S. l.*]: Siam, 2017. v. 2.

THOMAS, L.; OLIVER, R.; HAND, D. A survey of the issues in consumer credit modelling research. **Journal of the Operational Research Society**, [*S. l.*], v. 56, n. 9, p. 1006–1015, 2005. Disponível em: https://doi.org/10.1057/palgrave.jors.2602018. Acesso em: 09 jun. 2020.

VAPNIK, V. **The nature of statistical learning theory**. [*S. l.*]: Springer science & business media, 2013.

VERBRAKEN, T.; BRAVO, C.; WEBER, R.; BAESENS, B. Development and application of consumer credit scoring models using profit-based classification measures. **European Journal of Operational Research**, [*S. l.*], v. 238, n. 2, p. 505–513, 2014. Disponível em: https://doi.org/10.1016/j.ejor.2014.04.001. Acesso em: 09 jun. 2020.

WEST, D. Neural network credit scoring models. **Computers & Operations Research**, [*S. l.*], v. 27, n. 11-12, p. 1131–1152, 2000. Disponível em: https://doi.org/10.1016/S0305-0548(99)00149-5. Acesso em: 09 jun. 2020.

WHITLEY, D. A genetic algorithm tutorial. **Statistics and computing**, [*S. l.*], v. 4, n. 2, p. 65–85, 1994. Disponível em: https://link.springer.com/article/10.1007/BF00175354. Acesso em: 09 jun. 2020.

WOODS, K.; KEGELMEYER, W. P.; BOWYER, K. Combination of multiple classifiers using local accuracy estimates. **IEEE transactions on pattern analysis and machine intelligence**, [*S. l.*], v. 19, n. 4, p. 405–410, 1997. Disponível em: https://doi.org/10.1109/34.588027. Acesso em: 09 jun. 2020.

WRIGHT, R. E. Logistic regression. American Psychological Association, 1995.

XIA, Y.; LIU, C.; DA, B.; XIE, F. A novel heterogeneous ensemble credit scoring model based on bstacking approach. **Expert Systems with Applications**, [*S. l.*], v. 93, p. 182–199, 2018. Disponível em: https://doi.org/10.1016/j.eswa.2017.10.022. Acesso em: 09 jun. 2020.

XIA, Y.; LIU, C.; LI, Y.; LIU, N. A boosted decision tree approach using bayesian hyper-parameter optimization for credit scoring. **Expert Systems with Applications**, [*S. l.*], v. 78, 2017. Disponível em: https://doi.org/10.1016/j.eswa.2017.02.017. Acesso em: 09 jun. 2020.

XIAO, H.; XIAO, Z.; WANG, Y. Ensemble classification based on supervised clustering for credit scoring. **Applied Soft Computing**, [*S. l.*], v. 43, p. 73–86, 2016. Disponível em: https://doi.org/10.1016/j.asoc.2016.02.022. Acesso em: 09 jun. 2020.

XIAO, J.; XIE, L.; HE, C.; JIANG, X. Dynamic classifier ensemble model for customer classification with imbalanced class distribution. **Expert Systems with Applications**, [*S. l.*], v. 39, n. 3, 2012. Disponível em: https://doi.org/10.1016/j.eswa.2011.09.059. Acesso em: 09 jun. 2020.

ZURADA, J. M. **Introduction to artificial neural systems**. [*S. l.*]: West St. Paul, 1992. v. 8.

ATTACHMENT A - PUBLICATIONS

Journals:

– MELO JR, L.; NARDINI, F. M.; RENSO, C.; TRANI, R.; MACEDO, J. A. A novel approach to define the local region of dynamic selection techniques in imbalanced credit scoring problems. **Expert Systems with Applications**, [*S. l.*], v. 152, n. 113351, 2020. Disponível em: https://doi.org/10.1016/j.eswa.2020.113351. Acesso em: 09 jun. 2020. **[Qualis: A1]** In this paper, I present a novel procedure to define the local region of a dynamic ensemble selection classification approach. Chapter 4 contains this contribution.

Conferences:

– MELO JR, L.; MACEDO, J. F.; NARDINI, F. M.; RENSO, C. An empirical comparison of classification algorithms for imbalanced credit scoring datasets. In: IEEE. **2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA)**. 2019. p. 747–754. Disponível em: https://doi.org/10.1109/ICMLA.2019.00133. Acesso em: 09 jun. 2020. **[Qualis: B1]** This paper shows a benchmark comparison of traditional and novel classification approaches for credit scoring problem. classification approach. Chapter 3 contains this contribution.

– MELO JR, L.; NARDINI, F. M.; RENSO, C.; MACEDO, J. A. Knora-iu: Improving the dynamic selection prediction in imbalanced credit scoring problems. In: IEEE. **2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)**. 2019. p. 424–431. Disponível em: https://doi.org/10.1109/ICTAI.2019.00066. Acesso em: 09 jun. 2020. **[Qualis: B1]** In this paper, I evaluate the performance of a novel metric to define the local competence of base classifiers in a k-neraest oracles union dynamic ensemble selection technique. Chapter 5 contains this contribution.

– MELO JR, L.; NARDINI, F. M.; RENSO, C.; MACEDO, J. A. On combining dynamic selection, sampling, and pool generators for credit scoring. In: IBAI-PUBLISHING. **International Conference on Machine Learning and Data Mining in Pattern Recognition**. [*S. l.*], 2019. **[Qualis: B1]** This paper evaluates the performance of combinations of thechniques presented at Section 2.1.