NOVA IMS Information Management School

MDSAA

Master Degree Program in

Data Science and Advanced Analytics

Machine Learning Techniques to reveal abnormal behaviour in client profiles and vehicle characteristics in the non-life insurance context

Catarina Sofia Domingues Candeias

Internship Report

presented as partial requirement for obtaining the Master Degree Program in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

MACHINE LEARNING TECHNIQUES TO REVEAL ABNORMAL BEHAVIOUR IN CLIENT PROFILES AND VEHICLE CHARACTERISTICS IN THE NON-LIFE INSURANCE CONTEXT

by

Catarina Sofia Domingues Candeias

Internship report presented as partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics, with a Specialization in Business Analytics.

Supervisor: *Professor Dr. Mauro Castelli*

February 2023

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

Aveiro, 28/02/2023

In memory of my grandmother
Maria Luísa Jesus Marques Candeias,
who passed away on 26th December 2022.

I still hear her last breath.
I still remember her last words.
I still love her, as when I first met her.
I still love her, as if I had never lost her.

DEDICATION

Dear reader,

I have always wanted to write a novel.

My father is consistently advising me to save my life moments in a notebook or a diary, to keep me living those days as the wind blows, just as a breath of relief, and to publish them afterwards. My father has always encouraged me by saying: "You can make it".

My mother is the boat of my comfort zone. When in doubt, in sadness and mental torment, she has already prepared a speech that calms me down. My mother is my soulmate and would gladly say: "I am proud of you".

My brother is the one that would look at me with his ingenuity eyes and would tell me: "Have fun; you only live once". Those words let me live each day with passion and happiness, and above all, with some ingenuity that I sometimes miss.

My friends Catarina Urbano and Rita Ferreira are the ones that I left a part of me during the hard and good times. They supported me the most in this journey of knowledge, even though they were always trying to get me out of the office for a better purpose: to enjoy moments of true friendship.

My friends André Brito and Tiago Gonçalves are the ones who said several times: "You are insane for doing so many things simultaneously!" making me aware that we are not supposed to act like AI, and we should have a work-life balance. They were the ones that advised me the most and would be happy about my achievements.

My friend Francisco Lúcio is always with a busy life, from Milan to Dublin, but he would never leave me alone. He would update me with his new adventures and always make time to do a video call to see how everything was going. He is the one that would say: "Wow, Congratulations!".

My friend Filipe Boiça is the one whose feelings have been only affection and pride since our first year of studies. Consistently being present in my life moments, he would always make an effort to drive for hours to meet me wherever I was, creating more memories together and enjoying those little hours to the fullest.

I have always wanted to write a novel. It is not what my father imagined as my first, but it is my entry into this world. Academically, it was a rollercoaster of emotions; as a novelist, it would be even greater. "One day, dad, my book will be in the bookshop, and I will have the pleasure of giving you at first-hand with love and caring. However, until that day, I will still register some of my moments in the diary you offered me, to afterwards look at it, read it to my children and grandchildren and perceive how beautiful this journey was. I promise."

I have always wanted to write a novel, and I hope you enjoy this ride.

With love and gratefulness,

Catarina Candeias

ACKNOWLEDGEMENTS

I would like to express my gratitude to all individuals who crossed my academic path during these last two years, as all the knowledge acquired was mainly due to the conversations and debates taken.

A special thanks to Professor Dr. Mauro Castelli for being an available supervisor whenever a doubt arose and for his excellent guidance during this year full of uncertainties and challenges. I appreciate all his effort, shared knowledge and support.

Alongside, this project would not have been accomplished without the guidance of João Pedro Oliveira, with whom I feel utterly honoured to have worked. He gave all his time and extra more to ensure I would have the best first professional experience ever. My heart is full of gratitude.

Moreover, I want to dedicate some words to the remaining Pricing and Business Analytics team, especially to Kerem Tomasoglu, Carolina Pina, Guilherme Mousinho, João André and Manuel Viegas. I would not have conquered what I achieved if I did not have your support along the way; it was crucial, from the doubting sessions to the laughing moments in the cafeteria. I appreciate every moment we spent together and will always keep those memories alive in my heart.

Finally, thanks to all my friends, colleagues and professors of NOVA IMS. It was a pleasure to meet you and learn a new valuable topic within the scope of data science and business analytics day by day.

ABSTRACT

Anomalies are everywhere, and neither can we discard such truth in the business context. From intrusion detection for computer network systems to fraud detection and credit risk analysis, abnormalities are an unavoidable component of practically every known system. Insurance companies have registered significant growth over the last few years with the support of machine learning techniques and technological advancements. Several studies have discussed the best-unsupervised anomaly detection algorithm for each business problem and domain. Algorithms' enhancements and novel models' proposals are the most typical subject addressed. Nonetheless, fewer studies have been made regarding the identification of abnormal behaviour in client profiles and vehicle characteristics that may influence the two main measures in a non-life insurance field: the frequency and the severity. This project aims to respond to this need by experimenting with different clustering techniques, such as DBSCAN and OPTICS, and distinct unsupervised anomaly detection models, such as Isolation Forest, Extended Isolation Forest and Local Outlier Factor, on a real-world dataset provided by an insurance company that operates in Portugal. In doing so, its impact on the pricing and underwriting rules allows the attribution of an equitable tariff for the insurance entity and its customers. The implementation of the Isolation Forest algorithm for the whole dataset outperforms the remaining models by achieving an AUC score of approximately 0.86. The development of this project, besides supporting the decision-making process on identifying unsought clients in the insurance context, also contributes to broadening the knowledge of existing state-of-the-art anomaly detection algorithms and their performances.

KEYWORDS

Anomaly Detection; Unsupervised Learning; Clustering; Non-life Insurance; Isolation Forest; Area Under the Curve

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

2-D	2-dimensional
3-D	3-dimensional
AI	Artificial Intelligence
AUC	Area Under the Curve
CFS	Correlation-based Feature Selection
CRISP-DM	Cross-Industry Standard Process for Data Mining
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
EIF	Extended Isolation Forest
FPR	False Positive Rate
GDPR	General Data Protection Regulation
ID	Identification
IDE	Integrated Development Environment
iForest	Isolation Forest
IQR	Interquartile Range
KDD	Knowledge Discovery Databases
KNNimpute	Weighted K-nearest neighbours
LOF	Local Outlier Factor
ML	Machine Learning
MVI	Missing Value Imputation
OPTICS	Ordering Points To Identify the Clustering Structure
PBA	Pricing and Business Analytics
ROC	Receiver Operating Characteristic
SEMMA	Sampling, Exploring, Modifying, Modelling and Assessing
SHAP	Shapley Additive Explanations
TPR	True Positive Rate
t-SNE	t-distributed Stochastic Neighbour Embedding

e.g.	Exempli gratia ("for the sake of an example")
i.e.	Id est ("it is")

1. INTRODUCTION

Nowadays, with the rising popularity of network and information technology, human beings generate a massive amount of data exponentially (Ma, 2022), and anomalies are an inevitable component of practically every known system (Fahim & Sillitti, 2019). As early as the 19th century, identifying anomalies in data has been subject to study by the data mining community due to its high impact on distinct applications domains (Hawley & Gallagher, 1994; Nassif et al., 2021; Yepmo et al., 2022), since intrusion detection for computer network systems (Butun et al., 2014; García-Teodoro et al., 2009) to fraud detection and credit risk analysis (Abdallah et al., 2016; Hilal et al., 2022).

Anomaly detection encompasses the problem of finding the nonconforming patterns, in other words, the anomalies, concerning what is expected to be normal behaviour. Depending on the specific application domain, these so-referred anomalies are also known as outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants. Nonetheless, the terms "anomaly" and "outlier" usually are used interchangeably in the context of anomaly detection (Aggarwal, 2017; Chandola et al., 2009). For clarity, this manuscript treats both terms with the same meaning.

Several books, review articles, and research surveys have discussed the topic of anomaly detection, but the challenges associated with detecting anomalies still endure over time. These challenges are derived from several factors, as Chandola et al. (2009) state, and some of the most relevant ones are here pointed out:

- The difficulty of defining the normal region, whose instances represent every possible normal behaviour, and the lack of precision in defining the boundary between normal and abnormal behaviour can result in a misclassification data point.
- The idea of an anomaly is distinct when applied in different domains. Similar deviations can be classified as an anomaly in the medical sector, for example, but as normal behaviour in the financial industry, assessing the dependency of the studied domain.
- The developments and changes of the known normal behaviour over time are not recognised *a priori*, which can turn the decision of defining the behaviour as normal more exhaustive, complicated and less accurate for future analysis.
- The inexistence of labelled data for training and validation of models is a significant issue when applying anomaly detection techniques, as it does not incorporate *a priori* the differentiation between regular and anomalous instances (Theissler, 2017), as also the presence of noise that can be similar to the real anomalies, which turns out to be burdensome.

Therefore, with the advancements in technology and the wide range of possible applications, a reasonable number of anomaly detection algorithms have been developed and enhanced during the last decades (Xu et al., 2019).

Since machine learning (ML) and artificial intelligence (AI) emerged in the 1990s, actuarial science has undergone several transformations. Insurance firms have been evolving with Big Data Analytics, which has allowed the discovery of solutions for a customer market that nowadays is more informed and more demanding for requests and benefits. With the increase in awareness of the remaining competitors' supply through price comparison sites, there is a need to adapt to customers' expectations, retain the known customers and attract new ones (Hassani et al., 2020).

In addition, non-life insurance, particularly automotive insurance, has already been a domain of study in distinct areas, from the development of an optimal Bonus-Malus System (Frangos & Vrontos, 2001), to customer churn prediction (Spiteri & Azzopardi, 2018) and financial fraud (Hilal et al., 2022). As common acumen, vehicle owners look up automotive insurance firms for insurance to diminish costs in case of an unexpected accident. These costs cover the property, liability and medical situations. Moreover, the value charged by the company to the policyholder (the customer) registered in their insurance contract depends on several factors that can acknowledge whether the person is a desired customer for the company (Hanafy & Ming, 2021).

Many researchers have used anomaly detection algorithms to detect fraudulent claims in automobile insurance. Sometimes, the claim processing system is manipulated by the claimants and providers, frequently resulting in unnecessary financial losses for insurance companies (Chandola et al., 2009). In the literature, unsupervised auto insurance fraud detection is scarce and harder than supervised learning due to two main reasons:

(1) The inexistence of labelled targets, expensively cumbersome if executed by humans, makes assessing the model performance tougher (Domingues et al., 2018).
(2) The selection of the main variables to perform distinct unsupervised learning methods is a great challenge for the data mining community.

As a result of these difficulties, in the existing literature, there are usually more approaches to auto insurance fraud detection as a supervised learning problem rather than an unsupervised one (Nian et al., 2016).

For this reason, this document aims to reinforce the importance of unsupervised learning to perform anomaly detection problems. To the best of our knowledge, little research has been done regarding the application of anomaly detection algorithms on identifying abnormal behaviour in client profiles and vehicle characteristics in the motor insurance context and understanding if it influences claims' frequency and severity, i.e., on evaluating the level of risk a driver poses to insurance companies.

To address this study, some clustering techniques with the possibility to identify outliers and unsupervised anomaly detection models were applied. Specifically, the Density-Based Spatial Clustering of Applications with Noise model, the Ordering Points To Identify the Clustering Structure algorithm, the Isolation Forest, the Extended Isolation Forest and the Local Outlier Factor methods.

In light of the CRISP-DM methodology and the available data provided by an insurance entity, the Isolation Forest algorithm outperforms the remaining anomaly detection models, enabling the detection of isolated instances with overall good performance.

1.1. COMPANY OVERVIEW

The work presented in this report was executed within the scope of an internship at an insurance company. The company is known to be one of the most significant insurance groups, whose headquarter is in Belgium and whose entity's objective is to support distinct clients to manage, anticipate and protect them from several damages by offering them a vast collection of products developed to answer their necessities, nowadays and in the future.

In Europe, the respective insurance company is active in four distinct markets, Portugal being one of them. Clients can access insurance products through various channels, including bank branches, brokers, agents, high-profile affinity partners and directly through its brand. In Portugal, the products are from a wide range of varieties, but two main market segments are highlighted: *Life* and *Non-Life*. Focusing on *Non-Life* products, the domain carried out in this work is automotive insurance.

Given the increasing amount of data generated and the modernisation of computing capabilities, data science has been one of the most auspicious fields in almost every business area (Albrecher et al., 2019). Therefore, the insurance industry was not indifferent to the influence of data science on decision-making as it allows the identification of risk factors and the perception of behaviours (Berthelé, 2018).

Consequently, the insurance company has created within the Operations *Non-Life* department a sub-department dedicated to Pricing and Business Analytics tasks. A partial organogram is presented in Figure 1.

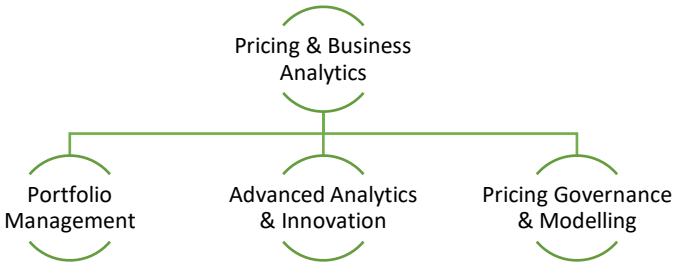


Figure 1 - Partial Organogram of the Operations *Non-Life* department

Although all the divisions are interrelated, most of the work developed was within the Advanced Analytics & Innovation team, composed chiefly of data scientists. Nonetheless, to perceive the business expertise, the remaining teams were vital to understanding the importance of distinct variables and the nomenclatures or definitions of specific insurance terms and pricing rules.

As a rule of thumb, the gathered data for the development of this project is accordingly the compliance with the European Union's General Data Protection Regulation (GDPR). For this reason, all the data presented here is anonymised for not being associated directly with any client.

1.2. Problem Definition

Since the recognition of how information technology helps distinct companies evolve, insurers were among the first to use this technological opportunity to perform several successive and segmented responsibilities. Undoubtedly, data is vital for any insurance company, as it is indeed needed for product pricing, reserving and claims management. Additionally, insurers are responsible for understanding the risk and the client's behaviour; these are essential in guaranteeing the insurance business's maintenance and competitive position in the insurance market (Berthelé, 2018).

For automotive insurance companies, the target clients are the ones whose risk of having accidents is low to refrain from covering their expenses in the event of a claim. For this reason, it is crucial to understand the factors, which can vary from clients' characteristics to vehicle attributes, that can be seen as out of the norm and perceive whether it influences claims' frequency and severity. According to the literature, one of the main approaches to investigate this problem is the implementation of anomaly detection models. In this thesis, some known algorithms were studied to address the business case at hand.

This project aims to support the decision-making on pricing and subscription rules and retain clients with *low-risk* profiles. Concerning the several applications, the main objective is to restrict specific clients' segments from entering the system, as they are inconsistent and may impact claims' frequency and severity. In this order, for example, they can be retrieved from the system or blocked at an initial stage, as they are considered anomalous observations and unsought clients.

Problem Definition:

Detecting the unusual behaviour in client profiles and vehicle characteristics. Do those instances impact the claims' frequency and severity?

Within this context, the following six research questions have been assessed:

(1) Which insights are possible to retrieve from the initial data?
(2) How to distinguish between outliers from erroneous information?
(3) How to perform the imputation of missing values to be as accurate as the original values?
(4) Which feature selection and encoding methods enhance the distinct models?
(5) How to evaluate unsupervised anomaly detection algorithms? How to compare them?
(6) Which anomaly detection method is better according to the data structure and the problem at hand?

1.2.1. Constraints and Limitations

An insurance company provided the collected data for the development of this project as part of an academic internship taken during the second year of my master's studies. Concerning the European Union's General Data Protection Regulation (GDPR) and the insurance's data protection policies, the data was anonymised, including clients' fiscal identification numbers and vehicles' license plates that could directly associate with the client and/or the driver.

Furthermore, due to some limitations of IT resources, the modelling phase was carried out using samples. Despite all efforts to reduce the data, it was computationally expensive and insanely time-consuming to perform the several anomaly detection models presented in this project with the amount of data retrieved.

Moreover, the number of benchmark datasets for this specific problem is very scarce, which results in greater difficulty in assessing the veracity of the anomaly score attribution, whereas in a supervised learning setting is immediate.

1.2.2. Thesis Structure

The present manuscript is structured into six chapters, including *Chapter 1 - Introduction,* which was already overviewed. In the next chapter, *Chapter 2 – Literature review*, a summary of the theoretical background will be addressed to provide all the basic concepts needed to support this document. In short, a brief overview of the insurance business context and its definitions, as well as an extensive theoretical summary of anomaly detection and some of its associated algorithms, will be described. Furthermore, the novel method used for imputing the missing values and the categorical encodings needed will be introduced too. Finally, to support the decisions taken during the development of this work, the *Related work* subchapter is explicitly highlighted.

Chapter 3 – Methodology discusses the CRISP-DM research framework and the tools and technologies used to proceed with the workflow. Moreover, all sub-tasks related to data processing and modelling are mentioned and analysed in detail in this chapter.

Chapter 4 – Results and discussion, as the name suggests, covers the generated outcomes and the models' performance comparisons. Nonetheless, a more in-depth analysis of the best overall model is discussed, considering the problem at hand.

The last two chapters, *Chapter 5 – Conclusion* and *Chapter 6 – Limitations and recommendations for future works*, present a reflection of the project's findings and contributions in the context of the internship, the restraints on developing this project and the possible steps to adopt to enhance the further investigation on this subject.

2. LITERATURE REVIEW

In this chapter, a description of the theoretical framework is presented to support the decisions taken throughout the project's development. For better comprehension, this chapter is divided into four sections. *Section 2.1.* introduces an outline of the insurance business, highlighting some critical concepts and terminology crucial to perceive the inherent project. *Section 2.2.* presents an extensive literature review on anomaly detection from its essential basic notions to its algorithms. Nonetheless, in this project, the imputation of missing values was a vital step in preparing the data for the modelling phase, so *Section 2.3.* is dedicated to that matter by explaining the *MissForest* algorithm. *Section 2.4.* aims to review different categorical encoding techniques used to enhance the performance of distinct anomaly detection models and convert categorical variables into numerical features. Finally, *Section 2.5.* focuses on previous work in anomaly detection in different business domains.

2.1. INSURANCE

According to the *Dictionary of Political Economy*, "the whole theory of insurance rests on the fundamental notion of risk" (Say & Chailley-Bert, 1891), emphasising risk's importance in perceiving what insurance is and why it is needed.

In common parlance, some possible risk synonyms are danger or an unfortunate event that someone was exposed to. However, in line with the insurance business acumen, this concept designates a particular way of assessing certain circumstances that can occur to values, capital owned, or even a population. Therefore, the definition of risk comes along with two branches. The first branch invokes chance, hazard, probability, eventuality or randomness; the second refers to loss or damage. When these two branches are merged, it results in the concept of *accident*. For this reason, according to the literature, every occurrence is classified as an accident, and insurance is ruled by a specific type of rationality: the calculus of probabilities (Ewold, 1991).

Once it is perceived the meaning of risk, the importance of its measurement is notable for the insurance business. For this reason, two elements are indispensable for analysing each risk exposure: the frequency of occurrence and the severity (average cost) of the losses that may occur. For a clear view of these losses, those are mainly related to (1) ownership exposures, such as repairing the destroyed property, (2) liability exposures and (3) personnel exposures, such as costs associated with death or injury (Outreville, 1998).

Consequently, with respect to the non-life insurance business, it is negotiated a legal contract between the client (policyholder) and the insurance company, known as a non-life insurance policy, in which the insurer agrees to reimburse the client for some unforeseeable losses during a specific time frame in exchange for a payment - the *premium* value (Ohlsson & Johansson, 2010).

For a long time, actuaries have been relying on traditional methods to perform pricing rules. The set of characteristics used includes the policyholder's age, address and occupation, as also the age of the policy, the claim history, the type of insurance package and, particularly concerning the motor insurance policy, the power and the type of vehicle (Albrecher et al., 2019). Nevertheless, due to the

increasing volume of information and technological advancements, actuaries are getting their hands dirty with new tools and techniques, enriching their decision-making with more insights and unseen patterns inaccessible before the boom of data analytics (Hassani et al., 2020).

2.2. ANOMALY DETECTION

2.2.1. Definitions and Terminology

During the last century and a half, the research on outliers has been one of the eldest discussions among the statistician community by composing colossal literature on the given subject. The contemporary probe has solved several age-old problems and identified new ones in outlier theory (Hawkins, 1980).

Therefore, various definitions of *anomaly detection* have emerged in distinct research papers and articles. For Chandola et al. (2009), anomaly detection reflects the problem of finding the nonconforming patterns in data with respect to what is assumed to be the expected behaviour. Another point of view is emphasised by Pang et al. (2022), who consider anomaly detection, also recognised as outlier detection or even novelty detection, the procedure of pinpointing data observations that significantly deviate from the remaining majority. These two interpretations aligned are the anomaly detection definition used throughout this thesis.

Henceforth, the concepts “outlier”, “discordant observations”, “exceptions”, “aberrations”, “surprises”, “peculiarities”, “contaminants”, “abnormality”, and “anomaly” are used interchangeably, as well as in the case of “customer”, “client” and “policyholder”. This document treats those terms with the same meaning for ease of reference.

2.2.2. Supervised, Semi-supervised and Unsupervised Learning

For any machine learning problem, one of the primary principal assessments is recognising which type of datasets we have at hand, in other words, whether we are in the presence of a supervised, semi-supervised or unsupervised learning problem. Henceforward, these three broad categories are vital to distinguish before applying any anomaly detection algorithm, as each has its own uniqueness and properties.

The category is decided based on the existence or inexistence of labels in the dataset. Figure 2 represents an illustrative representation of these three modes.

2.2.2.1. Supervised Anomaly Detection

Briefly, when encountering a supervised anomaly detection, the anomalous data is detected in accordance with the abnormality and regular labels in a training dataset. In this case, creating a predictive model for a binary classification task is the standard procedure. Any unseen observation is allocated to one of the classes according to the model developed (Chandola et al., 2009).

However, Domingues et al. (2018) state that the requirement of a labelled dataset is burdensome when human intervention is needed, which in turn, Chandola et al. (2009) underline the exigent challenge of getting accurate and representative labels for the contaminant class, even though the existence of numerous methods to generate artificial outliers. Moreover, in supervised anomaly detection, the imbalanced class distribution is highly noticeable as there is a residual number of anomalous instances compared to normal ones in the training data, which deteriorates the efficiency of the methods applied (Chandola et al., 2009; Domingues et al., 2018).

Although possessing *a priori* the designation of each instance as anomalous or normal, there is still, in practice, a tremendous difficulty in getting access to a dataset with such information filled. Furthermore, not all anomalous instances are probably recognised *per se* when developing the model because new ones may arise, which should also be identified (Yepmo et al., 2022).

2.2.2.2. Semi-supervised Anomaly Detection

Concisely, semi-supervised anomaly detection, also known as novelty detection, is a category betwixt supervised and unsupervised modes (Yepmo et al., 2022). The training data labels the normal instances, and consequently, the generated model is based only on those observations. Therefore, anomalies are detected in the test dataset that do not fit the developed model (Chandola et al., 2009).

Withal, using this mode is usually not common because it is complicated to get a training dataset that includes all kinds of possible abnormal behaviour (Chandola et al., 2009).

2.2.2.3. Unsupervised Anomaly Detection

In simplified terms, if there is no differentiation between the training and test data, and labels are absent, we are in an unsupervised anomaly detection mode. This category assumes that there are more regular observations than anomalies, which allows the algorithm to identify outliers by scoring each instance with the level of how much is considered an anomaly (Goldstein & Uchida, 2016).

As a side note, most semi-supervised algorithms can be tailored to run in an unsupervised mode by presenting a training dataset as a sample without labelled targets. Consecutively, these adjustments recognise that when developing the model, this is robust enough not to be perturbed by a few anomalies that the test dataset might contain (Chandola et al., 2009).

Moreover, as Yepmo et al. (2022) highlight, there is a slight disagreement when projecting the unsupervised algorithms into the same baskets. Some authors believe that there are five groups of unsupervised algorithms: nearest-neighbour based, clustering-based, statistical, subspace-based and classifier-based, whereas other authors defend that there are only three groups: density-based, distance-based and model-based. In Yepmo et al. (2022) perspective, it is preferable to split into three groups as some methods share the same properties or intrinsic reasonings (for example, the nearest-neighbour-based, distance-based and density-based methods rely on distances computations). For this reason, their proposal's division is into distance-based, model-based and

neural-network-based methods. The model-based method is the aggregation of clustering-based and all the semi-supervised methods.

This manuscript only focuses on unsupervised machine learning algorithms to detect abnormal behaviour.

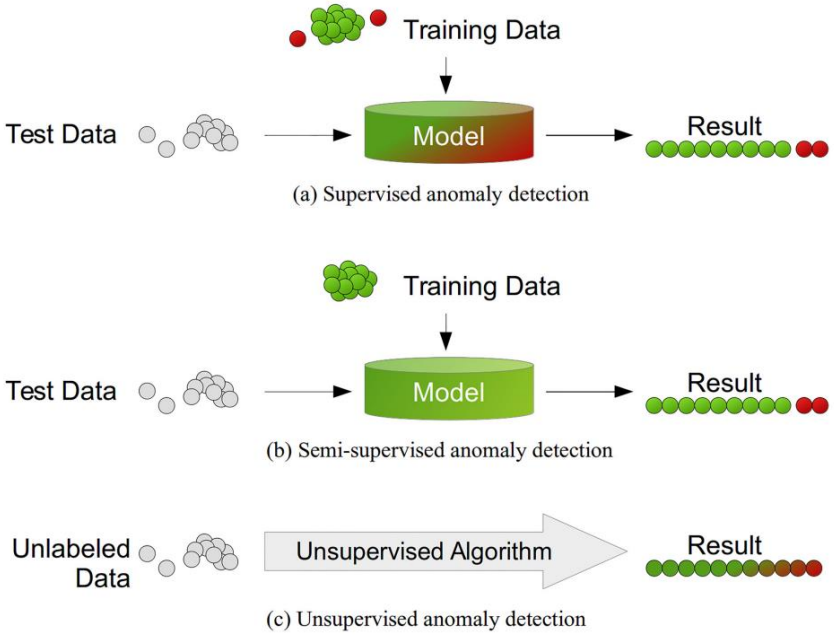


Figure 2 - Distinct anomaly detection categories[1]

2.2.3. Types of Anomalies

According to Hilal et al. (2022), for any anomaly detection technique, it is vital to distinguish each type of anomaly. Although an anomaly is always considered a rare single observation, in practice, it is not as linear as it sounds (Goldstein & Uchida, 2016). Hence, three categories are recognised by several researchers.

2.2.3.1. Point Anomalies

Most anomaly detection algorithms are designed to isolate single instances that differ significantly from the remaining data. This is suitable when referencing point anomalies (Chandola et al., 2009) as

[1] Source: Goldstein & Uchida, 2016.

demonstrated in Figure 3 through x_1, x_2, x_3 points and instances in region c_3, as these are completely far away from the rest of data.

To be more precise, instances x_1 and x_2 are considered *global anomalies* because they are very discrepant from the dense regions regarding their characteristics. On the other hand, x_3 is denominated as a *local anomaly* when compared to cluster c_2 if it is ignored all the remaining data. In this case, the comparison is based on close-by neighbourhoods. Finally, the points of cluster c_3 can be seen as three isolated data anomaly points or a small cluster. The designation of this phenomenon is the *micro cluster*. For this reason, when assigning the anomaly scores for each instance, unsupervised anomaly detection algorithms probably allocate smaller values than the undoubted anomalies but greater values than regular instances (Goldstein & Uchida, 2016).

On a daily basis, consider a bag full of marbles. For the sake of simplicity, let us assume each marble has a primary colour. The only feature being assessed is its colour. If we extract a sample and just one marble has a colour distinct from the others, it is considered an outlier, precisely, a point anomaly.

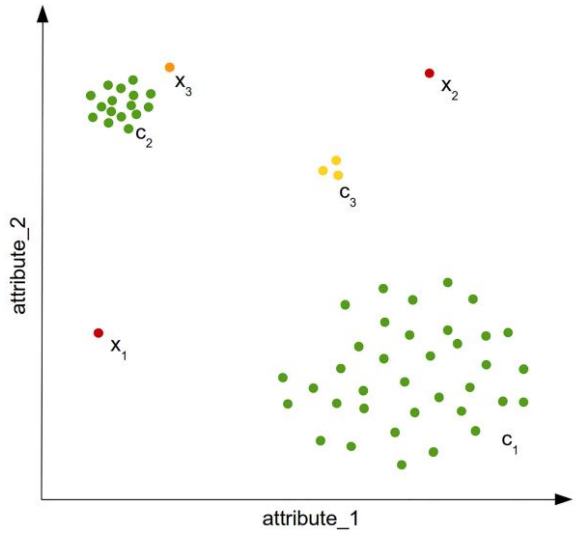


Figure 3 - Graphical illustration of three types of point anomalies[2]

2.2.3.2. Contextual Anomalies

As the name suggests, contextual anomalies, also known as conditional anomalies (Song et al., 2007), are data points that can be classified as outliers or not, depending on the current circumstance. Frequently, this type of anomaly is more presented in time-series data (Salvador & Chan, 2005) and spatial data (Shekhar et al., 2001).

In a nutshell, two attributes are crucial to understanding whether we are in the presence of a contextual anomaly: contextual attributes and behavioural attributes. The first refers to the determination of the neighbourhood or context for that data point, such as the longitude and

[2] Source: Goldstein & Uchida, 2016.

latitude of a location in spatial data (Chandola et al., 2009). The latter comprises all non-contextual characteristics of a data point, such as the average arsons of a specific location.

As a real-world example, assume that the temperature range of a particular location is between -5^{o}C and 25^{o}C. An anomaly instance could be a value of -1^{o}C during summertime, taking into account the season, even though it is considered a suitable temperature for that location during wintertime.

Having said, when employing a contextual anomaly detection technique, it is vital to possess contextual attributes in order to make sense of its application because it is not always easy to define the context. For this reason, the decision to use a contextual anomaly detection technique is specified by the relevance of the contextual anomalies in the target application field (Chandola et al., 2009).

2.2.3.3. Collective Anomalies

Briefly, when data instances are related within a dataset, it is possible to occur collective anomalies. This type of anomaly is characterised by an anomalous situation represented as a set of several instances, where each instance may not be considered as an anomaly *per se*, but their occurrence as a collection is abnormal. Typically, collective anomalies are studied in sequence, graph, and spatial data (Chandola et al., 2009).

Nowadays, intrusion detection has been one of the domains for applying anomaly detection algorithms (Jiang et al., 2006), and collective anomalies are usually present when accessing the computer with specific access patterns from a remote machine that deviates from the norm. Nonetheless, when considering only individual events, those are not seen as anomalies (Chandola et al., 2009).

2.2.4. Anomaly Detection Algorithms

This section presents all the essential theoretical backgrounds on certain anomaly detection algorithms utilised to accomplish the goal of this project. From tree-based algorithms to clustering techniques, all are explicitly described by mentioning some advantages and/or disadvantages and their implementation procedure.

2.2.4.1. Isolation Forest

Isolation Forest, also known as iForest, is a distinct outstanding type of model-based method, according to Liu et al. (2008). They proposed a model that explicitly isolates anomalies rather than profiles normal instances, as in their perspective, general anomaly detection algorithms have two pitfalls:

1) The optimisation of the anomaly detector is dedicated to profiling normal instances, but not to detect anomalies, which in turn results in worse outcomes, causing too many false alarms (regular data points are identified as abnormalities) or too few anomalies being identified.

2) Few methods can be applied for high dimensional data, as most are designated to perform only for low dimensional data since it is computationally expensive.

For this reason, this model seeks to overcome these drawbacks by handling extremely high-dimensional issues and large data sizes with a greater number of irrelevant features. Furthermore, this algorithm builds a partial model by using sub-sampled data, and it has a linear time complexity with a low memory requisite.

Succinctly, the basic assumption for all instances to be isolated is to randomly repeat its partitioning recursively, represented by a tree structure. When performing such arbitrary partitioning, the generated shorter paths are the ones where anomalies are encountered due to two main reasons:

1) Shorter paths mean a smaller number of partitions, in other words, a shorter path length from the root node to a terminating node, which only occurs when there is a low number of anomalies.
2) For each feature, instances with highly discrepant values are more likely to be split at the beginning of the partitioning.

Thus, the contaminant point is exemplified by having the shorter path delimited by the red line in the treetop and, on the contrary, the regular point whose trajectory is defined by the blue line has the tree's maximum depth, as it is possible to verify in Figure 4.

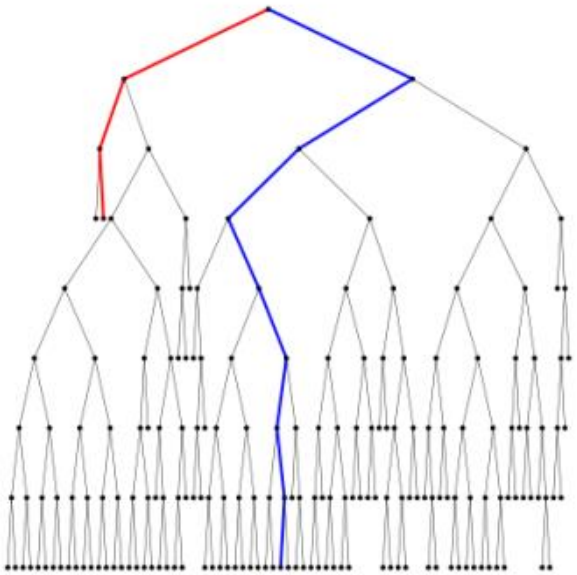


Figure 4 - Illustration of a single tree in a forest[3]

Given that each partition is arbitrarily created, distinct sets of partitions produce singular trees (known as *iTrees*). For this reason, path lengths are averaged over a number of trees to discover the estimated path length.

[3] Source: Hariri et al., 2021.

As a result, an anomaly score is generated to identify which instances are considered abnormalities by the model, which can be calculated according to Equation 1:

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \quad (1)$$

Where $E(h(x))$ is the average value of the depths an individual instance x gets in all trees and $c(n)$ is the normalizing factor as the average path length of unsuccessful search in Binary Search Tree.

According to the resulting value s, three hypotheses may arise concerning the instance classification:

(1) If $s \sim 1$, then those data points are definitely anomalies.
(2) If $s \ll 0{,}5$, then those instances are undoubtedly considered normal points.
(3) If $s \approx 0{,}5$ for all instances, then that specific sample does not present any distinct anomaly.

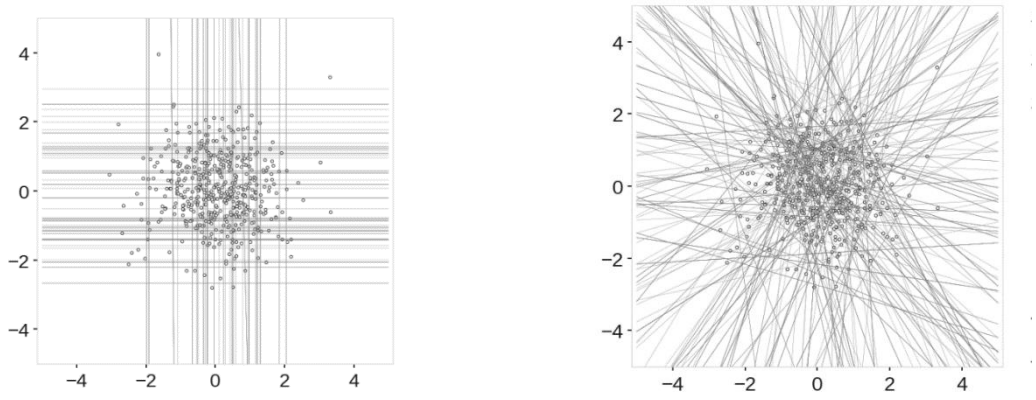
In practical terms, using the Python ML library *scikit-learn* created in 2016, the iForest algorithm can be performed. The anomaly score obtained is the opposite of the anomaly score described in the original paper, meaning that when the score is higher, we are in the presence of a regular data point and vice-versa. Regarding the other functions, all of them respect the original paper, taking only into consideration that the estimator used to grow the ensemble is the *ExtraTreeRegressor*, an extremely randomised tree regressor (Pedregosa et al., 2012).

2.2.4.2. Extended Isolation Forest

As the name suggests, the Extended Isolation Forest, EIF, is an improvement of the algorithm previously described, the iForest. According to Hariri et al. (2021), the iForest suffers from a bias as a consequence of how *iTree*'s branching comes about. In this way, the authors define that the EIF can surmount this hurdle by adjusting certain details of the original implementation, making it more widely applicable.

Straightforward, when applying the iForest algorithm, an arbitrary feature and a corresponding random value are selected. Therefore, the branch cuts are always vertical or horizontal, resulting in areas with inconsistent anomaly scores. In Figure 5a, it is perceptible that as the number of branching operations is directly related to the anomaly score, instances near (4,0) will suffer from more branch cuts rather than an instance that is near (3,3), resulting in data points with very distinct anomaly scores.

To overcome this shortcoming, EIF performs its branching operation in random directions, taking into consideration the current instances on the tree node. As such, until all instances are isolated or it is achieved the maximum depth, the branching hyperplanes can have any random slope. Using the same data distribution presented in Figure 5a, Figure 5b exemplifies the branch cuts utilised in EIF.



(a) (b)

Figure 5 - For the same data distribution, 5a represents the possible branch cuts generated by iForest, and 5b emphasises the possible branch cuts resulting from the implementation of EIF[4]

To sum up, besides being appropriated for high-dimensional datasets, EIF presents more reliable and robust anomaly scores without being computationally expensive. In practice, Hariri et al. (2021) developed an intuitive Python implementation package with the most recent version released in 2019.

2.2.4.3. Local Outlier Factor

Breunig et al. (2000) consider that most of the available anomaly detection modelling proposals classify an outlier as being a binary property, in other words, either the data instance is undoubtedly a regular data point or, indeed, it is a contaminant object. In their perspective, it is more meaningful whether it is attributed a *degree* of being an outlier to each instance rather than the traditional approaches, which led to the development of the Local Outlier Factor algorithm.

Local Outlier Factor, alias LOF, is a density-based unsupervised outlier detection variation that implements the nearest neighbour search to detect outliers in a multidimensional dataset. For each instance, an outlier factor is assigned to the local degree of being outlying, meaning that only a particular limited neighbourhood of each object is considered.

For employing this model, certain calculations need to be performed, and a specific parameter needs to be indicated. Regarding the latter, to clarify the notion of density, it is specified the value of *MinPts* that designates the minimum number of nearest neighbours to delimit the local neighbourhood of the instance. Additionally, knowing that LOF considers an instance's density and its neighbourhood's density, the principal calculation, according to Kotu & Deshpande (2019), is the relative density of a certain instance x, expressed in Equation 2.

[4] Source: Hariri et al. (2021).

$$Relative\ Distance\ of\ x = \frac{Density\ of\ x}{Average\ density\ of\ all\ k\ neighbors} \quad (2)$$

According to the resulting value of the final main equation, described in the original paper by Breunig et al. (2000), two possible scenarios can occur:

(1) LOF ≈ 1, then that instance is deep inside a cluster, so it is classified as a regular one.
(2) LOF > 1, then that instance is classified as an outlier.

To conclude, LOF outperforms the traditional methods by using the concept of local outliers, as demonstrated in Figure 6, highlighting this algorithm's importance. As it is possible to observe, even with the existence of two clusters, C_1 and C_2, with distinct density distributions, LOF can identify meaningful local outliers, as instances O_1 and O_2.

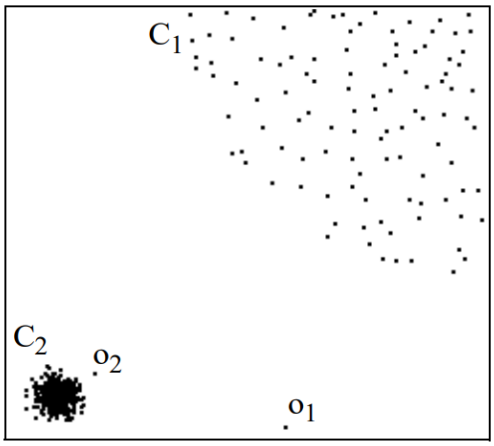


Figure 6 - Illustration of the applicability of LOF[5]

With the assistance of the Python ML library *scikit-learn*, it is possible to implement the LOF model. Even so, some remarks need to be made: (1) the metric used for computing the distance is the Euclidean distance if maintaining the default values; (2) the outlier factor obtained is negative, which means that the higher the value of the degree (closer to -1), an inlier object is identified; otherwise, it is considered an outlier (lower than -1) (Pedregosa et al., 2012).

[5] Source: Breunig et al. (2000).

2.2.4.4. Density-Based Spatial Clustering of Applications with Noise

Density-Based Spatial Clustering of Applications with Noise, alias DBSCAN, is the first density-based clustering technique capable of discovering clusters of arbitrary shape and with the ability to identify noise and outlier instances (Singh et al., 2022).

As proposed by Ester et al. (1996), this algorithm has the benefit of being efficient in a large spatial database and even supports the user in defining suitable values for the input parameters. On the other hand, Singh et al. (2022) have recently argued that the major difference between the traditional cluster-based algorithms and DBSCAN is the performance relied on detecting anomalies, which makes the DBSCAN a preferable model in comparison to others that are poorly effective.

Overall, the DBSCAN is easily understandable from a mathematical and an implementation point of view. Firstly, two key parameters are strictly necessary to initialise the algorithm:

(1) *Eps-neighbourhood*, ε: the maximum radius of the neighbourhood used to determine the region's density.
(2) *MinPts*: the minimum number of points in ε to form a cluster.

Ester et al. (1996) suggest a simple and effective heuristic to determine these key parameters of the least dense cluster in the dataset, even though DBSCAN does not change its values according to each cluster's properties.

In order to determine ε, it is constructed a sorted *k*-dist graph (*k*-distances of all objects, sorted in decreasing order, being *k*-distance(*p*) the distance from a point *p* to its *k*-nearest neighbour), where the threshold point is the one located in its first "valley", as illustrated in Figure 7. The ε will be the *k*-dist value of that threshold.

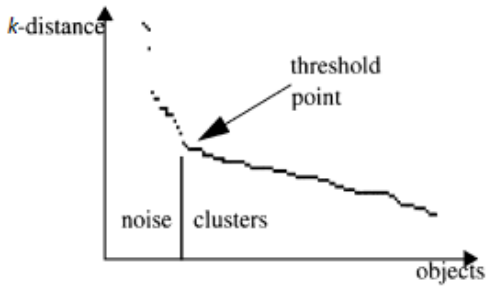


Figure 7 - Sorted *k-dist* plot for a sample database (Adapted) [6]

Regarding the *MinPts*, the originators of the DBSCAN algorithm affirm that for a 2-D dataset, a value of 4 is suitable according to their research, so the only input parameter to define is ε. On the contrary, Sander et al. (1998) assert that if the user prefers to choose its value, a possible heuristic is to define *k* as 2*dimension – 1 and then *MinPts* as equal to *k* + 1. Another possibility is offered by

[6] Source: Ester et al. (1996).

Sawant (2014), who estimates the value of *MinPts* for each distinct value of ε, as Equation 3 demonstrates.

$$MinPts = \frac{1}{n}\sum_{i=1}^{n} P_i \quad (3)$$

Where P_i is the number of instances in ε of point i.

Following the choice of each input value for DBSCAN's parameters, each instance is classified according to certain characteristics (Singh et al., 2022) and, for a better apprehension,

Figure 8 depicts its differences in an illustrative manner:

(1) *Core* (X) if there are at least *MinPts*, including the point itself, in its ε.
(2) *Border* (Y) if it has one or more *core* instances within ε but does not respect the minimum of *MinPts*.
(3) *Noise* (Z) if the instance is neither a core nor a border point and it is not accessible from any core objects.

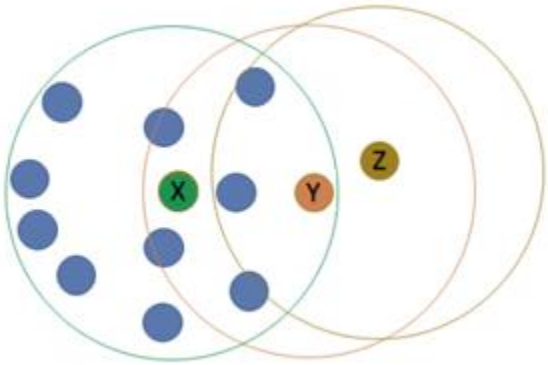


Figure 8 - Three types of instances in DBSCAN (Adapted) [7]

For this reason, it is noticeable that DBSCAN covers all the existent instances in the dataset, probably more than once, and for practical reasons, the time complexity is considerably higher. Without the use of an indexing structure, the worst-case scenario is a memory of $O(n^2)$ (Suthar et al., 2013).

For implementation, the Python ML library *scikit-learn* presents a class dedicated to clustering algorithms, including the DBSCAN. By default, the Euclidean distance is the distance metric used, and the noisy samples are labelled as -1 to distinguish them from the regular data points (Pedregosa et al., 2012). Furthermore, in this document, the approach of Sander et al. (1998) for choosing the value of *MinPts* was considered.

[7] Source: A Practical Guide to DBSCAN Method | by Amit Shreiber | Towards Data Science.

2.2.4.5. Ordering Points To Identify the Clustering Structure

Despite DBSCAN being a successful algorithm in many real-world applications (Schubert et al., 2017), it also has some drawbacks, such as the incapability to handle high-dimensional data and the inability to cluster datasets with varying densities. Moreover, as a requirement to initialise the DBSCAN algorithm, the user needs to introduce two parameters that are hard to determine and have a significant impact on the clustering result (Sawant, 2014; Smiti, 2020; Suthar et al., 2013).

To overcome the shortcomings mentioned above, especially the determination of parameters and the clustering of datasets with changeable densities, the Ordering Points To Identify the Clustering Structure, alias OPTICS, is used (Sawant, 2014). Ankerst et al. (1999) propose this new algorithm that creates an augmented ordering of the database that represents its density-based clustering structure rather than producing a clustering of a dataset explicitly.

Hence, OPTICS is an extension of the DBSCAN algorithm. The only distinction between these two algorithms is that instead of assigning cluster memberships as the DBSCAN does, it stores the order in which the data points are processed when expanding a cluster and the information regarding the core-distance and the reachability-distance for each instance (Ankerst et al., 1999).

To put it simply, Fathnia & Bayaz (2018) explain that as the DBSCAN algorithm requires a constant value of *MinPts,* consequently higher-density clusters (whose ε is smaller) are entirely within the lower-density clusters (whose ε is larger). For this reason, firstly, the object that requires the lowest ε for cluster membership must be selected. To this end, it is important to underline two distances:

(1) Core-distance: the smallest ε value that defines the object *p* as a core point. If *p* is not a core point, then the core-distance of *p* is classified as undefined.
(2) Reachability-distance: of an object *p* with respect to another *q* point is the greater value of the core-distance *q* and the Euclidean distance between *p* and *q*. If *q* is not a core point, the reachability-distance between *p* and *q* is classified as undefined.

Figure 9 demonstrates an example of these two concepts. Consider that ε = 6 and *MinPts* = 5. ε' is the distance of the core-distance of *p*, between *p* and the fourth nearest data instance. With respect to *p*, the reachability-distance of q_1 is the core-distance of *p* (i.e., ε' = 3) since this is greater than the Euclidean distance from *p* to q_1. Again, with respect to *p*, the reachability-distance of q_2 is the Euclidean distance from *p* to q_2 as this is greater than the core-distance of *p* (Fathnia & Bayaz, 2018).

Regarding the identification of outliers, as those are more distant from the objects within a cluster, they have a higher reachability-distance (Fathnia & Bayaz, 2018).

In short, the main advantage of OPTICS is that it does not stick to a single global parameter setting. As a result, it is a flexible foundation for automatic and interactive cluster analysis. Additionally, OPTICS performs well with changeable densities. Howbeit, as OPTICS works like an extended DBSCAN algorithm, the run-time is similar, and it is still infeasible to apply to databases covering several million high-dimensional points (Ankerst et al., 1999).

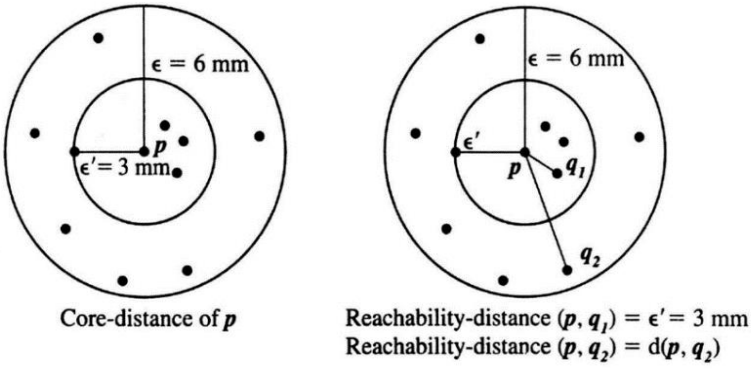


Figure 9 - OPTICS terminology[8]

In the clustering class provided by the Python ML library *scikit-learn*, it is also possible to perform the OPTICS algorithm. The distance metric and the labelling of noisy points are the same as those used in the DBSCAN implementation. Nonetheless, it is pointed out that this application differs from the original paper by first performing the *K*-nearest neighbourhood searches on all objects to recognise core sizes and then computing only the distances to unprocessed objects when making the cluster order (Pedregosa et al., 2012).

2.3. MISSFOREST

Daily, a significant amount of data is generated by humans, and it usually faces the problem of containing one or more missing attribute values. The simplest way to tackle this issue, as quoted by Strike et al. (2001), is the removal of observations if their missing values' proportion is up to 15% for the whole dataset, as it will not have a significant effect on the final result and it performs remarkably well. This method is known as *listwise deletion*. However, other approaches should be taken for a proportion higher than 15%. Missing value imputation, alias MVI, is the common solution for such cases.

MVI is the process of using a statistical or machine learning technique to replace the missing data with estimated values. According to Lin & Tsai (2020), the most widely used statistical techniques are expectation management, linear regression, least squares, and mean/mode, the latter being the simplest imputation method. On the other hand, the four main machine learning techniques applied are clustering, decision tree, *K*-nearest neighbour and random forest.

Nonetheless, most of these imputation methods are limited to one type of feature: categorical or continuous. For mixed-type data, the distinct types are usually conducted separately, discarding eventual relations between each other. Moreover, all these methods need prior knowledge about the distribution of the data, which can lead to inaccurate assumptions (Stekhoven & Buhlmann, 2012).

[8] Source: Fathnia & Bayaz (2018).

As a result, Stekhoven & Buhlmann (2012) propose a new approach, missForest, based on the random forest algorithm. In a brief explanation, the missForest starts by filling the missing values using the mean imputation or another imputation method (as the mode for categorical features). Then, it sorts the features according to the number of missing values in ascending order. For each feature, missing values are imputed by first fitting a random forest with response to non-missing values of that specific variable and the predictors (the values of the remaining variables whose index is the same). Next, for each variable, missing values are predicted by applying the trained random forest to the values of the variables whose index is also the same. This process is repeated until a stopping criterion is met or after a certain number of iterations has elapsed.

Based on the experimental study's results conducted by Stekhoven & Buhlmann (2012), even though the weighted *K*-nearest neighbours (KNNimpute) algorithm, introduced by Troyanskaya et al. (2001), is the most well-known technique for imputing continuous datasets, it made twice as much error on the categorical features compared to missForest. Hence, missForest outperformed KNNimpute by reducing imputation error in several cases by over 50%.

Thence, besides missForest being able to handle multivariate categorical and continuous data simultaneously, it can also be applied to high-dimensional datasets without the need for tuning parameters or making assumptions about the distribution of the data.

For implementation, *missingpy*[9] is a library for missing data imputation in Python with an API consistent with *scikit-learn*. One of the algorithms that the library supports is the missForest, which was applied in this thesis.

2.4. CATEGORICAL ENCODING

Commonly, tabular datasets contain categorical and numerical attributes. Nonetheless, not all models can perform well with mixed data types. For this reason, a numerical representation is required for all entries, which leads to the essentiality of developing an encoding (Cerda & Varoquaux, 2022).

According to the literature review, this section briefly discusses two encoding methods and its advantages and disadvantages.

2.4.1. One-hot Encoding

The most common approach to encoding categorical data into numerical is one-hot encoding. Basically, the original feature vector is expanded to a multidimensional matrix, so the matrix's dimension corresponds to the number of categories in this variable. Consequently, each dimension represents a specific category, which is filled with 1 and the remaining with 0 within the same feature matrix (Yu et al., 2022). Thus, the vectors created are orthogonal and equidistant (Cohen et al., 2013).

[9] Source: GitHub - epsilon-machine/missingpy: Missing Data Imputation for Python.

Figure 10 illustrates a simple example of how one-hot encoding is processed. By analysing each variable, it is possible to recognise that, in the resulting matrix, the number of dimensions of each feature matrix is equal to the number of categories within each feature in the original dataset. Thoroughly, the feature *Gender* presents three possible values: *male*, *female* and *unknown*, generating three columns in the one-hot encoded data. The process is applied to the remaining features (Yu et al., 2022).

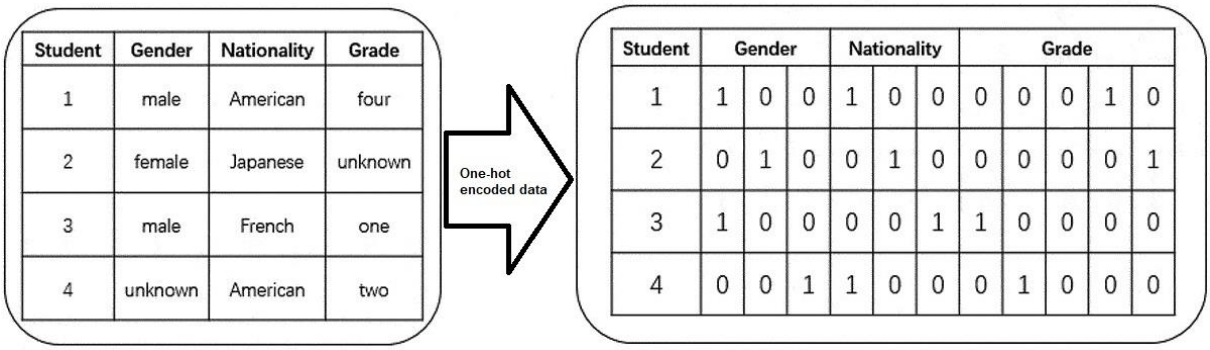


Figure 10 - Processing of one-hot encoding (Adapted)[10]

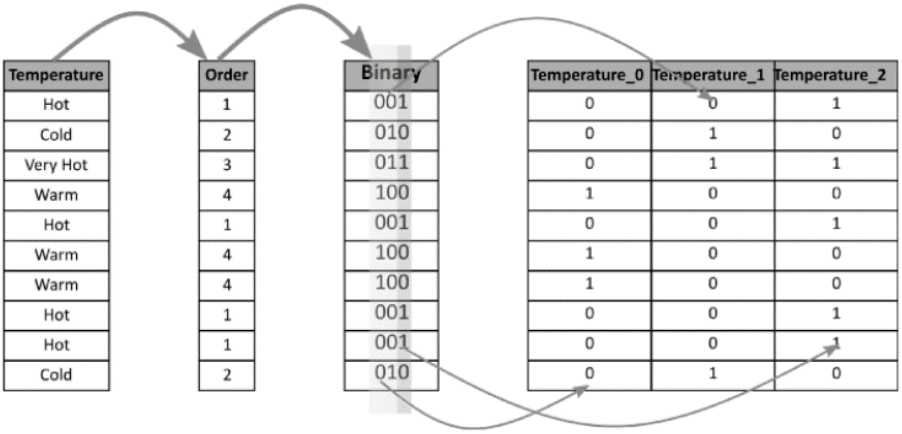
The main advantage of using one-hot encoding is its simple application, which makes it a widely used encoding method, despite more advanced coding structures being available (Davis, 2021). Nonetheless, this encoding technique brings a significant disadvantage for high-cardinality categories: it originates feature vectors of high-dimensionality, leading to *curse of dimensionality* and computational and statistical problems (Cerda & Varoquaux, 2022).

2.4.2. Binary Encoding

As Potdar et al. (2017) describe, in binary encoding, the categories are first encoded as ordinal for a particular categorical feature, and then those integers are transformed into binary code. Subsequently, the digits resulting from that binary string are divided into separate columns. Having that said, for a feature with n-unique values, this results in a $\log_2(n)$-number of on or off discrete values (Seger, 2018).

As an illustrative example, consider the categorical feature *Temperature* in Figure 11. In order to encode the categorical feature through binary encoding, firstly it was encoded the categories as ordinal, resulting in 4 values, and then those integers were converted into binary code (*001*; *010*; *011*; *100*). Next, each digit was split into separate columns, obtaining, in this case, three columns.

[10] Source: (Yu et al., 2022).



Temperature	Order	Binary	Temperature_0	Temperature_1	Temperature_2
Hot	1	001	0	0	1
Cold	2	010	0	1	0
Very Hot	3	011	0	1	1
Warm	4	100	1	0	0
Hot	1	001	0	0	1
Warm	4	100	1	0	0
Warm	4	100	1	0	0
Hot	1	001	0	0	1
Hot	1	001	0	0	1
Cold	2	010	0	1	0

Figure 11 - Processing of binary encoding (Adapted)[11]

2.5. RELATED WORK

Several authors have applied anomaly detection algorithms in various domains, from credit card fraud to network intrusion detection. However, to the best of our knowledge, fewer studies have been done regarding the detection of abnormal behaviour in client profiles and vehicle characteristics in an insurance context. For this reason, this review gives an overview of the current anomaly detection state-of-the-art and describes the overall research related to anomaly detection in distinct domains.

Firstly, referring to the views of Domingues et al. (2018), anomaly detection methods can be split into different groups. More specifically, the groups are divided into probabilistic, distance-based, density-based, neighbour-based, information theory, neural networks, domain-based, and isolation methods. When comparing some of these techniques, the iForest algorithm was the standout model by being efficient and high-scalable on high-dimensionality data and less computationally demanding.

On the other hand, Sun et al. (2016) compared the performance of iForest with iForest with categorical data for detecting anomalous user behaviour in the network systems to prevent malicious attacks. The research demonstrated that although iForest is an adequate model, it showed promising outcomes with a real-world dataset when including categorical attributes.

Nevertheless, a theoretical comparison between the EIF and IF was made by Panja et al. (2022) to discover uncommon occurrences of conventional behaviour in IoT systems. When applying the EIF method, its model's accuracy score was 93% and F1-Score was 96%. Furthermore, due to its easy interpretability, the authors retrieved the top 12 features for anomaly detection.

[11] Source: All about Categorical Variable Encoding | by Baijayanta Roy | Towards Data Science

Nowadays, unsupervised learning is also widely used for abnormal data identification, for instance, for medical insurance transactions. Zhang et al. (2020) applied several models, such as DBSCAN, iForest and LOF, and the best detection rate was obtained using the iForest algorithm. Therefore, it was possible to conclude that the ability of iForest to deal with irrelevant attributes adds value to the model's performance and highlights its relevance in this area.

Despite iForest being a recognisable unsupervised method, clustering algorithms such as DBSCAN can also be employed efficiently to detect anomalies (C. Zhang et al., 2020). In fact, as Alhussein & Ali (2020) point out, DBSCAN is suitable for detecting anomalies defined as very low-density data compared to natural data. One application of this algorithm concerns the early identification of unknown events within routine flight data due to safety precautions and to prevent some possible incidents. As a result, the DBSCAN worked very well when isolating samples with significant distinct characteristics. Another application focuses on credit card fraud transactions. Panigrahi et al. (2009) state that the noises, in this case, are considered credit card fraud that do not belong to any cluster. In this research, only a proportion of variables was used, despite the authors' suggestion to include other attributes as well.

Regarding the OPTICS algorithm, which is a novel clustering technique, there is little research on its applicability. Nonetheless, N. et al. (2022) applied this method to defend against selective forwarding attacks in wireless sensor networks. The outcomes proved that OPTICS models obtained a mean accuracy of 100% detection rate and 0% of missed detection rate.

Moreover, the LOF method is also considered a possible anomaly detection algorithm that Tahir et al. (2019) implemented for investigating anomalous changes in network user behaviours. Having said that, LOF had excellent results by obtaining a 100% true positive rate (TPR), but only at the expense of a 2% false positive rate (FPR). Nonetheless, for another experimental setting, it achieved a 47.2% TPR with a 1.1% FPR. On the contrary, Fan et al. (2021) studied the accuracy and stability of LOF and iForest in Diabetes and Shuttle datasets from the UCI ML Repository. The accuracy obtained through the ROC curves indicated that the iForest had higher stability and accuracy. As the authors point out, the iForest is more indicated when applied to industries and systems that need to process a large amount of data, which is not the case of the LOF algorithm as it is relatively complex.

Finally, Xiaoyun & Danyue (2010) developed a hybrid outlier mining algorithm in the insurance context. The main purpose of its study was to evaluate the perception of client moral risk that could cause many problems, such as insurance fraud, high loss ratio and adverse selection. To this end, anomaly detection was implemented to understand if the instance was a normal or abnormal claim. Among all the ingested attributes, the authors emphasise the policyholder's age, marital status, job grade, insurance coverage, insurance premium value and former credit record that could be related to their future behaviours, and so it could indicate the possible moral risk of the insured person. This research motivates the use of certain variables for detecting anomalies in this project.

To sum up, our contribution is to spread the applicability of anomaly detection algorithms in the motor insurance context by reinforcing the impact of abnormal client profiles and vehicle characteristics in claims' frequency and severity. As mentioned previously, studies on this concrete topic are very scarce, so this project aims to promote further discussion on this subject. Although most research papers depict different application domains, the outcomes can be adapted towards anomaly detection in the auto insurance context and are helpful for further work.

3. METHODOLOGY

In this chapter, the description of the research methodology used for developing this case study is presented. The following four sections aim to explain in detail distinct subject matters. *Section 3.1.* contains information about the CRISP-DM methodology, which was the framework used for this project. *Section 3.2.* states the tools and technologies available and used to proceed with the analysis. *Section 3.3.* overviews all the data processing tasks to perform the modelling phase discussed in *Section 3.4.*

3.1. RESEARCH FRAMEWORK

In 2000, a non-proprietary and freely available standard process model was released to service the data mining community. This model was intended to be industry-, tool-, and application-neutral. Cross-Industry Standard Process for Data Mining, alias CRISP-DM, is the model announced with such purpose (Chapman et al., 2000).

In depth, the CRISP-DM methodology is described in terms of a hierarchical process model comprising four levels of abstraction (from general to specific): *phases*, *generic tasks*, *specialised tasks* and *process instances*. The reference model, despite not being possible to identify all relationships, gives an overview of the life cycle of a data mining project. It consists of six iterative phases, from business understanding to deployment, shown in Figure 12 (Schröer et al., 2021).

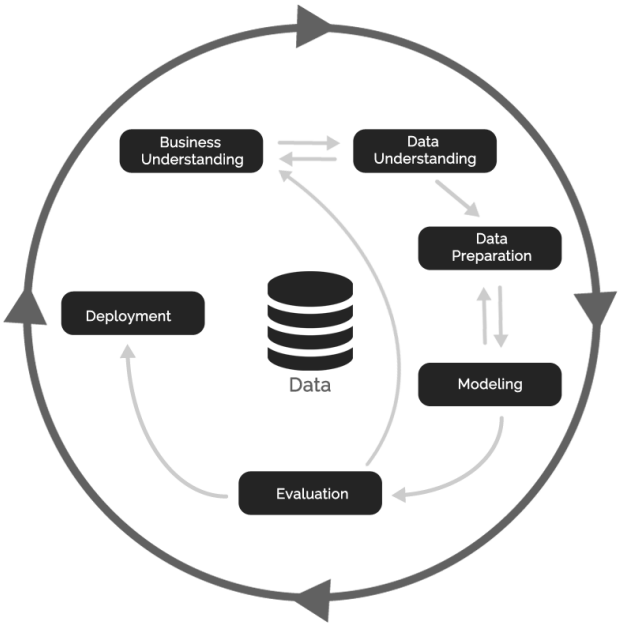


Figure 12 - CRISP-DM reference model phases (Adapted)[12]

[12] Source: CRISP-DM and what I did wrong | by Gabriela Moreira Mafra | Medium

As it is possible to assess, the sequence of the phases is not strict. In fact, the most important and frequent dependencies between phases are shown through the displayed arrows. Depending on the project, it is required to backtrack to previous phases and repeat certain tasks according to the outcome of each phase or task. Furthermore, the outer circle in Figure 12 represents the cyclic nature of data mining itself. All the knowledge obtained during the process and even the deployed outcome can generate new and more focused business questions. Briefly, it is a continuous learning (Wirth & Hipp, 2000).

This document follows the CRISP-DM methodology, although the existence of other processes/methodologies to support the progression of data mining projects, such as the Knowledge Discovery Databases process, alias KDD, and Sampling, Exploring, Modifying, Modelling, and Assessing methodology, alias SEMMA. However, according to Shafique & Qaiser (2014), regardless of researchers' and data mining experts' beliefs that KDD is more accurate, most companies and industries use CRISP-DM and SEMMA, concluding that CRISP-DM is more detailed and therefore completer than the latter.

In short, the six phases of CRISP-DM are described succinctly by Chapman et al. (2000).

(1) Business Understanding: This initial phase uncovers and focuses on perceiving the project objectives and requirements from a business perspective, converting this knowledge into a data mining problem definition and a preliminary plan to accomplish the objectives outlined.
(2) Data Understanding: This second phase begins with an initial data collection and focuses on activities that allow us to get familiar with data, identify data quality problems and explore first insights into the data, and/or detect interesting subsets to form hypotheses for hidden information.
(3) Data Preparation: This third phase reports all activities required to build the final dataset, whose tasks are likely to be performed several times without a fixed order, e.g., transformation and cleaning of the data.
(4) Modelling: This fourth phase covers the selection and application of various modelling techniques and the calibration of their parameters to optimal values. Nonetheless, specific techniques need a particular form of data, which leads back to the previous phase.
(5) Evaluation: This fifth phase encloses the model (or models) that seemed more accurate from a data analysis perspective by evaluating and reviewing every step executed to create it. The final outcome should be the decision to use the data mining results achieved.
(6) Deployment: This sixth phase, and last, usually is the customer that carries out the deployment steps, as it can be simply the creation of a report or the implementation of a repeatable data mining process across the organisation. For this reason, the data mining project does not end with the creation of the model; it goes beyond it.

To sum up, CRISP-DM is the *de facto* standard and an industry-independent process model for applying data mining projects (Schröer et al., 2021), which was the guideline for conducting this work. Hence, *Chapter 1 – Introduction* provides all the information regarding the first phase of CRISP-DM. *Chapter 3 – Methodology* covers the CRISP-DM's second, third and fourth phases, and *Chapter 4 – Results and discussion* introduces the evaluation phase and the most suitable final model for this

project. As a side note, the deployment phase was not reached at the time of writing, even though it was being prepared. For this reason, it is not going to be assessed in this manuscript.

3.2. TOOLS AND TECHNOLOGIES

For any data science project, tools and technologies are necessary to develop the data analysis and achieve the stipulated goals. In this thesis, Python was the primary tool used for all the data exploration and preparation, alongside SAS® Enterprise Guide, as it is the main tool used by the entity and where the data was stored. Additionally, the DataRobot was mainly used for modelling, as it is the AI Cloud leader with a vision to accelerate the delivery of AI to production for every organisation.

According to Raschka & Mirjalili (2019), Python is one of the most popular programming languages for data science and therefore contains several useful add-on libraries developed by its great community. The inventor of Python was Guido Van Rossum, who describes Python as being an easy-to-learn, powerful programming language since it has efficient high-level data structures and a simple but effective approach to object-oriented programming. Furthermore, the creator defends that Python has elegant syntax and dynamic typing, making it a suitable language for scripting and fast application development in diverse fields on most platforms (van Rossum, 1995). Moreover, Python is free software that can be used with GNU (GNU/Linux), Unix, Microsoft Windows and many other systems (van Rossum & Drake, 2009).

The use of Python has reached unprecedented levels, especially around freely available tools and libraries, mainly due to its reliance via *NumPy* and *SciPy* wrappers on the fast implementations of a large number of scientific algorithms. Nonetheless, the best and most used Python library in data preparation nowadays is *pandas* since it has powerful querying possibilities, statistic calculations and basic visualisations. Regarding data visualisation, *Plotly* supports most of the standard plots used in data mining and machine learning, despite the existence of *seaborn* and *Matplotlib*, the latter having fewer capabilities. With respect to machine learning, the most popular library used is *scikit-learn*, as it contains a significant number of algorithms to implement (Stančin & Jovic, 2019).

All analyses with Python are run in Jupyter Notebook within the Anaconda open-source toolkit. Jupyter Notebook is an integrated development environment (IDE) that operates as a web-based interactive environment and enables users to edit, run and present code. Through Anaconda, all the necessary software packages are retrieved and managed, as it handles multiple data environments that can be maintained and run separately without interference from each other (Persson & Khojasteh, 2021).

Regarding SAS® Enterprise Guide, it is the newest point-and-click interface from SAS that replaced the Analyst interface, providing easier access to many SAS statistical analyses without the need to learn how to write the SAS code underlying its procedures (Meyers et al., 2009). Moreover, in the era of big data, SAS® Enterprise Guide allows the user to extract deep insights from the huge volume of data and its fundamental patterns (Parr-Rud, 2014). The data repository consisted of a set of SAS tables, as it is the principal tool that the entity employs.

Concerning the DataRobot platform, as its website[13] highlights, it is well-known to be the AI Cloud leader, enabling entities to leverage the transformational power of AI, which is trusted globally by several customers across industries, including a third of the Fortune 50. The organisation gave the possibility of using this platform to support the modelling phase.

Overall, several tools, technologies and a specific IDE were used to execute all analyses and developments of this project.

3.3. DATA PROCESSING

In this subchapter, an overview of data understanding and preparation phases is described. Furthermore, the following sub-subchapters are dedicated to understanding the information retrieved from the initial analysis, the difference between noise and outliers and the imputation of missing values. Moreover, all the feature engineering and selection are studied in detail.

3.3.1. Data Understanding

This sub-subchapter explains in an overview the data collection, exploration and quality of the initial dataset to get preliminary insights into the data.

3.3.1.1. Data Collection

In summary, the project aims to detect unusual behaviour from client profiles and vehicle characteristics that may impact claims' frequency and severity. As this project was developed within an insurance company, the data extracted was obtained through the main tool used by the entity: SAS® Enterprise Guide. The data repository consisted of 15 SAS data tables, stored in 4 distinct libraries managed by the Pricing & Business Analytics team. This investigation was applied to the automobile segment in three distinct entity's brands, in which the data corresponded to clients' characteristics, vehicles' attributes and the main features of insurance, such as frequency, severity and all characteristics related to policy coverages and products.

In order to respect the compliance of the European Union's GDPR, the data presented here is anonymised to not to be associated directly with any client. For this reason, pseudo-unique identifiers were created for specific variables, such as the Taxpayer Identification Number and the license plate attributes.

The raw data comprised 301 features and approximately 77 million instances, which is a high volume of data; hence, data processing was a vital and hugely time-consuming task. The data collected corresponded to the period between 1st January 2016 and 31st December 2021, as a 5-year historical data could bring more insights into the analysis.

One particularity about this dataset is the fact that the information is updated monthly, which evinces the existence of a high percentage of duplicates (when for the same client, there is not any monthly modification and still a row is introduced), but also that a new instance is always generated if only one attribute value is different from the previous record within the same client and month,

[13] Source: About DataRobot - DataRobot AI Platform

maintaining everything else constant. This means that N variations in one variable bring N extra rows for the same client. The sub-subchapter *3.3.2 - Data Preparation* will address this situation in detail.

On the other hand, this project only considered the particular clients and the known drivers, as motor insurance can also be applied to an enterprise customer, and hence it does not provide any information regarding the driver. Furthermore, it is essential to underline that although the policyholder is always deemed the client, the driver does not necessarily imply the same reasoning since the policy can be registered on behalf of a person different than the driver; for example, a parent oversees the vehicle insurance policy even though the son drives the respective vehicle.

3.3.1.2. Data Exploration

To better perceive the structure of the dataset, an initial exploratory data analysis was carried out. As mentioned previously, the raw dataset was huge, and some prior knowledge was needed to begin this project. Thus, as Agrawal et al. (2015) defend, data visualisation is very useful for understanding data in a graphical manner, but the challenges associated with the amount of data generated daily turn big data visualisation into a more demanding task. Nonetheless, it was still possible to deploy some visual graphics to get preliminary data insights.

Firstly, some clients' characteristics were assessed to perceive the client portfolio within the auto insurance segment. As stated before, the driver and the policyholder may not be the same person, so most of the analysis was based on the driver's characteristics, whose risk profile is evaluated when getting an auto insurance policy. In Figure 13, the pie chart represents the proportion of the policyholder being the driver (*1*) versus the policyholder not being the driver (*0*). Hence, it is possible to infer that approximately 77% of clients are policyholders that drive the vehicle insured. According to the data, most of the remaining percentage corresponds to individuals with insurance policies associated with their parents or spouses.

Regarding our driver portfolio, 69.13% are male, 73.45% are married, and 50% were born between 1960 and 1980. Moreover, the geographic predominance of clients corresponds to urban areas (53.28%), making Lisbon the most prevalent one (21.45%), followed by Porto (19.31%) and Aveiro (11.28%).

Concerning the vehicle portfolio, 51.06% corresponds to passenger vehicles, more precisely cars, and the major brand insured is Renault (12.32%), followed by Opel (9.3%), Peugeot (8.29%) and Volkswagen (8.27%). Most vehicles have a diesel engine (56.46%) or a gasoline engine (42.92%), and the vehicle's production year ranges mostly between 1995 and 2010.

From March 2020 to August 2021, due to public health conditions (Covid-19), a quarantine period affected several corporate fields. This situation encouraged some people to cancel their active motor policies. Figure 14 represents that proportion (32%) by comparing the policy cancellation between such period and the remaining one. If the policy's cancellation occurred between March 2020 and August 2021, *IND_COVID* takes the value 1; otherwise, it takes 0.

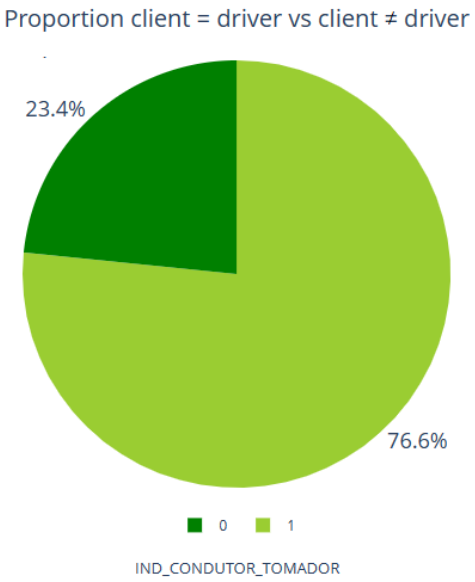


Figure 13 - Proportion between policyholder = driver (1) versus policyholder ≠ driver (0)

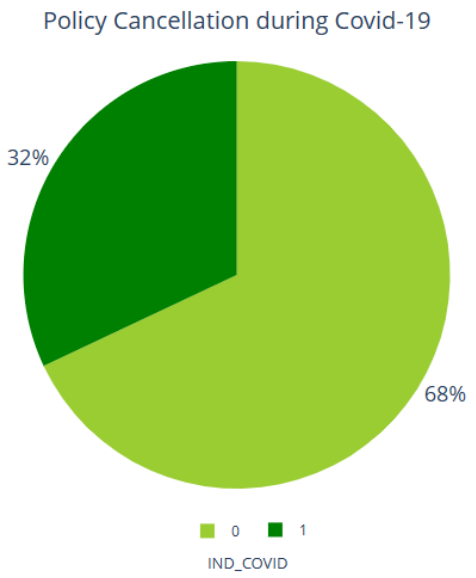


Figure 14 - Policy cancellation during Covid-19

Overall, 75% of active policies encompass 4, 5 or 6 insurance coverages, three of them being obligatory in Portugal by law[14] (vehicle damage, injury costs and damage to personal property), also known as mandatory third-party insurance. This emphasises the extra add-ons requested by the policyholders, varying from fire damage to theft protection or even fully protected against all risks.

Frequency and severity are two main variables that identify the number of claims on the policy's exposure and its associated average cost, respectively. Some interesting findings were retrieved from the initial data that are shown in Figures 15 and 16.

Figure 15 highlights the comparisons between different drivers' genres and marital statuses with severity. First of all, when comparing the different marital statuses, widowers have a higher severity, especially when a woman (*Driver's Genre = 1*) drives the vehicle. Nonetheless, when married, divorced or single, the difference between genres is hardly noticeable with respect to severity; however, single men drivers have a slightly higher severity than single women drivers.

In turn, Figure 16 emphasises the distinct insurance coverage packages and the associated average frequency. As it is possible to assess, mandatory third-party insurance has the lowest average frequency, and collision or rollover insurance has the highest value of average frequency. To conclude, getting extra add-on insurance coverage increases the likelihood of a claim occurring.

[14] Source: Seguro Automóvel | Autoridade de Supervisão de Seguros e Fundos de Pensões

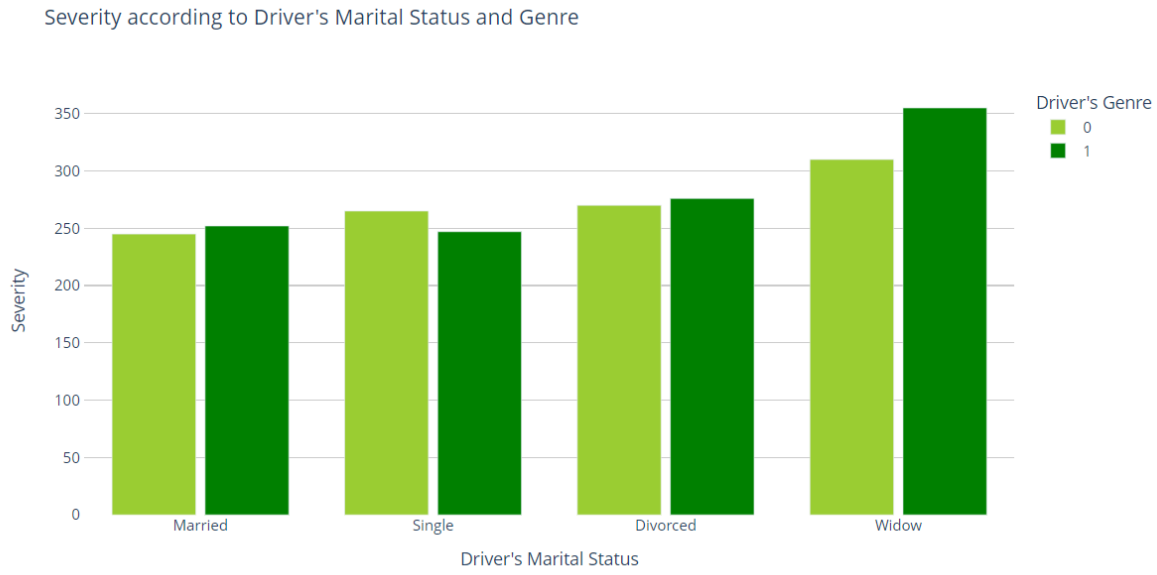


Figure 15 - Severity according to driver's marital status and genre

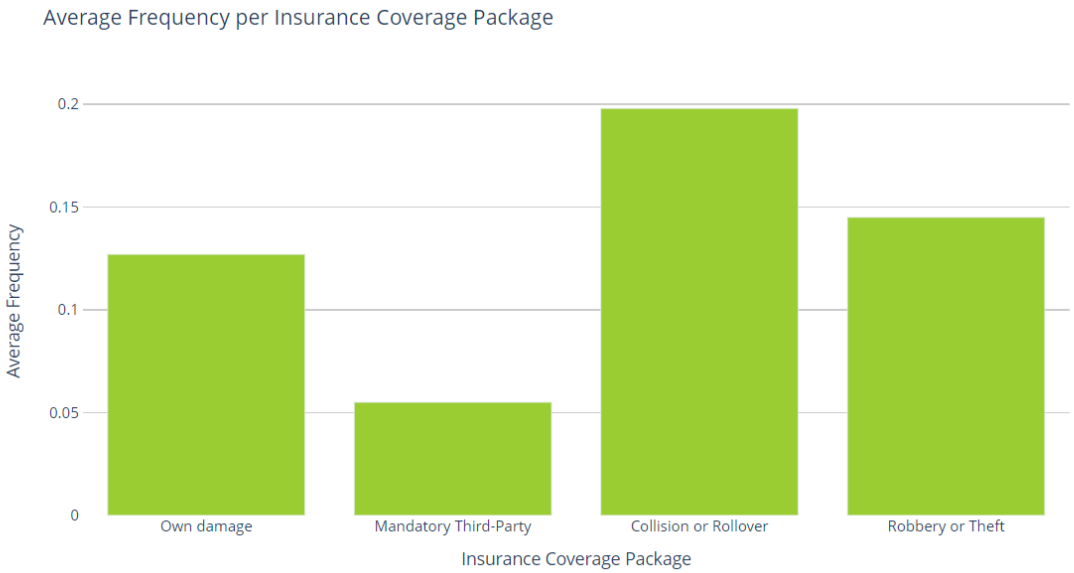


Figure 16 - Average frequency per insurance coverage package

3.3.1.3. Data Quality

Since the obtained data had a significant volume of instances and features and knowing that most of the information is inserted manually by the insurance agents, it is imperative to distinguish between noise and outliers, as Smiti (2020) defends. According to the author, while noises are useless and must be eliminated, outliers can give both useless and interesting information. Cases of noisy data can be attributed to incorrect data type, erroneous data values (i.e., instead of "88" for the *age* feature, it was introduced "888") and missing values. On the other hand, the presence of outliers in

data can be assigned to measurement or recording error, exceptional but true values, misreporting or even sampling error. Yepmo et al. (2022) highlight this difference in the banking domain, where fraudulent credit card transactions are considered anomalies because they were not performed by the card owner. Nonetheless, in this specific case, outliers can be detected when the cardholder makes a very high one-time payment compared to their habits, and noise can be related to a negative value for an amount, due to an error in the system, for example.

With that said, it was performed an extensive examination of the noisy data alongside business experts and documentation support to correct most of those cases. The policyholder's age was one of the features in such circumstances, with values of "9999", as also the vehicle valuation attribute when presenting values below zero.

Regarding missing data, the percentage was higher than expected, as more than 50% of variables presented at least 40% of missing values. Nonetheless, in some numerical features, the missing values corresponded to zero, as in the case of the feature related to the number of claims processed.

Moreover, some categorical features, for example related to the brand or the model of the vehicle, presented inconsistencies due to the manual insertion by insurance agents, making it difficult to extract some insights from it. As a result, more than one value was generated for the same characteristic. For instance, "Renault" and "renault" are considered two different categories, even though it is respected to the same brand.

Due to all the aforementioned reasons, an enormous amount of time was spent in the data preparation phase, which will be described in the following subchapter.

3.3.2. Data Preparation

Data Preparation is a fundamental phase of data analysis, and it generates a dataset smaller than the original one, enhancing the efficiency of data mining (S. Zhang et al., 2003). This subchapter englobes all the main tasks within this phase. Data selection, cleaning, construction, integration and formatting tasks are described here before proceeding to the modelling phase.

3.3.2.1. Data Selection

The raw dataset consisted of 301 features and approximately 77 million instances, comprehending 5-year historical data. To perform clustering techniques, such as DBSCAN, it is important to diminish the size of the dataset; otherwise, it may not work effectively and efficiently in high-dimensional space due to the phenomena of *curse of dimensionality* (Hinneburg & Keim, 1999).

The initial selection of variables was based on business knowledge and the information necessary to achieve the data mining goal. Having said that, the dataset was reduced from 301 features to 200 variables.

Thereafter, according to Nassif et al. (2021), Correlation-based Feature Selection (CFS) and Principal Component Analysis are the most used feature selection techniques in anomaly detection problems. For this reason, being CFS the simplest and the most straightforward feature selection technique to apply in any dataset, it eliminated 94 numerical features and 20 categorical features. Nonetheless,

each variable was assessed according to its relevance to the business goal before it was indeed discarded. With that said, redundant features were screened out as they were highly correlated with one or more of the remaining features (Hall, 1999). For numerical features, if Spearman's correlation coefficient was higher than 0.85, the numerical attribute would be removed. On the other hand, for categorical features, if Cramer's V correlation coefficient was higher than 0.30, the feature was dropped. The choice of these two correlation coefficients and the measure of the strength of the relationship was based on the study done by Akoglu (2018). Succinctly, the use of Spearman's correlation coefficient was due to the existence of outliers, and the choice of Cramer's V correlation coefficient was due to the large dimension of the dataset.

Additionally, when the driver or the policyholder was undefined and it was not possible to cross information between these two individuals (i.e., when they were different people), or when the company owned the policies as a fund to claims costs that were not assigned to the appropriate policy, or when the vehicles had erroneous license plate making harder its identification, the removal of all those cases was the chosen step as they were useless. This filter eliminated 7.4% of the dataset.

3.3.2.2. Data Cleaning

One of the main goals of the Data Preparation phase is to generate quality data (S. Zhang et al., 2003). As this project aims to detect abnormal behaviour, the removal of outliers was not performed. Nonetheless, the correction of erroneous data, the imputation of missing values and the text pre-processing of high-cardinality categorical features were crucial to continue developing this project.

Before all else, as emphasised in *3.3.1.3. – Data Quality*, erroneous information and outliers are different concepts. The treatment taken to correct erroneous data was to evaluate the discrepant values that could directly state the inconsistency of the feature, i.e. typos inserted manually by insurance agents (e.g., the value registered in the *age* feature should have been allocated in the *genre* feature) or system errors (e.g., according to the value of *age* feature [assuming 18], a binary value is inserted in *retirement* attribute [incorrectly the system declares 1, meaning the individual is elder]). With that said, the data was assessed intensively with documentation and business experts' support to correct most of those cases. When the values were identified as erroneous, but it was not possible to infer the actual value, a missing value would replace the erroneous value.

Regarding the imputation of missing values, four procedures were adopted according to their percentages and their reasons for the missingness. Straightforward, (1) 22 variables with more than 40% of missingness were removed, knowing that 17 features had a proportion higher than 65%; (2) with auxiliaries SAS data tables, it was possible to reduce the number of missing values of variables associated with the driver and policyholder's birthday year, genre and marital status, and of variables related to vehicle's number of doors, gross weight and valuation; (3) for some numerical features, the missingness was the default option, which was replaced by zero, because it concerns to the inexistence of claims, costs or vehicle additional accessories and equipment; (4) finally, for variables with less than 25% of missing values, it was applied the missForest algorithm. The missForest algorithm allows the imputation of categorical and numerical data simultaneously without the need to know *a priori* the data distribution or to tune parameters. Moreover, missForest works well in high-dimensional data and can parallelise the process of doing such imputations. Additionally, it

outperformed KNNimpute by diminishing imputation error in several cases by over 50% (Stekhoven & Buhlmann, 2012). All these reasons reinforce the use of this method to impute the remaining missing values, which was performed in four distinct groups to reduce the time complexity according to (1) client features, (2) geographical characteristics, (3) insurance variables and (4) vehicle attributes.

Lastly, categorical features introduced manually by the insurance agent can result in high cardinality and many inconsistencies. The vehicle brand and the vehicle model features had more than 500 and 3000 distinct unique values, respectively. When carefully analysing these two attributes, it was evident that the same brand or model had different nominations, originating distinct categories, e.g., "fOrd" and "Ford". To handle this situation, researchers follow a typical procedure of text pre-processing steps as Denny & Spirling (2018) describe, which was the baseline adopted here by applying two of them: (1) *punctuation*: it was discarded all the nonletter characters, such as "&" or "-", because were considered uninformative; (2) *lowercasing*: it was converted all the words into lowercase, as it does not make any sense to have two separate categories when the two values have the same meaning. This decision reduced approximately 40% of categories in both variables.

3.3.2.3. Data Construction

When preparing the data, generating new features may improve models' efficiency and performance, adding more insights into the analysis. As the purpose of this project is to detect abnormal behaviour in client profiles and vehicle characteristics, some features were derived from existing ones, such as the driver's birthday year, license driver's year, license plate year and the policy's starting year.

3.3.2.4. Data Integration

The dataset collected was composed of approximately 77 million instances with several records concerning the same person because of how the information is introduced and updated monthly by the technical team, as explained in *3.3.1.1. - Data Collection.*

In order to condensate the information and by being aware that the driver's risk profile is the mainly evaluated criterion when getting an auto insurance policy, a primary key was created by associating the driver's identification (ID) with the license plate's ID. The reasons behind this decision are: (1) one individual can drive multiple vehicles, and gathering the information without the identification of the vehicle could lead to misleading insights as several attributes are respected to it; (2) the existence of duplicates would bias the identification of abnormal data and decrease the performance of models, and (3) for modelling processing, the aggregation of information was the most reasonable decision to take; otherwise, it would be computationally expensive and extremely time-consuming.

Having said, on the one hand, it is obtained the 5-year historical data with accumulative values in specific attributes, such as the number of total claims, the total premium value and the accumulated claims' costs; on the other hand, it is acquired the most updated information related to, for example,

the number of years without any registered claims, the introduction of extra vehicle accessories or equipment and the quantity of insurance coverages.

This way, the dataset was reduced by 98%, containing all the 5-year historical information of each driver and vehicle.

3.3.2.5. Data Formatting

Ultimately, the resulting data set comprised 1 514 779 instances and 55 variables (20 metric and 35 non-metric features, being 24 binary variables). Before the application of unsupervised anomaly detection algorithms, some final steps were performed. For this project, normalisation and categorical encoding were two procedures taken to satisfy specific modelling requirements.

Normalisation

When different attributes have distinct measurement units, normalising the data is very important, especially when the selected approaches directly use distances between instances. A study by Campos et al. (2016) demonstrates that anomaly detection techniques on normalised data perform better than on un-normalised data.

Typically, the most used normalisation technique in anomaly detection problems is *Min-Max* scaling, but it can also be contra-productive, as Goldstein (2014) highlights. According to the author, for a dataset containing categorical and numerical attributes, the *Mixed Euclidean Distance* is advised to be computed, diminishing the influence of categorical distances in relation to numerical distances.

Nonetheless, the effect of four distinct normalisation methods for outlier detection was assessed by Kandanaarachchi et al. (2020). The conclusion retrieved from this study reinforces the dependency on the direction where the outlier is placed. When the outlier is in a direction other than the highest range, *Min-Max* normalisation is the preferred technique, even though it is influenced by outliers; on the other hand, *Median-IQR* is more robust to anomalies and, for a high dimensional dataset, it may give better results, since outliers may occur in directions of high variation. However, for the models applied in this project, the study emphasises that, on average, iForest does not have a preferred normalisation method and LOF performs slightly better when *Min-Max* normalisation is applied.

Another research was conducted by Ramsauer et al. (2021), where it was applied the DBSCAN algorithm for outlier detection in driveability data. For this model, the most relevant results were obtained with *Min-Max* normalisation, as the output of standardisation (*Z-Score* normalisation) and *Median-IQR* were unsatisfactory.

For all these reasons, it was performed the *Min-Max* normalisation available in Python *preprocessing* library *scikit-learn*, which corresponds to Equation 4.

$$Min\_Max\colon x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{4}$$

Where x_{min} and x_{max} are the range of each column x. All features will be transformed into the range [0,1].

Categorical Encoding

Another essential aspect before ingesting the data into different models is the conversion of categorical attributes into numerical features. Although most outlier detection models use homogeneous datasets with only one unique type of attributes, i.e., numerical or categorical features, when working with real-world datasets, the probability of being composed of mixed data types is very high (K. Zhang & Jin, 2010).

One-hot encoding is widely used for converting categorical features into numerical attributes; nonetheless, the data becomes sparse after applying this transformation because it increases the number of variables and, in turn, the dimensionality of the data (Ul Haq et al., 2019).

On the other hand, binary encoding is a substitute for one-hot encoding, especially when there is a high number of unique categories, i.e., if a particular feature contains n unique categories, one-hot encoding will generate n or $n-1$ features, whereas binary encoding will produce only $\log_2(n)$ features. So, suppose one categorical feature presents 100 unique values. In that case, the one-hot encoding needs at least 99 variables, whereas the binary encoding needs only 7 features, which is a substantial reduction in dimensionality and a significant advantage over one-hot encoding.

As Seger (2018) states, an interesting approach to consider is mixing different encoding techniques. For very high cardinality attributes, it might be suitable to use a compressed representation, but applying one-hot encoding might make more sense for the remaining features. For this reason, binary encoding was applied for variables with high cardinality; otherwise, one-hot encoding was used.

3.4. MODELLING

As mentioned previously, the objective of this project is to detect abnormal behavior in client profiles and vehicle characteristics. Hence, to proceed with the application of unsupervised anomaly detection algorithms, the data was divided into two segments, bearing in mind that some attributes concerning the insurance policy conditions are necessary to consider in both parts. Nonetheless, except for clustering algorithms (DBSCAN and OPTICS), due to the problem of *curse of dimensionality* and high computational costs, it was used all the pre-processed data to perform outlier detection, which corresponds to 65 attributes. Furthermore, due to IT resource limitations, the modelling phase was carried out using a sample of 10% in Jupyter Notebook and in the DataRobot platform for fair comparisons.

3.4.1. Hyperparameters Tuning

For unsupervised ML methods, setting the hyperparameters is challenging, even though it can significantly affect models' performance. Additionally, measuring performance requests to possess labelled data, which is not pretended to be available in an unsupervised mode (Soenen et al., 2021). In the literature, default hyperparameters are usually the most common choice since its tunning is hardly achievable, given the unavailability of labels to optimise on and its associated high

computational costs (Ahmed & Courville, 2020; Datta et al., 2020; Domingues et al., 2018). Nevertheless, to fairly compare the models, the *contamination* level was defined *a priori* to 0.1 (10%). This hyperparameter concerns the expected proportion of outliers in the dataset and is utilised to delimit a threshold on the decision function.

Still, for the DBSCAN algorithm, as described in *Chapter 2 – Literature review*, the choice of *MinPts* value was based on the heuristic provided by Sander et al. (1998), which corresponds to *k + 1*, being *k* equal to *2*dimension – 1*. The ε value was calculated according to Ester et al. (1996) method that first computes a sorted *k*-dist plot, and then it finds the *k*-dist value of the threshold point located in its first "valley" that corresponds to the most suitable value for ε. Additionally, for OPTICS, the value of *MinPts* was based on the same heuristic as for the DBSCAN. Finally, for EIF, according to Hariri et al. (2021), the *ExtensionLevel* hyperparameter represents the extension level to be used in creating the splitting criteria, which depends on the number of features, and so it is equal to the *number of dimensions - 1*.

Table 1 presents the default hyperparameters for each anomaly detection model used in this experiment. Table 2 focuses only on the parameters for DBSCAN, OPTICS and EIF algorithms for each data segment defined by data-driven decisions. The sorted *k*-dist plot exhibited in Figure 17 supports the choice of ε value for DBSCAN for both segments since the obtained ε value was equal.

Table 1 - Default hyperparameters values for each anomaly detection method.

Model	Hyperparameter	Values
IF	n_estimators	100
	max_samples	'auto'
	max_features	1.0
	bootstrap	False
EIF	ntrees	100
	sample_size	256
LOF	n_neighbors	20
	algorithm	'auto'
	metric	'euclidean'
DBSCAN	metric	'euclidean'
	algorithm	'auto'
OPTICS	max_eps	np.inf
	metric	'euclidean'
	cluster_method	'xi'
	algorithm	'auto'

For the avoidance of doubt, the default hyperparameter *cluster_method* in OPTICS defined as 'xi' corresponds to the automatic technique to generate the clustering model proposed in Ankerst et al.

(1999) research paper. For a more detailed description, it is highly recommended to execute a close reading of the respective paper research, as it is not going to be reviewed in this document. Moreover, in the same model, the default hyperparameter *max_eps* corresponds to the identification of clusters across all scales, which in turn increases the time complexity to $O(n^2)$.

Table 2 - Hyperparameters defined according to data-driven decisions.

Model	Hyperparameter	Client Data	Vehicle Data	All Data
DBSCAN	eps	0.13	0.13	-
	min_samples	26	32	-
OPTICS	min_samples	26	32	-
EIF	ExtensionLevel	39	37	65

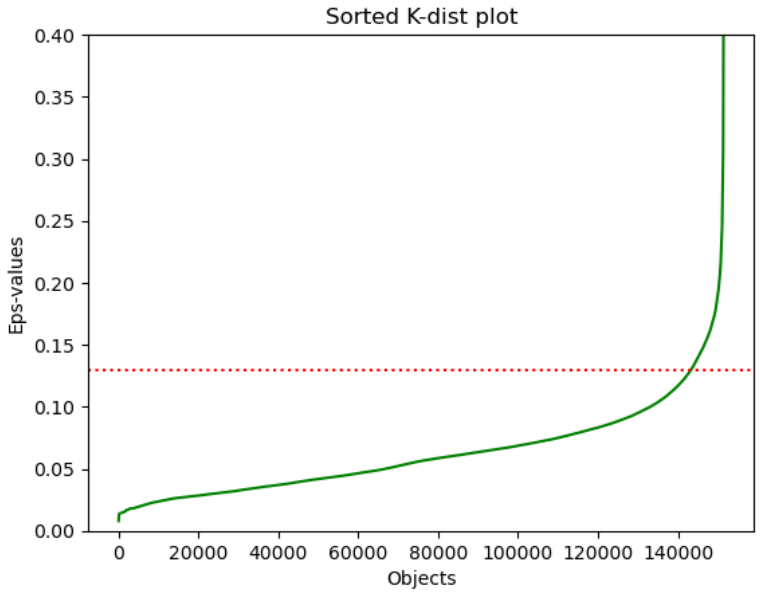


Figure 17 - Sorted *k-dist* plot for determining ε for both segments

3.4.2. Evaluation Metrics

Regarding the evaluation of unsupervised anomaly detection algorithms, it is not as straightforward as in the supervised mode (Goldstein & Uchida, 2016), and indeed it is a constant challenge in data mining research (Campos et al., 2016). Unfortunately, without a labelled dataset, having access to criteria able to discriminate between unsupervised algorithms is extremely difficult and very limited in the proposed literature (Goix, 2016; Goldstein, 2014). To that end, the most popular evaluation strategy on unsupervised outlier detection corresponds to ranking the results according to the anomaly score and then iteratively defining a threshold from the first to the last rank (Campos et al., 2016; Goldstein & Uchida, 2016). This approach is based on the *Receiver Operating Characteristic*

(ROC) curve, whose computation acts in accordance with the following procedure: (1) the true positive rate (TPR) corresponds to the ratio of observations correctly identified as anomalies to all outlying observations; (2) the false positive rate (FPR) coincides with the ratio of observations incorrectly designated to the anomalous class compared to the total number of normal observations; (3) both ratios are plotted against each other, obtaining the ROC curve, where a random outlier ranking produces a curve close to the diagonal, whereas a perfect algorithm generates a curve making up of a vertical line at FPR equal to 0 and a horizontal line at TPR equal to 1 (Goldstein, 2014). Figure 18 demonstrates these phenomena for a 2-D artificial dataset, where a perfect algorithm (green ROC curve) is applied and where a random guessing algorithm is created (red ROC curve), being LOF an almost perfect algorithm (blue ROC curve).

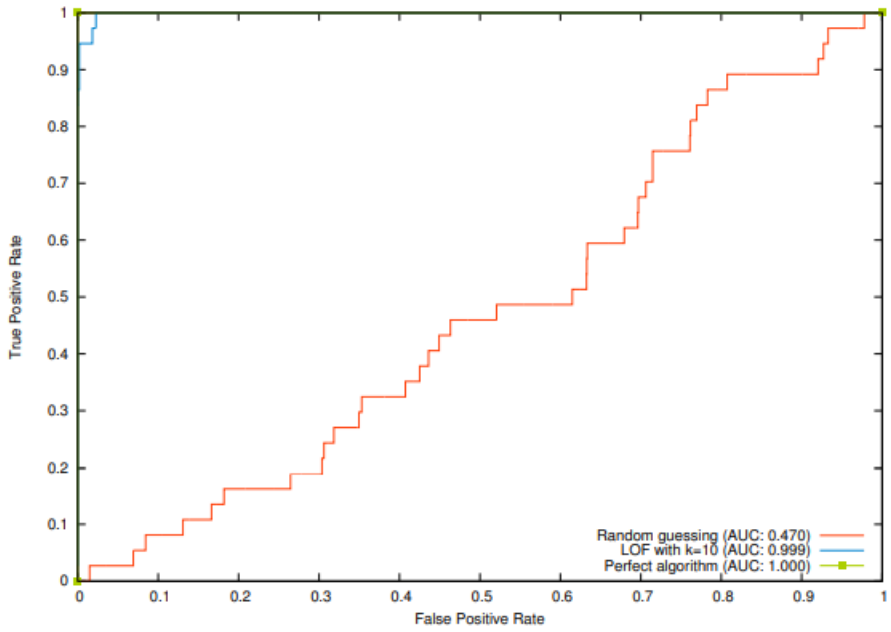


Figure 18 - Demonstration of distinct ROC curves according to the type of algorithm[15]

Another key fact to remember is related to the difference between the ROC curve calculation in a classification problem and in an unsupervised mode, the latter showing the ranking quality of the algorithm (Goldstein, 2014). Additionally, the ROC curve presents a significant advantage by naturally adjusting according to the imbalance of class sizes, usually represented on outlier detection tasks (Campos et al., 2016).

Conversely, the ROC curve can be summarised by a unitary value known as the *Area Under the Curve* (AUC), which corresponds to the integral of the ROC curve. Hence, AUC is also used as a detection performance quality measure, whose interpretation follows the proof from Fawcett (2006) and Hanley & McNeil (1982), which corresponds to the likelihood of allocating a lower score to a randomly chosen normal observation than a randomly chosen outlying observation, in an anomaly detection context. In other words, a perfect ranking would result in an AUC value equal to 1, whereas

[15] Source: Goldstein, 2014.

an imperfect ranking would produce an AUC value near 0. If the ranking is randomly produced, AUC would be close to 0.5 (Campos et al., 2016; Goldstein, 2014). For all these reasons, AUC is considered a great quality measure and is used for comparing the performance of the distinct unsupervised detection models applied in this project.

For transparency, the DataRobot platform assesses the performance of anomaly detection models based on a synthetic AUC metric. It is generated two synthetic datasets from the validation sample; one made more normal and another more anomalous[16]. Nonetheless, the resulting AUC score is based on the same proof mentioned previously.

Finally, for non-hierarchical clustering techniques, such as DBSCAN and OPTICS, that enable the detection of abnormalities when grouping similar observations, some possible evaluation measures to assess the clusters' quality are the *R-Squared* (Halkidi et al., 2002) and the *overall average silhouette score width* (Rousseeuw, 1987).

By referring to the views of Halkidi et al. (2002), the *R-Squared* value is used to understand if there is a significant difference among instances in distinct groups and whether observations within the same cluster have high similarity. The values of this measure range between 0 and 1; the minimum value indicates no differentiation among groups, and the maximum value suggests a significant dissimilarity. Equation 5 represents the mathematical definition of the *R-Squared* value.

$$R\_Squared = \frac{SS_t - SS_w}{SS_t} \quad (5)$$

Where SS_t is the total sum of squares of the whole dataset and SS_w is the sum of squares within the same cluster.

Concerning the *overall average silhouette score width* measure, it provides an evaluation of clustering validity for the entire plot. As Rousseeuw (1987) explains, this measure represents the average of how well all instances match the clustering at hand (i.e., how well they have been classified) in the whole dataset, whose values range between -1 and 1. A value near -1 indicates that the overall classification of observations was wrongly made, whereas a value near 0 means that there is not any discrepancy between clusters (as the data points might belong to some other clusters) and a value near 1 suggests that the overall classification of instances was correctly made. Succinctly, the larger the *overall average silhouette score width*, the better the discrimination of clusters.

As a final note, despite the objective of this project is not to obtain a perfect clustering result, it is important to perceive if the applied clustering techniques are the most suitable for the data at hand. For this reason, both previously mentioned measures are calculated, and if appropriated, those models are considered for the outlier detection problem.

[16] Source: Anomaly detection: DataRobot docs

3.4.3. Model Explainability

When it comes to complex ML models, especially for large modern datasets, it is more difficult to perceive which input features drive the prediction outcome. In a corporate context, interpreting and explaining what happens within the distinct models is crucial to make them more transparent and trustworthy for presenting to an audience and captivating the stakeholders' attention.

To address this problem, Lundberg et al. (2017) developed a unified framework known as Shapley Additive Explanations (SHAP). Essentially, the SHAP values work as a cohesive measure of feature importance that results from the game theory's principles. In fact, the SHAP values allow local and global interpretability; however, in this project, the latter is the only case to be addressed. Succinctly, the global explainability transmits which features are important to the model overall.

In the anomaly detection context, it might be helpful to comprehend which attributes impact the anomaly score. To this end, the target here is the prediction, i.e., if the instance is detected as an outlier or a regular one. For instance, Antwarg et al. (2021) demonstrated the effective use of SHAP values to explain irregularities detected by an autoencoder, which is an unsupervised model.

All things considered, Lundberg et al. (2017) admitted that SHAP is better aligned with human intuition than previous techniques, emphasising its higher impact on interpreting the model's output.

Through the DataRobot platform, the SHAP values are automatically computed for each feature by getting a sample average of the SHAP absolute values.[17]

[17] Source: SHAP reference - DataRobot AI platform

4. RESULTS AND DISCUSSION

In this chapter, the outcomes of distinct anomaly detection models and their performances are assessed. As mentioned in *Section 3.4. - Modelling*, for clustering techniques, the data was only segmented into client profiles and vehicle characteristics. In contrast, for the remaining unsupervised anomaly detection algorithms, it was used all the pre-processed data, as well as the segmented data. For this reason, this chapter will be divided into three main groups: the client segment, the vehicle segment and the whole dataset segment. Next, according to the performance of each model with respect to its evaluation metric result, the best overall method will be in-depth analysed. Note that in all cases is only used 10% of the dataset.

4.1. CLIENT SEGMENT

For the client segment, 13 attributes were introduced into the modelling task. Table 3 presents the *R-Squared* and *overall average silhouette score width* for clustering techniques, and Table 4 exhibits the obtained AUC scores for the remaining unsupervised anomaly detection models. In bold, the highest AUC is highlighted, representing the best model.

Table 3 - Evaluation performance metrics for clustering techniques and the number of clusters.

Model	# Clusters	R-Squared	Silhouette Score
DBSCAN	13	0.14	- 0.51
OPTICS	58	0.83	- 0.76

In light of Table 3, the clustering techniques have poor performance according to the *overall average silhouette score width* with values below zero, meaning that most of the observations' classification was wrongly made. Additionally, the *R-Squared* value for the DBSCAN algorithm reinforces its poor performance. Nonetheless, despite the higher *R-Squared* value in the OPTICS model, it is crucial to address the overall spectrum and the discrimination between clusters. For this reason, it is assumed that grouping the data to detect anomalies is not the most suitable approach with the algorithms tested, even though the presence of a good value in the *R-Squared* score for the OPTICS model. *Appendix A – Client Segment* presents the distribution of each feature for each cluster with the support of box plots for both clustering techniques.

Table 4 - AUC scores for EIF, iForest and LOF algorithms.

Model	AUC
EIF	0.7759
iForest	**0.7906**
LOF	0.6743

As demonstrated in Table 4, the obtained scores induce that the best model to apply in this segment is the iForest, with an AUC score of approximately 0.79. Since the difference between EIF and iForest scores is not significantly discrepant, it was created a forest visualisation to provide an instant view of how on average anomalous observations reach much smaller depths than the considered normal observations. Figure 19 exhibits the forest visualisation for iForest and EIF, where each radial line is associated with a tree. The grey circle corresponds to the depth limit each tree can reach, whereas the yellow lines represent the depth each normal observation reached on each tree, and the green lines show the reached tree depth for each anomalous observation. Having said that, it is possible to perceive that there is a tendency for normal observations to reach the maximum depth of the tree in EIF rather than in iForest. Regarding anomalous observations, on average, the depths in iForest seem much smaller than in EIF.

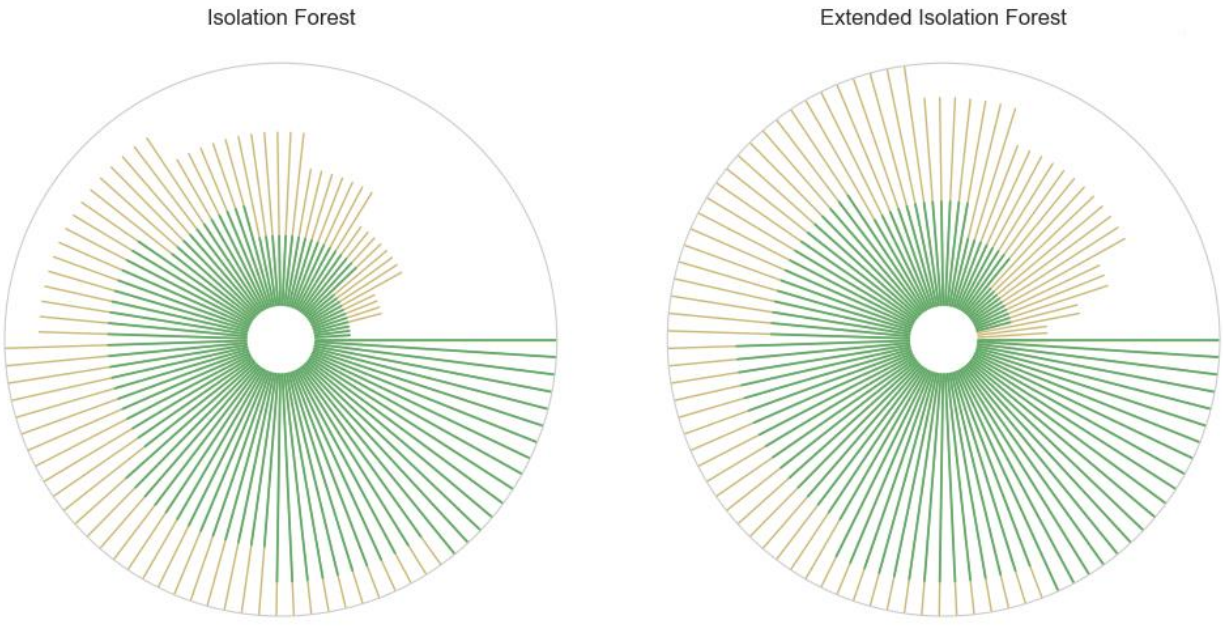


Figure 19 - Forest visualisation of iForest and EIF for the client segment

Concerning the best model, iForest, it is important to understand which features are needed to take model decisions. To this end, SHAP values are measured through the DataRobot platform. According to Figure 20, the top five features that have the highest impact on the model output, i.e., on the anomaly score, are related to the driver's civil status, the coverage package each policyholder holds, and the number of claims reported.

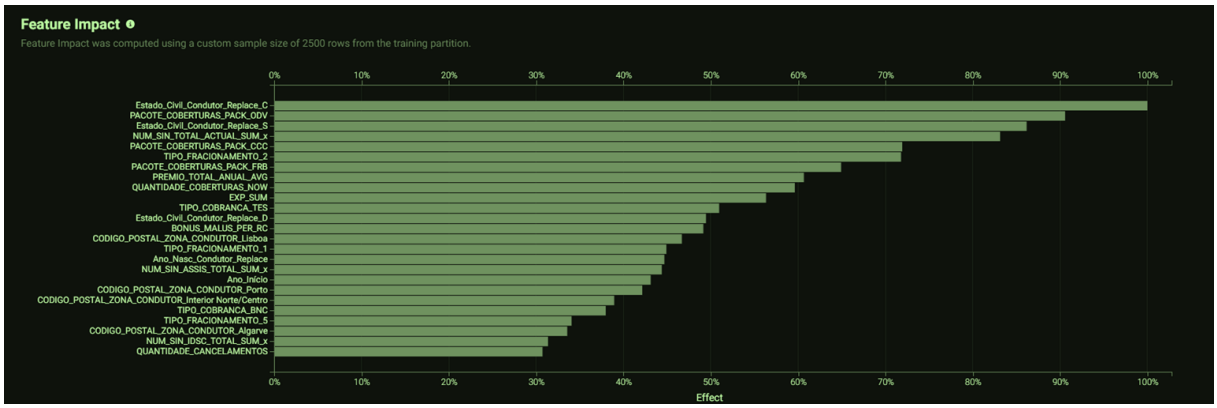
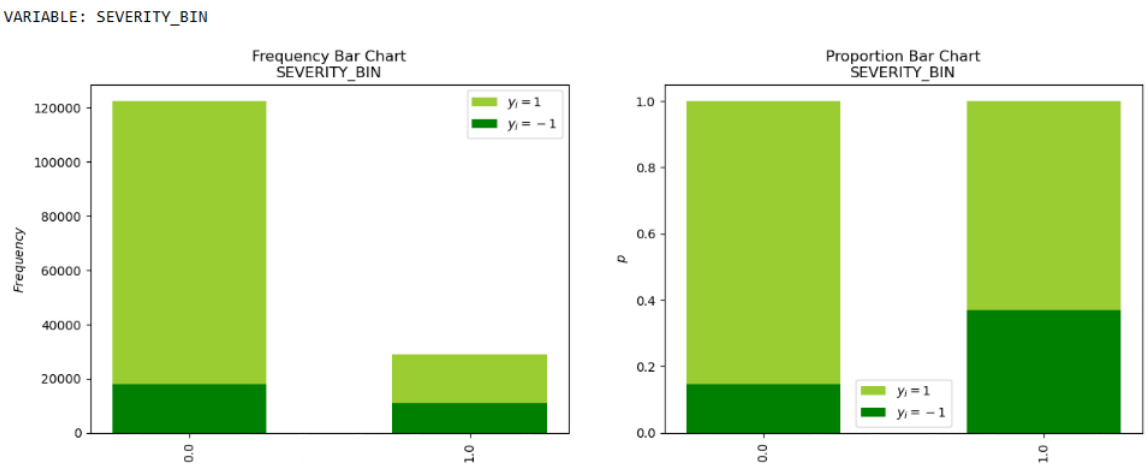


Figure 20 - Feature impact with SHAP values for the client segment

Finally, as the main objective is to perceive whether the detected abnormal behaviour in the client segment influences the frequency and the severity of claims, it was analysed in terms of proportion if the assigned outlier observations have a frequency and a severity higher than zero. Figure 21 demonstrates that less than 20% was considered an outlier when the client presented a *low-risk* profile. Moreover, it is shown that approximately 40% was identified as an abnormal observation for both insurance measures when their values were higher than zero, which is already a great indicator regarding the discrepancy of client profiles. As an afterthought, -1 indicates the anomalies, whereas 1 represents the normal instances.

On a side note, the frequency bar plot exhibits a significant quantity in zero, as the most preferred clients are the ones with a *low-risk* profile, i.e., who present zero-frequency and zero-severity, in order to diminish the entity's costs. So, the value 1 for both measures represents all instances with values higher than zero.



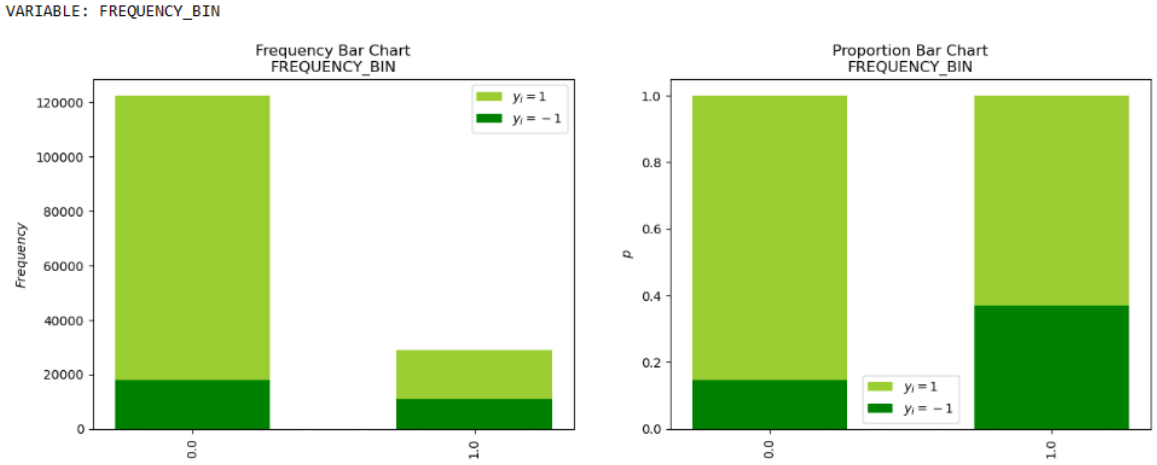


Figure 21 - Frequency and Proportion Bar Plots for frequency and severity attributes within the client segment

4.2. VEHICLE SEGMENT

For the vehicle segment, 16 features were used to extract some insights. Table 5 presents the *R-Squared* and *overall average silhouette score width* for clustering techniques, and Table 6 exhibits the obtained AUC scores for the remaining unsupervised anomaly detection models. In bold, the highest AUC is highlighted, representing the best model.

Table 5 - Evaluation performance metrics for clustering techniques and the number of clusters.

Model	# Clusters	R-Squared	Silhouette Score
DBSCAN	13	0.33	- 0.50
OPTICS	8	0.85	- 0.68

Concerning Table 5, the *R-Squared* and the *overall average silhouette score width* for DBSCAN demonstrate again the poor performance of such model in clustering the ingested data and, simultaneously, identifying outliers. Hence, it is reasonable to infer that the shape of the data may not be density-based algorithms friendly. Regarding the OPTICS algorithm, the *R-Squared* value is relatively high, expressing a good performance; nonetheless, when considering the *overall average silhouette score width*, the score value indicates that the overall classification of observations was wrongly made, contradicting the good performance evoked by the *R-Squared* value. *Appendix B – Vehicle Segment* exhibits, for each clustering technique, the distribution of each feature for each cluster with the aid of boxplots.

Table 6 - AUC scores for EIF, iForest and LOF algorithms.

Model	AUC
EIF	0.8145
iForest	**0.8274**
LOF	0.7265

Comparing the AUC scores in Table 6, the iForest model outstands with approximately 0.83 AUC value, whereas EIF and LOF take the second and third place, respectively. Due to their similar AUC scores, the forest visualisation was performed to obtain more insights regarding how, on average, the considered outliers reach much smaller depths than the normal instances. Figure 22 displays the forest visualisation for iForest and EIF algorithms. Perhaps, it is understandable that there is a higher propensity for anomalous instances to reach the maximum depth in the tree more frequently in iForest rather than in EIF. This may happen because of the difference in how the branch is cut, i.e., in iForest, the branch cuts are always vertical or horizontal, which can result in areas with inconsistent anomaly scores, whereas the EIF performs its branching operation in random directions, having into consideration the current instances on the tree node (Hariri et al., 2021).

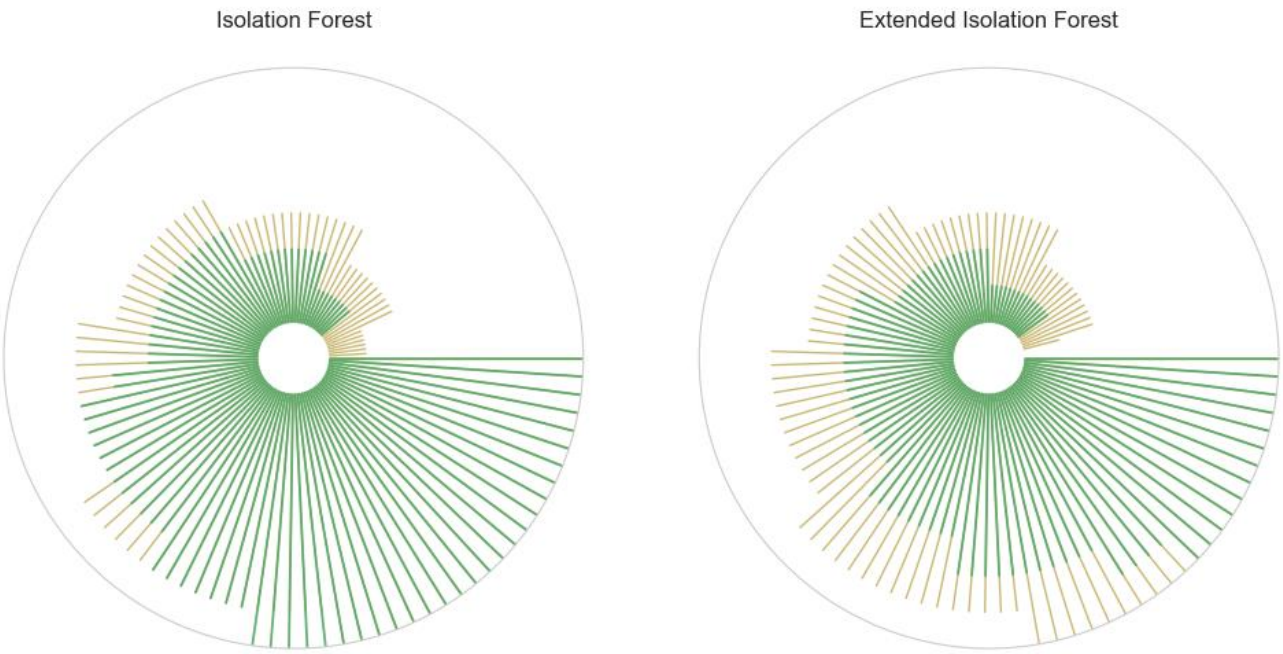


Figure 22 - Forest visualisation of iForest and EIF for the vehicle segment

Nonetheless, the DataRobot platform spotlights five features for the iForest model, which was the one with the highest AUC score. The impact of these features on the anomaly score is considered to be the topmost. Those are related to the vehicle valuation, cubic capacity, engine power and quantity of doors, and the total average annual premium value each policyholder supports, as shown in Figure 23.

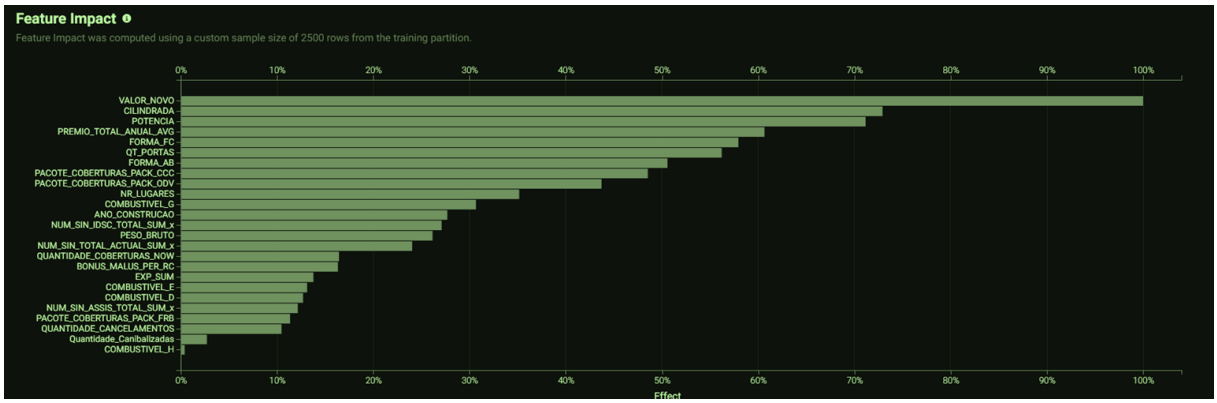
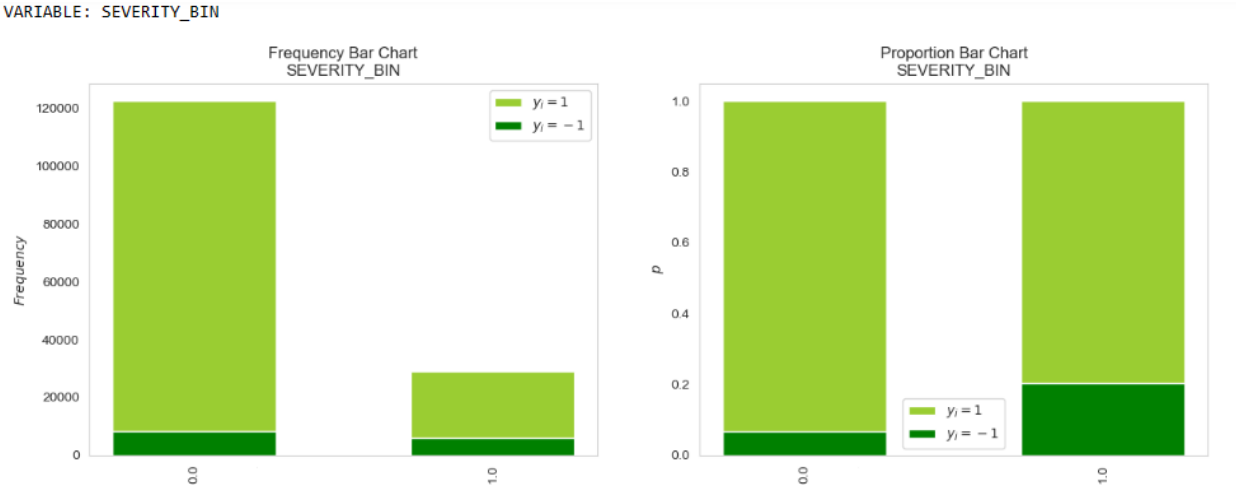


Figure 23 - Feature impact with SHAP values for the vehicle segment

Regarding the influence of anomalous vehicle characteristics on claims' frequency and severity, only approximately 20% of instances with values higher than zero, either in terms of severity or frequency, were considered anomalies, as demonstrated in Figure 24. This may indicate that the combinations of vehicle attributes may not impact the claim's frequency and severity.



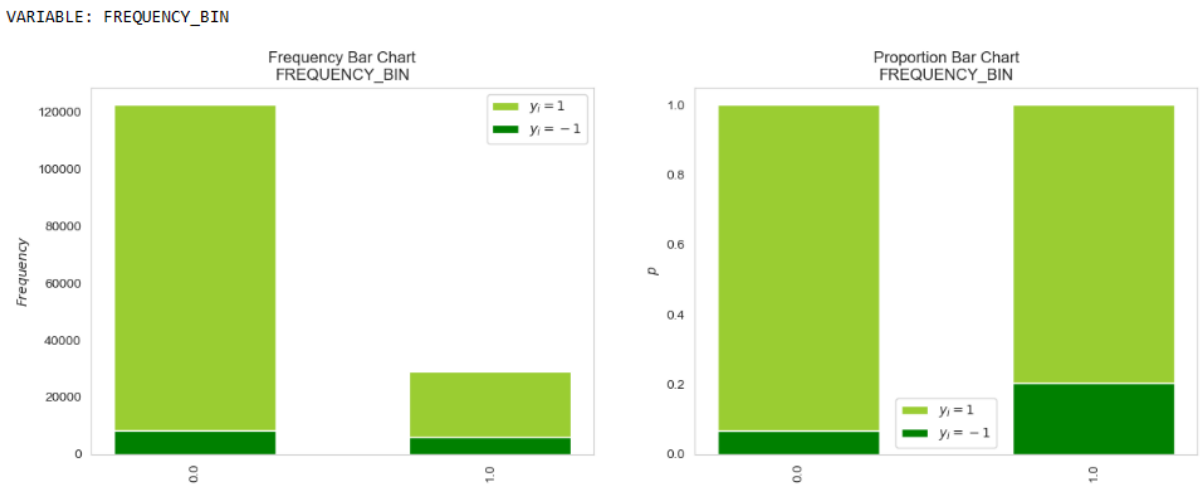


Figure 24 - Frequency and Proportion Bar Plots for frequency and severity attributes within the vehicle segment

4.3. WHOLE DATASET SEGMENT

For the whole dataset segment, due to computational resources and time consumption, it was performed only some algorithms, in this case, EIF, iForest and LOF models. To this end, Table 7 exposes the AUC scores for each unsupervised anomaly detection algorithm and, in bold, it is highlighted the best model according to the obtained AUC scores.

Table 7 - AUC scores for EIF, iForest and LOF algorithms.

Model	AUC
EIF	0.8578
iForest	**0.8619**
LOF	0.7225

By analysing the different models, the overall model performance is increased when ingesting the whole dataset, i.e., all the attributes related to the client and vehicle segments. The iForest model obtains the best AUC score of approximately 0.86. Once more, the EIF and iForest algorithms present similar AUC scores, and hence the forest visualisation was implemented to extract more insights. Figure 25 evokes the difference regarding the average depth of the trees between the EIF and iForest models. Straightforward, the regular instances overall achieve more extended depth trees in iForest rather than in EIF, and as demonstrated in the EIF's forest visualisation, there is not a significant discrepancy between the tree depth of regular instances and anomalous ones, which reinforces the better AUC score for iForest.

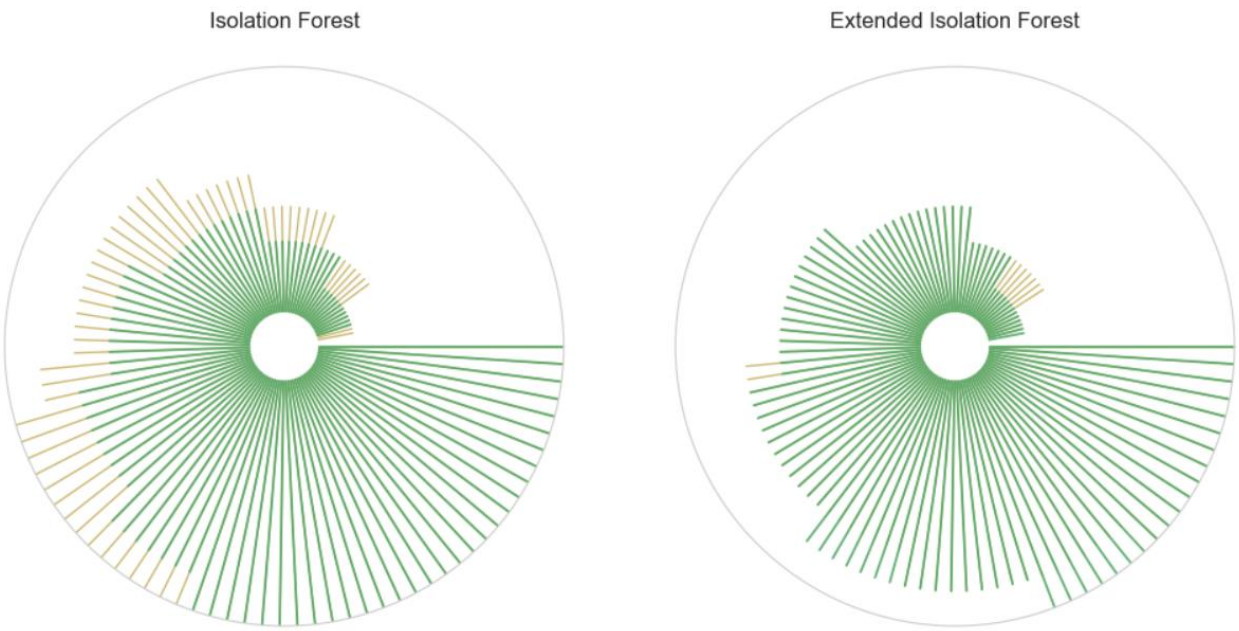


Figure 25 - Forest visualisation of iForest and EIF for the whole dataset segment

For the best algorithm, iForest, the feature impact was accessed according to the SHAP values performed by the DataRobot platform. As a result, it is vital to highlight the presence of certain top-ranked features that also had a more significant impact when the segments were evaluated separately. As Figure 26 displays, the total average annual premium value paid by the policyholder, the driver's civil status, the vehicle valuation and engine power are among the top five most important attributes that influence the computation of the anomaly score. Nonetheless, the top-25 features are emphasised in the bar plot to perceive better its influence on detecting abnormal behaviour.

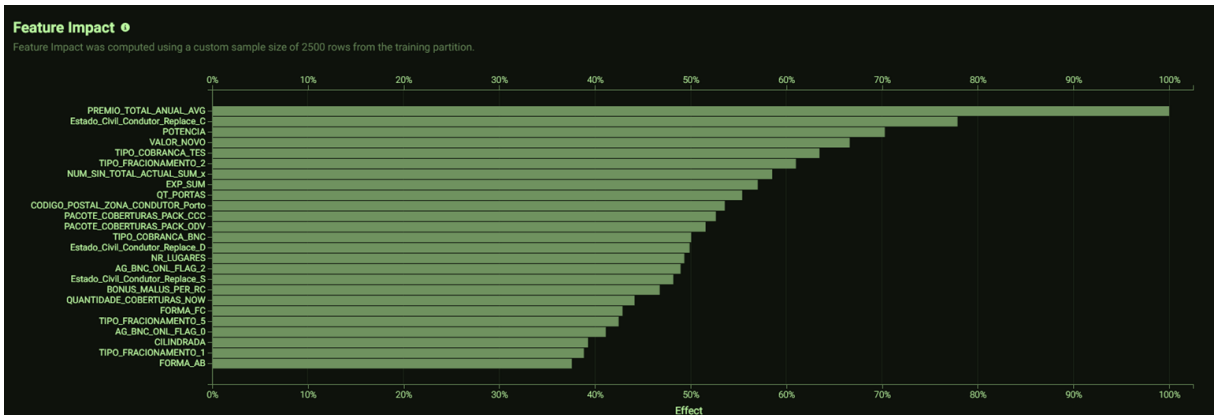


Figure 26 - Feature impact with SHAP values for the whole dataset segment

In addition, as the main focus of detecting anomalous instances is to perceive the influence on the claims' frequency and severity, Figure 27 indicates that approximately 60% of abnormal instances have values higher than zero for frequency and severity attributes, which is a good indicator. By contrast, approximately 10% of anomalous observations still present zero-frequency and zero-severity. Thus, when defining pricing and subscription rules, the previously mentioned features that had a more significant impact on defining the anomaly score should be analysed with careful attention in order to define a fairer tariff for both the insurance entity and the client; or even to identify unsought customers that may not have a *low-risk* profile, which is not what is desired for the insurance business.

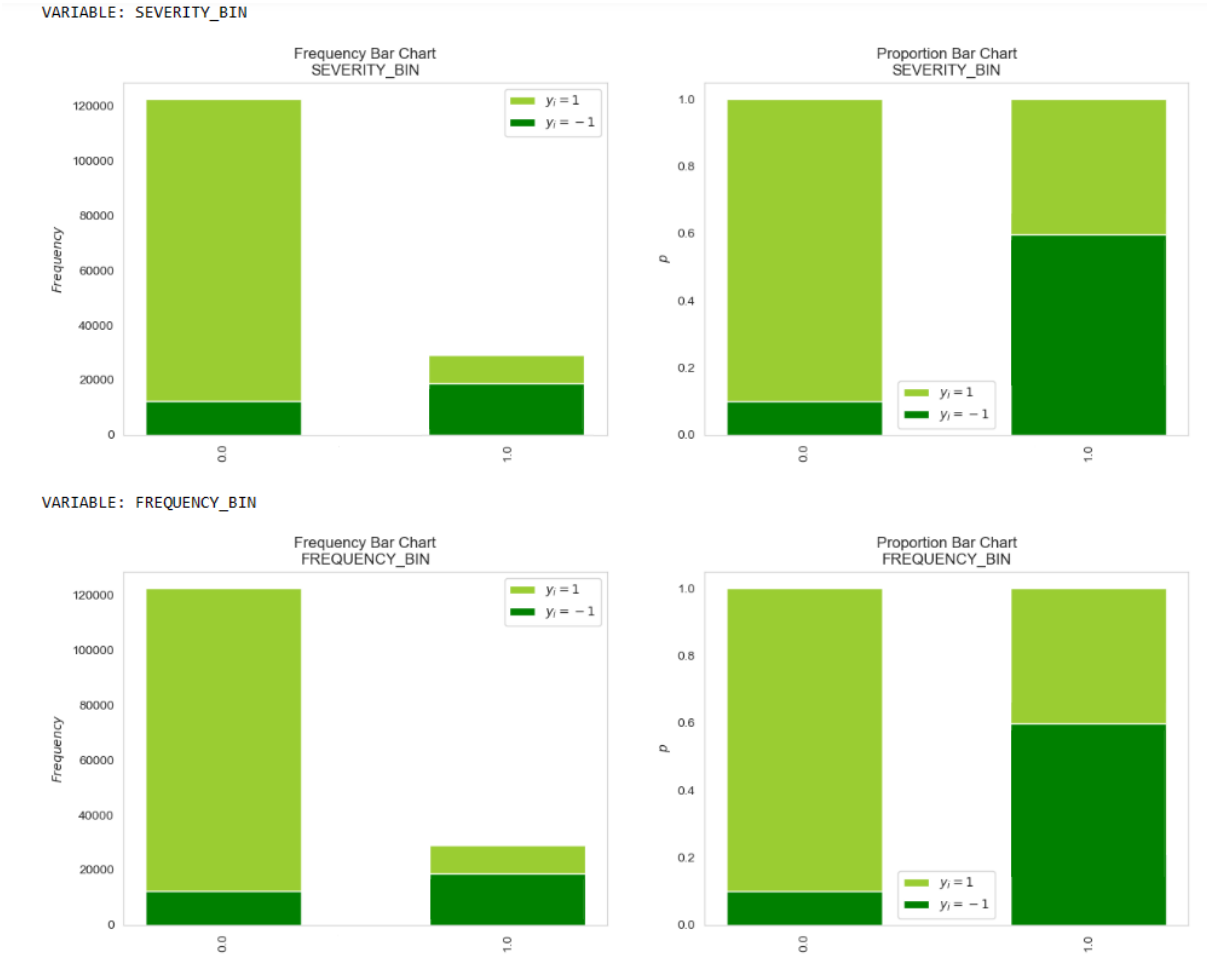


Figure 27 - Frequency and Proportion Bar Plots for frequency and severity attributes within the whole dataset segment

Nonetheless, when comparing this outcome with the ones previously mentioned, i.e., for separated segments, it is evident that when using all attributes, the percentage of anomalies within the non-zero frequency and severity group is higher, which is a satisfactory outcome for the purpose of this project. Furthermore, as the AUC score for iForest was approximately 0.86, it strengthens the decision to deploy this model according to the data at hand, making it the most suitable one. For a

more in-depth analysis, the t-distributed Stochastic Neighbour Embedding (t-SNE), a nonlinear dimensionality reduction technique, was performed to visualise high-dimensional data in a lower dimensional space, according to the generated matrix of pairwise similarities (van der Maaten & Hinton, 2008). The anomalies are coloured green, whereas the regular instances are coloured black. According to Figure 28, some isolated instances were not identified as anomalies, and the considered ones are more concentrated in one spectrum. Thus, according to the data distribution, there is always space for improvement and enhancement of the iForest model for unsupervised anomaly detection tasks.

In *Appendix C – t-SNE 3-D visualisation*, it is shown the t-SNE visualisation for a three-dimensional space, even though with a 2-D visualisation, some inputs can already be retrieved.

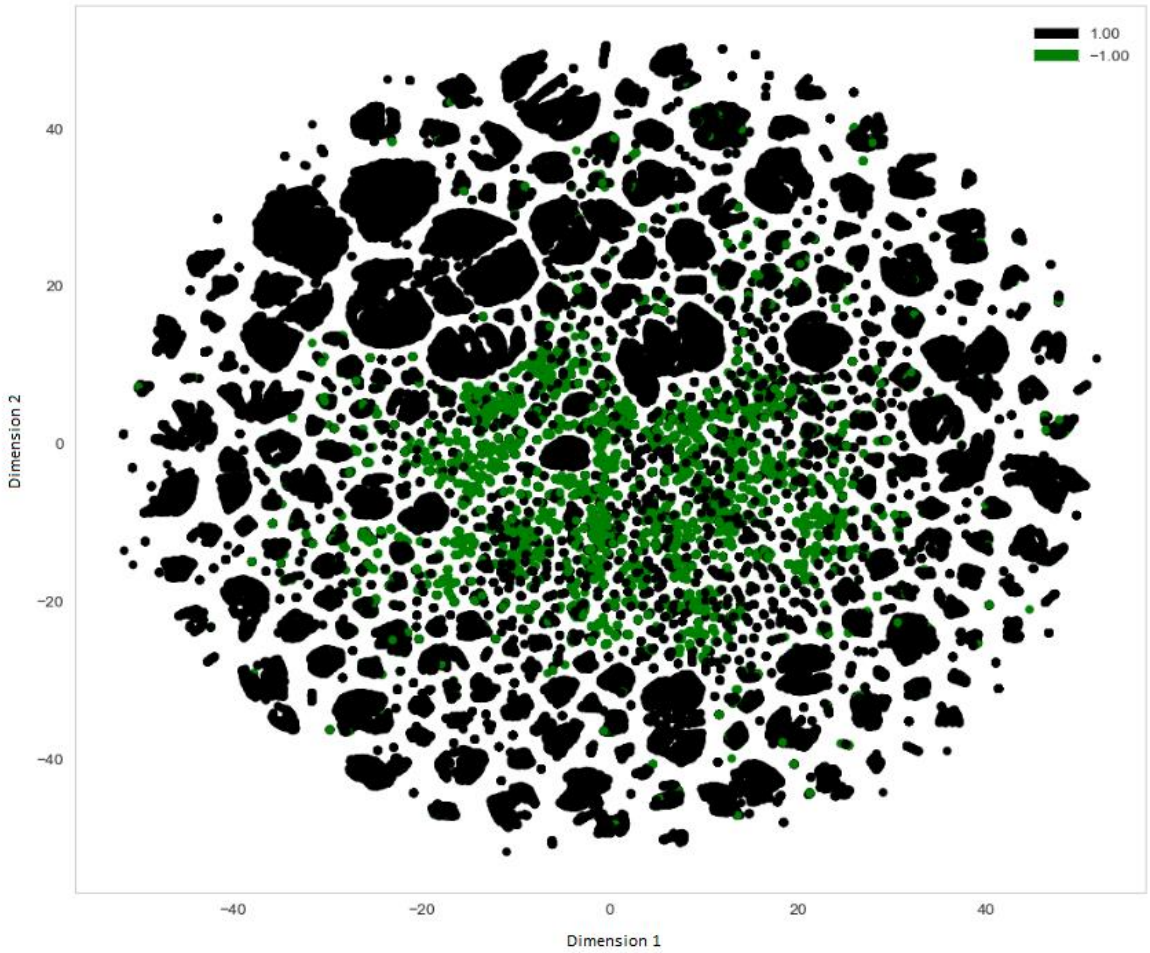


Figure 28 - The t-SNE visualisation of regular and anomalous data points into a 2-D space, according to the iForest model

5. CONCLUSIONS

Over the last few years, the insurance business has been evolving with technological advancements since the development of Bonus-Malus systems to the detection of fraudulent financial actions. One major priority of insurance companies is to provide a fairer tariff for the entity itself and its customers by stipulating adequate pricing and underwriting rules. However, with the amount of data generated nowadays, abnormalities are inevitable and may impact decision-making. Thus, even though the scientific community has experimented with and enhanced several unsupervised anomaly detection algorithms, to the best of our knowledge, fewer studies have been made regarding the identification of anomalous client profiles and vehicle characteristics that may influence the claims' frequency and severity, which are two crucial measures to evaluate the clients' risk-profile.

Based on a real-world dataset provided by a private insurance company that operates in Portugal, this thesis addresses two clustering techniques with the possibility to recognise outliers - the DBSCAN and the OPTICS algorithms - as well as three unsupervised anomaly detection models, such as Isolation Forest, Extended Isolation Forest and Local Outlier Factor, by using the DataRobot platform and also the Python ML library *scikit-learn*.

For conducting this project, the CRISP-DM methodology was the guideline followed. Firstly, an overview of the business goals and the problem definition were done in *Chapter 1 – Introduction* to present the purpose of this research and some initial limitations and constraints. Secondly, a descriptive and exploratory analysis of the anonymised data was performed to retrieve the first preliminary insights and verify the data quality. Thirdly, to obtain a dataset with greater consistency, the missForest algorithm was executed for the imputation of missing values, as the dataset was composed of mixed-type attributes. Additionally, the differentiation between a noisy instance and an outlier in the insurance context was also taken into account, as well as the distinct encoding techniques to convert high-cardinality categorical features into numerical attributes, for instance, the application of one-hot encoding and binary encoding. Finally, the normalisation of the data through *Min-Max* scaling and the data compression to obtain an accumulative 5-year historical information of each driver and vehicle were achieved. All things considered, the data was prepared to be ingested into the Modelling phase and, therefore, into the Evaluation phase.

During the Modelling phase, two experimental segments were created by splitting the data into client and vehicle attributes. Nonetheless, it was also used the whole dataset as a third segment. As suggested in several research papers, an evaluation of the models' performance outcomes was undertaken by applying the distinct algorithms with the default hyperparameters. As such, the *R-Squared* and the *overall average silhouette score width* were taken as model evaluation criteria for clustering techniques. On the other hand, for the remaining unsupervised anomaly detection algorithms, the *Area Under the Curve* score was the chosen model evaluation criterion. In addition, for model explainability, using SHAP values allowed us to perceive the features' importance and interpret the different results.

When analysing the obtained experimental outcomes, the Isolation Forest algorithm outperformed the remaining models in each segment, particularly when using the whole dataset, by achieving an AUC score of approximately 0.86. Compared with its improved model, the Extended Isolation Forest, the AUC scores did not vary significantly, regardless of the segment type dealt with. However, when

the forest visualisation for these algorithms was created, it was possible to verify the difference between the overall trees' depth when comparing an anomalous observation with a regular instance. For the whole dataset segment, it was concluded that, on average, anomalous observations reach much smaller depths than the normal ones when applying the Isolation Forest, reinforcing the better AUC score for this model.

In general, referring to the views of Breunig et al. (2000), the poor performance of clustering methods might be related to their intrinsic property of being optimised to detect clusters rather than to find outliers. Furthermore, it may be reasonable to assume that the shape of the data at hand may not be density-based algorithms friendly. Nonetheless, the main benefit of the OPTICS method is the needlessness of defining *a priori* the ε parameter, although increasing the time-complexity, which enhances the model performance, according to the obtained *R-Squared* value.

Finally, by focusing on the best model for the whole dataset segment, Isolation Forest, specific attributes were classified as having more impact on the attribution of the anomaly scores. Specifically, for the considered dataset, the total average annual premium value paid by the policyholder, the driver's civil status, vehicle valuation and engine power are among the top five most important features.

Moreover, when using all attributes, approximately 60% of abnormal instances had values higher than zero for claims' frequency and severity attributes, which was a good indicator considering that the principal focus of detecting anomalous instances was to perceive the influence on these two measures.

To sum up, the proposed research questions mentioned in *Chapter 1 - Introduction* were all addressed throughout this project, fulfilling its main goals. From now on, the insurance company has the possibility to provide more added value in the decision-making process on pricing and underwriting rules, as well as to identify unsought clients that may not have a *low-risk* profile according to their characteristics or even when assessing the attributes of the insured vehicle.

6. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORKS

The development of this research was based on a case study using a real-world dataset given by a private insurance entity. Due to confidentiality protocols, the data was anonymised, and additionally, it was subjected to business model restrictions, which limited some potential approaches.

One of the significant adversities during the realisation of this project was the several inconsistencies presented in the databases' historical data, which was translated into a data processing phase that was more demanding and extremely time-consuming. Likewise, because of the size of the data and its magnitude, as it was initially composed of 301 features and approximately 77 million instances, the modelling phase was carried out using a sample of 10% due to IT resource limitations, even though all efforts have been taken to reduce it. A suggestion for future work could be to use the full processed dataset and assess if the performance of distinct models corresponds to the equivalent outcome using the retrieved subset of instances.

Another aspect that should be highlighted concerns the inexistence of a labelled dataset, which makes detecting anomalies even more arduous. When using unsupervised learning approaches, evaluating the ground truth of the considered anomalies is complex and made basically through business knowledge. Nonetheless, Goix (2016) suggests two novel criteria that do not require labels and may be worth trying. Those are based on existing Excess-Mass and Mass-Volume curves, which generally are not well estimated in large dimensions. For this reason, it was not applied in this project, although being a recommendation for future work.

Furthermore, regarding the data preparation phase, some other techniques that were not mentioned during this project were tested to perceive if there was indeed an increase in the models' performance. For instance, some categorical encoding techniques tested were hashing and frequency encoding. Also, some dimensionality reduction techniques were performed, such as Principal Component Analysis and Factor Analysis of Mixed Data. However, the outcomes were not satisfactory as the ones described previously, but it highlights the importance of the techniques' choice. With that said, other approaches may be interesting to adopt, as they can enhance the models' performance and enrich the research.

Finally, the models' performance is affected by the setting of each algorithm's hyperparameters, and it usually is a decision made by the user. Consequently, Soenen et al. (2021) propose using a small validation set to tune an anomaly detector's hyperparameters on a per-dataset basis. Although acquiring labelled data is costly, it would be a smaller proportion in this case, balancing this spending with a fairer selection of hyperparameters. This approach could bring more feasible outcomes and add value to this research.

7. REFERENCES

Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. *Journal of Network and Computer Applications*, *68*, 90–113. https://doi.org/10.1016/j.jnca.2016.04.007

Aggarwal, C. C. (2017). An Introduction to Outlier Analysis. In *Outlier Analysis* (pp. 1–34). Springer International Publishing. https://doi.org/10.1007/978-3-319-47578-3_1

Agrawal, R., Kadadi, A., Dai, X., & Andres, F. (2015). Challenges and opportunities with big data visualization. *Proceedings of the 7th International Conference on Management of Computational and Collective IntElligence in Digital EcoSystems*, 169–173. https://doi.org/10.1145/2857218.2857256

Ahmed, F., & Courville, A. (2020). Detecting Semantic Anomalies. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(04), 3154–3162. https://doi.org/10.1609/aaai.v34i04.5712

Akoglu, H. (2018). User's guide to correlation coefficients. *Turkish Journal of Emergency Medicine*, *18*(3), 91–93. https://doi.org/10.1016/j.tjem.2018.08.001

Albrecher, H., Bommier, A., Filipović, D., Koch-Medina, P., Loisel, S., & Schmeiser, H. (2019). Insurance: models, digitalization, and data science. *European Actuarial Journal*, *9*(2), 349–360. https://doi.org/10.1007/s13385-019-00209-x

Alhussein, I., & Ali, A. H. (2020). Application of DBSCAN to Anomaly Detection in Airport Terminals. *2020 3rd International Conference on Engineering Technology and Its Applications (IICETA)*, 112–116. https://doi.org/10.1109/IICETA50496.2020.9318876

Ankerst, M., Breunig, M. M., Kriegel, H.-P., & Sander, J. (1999). OPTICS. *ACM SIGMOD Record*, *28*(2), 49–60. https://doi.org/10.1145/304181.304187

Antwarg, L., Miller, R. M., Shapira, B., & Rokach, L. (2021). Explaining anomalies detected by autoencoders using Shapley Additive Explanations. *Expert Systems with Applications*, *186*, 115736. https://doi.org/10.1016/j.eswa.2021.115736

Berthelé, E. (2018). Using Big Data in Insurance. In *Big Data for Insurance Companies* (pp. 131–161). John Wiley & Sons, Inc. https://doi.org/10.1002/9781119489368.ch5

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF. *ACM SIGMOD Record*, *29*(2), 93–104. https://doi.org/10.1145/335191.335388

Butun, I., Morgera, S. D., & Sankar, R. (2014). A Survey of Intrusion Detection Systems in Wireless Sensor Networks. *IEEE Communications Surveys & Tutorials*, *16*(1), 266–282. https://doi.org/10.1109/SURV.2013.050113.00191

Campos, G. O., Zimek, A., Sander, J., Campello, R. J. G. B., Micenková, B., Schubert, E., Assent, I., & Houle, M. E. (2016). On the evaluation of unsupervised outlier detection: measures,

datasets, and an empirical study. *Data Mining and Knowledge Discovery*, *30*(4), 891–927. https://doi.org/10.1007/s10618-015-0444-8

Cerda, P., & Varoquaux, G. (2022). Encoding High-Cardinality String Categorical Variables. *IEEE Transactions on Knowledge and Data Engineering*, *34*(3), 1164–1176. https://doi.org/10.1109/TKDE.2020.2992529

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection. *ACM Computing Surveys*, *41*(3), 1–58. https://doi.org/10.1145/1541880.1541882

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*. DaimlerChrysler.

Cohen, J., Cohen, P., G. West, S., & S. Aiken, L. (2013). *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. Routledge. https://doi.org/10.4324/9780203774441

Datta, D., Islam, M. R., Self, N., Meadows, A., Simeone, J., Outhwaite, W., Hin Keong, C., Smith, A., Walker, L., & Ramakrishnan, N. (2020). Detecting Suspicious Timber Trades. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(08), 13248–13254. https://doi.org/10.1609/aaai.v34i08.7032

Davis, M. J. (2021). Contrast Coding in Multiple Regression Analysis: Strengths, Weaknesses, and Utility of Popular Coding Structures. *Journal of Data Science*, *8*(1), 61–73. https://doi.org/10.6339/JDS.2010.08(1).563

Denny, M. J., & Spirling, A. (2018). Text Preprocessing For Unsupervised Learning: Why It Matters, When It Misleads, And What To Do About It. *Political Analysis*, *26*(2), 168–189. https://doi.org/10.1017/pan.2017.44

Domingues, R., Filippone, M., Michiardi, P., & Zouaoui, J. (2018). A comparative evaluation of outlier detection algorithms: Experiments and analyses. *Pattern Recognition*, *74*, 406–421. https://doi.org/10.1016/j.patcog.2017.09.037

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996, December 31). A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*.

Ewold, F. (1991). Insurance and risk. In G. Burchell, C. Gordon, & P. Miller (Eds.), *The Foucault effect: Studies in governmentality* (pp. 197–210). Harvester Wheatsheaf.

Fahim, M., & Sillitti, A. (2019). Anomaly Detection, Analysis and Prediction Techniques in IoT Environment: A Systematic Literature Review. *IEEE Access*, *7*, 81664–81681. https://doi.org/10.1109/ACCESS.2019.2921912

Fan, L., Ma, J., Tian, J., Li, T., & Wang, H. (2021). Comparative Study of Isolation Forest and LOF algorithm in anomaly detection of data mining. *2021 International Conference on Big Data, Artificial Intelligence and Risk Management (ICBAR)*, 1–5. https://doi.org/10.1109/ICBAR55169.2021.00008

Fathnia, F., & Bayaz, M. H. J. D. (2018). Anomaly Detection in Smart Grid with Help of an Improved OPTICS Using Coefficient of Variation. *Electrical Engineering (ICEE), Iranian Conference On*, 1044–1050. https://doi.org/10.1109/ICEE.2018.8472534

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Frangos, N. E., & Vrontos, S. D. (2001). Design of Optimal Bonus-Malus Systems With a Frequency and a Severity Component On an Individual Basis in Automobile Insurance. *ASTIN Bulletin*, *31*(1), 1–22. https://doi.org/10.2143/AST.31.1.991

García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*, *28*(1–2), 18–28. https://doi.org/10.1016/j.cose.2008.08.003

Goix, N. (2016). How to Evaluate the Quality of Unsupervised Anomaly Detection Algorithms? *ArXiv Preprint*. https://doi.org/10.48550/arXiv.1607.01152

Goldstein, M. (2014). *Anomaly Detection in Large Datasets* [PhD-Thesis]. Technische Universitaet Kaiserslautern, Dr. Hut Verlag Muenchen, 2/2014. ISBN: 978-3-8439-1572-4.

Goldstein, M., & Uchida, S. (2016). A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. *PLOS ONE*, *11*(4), e0152173. https://doi.org/10.1371/journal.pone.0152173

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2002). Clustering validity checking methods. *ACM SIGMOD Record*, *31*(3), 19–27. https://doi.org/10.1145/601858.601862

Hall, M. A. (1999). *Correlation-based Feature Selection for Machine Learning* [PhD Thesis]. University of Waikato.

Hanafy, M., & Ming, R. (2021). Machine Learning Approaches for Auto Insurance Big Data. *Risks*, *9*(2), 42. https://doi.org/10.3390/risks9020042

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, *143*(1), 29–36. https://doi.org/10.1148/radiology.143.1.7063747

Hariri, S., Kind, M. C., & Brunner, R. J. (2021). Extended Isolation Forest. *IEEE Transactions on Knowledge and Data Engineering*, *33*(4), 1479–1489. https://doi.org/10.1109/TKDE.2019.2947676

Hassani, H., Unger, S., & Beneki, C. (2020). Big Data and Actuarial Science. *Big Data and Cognitive Computing*, *4*(4), 40. https://doi.org/10.3390/bdcc4040040

Hawkins, D. M. (1980). *Identification of Outliers*. Springer Netherlands. https://doi.org/10.1007/978-94-015-3994-4

Hawley, R. W., & Gallagher, N. C. (1994). On Edgeworth's method for minimum absolute error linear regression. *IEEE Transactions on Signal Processing*, *42*(8), 2045–2054. https://doi.org/10.1109/78.301827

Hilal, W., Gadsden, S. A., & Yawney, J. (2022). Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. *Expert Systems with Applications*, *193*, 116429. https://doi.org/10.1016/j.eswa.2021.116429

Hinneburg, A., & Keim, D. A. (1999). Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering. *Proceedings of the 25th International Conference on Very Large Data Bases*, 506–517.

Jiang, S., Song, X., Wang, H., Han, J.-J., & Li, Q.-H. (2006). A clustering-based method for unsupervised intrusion detections. *Pattern Recognition Letters*, *27*(7), 802–810. https://doi.org/10.1016/j.patrec.2005.11.007

Kandanaarachchi, S., Muñoz, M. A., Hyndman, R. J., & Smith-Miles, K. (2020). On normalization and algorithm selection for unsupervised outlier detection. *Data Mining and Knowledge Discovery*, *34*(2), 309–354. https://doi.org/10.1007/s10618-019-00661-z

Kotu, V., & Deshpande, B. (2019). Anomaly Detection. In *Data Science* (pp. 447–465). Elsevier. https://doi.org/10.1016/B978-0-12-814761-0.00013-7

Lin, W.-C., & Tsai, C.-F. (2020). Missing value imputation: a review and analysis of the literature (2006–2017). *Artificial Intelligence Review*, *53*(2), 1487–1509. https://doi.org/10.1007/s10462-019-09709-4

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. *2008 Eighth IEEE International Conference on Data Mining*, 413–422. https://doi.org/10.1109/ICDM.2008.17

Lundberg, S. M., Allen, P. G., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, *30*. https://github.com/slundberg/shap

Ma, M. (2022). The Operation of China's Insurance Industry in the Context of Big Data: Dilemmas, Challenges and Countermeasures. *Beijing Law Review*, *13*(04), 853–863. https://doi.org/10.4236/blr.2022.134056

Meyers, L. S., Gamst, G., & Guarino, A. J. (2009). *Data Analysis Using SAS Enterprise Guide*. Cambridge University Press. https://doi.org/10.1017/CBO9780511804786

N., D., J., S., & P., S. (2022). Intrusion Detection in Wireless Sensor Networks using Optics Algorithm. *2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, 1265–1272. https://doi.org/10.1109/ICAAIC53929.2022.9793233

Nassif, A. B., Talib, M. A., Nasir, Q., & Dakalbab, F. M. (2021). Machine Learning for Anomaly Detection: A Systematic Review. *IEEE Access*, *9*, 78658–78700. https://doi.org/10.1109/ACCESS.2021.3083060

Nian, K., Zhang, H., Tayal, A., Coleman, T., & Li, Y. (2016). Auto insurance fraud detection using unsupervised spectral ranking for anomaly. *The Journal of Finance and Data Science*, *2*(1), 58–75. https://doi.org/10.1016/j.jfds.2016.03.001

Ohlsson, E., & Johansson, B. (2010). *Non-Life Insurance Pricing with Generalized Linear Models*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-10791-7

Outreville, J. F. (1998). Insurance Concepts. In *Theory and Practice of Insurance* (pp. 131–146). Springer US. https://doi.org/10.1007/978-1-4615-6187-3_8

Pang, G., Shen, C., Cao, L., & Hengel, A. van den. (2022). Deep Learning for Anomaly Detection. *ACM Computing Surveys*, *54*(2), 1–38. https://doi.org/10.1145/3439950

Panigrahi, S., Kundu, A., Sural, S., & Majumdar, A. K. (2009). Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning. *Information Fusion*, *10*(4), 354–363. https://doi.org/10.1016/j.inffus.2008.04.001

Panja, S., Patowary, N., Saha, S., & Nag, A. (2022). *Anomaly Detection in IoT Using Extended Isolation Forest* (pp. 3–14). https://doi.org/10.1007/978-3-031-22485-0_1

Parr-Rud, O. (2014). *Business Analytics Using SAS Enterprise Guide and SAS Enterprise Miner* (1st ed.). SAS Institute.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Müller, A., Nothman, J., Louppe, G., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2012). *Scikit-learn: Machine Learning in Python*.

Persson, I., & Khojasteh, J. (2021). Python Packages for Exploratory Factor Analysis. *Structural Equation Modeling: A Multidisciplinary Journal*, *28*(6), 983–988. https://doi.org/10.1080/10705511.2021.1910037

Potdar, K., S., T., & D., C. (2017). A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers. *International Journal of Computer Applications*, *175*(4), 7–9. https://doi.org/10.5120/ijca2017915495

Ramsauer, A., Baumann, P. M., & Lex, C. (2021). The Influence of Data Preparation on Outlier Detection in Driveability Data. *SN Computer Science*, *2*(3), 222. https://doi.org/10.1007/s42979-021-00607-7

Raschka, S., & Mirjalili, V. (2019). *Python Machine Learning* (J. Malysiak, S. Jain, J. Lovell, C. Nelson, S. D'silva, & R. Atitkar, Eds.; 3rd ed.). Packt Publishing Ltd.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Salvador, S., & Chan, P. (2005). Learning States and Rules for Detecting Anomalies in Time Series. *Applied Intelligence*, *23*(3), 241–255. https://doi.org/10.1007/s10489-005-4610-3

Sander, J., Ester, M., Kriegel, H.-P., & Xu, X. (1998). Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Mining and Knowledge Discovery*, *2*(2), 169–194. https://doi.org/10.1023/A:1009745219419

Sawant, K. (2014). Adaptive Methods for Determining DBSCAN Parameters. In *IJISET-International Journal of Innovative Science, Engineering & Technology* (Vol. 1). www.ijiset.com

Say, L., & Chailley-Bert, J. (1891). *Nouveau dictionnaire d'économie politique* (Vol. 2). Guillaumin et Cie.

Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, *181*, 526–534. https://doi.org/10.1016/j.procs.2021.01.199

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017). DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Transactions on Database Systems*, *42*(3), 1–21. https://doi.org/10.1145/3068335

Seger, C. (2018). *An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing* [Dissertation]. KTH, School of Electrical Engineering and Computer Science.

Shafique, U., & Qaiser, H. (2014). A Comparative Study of Data Mining Process Models (KDD, CRISP-DM and SEMMA). *International Journal of Innovation and Scientific Research*, *12*(1), 217–222.

Shekhar, S., Lu, C.-T., & Zhang, P. (2001). Detecting graph-based spatial outliers. *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 371–376. https://doi.org/10.1145/502512.502567

Singh, H. V., Girdhar, A., & Dahiya, S. (2022). A Literature survey based on DBSCAN algorithms. *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 751–758. https://doi.org/10.1109/ICICCS53718.2022.9788440

Smiti, A. (2020). A critical overview of outlier detection methods. *Computer Science Review*, *38*, 100306. https://doi.org/10.1016/j.cosrev.2020.100306

Soenen, J., van Wolputte, E., Perini, L., Vercruyssen, V., Meert, W., Davis, J., & Blockeel, H. (2021). The Effect of Hyperparameter Tuning on the Comparative Evaluation of Unsupervised Anomaly Detection Methods. *ODD'21*.

Song, X., Wu, M., Jermaine, C., & Ranka, S. (2007). Conditional Anomaly Detection. *IEEE Transactions on Knowledge and Data Engineering*, *19*(5), 631–645. https://doi.org/10.1109/TKDE.2007.1009

Spiteri, M., & Azzopardi, G. (2018). Customer Churn Prediction for a Motor Insurance Company. *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, 173–178. https://doi.org/10.1109/ICDIM.2018.8847066

Stančin, I., & Jovic, A. (2019). An overview and comparison of free Python libraries for data mining and big data analysis. *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 977–982. https://doi.org/10.23919/MIPRO.2019.8757088

Stekhoven, D. J., & Buhlmann, P. (2012). MissForest--non-parametric missing value imputation for mixed-type data. *Bioinformatics*, *28*(1), 112–118. https://doi.org/10.1093/bioinformatics/btr597

Strike, K., el Emam, K., & Madhavji, N. (2001). Software cost estimation with incomplete data. *IEEE Transactions on Software Engineering*, *27*(10), 890–908. https://doi.org/10.1109/32.962560

Sun, L., Versteeg, S., Boztas, S., & Rao, A. (2016). *Detecting Anomalous User Behavior Using an Extended Isolation Forest Algorithm: An Enterprise Case Study*.

Suthar, N., jeet Rajput, I., & kumar Gupta, V. (2013). A Technical Survey on DBSCAN Clustering Algorithm. *International Journal of Scientific & Engineering Research*, *4*(5). http://www.ijser.org

Tahir, M., Li, M., Zheng, X., Carie, A., Jin, X., Azhar, M., Ayoub, N., Wagan, A., Aamir, M., Ali, L., Asif, M., & Hussain, Z. (2019). A Novel Network user Behaviors and Profile Testing based on Anomaly Detection Techniques. *International Journal of Advanced Computer Science and Applications*, *10*(6). https://doi.org/10.14569/IJACSA.2019.0100641

Theissler, A. (2017). Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection. *Knowledge-Based Systems*, *123*, 163–173. https://doi.org/10.1016/j.knosys.2017.02.023

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., & Altman, R. B. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics*, *17*(6), 520–525. https://doi.org/10.1093/bioinformatics/17.6.520

Ul Haq, I., Gondal, I., Vamplew, P., & Brown, S. (2019). *Categorical Features Transformation with Compact One-Hot Encoder for Fraud Detection in Distributed Environment* (pp. 69–80). https://doi.org/10.1007/978-981-13-6661-1_6

van der Maaten, L., & Hinton, G. (2008). Visualizing Data using t-SNE. In *Journal of Machine Learning Research* (Vol. 9).

van Rossum, G. (1995). *Python tutorial: Vol. 1.2*. Centrum Wiskunde & Informatica.

van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual* (1st ed.). CreateSpace.

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a Standard Process Model for Data Mining. *Practical Application of Knowledge Discovery and Data Mining*, 29–40.

Xiaoyun, W., & Danyue, L. (2010). Hybrid outlier mining algorithm based evaluation of client moral risk in insurance company. *2010 2nd IEEE International Conference on Information Management and Engineering*, 585–589. https://doi.org/10.1109/ICIME.2010.5478070

Xu, X., Liu, H., & Yao, M. (2019). Recent Progress of Anomaly Detection. *Complexity*, *2019*, 1–11. https://doi.org/10.1155/2019/2686378

Yepmo, V., Smits, G., & Pivert, O. (2022). Anomaly explanation: A review. *Data & Knowledge Engineering*, *137*, 101946. https://doi.org/10.1016/j.datak.2021.101946

Yu, L., Zhou, R., Chen, R., & Lai, K. K. (2022). Missing Data Preprocessing in Credit Classification: One-Hot Encoding or Imputation? *Emerging Markets Finance and Trade*, *58*(2), 472–482. https://doi.org/10.1080/1540496X.2020.1825935

Zhang, C., Xiao, X., & Wu, C. (2020). Medical Fraud and Abuse Detection System Based on Machine Learning. *International Journal of Environmental Research and Public Health*, *17*(19), 7265. https://doi.org/10.3390/ijerph17197265

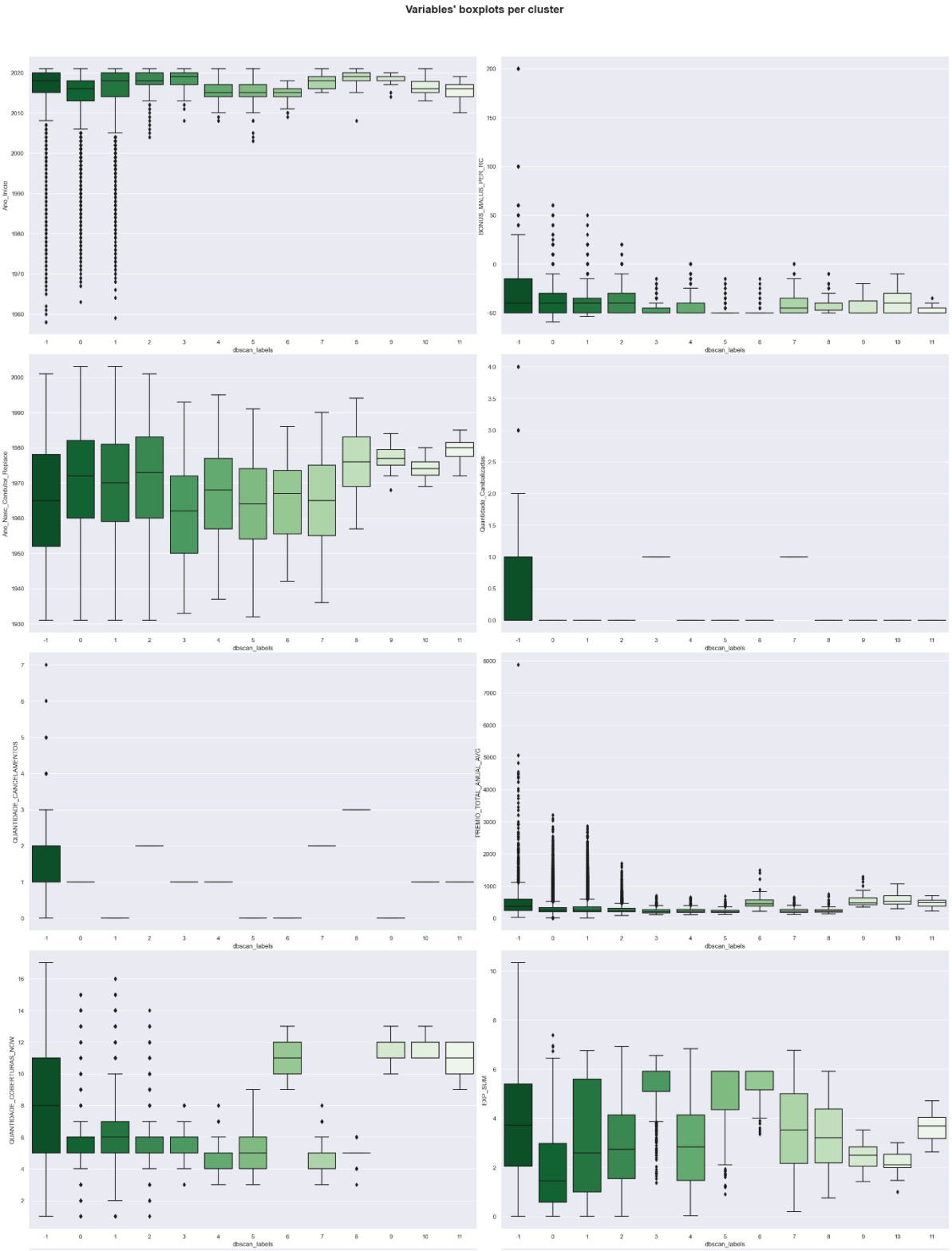
Zhang, K., & Jin, H. (2010). *An Effective Pattern Based Outlier Detection Approach for Mixed Attribute Data* (pp. 122–131). https://doi.org/10.1007/978-3-642-17432-2_13

Zhang, S., Zhang, C., & Yang, Q. (2003). Data preparation for data mining. *Applied Artificial Intelligence*, *17*(5–6), 375–381. https://doi.org/10.1080/713827180

8. APPENDIX

8.1. APPENDIX A – CLIENT SEGMENT

Here it is exhibited the boxplots related to each cluster and metric feature for both DBSCAN and OPTICS models. For the DBSCAN algorithm, Figure 1 presents the respective boxplots.



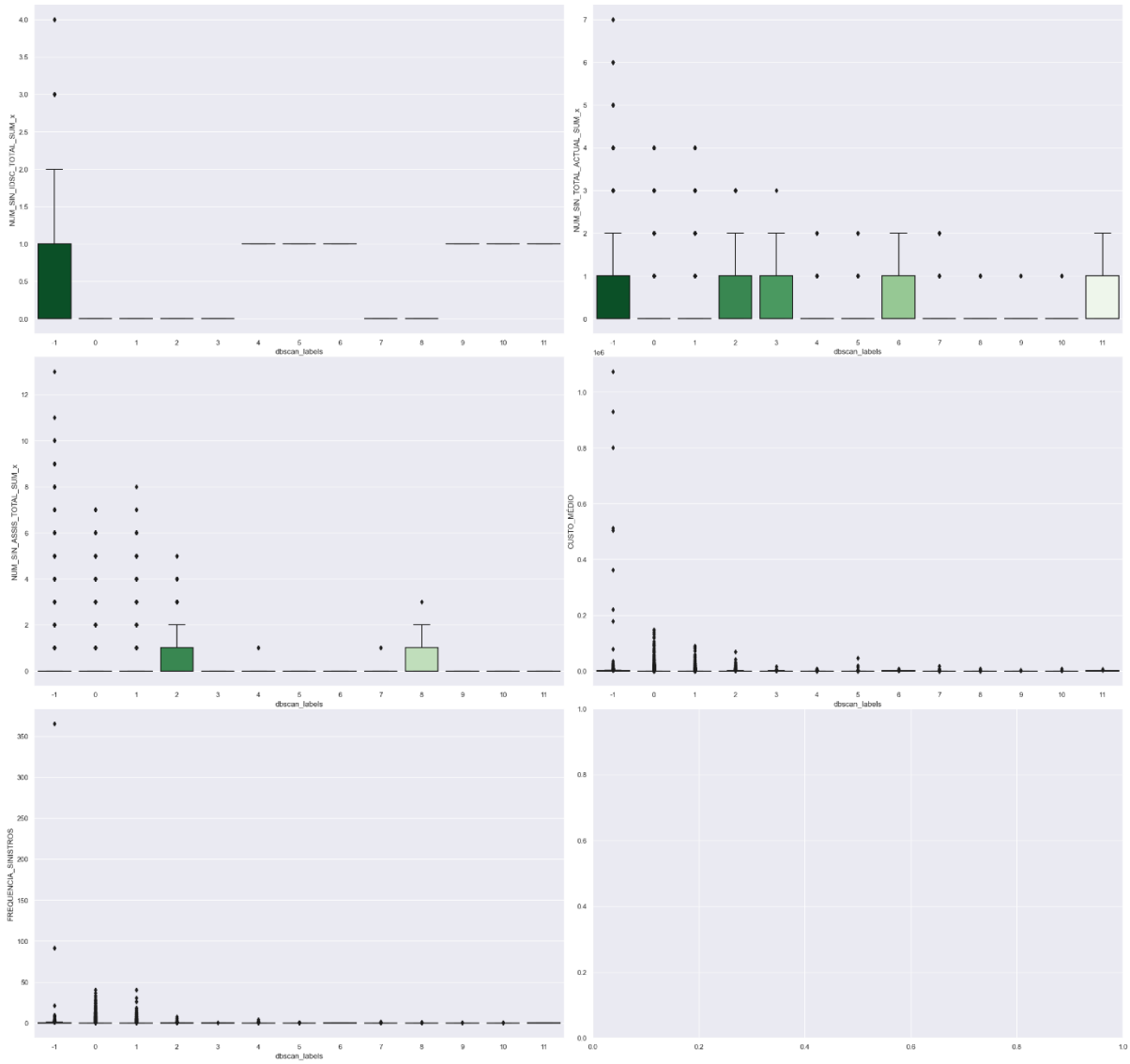
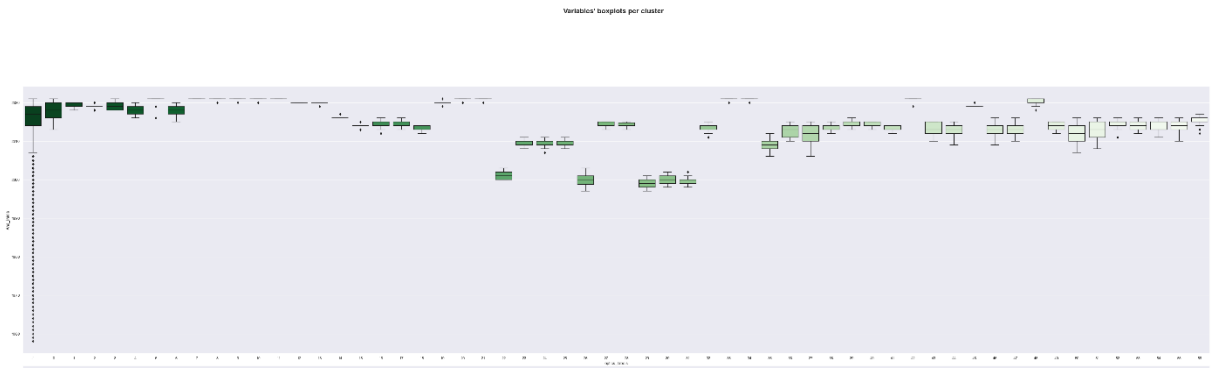
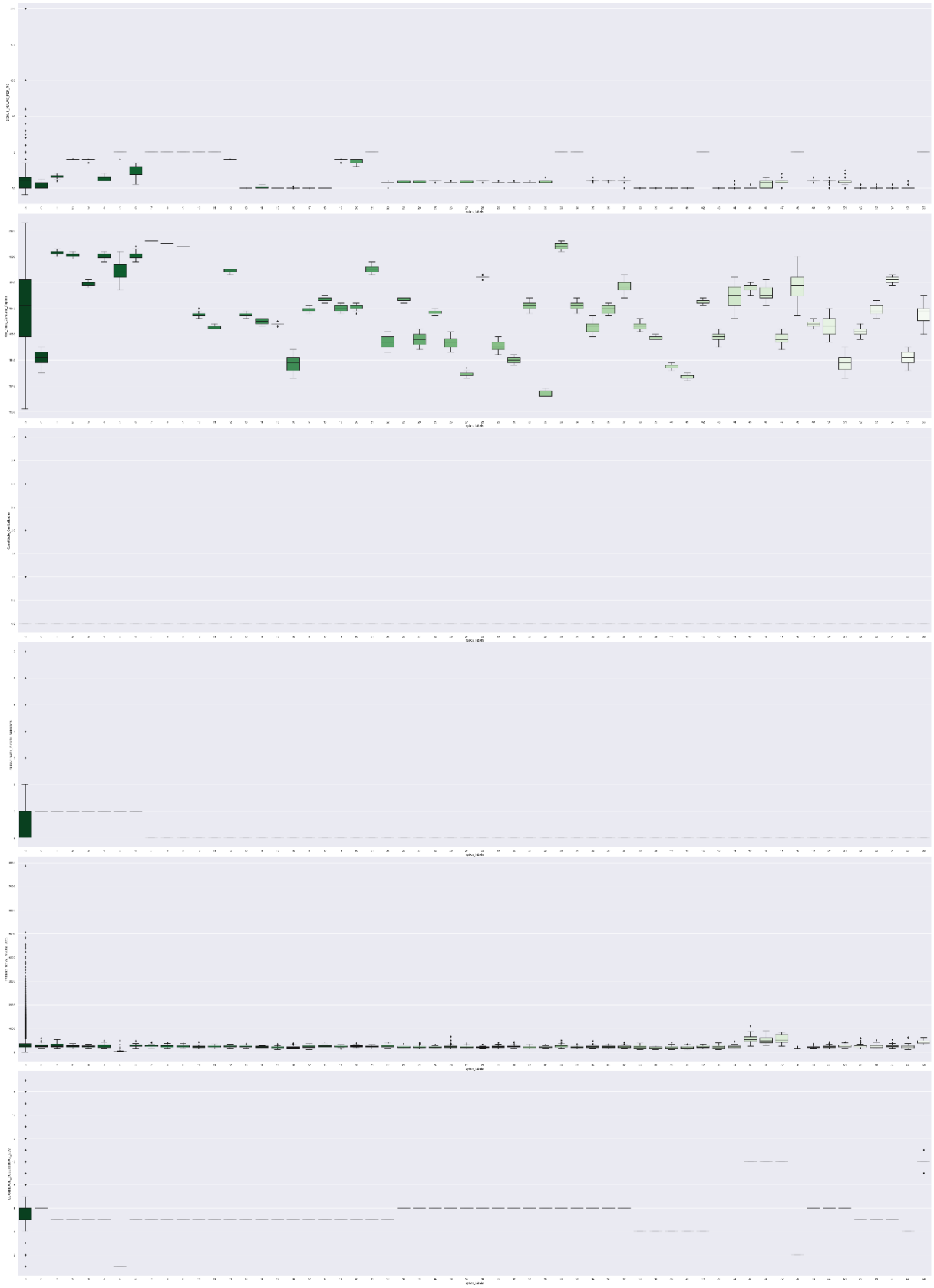


Figure 1 – Clients' variables distribution per cluster through the DBSCAN algorithm

For the OPTICS algorithm, Figure 2 shows the generated boxplots.





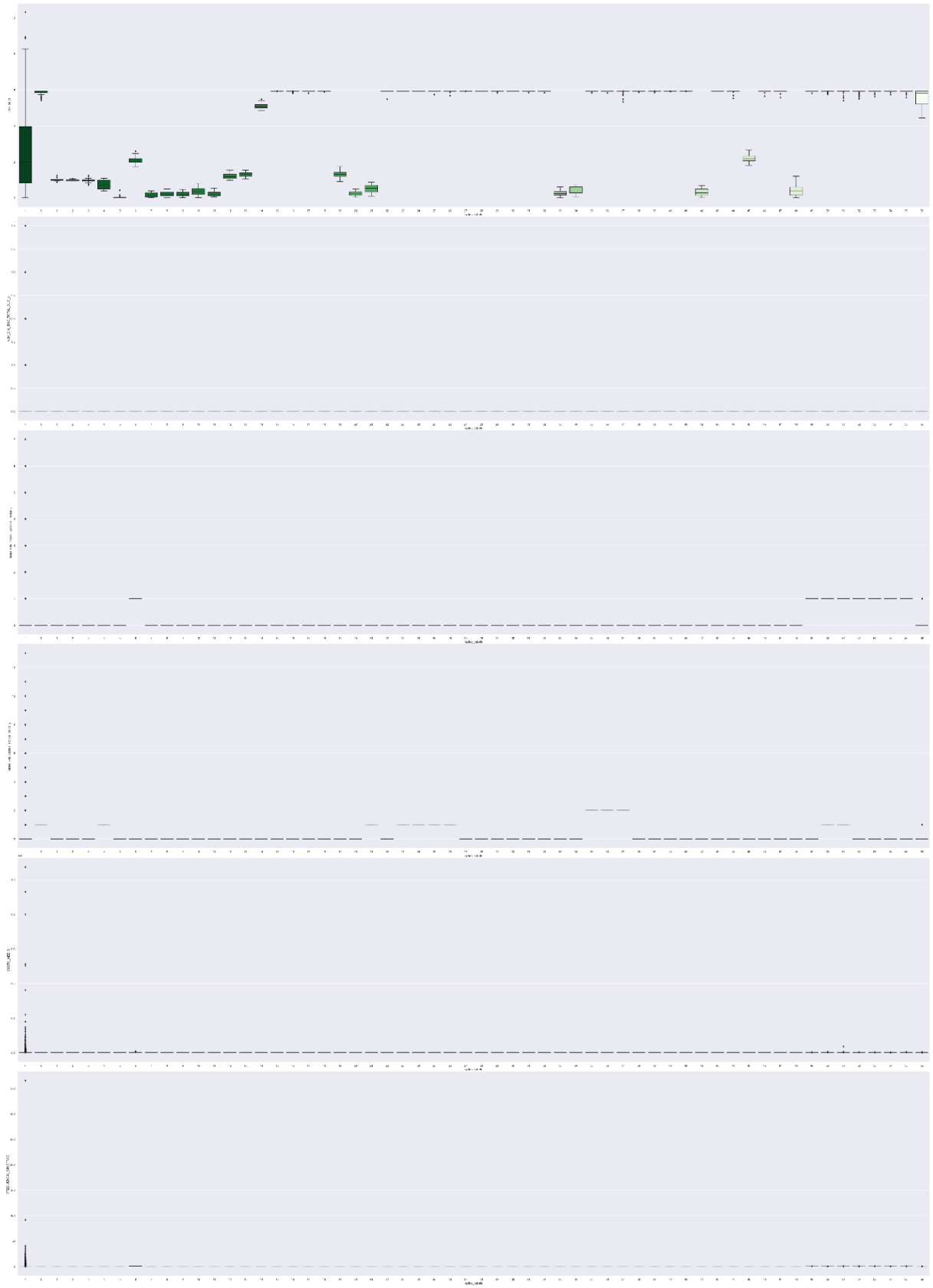
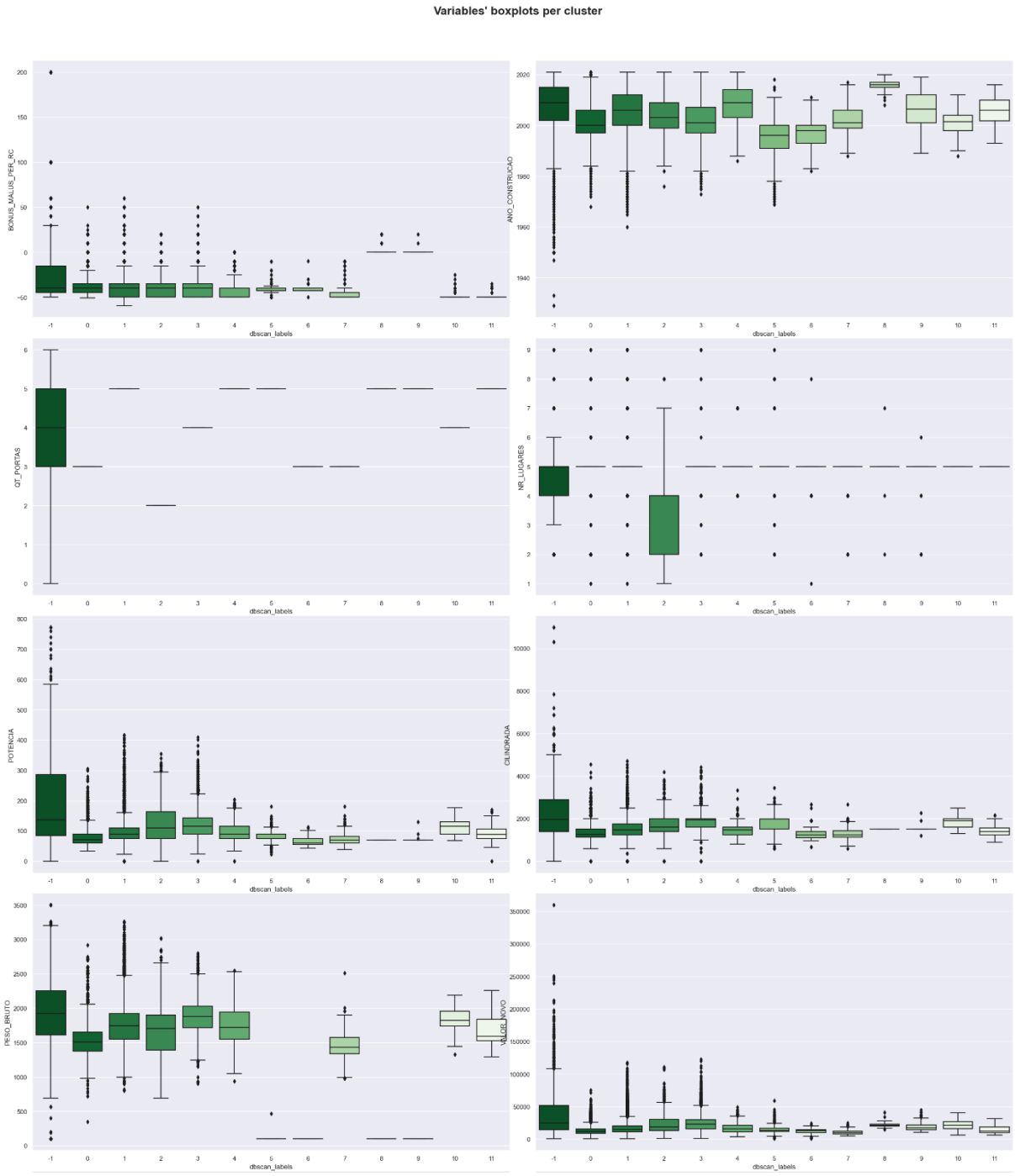


Figure 2 – Clients' variables distribution per cluster through the OPTICS algorithm

8.2. APPENDIX B – VEHICLE SEGMENT

This appendix shows the boxplots related to each cluster and metric feature for DBSCAN and OPTICS, regarding only the vehicle segment.

For the DBSCAN algorithm, Figure 1 presents the respective boxplots.



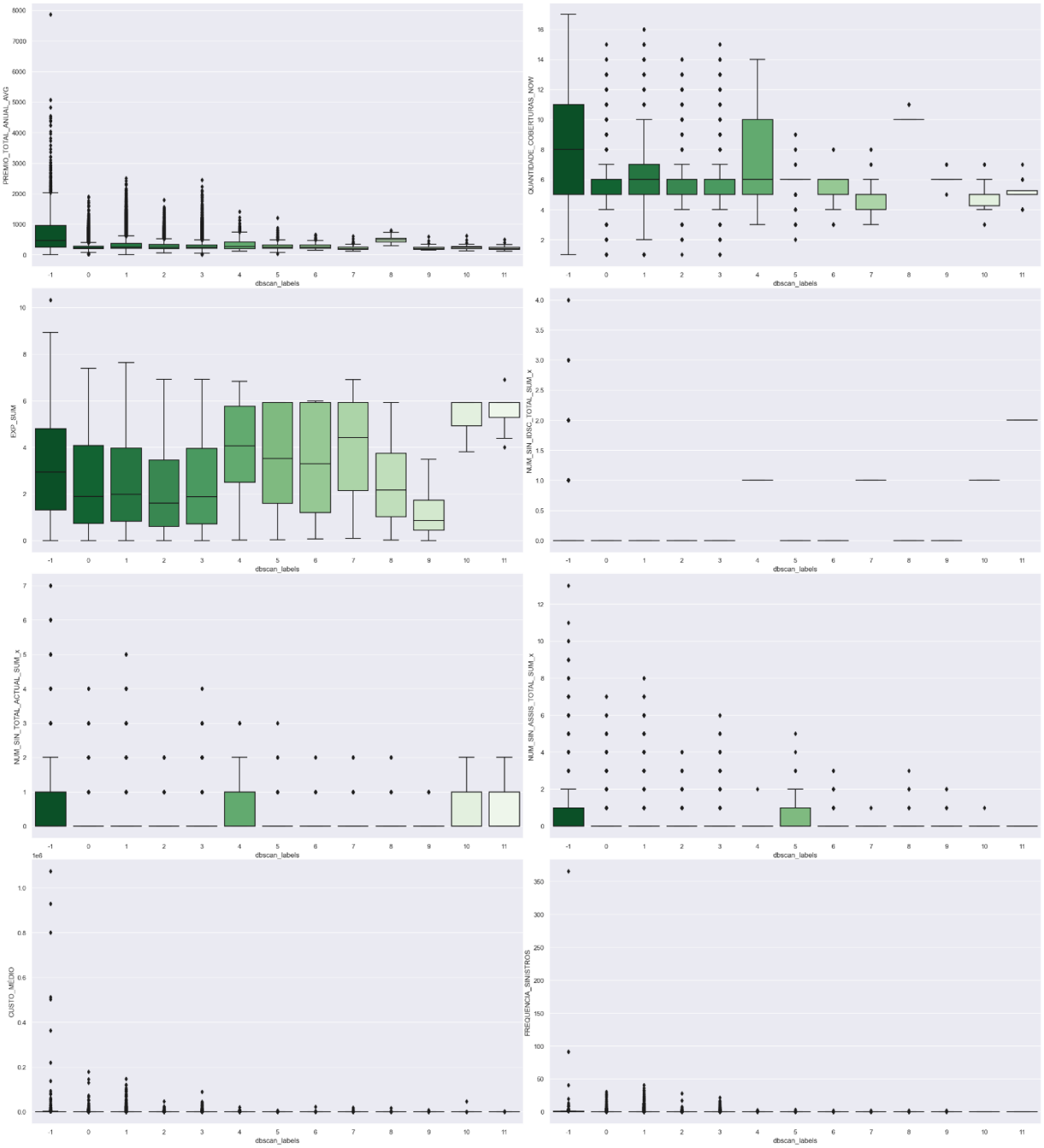
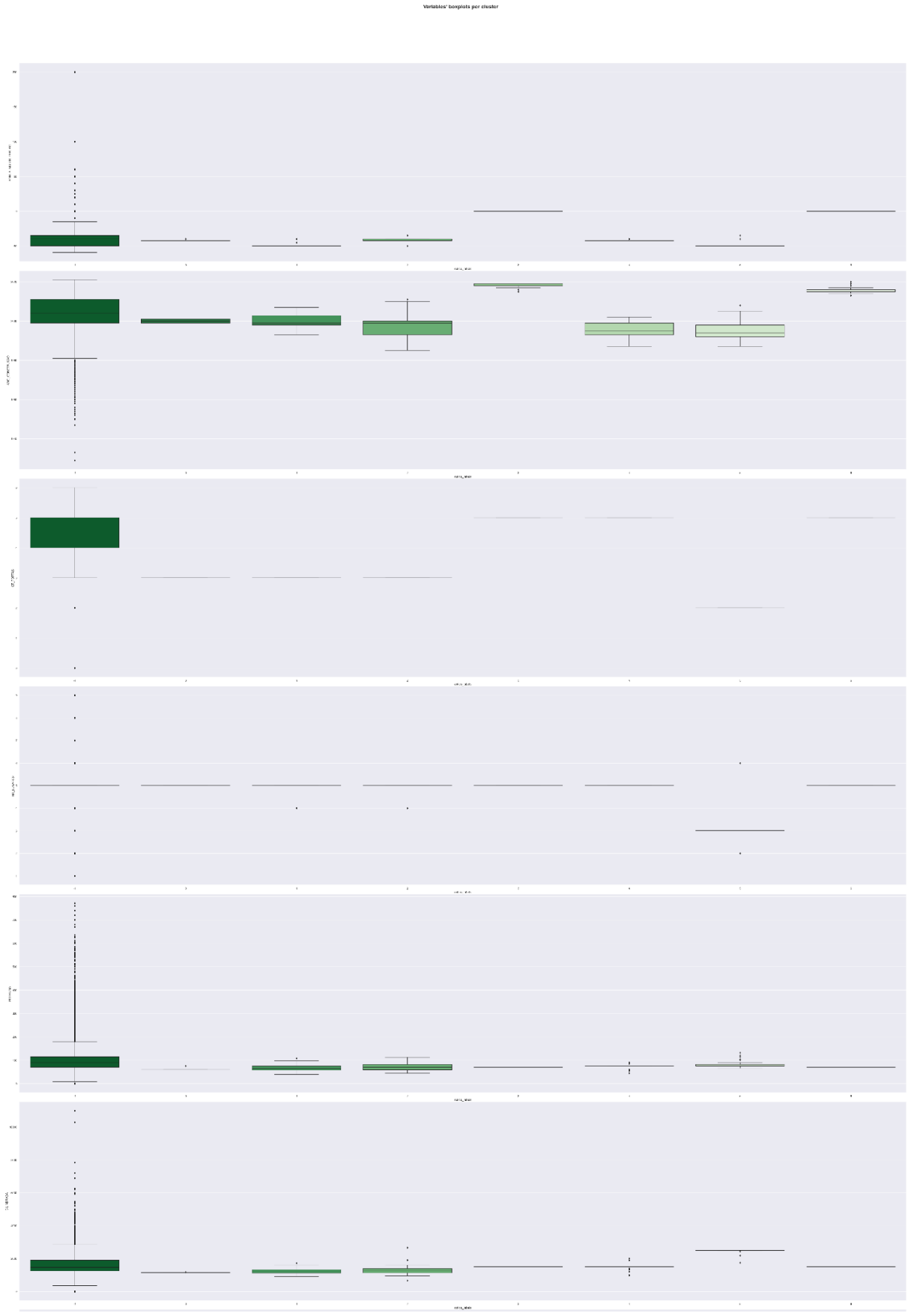
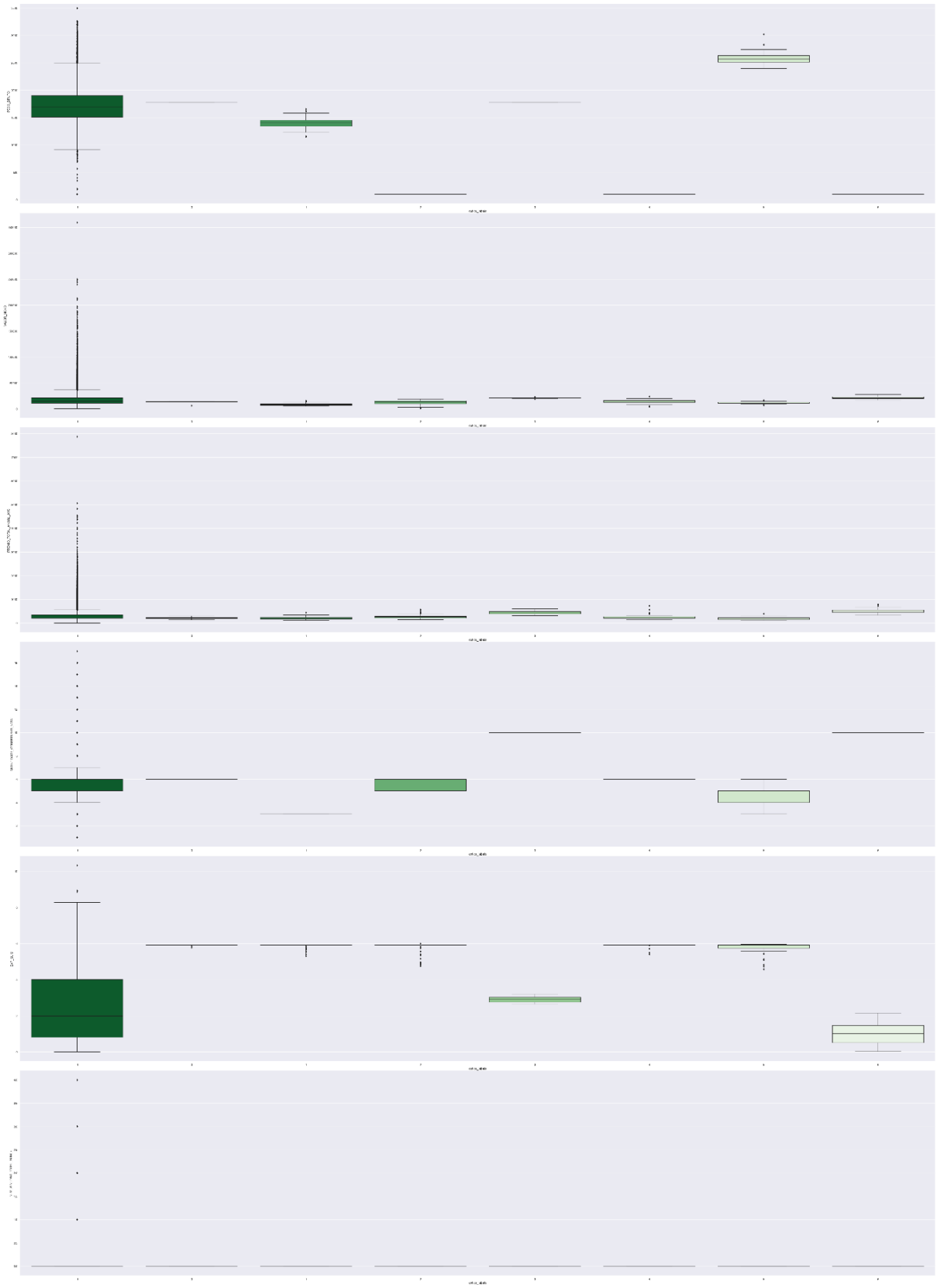


Figure 1 – Vehicles' variables distribution per cluster through the DBSCAN algorithm

For the OPTICS algorithm, Figure 2 shows the generated boxplots.





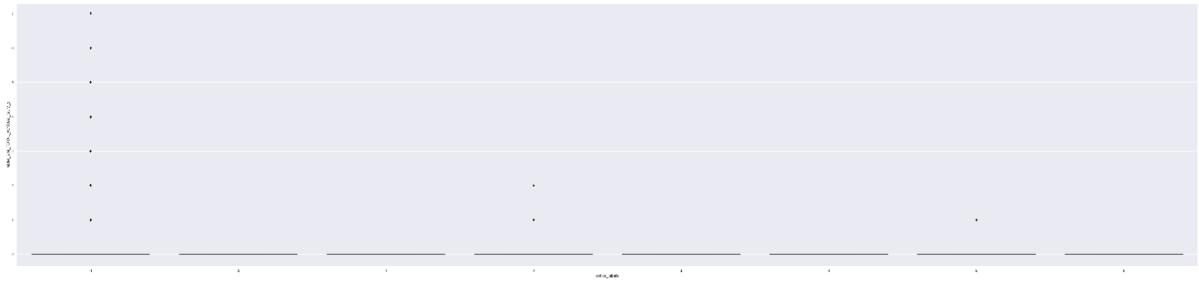
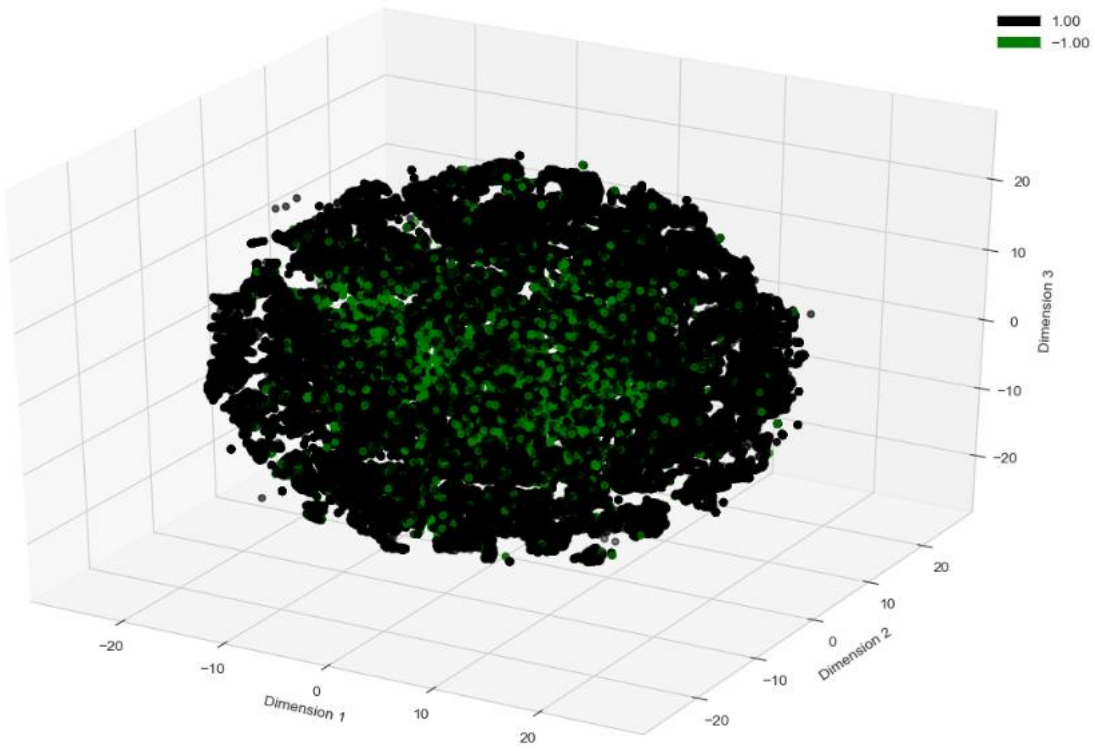


Figure 2 – Vehicles' variables distribution per cluster through the OPTICS algorithm

8.3. APPENDIX C – T-SNE 3-D VISUALISATION



9. ANNEXES

9.1. ANNEX A – DATAROBOT AUC SCORES

One of the tools used to evaluate the obtained models was the DataRobot platform, an AI leader that enables the delivery of AI to production for every organisation. As shown in Figure 1, several IF and LOF models were performed for distinct segments, which is explicitly emphasised in the column "Feature List & Sample Size". On a side note, the Informative Features set comprehends the client and vehicle segments. Then for each model, the AUC score is calculated.

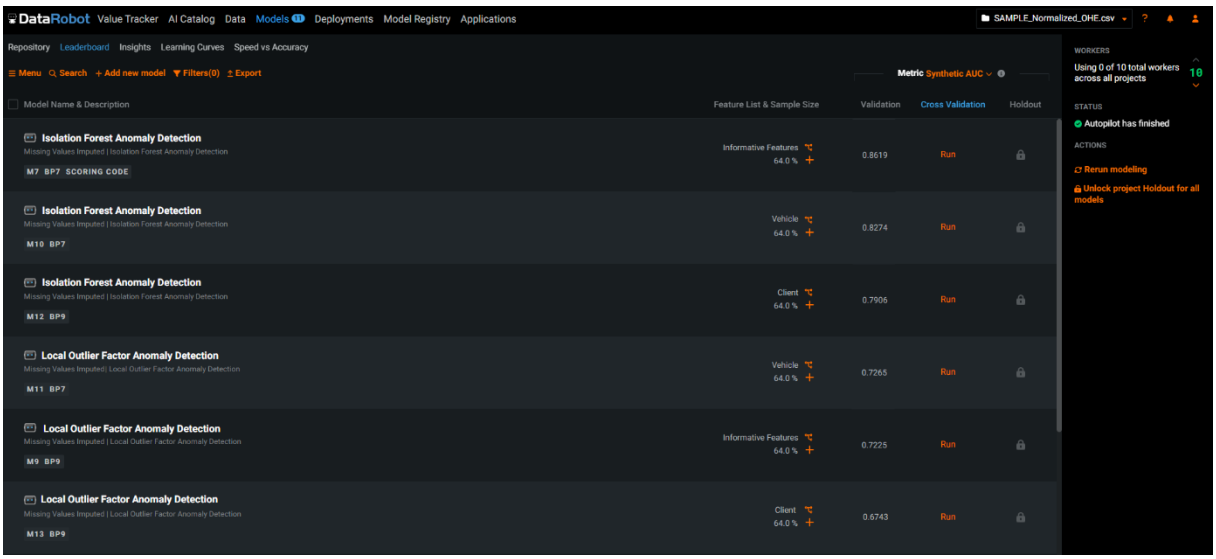


Figure 1 – Calculation of AUC scores through the DataRobot platform

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa