Interpretable Unsupervised Fraud Detection in Financial Services

Michalopoulos, Panagiotis

Award date:
2020

Master Thesis

Department of Mathematics and Computer Science
Data Mining Research Group

Interpretable Unsupervised Fraud Detection in Financial Services

Panagiotis Michalopoulos

Academic Supervisor:
prof.dr. Mykola Pechenizkiy

Version 1.0

Eindhoven, September 29, 2020

[This page intentionally left blank]

Abstract

AI systems demonstrate terrific potential. Yet, few FIs have employed AI in their anti-fraud efforts - only 5.5% of them have adopted real AI systems. In an attempt to reduce fraud incidences, it becomes urgent to leverage AI systems that can discern whether given financial activities seem suspicious or not, and alert fraud analysts to take immediate action through predefined workflows. In this thesis, we expose the ML advancements in the industrial world via studying the applicability of the Isolation Forests on detecting suspicious credit applications. Unsupervised learning unlocks the potential of supervised learning the moment we determine a threshold that gives us a rough estimation of whether an application is considered fraudulent or not. Here, we incorporate a small fraction of labeled data to set the threshold to the anomaly score that maximizes the AUC-PR, which is a suitable metric in fraud scenarios.

Modeling a region that captures normality can be extremely difficult, especially in financial data applications where the boundaries between normal and fraudulent behavior are usually blurred. The Isolation Forest differs from the existing model-based anomaly detection methods as it isolates anomalies explicitly, without profiling normal data points. It is optimized to detect anomalies, reduce the number of false positives, and deal with high computational complexity. Furthermore, it has linear time complexity with a low constant and memory requirement while utilizes no distance or density measures to detect anomalies, which eliminates the substantial computational cost of popular distance and density-based methods.

Extended experimentation with both the standard and the extended version of the Isolation Forest shows promising results when applied to segmented data. Financial applications are characterized by high heterogeneity that leads to high complexity and makes it tough for ML algorithms to parse signals from noise. By applying segmentation, we achieve the level of homogeneity to assume that outlying behavior characterizes suspicious activity. Despite that fraudulent and legitimate instances remain non clearly separable, the noticeable enhancement cannot be disregarded, especially in S3.

Industrial datasets often consist of high-dimensional feature spaces that are difficult to inspect. Being an outlier does not necessarily imply that a particular application is fraudulent. Thus, it is crucial to be able not only to evaluate an instance given its anomaly score but also to understand the drivers behind the model decision. The involved business experts acknowledge that the SHAP explanations increase task effectiveness on application processing compared to the model's anomaly scores alone, as they enhance the model credibility by providing insights on the features that contribute to the prediction. Furthermore, SHAP values contributed to task efficiency by enabling domain experts to process applications with a large number of features faster. The high-dimensionality of the working dataset also affects mental efficiency. The business experts stated that the provided SHAP explanations guided them to focus on a subset of features and assisted them in interpreting the model prediction and assessing the rationale of the decision.

Keywords: Credit applications, Anomaly Detection, Isolation Forest, Interpretable Machine Learning, SHAP

Preface

This thesis is the result of the master graduation project for Data Science Engineering at the Eindhoven University of Technology (TU/e). The research has been conducted within the Data Mining Group of the TU/e in collaboration with a Financial Institution. For privacy reasons, we will refer to the institution as PGS. The name is fictional; therefore, any resemblance to actual companies is purely coincidental.

PGS is committed to ensuring data is secure. To prevent unauthorized access or disclosure, PGS has technical and organizational measures to safeguard and secure the data. All PGS personnel and third parties are obliged to respect the confidentiality of the data. The thesis does not include any information concerning persons and attributes. The displayed results, graphs, and figures only present high-level insights into the applied methods.

I would like to thank my supervisor at PGS for the help and guidance since the very first day of this project. I would also like to thank professor Mykola Pechenizkiy for the given opportunity to work on a challenging yet exciting data science area. In this long journey until graduation, which included difficulties, hard decisions, and a pandemic, nothing would have been possible without the unconditional support of my family and friends.

Contents

Acronyms

AI Artificial Intelligence.

AUC Area Under the Curve.

BST Binary Search Tree.

DL Deep Learning.

FI Financial Institution.

iForest Isolation Forest.

iTree Isolation Tree.

LOF Local Outlier Factor.

ML Machine Learning.

NN Neural Network.

OOD Out-of-Distribution.

PR Precision-Recall.

ROC Receiver Operating Characteristic.

XAI Explainable Artificial Intelligence.

Glossary

Credit Application Documentation that is completed by an individual or business seeking to apply for a line of credit with a lending institution. The information on the application is used to determine the borrower's credit history, employment status, and his/her ability to repay the loan amount..

Debtor A debtor is someone who owes money.

End-user End User means any person or entity (e.g., contractor, contractor's employee, business associate, subcontractor, other downstream user, etc.) that uses the Financed Items pursuant to an End-User Contract.

Financial Institution A financial institution (FI) is a company engaged in the business of dealing with financial and monetary transactions such as deposits, loans, investments, and currency exchange.

Vendor Financing Vendor financing is a financial term that describes the lending of money by a vendor to a customer who uses that capital to purchase that specific vendor's product or service offerings.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Fraud in Financial Services

Financial Institutions process Personal Data to ensure the security of the company and their partners, and, eventually, the security of the financial sector. They aim at preventing fraud, money laundering, and the financing of terrorism.

Fraud is any irregular or illegal activities carried out for intentional deception and results in damage to PGS or its members, either financially, reputationally, or some other tangible way. It can be either internal or external. Internal fraud is a type of fraud, perpetrated by, or in collusion with, a member of PGS, when, for example, an agent exploited his/her position at PGS for personal gain. On the other hand, external fraud is perpetrated by third parties against PGS.

Figure 1.1: Fraud Risks in Financial Industry.

Fraud tactics range from personal identity theft to phishing emails or fictitious accounts. Financial systems are interconnected - linking partners, customers, service providers, and banks - and criminals attack the weakest link to infiltrate systems and make fraudulent purchases or claims. Figure 1.1 illustrates a host of crimes we usually encounter in financial services.

1.2 AI Adoption in Financial Industry

According to the AI Innovation Playbook, a recent survey published by PYMNTS in collaboration with Brighterion, 63.6% of fraud specialists believe AI is an effective tool for stopping fraud before it happens. In the meantime, 80% stated they have seen AI-based platforms reduce false positives, payment fraud, and prevent fraud attempts. The entirely new knowledge gained from unsupervised machine learning algorithms, coupled with the ability of supervised machine learning to interpret trend-based insights, is reducing payment fraud and propel more and more financial institutions (FIs) to AI.

AI systems demonstrate terrific potential. They can process large volumes of data in real-time and learn fast to identify suspicious financial activity. Yet, few FIs have employed AI in their anti-fraud efforts - only 5.5% of them have adopted real AI systems. Instead, FIs' anti-fraud departments continue using less-sophisticated learning systems, like data mining and business rule management systems (BRMS). This translates to only 12.5% of fraud specialists using AI, while 92.5% use data mining, and 65% use BRMS.

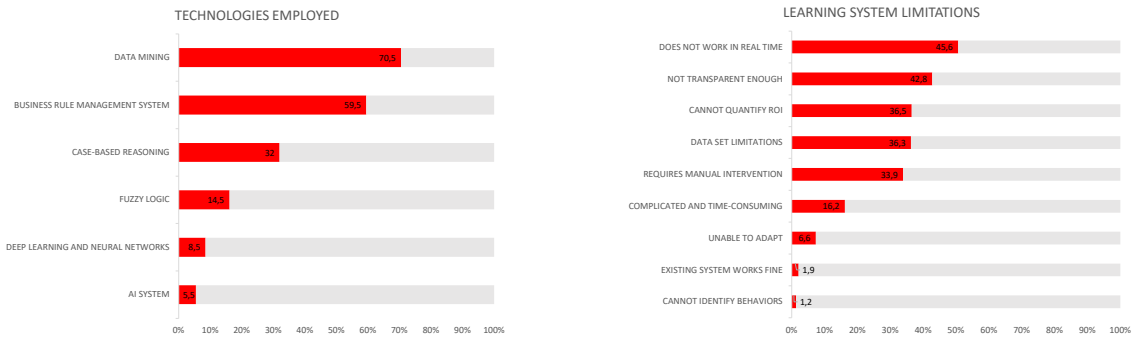


Figure 1.2: As of 2019, Data Mining remained the most employed technology by FIs, while AI systems show slow adoption due to their perceived limitations. Working in real-time, addressing transparency issues, and quantifying ROI, are the most urgent shortcomings to overcome to achieve greater adoption (Source: AI Innovation Playbook).

The black-box nature of AI systems explains the slow adoption at a business level: Decision-makers do not always understand why AI systems make the decisions they do. In fact, 60% of bank fraud specialists believe AI technology is not transparent enough, and the same portion views it as complicated and time-consuming. This speaks not only to their misunderstanding of how AI systems make decisions but to the fact that many lack the expertise to understand how to use them.

The financial industry faces unique security challenges. Not only is it a target due to direct money involvement, but the number of transactions along with potential targets — clients, dealers, and financial institutions - makes security more complex and critical.

FIs suffer millions in losses due to Fraud incidences annually. Thus, it becomes urgent the need to leverage AI systems that can discern whether a series of financial activities are suspicious or not, and alert fraud analysts to take immediate action through developed workflows. Payment fraud schemes are growing in complexity, often incorporating a completely different digital footprint or sequence and structure, which makes it extremely difficult for ruled-based logic to handle. By employing AI, FIs will be in a position to spot anomalies in their large-scale datasets in seconds. The analysis of historical data is the key to understand patterns and create an integrated picture of customer behavior.

The use of AI and particularly machine learning to detect and prevent fraud, is gaining traction, with several financial institutions starting to deploy these technologies as a direct response to increasingly sophisticated cybercrime. Despite recent studies show that the rate of AI and machine

learning deployment as a fraud management measure remains relatively low, we expect it to rise significantly in the next 2-3 years. Traditional rule-based approaches, which focus on identifying potential fraud based on pre-defined rules, are getting less effective due to the evolution and complexity of modern fraud. Fraudsters' attacks nowadays employ distributed networks, big data, and the dark web to detect vulnerabilities. They also mimic good customer behavior to delude the system. Rule-based protection methods are slow-learning and can't cope up with the velocity of transactions. A fully dynamic system compels financial services providers to look beyond traditional 'one size fits all' rule-based methods and build a hybrid system, combining a range of AI and machine learning-based approaches.

Machine learning improves accuracy once it has access to a large volume of data to learn from. In fraud detection, millions or even billions of data points enable the machines to build a comprehensive understanding to distinguish between illicit and legitimate behavior. The evolving nature of the crime requires machines to encounter as many examples as possible to become effective. While the initial data input is important, state-of-the-art models use adaptive technologies that continually learn from any additional input data so decisions can be adjusted based on current environments.

Despite recent attempts, we have a long way to go before fraud prevention systems become commonplace. The world's financial systems are not yet broadly connected and tend to function in isolation, making it difficult to identify patterns across a bigger ecosystem. AI offers an opportunity to connect the dots, and if executed properly, the ability to save a significant amount of time and money for organizations, while also minimizing potential reputational damage as a result of fraud-related incidents. It is crucial to ensure the right people and partners are on hand to assist in navigating the maze to develop a complex and multi-faceted but yet effective strategy.

1.3 Digital Twins

According to Bolton et al.[8], a digital twin is a dynamic virtual representation of a physical object or system across its lifecycle, using real-time data to enable understanding, learning, and reasoning. It has been adopted widely in solving actual industrial problems, as it allows us to tune the current conditions and predict the future states of an industrial system. The success of Digital Twins relies on domain knowledge, probabilistic inference, and deep learning, the three pillars depicted in figure 1.3.

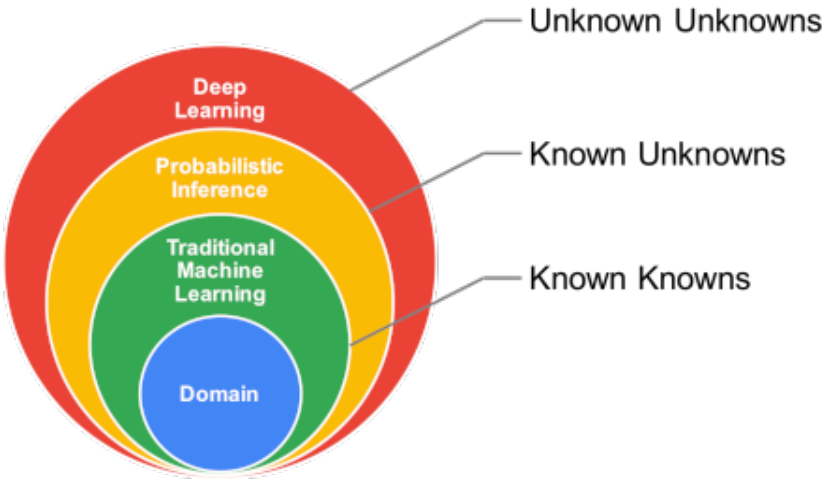


Figure 1.3: The three pillars to build Digital Twins: Domain Knowledge, Probabilistic Inference, and Deep Learning (source: TensorFlow Blog).

- ***Known Knowns*** refers to problems whose solution involves a blend of domain expertise and traditional ML. Traditional Machine Learning consists of any method or algorithm that has been deployed and used over the years, including clustering, classification, and regression. Predicting a fraud case under various behavioral patterns is an example of domain expertise combined with traditional ML: it requires a thorough understanding of how fraudsters behave under certain conditions and a set of ML methods to compute problem-specific coefficients.
- ***Known Unknowns*** refers to systematic ways to quantify uncertainty. Probabilistic inference constitutes the most effective way to model such uncertainty, besides understanding the phenomena. In fraud detection, we tend to study behavioral patterns, but we also need to account for un-modeled factors such as the integrity of the system or the malfunction of a credit card.
- ***Unknown Unknowns*** refers to unknown patterns and behavior that advanced Machine Learning has the potential to identify and predict. In fraud detection, with probabilistic inference, we could predict a fraudulent attempt only under a particular set of conditions with known uncertainties. On the contrary, a deep learning model for anomaly detection, such as an autoencoder, is capable of catching anomalous conditions that the hard-rules based model would not pick up. Generative models' characteristic to recreate the input data - alternatively, the unlikelihood to generate previously non observed data points - can be leveraged to spot normality deviations when they are trained on a large volume of anomaly-free data. Numerous studies demonstrate the ability of advanced ML to shine on problems where anomalies are not known a priori.

1.4 Problem Description

The motivation behind the master thesis is the design and the implementation of a system capable of identifying suspicious patterns and detecting fraudulent behavior. In general, anomalies are distinct data points in the dataset that do not conform to a well-defined notion of normality. However, the straightforward approach to model normality and classify anything that lies outside the defined boundaries as an anomaly faces various challenges.

- Modeling a region that captures normality can be extremely difficult, especially in financial data applications where the boundaries between normal and fraudulent behavior are usually blurred. Malicious activities tend to adapt rapidly to system requirements and appear normal at first sight.
- The notion of normality changes continually, as what we consider normal today might not be valid in the future. A powerful application should be able to generalize and adapt to the applied changes.
- Gathering labeled data of abnormal behavior is a primary challenge as it is often expensive and time-consuming. In most cases, we have access to an abundance of normal observations, but it is tough to gather enough data on abnormal behavior to train a good model.

We could reinforce supervised machine learning models by leveraging the benefits of semi-supervised learning space. Semi-supervised is a concept that leverages unsupervised machine learning methods to learn some notion of normality from the abundance of normal cases and then, it incorporates a small fraction of labeled data to fine-tune the resulted perception. Unsupervised learning could unlock the potential of supervised learning the moment we manage to define a threshold that gives us a rough estimation of whether a transaction is considered fraudulent or not.

The current work will try to answer the following **research questions**:

1. *How can financial institutions leverage unsupervised learning to uncover the evolving nature of financial crime in vast amounts of incoming credit application data?*

2. *Can we expect financial institutions to rely more on findings provided by an unsupervised model to develop preventive anti-fraud systems?*
3. *How Isolation Forests perform in the given setup?*
4. *How would interpretable machine learning contribute to the adoption of such systems?*

1.5 Dataset

The current thesis has been conducted in collaboration with PGS as part of a fraud detection project. The working dataset contains numerous independent variables that describe the characteristics and the behavior associated with each unique credit application as of the moment of entry into PGS's system. The feature dataset contains 441.757 observations, uniquely identified by an application key, and a total of 2.375 variables.

1.6 Challenges

The current advancements in AI create a momentum for the anti-crime domain. However, there are still challenges we must address before such applications gain widespread use. A few of the most critical hurdles to consider are listed below[9][10].

- **Data Volume**. The performance of many AI systems relies on large amounts of data, proportional to the number of parameters of the applied architecture. Gathering such an amount of data to train machine learning models effectively can be challenging both in terms of cost and availability. Credit data are subject to privacy constraints that make them hardly accessible for research.
- **Data Quality**. Fraudulent cases usually represent only a very small fraction of the total transactions, therefore fraud detection systems should be able to handle extremely imbalanced datasets with skewed distributions.
- **Class Imbalance**. Credit data are characterized by high heterogeneity that leads to high complexity and makes it tough for ML algorithms to parses signal from noise.
- **Overlapping Data**. Modeling a region that captures normality can be extremely difficult, especially in financial applications where the boundaries between normal and fraudulent behavior are usually blurred. Malicious activities tend to adapt rapidly to system requirements and appear normal at first sight.
- **Generalizability**. The questioned ability of ML models to generalize beyond the training set is another notable concern that follows such applications. Machine learning algorithms tend to overfit to the statistical characteristics of the training data, making them hyperspecialized for particular purposes and less efficient on the population at-large. Furthermore, the notion of normality changes continually, as what we consider normal today might not be valid in the future. Robust systems should be able to generalize and adapt to the applied changes.
- **Interpretability**. The concept of interpretability of machine learning algorithms has been in the spotlight the recent years. ML has grown to a powerful tool in problem-solving, yet mapping complex, nonlinear functions is difficult to interpret. In fraud, the ability to identify drivers of outcomes is critical to convince domain professionals to rely on the detections of such systems.
- **Ethical**. Special care should be taken into consideration when dealing with underrepresented populations. The tendency of ML algorithms to fit the characteristics of the training set

could lead to the incidental introduction of bias and inequities against specific individuals or entities[11].

1.7 Technical Approach

Unsupervised learning offers a variety of methods and algorithms to tackle different problems. The current state-of-the-art methods lack the maturity to handle industry-level applications with ease, despite their potential to cope up with the increasing complexity and high-dimensional inputs. The ongoing thesis aims to leverage machine learning advancements and study the suitability of Isolation Forests for industrial applications. In the current structure, we provide an overview of anomaly detection techniques for completeness, and then we focus our study on Isolation Forests.

The concept of Isolation Forests differs from other popular outlier detection methods as it targets explicitly anomalies rather than profiling normal data-points. It has its basis on decision trees, where partitions are created by first randomly selecting a feature and then selecting a random split between the minimum and maximum value of that feature. In principle, by using such random partitioning, outliers should be identified closer to the root of the tree, with fewer splits necessary, as they are less frequent compared to regular observations, and they lie further away in the feature space.

Despite the advancements in Machine Learning and the recent efforts of the scientific community towards explainable models, yet many are considered black boxes. Unlike their tremendous potential, advanced machine learning models are still far away from mass adoption, especially in industrial domains such as finance, where the justification of any decision plays a crucial role. The ability to explain your results increases the trustworthiness of your system, which is often preferable to the detriment of pure performance.

The black box accountability is associated with supervised learning due to the necessity to understand predictions, but it scales up to cover most machine learning problems. Similar questions arise when an unsupervised model flags data points as outliers, and data scientists seek the reasoning behind each decision. In fraud, the abundance of variables that accompany a credit application may contribute to a powerful detector; however, it remains a question of how to quantify that contribution and evaluate whether humans influence the decision of the model.

Industrial datasets often consist of high-dimensional feature spaces that are difficult to inspect. Being an outlier does not necessarily imply that a particular application is fraudulent. Thus, it is crucial to be able not only to evaluate an instance given its anomaly score but also to understand the drivers behind the model decision. In this work, we introduce the SHAP explanations to business experts, and we seek to study whether their use increases task effectiveness, task efficiency, and mental efficiency during processing.

In Chapter 3, we discuss all related topics in more detail.

1.8 Thesis Outline

In this section, we describe the structure of the current work. The thesis is organized into six chapters. In Chapter 1, we introduce the concept of fraud in financial services, alongside an overview of the adoption of AI systems in industrial applications. The dataset description is followed by the research questions, the technical approach, and the challenges we have to overcome to achieve our goals.

We aim to study the applicability of Isolation Forests for anomaly detection in high-dimensional credit application data. The concept of Isolation Forests differs from other popular outlier detection

methods (Chapter 2) as it targets explicitly anomalies rather than profiling normal data-points. In addition to the standard analysis, we experiment for the first time in a large-scale application with the extended version of the algorithm (Chapter 3). The extended version aims to enhance the robustness and reliability of the anomaly scoring.

In Chapter 4, we present the results of experimenting with both versions of the Isolation Forest before and after applying data segmentation. For data segmentation, we employ domain knowledge in an attempt to ensure the homogeneity required to identify suspicious financial applications using anomaly detection techniques. For the performance evaluation, we leverage the few available labels to set the anomaly threshold to the score that maximizes the AUC-PR. In the same Chapter, we also introduce the required preprocessing steps to create features that translate different anomaly types into point anomalies, which can be captured from our models. Ultimately, the use of Isolation Forests coupled with data segmentation shows promising results in the current setup.

Industrial datasets often consist of high-dimensional feature spaces that are difficult to inspect. Being an outlier does not necessarily imply that a particular application is fraudulent. Thus, it is essential to be able not only to evaluate an instance given its anomaly score but also to understand the drivers behind the model decision. In Chapter 2, we provide an overview of different interpretable machine learning techniques, and in Chapter 3, we introduce in detail the SHAP value estimation methods. Essentially SHAP aims for reporting machine learning interpretability and reporting model predictions through an improved alignment with human intuition. In chapter 4, we discuss the feedback provided by the business experts on the utility of such explanations in terms of task effectiveness, task efficiency, and mental efficiency. Ultimately, the business experts stated that the proposed methods boost the user's trust in machine learning.

In Chapter 5, we conclude our analysis with the contribution of the current work, the limitations, and future steps.

Chapter 2

Related Work

2.1 Types of Machine Learning

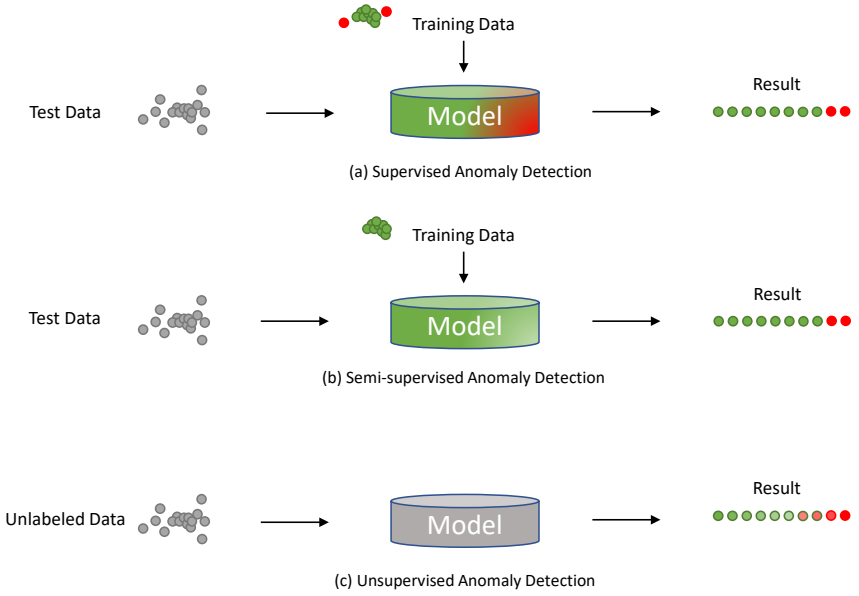


Figure 2.1: The different anomaly detection modes are associated with label availability in the dataset. (a) supervised learning leverages a fully labeled dataset to classify anomalies. (b) semi-supervised learning uses an anomaly-free dataset to model normal behavior and identify anomalies as deviations. (c) Unsupervised learning discovers similarities between items based on their features, therefore it can be used to detect anomalies as isolated observations[1].

In contrast to the widely-explored supervised problems, where input vectors along with their corresponding target are used to train a classifier and test data measures performance afterward, there are multiple ways to describe possible anomaly detection setups[1]. In particular, the choice between alternative anomaly detection setups relies on the labels available in the dataset. We can categorize the possible setups in three main types as depicted in Figure 2.1:

1. Supervised Anomaly Detection refers to tasks that consist of fully labeled datasets. Models are fit on training data comprised of inputs and outputs and used to make predictions on test sets where the target is not available. This scenario only differs from traditional pattern recognition on the fact that the classes are typically extremely imbalanced. In a typical credit card fraud detection problem, the fraudulent transactions comprise only 0.01% of the total,

hence special treatment is often required. However, this setup assumes that anomalies are known in advance and labeled accordingly when in many real-case scenarios occur as novelties in the test phase.

2. Semi-supervised Anomaly Detection works similarly to supervised, except the training set only consists of normal data points without any anomalies. This method, also known as one-class classification, models the normal behavior and classifies as anomaly any instance that deviates from that. One-class SVM[12] and Autoencoders[13] are the trending algorithms for this setup.

3. Unsupervised Anomaly Detection operates upon only the input data without target variables available, which makes it more flexible compared to the previous setups. Most unsupervised models, such as the trending isolation forests, process the data solely based on their intrinsic properties and assign them a score that estimates what is normal and what is an outlier.

2.2 Anomaly Types in Literature

Before we move on, it is essential to define the different types of anomalies we meet in the literature. Based on their unique characteristics, anomalies can be:

1. **Point Anomalies:** It refers to single data points that represent an irregularity or deviation that occurred randomly. Figure 2.2 demonstrates such an example, where the amount spent in Monaco Café deviates significantly from the rest of the transactions.

2. **Contextual or Conditional Anomalies:** It refers to anomalous behavior under a specific context. In Figure 2.3, the temperature at time t_1 and time t_2 might be the same, but t_1 occurs in a different context and, hence it is treated differently.

3. **Collective or Group Anomalies:** It refers to individual data points that appear normal, whereas exhibit unusual characteristics when observed collectively. A single transaction of "pcexperts.com" as illustrated in Figure 2.2, might not seem suspicious but the identical repetition candidates for collective or group anomaly.

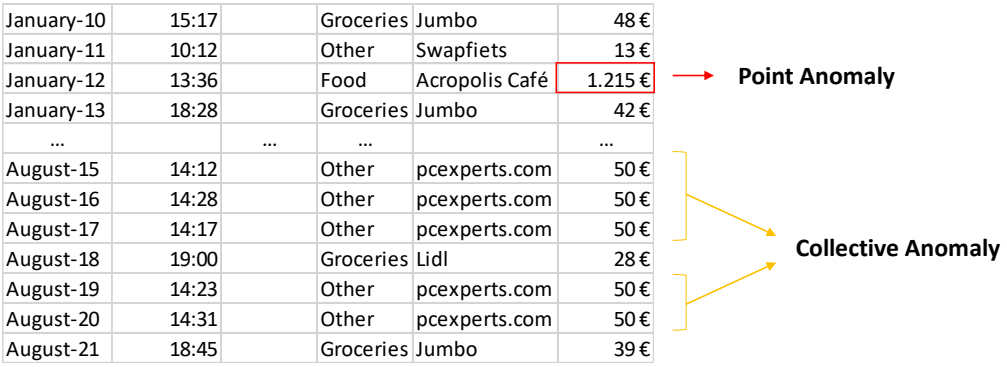


Figure 2.2: Illustration of point and collective anomalies in credit card fraud detection[2].

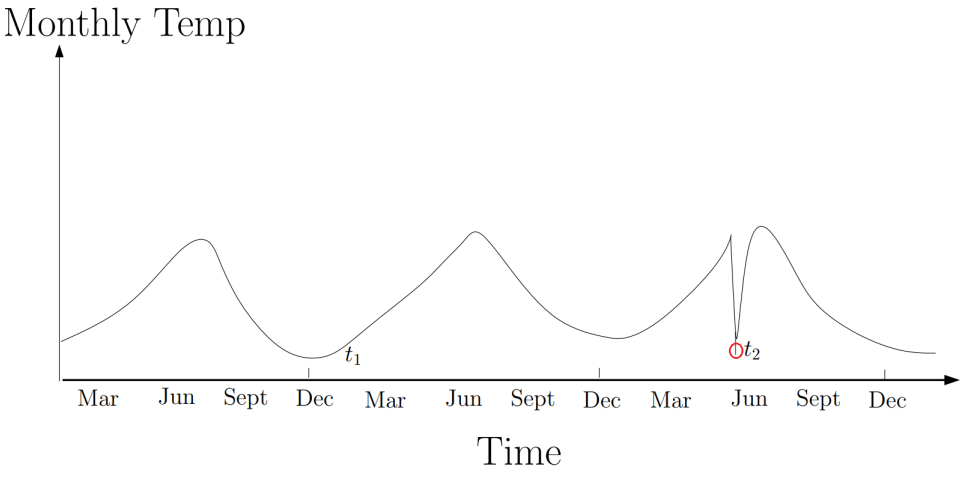


Figure 2.3: Illustration of contextual anomaly detection[2].

2.3 Anomaly Detection Methods

Over the years, many Anomaly Detection techniques have been employed for fraud detection and prevention. Fraud constitutes an adaptive crime, which often requires real-time systems in place. In the banking domain, user profiling used to be the favored approach to monitor user behavior and detect credit card fraud. However, maintaining billions of user-profiles is not scalable. Furthermore, traditional machine learning requires manual expertise to extract robust features that often fail to adapt to the evolving nature of fraud. This section summarizes the most commonly used Anomaly Detection techniques up-to-date, as presented in several surveys [2][3][10][14].

Adewumi et al.[10] contributed with a survey of machine learning and nature-inspired based credit card fraud detection techniques that financial institutions and individuals can use to develop anti-fraud systems. They claim that hybridized methods produce better results when fighting against conventional fraud like money laundering, computer intrusion, and credit card fraud. Hybridized methods can be either composition of multiple classifiers, blends of nature-inspired and machine learning algorithms, or just ensembles.

Hidden Markov Model (HMM) is the basis of several anti-fraud systems. Khan et al.[10] combined HMM with K-clustering to develop a method that calculates the probability of acceptance for a new transaction based on the spending behavior of the cardholder. Mhamane and Lobo[10] proposed a different HMM-based system composed of 10 modules that guarantee secure transactions.

Support Vector Machines (SVM) is another common technique in the fight against crime. Lu and Ju[10] combined PCA and Imbalanced Class Weight SVM (ICW-SVM) to develop a credit card fraud detection model able to handle data imbalance. They used PCA for selecting the features with the highest contribution rate, and ICW-SVM for classification.

Pun[10] designed a system to filter transactions from Falcon Fraud Manager used by the major Canadian banks. Three classifiers, k-nearest neighbor, decision tree, and naive Bayes, compose a **meta-learning** technique that reports an improvement of 24-34% compared to the existing solution. Stolfo et al.[10] proposed another meta-learning based system that aims at a distributed fraud detection system for financial institutions, as a secure way to share fraudulent models. Sen and Dash[10] experimented with **Classification and Regression Tree (CART)**, **AdaBoost**, **Bagging**, **LogitBoost**, and **Grading** to find that Bagging performed best in classifying fraudulent transactions.

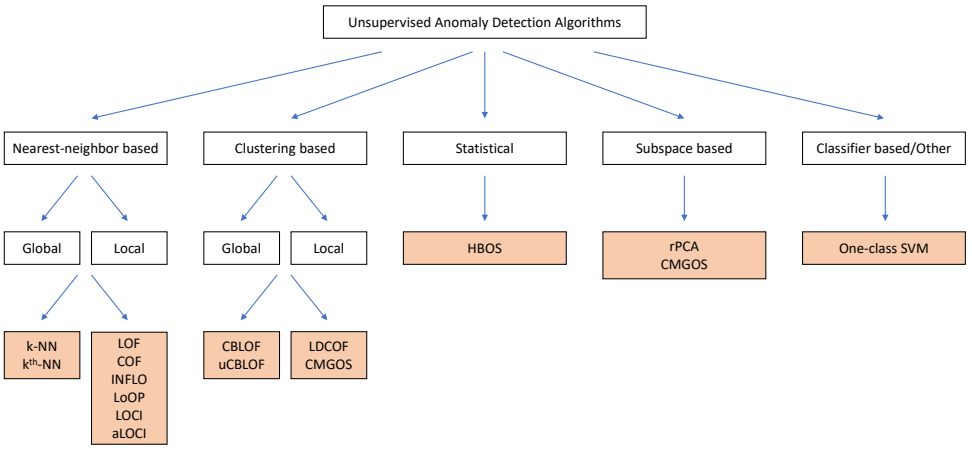


Figure 2.4: A taxonomy of unsupervised anomaly detection algorithms as organized in a survey conducted by Goldstein and Uchida[3].

In contrast to the survey of Adewumi et al., where most presented methods belong to the family of supervised machine learning, Goldstein and Uchida[3] focused on the evaluation of unsupervised anomaly detection algorithms for multivariate data. For the interested reader, their survey evaluates 19 different unsupervised algorithms (Figure 2.4) on 10 different datasets from multiple application domains.

In general, we mark a data point as anomalous either by using labels or a score. The label is a binary indicator stating whether a data point is anomalous or not, while the score is a continuous variable that describes the degree of abnormality. The **k-nearest neighbor** is a widely used algorithm for detecting global anomalies in literature. For every instance, we first determine the k-nearest neighbors, and then we calculate the anomaly score as the distance to the k^{th}-nearest-neighbor or the average distance to all k-nearest-neighbors. We often refer to the former as k^{th}-NN and the latter as k-NN.

Local Outlier Factor (LOF) is the most commonly used method for detecting local anomalies, based on the principles of k-NN, where the anomaly score is estimated as a ratio of local densities. Since the density of an instance equals the densities of its k-nearest neighbors, we expect normal data points to score around 1.0 and anomalies considerably higher. Despite the algorithm can also be used to detect global anomalies, experimentation has shown that generates many false alarms.

Connectivity-Based Outlier Factor (COF) only differs from LOF in the way it performs the density estimation. Here the local density of the neighborhood is estimated by employing a shortest-path approach, called the chaining distance. The chaining distance targets the shortcoming of assuming spherical distribution to features that are linearly correlated.

Influenced Outlierness (INFLO) is a method that aims to address cases where LOF fails to score correctly instances that lie in the borders of clusters with different densities. For that purpose, INFLO incorporates a reverse nearest neighborhood set along with k-NN.

As we have already seen, LOF produces an anomaly score that we can transform into a binary label, indicating whether a data point is normal or anomalous, by setting a threshold. However, it can be tricky to define such a threshold as we rarely know where the boundaries lie. **Local Outlier Probability (LoOP)** attempts to address this issue by producing an anomaly probability instead, as a result of normalization and a Gaussian error function.

All algorithms presented to this point are sensitive to the parameter k of k-NN. This deficiency is tackled by the **Local Correlation Integral (LOCI)** that leverages a maximization approach to determine the best k. Yet it comes at the price of increasing the computational complexity to $O(n^3)$, which makes it inefficient for large datasets. **Approximate Local Correlation Integral**

(aLOCI) comes to its rescue as it employs quadtrees to optimize the execution time. However, aLOCI receives criticism as the approximation often leads to poor performance.

Cluster-Based Local Outlier Factor (CBLOF) uses clustering to determine dense areas and afterward heuristics to classify the outcome into small and large clusters. The anomaly score results as the distance of each data point to its center cluster. The authors claim that if you multiple the computed distance to the number of cluster members, you get an estimation of the local clustering density. It is proven though that this assumption does not hold in practice, therefore exists a modified version of the algorithm, called **unweighted-Cluster-Based Local Outlier Factor (uCBLOF)**, that neglects the weighting. Whilst uCBLOF works out the controversial weighting, it is not a local anomaly detection method anymore. **Local Density Cluster-based Outlier Factor (LDCOF)** addresses this by dividing the instance distance to its cluster center by the average distance of all cluster members to the centroid.

Clustering-based Multivariate Gaussian Outlier Score (CMGOS) is another improvement of cluster-based detection methods. A multivariate Gaussian model is used to estimate the local density, while the anomaly score results from dividing the Mahalanobis distance of an instance to its nearest cluster center by the chi-squared distribution with a set confidence interval.

More methods include the **Histogram-based Outlier Score (HBOS)**, **One-Class Support Vector Machine**, and **Robust Principal Component Analysis (rPCA)**. HBOS is a statistical method, similar to Naive Bayes, that relies on histograms of independent variables to score data points. One-Class SVM is trained on anomaly-free data and classifies as anomalies the instances that contribute the less to the decision boundary. Last, rPCA leverages the principle components to determine global deviations through major components or local through minor components.

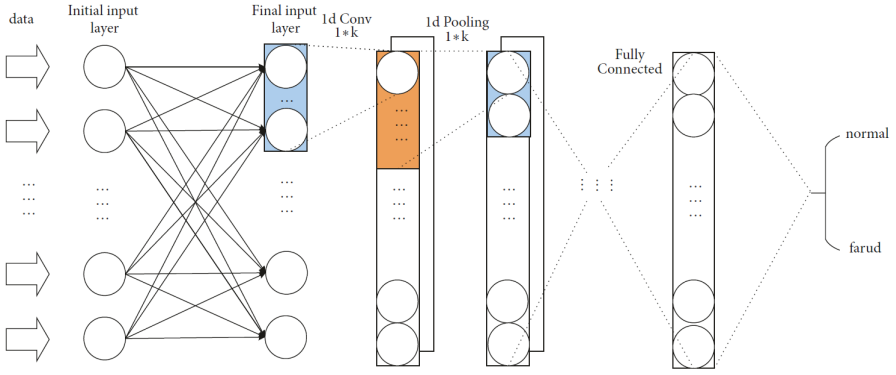


Figure 2.5: Overall network structure as presented in [4].

Deep Neural Networks (DNN) refer to deep architectures designed to overcome the limitations of traditional Machine Learning[2]. **Deep Belief Networks (DBN)** are class of deep neural networks, alternatively generative graphical models, composed of multiple connected layers of latent variables. Their structure consists of an unsupervised network, such as **Restricted Boltzmann Machines (RBM)** or Autoencoders, where each subnetwork's hidden layer serves as a visible layer to the next. DBNs fail to capture the characteristic variations of anomalous samples, which makes them suitable for anomaly detection problems.

Generative models aim at reproducing the input images by learning the original distribution with some variations. **Variational Autoencoders (VAE)** and **Generative Adversarial Networks (GAN)** are the most commonly used generative methods to-date. The ability of such models to learn the input distribution has proven effective in identifying outliers in high-dimensional datasets with complex structures.

Convolutional Neural Networks (CNN) are used widely in computer vision, medical image analysis, natural language processing, and financial time-series. Recently, researchers started

investigating the application of such architectures in anomaly detection that led to promising results. Zhang et al.[4] proposed a fraud detection model, based on CNNs, that achieves remarkable performance on transaction data from a commercial bank without derivative features (Figure 2.5).

Fraud detectors that employ **Autoencoders (AE)** assume higher reconstruction error for outliers compared to inliers, as they are trained solely on normal data instances. Despite both Autoencoders and PCA aim at dimensionality reduction, Autoencoders extend PCA by enabling both linear and nonlinear transformations. Various autoencoder architectures have been proposed and tested effective in anomaly detection tasks. The choice between the different architectures is often a matter of the nature of data. **Long Short Term Memory Networks (LSTM)** are preferable when dealing with sequential data while CNNs work better with image datasets. Efforts to combine models, such as **Gated Recurrent Unit Autoencoders (GRU-AE)**, **Convolutional Neural Network Autoencoders (CNN-AE)**, and **Long Short-Term Memory Autoencoders (LSTM-AE)**, eliminates the need for sophisticated feature engineering and facilitates the employment of raw data with minimal preprocessing in anomaly detection tasks.

2.4 Post-hoc Anomaly Detection Techniques

The hype of Artificial Intelligence offers incredible opportunities for anomaly detection problems. Machine learning appears robust but yet has to address numerous challenges to reach mass adoption, especially in industrial applications, where decision making demands high precision. Most ML methods imply ideal conditions and rely on the assumption that training and test data follow the same distribution. However, this assumption is rarely satisfied in real-world applications, where test data are noisy, exposed to adversarial corruptions, and other temporal and spatial effects that impact on their distribution. These deviations are the so called anomalies or outliers.

Machine learning models are highly sensitive to such anomalies hence it is critical to determine the level of difference between the training and test data. Then and only then, we can determine whether the detector is reliable or it generates noisy decisions. The most common setup up-to-date consists of a training phase on in-distribution data and a prediction phase on both in-distribution and out-of-distribution test samples. Even though we expect the detector to perform well on in-distribution data samples, it remains challenging to achieve high reliability on non-conforming test samples. The latter led to the idea of post-hoc anomaly detectors.

For completeness, we classify anomalies into unintentional and intentional. Unintentional anomalies refer to data points independent of the machine learning model when intentional anomalies are model dependent as they constitute attempts to force incorrect results. Bulusu et al.[15] have conducted a recent survey on anomalous instance detection in deep learning with the focus on post-hoc anomaly detection techniques. Even though Deep Learning is out of the scope of the current thesis, we invite the interested reader to consult the original paper for detailed information and references.

2.5 Interpretable Machine Learning

Despite the advancements in Machine Learning and the recent efforts of the scientific community towards explainable models, yet many models are considered as black boxes. Despite their tremendous potential, they are still far away from mass adoption, especially in industrial domains such as finance, where the justification of any decision plays a crucial role. The ability to explain your results increases the trustworthiness of your system, which is often preferable to the detriment of pure performance.

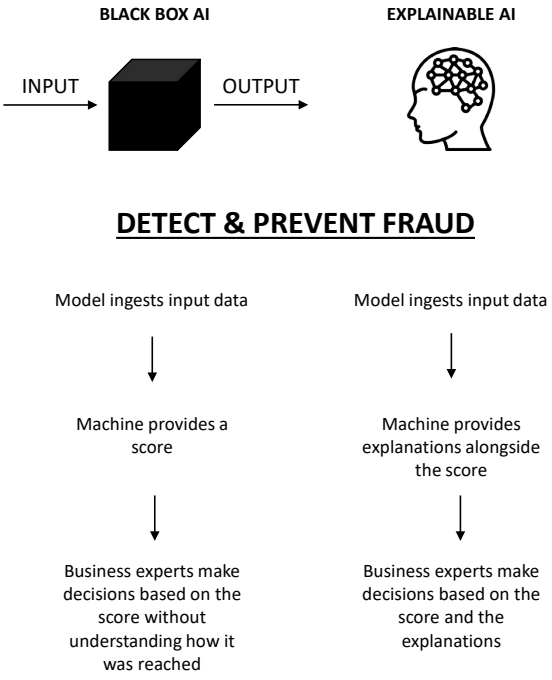


Figure 2.6: Explainable Artificial Intelligence (XAI) enhances the fairness of anti-fraud systems.

The black box accountability is associated with supervised learning due to the necessity to understand predictions, but it scales up to cover most machine learning problems. Similar questions arise when an unsupervised model flags data points as outliers, and data scientists seek the reasoning behind each decision. In fraud, the abundance of variables that accompany a credit application may contribute to a powerful detector; however, it remains a question of how to quantify that contribution and evaluate whether humans influence the decision of the model.

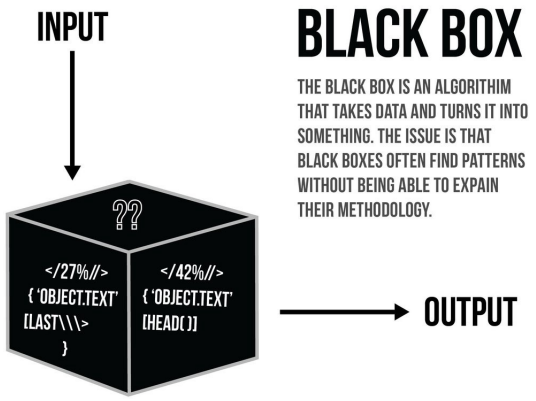


Figure 2.7: The black box concept (source: Medium).

Visualization, such as the widely used dimensionality reduction methods that allow us to project high-dimensional data in a lower-dimensional space, constitutes the core of most attempts to interpret machine learning models. Despite the effort to develop tools and methods to explain learned features and complex representations, we struggle to guarantee the transparency in decision making and the statistical validity of the results. Machine Learning models are data-driven, which justifies their black-box nature to a certain extent, however, it is essential to develop the right methods to be able to explain them efficiently. The selection of an algorithm over another is always a trade-off between accuracy and interpretability. In high-risk environments, we tend to sacrifice accuracy for explainability when in low-risk environments there is no such need.

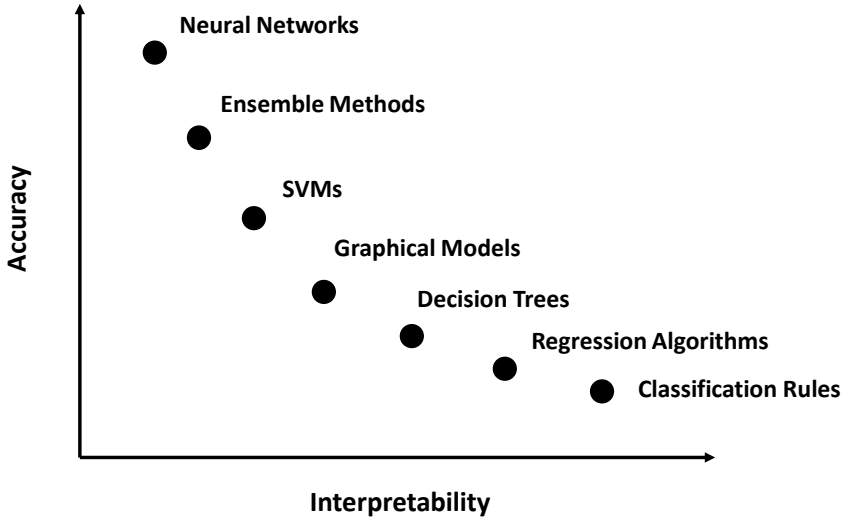


Figure 2.8: Very often high accuracy is achieved at the expense of interpretability.

The distinction between different model visualization methods and interpretability metrics lies in a two-dimensional space, where the x-axis ranges from local to global and the y-axis from model-specific to model-agnostic methods, respectively. Local methods refer to the analysis of parts of the network (e.g. individual layers) when global methods study the model as a whole (e.g. visualization of the weight distribution in a deep net). On the other hand, a model-agnostic method examines the

input or output data distribution when model-specific methods focus on the structure of algorithms and representations. The most popular methods in use up-to-date are the following[5]:

- **Partial Dependence Plots**[16] show the marginal effect one or two features have on the predicted outcome of a machine learning model. In fraud, for instance, we can visualize the influence of variables such as the years in business or the requested finance amount on the predictions of the model. Despite the intuitive nature of partial dependence plots and the simple implementation, the realistic restriction on two features and the assumption of independence are strong disadvantages for real-world applications.
- **Individual Conditional Expectation (ICE)** is similar to the Partial Dependency Plots except for displaying a different line per instance in the dataset that shows the instance-specific dependency of a feature variable. Figure 2.9 illustrates such plots in the example of bicycle rental prediction. We see the association between the prediction of each instance and features like temperature, humidity, and wind speed. In fraud, similar plots can reveal dependencies and help understand better the drivers behind model predictions. Unlike Partial Dependence Plots, ICE plots have an even more intuitive design that can uncover heterogeneous relationships. Yet, ICE can only display one feature meaningfully otherwise the plot becomes overcrowded and hardly visible.

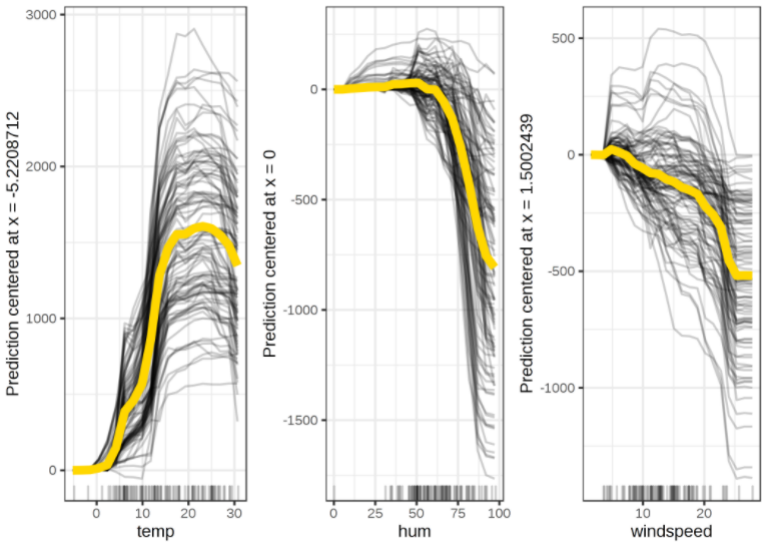


Figure 2.9: Centered ICE plots of predicted number of bikes by weather condition[5].

- **Accumulated Local Effects (ALE) Plots** is a faster and unbiased alternative to PDPs that describe the average influence of the features on the model predictions. ALE plots address the major issue of PDPs, which fail to produce meaningful explanations when the features are correlated, by calculating differences in predictions instead of averages.
- **Permutation Feature Importance** exposes relationships between features and the outcome by measuring the change in prediction error after permuting the feature's value. A feature is considered important when permuting its values increases the prediction error and unimportant when the error remains unchanged. Despite the trivial interpretation of this method, it requires prior knowledge of the outcome, hence it is not suitable for unsupervised learning.
- **Global Surrogate** refers to the method that leverages an interpretable model to explain the predictions of a black-box model. Linear regression, logistic regression, decision trees, and k-nn are among the most popular interpretable models that can be used to approximate

the predictions of the original model. Surrogate models appear flexible and straightforward, however, it is often difficult to decide what is a good approximation.

- **LIME (Local Interpretable Model-Agnostic Explanations)**[17] is an implementation of Local Surrogate models and constitutes a model-agnostic method. The basic idea lies in creating a dataset, consisting of permuted samples and the corresponding predictions of the black-box model, and training an interpretable model to closely approximate local predictions. Interpretable models can be Lasso or a decision tree that results in short and easy explanations that are often preferred as human-friendly. Besides, the black-box model approximation gives us an estimation of how reliable the generated explanations are. However, the simplicity of LIME is not always preferred, as in sectors where you are legally required to explain the predictions in detail, blocking them from mass adoption.
- **Anchors**[18] deploy a perturbation-based strategy to generate local explanations for predictions of black-box machine learning models. Unlike surrogate models used by LIME, the resulting explanations are expressed as intuitive IF-THEN rules, called anchors. Key advantages of using anchors include easily interpretable rules, the notion of coverage as a measure of importance, and great functionality when model predictions are non-linear or complex. On the other hand, anchors require hyperparameter tuning to yield meaningful results, as well as discretization to avoid too specific explanations with low coverage.

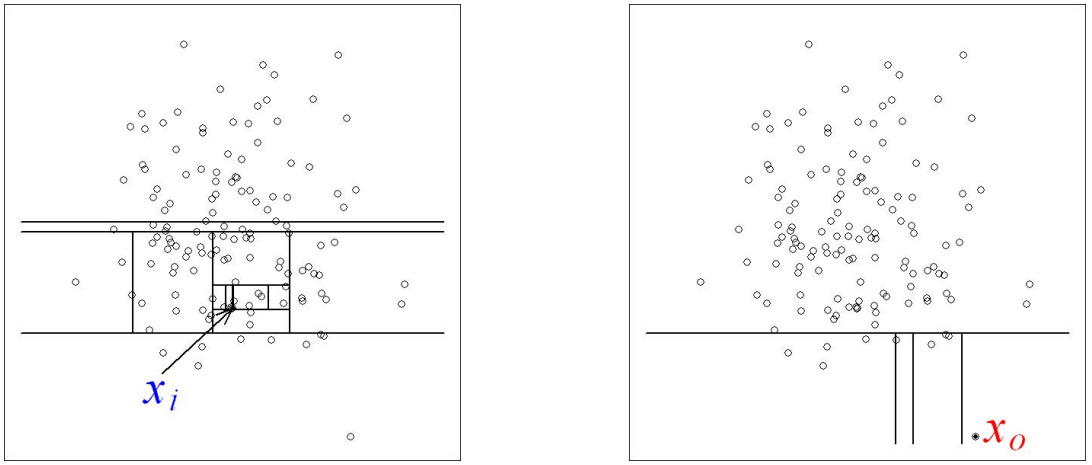
```
IF years_in_business = 5
AND financed_amount = 1.000.000
THEN PREDICT Fraud = true
WITH PRECISION 98%
AND COVERAGE 16%
```

- **SHAP (SHapley Additive exPlanations)**[19] is a game theory method based on the theoretically optimal Shapley Values. Briefly, the Shapley Values is an adapted method from coalitional game theory, which assumes each feature value of an instance as a player in a game where the prediction is the payout. In Machine Learning, each feature is a player that contributes differently to the output value. Section 3.3 discusses SHAP in detail.
- **Counterfactuals**[20] search for the smallest change possible to the feature space that could lead to a predefined output. In financial applications, a counterfactual can be formulated as: "if Farmland was a client of PGS for 10 years without ever being on default, the application would not have been fraudulent". Counterfactual explanations are very clear to interpret and can be reported either by using the counterfactual instance itself or highlighting the key changes between the instance of interest and the counterfactual. On the other hand, the counterfactuals of a given instance can be multiple and the method cannot handle categorical variables with many different levels.

Chapter 3

Anomaly Detection Framework & Explanations

3.1 Isolation Forest



(a) Isolating x_i.

(b) Isolating x_0.

Figure 3.1: Anomalies are more susceptible to isolation and hence have short path lengths. Given a Gaussian distribution (135 points), (a) a normal point x_i requires 12 random partitions to be isolated; (b) an anomaly x_0 requires only 4 partitions to be isolated [6].

The vast majority of existing model-based anomaly detection methods rely on profiling normal data points and, then, flag instances that do not confront the normal profile as anomalies. Different algorithms have their own different way of defining a normal point; some employ statistical methods, others use classification or clustering but, ultimately, the process remains the same — model normality and filter out everything else. The Isolation Forest[6] is unique to its kind as it can isolate anomalies explicitly, without profiling normal data points. Traditional methods are designed to profile normal instances but they are not optimized to detect anomalies, reduce the number of false positives, or deal with high computational complexity. Controlling false alarms, for instance, is considered equally important for financial institutions to pure performance.

Similar to any tree ensemble method, the isolation forest is designed upon the basis of decision trees. Isolation works very well on the assumption that anomalies belong to the minority class and their attribute values are unusual comparing to those of normal instances. In a tree structure, partitions are created by first randomly selecting a feature and then selecting a random split value in the minimum-maximum range of the selected feature (Figures 3.1 and 3.2). Due to anomalies'

susceptibility to isolation, we expect them to get isolated closer to the root, whereas normal data points are isolated at the deeper end of the tree. Ultimately, anomalies are instances that have short average path lengths on the isolation trees.

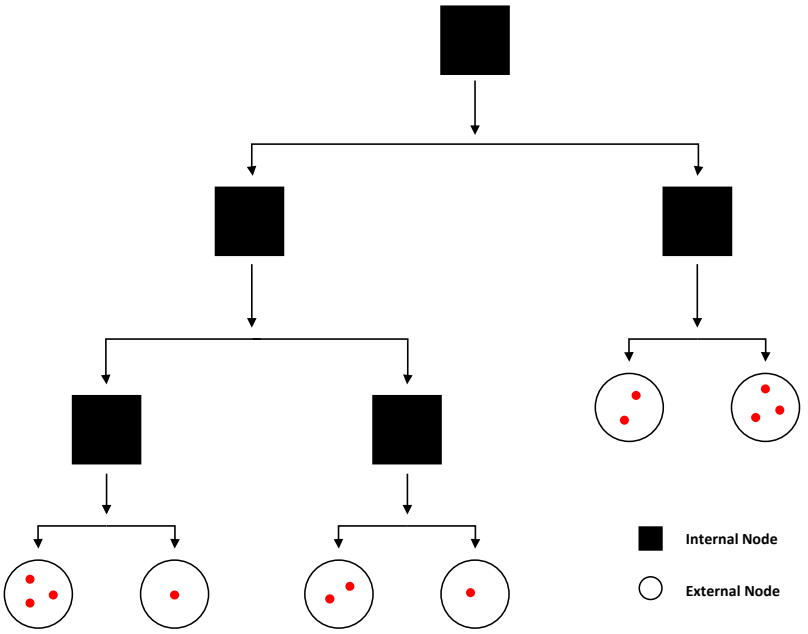


Figure 3.2: An iTree can have two types of nodes: internal and external. Internal nodes are non-terminal nodes that contain the split value, the selected feature, and pointers to exactly two child nodes. External nodes are terminal nodes that cannot split further. Each external node holds the size of the non-generated subtree which contributes to the anomaly score. The iTree is a proper binary tree. (Source: Medium).

According to the authors of the original paper, the main advantages of Isolation Forest compared to model-based, distance-based, and density-based approaches are the following:

1. The ability to isolate iTrees enables us to build partial models and exploit sub-sampling to the extent that it is not feasible in existing methods. We can avoid the construction of a large part of an iTree that isolates normal points as it is not necessary for anomaly detection. A small sample size produces better iTrees due to the reduced swamping and masking effects. Figure 3.3 depicts the result of sub-sampling.
2. iForest utilizes no distance or density measures to detect anomalies. This characteristic eliminates the computational cost of distance calculation in all distance and density-based methods.
3. iForest has a linear time complexity with a low constant and low memory requirement.
4. iForest is suitable for handling big data size and high-dimensional problems with a large number of irrelevant attributes.

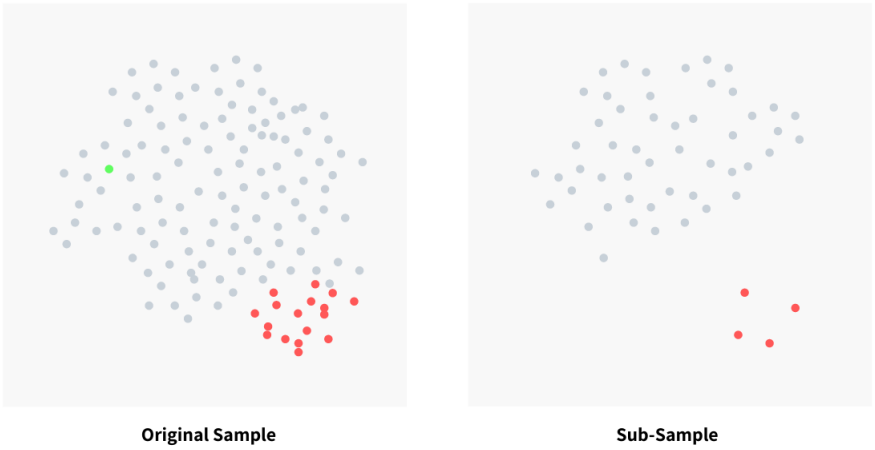


Figure 3.3: Isolation differs from most algorithms in the sense that it works better will fewer data. Sub-sampling enables the algorithm to clearly distinguish between normal and anomalous instances that might interfere in the entire dataset. Here we see how sub-sampling leads to a clear separation between normal points and anomalies. In the original sample, normal points are close together with anomalies that make the detection extremely challenging. Sub-sampling improves the accuracy of the model and results in less false positives. (Source: Medium).

Figure 3.1 illustrates the random partitioning of a normal data point (Figure 3.1a) compared to an anomaly (Figure 3.1b). We can see that normal data points require more partitions to be isolated, whereas anomalous require considerably less. The recursive partitioning represents a tree structure. Thus, the number of partitions required to isolate a data point is equivalent to the path length from the root to a leaf. In this example, the path length of x_i is greater than the path length of x_0.

Similarly to any outlier detection method, an anomaly score is essential for decision making. The challenge to derive such score lies in the peculiarity that while the maximum possible height of iTree grows in the order of n, the average height grows in the order of log n. The potential normalization of *h(x)*, where *h(x)* refers to the path length of a point x, would be either unbounded or not directly comparable.

However, iTrees' similar structure to Binary Search Trees (BST)[21] enables us to estimate the average *h(x)* for external node terminations as the unsuccessful search in BST. Given a dataset of n instances, we define the average path length of unsuccessful search in BST as:

$$s(x, n) = 2^{-}\frac{E(h(x))}{c(n)} \tag{3.1}$$

where *H(i)* is the harmonic number, which can be estimated by $\ln(i) + 0.5772156649$ (Euler's constant). As *c(n)* is the average of *h(x)* given n, we use it to normalize *h(x)*. We define the anomaly score s of an instance x as:

$$c(n) = 2H(n-1) - (2(n-1)/n) \tag{3.2}$$

where *E(h(x))* is the average of *h(x)* from a collection of isolation trees. Ultimately, each observation is given an anomaly score s and decisions can be made on the following basis:

- Scores very close to 1 indicate strong anomalies.
- Scores much smaller than 0.5 indicate normality.
- If all scores are close to 0.5, then the entire sample suggests non-recognizable anomalies.

Figure 3.4 illustrates an example of a contour of anomaly scores, which is generated by passing a lattice sample through a collection of isolation trees. The contour facilitates a detailed analysis of the detection results and enables us to visualize anomalies in the instance space.

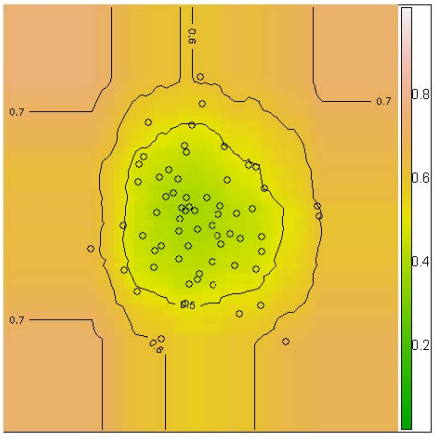


Figure 3.4: Anomaly score contour of iForest for a Gaussian distribution of 64 data points. Contour lines for s = 0.5, 0.6, 0.7 are illustrated. Potential anomalies can be identified as points where s $s \geq 0.6$[6].

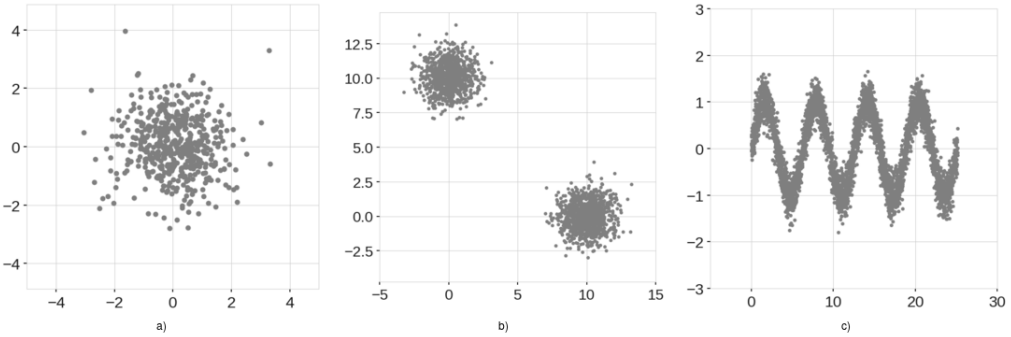
Hyperparameter tuning applies to the following parameters for optimal performance:

- **n_estimators**: The number of base estimators in the ensemble. The default is 100.
- **max_samples**: The number of samples to draw from dataset X to train each base estimator. Given large datasets, training on random subsets decreases the training time.
- **contamination**: The proportion of outliers in the dataset. As it is used when fitting to define the threshold on the scores of the samples, it often requires some trial and error combined with scatterplot visualization.
- **max_features**: The number of features to draw from dataset X to train each base estimator. Normally equals the number of variables available for training. However, this feature allows for univariate outlier detection on single variables without specifying a standard deviation threshold.

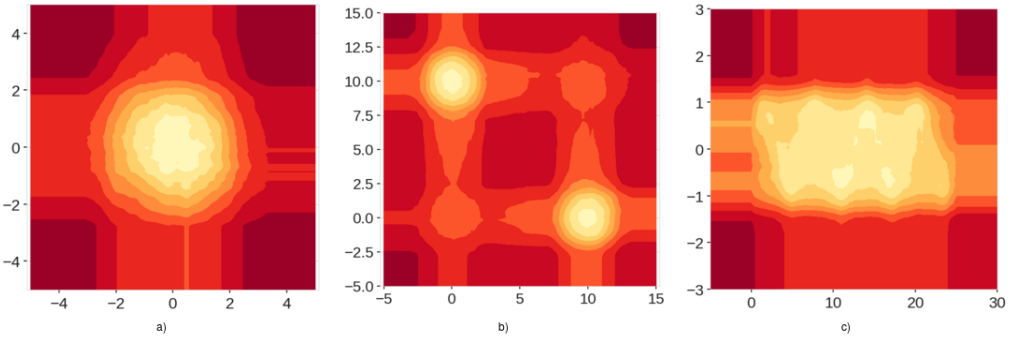
3.2 Extended Isolation Forest

Despite the success of Liu's method, the standard Isolation Forest often produces inconsistent anomaly score maps that tend to suffer from an artifact generated as a result of how we define the criteria for branching operation of the binary tree. To overcome this limitation, Hariri et al.[7] implemented an extension to the standard model-free anomaly detection algorithm. The extension, named Extended Isolation Forest (EIF), addresses the consistency and reliability problems of the anomaly score caused by the standard methods for a given data point.

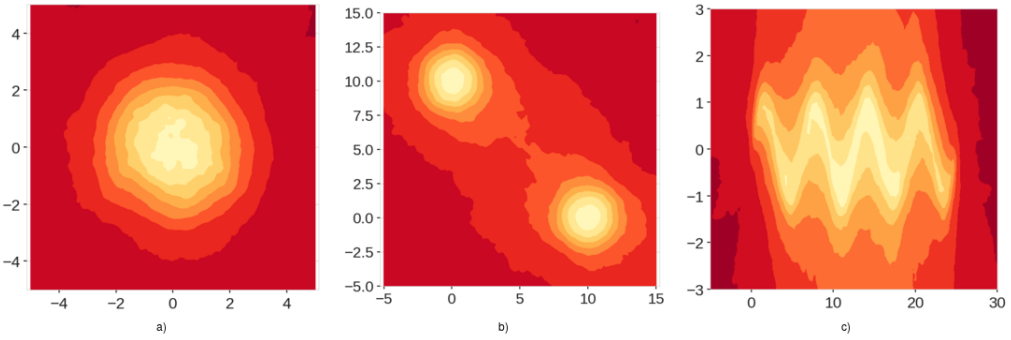
The extended method substantially improves the consistency and reliability of the algorithm by incorporating hyperplanes with random slopes for the slicing of the data. Figure 3.5 demonstrates the need for the proposed enhancement on scoring anomalies and the achieved robustness of the score maps, especially around edges of nominal data. The standard Isolation Forest is a specialized case of the extended version presented in [7]. For an N-dimensional dataset, the Extended Isolation Forest has N levels of extension, with 0 representing the standard Isolation Forest and N-1 being the fully extended version.



(a) Example training data. Left: Normally distributed cluster. Middle: Two normally distributed clusters. Right: Sinusoidal data points with Gaussian noise.



(b) Score maps using the standard Isolation Forest



(c) Score maps using the Extended Isolation Forest

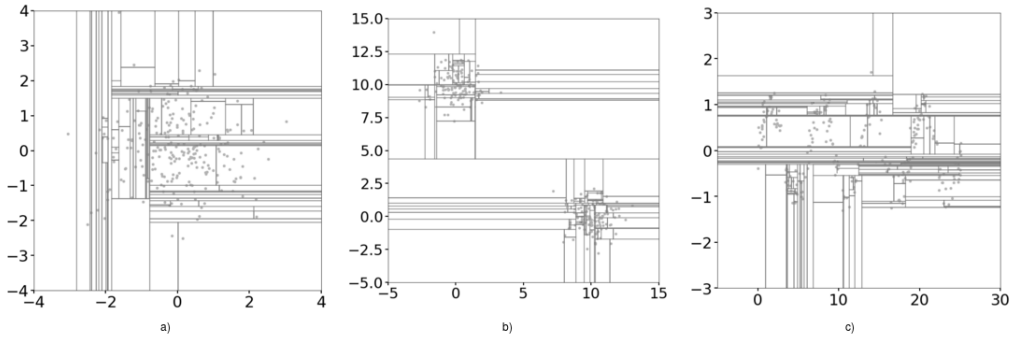
Figure 3.5: The Extended Isolation Forest substantially improves the consistency and reliability of the standard method by incorporating hyperplanes with random slopes for the slicing of the data[7].

Figure 3.5 depicts the inconsistencies in the score maps caused by the standard Isolation Forest. Looking at the left map of Figure 3.5b, we observe a region of low anomaly score in the center that aligns with our intuitive understanding of the example data in 3.5a, but we also see regions aligned with x and y axes passing through the origin that have lower anomaly scores compared to the four corners of the region. Similarly, in the middle map of 3.5, we experience an amplified problem that shows artificially low anomaly score regions to intersect close to points (0,0) and (10,10) where there is no data. Last, on the right map of 3.5, we encounter a completely lost data structure as the sinusoidal shape is essentially treated as one rectangular blob.

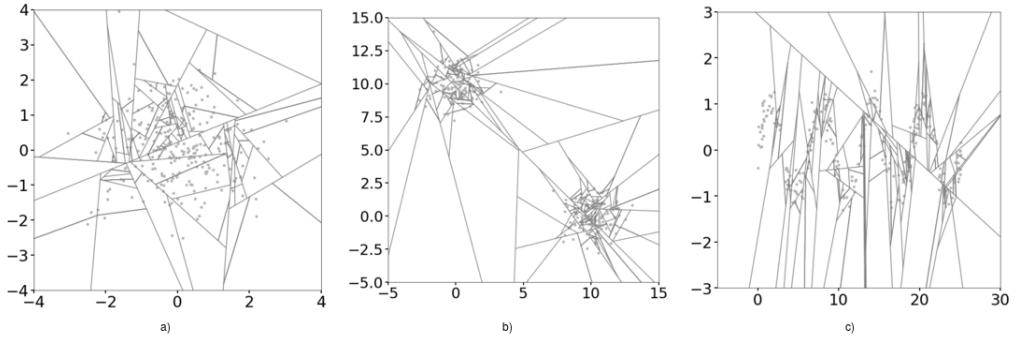
It turns out the splitting process of the standard method is the main source of the bias observed in the score maps. Figure 3.6a illustrates the standard splitting process for each of the training examples in Figure 3.5a. As the branch cuts can only be parallel to the axes, regions in the domain

that don't occupy any data points receive improper anomaly scores following the construction of many trees.

The Extended Isolation Forest remedies this problem by allowing the branching to occur in every direction. Instead of selecting a random feature along with a random value, Hariri et al.[7] have modified the process of choosing branch cuts so that at each node we pick a random normal vector along with a random intercept point. The modified process divides the region much more uniformly and eliminates the bias introducing the effects of the coordinate system. In the updated version of the algorithm, the anomaly score continues to be the aggregated depth that a given point reaches on each iTree. As we see in Figure 3.5c, the applied alterations cure the issue with the score maps that we encountered before and produce reliable results. The resulted score maps are a much better representation of anomaly score distributions.



(a) Isolation Forest



(b) Extended Isolation Forest

Figure 3.6: (a) Splitting of data in the domain during the process of construction of one tree using (a) the Standard and (b) the Extended Isolation Forest[7].

3.3 SHapley Additive exPlanations (SHAP)

The increasing complexity of modern machine learning models has led the research field of model interpretability to substantial growth. Figure 2.8 presented clearly the trade-off between model complexity and performance, where complex machine learning models such as Deep Learning are often treated as black boxes[17]. The ability to correctly interpret the model predictions is essential as it enhances the trustworthiness of your system.

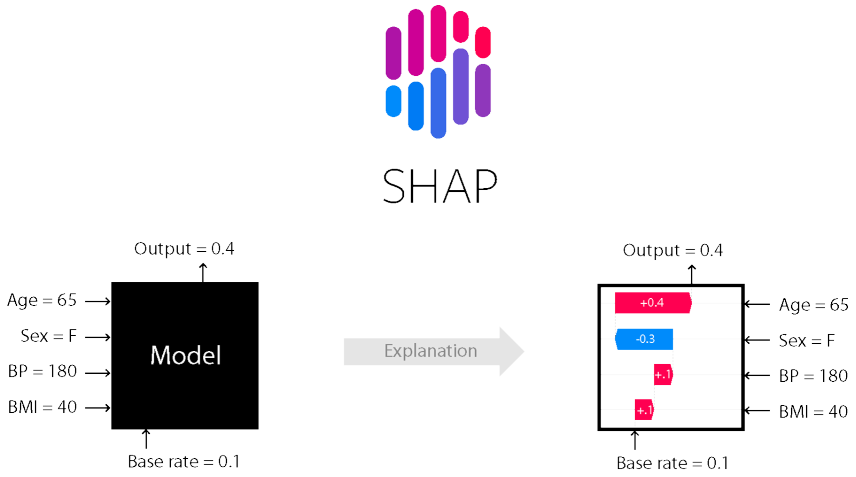


Figure 3.7: SHAP (SHapley Additive exPlanations) is a game theoretic approach to explain the output of any machine learning model. It connects optimal credit allocation with local explanations using the classic Shapley values from game theory and their related extensions (Source: GitHub).

SHAP value estimation methods[19] are novel techniques that aim for machine learning interpretability through an improved alignment with human intuition and their ability to assess better the model output than several existing methods. They boost user's trust in machine learning by providing insights on how models behave and what could improve the predictions.

Both SHAP and LIME[17] are surrogate models designed to approximate the predictions of black-box machine learning models. Similarly to sensitivity tests, they tweak the input slightly -the tweak should lie in the local region of the original data point- and evaluate the effect in prediction. This technique allows us to quantify the effect of certain variables to the prediction, as we can assume that a variable is not an important predictor if tweaking its value has no effect on the prediction. Ultimately, surrogate models are model agnostic machine learning models employed to conclude the original black box architectures.

At this point, it is crucial to understand the concept of fairness derived from Shapley values. In game theory, a game is considered fair when it meets the following conditions:

1. The sum of individual rewards should equal the total reward.
2. Multiple persons that contributed equally to the game should receive the same reward.
3. A person that did not contribute to the game should receive no reward.
4. In the case of multiple games, the total reward of an individual should equal the sum of rewards of each separate game.

In machine learning, game participants and rewards are translated to features and predictions, respectively. Lloyd Shapley proved in 1953 that there is only one method of calculating coefficients that respect all aforementioned rules. The Shapley value for a certain feature i given a prediction p is:

$$\phi_i(p) = \sum_{S \subseteq N/i} \frac{|S|!(n - |S| - 1)!}{n!}(p(S \cup i) - p(S)), \tag{3.3}$$

which, at a high level, computes the feature importance by comparing the prediction of the model with and without the feature i. However, since the order in which we add features might affect the result, $S \subseteq N/i$ describes the permutation over all possible sets of S of feature groupings to calculate a weighted sum of the final contribution.

The strong theoretical background of SHAP values, enables them to be the only explanations within a broad class of alternatives that satisfy three useful properties termed by Lundberg et al.[19]:

1. **Local Accuracy**. The approximate model matches the output of the real model for a given input.
2. **Missingness**. It requires features missing in the original input to have no impact.
3. **Consistency**. Whenever we tweak the original model such that it relies more on a feature, then the attributed importance for that feature should not decrease. Consistency allows the meaningful comparison of the feature importance between two models.

All in all, when utilizing the SHAP values, we should be aware of the following[5]:

- The efficiency property of Shapley values guarantees that the difference between the single and the average prediction is fairly distributed among the feature values of the instance, which does not hold in other methods such as LIME. This positions the Shapley values as the only method to deliver a full explanation due to their solid theoretical foundation and the ability to distribute fairly the effects in the feature space.
- In contrast to LIME, Shapley values allow to compare a prediction to a subset or even to a single data point besides comparing it to the average prediction of the entire dataset. For example, it may be more meaningful to explain a fraud prediction concerning how it deviates from dealers who have not committed fraud. In this case, we might want to use the dataset of dealers who have not committed fraud as our background dataset.
- Going through all possible combinations of features to calculate their contribution is computationally expensive. For large datasets, it would have been possible to subsample the data and rely on approximations with potential implications for the accuracy of the explanation. Luckily, the current SHAP implementation employs model-specific algorithms to optimize the computation of Shapley values, especially for tree-based models.
- In practice, we should apply a few simplifications to compute Shapley values. How we simulate the adding or removal of features while computing model prediction can be challenging as there is no straightforward way to remove a feature for most predictive models at the moment of testing. We could only replace the feature with its mean or median value. The SHAP library implementation simulates a missing feature by replacing it with the values it takes in the background dataset.
- We can utilize either the conditional or the marginal distribution to sample the absent features. The marginal distribution assumes feature independence to simplify the approximation and can provide causality; however, this assumption rarely holds in practice and we end up misleading the approximation model. On the other hand, conditional distribution solves the problem with correlated features, yet features with no influence on the prediction can get a non-zero SHAP value, which violates one of the Shapley values properties. Ultimately, there is no ideal distribution to use. We tend to use marginal distributions when background data is available, and conditional distributions in Tree-based models when there is no dataset available.
- The Shapley values can be misinterpreted. It should become clear that the Shapley value of a given feature is not the difference between the predicted value after removing the feature

from the model. On the contrary, given the current set of feature values, the Shapley value estimates the contribution of a feature value to the difference between the actual prediction and the mean prediction.

Chapter 4

Experiment

In this section we introduce the dataset used for this study and the feature engineering steps we have followed, we evaluate the proposed methods, and we conclude with a discussion over the results and the future steps.

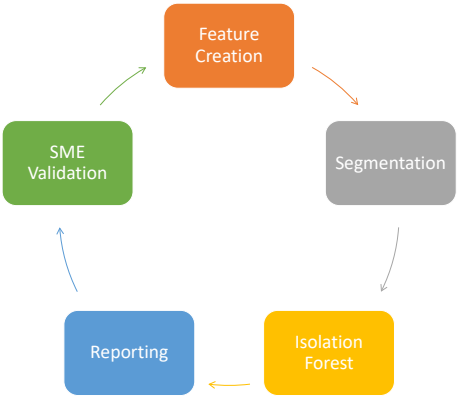


Figure 4.1: Project development workflow.

4.1 Experiment Goals & Setup

The experimental setup follows the project direction introduced in Section 1.4. In the research questions, we wondered how financial institutions could leverage unsupervised learning to uncover the evolving nature of financial crime in vast amounts of incoming credit application data. PGS provided us with an anonymized industrial dataset to experiment and reach handy conclusions regarding the applicability of the Isolation Forest in the applied domain. The performance of the employed models in the given setup, where we coupled state-of-the-art anomaly detection techniques with domain knowledge to segment our data, will eventually lead us to determine to what extent financial institutions can trust the predictions of an unsupervised learning model. The quality of the results will also determine whether the initial assumption of outlying behavior in suspicious applications holds in practice.

Besides testing the applicability of Isolation Forests in detecting fraudulent applications, we aim to study how interpretable machine learning would enhance the reliability and eventually contribute

to the adoption of such systems in industrial applications. Unlike supervised learning, where we can quickly assess the model's performance using traditional metrics, the boundaries between normal and fraudulent behavior are usually blurred. It is a common practice in large-scale applications to collect the top-n generated anomalies and send them out to business experts for inspection. However, this time-consuming process undermines the commitment to the proposed methods. Here, we have designed the experiment, so we report the predictions to the experts using SHAP explanations, and we aim to study whether their use enriches data understanding while reducing the processing time. Ultimately, we seek feedback from our business experts to assess the utility of such explanations in terms of task effectiveness, task efficiency, and mental efficiency.

4.2 Data Overview

The working dataset consists of credit applications submitted to PGS. It contains several independent variables that describe the characteristics and the behavior associated with each unique credit application as of the moment of entry into PGS's system. The pre-scoped dataset contains 441.757 observations, uniquely identified by an application key, and a total of 2.375 variables. After data scoping and preparation, we have limited the instances to 249.690. The dataset also contains a target variable that describes whether an application is legitimate or fraudulent. In our anonymized dataset, 135 contracts are labeled as fraudulent.

4.3 Feature Engineering

In their survey, Adewumi et al.[10] presented feature types used in constructing classifiers for credit card detection systems. Briefly, the feature space is organized as:

1. Features related to accounts: they include account number, account type, date of account opening, date of the last transaction, balance available in the account, card expiry date, etc.
2. Features related to transactions: they include the transaction reference number, account number, type of transaction, currency of transaction, timestamp of transaction, terminal reference number, etc.
3. Features related to customers: they include the customer number, the branch of the customer, type of customer, etc.
4. Financial Ratios[22]: they refer to information about the operating performance, the financial position, and the cash flow of a company. They are further classified as:
   - Profitability ratios: they include the operating profit, the gross profit, the net profit, the total assets, the equity, the current liabilities, etc.
   - Liquidity ratios: they include the current assets, inventories, etc.
   - Solvency ratios: they include long term debts, fixed assets, etc.
   - Activity ratios: they include operating expenses, inventories, accounts receivable, etc.
   - Structure ratios: they include retained earnings, current liabilities, cash, etc.

The current project leverages internal and external information accompanying credit applications to create features that can capture the dynamic characteristics concerning the working dataset. Confidentiality restricts us from describing the feature engineering steps in detail. After applying the data cleaning and feature engineering steps, the dataset consists of 241.234 instances and 44 numerical variables. In total, 99 applications are known as fraudulent.

4.4 Normalization

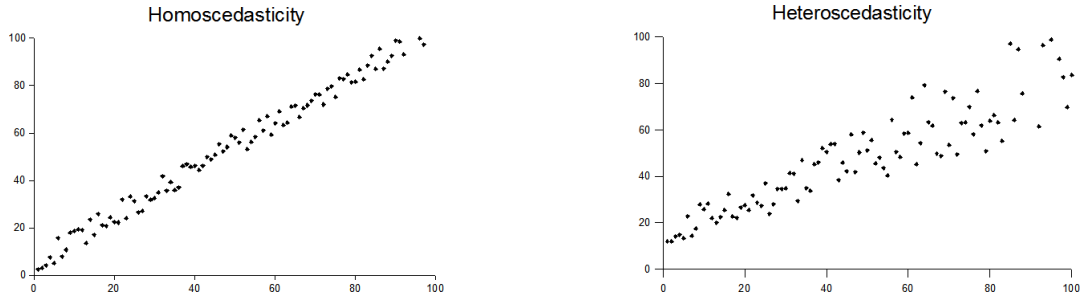
Normalization is a commonly used technique to transform features to be on a similar scale. Scaling to a range, clipping, log scaling, and z-score are some useful techniques to improve the performance and training stability of any machine learning model. Kandanaarachchi et al.[23] showed that the performance of various outlier detection methods depends sensitively on data normalization schemes employed.

In section 2.2, we presented the types of anomalies we meet in the literature. Point anomalies refer to single data points that represent an irregularity or deviation that occurred randomly. Feature engineering was employed, so our experiment constitutes a point anomaly detection problem, where outliers imply suspicious activity. Before feeding the data to the isolation forest, min-max normalization is applied coupled with background knowledge. The min-max normalization converts floating-point features from their natural range into a standard 0-1.

When to use normalization instead of standardization, where features are transformed, so their mean is zero and the standard deviation one, is an eternal question. Briefly, we tend to normalize our data when we know that they do not follow a normal distribution, while standardization can be useful in the opposite scenario. In our experimental setup, we cannot assume that our data are normally distributed, therefore we applied normalization.

It is worth mentioning that normalization can also be contra-productive. In a scenario where we have a categorical binary feature converted to [0, 1] and a numerical value measuring a length normalized to [0, 1], the influence of categorical variable to the prediction, which results in distances being either one or zero, is much higher compared to the numerical value. Thus, it is crucial to couple normalization with background knowledge to avoid such biases.

4.5 Segmentation



(a) Homoscedasticity (b) Heteroscedasticity

Figure 4.2: In statistics, a sequence of random variables is homoscedastic if all its random variables have the same finite variance. On the other hand, a vector of random variables is heteroscedastic if the variability of the random disturbance is different across elements of the vector. In financial services, we employ segmentation to achieve the level of homogeneity required to identify suspicious applications using anomaly detection techniques (Source: Wikipedia)

Segmentation is a common practice in Machine Learning. Imagine that FIs develop anti-fraud systems that treat disparate credit applications uniformly. This approach can only cause dissatisfaction and mislead any deployed system. Segmentation enables FIs to identify similar characteristics in credit applications and generalize those characteristics into groups to develop anti-fraud measures with various strategies.

Anti-fraud strategies enable financial institutions to take targeted measures against a specific group of customers, which increases the chances of being effective. They allow them to customize the red flags per unique group case and activate different scenarios accordingly. For instance, an

effective anti-fraud system considers suspicious a small enterprise to do business overseas when there are usually contracts coming from close areas. On the other hand, an oversea lease from a multinational that operates worldwide should not raise an alert. Segmentation helps companies in employing reliable systems while minimizing false alerts.

There are several methods to perform segmentation on your data. Unsupervised learning algorithms are often employed for this purpose. In the current work, segmentation is a result of applying the domain knowledge provided by the domain experts.

4.6 Performance Metrics

Despite the unsupervised learning approach in our work, performance metrics are essential to evaluate the applied models. In this work, the metrics contribute to defining the cutoff threshold for anomalies, fine-tune the isolation forests, and understand the effect of the features we created on the model. Systems that rely solely on human input for reviewing individual predictions are expensive and time-consuming.

Below we summarize the list of performance metrics we consult when evaluating our models. We clarify that we leverage the few available labels only for evaluation and thresholding in a later stage, and no model is aware of their existence at the moment of training.

- **Accuracy**: Correct Predictions / Total Predictions
- **Precision**: True Positive / (True Positive + False Positive), which represents the proportion of positive cases correctly identified.
- **Recall**: True Positive / (True Positive + False Negative), which represents the proportion of actual positive cases correctly identified.
- **AUC-ROC**: the entire 2D area under the ROC (Receiver Operating Characteristic) curve, which represents the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative. Although ROC graphs are widely used to evaluate classifiers under the presence of class imbalance, there is a drawback: the estimates can be unreliable under class rarity where the class imbalance is associated with the presence of a low sample size of minority instances (Figure 4.3).
- **AUC-PR**: the entire 2D area under the Precision-Recall curve. The precision-recall curve (or PR Curve) is a plot of the precision and the recall for different probability thresholds. The focus of the PR curve on the minority class makes it suitable for imbalanced binary classification models. Precision-recall curves (PR curves) are recommended for highly skewed domains where ROC curves may provide an excessively optimistic view of the performance[24].

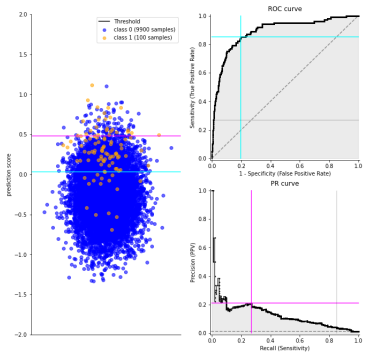
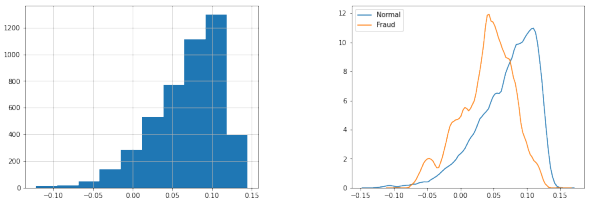


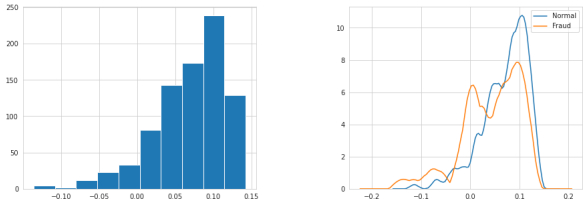
Figure 4.3: The effect of imbalanced data. PR curves are preferred when dealing with imbalanced datasets, as normally in fraud scenarios, where we are interested in the minority class (Source: Medium).

4.7 Results

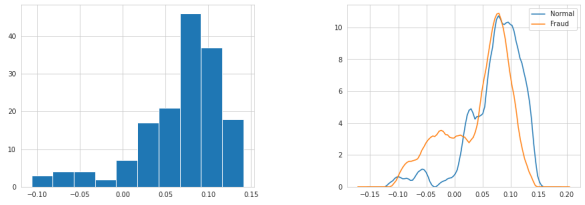
In this section, we summarize the results we have obtained experimenting with both the standard and the extended version of the isolation forest in different segments. In total, we have defined more than 200 segments of varying sizes based on domain knowledge. We could include results obtained from all segments; however, the limited labels would obstruct us from evaluating the models with objective criteria. Thus, we decided to only focus on segments with a sufficient number of fraudulent applications. For this reason, we assess the performance of the models in a setup without segmentation, which encompasses all 99 fraudulent cases, and three additional compositions (S1-S3) that involve segmentation. In S1-S3, the number of fraudulent applications is 45, 32, and 10, respectively. No segmentation refers to mixing the applications of all segments with at least one known fraud and not to the entire set of applications.



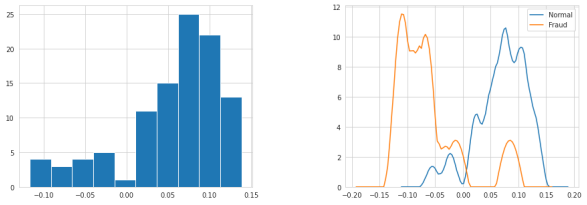
(a) Combined segments



(b) Segment 1

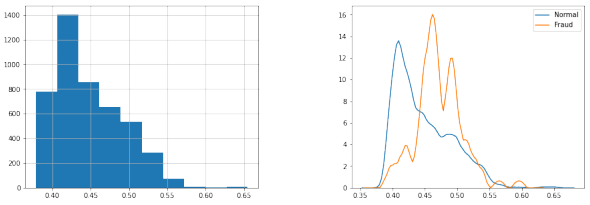


(c) Segment 2

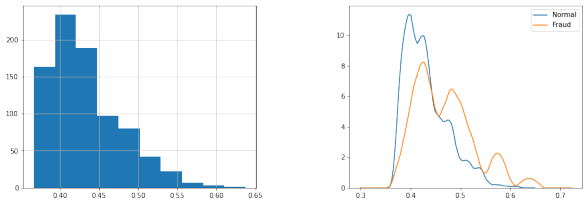


(d) Segment 3

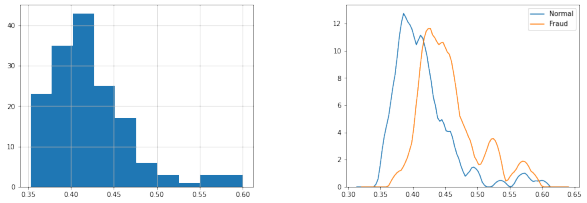
Figure 4.4: On the left, we see the distribution of anomaly scores as produced by the isolation forest with and without segmentation. Negative scores correspond to anomalous data points. On the right, the Kernel Density Estimate plot illustrates the probability density of anomaly scores that correspond to fraudulent applications against that of legitimate applications.



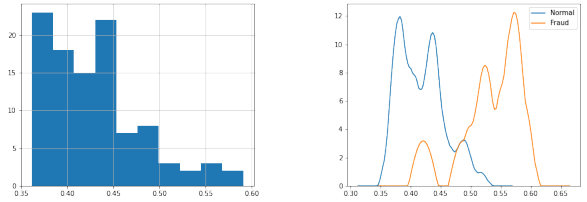
(a) Combined segments



(b) Segment 1

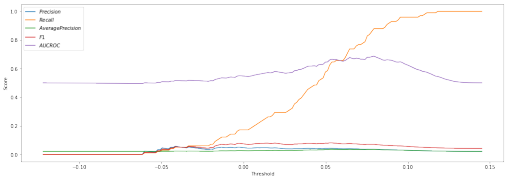


(c) Segment 2

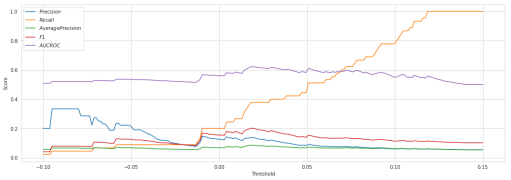


(d) Segment 3

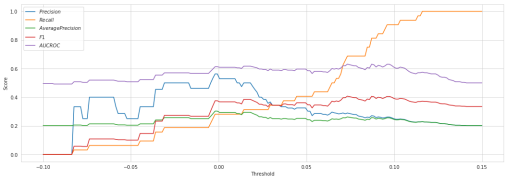
Figure 4.5: On the left, we see the distribution of anomaly scores as produced by the extended isolation forest with and without segmentation. High positive scores correspond to anomalous data points. On the right, the Kernel Density Estimate plot illustrates the probability density of anomaly scores that correspond to fraudulent applications against that of legitimate applications.



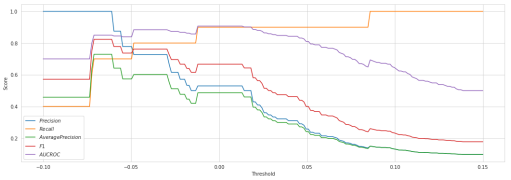
(a) Combined segments



(b) Segment 1

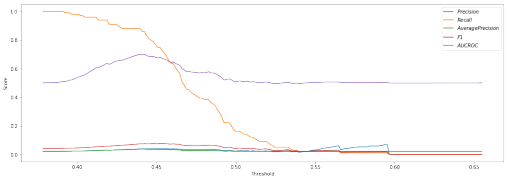


(c) Segment 2

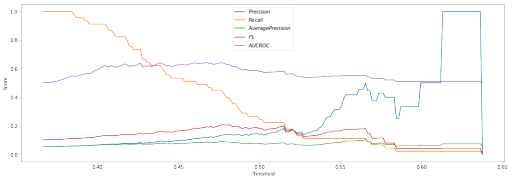


(d) Segment 3

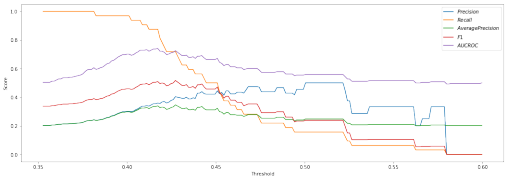
Figure 4.6: Performance Metrics for the Isolation Forest in different anomaly thresholds and segments.



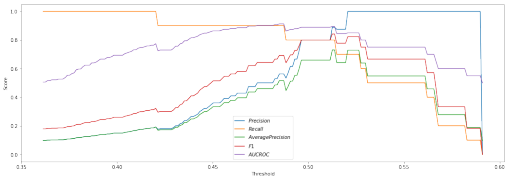
(a) Combined segments



(b) Segment 1

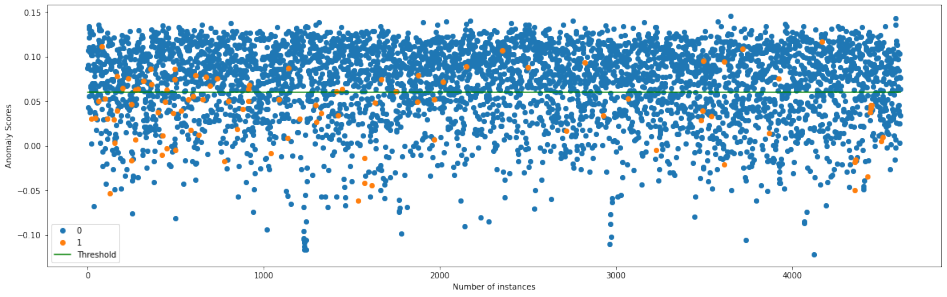


(c) Segment 2

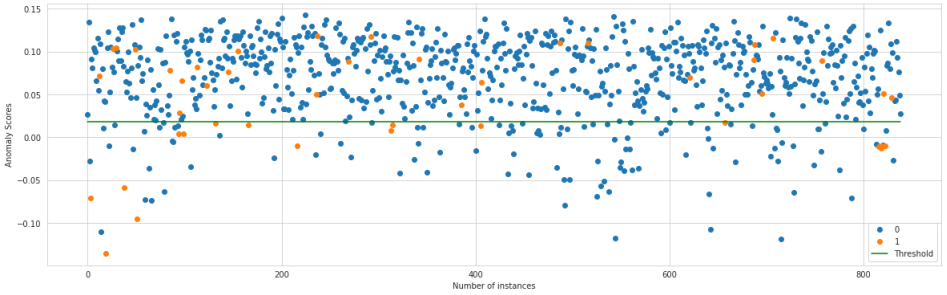


(d) Segment 3

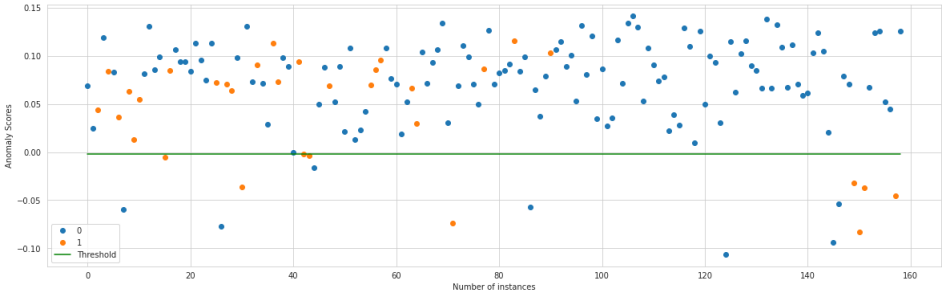
Figure 4.7: Performance Metrics for the Extended Isolation Forest in different anomaly thresholds and segments.



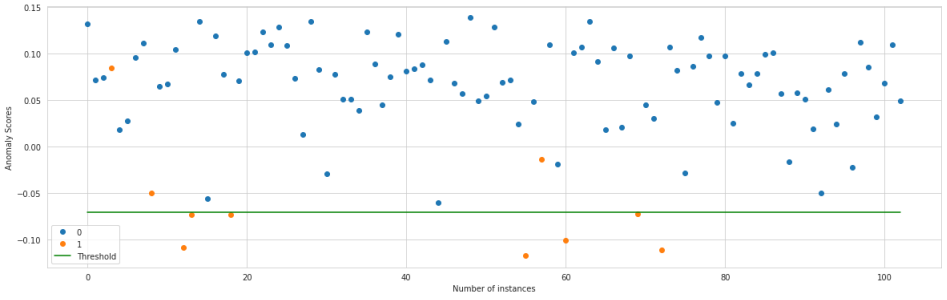
(a) Combined segments



(b) Segment 1

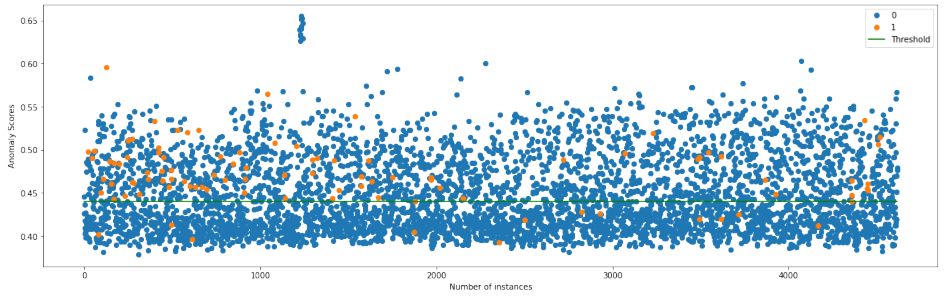


(c) Segment 2

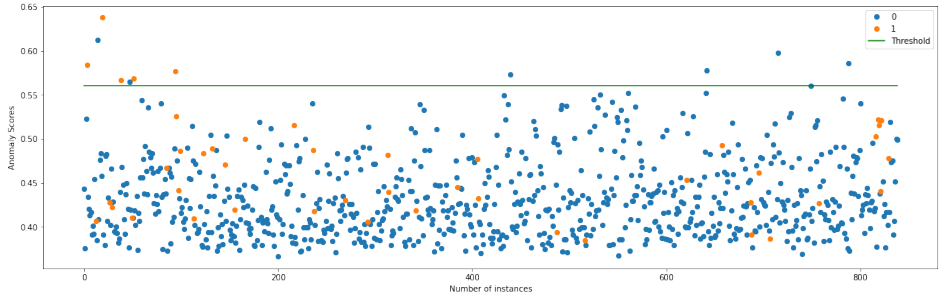


(d) Segment 3

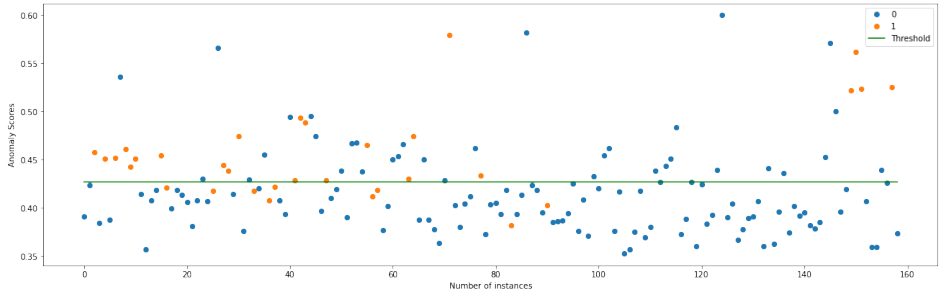
Figure 4.8: The scatter plot of the isolation forest scores assigned to each instance. Blue data points correspond to legitimate applications and orange to fraudulent. The threshold is set to the score that maximizes the AUC-PR for the given segment.



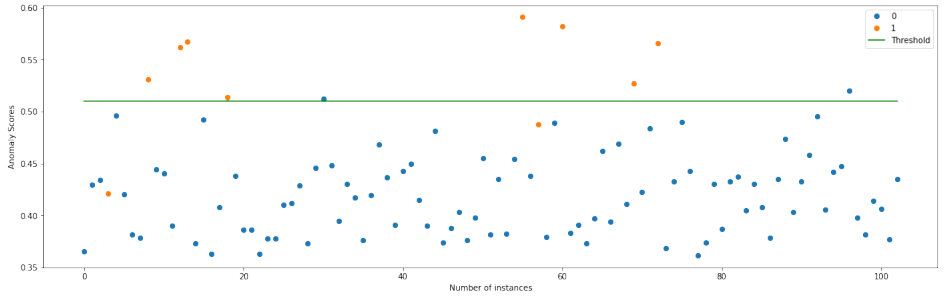
(a) Combined segments



(b) Segment 1



(c) Segment 2



(d) Segment 3

Figure 4.9: The scatter plot of the extended isolation forest scores assigned to each instance. Blue data points correspond to legitimate applications and orange to fraudulent. The threshold is set to the score that maximizes the AUC-PR for the given segment.

(a) Combined segments (b) Segment 1 (c) Segment 2 (d) Segment 3

Figure 4.10: Confusion Matrices for the Isolation Forests in different segments. The anomaly threshold has been set to the score that maximizes the AUC-PR for the given segment.

(a) Combined segments (b) Segment 1 (c) Segment 2 (d) Segment 3

Figure 4.11: Confusion Matrices for the Extended Isolation Forests in different segments. The anomaly threshold has been set to the score that maximizes the AUC-PR for the given segment.

Metric	LOF			
	S0	S1	S2	S3
Threshold	-1.16	-2.3	-1.12	-0.9
Accuracy	78.2%	94.7%	66%	16.5%
Precision	4.4%	60%	30.3%	10.4%
Recall	44.4%	6.7%	53.1%	100%
AUC-ROC	61.7%	53.2%	61.2%	53.7%
AUC-PR	3.1%	9%	25.6%	10.4%

Table 4.1: The overall performance of the Local Outlier Factor in the defined segments. S0 refers to no segmentation while S1-S3 to defined segments. The anomaly threshold has been set to the score that maximizes the AUC-PR for the given segment. The table serves as a baseline for comparison with the Isolation Forests. For the experiment, we have used the default parameters of the algorithm.

Metric	Standard IFO				Extended IFO			
	S0	S1	S2	S3	S0	S1	S2	S3
Threshold	0.06	0.02	-0.001	-0.07	0.44	0.56	0.43	0.51
Accuracy	64.9%	84%	81.1%	97.1%	53.1%	94.4%	72.3%	96.1%
Precision	3.9%	13.8%	56.2%	100%	3.9%	41.7%	39.7%	80%
Recall	65.6%	37.8%	28.1%	70%	87.9%	11.1%	71.9%	80%
AUC-ROC	65.2%	62.2%	61.3%	85%	70.1%	55.1%	72.1%	89%
AUC-PR	3.3%	8.5%	30.3%	73%	3.7%	9.4%	34.2%	65.9%

Table 4.2: The overall performance of the standard and the extended isolation forest in the defined segments. S0 refers to no segmentation while S1-S3 to defined segments. The anomaly threshold has been set to the score that maximizes the AUC-PR for the given segment. The table shows that AUC-PR improves substantially with segmentation. Moreover, we see the extended isolation forest performing better in most cases on the current setup. For the experiment, we have set the number of base estimators to 100 and max_samples equal to the min(256, n_samples). For the extended IF, the Extension Level equals the number of variables.

Local Outlier Factor (LOF) is a widely used method for detecting local anomalies, based on the principles of k-NN, where the anomaly score is estimated as a ratio of local densities. Since the density of an instance equals the densities of its k-nearest neighbors, we expect normal data points to score around 1.0 and anomalies considerably higher. Due to its ability to detect local anomalies, similar to isolation forests, we have also experimented with LOF to use it as a baseline for comparisons. In the scikit-learn implementation, the anomaly score is the opposite LOF of the training samples. Thus, inliers tend to have negative_outlier_factor_ close to -1, while outliers tend to have a large negative score. Table 4.1 summarizes the performance of LOF in segmentation scenarios. This table serves as a baseline for later comparisons with the Isolation Forests. Default parameters have been used to obtain these results.

Extended experimentation with Isolation Forests showed substantial performance improvement when segmentation has been applied. Table 4.2 summarizes the results of both the standard and the extended version of Isolation Forest in different segmentation settings. Figures 4.4, 4.5 illustrate the anomaly score distribution between the different versions of the model and the segmentation setups. In the standard version, negative scores indicate anomalies while in the extended occurs the opposite. The graphs demonstrate the difficulties to distinguish the anomaly scores between fraudulent and legitimate applications when no segmentation is applied. Figures 4.8a, 4.9a communicate the above in a more precise way. We see fraudulent data points mixed with legitimate in terms of the anomaly score.

Figure 4.12 illustrates the pair plot for a set of financial ratios when we combine data points of several segments to construct a no segmentation setup. The plot allows us to inspect both the distribution of a single variable and the relationship between pairs. When we do not apply segmentation, we see no clear separation between fraudulent and legitimate applications for the given set of features. In Appendix B, the reader can consult the pair-plots that correspond to segmented datasets.

In Section 1.6, we highlighted several challenges of anomaly detection in financial services. Financial applications are characterized by high heterogeneity that leads to high complexity and makes it tough for ML algorithms to parse signals from noise. Modeling a region that captures normality can be extremely difficult, especially in financial applications where the boundaries between normal and fraudulent behavior are usually blurred. By applying segmentation techniques, we achieved a certain level of homogeneity to be able to identify suspicious activity. Despite fraudulent and legitimate instances remain not clearly separable, the enhancement cannot be disregarded. Figures 4.4, 4.5, 4.8, 4.9 depict the above findings.

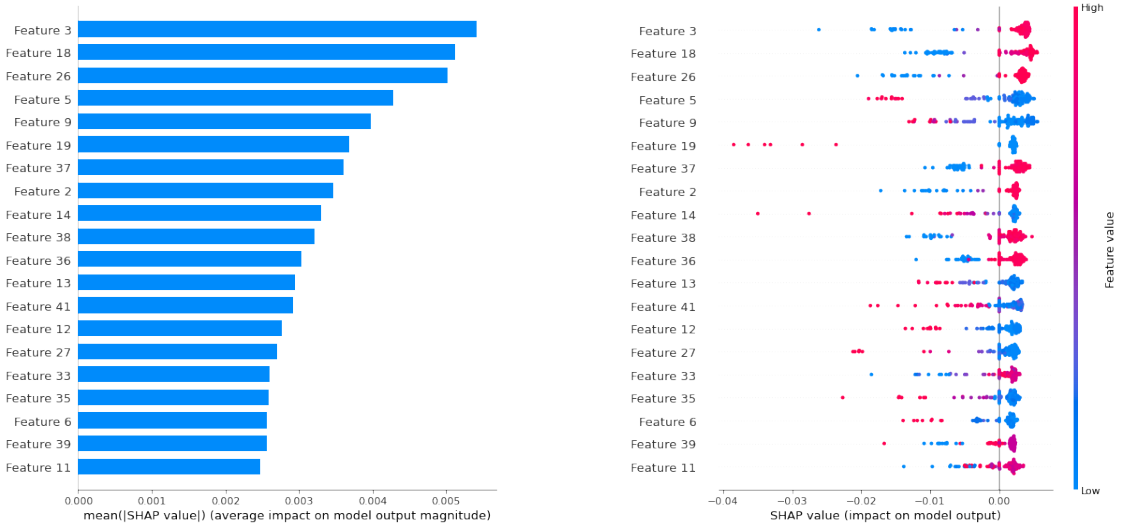
Fraudulent cases usually represent only a very small fraction of the total transactions. Although ROC graphs are widely used to evaluate models under the presence of class imbalance, there is a drawback: the estimates can be unreliable under class rarity where the class imbalance is associated with the presence of a low sample size of minority instances. This scenario is often the case in fraud detection. On the other hand, the focus of the PR curve on the minority class makes it

suitable for such scenarios. Figures 4.6, 4.7 demonstrate the model performance under several evaluation metrics. To set the anomaly score threshold, we have leveraged the AUC-PR due to the suitability in fraud projects. The above decision results in the confusion matrices presented in Figures 4.10, 4.11. Table 4.2 also shows that performance is negatively correlated with the threshold. By applying segmentation, we have not only achieved better performance, but we also experience lower thresholds. Since we expect anomalies to have a negative score, we can state from the results that the initial assumption of outlying behavior in fraudulent applications holds in several scenarios.

Figure 4.12: The figure illustrates the pair-plot for a set of financial ratios in no segmentation setup. This plot allows us to inspect both the distribution of a single variable and the relationship between pairs. When we do not apply segmentation, we see no clear separation between fraudulent and legitimate applications for the given features.

4.8 Reporting

Better interpretability leads to transparency and eventually to mass adoption. A sophisticated machine learning model can achieve accurate predictions, but its black box nature does not help adoption at all. By utilizing the state-of-the-art SHAP values, we look under the hood of machine learning predictions, and we boost the business engagement to the evaluation of the applied models. In this section, we present the SHAP plots we leverage for reporting anomaly detection results.



(a) (b)

Figure 4.13: The SHAP Variable Importance Plots

Figure 4.13 is an example of a global interpretability graph — the collective SHAP values depict how much each predictor contributes, either positively or negatively, to the target variable. Besides being a variable importance plot, it can reveal the positive or negative relationship for each variable with the anomaly score (Figure 4.13b). Figure 4.13a presents the most significant variables in descending order. The top variables contribute more to the model compared to those at the bottom and thus have high predictive power.

Briefly, the SHAP value plot (Figure 4.13b) communicates the following information:

- *Feature importance:* Variables are ranked in descending order.
- *Impact:* The horizontal axis reveals whether the effect of a certain feature value is associated with a high or low anomaly score.
- *Original value:* Color depicts the range of feature values from high (in red) to low (in blue) for all observations.
- *Correlation:* A high level of *Feature 3* has a high and positive impact on the anomaly score. The *high* derives from the red color, and the *positive* impact is visible on the x-axis. Similarly, we observe *Feature 9* is negatively correlated with the target variable.

Besides global interpretability, SHAP provide a local view to the model — each observation gets its own set of SHAP values (Figures 4.14, 4.15). Local interpretation substantially increases transparency. We can explain why the model flags an applications as anomalous and what contributes to the received score. Traditional algorithms only provide us with a feature importance overview across the entire population but not on individual observations. The local interpretability enables us to pinpoint and contrast the impact of different factors.

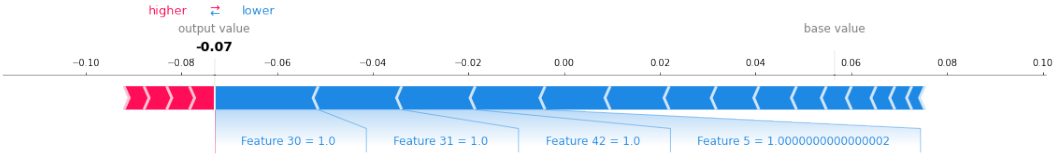


Figure 4.14: Individual SHAP Value Plot for the highest scored anomaly in segment S3.

Figure 4.14 illustrates the SHAP force plot of a high scored anomaly in segment 3. It delivers the following insights:

- The output value is the anomaly score for that observation, which is -0.07.
- According to the original paper[19], the base value is the value that the model would predict if no features are known for the current output. In simple words, it is the mean anomaly score across all observations. In segment 3, the average anomaly score of all applications is 0.056.
- Red/Blue: The red color represents features that push the prediction higher (to the right) while those pushing the prediction lower are shown in blue.
- Feature 30 has a negative impact on the anomaly score. The instance value equals 1, which is higher than the average value of 0.03. It pushes the prediction to the left.
- Feature 31 also has a negative impact on the anomaly score. The higher than the average feature value ($1 > 0.058$) drives the anomaly score to the left.
- Feature 42 is negatively correlated to the anomaly score (($1 > 0.058$)), which results in pushing the score to the left.

Besides force plots, we leverage the SHAP decision plots (Figure 4.15) to show how the models arrive at their predictions. It is important to note that decision plots are not informative by themselves; we mainly use them to illustrate primary concepts.

- The x-axis represents the model's output. In this case, it shows the anomaly scores.
- The plot is centered on the x-axis at the explainer's expected value (the mean of all anomaly scores). Similarly to the effects of a linear model to the intercept, all SHAP values are relative to the model's expected value.
- The y-axis lists the model's features, which, by default, are ordered by descending importance. The importance is calculated over the observations plotted. This is usually different than the importance of ordering for the entire dataset.
- The colored lines represent each observation's prediction. At the top of the plot, the line strikes the x-axis at its corresponding observation's output value (-0.07 in the example of Figure 4.15). The output value is responsible for the color of the line on a spectrum.
- The SHAP values for each feature are added to the model's base value, moving from the bottom of the plot to the top. This demonstrates how each feature contributes to the overall prediction.
- Ultimately, the observations converge at the explainer's expected value, which we see at the bottom of the plot.
- In scenarios where there are a large number of significant features involved, a decision plot can be more helpful than a force plot. In such cases, the force plot can be hard to read.

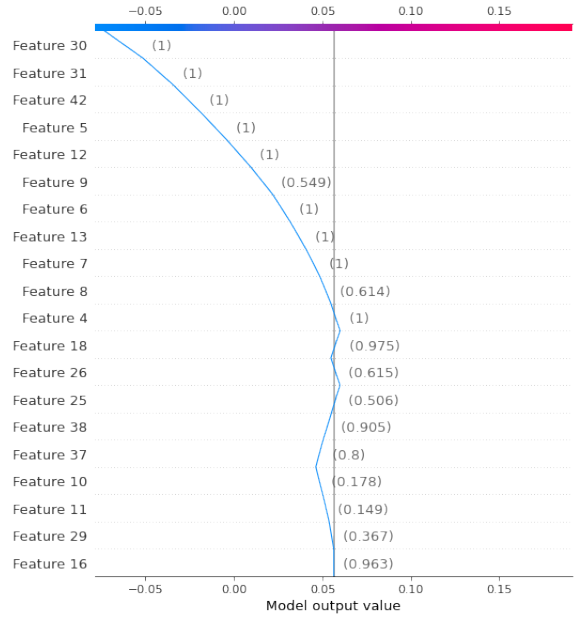


Figure 4.15: Individual SHAP Decision Plot for the highest scored anomaly in segment S3.

The additive nature of Shapley values guarantees that they always sum up to the difference between the game outcome when all players are present and the game outcome when no players are present. For machine learning models this means that SHAP values of all the input features will always sum up to the difference between baseline (expected) model output and the current model output for the prediction being explained. The easiest way to see this is through a waterfall plot (Figure 4.16) that starts with our background prior expectation for an anomaly score $E[f(X)]$, and then adds features one at a time until we reach the current model output $f(x)$.

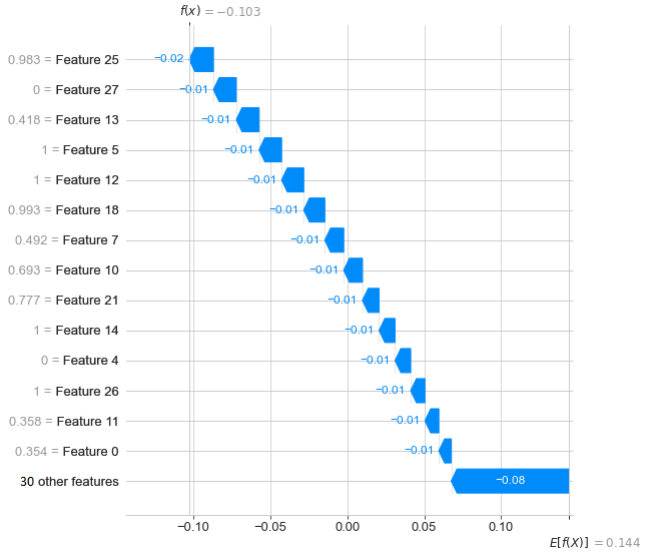


Figure 4.16: Interpretation of SHAP values away from the mean. The waterfall plot demonstrates how the SHAP values of each feature move the anomaly score of this instance from our prior expectation under the background data distribution of 100 manually selected normal points to the final model prediction, given the evidence of all the features.

The SHAP value of a feature represents the impact of the evidence provided by that feature on the model's output. The waterfall plot is designed to visually display how the SHAP values (evidence) of each feature move the model output from our prior expectation under the background data distribution to the final model prediction given the evidence of all the features. Features are sorted by the magnitude of their SHAP values with the smallest magnitude features grouped at the bottom of the plot. Figure 4.16 illustrates how the features of a high-scored anomaly move the model output from the expectation under the background distribution consisted of manually selected normal data points.

4.9 Discussion

For the inexperienced audience, the use and the capability of SHAP values might confuse. People hurry to immediately adopt SHAP values and seek ways to share the output of decision plots with their clients. They often skip questions regarding the calculation of SHAP values, and their only concern remains how to identify the business drivers to launch profitable strategies. Despite the importance of reasoning over business decisions, this topic concerns more correlation and causality. The SHAP values do not provide causality, which we can identify by experimental design or similar approaches.

Over the years, many explanation methods have been proposed to interpret machine learning predictions. However, the scientific community still questions the utility of such techniques in industrial applications. Weerts et al.[25] conducted a human-grounded evaluation of SHAP for alert processing in which they researched whether this local model-agnostic explanation method can be useful for real human domain experts to assess the correctness of positive predictions. Their qualitative analysis studied the decision-making process when performing alert processing with and without SHAP information in the supervised machine learning space, and it inspired us to investigate similar concepts in anomaly detection tasks. The current discussion delivers the feedback we received from the business experts when they had to evaluate anomalies with and without SHAP explanations.

Anomaly detection is the process of identifying suspicious activity in large data volumes. Industrial datasets often consist of high-dimensional feature spaces that are difficult to inspect. Recall also the assumption we made at the beginning where we related suspicious financial activity in credit applications with outlying behavior. Being an outlier does not necessarily imply that this particular application is fraudulent. Thus, it is essential to be able not only to evaluate an instance given its anomaly score but also to understand the drivers behind the model decision. In financial services, for example, given the tendency for annual engagements, the model might consider suspicious short-term contracts, which disagrees with the business logic.

Weerts et al.[25] measured task performance by introducing the terms of task effectiveness, task efficiency, and mental efficiency. In our work, we leverage these terms to translate the received feedback from business experts when we requested them to evaluate anomalies with and without SHAP explanations. In a similar study, Antwarg et al.[26] also evaluated the utility of SHAP values on explaining anomalies detected by Autoencoders and stated that such methods appear useful based on experts' feedback.

Task effectiveness refers to the extent to which SHAP explanations increase the ability of the user to assess the model's credibility. For example, if a certain feature contributes substantially to the anomaly score, but the expert considers this reasoning as counter-intuitive as with short-term contracts, it will be more likely to question the model's prediction. Moreover, SHAP may guide the domain experts towards important features they would not have considered without being exposed to the explanation. The business experts acknowledge that the SHAP explanations increase task effectiveness of application processing compared to the model's anomaly scores alone.

Task efficiency refers to the time we invest in a particular task and the extent to which the

explanations help us to perform it more efficiently. We already highlighted the importance of understanding the feature space in unsupervised machine learning scenarios to evaluate whether the initial assumption of outlying behavior holds for the applied domain. SHAP explanations reveal which feature values are relevant for the model's decision, and can be used to determine whether the model's output is reasonable given domain knowledge. In scenarios like ours, where the number of features of an instance is sufficiently large, SHAP helps the domain experts to process applications faster.

Mental efficiency refers to the required mental effort to perform a task. For low-dimensional instances, the extra information that the SHAP explanations deliver may cause a counter-effect as it is likely to increase mental effort. On the other hand, a complex instance with a sufficiently large number of features is unlikely to fit into working memory, which has a limited capacity according to the cognitive load theory. According to the experts, in such cases, the provided SHAP explanations guided them to focus on a subset of features to help them understand the model prediction and assess the rationale of the decision.

In the discussion, we summarized the key findings of using SHAP explanations to assess anomalous applications. The business experts endorsed the utility of such techniques in the financial domain, as they facilitate a better understanding of the proposed models and enable them to evaluate faster the generated anomalies. However, this design lacks the statistical analysis to validate that there is a significant difference in task utility metrics between tasks for which an explanation was available and tasks in which it was not provided and whether the experts' perception holds on a large scale. Confidentiality reasons restricted us from sharing features and results outside PGS, so we limited our evaluation in discussing only with business experts involved in this project.

Chapter 5

Conclusion

5.1 Contribution

FIs suffer millions in losses due to Fraud incidents annually. Thus, they attempt to leverage AI systems able to discern whether given financial activities are suspicious or not, and alert fraud analysts to take immediate action through predefined workflows. Payment fraud schemes constantly grow in complexity, often incorporating a completely different digital footprint or sequence and structure. By employing AI, FIs seek to be in a position to spot anomalies in their large-scale datasets in seconds.

AI systems demonstrate terrific potential. They can process large volumes of data in real-time and learn fast to identify suspicious financial activity. Yet, few FIs have employed AI in their anti-fraud efforts - only 5.5% of them have adopted real AI systems. The motivation behind the master thesis was the design and the implementation of a system capable of identifying suspicious patterns and detecting fraudulent behavior. In general, the straightforward approach to model normality and classify anything that lies outside the defined boundaries as an anomaly faces various challenges.

Modeling a region that captures normality can be extremely difficult, especially in financial data applications where the boundaries between normal and fraudulent behavior are usually blurred. Malicious activities tend to adapt rapidly to system requirements and appear normal at first sight. Moreover, the notion of normality changes continually, as what we consider normal today might not be valid in the future. A powerful application should be able to generalize and adapt to the applied changes. However, gathering labeled data of abnormal behavior is a primary challenge as it is often expensive and time-consuming. In most cases, we have access to an abundance of normal observations, but it is tough to gather enough data on abnormal behavior to train a good model.

Here, we aimed to reinforce supervised machine learning models by leveraging the benefits of unsupervised machine learning space and limited label availability. Unsupervised learning unlocks the potential of supervised learning the moment we manage to define a threshold that gives us a rough estimation of whether a transaction is considered fraudulent or not. For this reason, we incorporated a small fraction of labeled data to fine-tune the resulted notion of normality based on the score that maximizes the AUC-PR. The current state-of-the-art solutions lack the maturity to handle industry-level problems with ease, despite their potential to cope up with the increasing complexity and high-dimensional inputs. The thesis aimed to expose and leverage the ML advancements in the industrial world.

The vast majority of existing model-based anomaly detection methods rely on profiling normal data points and, then, flag instances that do not confront the normal profile as anomalies. Different algorithms have their own different way of defining a normal point; some employ statistical methods, others use classification or clustering, but, ultimately, the process remains the same — model normality and filter out everything else. The Isolation Forest is unique to its kind as it can isolate anomalies explicitly, without profiling normal data points. It has its basis on decision trees, where

partitions are created by first randomly selecting a feature and then selecting a random split between the minimum and maximum value of that feature. Traditional methods are designed to profile normal instances, but they are not optimized to detect anomalies, reduce the number of false positives, or deal with high computational complexity. Controlling false alarms, for instance, is considered equally important for financial institutions to pure performance. Ultimately, iForest has linear time complexity with a low constant and memory requirement and utilizes no distance or density measures to detect anomalies, which eliminates the substantial computational cost in all distance and density-based methods.

In Section 4.7, we summarized the results we obtained experimenting with both the standard and the extended version of the isolation forest in different segments. For the experiment, we defined more than 200 segments of varying sizes based on domain knowledge and the feedback provided by the business experts. Instead of including results from all segments, we only presented predictions for segments with a sufficient number of fraudulent applications. In a different scenario, it would have been difficult to determine an objective criterion for the performance of our models.

Extended experimentation with both the standard and the extended version of the Isolation Forest showed substantial performance improvement when we apply segmentation. Financial applications are characterized by high heterogeneity that leads to high complexity and makes it tough for ML algorithms to parse signals from noise. Modeling a region that captures normality can be extremely difficult, especially in financial applications where the boundaries between normal and fraudulent behavior are usually blurred. By applying segmentation techniques, we achieved the required level of homogeneity to be able to identify suspicious activity with outlying behavior. Despite fraudulent and legitimate instances remain non clearly separable, the obvious enhancement cannot be disregarded especially in S3. All in all, the overall performance demonstrates the suitability of the proposed state-of-the-art methods on large-scale tasks applications with heterogeneous datasets as we managed to identify fraudulent applications with outlying behavior in several scenarios. However, our work remains exploratory, and further evaluation with more segments is required to become conclusive.

Despite the advancements in Machine Learning and the recent efforts of the scientific community towards explainability, yet many models are considered as black boxes. Despite their tremendous potential, they are still far away from mass adoption, especially in industrial domains where the justification of any decision plays a crucial role. The ability to explain your results increases the trustworthiness of your system, which is often preferable to the detriment of pure performance.

Industrial datasets often consist of high-dimensional feature spaces that are difficult to inspect. Being an outlier does not necessarily imply that a particular application is fraudulent. Thus, it is essential to be able not only to evaluate an instance given its anomaly score but also to understand the drivers behind the model decision. SHAP value estimation methods are novel techniques that aim for machine learning interpretability through an improved alignment with human intuition and their ability to assess better the model output than several existing methods. They boost user's trust in machine learning by providing insights on how models behave and what could improve the predictions.

In the current thesis, we leveraged the terms of task effectiveness, task efficiency, and mental efficiency to evaluate the utility of SHAP explanations in unsupervised scenarios. The analysis concerns the feedback we received from the business experts when assessed anomalies with and without explanations.

The business experts acknowledged that the SHAP explanations increase the model credibility as they provide insights on the features that contribute to the predicted anomaly score, and therefore, increase task effectiveness on application processing compared to the model's anomaly scores alone. In cases where certain features seemed to contribute substantially to the anomaly score, but the expert considered this reasoning as counter-intuitive, it was easier to question the model's prediction and suggest improvements. Besides task effectiveness, SHAP values contributed to task efficiency. In financial services, where the number of features is large, SHAP helped the domain experts to process applications faster. The high-dimensionality of the working dataset also affects mental efficiency. The instance complexity coupled with a large number of features limits the received

information from fitting into working memory. The business experts stated that the provided SHAP explanations guided them to focus on a subset of features and assisted them to understand the model prediction and assess the rationale of the decision.

Interpretability leads to transparency and eventually to mass adoption. A sophisticated machine learning model can achieve accurate predictions, but its black box nature does not help adoption at all. By utilizing the state-of-the-art SHAP values, we looked under the hood of machine learning predictions, and we boosted the engagement of business experts to the evaluation of the applied models.

5.2 Limitations & Future Work

Machine learning improves accuracy once it has access to a large volume of data to learn from it. In fraud detection, millions or even billions of data points enable the machines to build a comprehensive understanding to distinguish between illicit and legitimate behavior. The evolving nature of the crime requires machines to encounter as many examples as possible to become effective. Unlike credit card fraud, where we have access to an abundance of daily transactions, in credit applications, the frequency is considerably lower. Continuous collection of data is the keystone for developing effective anti-fraud systems.

Despite the valuable business insights, the very few available labels impede an extensive evaluation of our work. We have limited the presented results into three segments with a sufficient number of fraud cases; however, it would have increased the trustworthiness of the output if we could demonstrate the improvements of our method in more segments. An alternative would be to also present the top anomalies for the entire dataset but, the absence of an objective criterion to the evaluation of the model performance would have been an undeniable drawback.

The small number of labels also affects setting the anomaly threshold. In our work, we leveraged the labels to set the threshold to the score that maximizes the AUC-PR. There is an ongoing debate related to thresholding among the scientific community, without a winning opinion. Negative anomaly score, contamination ratio, the interquartile range of the anomaly scores, labels, business logic, manual inspection are just a few techniques utilized to define such a threshold. Future experimentation within the same setup could profoundly study the effects of different methods in the produced results. Gathering labeled data of abnormal behavior is expensive and time-consuming; hence developing reliable models that require no labels should become a priority for the industrial world.

The concept of interpretable machine learning has been in the spotlight the recent years. Machine learning has grown to a powerful tool in problem-solving, yet mapping complex, nonlinear functions is difficult to interpret. In fraud, the ability to identify drivers of outcomes is critical to convince domain professionals to rely on the detections of such systems. In the current work, we presented key findings of using SHAP explanations to assess anomalous applications. The business experts endorsed the utility of such techniques in the financial domain, as they facilitate a better understanding of the proposed models and enable them to evaluate faster the generated anomalies. However, confidentiality restricted us from conducting a statistical analysis to validate that there is a significant difference in task utility metrics between tasks for which an explanation was available and tasks in which it was not provided and whether the experts' perception holds on a large scale. Future work should focus on getting a fully anonymized copy of the same dataset to validate the benefits of utilizing SHAP values in different scenarios. This statistical analysis would eventually lead to the adoption of such methods in industrial applications.

Bibliography

[1] Markus Goldstein and Seiichi Uchida. A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PLoS ONE*, 11, 2016.

[2] Raghavendra Chalapathy and Sanjay Chawla. Deep Learning for Anomaly Detection: A Survey. pages 1–50, 2019.

[3] Markus Goldstein and Seiichi Uchida. A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PLoS ONE*, 11(4):1–31, 2016.

[4] Zhaohui Zhang, Xinxin Zhou, Xiaobo Zhang, Lizhi Wang, and Pengwei Wang. A Model Based on Convolutional Neural Network for Online Transaction Fraud Detection. *Security and Communication Networks*, 2018, 2018.

[5] Christoph Molnar. *Interpretable Machine Learning*. 2019. https://christophm.github.io/interpretable-ml-book/.

[6] F. T. Liu, K. M. Ting, and Z. Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, 2008.

[7] Sahand Hariri, Matias Carrasco Kind, and Robert J. Brunner. Extended isolation forest. *ArXiv*, abs/1811.02141, 2018.

[8] Ruth Bolton, Janet McColl-Kennedy, Lilliemay Cheung, Andrew Gallan, Chiara Orsingher, Lars Witell, and Mohamed Zaki. Customer experience challenges: bringing together digital, physical and social realms. *Journal of Service Management*, 09 2018.

[9] Aly Al-Amyn Valliani, Daniel Ranti, and Eric Karl Oermann. Deep learning and neurology: A systematic review. *Neurology and Therapy*, 8(2):351–365, Dec 2019.

[10] Aderemi O Adewumi and Andronicus A Akinyelu. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *International Journal of Systems Assurance Engineering and Management*, 8:937–953, 2017.

[11] Ziad Obermeyer and Sendhil Mullainathan. Dissecting racial bias in an algorithm that guides health decisions for 70 million people. In *FAT*, 2019.

[12] Bernhard Schölkopf, Robert Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. volume 12, pages 582–588, 01 1999.

[13] Dana H. Ballard. Modular learning in neural networks. In *AAAI*, 1987.

[14] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41, 07 2009.

[15] Saikiran Bulusu, Bhavya Kailkhura, Bo Li, Pramod K. Varshney, and Dawn Song. Anomalous Instance Detection in Deep Learning: A Survey. 2020.

[16] Jerome Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 10 2001.

[17] Marco Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. pages 97–101, 02 2016.

[18] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI*, 2018.

[19] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. 12 2017.

[20] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr, 2017.

[21] Bruno Preiss. *Data Structures and Algorithms with Object-Oriented Design Patterns in Java.* 01 2000.

[22] Rasa Kanapickienė and Živilė Grundienė. The Model of Fraud Detection in Financial Statements by Means of Financial Ratios. *Procedia - Social and Behavioral Sciences*, 213:321–327, 2015.

[23] Sevvandi Kandanaarachchi, Mario Andrés Muñoz Acosta, Rob Hyndman, and Kate Smith-Miles. On normalization and algorithm selection for unsupervised outlier detection. *Data Mining and Knowledge Discovery*, 11 2019.

[24] Paula Branco, Luis Torgo, and Rita Ribeiro. A survey of predictive modelling under imbalanced distributions, 2015.

[25] Hilde J. P. Weerts, Werner van Ipenburg, and Mykola Pechenizkiy. A human-grounded evaluation of shap for alert processing, 2019.

[26] Liat Antwarg, Bracha Shapira, and L. Rokach. Explaining anomalies detected by autoencoders using shap. *ArXiv*, abs/1903.02407, 2019.

Appendices

Appendix A

Pair plots in different segments

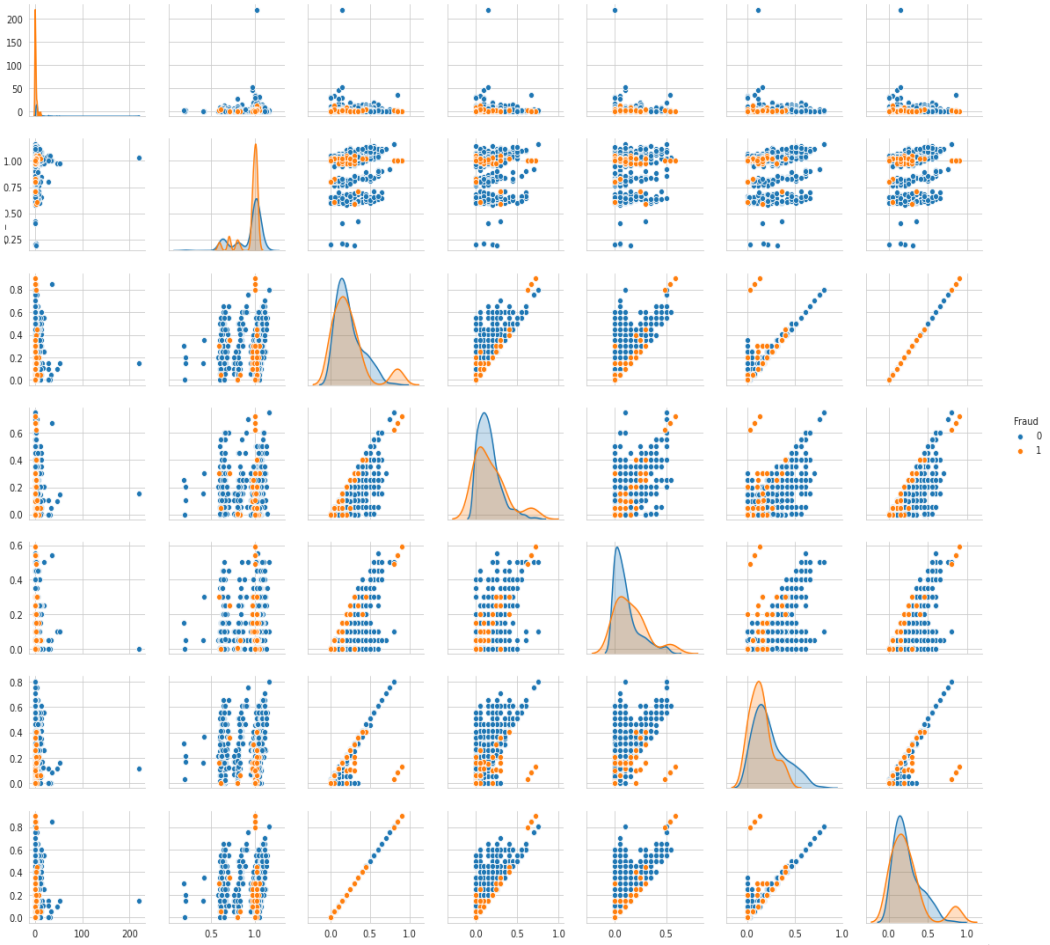


Figure A.1: Pair plot of selected financial ratios in segment 1.

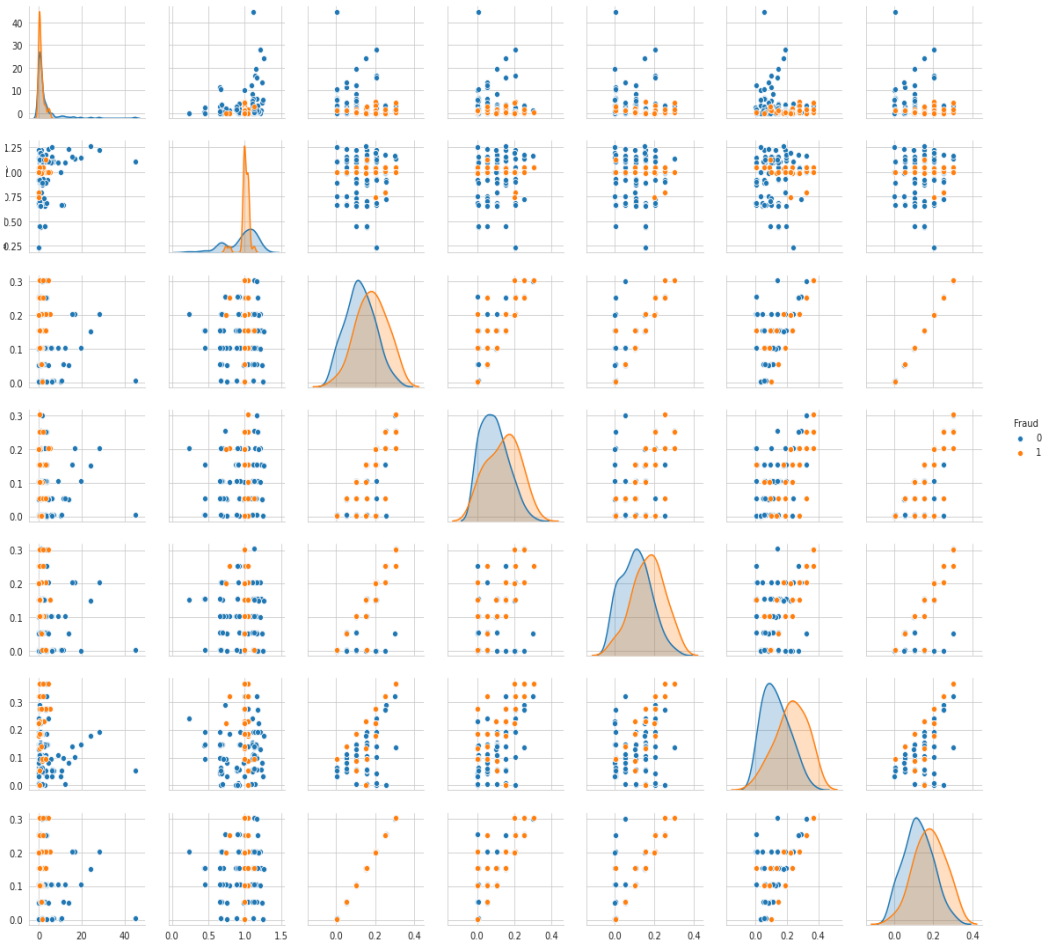
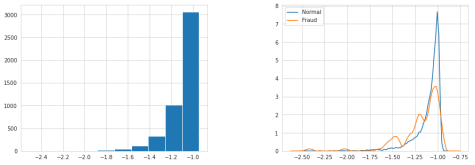


Figure A.2: Pair plot of selected financial ratios in segment 2.

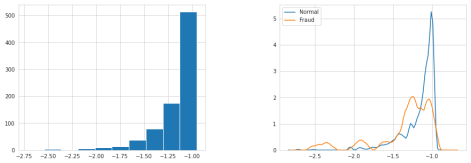
Figure A.3: Pair plot of selected financial ratios in segment 3.

Appendix B

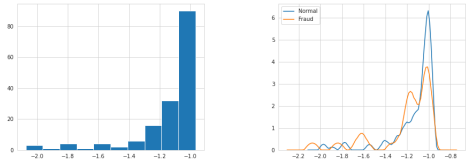
Local Outlier Factor Results



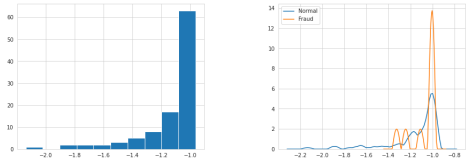
(a) Combined segments



(b) Segment 1

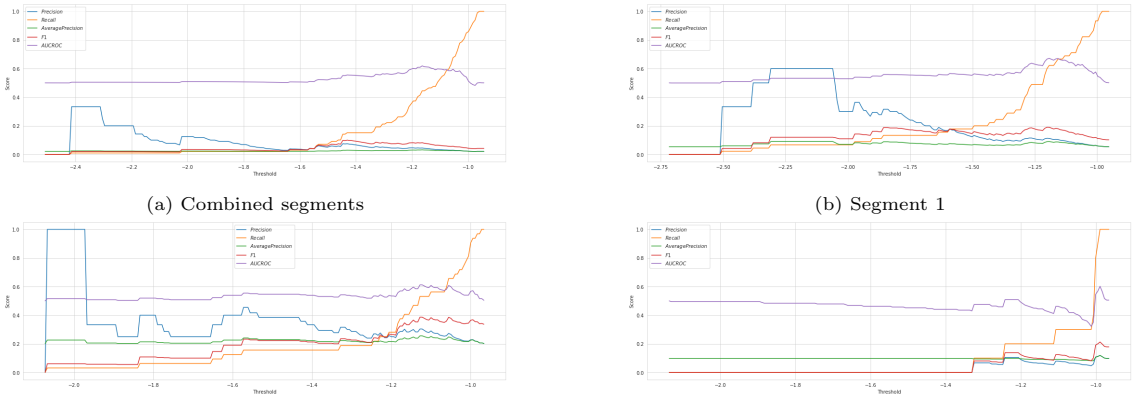


(c) Segment 2



(d) Segment 3

Figure B.1: On the left, we see the distribution of anomaly scores as produced by the Local Outlier Factor with and without segmentation. Large negative scores correspond to anomalous data points while scores close to -1 correspond to inliers. On the right, the Kernel Density Estimate plot illustrates the probability density of anomaly scores that correspond to fraudulent applications against that of legitimate applications.



(a) Combined segments

(b) Segment 1

(c) Segment 2

(d) Segment 3

Figure B.2: Performance Metrics for the Local Outlier Factor in different anomaly thresholds and segments.

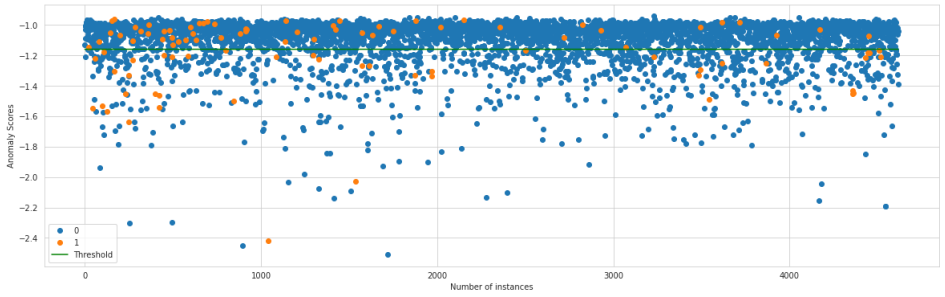
(a) Combined segments

(b) Segment 1

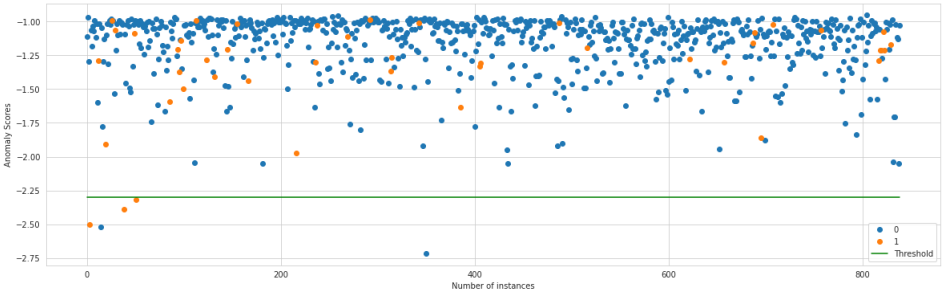
(c) Segment 2

(d) Segment 3

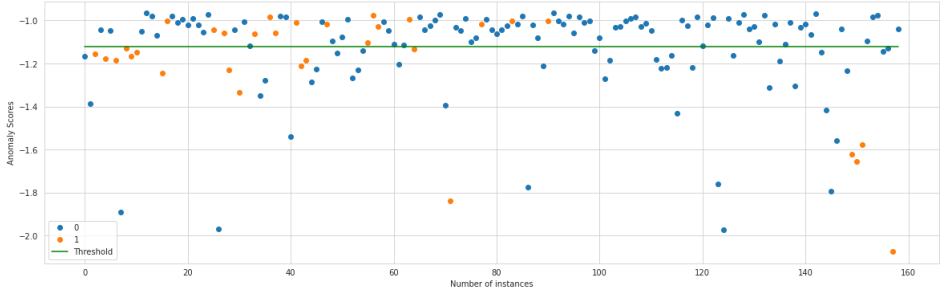
Figure B.3: Confusion Matrices for the Local Outlier Factor in different segments. The anomaly threshold has been set to the score that maximizes the AUC-PR for the given segment.



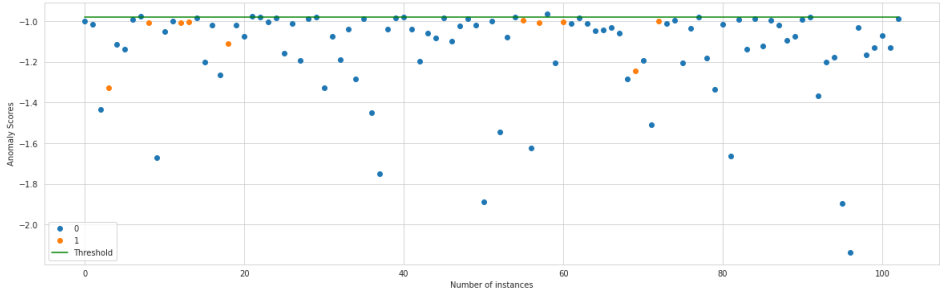
(a) Combined segments



(b) Segment 1



(c) Segment 2



(d) Segment 3

Figure B.4: The scatter plot with the Local Outlier Factor scores assigned to each instance. Blue data points correspond to legitimate applications and orange to fraudulent. The threshold is set to the score that maximizes the AUC-PR for the given segment.