NOVA IMS Information Management School

MDSAA

Master's Degree Program in
Data Science and Advanced Analytics

Enhancing Fraud Detection in the Insurance Industry:
Integrating SHAP explanations into an Unsupervised Anomaly Detection Model with Autoencoder-based Dimensionality Reduction

João Morais Costa

Dissertation

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

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

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

ENHANCING FRAUD DETECTION IN THE INSURANCE INDUSTRY: INTEGRATING SHAP EXPLANATIONS INTO AN UNSUPERVISED ANOMALY DETECTION MODEL WITH AUTOENCODER-BASED DIMENSIONALITY REDUCTION

by

João Morais Costa

Dissertation presented as partial requirement for obtaining the Masters's Degree in Data Science and Advance Analytics, with a Specialization in Data Science

Supervisor: Professor Roberto Henriques

November 2023

STATEMENT OF INTEGRITY

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

João Morais Costa

Lisbon, November 2023

ACKNOWLEDGMENTS

First, I would like to express my gratitude to Future Healthcare Group for providing me with the opportunity to undertake my internship. My sincere appreciation goes to Ana Pina and Ricardo Galego, my co-supervisors, for their invaluable guidance and support throughout this project. Their expertise and mentorship greatly contributed to the development and success of this work, as well as to my overall growth during this internship. This project is the result of an intensive group working and their brilliance. I am also grateful to my supervisor, Professor Roberto Henriques, for his continuous support and valuable insights. I would like to extend my thanks to my parents and to my partner for their unwavering support and understanding during this significant transition in my career. Their encouragement and belief in me were vital in overcoming challenges and pursuing this opportunity.

ABSTRACT

In the insurance sector, machine learning techniques are widely employed to aid auditing teams in identifying potentially fraudulent claims. At Future Healthcare Group, an unsupervised anomaly detection (UAD) model has been deployed to support a dedicated team in the audit process. This model incorporates an autoencoder for dimensionality reduction of part of its feature space. This project starts with the question: *'Is it possible to increase the efficiency of the current UAD model by increasing its interpretability with SHapley Addictive Explanations (SHAP)*?'. Due to its 'nested architecture' the direct implementation of SHAP explanations directly into this model poses computational challenges namely in uncovering the information compressed by the autoencoder. This project aimed at developing a framework that efficiently integrates SHAP explanations into the unsupervised anomaly detection model. This project is divided in two steps: In the first step, it focuses on building the framework; in the second, the framework output is evaluated. The framework increased the efficiency of the model. This was achieved mainly by indirectly increasing the UAD model performance. The presence of the explanations allowed to uncover observations classified as anomalous due to its rarity that were not true anomalies by business definition. This allowed the pre-filtration of these, which contributed indirectly to the increased performance of the based model. In summary, the developed framework offers an efficient solution for integrating SHAP explanations into an unsupervised anomaly detection model, particularly when a part of the feature space undergoes compression via an autoencoder.

KEYWORDS

Health Insurance; Fraud Detection; Unsupervised Learning; Anomaly Detection; Autoencoder; SHAP

Sustainable Development Goals (ODS):

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS AND ACRONYMS

AI	Artificial Intelligence
GDPR	General Data Protection Regulation
FHG	Future Healthcare Group
LIME	Local-Interpretable Model Agnostic Explanations
ML	Machine Learning
SHAP	SHapley Addictive exPlanations
UAD model	Unsupervised Anomaly Detection Model
XAI	Explainable Artificial Intelligence

1. INTRODUCTION

This project was developed in the context of an academic traineeship at the Future Healthcare Group (FHG), under the supervision of Professor Roberto Henriques and the practical guidance of Ana Pina and Ricardo Galego. FHG is a Portuguese private group founded in 2003, mainly focused on healthcare plans and healthcare insurance management that operates in Europe and in South America. As a common practice of the global insurance sector, including the healthcare insurance industry, there is a dedicated team within the group focused on the claim assessment. It seeks to, among other objectives, actively identify and analyze potentially abusive and fraudulent behavior.

Healthcare fraud is a pervasive global phenomenon that exerts a substantial economic impact, leading to reported losses estimated in the billions of dollars worldwide (*Morris,2009; NHCAA,2021; Glee et al, 2010; Zhang et al, 2020*). The International Social Security Association (*ISSA, 2022*) recognizes the healthcare insurance sector as a prime target for healthcare fraud which is by nature difficult to evaluate. The same author state that shifting the paradigm from post-detection to prevention, has emerged as a more effective approach in managing improper healthcare expenditures, and that emergent technologies such as Artificial Intelligence (AI) and advanced analytics are in the leading front of this change.

In recent years, AI namely through Machine Learning (ML) has been progressively used as an extension and an enhancement tool of the conventional ruled-based fraud detection systems, which despite their good results are traditionally less adaptive to sudden changes. The unique ability to learn from data, makes ML suitable to detect data patterns often not apparent to human eye or to ruled-based-systems (*Gupta, 2023*). However, the outputs generated by ML, despite their reported effectiveness, often pose challenges for human interpretability unlike the more transparent nature of rule-based systems. The tendency to behave like a '*black box*' has been a significant obstacle to its widespread adoption, a challenge not confined to the insurance sector alone (*Tu, 1996; Elshawi et al, 2019; Petch et al, 2022*).

Explainable Artificial Intelligence (XAI) is an advancing field committed to the improving transparency of machine learning (ML) automated systems. *(Abadi and Berrada, 2018).* Its primary objective is to enhance the understandability of the decision-making processes in these systems, driven by both development needs and legal requirements *(Goodman and Flaxman, 2017).* XAI techniques aim to demystify the *'black box'* models, providing insights into how these arrive to their predictions, thus fostering trust and accountability in automated decision systems *(Barreto Arrieta, et al, 2020)*. Several techniques have been developed and tested in the last years. One of the most popular is Shapley Additive Explanations (SHAP) developed by *Lundberg and Lee* (2017), based on the game theory concepts of Shapley Values introduced by *Lloyd Shapley* in 1953. SHAP has been extensively applied in various fields, including military, healthcare, and insurance (*Mihirette and Tan, 2022; Stenwig et al, 2022; Serré et al, 2021; Bora et al, 2021*). While acknowledging that SHAP

may be computationally less efficient than alternative methods, such as Local Interpretable Model-Agnostic Explanations (LIME), it is important to emphasize that SHAP was the chosen interpretability technique for this project. This decision followed initial testing, which included a comparison with LIME during the project's early phases. However, due to observed inconsistencies after several iterations with the same pool of observations, LIME was not included in the subsequent stages of this project. Notably, Molnar (2022), in the LIME chapter of his work 'Interpretable Machine Learning,' also highlights similar concerns about LIME's explanation variation, especially when sampling is repeated. The election of SHAP as the explainability technique to be used in this project, was made due to the recognized SHAP's theoretical robustness, which according with some authors (*Alvarez-Melis and Jaakkola, 2018; Molnar, 2022*) provides more consistency and less variables results when compared with LIME. This recognized robustness can be attributed to solid theoretical foundations and mathematical properties of SHAP values, which ensure more reliable and interpretable explanations (*Lundberg and Lee, 2017; Alvarez-Melis and Jaakkola, 2018; Molnar, 2022*). Hence, despite the potential higher computational resources demand, the consistency of results made SHAP the preferred choice as the interpretation technique in this project.

Currently in FHG, an unsupervised anomaly detection model (UAD model) based on the Isolation Forest algorithm, has been used to support the identification of potentially fraudulent claims, which before any decision are mandatory to be evaluated by a human. Due to the uninformative nature of the ML models, it's often not clear to the audit the reasons behind the model's classification of certain claims as anomalies. The inclusion of explanations to the model output could help the audit team and therefore increase the efficiency of the UAD model. Therefore, this project started with the goal of answering the following question: *'Is it possible to increase the efficiency of the current UAD model by increasing its interpretability with SHAP*?'. However, the direct implementation of an explainability technique such as SHAP, into the UAD model, poses some challenges since this model uses an autoencoder to compress part of its input feature space before the anomaly detection step. The architecture of this model is described further on this work, in section 1.2.

To tackle the challenge posed by the 'nested' architecture of the UAD model, our project aimed at developing a framework that could allow the explanation of suspicious claims identified by the UAD model in an efficient way without losing the quality of these explanations. By producing human understandable explanations for the output of the UAD model, this framework would help answer our research question: *'Is it possible to increase the efficiency of the current UAD model by increasing its interpretability with SHAP*?'.

1.1 – THESIS ORGANIZATION

The organization of this thesis proceed as follow:

Section 1 (Introduction): In this section, we introduce the project's topic, context, and the primary research question. We also provide a description of the Unsupervised Anomaly Detection (UAD) model to help readers understand the context of the explanation framework.

Section 2 (Literature Review): This section offers a comprehensive review of the existing literature on XAI techniques, specifically focusing on SHAP. We explore how these techniques are applied to explain the outputs of unsupervised learning models, namely when there is a degree of dimensionality reduction performed by autoencoders. Key theoretical frameworks and their applications in the insurance sector are examined.

Section 3 (Methodology): This section outlines the research methodology used in this thesis. It includes the experiences that contributed to both the construction and evaluation of the explanation framework. The Sections 3 to 5 are subdivided into two distinct components: 'Framework Construction' and 'Framework Evaluation,' each elucidating specific aspects of the research process. The framework construction section explores the experiences that led to the construction of the framework for generating explanations for the UAD model, namely in which step of the UAD model should SHAP explanations be performed, and how should the information compressed by the autoencoder be explained. The first experience starts with the hypothesis that performing SHAP at the Isolation Forest step is computational less expensive and offers the similar explanations as to englobe all UAD model in the SHAP explanations. The second experience tests the hypothesis that both SHAP and error reconstruction method offer concordant results in explaining the information compressed by the autoencoder. As for the framework evaluation section, it evaluates the output of the framework namely in the effect on the claim analysis time and on the overall UAD model's performance.

Section 4 (Results): In this section, we present the findings from our research experiences. We detail the most optimized way to explain the output of the UAD model and the results of the framework evaluation.

Section 5 (Discussion): The results are interpreted within the context of the existing literature, leading to conclusions about the initial hypotheses drawn from the explanation framework's construction and evaluation.

Section 6 (Conclusions): This section summarizes this thesis' key findings, namely the explanation framework's effect on the efficiency and interpretability of the current UAD model.

Through this structured approach, we aim to provide a comprehensive research analysis of this topic.

1.2 - UAD Model

On this sub-chapter, we aim at describing the UAD model in order to provide a better context to the main goal of this project. The current UAD model, which is an unsupervised anomaly detection model, relies on an Isolation Forest algorithm for the anomaly detection and on an autoencoder to provide dimensionality reduction to a part of the Isolation Forest's feature space, as illustrated in figure 1. The input feature space is then, comprised of two distinct components: a set of features ($F_1, F_2, \ldots, F_n$) that are not subject to any transformation and therefore directly presented to the Isolation Forest and a second component, a single feature, called on this project as Auto Error. This represents the sum of all reconstruction errors (mean absolute error) of the features ($X_1, X_2, \ldots, X_n$) that were compressed (encoder)

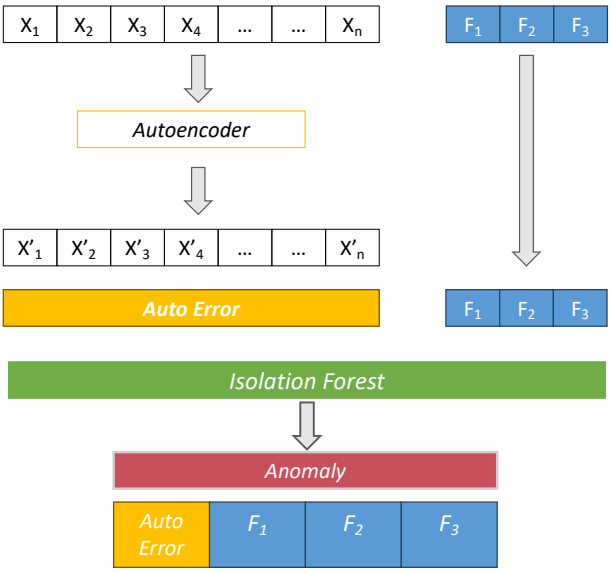

Fig. 1. UAD Model. F- Feature. X – Features at the input layer level of the auto encoder, which will be subjected to encoding and decoding. X' – Reconstructed features at the output layer of the auto encoder. Anomaly – Claim classified as anomaly by the Isolation Forest algorithm. Auto error – sum of all reconstruction errors (Mean absolute error) of the features that are processed on the auto encoder.

and reconstructed (decoder) in the autoencoder. At the Isolation Forest level, the Auto Error is then combined with the non-transformed features (F_1, F_2, …, F_n), which will then classify the observations according with the probability of being anomalous. This *'nested'* architecture presents a challenge when attempting to implement an explainability technique such as SHAP. The existence of a feature derived from an autoencoder (*Auto Error*) complicates the direct interpretation, particularly within the context of our primary objective, which is to enhance the interpretability of the UAD model as a tool for assisting in the claim auditing process. When performing a local interpretation with SHAP in a specific observation, especially in those where the Auto Error plays a role in the anomality of the observations it becomes unclear the meaning of this feature in that context. As the output of the UAD mode serves as an additional tool for the auditing team, it is important that the information it contain is easily interpretable and adds tangible value to the auditing process.

To tackle this problem there was the need to build a framework that: first, could help in the interpretation of the output of the UAD model including the original features that contributed to the specific value of the Auto Error; and second, implement it in a time efficient way. To approach this problem, we designed a framework that could explain this 'nested' explanation in a two-level process, as illustrated in figure 2.

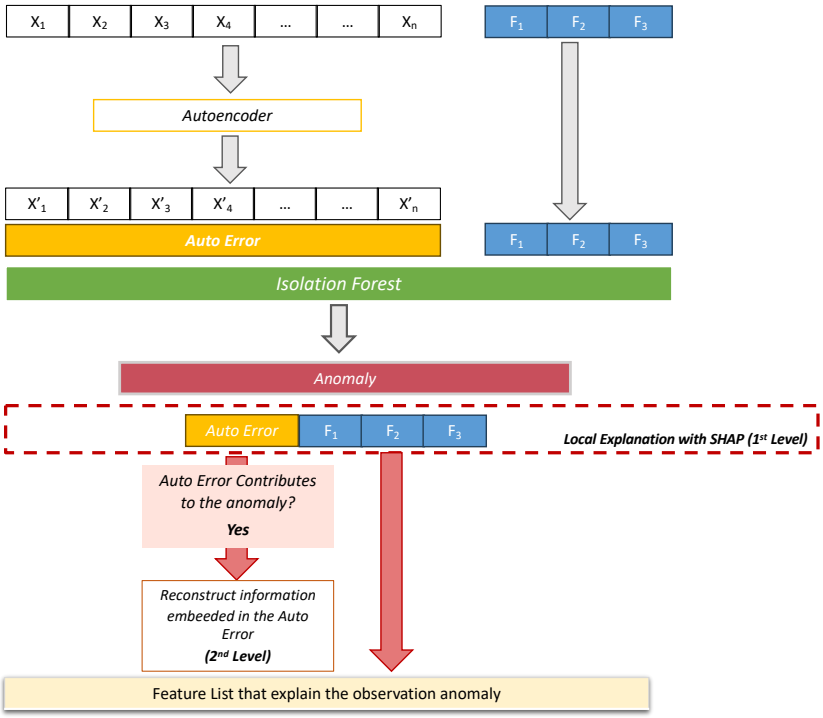

Fig. 2. Proposed Explanation Framework. F- Feature. X – Features at the input layer level of the autoencoder, which will be subjected to encoding and decoding. X' – Reconstructed features at the output layer of the autoencoder. Anomaly – Claim classified as anomaly by the Isolation Forest algorithm. Auto error – sum of all reconstruction errors (Mean absolute error) of the features that are processed on the autoencoder.

In a first level, when the UAD model classifies an observation as anomalous, a local explanation is conducted using SHAP to reveal the contribution of each feature to the anomaly classification. In instances where the Auto Error plays a role in explaining the anomaly, a secondary level is employed to identify the original features that contributed to the specific Auto Error value, and consequently, the anomaly classification. The output of the framework is a list of features sorted by their importance in explaining the observation's anomaly classification. Constructing this framework required addressing two fundamental questions: firstly, where to apply SHAP to efficiently explain the UAD model's output, and secondly, how to reconstruct the information embedded in the Auto Error.

This project is then developed in two steps. The first step is dedicated to the framework construction. The first sub-step was to understand the most efficient way to explain the UAD model output (the output of the Isolation Forest model). Our hypothesis was that the direct implementation of SHAP in all the UAD model (involving autoencoder and anomaly detection steps) would result in increased computational expense compared to using SHAP solely at the anomaly detection (Isolation Forest) step. Additionally, both methods were expected to exhibit a substantial overlap in their explanations. As for the reconstruction of the information contained in the Auto Error, two methods were considered: the first, by calculating the contribution of the primary features using SHAP; and the second method by calculating the error reconstruction of the primary features between the output and in the input layer of the autoencoder. The hypothesis is that error reconstruction method is less computational demanding while both methods provide similar explanations. This hypothesis is also

supported by *Antwarg et al* (2021) when stating that *'that in most of the cases, the features with the highest SHAP values were the same as the features with the highest reconstruction errors'*.

The second step of this project is dedicated to the evaluation of explanation framework previously built, using an audit claim specialist assessment on real data. The initial hypothesis was that the presence of SHAP explanations on the information provided to the audit team would result in a significant reduction in the time required for analyzing each claim. It was also hypothesized that the explanations produced by the framework, would not directly impact the UAD models performance. However, these explanations could offer insights about the output, which could then be used, in the future, to further tune the model.

2. LITERATURE REVIEW

As this project uses an explainability technique (SHAP) for enhancing the understanding of an unsupervised anomaly detection model, in which, part of its feature space is compressed by an autoencoder, this literature review delves into these two central subjects: eXplainable Artificial Intelligence (XAI) techniques and Autoencoders.

In the first chapter, titled 'Explainable Artificial Intelligence (XAI) and Shapley Additive Explanations (SHAP)', we start by addressing the *'black box*' paradigm in machine learning and its potential role in hindering the widespread adoption of AI technologies. We then introduce the emergent research field of XAI, which seeks to mitigate this issue. Subsequently, we present a current taxonomy for the existing XAI methods found in the literature, with a specific focus on the model-agnostic techniques, namely LIME and SHAP. At the same time, we highlight examples of their application, on the context of anomaly detection and insurance sector.

In the second chapter, 'Autoencoders', we explore the current definition, architecture and the several applications of autoencoders with a particular focus on dimensionality reduction. This chapter ends with some academic literature examples of the use of explainability techniques, namely SHAP, in models where autoencoders are used.

2.1 - EXPLAINABLE ARTIFICIAL INTELLIGENCE (XAI) AND SHAPLEY ADDITIVE EXPLANATIONS (SHAP)

In response to the escalating volume of generated data, organizations across diverse industries, including the insurance sector, are actively attempting to close the gap between the data produced and the data used. This effort is driven by the pursuit of not only a competitive edge, but also driven by the objectives of improving their existing business, customer experience and satisfaction *(Khan et al, 2014).*

Artificial intelligence (AI), notably through machine learning (ML), stands at the forefront of the current data revolution. Projections for the impact of AI in both American and European markets over the next decade are expected to reach billions of dollars. However, according to a recent report from *Capgemini Research Institute* (2022), just 18% of insurance companies had the necessary tools, technologies, personnel, processes, skills, and organizational culture to fully leverage the increasing volume of available data. Despite these challenges, ML emerges as pivotal field in pursuing the maximization of data utilization in the field of insurance. It plays a significant role in numerous applications, including personalized and dynamic pricing (*Grize et al.*, 2020; *Kaushik et al., 2022*), automating underwriting, customer segmentation, fraud and abuse detection, client churn prediction, life event marketing, personalized client product recommendation systems, client chatbots, and various other applications, as described by *Singh and Chivukula* (2020).

While ML plays a pivotal role in steering the AI revolution, a recurring impediment to its wider adoption is the issue of transparency within AI/ML-based systems. These are usually portrait as '*black box models'*, since its often hard or almost impossible for the system '*to provide a suitable explanation for how it arrived at an answer*' as stated by *Adadi and Berrada*

(2018). The issue of the lack of transparency of some AI/ML systems holds profound implications not only for developers and researchers who encounter challenges while attempting to improve these systems, but also to the end-users of these models, whose lives can be influenced by the predictions of this automated decision-making systems. This latter aspect holds exceptional significance and has been a focal point in the development of regulatory frameworks, exemplified by the General Data Protection Regulation (GDPR). The demand for explanations from AI-automated decision systems remains at the heart of the ongoing and robust debate surrounding the societal impact of AI, as discussed by *Ebers* (2021) on his work.

With the goal of tackling the '*black box*' paradox, a scientifical research movement named Explainable Artificial Intelligence (XAI), had been the subject of an increasing interest and development over the past two decades. Since 2016, XAI has gained significant momentum and has been progressively applied across a diverse spectrum of domain fields. Its main goal is to enhance the transparency and trust of AI-based systems, enabling human understanding, while attempting to not compromise the model's learning ability, and thus prediction power. Therefore, XAI has found applications in multiple fields, such finance and credit risk *(Gramegna and Guidici, 2021, Bussman et al, 2021)*, the militar field (*Luotsinen et al, 2019*), transportation systems (*Monje et al, 2022*) and even healthcare *(Stenwig et al, 2022, Adadi and Berrada, 2018)*.

Different taxonomies have been proposed to classify the interpretability methods in ML. In this work, the presented taxonomy follows the framework established by *Molnar* (2022). This author makes a distinguish between intrinsic interpretability and post-hoc (model-agnostic) interpretability. Intrinsic interpretability refers to the inherent characteristics of models that exhibit a simple structure, such as decision trees. The term *'post-hoc interpretation*' refers to methods applied after the model training, which can be model-specific (Model-Specific Interpretation), such as *DeepLift* for neural networks, or can be used with any ML model after it has been trained (Model-Agnostic Techniques). It's worth mention that model-specific interpretation is not exclusively to the post-hoc techniques since is a concept that also connected with intrinsic interpretability. One last criterion, which is extremely important, is the scope of interpretability. This distinguishes, methods that enable the explanation of individual observations/predictions (local interpretability) from methods that enable the explanation of the entire model (global interpretability). While global interpretability offers an understanding on how the model makes predictions from a global perspective, local interpretability takes interpretability one step further, allowing us to zoom in and identify how the feature space contributed to a specific prediction. As this work focuses on local interpretability, the next paragraphs will be dedicated to this topic. For a more comprehensive and exhaustive reading on global interpretability strategies and methods, please refer to *Adadi and Berrada* (2018) and *Molnar* (2022), particularly the chapter on *'Global Model-Agnostic Methods*'."

In the local model-agnostic methods category, there are different approaches available such as Individual Conditional Expectation (ICE), Local-Interpretable Model Agnostic

Explanations (LIME) and SHapley Additive exPlanations (SHAP), among others. ICE which mimics the concept of partial dependent plot in an individual observation level, allows the visualization of *'the functional relationship between the predicted response and the feature for individual observations*' (*Goldstein et al, 2017*). Among all the methods available, LIME and SHAP are the ones which are more commonly used to explain individual predictions. While LIME and SHAP share some common methodological concepts, their implementations differ, as described below.

LIME, as proposed by *Ribeiro et al* (2016), addresses the challenge of local interpretability by training model approximations of the base model for the individual predictions (local surrogate models) that need to be explained. According to *Molnar* (2022), LIME recreates variations of the selected observation and then feeds them to the base model to obtain predictions for these perturbed samples. LIME subsequently assigns weights to these new samples based on their proximity to the selected one. Finally, it trains a local surrogate model, typically a decision tree or a lasso regression, using these perturbed samples. Finally, LIME produces the explanation by interpreting the local model it created. According to the same author, LIME is still in the development phase, primarily due to the neighborhood definition, a parameter for choosing the instances around the one selected to be explained, which in its opinion, remains an unsolved problem for LIME in tabular data. He also states that this can lead to unstable explanations, being advisable to experiment with different kernel settings for each application to determine if the explanations are meaningful.

This contrasts with the more theoretical robust approach of SHAP, a popular XAI and local model-agnostic method developed by *Lundberg and Lee* (2017). SHAP is based on the game theory concept of Shapley values, which were introduced by *Lloyd Shapley* in 1953. Shapley values provide a theoretical framework for allocating rewards to participants in a given game based on their proportional contribution to the game's result or profit. If we consider the prediction as the reward, the Shapley values helps explain what is the contribution that each feature had for a specific prediction of a specific observation. This is achieved by computing the marginal contribution of a feature value across all possible coalitions, meaning all the possible feature combinations within an observation (*Molnar*, 2022).

According to the same author, the calculation of Shapley values follows the algorithm below:

1. Choose an instance x, the feature j, and set the number of iterations m.
2. In each iteration:
    - Select a random instance z from the dataset.
    - Generate a random order of features.
    - Create two new instances:
        - One, $x_{+j,}$ is the same as the instance of interest x, but with values after feature j replaced by values from sample z.

- The other, x_{-j}, is similar to $x_{+j,}$ but with feature j replaced by the value for feature j from sample z.
- Calculate the prediction differences for each iteration:

3. Average these differences over all iterations to obtain the feature' Shapley value.
4. Repeat this procedure for each feature to compute all the Shapley values.

As the dataset size increases, the computation needed for the calculation of the Shapley values becomes very expensive. In response to this challenge, *Lundberg and Lee* (2017) introduced an innovative method known as KernelSHAP, which leverages a kernel-based approach to enhance the efficiency of Shapley value calculations. Together with an optimized method for ensemble of trees TreeSHAP, contributed to the recognition and adoption of SHAP framework across multiple fields.

The general idea of KernelSHAP is to explain predictions made by machine learning models, by generating random coalitions of features, computing their contributions to predictions, and fitting a weighted regression model to approximate the Shapley values. Accordingly with Molnar (2022), KernelSHAP follows the approach described in the steps below:

1. Generate Coalition z: Random generated and in which represents a set of features. Each feature can be either absent (0) or present (1).
2. Convert z to the Original Feature Space: When a feature in z is absent (0), the corresponding values are replaced by randomly sampling values from the dataset.
3. Input the transformed feature set into the ML model to obtain the prediction. This prediction is noted as $f(h(z))$.
4. The step above is repeated multiples times to obtain multiples values of $f(h(z))$.
5. Compute the weight for each z.
6. Build the weighted regression model using the weights obtain previously. This model aims to fit the Shapley values, as Shapley values are additive.
7. Optimize the loss function, which lead to the calculation of the Shapley values, noted as Φ, which represents the contributions of each feature into the final prediction.

2.2 - AUTOENCODERS

Autoencoders are a specific type of artificial neural networks, used in different fields, such as image classification, natural language processing, anomaly detection, network security and financial analysis (*Li et al., 2023*). Autoencoders possess versatile applications owing to their architecture, encompassing dimensionality reduction (*Wang et al., 2016*),

image and video compression (*Habibian et al., 2019; Pessoa et al., 2020*), image noise removal (*Bajaj et al., 2020*), and anomaly detection (*Antwarg et al., 2021*). The primary structural objective of the autoencoders is to acquire a condensed and lower-dimensional representation of input data, while trying to minimize the reconstruction error between the input and the output data (*Li et al, 2023*). Autoencoders are categorized as unsupervised learning methods due to their reliance on unlabeled data during the training phase. Despite ongoing research leading to various architectural advancements, the fundamental structure of the autoencoder has remained consistent. It comprises two integral components: the encoder and the decoder. In the encoder phase, the input feature space learning takes place, typically involving data compression and consequent dimensionality reduction. Subsequently, during the decoder phase, the low-dimensional data is reconstructed to its original space or representation. The simplest architecture comprises an input and output layer with an equal number of neurons, along with a hidden layer. When the hidden layer contains fewer neurons than the input layer, compression occurs, and this layer is often referred to as the 'bottleneck' or 'sparse structure' (*Li et al, 2023*).

The autoencoder stands out from other dimensionality reduction methods, such as principal component analysis or linear discriminant analysis, primarily due to its inherent non-linearity (*Wang et al., 2016*). This non-linearity is attained through the utilization of non-linear activation functions, such as the sigmoid function and hyperbolic tangent function (*tanh*) (*Li et al, 2023*).

Autoencoders do exhibit certain limitations, including extended training durations and limited interpretability of low-dimensional features. Over the past decade, intensive research has been dedicated to autoencoders, particularly at the hidden-layer level. This research has resulted in diverse architectures variations and corresponding applications. Notable examples include denoising autoencoders, designed to filter noisy data, enhancing the network's ability to capture underlying data patterns and relationships. Additionally, convolutional autoencoders have emerged, bringing together the strengths of both convolutional neural networks and autoencoders (*Li et al, 2023*). In the scientific literature, there can be found several examples of the use of XAI techniques, particularly SHAP, to elucidate the outputs of autoencoders. However mostly of these works (*Roshan and Zafar, 2021; Gnoss et al, 2022; Antwarg et al, 2021)* make use of SHAP to explain the output of an autoencoder when this is used as anomaly detection model, rather than a dimensionality reduction technique. One work (*Júnior and Eler, 2022*), use SHAP to explain the use of autoencoders as a dimensionality reductor, however in this work the focus is the global interpretability and not the local one. To the best of our knowledge, there is no published work that related the use of SHAP to explain an unsupervised anomaly detection model that incorporate autoencoders as dimensionality reductors of part of its input space.

3. METHODOLOGY

3.1 - FRAMEWORK CONSTRUCTION

3.1.1 - Experience 1 – Where to use SHAP?

Python's SHAP implementation, make use of the Explainer class (`shap.Explainer`) in order to produce the explanations of the desired ML model. It works by computing the Shapley values. In order to generate the explanations' computations, this class (`shap.Explainer`) requires a function or a model object to be passed as a mandatory parameter together with dataset *(SHAP documentation, 2018).*

The *UAD model* in study, can be considered a 'nested' model since it makes use of two distinct algorithms in its pipeline (namely the autoencoder and the Isolation Forest). This led to an increased model complexity with multiple layers of processing. So, in order the explain the output of this model, the first approach would be to wrap all the UAD model and pass it as the mandatory parameter of the Explainer. However, the use of all UAD model as a model function in the Explainer to compute all the explanations, would be computational demanding and time consuming. The alternative approach considered was to make use of the SHAP Explainer solely at the Isolation Forest step, by passing the Isolation Forest model function and its input data in the Explainer. Both approaches are displayed in fig. 3.

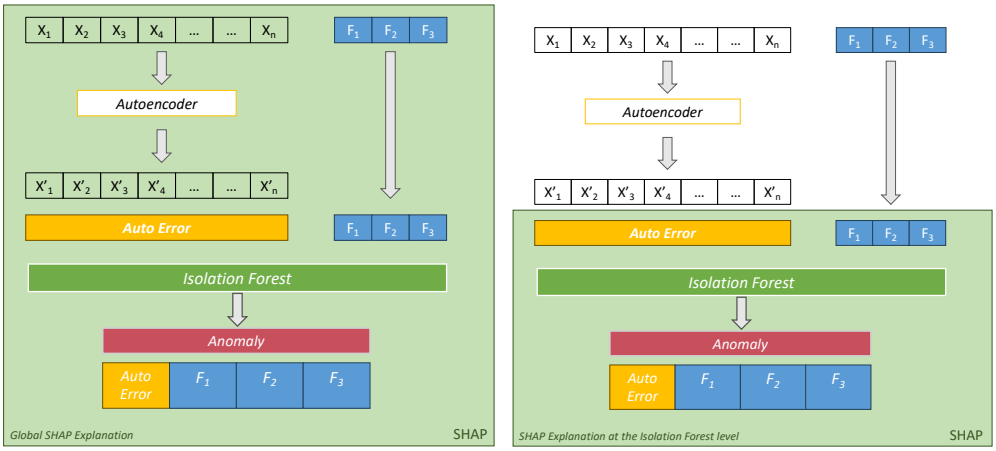

Fig. 2. Method A (*Global SHAP explanation*) and Method B (*SHAP explanation at the Isolation Forest level*)

Our hypothesis was that method B, by generating the SHAP explanations only at the Isolation Forest level, would be less computational expensive when compared with the first approach (where the entire *UAD model* is used to produce the explanations calculation), and that the explanations generated by both methods would have a significant degree of similarity. To test these two sub-hypotheses, two experiences were performed using two datasets of completely anonymized real claim data. The first dataset (*Dataset 1*) comprised of 66599 observations and the second (*Dataset 2*) comprised of 46823 observations. For both datasets, the SHAP explanations were calculated by using both methods: In method A, the all UAD

model was wrapped as a function and then passed as an the model parameter in the explainer (`shap.Explainer`); and in method B, only the Isolation Forest output function (Isolation Forest `decision_function`) was used as the model function parameter of the explainer.

To test the sub-hypothesis that method A was computational more demanding, the execution times of both method for Datasets 1 and 2 were registered while performing the SHAP explanations. The execution times of both methods in the two datasets, were compared using a simple ratio.

To test the sub-hypothesis that both methods are expected to exhibit a significant degree of similarity in their explanations, we use the same datasets (Dataset 1 and 2). The SHAP values calculated for each feature for each observation and using each method were stored. In a second step, these stored values were used to calculate the Pearson's correlation coefficient between the produced SHAP values for each feature of every observation in the tested datasets from both methods. By calculating the magnitude of the correlation coefficient between the explanations produced by both methods for the features within the tested datasets, we can obtain a quantitative measure of the similarity or dissimilarity between the two methods and assess the level of concordance in their explanatory outputs.

3.1.2 - Experience 2 – How to reconstruct the information contained in the Auto Error?

In the previous step (Experience 1), the goal was to determine what was the most efficient way to produce the SHAP explanations when using a model that has a nested architecture, as seen in the UAD model. However, when the feature Auto Error plays a role in explaining the observation's output, especially when this is classified as anomalous, this feature alone doesn't provide sufficient information. Since it originates from the compression of an autoencoder, the inclusion of this feature in the final information made available to the audit team is uninformative. Hence, it is crucial to reconstruct the information it contains, specifically to elucidate the role of each original feature in influencing the Auto Error value. Therefore, there was a need to develop a method that could explain this feature within the original autoencoder input feature space and integrate this method into an explanation Framework. This is particularly important in the context of the original project goal – increasing the efficiency of the current UAD model through the enhancement of its interpretability.

To determine the contribution of the original features in the Auto Error SHAP value, obtained after the anomaly detection step, we considered two different methods: In the first method (Method 1), SHAP explanations were employed to determine the contribution of each feature from the initial feature space to the Auto Error value. This was accomplished by utilizing the autoencoder model function as the required model function for the Explainer. In the second method (Method 2), it was used the error reconstruction method between the input and the output produced by the autoencoder. By identifying the original features associated with the highest reconstruction error, these features were used to explain their contribution to the Auto Error value. Our hypothesis is that both methods would yield similar

explanations. This hypothesis as formulated after considering *Antwarg et al* work in 2021, where SHAP was used for explaining the output of an autoencoder in anomaly detection task. They noted that '*in most cases, the feature with the highest SHAP values were the same as the features with the highest reconstruction error*'. In this experience four different datasets ($Dataset_A$, $Dataset_B$, $Dataset_C$, $Dataset_D$) which comprise of totally anonymized batches of real data obtained from the FH group databased were used. Each dataset consisted of approximately 40 000 claims and represented different temporal batches of claims that shared similar business characteristics.

In a first filtering step, there were only considered claims previously classified as anomalies by the UAD model and in which the Auto error was the single explanation factor considered. In a second step, after the previous filtration, the Auto error of the selected observations in the four datasets were subjected to the generation of the SHAP explanations (Method 1) and to the calculation of the error reconstruction (Method 2). In a third step, both lists resulting from Method 1 and Method 2 were compared. To avoid considering the noise introduced by the autoencoder during the decoding phase and focus exclusively on relevant features for the comparison method, we applied an extra filtering process. Only features falling within an interval of 5 standard deviations were considered in the distribution for each method. Among these, we retained only the top five features with the highest values for each method. Subsequently, the respective top feature sets from both methods were used to calculate the overlap coefficient (Szymkiewicz–Simpson coefficient), as described in Equation 1. This coefficient measures the similarity and overlap between two sets (*Vijaymeena and Kavitha, 2016*).

$$\frac{|\,Set1 \;\cap\; Set2|}{min\,(\,|Set1|,\;|Set2|\,)}$$

Equation 1. Overlap Coefficient

3.2 - FRAMEWORK EVALUATION

After the experience steps described above that led to the construction of the explanation framework, there was the need of evaluate its effectiveness. For this purpose, we sought the expertise of a domain specialist, specifically, a group insurance claim auditing specialist from FHG. His validation was instrumental in assessing the framework's results.

The goal of this evaluation step was to test the initial hypothesis that the inclusion of explanations produced by the framework, on the information provided to the audit team, would result in a significant reduction in claims analysis time. Which, in turn, would indirectly increase the overall effectiveness of the UAD model process.

To ensure that the effect of generated explanations on the information could be studied, only 50% of the cases identified as potential anomalies, were randomly assigned to contain the explanations generated by the framework, whereas the other 50% were sent without any explanations other than UAD model classification. The aim was to facilitate comparability between the two groups of cases and enable the study and testing of the effects

of the explanations using a control group (the group of data without explanations). Claims were submitted for audit on a weekly basis over an 8-week period, spanning from the eighth to the fifteenth week of the year 2023. In total, 234 claims were generated and submitted for audit. Of these, 115 included explanations produced by the framework, while the remaining 119 were submitted without any explanations.

Several indicators were evaluated and compared between both groups (with and without explanations). To test the hypothesis that the presence of the explanations could reduce the claim analysis time in the auditing process, the specialist was tasked to record the time taken to individually analyze each sent case, irrespective of whether it had been submitted with or without explanations. The recorded times for both groups were then later subjected to statistical analysis in order ot ascertain whether there was any significant difference.

Furthermore, the specialist was requested to o assess the utility of explanations provided with the submitted cases, assigning a score of 1 for *'Yes'* and 0 for *'No'* based on their usefulness in the claim analysis and decision-making process. This qualitative assessment provides a subjective dimension to the evaluation, helping us understand the value added by the explanations in the decision-making process.

Finally, the impact of the framework explanations on the performance of the UAD model was systematically examined, aiming to challenge the hypothesis that these explanations would not directly influence UAD precision but rather facilitate further model tuning by offering insights into the model output. An assessment was conducted by seeking the definitive classification of submitted cases based on the specialist's evaluations. This comprehensive evaluation not only allowed us to assess the immediate impact on precision, but also provided insights into its performance over time. Precision, which represents the number of cases correctly identified as anomalies among all cases identified as anomalies, was tracked on both a weekly and cumulative basis. This method allowed for a nuanced analysis of the stability and effectiveness of the framework's explanatory capabilities, providing valuable insights into its contribution and impact on the overall performance of the UAD model.

3.3 - TOOLS AND TECHNOLOGIES

In the course of this project, the computational experiments described above, were conducted on a standard laptop equipped with an Intel Core i7 2.80GHz processor and 32GB of RAM. The experiments were implemented using Python 3.8, and the SHAP package version 0.41.0.

4. RESULTS

4.1 - FRAMEWORK CONSTRUCTION

4.1.1 - Experience 1 – Where to use SHAP?

In this experience, it was tested the hypothesis that method B (generation of SHAP explanations at the Isolation Forest level) would be less computational expensive when compared with Method A (using all UAD model to produce the SHAP explanations), and that the explanations for both methods would have a significant degree of similarity.

So, in the first part of the experience where the execution times for both methods were compared for the two different datasets (Dataset 1 and 2), Method B was significant faster than the first method, as displayed in table 1.

Table 1 – Execution time of explanations generated by Method A (SHAP global explanations) and Method B (SHAP) explanations at the Isolation Forest Level) for Dataset 1 and Dataset 2

	Method A *SHAP global explanations*	**Method B** *SHAP explanations at the Isolation Forest level*
Dataset 1 *(N=66599)*	27:34:54	00:05:24
Dataset 2 *(N=46283)*	6:01:56	00:01:50

For the first dataset (Dataset 1), Method A took approximately 27h30 min to generate the calculations, while Method B generated the same number of calculations in 5mins and 24 seconds, which was 330 times faster than the first method for this dataset. As for the second dataset, Method A took approximately 6h to generate the explanations, while the second method perform the same task 200 times faster, taking roughly 2 minutes to perform the calculations.

In the second part of this experience, it was tested the sub-hypothesis that the explanations generated by both methods would show a significant similarity. The results for both datasets, show a very high correlation (over 0.9) between the SHAP values generated by both methods of the features used at the Isolation Forest level.

Table 2 – Correlation between the SHAP values generated by both methods used at the Isolation Forest level for two tested datasets (Dataset 1 and Dataset 2).

	Dataset 1 *(N=66599)*		**Dataset 2** *(N=46283)*
	Correlation between Method A and B		***Correlation between Method A and B***
Feature A_1	0.962	*Feature B_1*	0.984
Feature A_2	0.996	*Feature B_2*	0.997
Feature A_3	0.997	*Feature B_3*	0.996
Feature A_4	0.989	*Feature B_4*	0.968

4.1.2 - Experience 2 – How to reconstruct the information contained in the Auto Encoder?

In this experience it was tested the hypothesis that both methods (Method 1 that explain the contribution of each original feature in the Auto Error by calculating the Shapley values, and Method 2 that obtain the contribution by calculating the reconstruction error) would yield concordant results. For the datasets tested, and using only anomalous observations in which the Auto Error was the only feature that explained the observation's anomaly, the contributing original features perceived by both methods overlap significantly. The overlap coefficients were always greater than 0.5 for all four tested datasets as represented in Table 3. The minimum overlap coefficient value (0.774) was obtained in $Dataset_B$ and the highest (0.940) in $Dataset_C$.

Table 3 – Overlap coefficient between the contributing features of both methods for each observation's Auto error in the tested datasets

	$Dataset_A$ *N=495 after filtering*	**$Dataset_B$** *N=488 after filtering*	**$Dataset_C$** *N=580 after filtering*	**$Dataset_D$** *N=506 after filtering*
Overlap Coefficient	0.839	0.774	0.997	0.940

4.2 - FRAMEWORK EVALUATION

The first hypothesis tested was that the presence of SHAP explanations on the information provided for further auditing would result in a significant reduction in the claim analysis time. As mentioned previously, to conduct this test, the specialist was asked to record the time taken to individually analyze each claim, regardless of whether it had explanations or not. These times were recorded and analyzed both on a weekly and cumulative basis. In a cumulative analysis covering the 8 weeks of observations, the group of data sent with explanations generated by the framework took 7 seconds less to analyze per case, resulting in a cumulative average of 1 min and 17 seconds per case, as illustrated in figure 4. In contrast, the group of data sent without explanations required an average of 1 minute and 24 seconds per case.

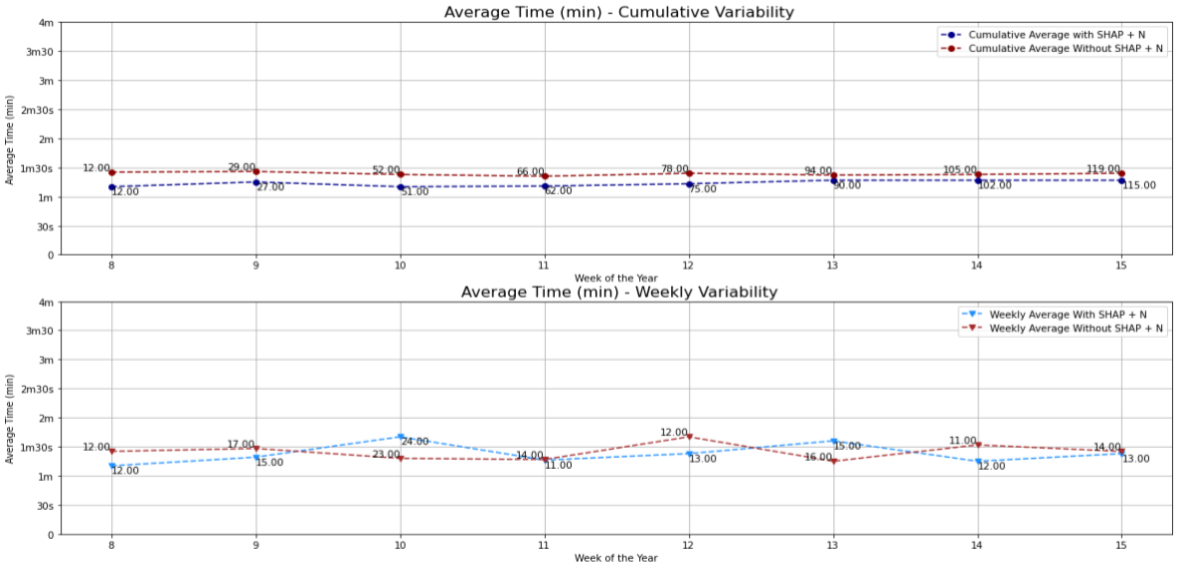

Figure 3 - Average time (in minutes) for individual claim auditing over time: weekly cumulative and weekly averages. N – Number/frequency of cases.

The weekly averages display some variability over time, with certain weeks in which the group without explanations show lower analysis times. Conversely, cumulative analysis demonstrates that the group with explanations required less time per case compared to the group without explanations. As mentioned before, the group of data with explanations cumulatively took in average, less 7 seconds to analyze per claim. The Mann-Whitney statistical test (p-value = 0.052, test statistic = 2504) suggests that, although the p-value is marginally higher than the conventional significance level of 0.05, there is a tendency, for this 7 second difference to be statistically significant.

In line with the methodology described earlier, the specialist was tasked with evaluating the usefulness of explanations accompanying the cases submitted for auditing. He provided a binary assessment by assigning a score of 1 for '*yes*' and 0 for '*no*,' indicating whether he considered the included explanations to be useful in the analysis and decision-making process. The results indicate that, overall, the specialist considered the explanations included in the information to be useful. As depicted in Figure 5, explanations accompanying 88% of the cases were deemed useful in the analysis process, while 12% were considered useless.

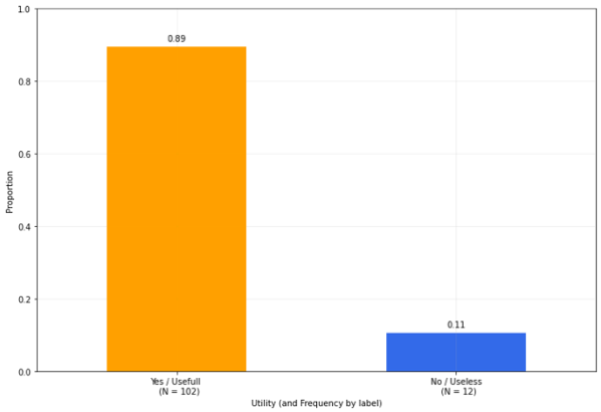

Figure 4 – Usefulness of the explanations generated by the Framework in the analysis and decision process. N – Number/Frequency of cases by label. N_{total}=114

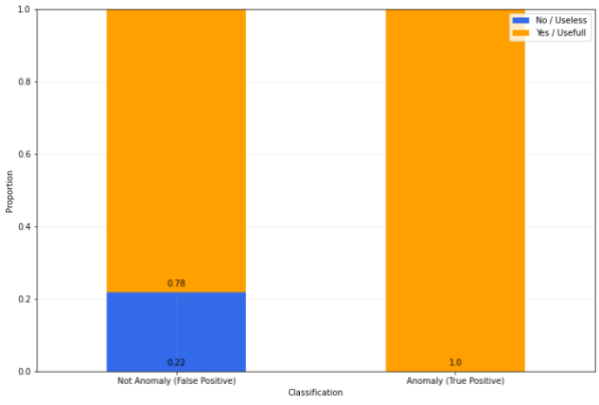

Figure 6 – Relationship between claim' classification and usefulness classification by the auditing specialist.

When we examine this relationship with the anomaly classifications performed by the specialist, as portrayed in Figure 6, it becomes evident that all explanations deemed useful belong to a group of claims classified as true anomalies. In contrast, for claims classified as false anomalies, a significant portion (22%) had explanations that were considered useless.

To examine the hypothesis that the presence of the framework's explanations would not directly impact the precision of the UAD model, the precision was calculated on a weekly and cumulative basis over an 8-week period of observations for both groups of data (with and without explanations). From a cumulative perspective, in the end of the 8th week of observations, the group of data sent with explanations achieved a precision of 0.51, which was 0.03 higher than the group sent without any explanations. This trend persisted throughout most of the observation period, as illustrated in Figure 7. However, the same trend was not observed in the weekly perspective, which showed a significant variability with some inversions in precision between the two groups. Despite the cumulative perspective showing consistently higher precision rates for the group with explanations compared to the group without explanations, this difference did not reach statistical significance. The chi-square test results (p-value = 0.997, Chi-square statistic = 0.164) suggest that we cannot conclude that these differences did not occur by chance.

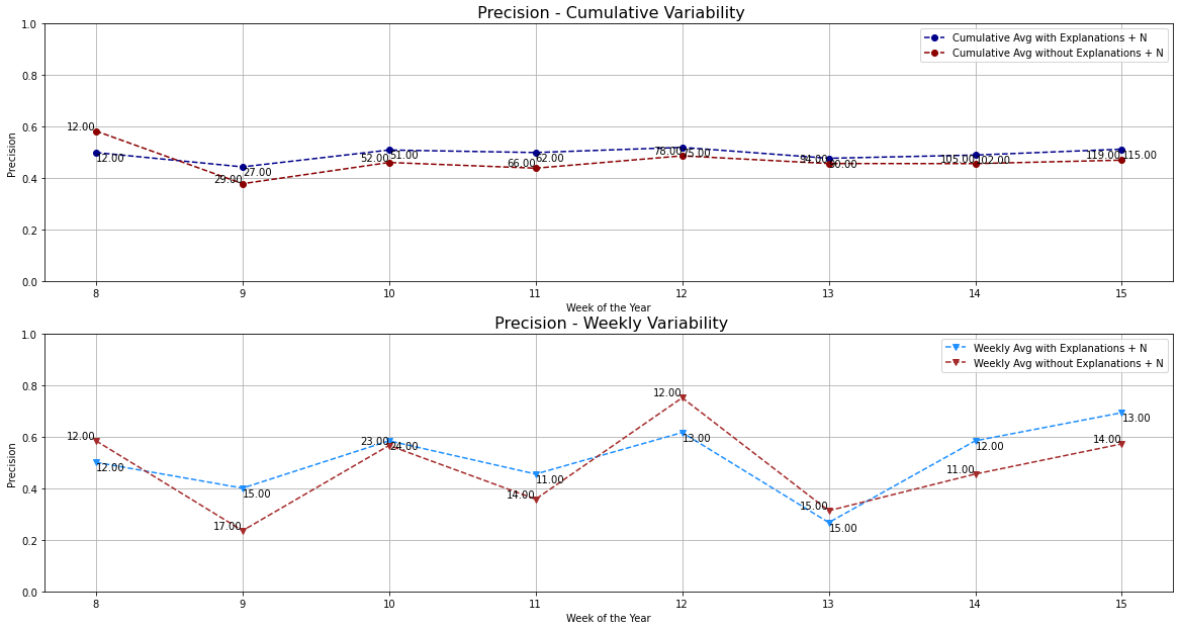

Figure 7 – Precision over time: weekly cumulative and weekly averages. N – Number/frequency of cases.

During the validation step and while receiving feedback from the specialist, as described above, it became noticeable that the explanations generated by the framework were helping in the identification of observations classified as anomalies by the model. However, some of these observations classified as anomalies by the UAD model were not true anomalies according with FH group own business rules, but only rare claims. By recognizing this, we started to remove these claims from the weekly batches' observations sent to the

specialist. This allowed us to calculate the UAD model precision considering this filtration and compare it with a scenario where these cases were not filtered and sent it to the specialist evaluation. This evaluation was conducted on both a cumulative and weekly basis, as illustrated in Figure 8.

By assuming that these rare claims would be classified as false anomalies, and by filtering them, the cumulative average of the group without the extra observations (Filtered data) outperformed the group-scenario where these observations were retained (All data), both on a weekly and cumulative basis. Once again, there is noticeable variability in precision on a weekly basis, but this time, there are no precision inversions between the two groups. This is even supported by the results of the Mann-Whitney test (p-value = 0.00015, U statistic = 64), that suggest that there is statistical difference between the precision between both groups.

It's also worth mentioning that in the test described above, comparing the effect of explanations on precision, we had already considered the filtration described here starting from the 9th week.

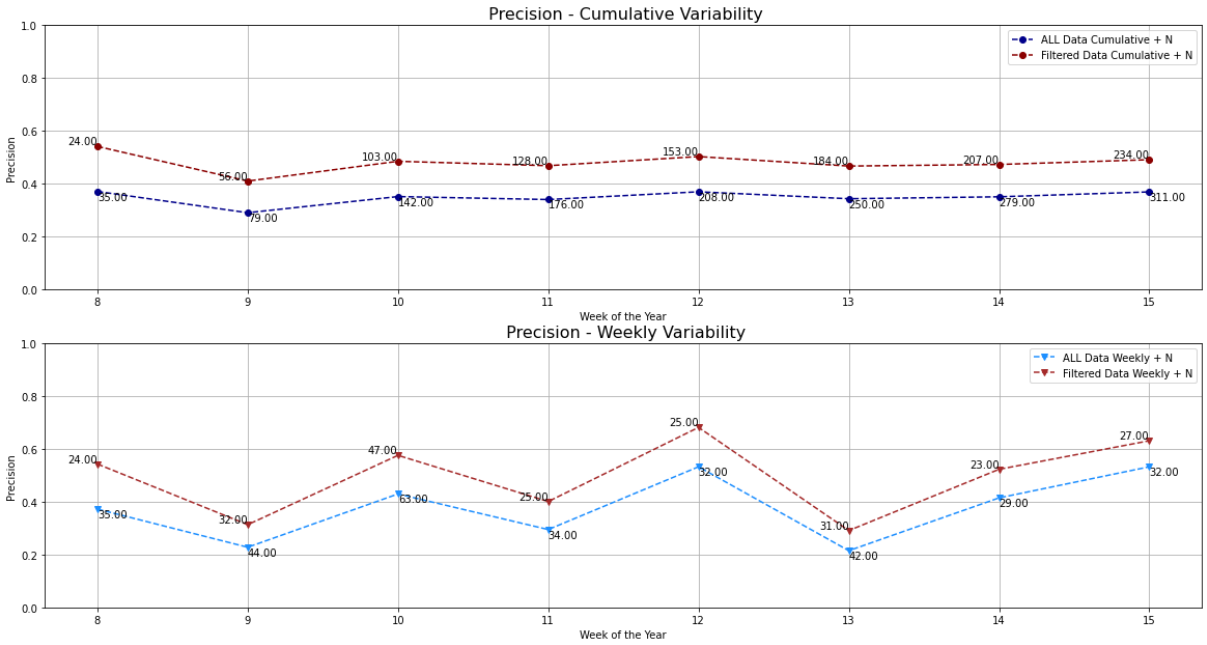

Figure 8 - Precision over time with and without data filtration: weekly cumulative and weekly averages. Filtered Data - group without the extra observations. All Data - group where all the observations were retained.

5. DISCUSSION

5.1 - FRAMEWORK CONSTRUCTION

5.1.1 - Experience 1 – Where to use SHAP?

The results from this experience confirm the initial hypothesis that performing the SHAP calculations at the Isolation Forest level in the UAD model provides a more time-efficient implementation without compromising the quality of the explanations generated. This assertion is supported by the strong correlation observed between the explanations generated using this approach and those produced when employing the entire UAD model in the SHAP explainer.

It is noteworthy that in this implementation it was employed the primary interface from the SHAP Python library (shap.Explainer), configuring the algorithm parameter to *'auto*.' This allows the Explainer to attempt to make the best choice given the passed model, accordingly with the SHAP documentation (2018). While the Isolation Forest, owing to its tree-based nature, appears to be a potential candidate for the explanation using the Tree SHAP algorithm (optimized for explaining the output of ensemble tree models), there was limited existing literature evidence available at the time of composing this discussion that demonstrated the advantages of employing this algorithm specifically with Isolation Forests in comparison to other algorithms. Some studies have utilized TreeSHAP with Isolation Forests, as exemplified by *Bergþórsdóttir* (2020). However, it is important to acknowledge the existence of contrasting viewpoints. *Anello* (2021), in a Towards AI post on the *'Interpretation of Isolation Forest with SHAP*,' expressed reservations regarding the effectiveness of the TreeExplainer—a method utilizing Tree SHAP—with Isolation Forests, suggesting that it "*doesn't work well with Isolation Forests*."

5.1.2 - Experience 2 – How to reconstruct the information contained in the Auto Error?

In this experience, we tested the hypothesis that both utilizing SHAP explanations and the Error reconstruction method, would provide similar information about the contribution of the original feature from the autoencoder input space in the Auto Error value. The results indicate substantial overlap in the explanations produced by both methods, with overlap coefficients consistently exceeding 0.5 in all the tested datasets. This finding aligns with the work of *Antwarg et al* (2021), who assert that *'in most cases, the feature with the highest SHAP values were the same as the features with the highest reconstruction errors*.' The determination of an acceptable threshold for coefficient overlap is subject to debate. In this study, a threshold of 0.5 was deemed the minimum for considering a significant overlap, leading to the conclusion that both methods offer significantly concordant explanations. It is worth noting that, although the execution time during this experiment was not formally

recorded, it was observed that the error reconstruction method was notably faster to compute compared to SHAP.

5.2 - Framework Evaluation

After completing the framework construction step, its output underwent evaluation by a claim auditing specialist. The specialist was tasked to evaluate and registered several parameters including claim analysis time, the usefulness of the explanations in the decision-making process, and the impact on the overall UAD model performance, namely the UAD precision.

Concerning claim analysis time, the initial hypothesis suggested a significant reduction with the inclusion of explanations. On average, each observation in the group of claims sent with explanations took seven seconds less to analyze. Although promising, the result from the Mann-Whitney test revealed a p-value (0.052), close but higher than the conventional significance level of 0.05. Considering the seven seconds difference across the 115 observations sent with explanations, this led to a total reduction of approximately 13 minutes and 20 seconds, constituting an 8% decrease in analysis time.

As for the usefulness of the explanations included in the information provided to the audit team, the overall results suggest that these explanations played a valuable role in the analysis process. It is essential to note that this metric holds a subjective nature and relies on the specialist's interpretation.

As for the impact on the explanations on the model performance, the obtained results also confirmed the initial sub-hypothesis that the inclusion of explanations in the information provided to auditing specialists would not directly enhance model performance. This was evident from the minimal difference in cumulative precision (0.03) between both groups, which lacked statistical significance (p-value = 0.997, Chi-square statistic = 0.164). However, the presence of framework explanations indirectly impacted the base model performance. As described earlier, during the validation step the presence of these explanations offered valuable insights about cases classified as anomalies. This was valuable in helping pre-identifying patterns of cases, classified as anomalies by the model solely due to their rarity, rather than anomalies by business definitions. The subsequent implementation of case filtration significantly enhanced the model performance, illustrating how SHAP can be utilized to help tune the model. In the literature there are some examples on how SHAP has been used to enhance indirectly model performance, such as the work of *Roshan and Zafar* (2021), in which SHAP was used to identify relevant features and subsequently retrain a new model based on these selected features. Or even the work of Arslan *and Lebichot* (2022), where the SHAP explanations of the output from one classifier, were used as input space to a second classifier.

6. CONCLUSION

In conclusion, the experiences conducted in this project have provided key insights regarding the construction of an explanation framework for the an unsupervised anomaly detection (UAD) model, based on an Isolation Forest and that uses an autoencoder to compress part of its impute feature space. From the results it was concluded, that for the UAD model case, the SHAP calculations only need to be performed at the anomaly detection level. This approach simplifies the implementation and expedites the explanation process. When the feature derived from the autoencoder, Auto Error, plays a role in explaining the UAD model's final output, the error reconstruction method emerges as a more efficient implementation compared to SHAP. This efficiency is supported by the consistent explanations achieved, as demonstrated by the substantial overlap in explanations between the two methods.

We can also conclude that, the explanation framework addresses the challenge posed by the nested architecture of the UAD model, by producing explanations that aggregate information from two levels, resulting in a comprehensive of features ordered by their contribution (SHAP values and reconstruction error) to the claim's anomaly classification. In essence, the framework generates the final output by explaining each individual claim or observation at the output of the anomaly detection level using SHAP. If Auto Error plays a role in explaining the anomaly, a second level is employed to identify the original features that contributed the most to the Auto Error value, based on reconstruction error method.

The evaluation steps of the framework, although failing to conclusively demonstrate a significant reduction in claim analysis time, it showed promising results that could have a relevance in a business practical context. The framework was considered helpful in the decision-making process by specialists. Additionally, it confirmed the hypothesis that the inclusion of explanations would not directly improve the UAD model's precision. However, a significant achievement was the framework's capacity to offer insights into observations classified as anomalies by the UAD model based on their rarity, rather than anomalies according to business-defined criteria. This unique capability facilitated model tuning by avoiding misclassification of such cases, ultimately contributing to enhance the model performance.

In conclusion, this research demonstrated that enhancing the interpretability of the UAD model indeed leads to improved efficiency, addressing the research question, *'Is it possible to increase the efficiency of the current UAD model by enhancing its interpretability with SHAP*?'. While the impact on reducing claim analysis time may be debatable in terms of its significance for increasing the efficiency of the UAD model, the undeniable effect of SHAP explanations on model performance, after model tuning through additional claim pre-filtration of rare but not anomalous claims, reinforces their role in improving efficiency.

The explanation framework employed in this study has proven to be an effective approach for addressing the challenge of explaining an unsupervised anomaly detection model, especially when part of the input space is compressed by an autoencoder. By shedding

light on the *'black box*' nature of such models, the framework contributes to the understanding and optimization of anomaly detection processes.

FUTURE WORK

In our opinion there are several avenues for future research from the findings of this study, namely:

- Testing the framework on a model where all the feature space is compressed by an auto encoder. It will give insights about the framework's adaptability to a broader range of data compression scenarios.

- Comparative analysis with TreeSHAP algorithm. Could be of great importance conduct an experiment comparing the implementation of this framework with an alternative approach involving the forced utilization of the TreeSHAP algorithm, theoretically optimized for ensemble of tress within the `shap.Explainer`. This will provide insights into the relative performance and effectiveness of these methods, particularly within the context of Unsupervised Anomaly Detection models.

- Testing the Framework in different business contexts, to ascertain the adaptability and practical utility of this framework in different business and data.

BIBLIOGRAPHIC REFERENCES

1. Adadi, A., & Berrada, M. (2018). Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). IEEE Access, 6, 52138-52160. doi:10.1109/ACCESS.2018.2870052

2. Alvarez-Melis, D., & Jaakkola, T. S. (2018). On the robustness of interpretability methods. arXiv preprint arXiv:1806.08049. Retrieved from https://arxiv.org/pdf/1806.08049.pdf

3. Anello, E. (2021, May 30). Interpretation of Isolation Forest with SHAP: An easy way to understand the most contributing features for anomaly detection. Towards AI. Retrieved from https://pub.towardsai.net/interpretation-of-isolation-forest-with-shap-d1b6af93ae71#c8e1

4. Antwarg, L., Miller, R. M., Shapira, B., Rokach, L. (2021). Explaining anomalies detected by autoencoders using Shapley Additive Explanations. Expert Systems with Applications, 186, 115736. doi:10.1016/J.ESWA.2021.115736

5. Arslan, Y., Lebichot, B., Allix, K., Veiber, L., Lefebvre, C., Boytsov, A., Goujon, A., Bissyandé, T.F. and Klein, J., (2022). Towards refined classifications driven by shap explanations. In International Cross-Domain Conference for Machine Learning and Knowledge Extraction (pp. 68-81). Cham: Springer International Publishing.

6. Bajaj, K., Singh, D. K., & Ansari, M. A. (2020). Autoencoders Based Deep Learner for Image Denoising. Procedia Computer Science, 171, 1535-1541. doi:10.1016/j.procs.2020.04.164

7. Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion, 58, 82-115. https://doi.org/10.1016/j.inffus.2019.12.012

8. Bora, A., Sah, R., Singh, A., Sharma, D., & Ranjan, R. K. (2022). Interpretation of machine learning models using XAI - A study on health insurance dataset. In 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) (pp. 1-6). doi:10.1109/ICRITO56286.2022.9964649

9. Capgemini Research Institute. (2022). The data-powered insurer: Unlocking the data premium at speed and scale. https://www.capgemini.com/wp-content/uploads/2022/03/Data-Masters-in-Insurance_FINAL_WEB-1.pdf

10. Ebers, M. (2021). Standardizing AI - The Case of the European Commission's Proposal for an Artificial Intelligence Act. In The Cambridge Handbook of Artificial Intelligence: Global Perspectives on Law and Ethics. Available at SSRN: https://ssrn.com/abstract=3900378

11. Elshawi, R., Al-Mallah, M. H., & Sakr, S. (2019). On the interpretability of machine learning-based model for predicting hypertension. BMC Medical Informatics and Decision Making, 19, 146. https://doi.org/10.1186/s12911-019-0874-0

12. Gee, J., Button, M., Brooks, G., & Vincke, P. (2010). The financial cost of healthcare fraud. Portsmouth: University of Portsmouth, MacIntyre Hudson. Retrieved December 20, 2010, from http://eprints.port.ac.uk/3987/1/The-Financial-Cost-of-Healthcare-Fraud-Final-(2).pdf

13. Gnoss, N., Schultz, M., Tropmann-Frick, M. (2022). Association for Information Systems Association for Information Systems XAI in the Audit Domain-Explaining an Autoencoder Model for XAI in the Audit Domain-Explaining an Autoencoder Model for Anomaly Detection Anomaly Detection. doi:10.23919/WI2022.2022.9552151

14. Goldstein, A., Kapelner, A., Bleich, J., & Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. Journal of Computational and Graphical Statistics, 24(1), 44-65.

15. Goodman, B., & Flaxman, S. (2017). European Union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, *38*(3), 50-57.

16. Gramegna, A., & Giudici, P. (2021). SHAP and LIME: An Evaluation of Discriminative Power in Credit Risk. Frontiers in Artificial Intelligence, 4. doi:10.3389/frai.2021.752558

17. Grize, Y. L., Fischer, W., & Lützelschwab, C. (2020). Machine learning applications in nonlife insurance. Applied Stochastic Models in Business and Industry, 36(4), 523-537.

18. Gupta, P. (2023). Leveraging Machine Learning and Artificial Intelligence for Fraud Prevention. SSRG International Journal of Computer Science and Engineering, 10(5), 47-52. https://doi.org/10.14445/23488387/IJCSE-V10I5P107

19. Habibian, A., van Rozendaal, T., Tomczak, J. M., & Cohen, T. S. (2019). Video Compression With Rate-Distortion Autoencoders. doi:10.1109/ICCV.2019.00713

20. Internal Social Security Association – ISSA (2022). "Fraud in Healthcare: Challenges and Emerging Technologies." Published July 4, 2022. Available from: https://www.issa.int/analysis/detecting-fraud-health-care-through-emerging-technologies

21. Kaushik, K., Bhardwaj, A., Dwivedi, A. D., & Singh, R. (2022). Machine Learning-Based Regression Framework to Predict Health Insurance Premiums. International Journal of Environmental Research and Public Health, 19(13), 7898. MDPI AG.

22. Khan, N., Yaqoob, I., Hashem, I. A. T., Inayat, Z., Kamaleldin, W., Alam, M., Shiraz, M., & Gani, A. (2014). Big data: Survey, technologies, opportunities, and challenges. The Scientific World Journal. Available at: https://www.researchgate.net/publication/312387451_Big_data_Survey_technologies_opportunities_and_challenges

23. Li, P., Pei, Y., & Li, J. (2023). A comprehensive survey on design and application of autoencoder in deep learning. Applied Soft Computing, 138, 110176. doi:10.1016/J.ASOC.2023.110176

24. Lundberg, S. (2018). SHAP documentation. Retrieved from https://shap.readthedocs.io/en/latest/index.html

25. Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In I. Guyon, U. von Luxburg, S. Bengio, et al. (Eds.), Advances in Neural Information Processing Systems, Vol. 30. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

26. Mihirette, S., & Tan, Q. (2022). SHAP Algorithm for Healthcare Data Classification. In Hybrid Artificial Intelligent Systems (pp. 363-374). Springer International Publishing.

27. Molnar, C. (2022). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable (2nd ed.). Retrieved from https://christophm.github.io/interpretable-ml-book/

28. Monje, L., Carrasco, R. A., Rosado, C., & Sánchez-Montañés, M. (2022). Deep Learning XAI for Bus Passenger Forecasting: A Use Case in Spain. Mathematics, 10(9), 1428. MDPI AG. Retrieved from http://dx.doi.org/10.3390/math10091428

29. Morris, L. (2009). Combating fraud in health care: An essential component of any cost containment strategy. Health Affairs, 28(5), 1351-1356. doi:10.1377/hlthaff.28.5.1351

30. NHCAA National Health Care Anti-Fraud Association. (n.d.). The Challenge of Health Care Fraud. Retrieved March 31, 2023, from https://www.nhcaa.org/tools-insights/about-health-care-fraud/the-challenge-of-health-care-fraud/

31. Pessoa, J., Aidos, H., Tomas, P., & Figueiredo, M. A. T. (2020). End-to-End Learning of Video Compression using Spatio-Temporal Autoencoders. In IEEE Workshop on Signal Processing Systems, SiPS: Design and Implementation (pp. 1-6). doi:10.1109/SiPS50750.2020.9195249

32. Petch, J., Di, S., & Nelson, W. (2022). Opening the Black Box: The Promise and Limitations of Explainable Machine Learning in Cardiology. Canadian Journal of Cardiology, 38(2), 204-213. https://doi.org/10.1016/j.cjca.2021.09.004

33. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Model-Agnostic Interpretability of Machine Learning. Retrieved from arXiv preprint arXiv:1606.05386

34. Roshan, K., Zafar, A. (2021). Utilizing Xai Technique to Improve Autoencoder Based Model for Computer Network Anomaly Detection with Shapley Add Explanation (SHAP). International Journal of Computer Networks and Communications, 13(6), 109-128. doi:10.5121/ijcnc.2021.13607

35. Serré, L., Amyot-Bourgeois, M., & Astles, B. (2021). Use of Shapley Additive Explanations in Interpreting Agent-Based Simulations of Military Operational Scenarios. In 2021 Annual Modeling and Simulation Conference (ANNSIM) (pp. 1-12). doi:10.23919/ANNSIM52504.2021.9552151

36. SHAP documentation, 2018 - Lundberg, Scott - Available from: https://shap.readthedocs.io/en/latest/index.html

37. Shapley, L. (1953). A value for n-person games. In H. Kuhn & A. Tucker (Eds.), Contributions to the Theory of Games (AM-28), Volume II (pp. 307-318). Princeton, NJ: Princeton University Press. https://doi.org/10.1515/9781400881970-018

38. Singh SK, Chivukula M (2020). A Commentary on the Application of Artificial Intelligence in the Insurance Industry. Trends Artif Intell. 4(1):75-79. Available from: https://pdfs.semanticscholar.org/69ee/1a5316d4bd5d131851bb947d61a8e407f25b.pdf

39. Stenwig, E., Salvi, G., Rossi, P. S., & Skjærvold, N. K. (2022). Comparative analysis of explainable machine learning prediction models for hospital mortality. BMC Medical Research Methodology, 22(1). doi:10.1186/s12874-022-01540-w

40. Tu, J. V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. Journal of Clinical Epidemiology, 49(11), 1225-1231. https://doi.org/10.1016/S0895-4356(96)00002-9

41. Vijaymeena, M. K., & Kavitha, K. (2016). A survey on similarity measures in text mining. Machine Learning and Applications: An International Journal, 3(2), 19-28.

42. Wang, Y., Yao, H., & Zhao, S. (2016). Auto-encoder based dimensionality reduction. Neurocomputing, 184, 232-242. doi:10.1016/J.NEUCOM.2015.08.104

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