
**MASTER IN**

DATA ANALYTICS FOR BUSINESS

**MASTER'S FINAL WORK**

INTERNSHIP REPORT

# Model Interpretability in Credit Insurance

ALESSANDRO
CONSIGLIO

March-2023


**MASTER IN**

DATA ANALYTICS FOR BUSINESS

**MASTER'S FINAL WORK**

INTERNSHIP REPORT

# Model Interpretability in Credit Insurance

ALESSANDRO
CONSIGLIO

**SUPERVISORS**:
JOÃO AFONSO BASTOS
BASILE CALDERAN

March-2023

*“I want to dedicate my work to*
*my dear family, who have been my constant source of*
*love, support, and encouragement.*
*Their unwavering belief in me*
*and sacrifices have made this achievement possible.*
*This thesis is a tribute to their love*
*and the values they have instilled in me.*
*Thank you for being my rock*
*and for always being there for me”*

# Contents

# List of Figures

# List of Tables

**Abstract**

The use of complex machine learning (ML) models has become increasingly popular in the credit insurance field due to their ability to accurately predict the probability of default. However, the lack of interpretability of these models has become a critical issue for businesses and regulatory bodies. This study focuses on the use of Explainable Boosting Machines (EBM) to develop an interpretable model for predicting the probability of default for buyers in credit insurance policies. The empirical analysis uses a dataset of credit insurance policies and compares the performance of the EBM model with state-of-the-art models in terms of accuracy and interpretability. The results show that the EBM model achieves high accuracy levels, comparable to the best-performing models, while maintaining a high degree of interpretability. The findings suggest that EBM can be a valuable tool for credit insurance companies to balance the need for accurate predictions with the need for transparency and accountability, in line with new policy restrictions.

***Keywords*—** Interpretability, Explainability, Explainable Boosting Machine, Shamrock

# 1 Introduction

Used responsibly, artificial intelligence (AI) and data technologies can help companies to make their operations more efficient as well as improve the customer experience. In insurance as well, artificial intelligence can be an advantage in several areas. Within claims, knowing the probability that they occur is a must. This is where AI can benefit both insurers and clients by detecting and preventing future claims. Claims analysis is the most important topic in the insurance business and artificial intelligence can learn automatically from past default cases, immediately apply the acquired knowledge and get better and better in identifying and preventing them with time. Moreover, having transparent and interpretable methods to do so, will be a competitive advantage among competitors in the industry, since machine learning interpretable techniques can reveal undesirable patterns in data that models exploit to make predictions, potentially causing harms once deployed.

## 1.1 Objectives of the Internship

In the Allianz Trade Group Data Analytics (GDA), a project named Shamrock is designed to analyzes and classifies buyers as well giving a score to them. Shamrock is already live in Italy and in other countries but it is based on another famous machine learning model called *Extreme* Gradient Boosting (XGBoost) [1]. The goal behind this internship is to:

- Test a new framework for Shamrock in order to maintain a good trade-off between accuracy and interpretability, considered vital in the industry.
- Use the Group Data Analytics Data Lake to extract all useful information related to claims and buyers as a part of the data pre-processing phase.
- Collect data through different departments from other branches, analyse, clean, and transform them in order to conduct accurate analysis and comparisons.

## 1.2 Allianz Trade

Allianz Trade (AZT) (*ex*. Euler Hermes) is a credit insurance company that offers a wide range of bonding, guarantees and collections services for the management of Business to Business (B2B) trade receivables. As a subsidiary of Allianz, AZT is rated AA by Standard & Poor's. The Group posted a consolidated turnover of €2,9 billion in 2021. Allianz Trade employs more than 5500 employees with 80 nationalities in over 50 countries and insured global business transactions for €931 billion in exposure at the end of 2021. The Allianz Trade GDA is part of the « Group Transformation » department, in charge of various B2B business innovations, digital product management and partnerships. The team reports directly to the AZT France Executive Committee. The GDA is based in France but works in close collaboration with the other branches of the group (Italy, United Kingdom, Germany, Hong Kong), which support and deploys data science initiatives across the group and helps data scientists to develop their machine learning skills. The aim is

to help the different Allianz Trade departments to make strategic decisions based on quantitative studies and data analysis.

From a solid and historical base in Europe, Allianz Trade has expanded throughout the world by integrating the national leaders in all the main credit insurance markets and by opening subsidiaries in new markets such as Asia and Latin America. This strategy has enabled the group to build up a dense network and offer consistent high-quality services on five continents. The 7 Regions are America, France, Germany Austria & Switzerland (DACH), Mediterranean countries Middle East & Africa (MMEA), Northern Europe (NEUR), Asia Pacific (APAC) and Allianz Trade World Agency (Created in 2008, Allianz Trade World Agency is a one-stop shop for multinationals, it provides global companies with a whole range of products and services tailored on their needs). In addition to Allianz Trade's main business area in credit insurance, the group has diversified its activities by providing companies with a full range of services regarding the protection of business transactions and assets. During my internship, I worked for Allianz Trade France and as part of the GDA, I was assigned to the the Model Development team who is in charge of building machine learning models.

## 1.3 Credit Insurance

The main business of Allianz Trade, credit insurance, is an insurance policy and risk management product offered by private insurance companies and government export credit agencies to businesses wishing to protect their receivables from losses due to credit risks such as prolonged default, insolvency, or bankrupt. Figure 1 shows the whole trade credit insurance cycle in three major actors, the insurance company, the policy holder and the buyer. For example, Company A, also called Policy Holder (PH) is selling goods for Company B, also called Buyer or Debtor. Company A is insured by Allianz Trade. The Company B is a new client. It has just ordered goods from the Company A for a total of €100 000. Company A asks AZT for a Credit Limit of €100 000 upon Company B. Allianz Trade checks its database where millions of businesses are monitored. It appears that Company B experiences some cash difficulties. Therefore, AZT decides to deliver a credit limit of €50 000. It means that in case of non-payment, Company A is secured by Allianz Trade up to €50 000. The risk underwriting department decides to grant credit limits or not. Company A decides not to take risks and informs its customer (Company B) that it will deliver goods for €50 000. Company B receives its goods. At the end of the payment term, Company B cannot pay Company A. Company A declares the Payment Default to AZT by gathering all the relevant documents needed (this is called a Claim). Allianz Trade will now officially take action against the debtor. Debt collection teams will try to get the money back from Company B with extra-judicial and legal actions. Allianz Trade manages to get €20,000 back from Company B. The final payment default amount is therefore €30,000. AZT will pay the claim on this basis.

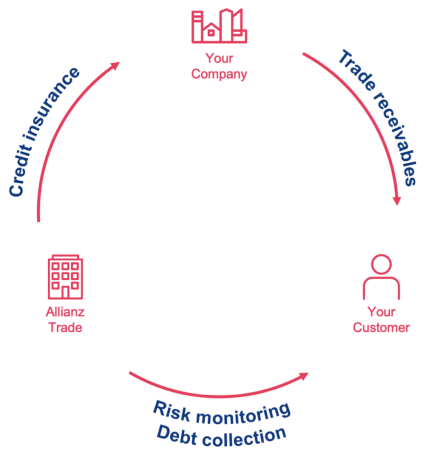


Figure 1: Allianz Trade Business Model

## 1.4 Allianz Trade GDA Main Projects

Allianz Trade has accompanied its clients to provide simpler and safer digital products, thus becoming a key catalyser in the world's commerce. It's very important to check the health of the company for the business of credit insurance. Allianz Trade manages more than 600,000 B2B transactions per month and performs data analytics from over 30 million companies worldwide. At-scale artificial intelligence and machine learning have become the heart of the business, Allianz Trade uses machine learning across a variety of use cases. In this section, I will present some of the main projects in general, including the two on which i worked directly 1.4.1 and 1.4.4

### 1.4.1 Shamrock

Shamrock is one of Allianz Trade's most important machine learning projects. It is first developed for Ireland in 2016. The aim is to provide a more powerful decision-making tool for the monitoring of credit-insurance exposures, while taking into consideration the actual requirements of the system (prediction accuracy, operational efficiency, understandable from a business perspective), the existing scoring framework, as shown in Figure 2, (rating scale from 1 to 10) and the future challenges in terms of data collection. The grading algorithm computes the grade on buyers with small sensitivity (S0/ S1) whereas credit risk analyst does analysis on larger buyer with high sensitivity (S2/ S3). Sensitivity is based on grade and exposure and the probability will represent the grade. The riskiness is function of a failure indicator given by official publications, for so the machine learning model also classifies buyers with a target feature namely failure or default indicator. The latter is a binary variable which describe the insolvency indicator setting 1 for the insolvent buyer and 0 otherwise.


Figure 2: Grading system of Allianz Trade

The Shamrock grading system has been first made in production in Ireland and then rolled out to other parts of the world. The result shows a very good impact on predictive performance which means more revenue and less loss for the company. The grading information is important to Allianz Trade since the main business is based on this system, and other machine learning projects also need the grading information as a parameter [2].

### 1.4.2 PRISM

Technical Pricing is an essential part of the Allianz Technical Excellence agenda. Technical Price is defined as the best estimate of the premium required for an individual insurance policy in order to achieve the long-term financial target of the Allianz Group. As of today, Market Management and Commercial Distribution teams use pricing tools to quote a trade credit insurance policy contract based on historical information or insurable turnover in addition to monitor policy profitability based on forward-looking estimates. Both tools are deemed to impact business units's profit loss, expected losses and contribution to capital requirements. Although, in the current stage, these tools have limited capabilities which prevent them from playing an active role in this particular context. Hence, PRISM's main aim is to generate the technical price for each policy. It allows us to ensure we are receiving the correct amount of premium so that we can cover the expected loss of a policy, the administration costs, and the cost of capital related to the capital requirement of the policy. It is important to note that the Technical Price is independent of any business plans, market conditions or strategic/tactical/individual pricing decisions. Technical price is neither a target price nor a commercial price, but rather a reference price to be established within the underwriting process. Any commercial considerations must come afterwards. Another objective of PRISM is to put sight on the policies that are harming the portfolio and which require pricing actions [3].

### 1.4.3 ENOLA

ENOLA is an anti-fraud solution designed to detect fraud on the Policy Holder side. Therefore, if we take the previous example explained in 1.3 and assume that Company A declares the payment default to Allianz Trade. Before taking action against the debtor (Company B), Allianz Trade must first ensure that the claim raised by Company A is real and not fraudulent and this is where ENOLA comes in. As previously stated, ENOLA is responsible for detecting fraud on the policy holder side. The same solution is available on the buyer's side and is called SHERLOCK. Figure 3 sums up the whole ENOLA pipeline.

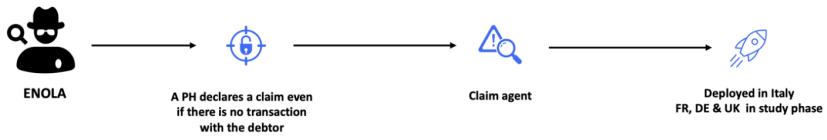


Figure 3: Enola Pipeline

As indicated in Figure 4, the strength of ENOLA lies in automating a heavy, risky, and time-consuming task which is manual monitoring of suspicious fraudulent claims especially that there's no systematic approach to detect suspicious fraud cases. That being the case, ENOLA helps minimizing Allianz Trade's financial loss and reputational risk by detecting suspicious cases in claim assessment at opening time, prioritizing fraud deep dive study by level of suspiciousness (Low, Medium, High) and most importantly monitoring automatically & continuously all claims regardless its amount.

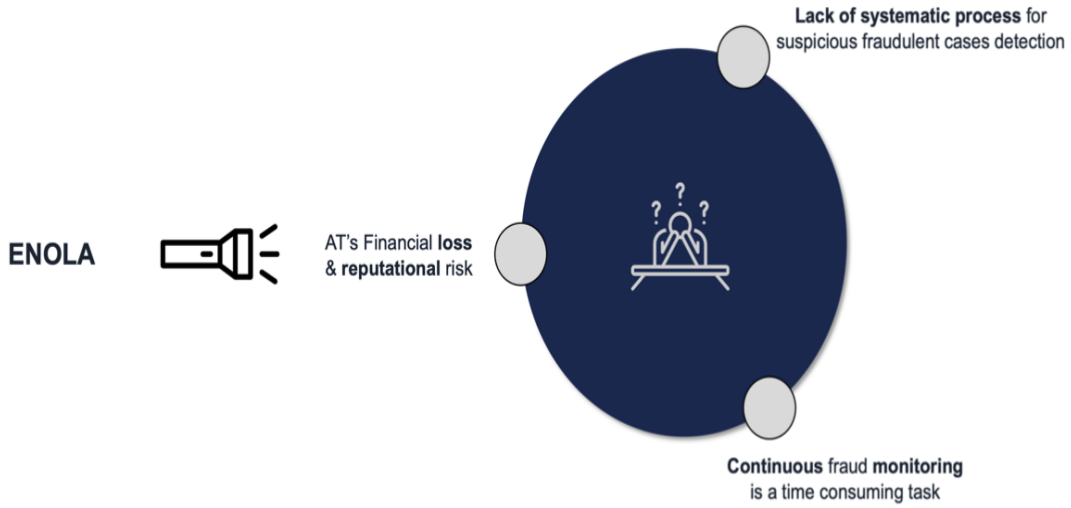


Figure 4: ENOLA's main strenght

### 1.4.4 ADT ++

ADT++ is a model designed to automate the underwriting process on top of the level currently provided by the Automatic Decision Threshold (ADT) model. It makes use of data grouped in 3 categories:

- Variables associated to the Buyer (i.e. that can be computed without a Credit Limit Request (CLR);
- Variables associated to the Request (e.g. requested amount, some ratios, etc.)
- Variables associated to the Policyholder performing the request (e.g. contract information, financials, . . . )

The model output is a score coming from a machine learning model, which is exactly the sum of sub-scores coming from the evaluation of all its input variables. When the score is above a parameterized threshold, then the request is validated, otherwise it is left for manual evaluation. The model has reduced the manual underwriting process by 30% and this is a crucial point for the business since they receive more than 200 request on average per day. Within the team has been created a simple dashboard based on the machine learning model used for the analysis where is possible to manually evaluate the score and define the limit on the variables for each buyer.

| # | Variable | Description | Type | Value | Score |
|---|---|---|---|---|---|
| 0 | RATIO_REQ_ADT | Ratio: CLR amount to ADT amount | continuous | 2 | 0.23 |
| 1 | REQT_M_EUR | Request amount (EUR) | continuous | 60000 | 0.33 |
| 2 | VAL_GRD_C | Buyer valid grade | continuous | 6 | 0.04 |
| 3 | EXPOSURE_M_EUR | Buyer exposure (EUR) | continuous | 493710 | -0.11 |
| 4 | ADT_M_EUR | Buyer ADT amount (EUR) | continuous | 30000 | -0.21 |
| 5 | WRK_IPG_MOTI_C | Job to do motive | categorical | MUW22 | -0.11 |
| 6 | FINANCIAL_ACTS_PRD_CLOSE_AGE | Duration since closing date of last financials | continuous | 414 | 0.20 |
| 7 | LEGAL_MEAN_EMP_Q | Buyer mean number of employee | continuous | 573 | 0.37 |
| 8 | PROD_FAM_TYPE_C | Contract family type | categorical | CRI | 0.01 |
| 9 | ADT_CREA_AGE | ADT creation age | continuous | 184 | 0.30 |
| 10 | TYP_GRD_C | Valid grade type | categorical | M | 0.26 |
| 11 | NUM_PH_WITH_LIMITS | Number of PH with limits on this buyer | continuous | 12 | 0.02 |
| 12 | LIM_CREA_AGE | Age of the limit (if any) | continuous | -1 | 0.06 |
| 13 | RATIO_REQ_PREV_LIMIT | Ratio: request amount to existing limit (if any) | continuous | -1 | -0.02 |
| 14 | RATIO_REQ_TURNOVER | Ratio: request amount to buyer turnover | continuous | 0.08279 | -0.15 |
| 15 | LEGAL_CREATE_AGE | Buyer creation age | continuous | 7437 | -0.04 |
| 16 | FINANCIAL_219000TR | Profit / Loss of the year (+/-) from Balance Sheet | continuous | 0 | 0.04 |
| 17 | FINANCIAL_380000TR | Net profit / loss (+/-) from Income Statement | continuous | 2931990 | 0.27 |
| 18 | THD_SGM_INT | PH segment | continuous | 1 | -0.06 |
| 19 | PROFITABLE | PH profitability flag | continuous | 0 | 0.00 |
| 20 | VAL_GRD_C x TYP_GRD_C | | interaction | - | -0.10 |
| 21 | VAL_GRD_C x LEGAL_MEAN_EMP_Q | | interaction | - | -0.14 |
| 22 | ADT_M_EUR x LEGAL_MEAN_EMP_Q | | interaction | - | -0.08 |
| 23 | RATIO_REQ_ADT x LIM_CREA_AGE | | interaction | - | 0.10 |
| | Baseline score | Baseline value for the score, should all contributions be zero. | constant | | -0.44 |
| | **Model score** | **Model score: sum of baseline + variables scores** | | | **0.75** |

Figure 5: ADT ++ Score Dashboard

# 2 Interpretability, Explainability and Intelligibility

"Interpretable and explainable machine learning techniques emerge from a need to design intelligible machine learning systems, i.e. ones that can be comprehended by a human mind, and to understand and explain predictions" [4]. For AI/ML methods, the terms interpretability and explainability are commonly interchangeable because there is no official agreement within the data science community. While they are very closely related, it is worth unpicking the differences, if only to see how complicated things can get once organizations start digging deeper into machine learning systems. Unfortunately, there is no formal definition for intrepretability and also there is no distinction from the task of interpretation. For example, Doshi-Velez and Kim define interpretability of ML systems as "the ability to explain or to present in understandable terms to a human" [5]. This interpretation lacks of mathematical rigour and it is too trivial. Nevertheless, the notion of interpretability often is determined by the specific field of application and sometimes is named as *intelligibility*. Another prevalent term in the literature is the *explainability* of the models. Again, there is no formal definition for it, but the most followed approach is the one defined in [4][6] where the author draws a clear line between interpretable and explainable ML: interpretable ML focuses on designing models that are inherently interpretable; whereas explainable ML tries to provide post hoc explanations for existing opaque models, i.e. models that are incomprehensible to humans. Since there is no general definition of either interpretability and explainability, the experts came up with some clear desired properties that interpretable ML models must have:

- Trust: Explanation should reflect the reality
- Causality: Model should reflect causality between variables, instead of mere associations.
- Robustness: Machine learning systems should be resistant to noisy inputs and domain shifts
- Fairness: Models should respect real life behaviours, but this is not a straightforward task.
- Privacy: can be of concern in systems relying on sensitive personal data, interpretations and explanations can help to understand if user privacy is preserved.

## 2.1 Glass-Box vs Black-Box

The introduction of these terms also led the experts to differentiate between the types of models used. They are called *Glass-Box* and *Black-Box* and often are cited by the professionals where they have to take decisions between model accuracy and explainability, but also when the complexity of the models should be reduced, since in general, simpler models tend to be more explainable than more complex ones.
*Glass-Box* models are structured for direct interpretability, meaning the explanations that are generated are exact and human interpretable. These models include all the desired properties discussed before and are really simple to understand given them intrinsic interpretability. In a *Glass-Box* model all parameters are known and is know exactly how the model comes to its conclusion, conferring full transparency.

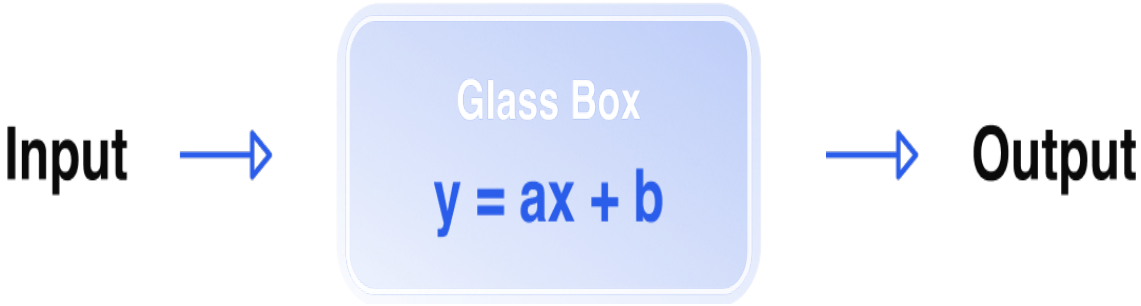


Figure 6: Glass-Box model representation

The advantage of using these models is that they are more practical in the business context. Since companies and organizations can understand how these frameworks come to their prediction it is easier to take actions on them. Businesses can use them to find tangible ways to improve their workflows and know what happened if something goes wrong. On the other hand, the cons of using these models are that often they are too simple and parsimonious to be applied in complex context and they are mostly linear, reason why are at times not used in practice even if they can reach reasonable prediction results. Within *Glass-Box* models family there are : *Logistic Regression* [7], *Linear Regression*, *Generalized Additive Models* [8], etc.
Contrastingly, *Black-Box* models produce useful information without revealing any information about its internal workings. These models are factually not interpretable because internally make complex calculations that are not inherently explainable. For so, these models must need *post-hoc* techniques to be understood and all these methods can only give an approximation of the real prediction given by the model.


Figure 7: Black-Box model representation

In some cases, it is not terribly important for humans to fully know how the model works or how it reaches its decisions. But, ethical AI is becoming more and more discussed since today the human touch is considered less important in decision making contexts. Nowadays, this family of models contain the best machine models on the market, i.e. *Neural Networks* [9], *Extreme Gradient Boosting* [1], reason why are broadly used, but sometimes the results are not entirely understood but are only accepted for their outstanding results in terms of performances. To summarize the concept of trade-off between performance and explainability of the two families of models, a simple illustration is given in Figure 8 where the main models of each family are included.

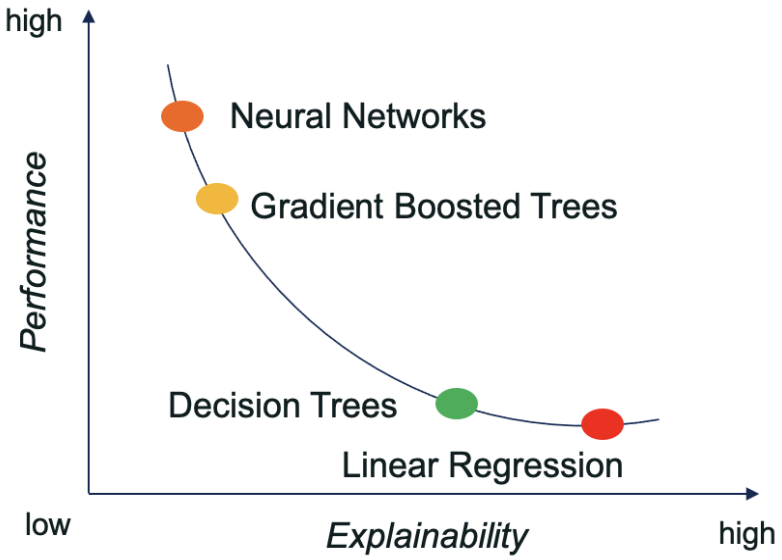


Figure 8: Performance-Explainability Trade-Off

## 2.2 European Regulatory Constraints

Regulations on the use of the described models are leading to a law shift regarding machine learning and artificial intelligence. The AI Act is a proposed European law on artificial intelligence – the first law on AI by a major regulator anywhere. The objectives of the proposed regulatory framework as summarized in [10] are:

- Ensure that AI systems placed on the Union market and used are safe and respect existing law on fundamental rights and union values
- Ensure legal certainty to facilitate investment and innovation in AI
- Enhance governance and effective enforcement of existing law on fundamental rights and safety requirements applicable to AI systems
- Facilitate the development of a single market for lawful, safe and trustworthy AI applications and prevent market fragmentation.

As described in [10] to achieve the outlined goals, the Artificial Intelligence Act draft combines a risk-based approach based on the pyramid of criticality, with a modern, layered enforcement mechanism. This means, among other things, that a lighter legal regime applies to AI applications with a negligible risk, and that applications with an unacceptable risk are banned. Between these extremes of the spectrum, stricter regulations apply as risk increases.

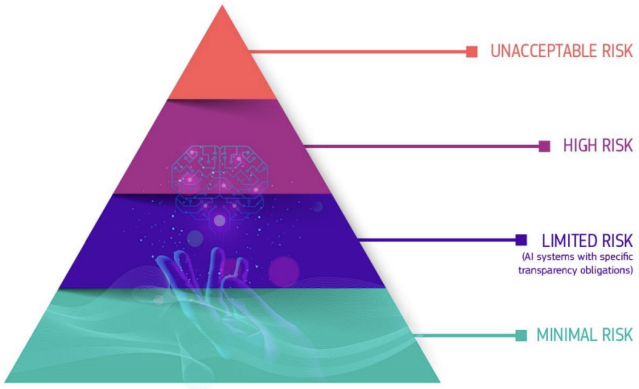


Figure 9: The Pyramid of Criticality for AI Systems

Like the EU's General Data Protection Regulation (GDPR) in 2018, the EU AI Act could become a global standard, determining to what extent AI has a positive rather than negative effect on human beings and businesses. For non-high-risk AI systems, only very limited transparency obligations are imposed, while for the very-high-risks the requirements of high quality data, documentation and traceability, transparency, human oversight, accuracy and robustness, are strictly necessary to mitigate the risks to fundamental rights and safety posed by AI and that are not covered by other existing legal frameworks. In a nutshell, for Allianz Trade and other companies a restriction on the use of the *Black-Box* models will be imposed and this will lead to a critical transformation in the processes of the organizations forcing them to find new approaches.

# 3 Explainable Boosting Machine

Due to these restrictions, academics and researchers crafted an outstanding new model that is considered a game changer within the machine learning community. Explainable Boosting Machine (EBM) is an algorithm used for supervised learning and classification problems. EBM is based on gradient boosting, which is a widely used ensemble machine learning technique. However, unlike traditional gradient boosting algorithms, EBM provides interpretable and transparent results, which makes it easier to understand and explain the model's predictions. EBM works by combining a large number of simple decision trees to form a complex model. The trees are trained sequentially, where each tree tries to correct the errors made by the previous trees. The decision trees used in EBM are shallow, meaning they have only a few splits, which makes it easier to understand the decisions made by the model. EBM builds upon or augments generalized additive models (GAMs) , one of the most used models when interpretability is required.

$$g(E[y]) = \beta_0 + \sum f_i(x_i) \tag{1}$$

where *g* is the link function that adapts the GAM to different settings such as regression or classification. A response variable $\hat{y}$ is predicted by learning an intercept ($\beta_0$) along with functions that describe the relationship between the response and each predictor variable. Essentially, the coefficients ($\beta_i$) in a multiple linear regression model are replaced with learned functions ($f_i$) that are not confined to a linear relationship. The model is additive because separate functions are learned for each predictor variable independently, which allows for an examination of the effect of each predictor variable separately [8]. One shortcoming of GAMs is that they ignore possible interactions between different features. And so, research had them included in what's known as Generalized Additive Models with Pairwise Interactions (GA²Ms) [11]:

$$g(E[y]) = \beta_0 + \sum f_i\left(x_i\right) + \sum f_{i,j}\left(x_i, x_j\right) \tag{2}$$

where $f_{i,j}$ represents the sum of all the learned functions of the pairwise interactions.
EBM expands upon GAMs to maintain interpretability but improve predictive performance. First, learns each feature function $f_i$ using modern machine learning techniques (boosting). The boosting procedure is carefully restricted to train on one feature at a time in round-robin fashion[1] using a very low learning rate. Due to this fact, only small updates to the model are made with the addition of each tree. This requires the model to be built by iterating through the training data over thousands of boosting iterations in which each tree only use one predictor variable. The algorithm developers argue that the low learning rate reduces the influence of the order in which features are used while iteratively cycling through the predictor variables using a round-robin method minimizes the impact of co-linearity to maintain interpretability. Second, EBM can automatically detect and include pairwise interaction terms as described in (2), and this can improve accuracy especially in the regression context. To take into account interactions between predictor variables, two-dimensional functions $f_{i,j}(x_i, x_j)$ can be learned to relate the response variable to pairs of predictor variables. The subset of available interactions included are selected using the FAST method proposed by that ranks all pairs of predictor variables. Explainable Boosting Machine is today the only glassbox model designed to have accuracy comparable to state-of-the-art machine learning methods like Random Forest and Boosted Trees, while being highly intelligibile and explainable. For so, it can be included in Figure 10 as below.

[1] In machine learning, round-robin fashion is a technique used to train models on a large dataset by splitting it into smaller subsets called batches. The model is trained on each batch in a circular order, with each batch getting an equal chance to be trained. This helps to train the model in a memory-efficient way when the dataset is too large to be loaded all at once.

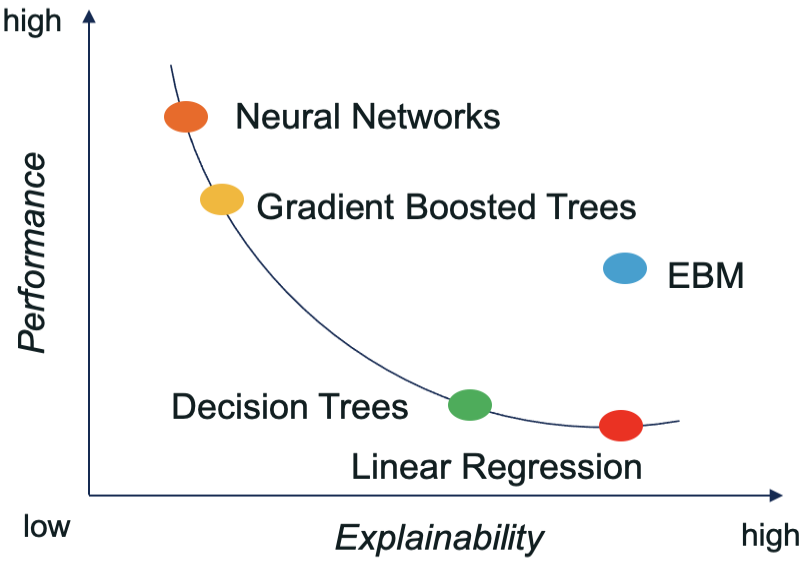


Figure 10: Performance-Explainability Trade-Off with EBM

## 3.1 Global and Local Explainability

EBM is highly intelligible because the contribution of each feature to a final prediction can be visualized and recognized by plotting $f_i$. Because EBM is an additive model, each feature contributes to predictions in a modular way that makes it easy to reason about the contribution of each feature to the prediction. To make individual predictions, each function $f_i$ acts as a lookup table per feature, and returns a term contribution. These scores contributions are simply added up, and passed through the link function $g$ to compute the final prediction. Because of the modularity (additivity), term contributions can be sorted and visualized to show which features had the most impact on any individual prediction.

Explainability of machine learning models is the ability to understand how a model arrived at its predictions. It can be divided into two categories: global and local explainability. Global Explainability refers to the overall understanding of how the model works, including the relationships between input features and the model's predictions. Global explainability can be achieved through techniques such as feature importance analysis, model interpretation, and visualization of model decision boundaries. A general interpretation is given by the Global Overall Importance plot, one of the built-in function of Interpret [2] package. It gives an overall importance about all the features showing the feature absolute value, similar to the SHAP values summary plot.

[2]https://interpret.ml/

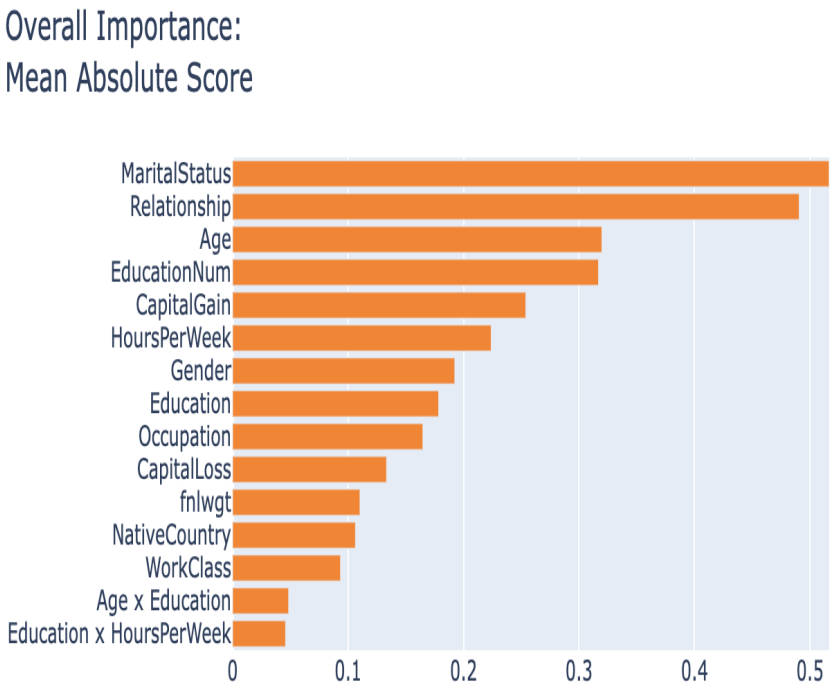


Figure 11: Overall Importance

Also, it is possible to visualize each function independently as function of a score which represents the impact on the target variable. An example is given in Figure 12:

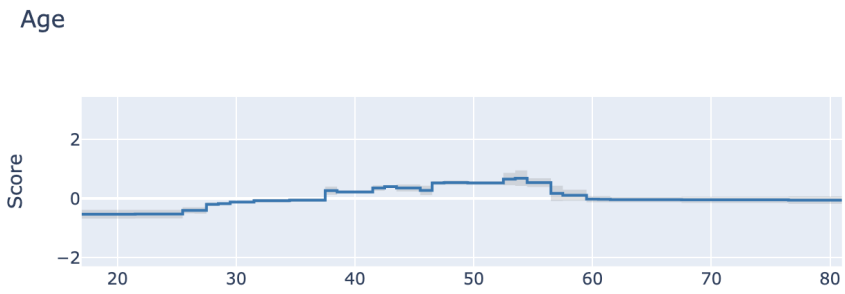


Figure 12: Single feature plot

In Explainable Boosting Machine, the scores for each feature are calculated by aggregating the information from the decision trees in the model. Each decision tree in EBM uses a single feature to split the data into two or more branches. For each tree, the model calculates the score by traversing the tree and summing the contributions of the individual leaves that are reached. The contribution of a leaf is the log odds ratio of the positive class to the negative class. The final score is the sum of the contributions from all the trees in the model used to classify that specific variable. An important differentiation has to be done regarding how the score should be interpreted. In regression, the explanation scores are in the units of the target. For example, in Figure 12 when a new data point come up at $x = 50$ the impact on the final prediction is about $+1$

units of the output. In classification (only binary for the moment), the scores are logits, or log odds. To convert these logits into a probability, they are summed up and passed through a logistic link function. This transformed value will represent the predicted probability to belong to one class for a new observation. This is achievable in Local Explainability which refers to the understanding of why a specific prediction was made by the model. Local explainability can be achieved through techniques such as instance-level interpretation, feature attributions, and saliency maps.

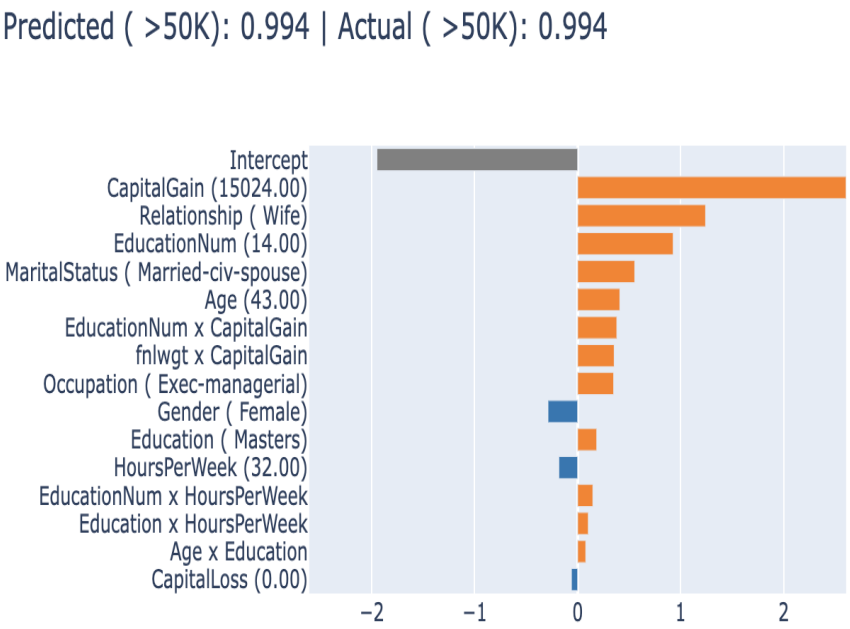


Figure 13: Local Explanation and Prediction

In the last plot a single prediction is shown. For each variable and interaction is possible to get how much it impacts on the final prediction (positively or negatively). In order to get the prediction it is needed to sum up all the values and pass them through a logistic regression to transform the sum from Log-odds to probability.

$$Pr(Y_i = 1|X_i) = \frac{exp(\beta_0 + \beta_1 X_i + \beta_2 X_2 + ... + \beta_n X_n)}{1 + exp(\beta_0 + \beta_1 X_i + \beta_2 X_2 + ... + \beta_n X_n)} \tag{3}$$

Where $Y_i$ is the target variable, $X_i$ the regressor or the interaction if present, and each $\beta_i$ is the estimated coefficient for each single feature or pairwise interaction, instead for $\beta_0$ which represents the intercept.

# 4 Empirical Application

This empirical application is based on the model introduced in 1.4.1, and the specific country analyzed for the MFW was Italy. The main analysis consisted in test different machine learning models and compare them with the performance of Explainable Boosting Machine, but also sec-

ondary analysis have been conducted to enhance and understand the performances of the model based on the given data.

## 4.1 Data Wrangling and Data Pre-Processing

In this chapter, I will explain the Data Wrangling process step by step from data collection to modelling and insights elaboration. Data pre-processing will also be discussed along with details of the data infrastructure and the databases used for the project. But, before going into the details of data wrangling, I would like to introduce the Allianz Trade data infrastructure. Allianz Trade uses Amazon Web Services (AWS) for its cloud provider. All streaming data is stored in the AWS Cloud.

### 4.1.1 Data Infrastructure

Allianz Trade receives 600 000 requests for credit limits each month relating to 30 million companies, 82% of which must be processed in real time. Recently, new platforms and marketplaces have been launched and threaten to disrupt the credit insurance industry. To maintain its position and drive innovation, Allianz Trade switched to Amazon Web Services (AWS) and noticed faster time to market for new services, better cost control and greater ability to integrate with services for new market entrants via APIs. This transformation therefore impacts the entire infrastructure. One important change has been the Infrastructure as Code (IaaC), which allows for faster and easier implementations. Infrastructure as Code also leverages serverless platforms, improving elasticity, resiliency, and security. The benefits of this infrastructure transformation are obvious: stability and performance of applications hosted on the public cloud landing zone.

### 4.1.2 Data Science Server

The server is based on an Amazon Elastic Compute Cloud (Amazon EC2) system. The EC2 instance is based on a Linux system as it can freely communicate with Amazon S3 (Amazon Simple Storage Service), where all snapshots of Allianz Trade live streaming data are stored. Each month, Amazon S3 data is updated with a snapshot of live streaming data. Within the data scientist server, it is possible to access data on a daily basis. For security reasons, data in Amazon S3 cannot be downloaded locally, all calculation must be done in the cloud, reason why all the data are encrypted with private codes. The GDA also have an Amazon Elastic File System (Amazon EFS), which provides a simple, scalable and fully managed elastic Network File System file system that we can save our scripts and notebooks when we shut down the instance. For this project, some of the data needed for training and production are stored in Amazon S3. Other data sources will be presented later. In Figure 14 it is possible to see and understand properly how the data lake is accessible for different entitities.

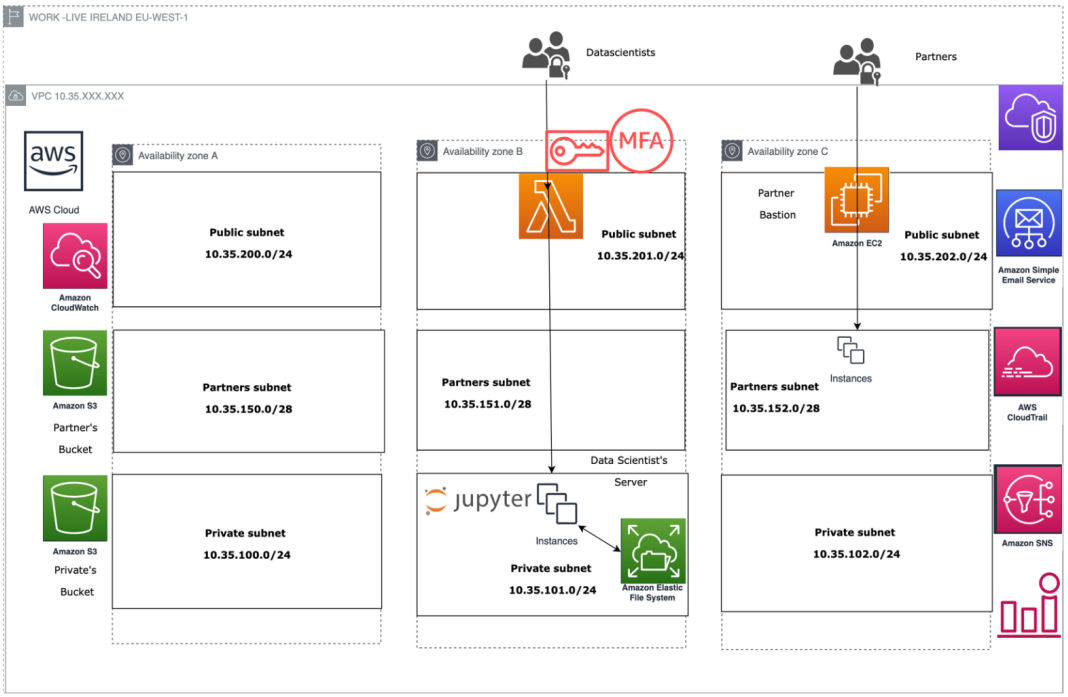


Figure 14: Allianz Trade AWS Infrastructure

## 4.2 Datasets

Allianz Trade has a very large dataset in the cloud with all companies registered. In this section, I will present the data used for the Shamrock project. At Allianz Trade, risk analysis and management are based on an integrated information system, called IRP [12], which will make it possible to manage all of the Group's commitments from a single program. This database is the key to all of Allianz Trade's activities. Every data analyst/scientist needs IRP information to examine the health of a business. IRP database contains a wide variety of data about a business, such as administrative information, financial information, grading information, etc. Allianz Trade has a rich collection of data available, coming from their data providers. The information is stored in the application system IRP, which is continuously fed with data updates for all the buyers. A data feed has been setup by IRP IT team, to download data in the Data Lake, since the data are updated once a month approximately and they come from four primary data providers.

### 4.2.1 Data Collection

Data has been collected through different channels. From IRP, 20 tables were extracted and each of them contains specific information related to claims, policy holders and debtors such as:

- Administrative data with every detail about the legal form of a company, the number of employees, the activity sectors, addresses, segment codes. These variables are codified as CODIFIED_VAL_******.

- Financial data with all the details regarding the company performances and the financial information contained in the companies yearly accounts. These variables are codified as FINANCIAL_****** or FINANCIAL_EVOL_******.
- Grading data to reflect the company's health situation and any fluctuation in grades, represented as GRD_VAL_*.
- Other data containing data about payment incidents, official publications and risk assessment, labelled as RISK_XX_****** or PAYMENT_INC_******.

### 4.2.2 Data Technical Pre-Processing

This process consist in extracting data from the previous table. The extraction is denoted as "technical pre-processing", as it is not driven by business constraints. The subsequent task denoted "functional pre-processing" is the result of the initial data analysis run by the data scientists, with the objective to prepare the data in the best possible way for their further injection in the machine learning grading engine. This first step consists in selecting the appropriate set of buyers on which the model will be trained. To do so, AZT filtered the input tables containing the buyers with exposures or requests, month by month which provides the aggregated limit requested by all policyholders on a buyer, per unique buyer identification number. Allianz Trade then filtered out buyers with no exposure nor grading request. The main reason for this is to be found in the availability of input data: as AZT purchase information on buyers with exposure or request, they do not systematically follow the remaining ones that are have been historically recorded in IRP. As Allianz Trade do not follow them, they have no information on their health/default status, which would introduce a bias in the sample. For this reason, they are excluded from the datasets. Once a list of buyers is obtained, the technical pre-processing pipeline will find, select and extract corresponding buyers' profile in the data lake (copy of IRP database), and filter out unwanted features. The data extraction could rely either on common pre-processing program (e.g. for TFBS items), or on local pre-processing program (e.g. for French official publications data). As a technical stage, this step is directly processed through a Python pipeline (i.e. an automated sequence of tasks), with the objective to select from the existing IRP data tables desired variables for a list of buyers' profiles during a specific moment (point-in-time data) [13].

### 4.2.3 Data Functional Pre-Processing

Once the sample of final buyers is obtained, data need to aggregate and transform (if necessary) the variables to obtain a workable dataset (usually a Pandas Dataframe). At the end of the technical preprocessing, the features are available in their raw forms. While some of them can be directly used without any new transformation, others need to be transformed so that the algorithm can use them [13]. Due to the merge of the previous tables, many duplicates have been created. In the first place, any duplicated column along with useless columns should be dropped. Next is dtype conversion, to each variable will be assigned its convenient type (*dates, floats, integers, etc*).

Other manipulations were done such as replacing erroneous values with *NaN* to make them easily detectable. Moreover, renaming columns following a standard notation was necessary to simplify manipulation in the coming steps.

### 4.2.4 Training and Test Data

The datasets obtained at the end of the data collection & preparation phase consists in two subsets: the train set on which the model will learn to discriminate good and bad buyers, and the test set on which we objectify how the model can generalize. As known, in machine learning fashion, it is important to have an independent measure through either an out-of-sample test set or an out-of-time test set. For so, the three dataframes used for the analysis have been the following:

- Train Datasets: the observation period is based on three years of defaults (2015, 2016 and 2017) for buyers with and without exposure. The dataset was composed of 315394 rows and 352 columns.
- Out-of-sample test set (*OOS*): a hold-out (sub)sample is left apart, not incorporate in the training set, so that the model is asked to predict a failure probability on a set of unknown buyers. The dataset was composed of 105678 rows and 352 columns.
- Out-of-time test set (*OOT*): a hold-out sample of buyers observed during a period on which the model has not been trained, 2018 in this case. Because this method is more independent from the train set, the risk of overfitting is limited (overfitting reduces generalization as the model is trained to predict the "white noise" in the dataset, which is not deemed to reproduce in the near future). Despite being more demanding (full set of observation, limiting the length of the train set observation frame), this method has been standardized for all implementations since 2019. The dataset was composed of 366642 rows and 352 columns.

All the three datasets are composed of only continuous variables and all of them were related to the buyers since those are the possible defaulters and because trivially reflect the objective of the model. Moreover, at the end of the pre-processing, we obtain the pre-processed data target, and the categorical dictionary to encode the categorical features in the test set (i.e. to have the same numerical values assigned to the categorical ones). The indicator used for such target was a binary variable [0,1] representing 1 as defaulted buyer and 0 otherwise.

## 4.3 EDA and Data Visualization

The first step to have a general idea of the situation includes the exploratory analysis of some variables. It helps to have a global idea about data distribution and major characteristics of potentially important variables. The analysis was split into two parts, one from claims perspective and the other one from features perspective. The last updated files containing claims included data from 2019 to 2021, the 2022 data were not yet available. The claims analysis for Italy have been done recently and the results have been compared with the previous dataset containing previous

claims including data up to 2020. There were some differences in the number of claims in the two datasets for the years 2019 and 2020 since the Italian law considers a company as a defaulter in a different way compared with other European countries. This because for the Italian legislation there is a way to pay the debt back and to save the company from failure, finding a deal with its creditors [3]. The business requires also that the claims for Italy must be only of two specific types, called Fido 1 and Fido 5. The *Fido Commerciale* is the value of the goods or services sold by the supplier and they will be payed with deferred payments by the buyer. It is based on the reliability and solvability of the buyer and depends on the contract stipulated with the supplier. After this preliminary analysis, the claims have been then stored and they have been used for backtesting purposes.

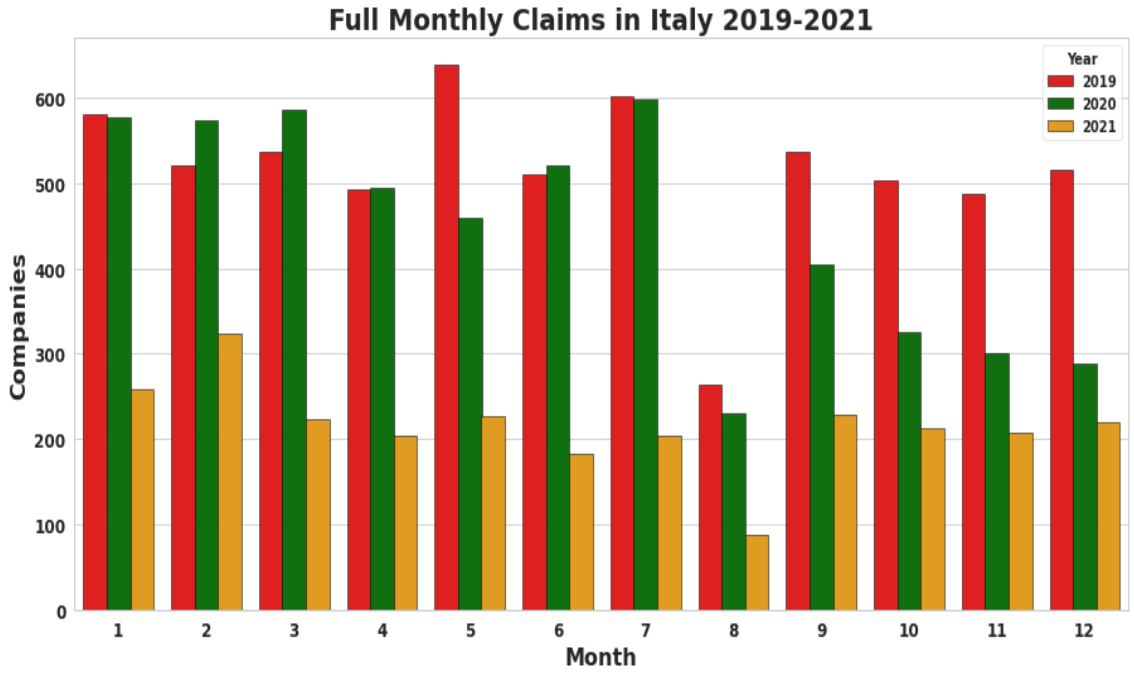


Figure 15: Monthly Claims Italy 2019-2022

From Figure 15 is possible to make different assumptions which could be related with future results of the models.

- 2019 and 2020 have been the years with more claims for all the months, probably due to COVID 19 pandemics
- August could be considered as a "calm" month in terms of claims, but this is probably due because in Italy the month of august is the closing period of almost all the companies
- Big drop in the number of claims probably caused by the financial aids to support the businesses after the crisis

[3]This law is regulated by Regio decreto del 16/03/1942 n. 267

Regarding the variables some of the features have been analyzed, the plots were mainly about the distribution of amounts, financial values, and industry codes. An example is shown in Figure 16 .

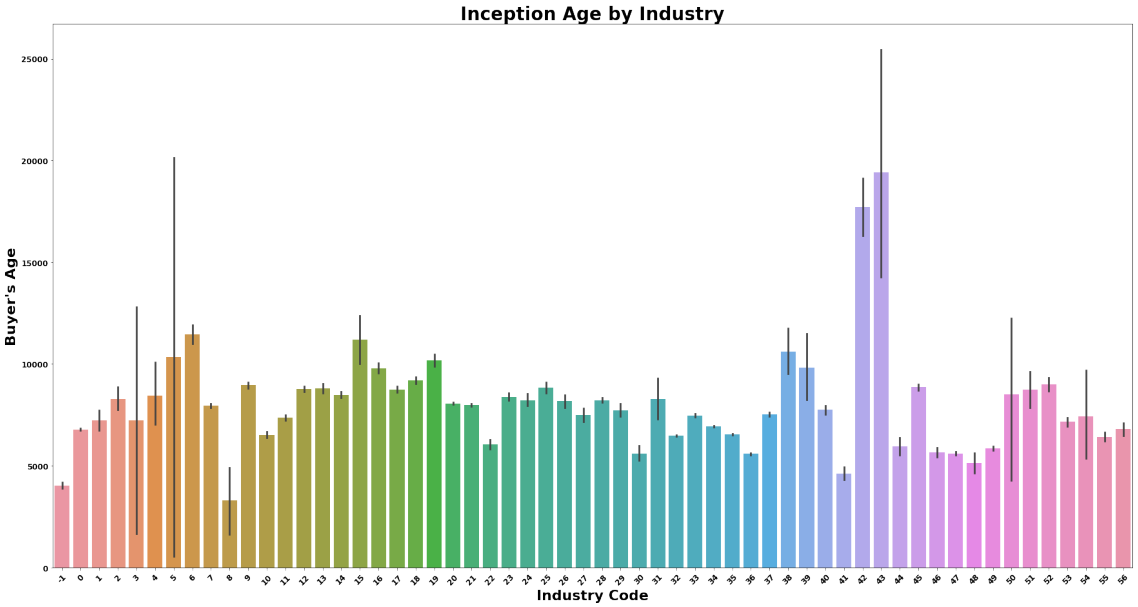


Figure 16: Buyer's age (in days) by industry code

The figure above clearly shows that industry code 42 (Construction) and 43 (Metallurgic) are the field with the higher buyer's age. In fact these two field can contain really old enterprises that have been stabilized during the years in the respective markets.

### 4.3.1 Correlation and Multicollinearity

In classification models, it is important to check for correlations between the features as well as multicollinearity, which can cause instability and inconsistency in the model's performance. Correlation refers to the linear relationship between two variables. In the context of classification, it is important to check for correlation between the input features and the target variable. This can help identify potential confounding variables that may affect the model's performance. Normally, for every 2 variables that are highly correlated, one should be dropped to avoid multicollinearity but this depends on the level of the correlation and to each specific problem because sometimes some correlation is needed. Figure 17 shows a heatmap of correlation between a subset of variables, where the degree of the correlation is between -1 (High negative correlation) and 1 (High positive correlation). Due to the high number of variables, a heatmap grouping all the dataset could not be plotted.

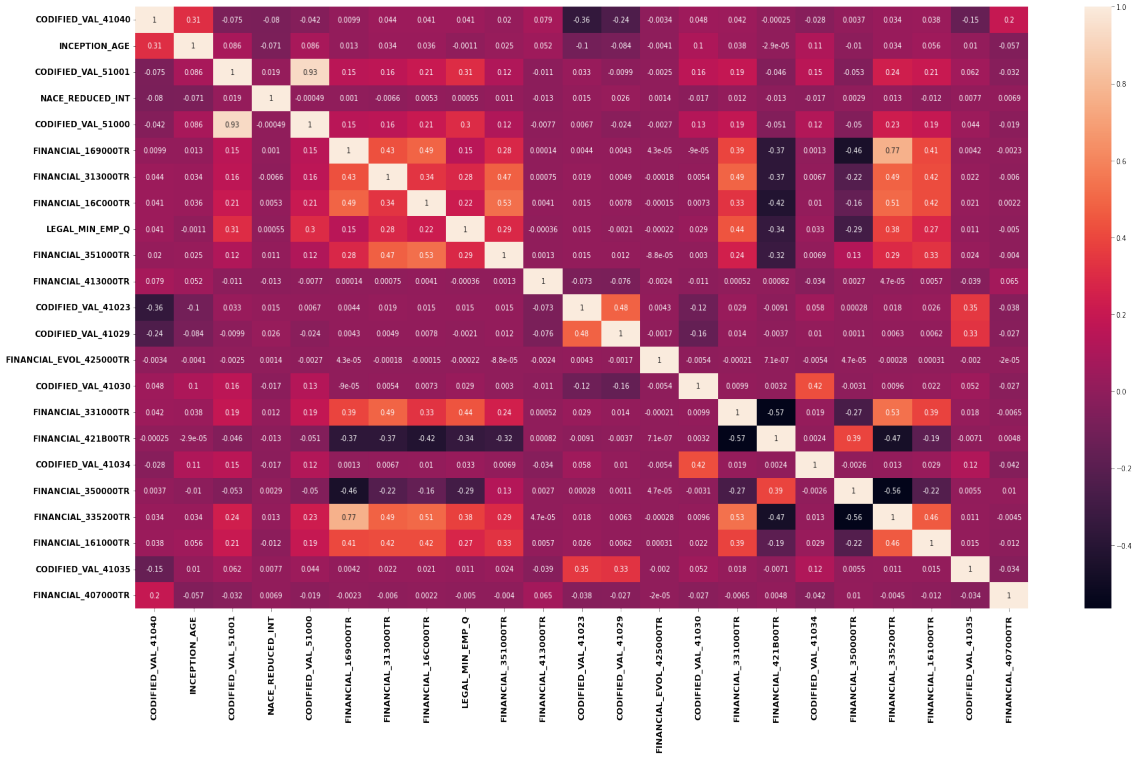


Figure 17: Heatmap plotting some variables used for the analysis

## 4.4 Hyperparameters Tuning

Hyperparameter tuning is an essential step in developing a successful machine learning model. In this project, we focused on tuning the hyperparameters of our Explainable Boosting Machine (EBM) classification model. While classic hyperparameter tuning methods, such as random search, have proven to be effective, they can be computationally expensive, especially for complex models like EBM. Therefore, we chose to use a more efficient approach that allowed us to tune the hyperparameters in a reasonable amount of time. Besides the values combination, a metric to measure performance should be chosen. For Shamrock the *AUC Score* has been chosen and the choice of these metric is justified by the project context. Actually, EBM should perform reasonably well out of the box with the default hyperparameters without any parameter tuning, as confirmed by the developers [4], but for a more detailed analysis, the hyperparameters [14] have been manually tuned. From previous analysis the parameters that affected the most the model are the following:

- *outer bags*: number of outerbags, is like wrapping a bagging process around the core algorithm, where individual EBMs are fit on different subsamples of the dataset. The final shape functions produced are an average of the shape functions learned in each outer bag.
- *inner bags*: number of subsamples drawn with replacement at the time of growing each individual tree in the boosting process. When the algorithm visits a single feature, it creates

[4] https://github.com/interpretml/interpret/issues/162

$n$ number of samples of the data, grows trees on each sample, and averages them together before creating the final update used in the gradient boosting process.

- *interactions*: Number of pairwise interactions between the variables.
- *max leaves*: Maximum leaf nodes used in boosted trees.
- *max bins*: Max number of bins per feature for pre-processing stage.
- *binning*: Method to bin values for pre-processing, the possible methods are: "uniform", "quantile" or "quantile humanized".
- *validation size*: Validation set size for boosting, represents the % of data used for train and test.

Due to the slow training time of the model, a method to run multiple EBM in parallel has been used. This technique created by the AZT Machine Learning Engineer team is based on batch computation. The great ability of this tool is the capability to work outside the internal server, leaving it free for other analysis. The characteristic of this method is that does not need scheduling or DAG (Directed acyclic graph) logic and can use customized infrastructure, manually selecting the number of vCPUs and memory wanted for the analysis. It has been a time saving instrument since it allowed to train multiple models with different values for the hyperparameters and then get a solution at the same time, without waiting each trained model to be ready for the comparison. Table 1 shows the hyperparameters that contributed to the creation of the "best" model according to the manual optimization, the remaining have been left as the default ones. The performance gained in using these parameters is noteworthy, with a *ROC-AUC Score* $5\%$ higher than the worst parameters set.

| Hyperparameter | Value |
|---|---|
| *outer bags* | 32 |
| *inner bags* | 16 |
| *interactions* | 0 |
| *max leaves* | 2 |
| *max bins* | 64 |
| *binning* | quantile humanized |
| *validation size* | 0.15 |

Table 1: Model's Hyperparameters

This was the hyperparameters set-up for the model without interactions, another model has been created using 10 interactions but since the difference in accuracy was not that diverse I preferred to loose in accuracy but gain in interpretability. Normally this type of decision are at the discretion of the data scientist and the business. By default the *interactions* value is set to zero because it increases the training time considerably, but at the cost of increasing training time, setting the interactions parameter to a non-zero number, which will automatically detect and introduce several pairwise interaction terms in the model. This can significantly improve performance on some datasets, especially in the regression setting. Unfortunately did not happen on the dataset used for this analysis. Again, increasing the number of *outer bags* and *inner bags* increase the training time but helps notably in terms of accuracy and smoothness of the plots of each function $f(x_i)$. Changing *max bins* can also help regularize the model if some of the observed graphs seems to be too jumpy on sparse regions of the data. The main idea of tuning the number of *max leaves* is connected to the fact that the boosting part should be executed on shallow trees for several boosting rounds, usually around $5000 - 10000$. Overall, is recommended to consider the learned graphs from the *global explanations* and letting that guide the manual parameters tuning.

## 4.5 Features Selection

The features selection is a crucial part in most of the data science project since the business part always requests models with a limited number of variables. For this specific project the request was to retain between 20 and 50 features, but often depends on each specific project since the accuracy of the model is highly affected by this factor. Furthermore, the business always wants features easy to understand and useful for the business purposes of the projects. Thanks to two different techniques based on machine learning is possible to obtain the "best" variables to include in the model and most of the time reflect the requests of the business, reason why they relies heavily on these methods.

### 4.5.1 EBM Overall Score

One of the main characteristics of EBM is the full explainability given by the overall importance plot. This graph represents all the variables that affects the most the score of the model in absolute terms. It simply plots the sum of each additive terms of the model for each feature giving an overall score of the model.

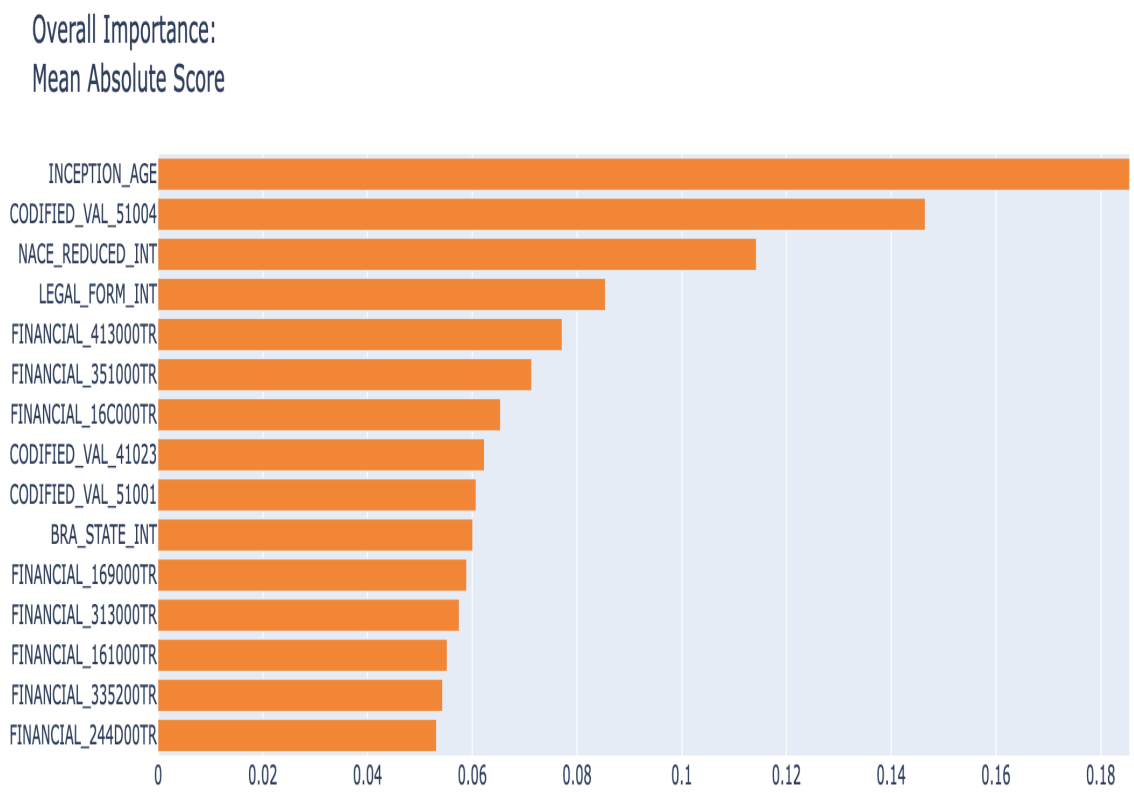


Figure 18: Top 15 variables from EBM overall importance (Italy model)

Looking at Figure 18 it is easy to understand that financial variables are the most effective in the model and that highly reflect the expectations of the business which are mainly interested in the financial wellness aspects of the buyers. The selected variables were 50 but it could not possible to give an explanation for all. The financial variables are mainly related to debt and payments like Debt-to-Equity Ratio (FINANCIAL_413000_TR), Interest Coverage Ratio (FINANCIAL_351000_TR), Debt-to-Asset Ratio (FINANCIAL_16C000_TR), Current Ratio (FINANCIAL_169000_TR), Accounts Payable Turnover (FINANCIAL_313000_TR) Ratio, Cash Conversion Cycle (FINANCIAL_313000_TR), Days Payable Outstanding (FINANCIAL_161000_TR), Return on Investment (ROI) (FINANCIAL_335200_TR) and Return on Asset (ROA) (FINANCIAL_244D000_TR). The Codified variables are non financial variables like trends or scores as it possible to see in this rank for *CODIFIED_VAL_41023* (payments trend) and *CODIFIED_VAL_51001* (score given by banks).

Narrowing down to the top 4 variables: *INCEPTION_AGE* which represents the legal age of the buyer has the highest significant impact on the default risk of the buyer with a value higher than 20%. The variable with the second most effective impact is *CODIFIED_VAL_51004* which is a codified act, representing a specific score called CGSX given by external company, with a total impact of almost 15% on the total . Concerning the *NACE_REDUCED_INT*, representing the trade sector ID with two digits, an impact of 13% is observable, meaning that the risk is dependent from sector to sector. The fourth variable *LEGAL_FORM_INT* represents the legal shape of the

company at its creation with an impact of 9%.

This is one of the two methods used for the features selection and it is possible to obtain a list of variables simply filtering by the value contribution of each variable sorted by importance. That will represent the $n$ variables that should be trained in the model.

### 4.5.2 SHAP Values

The second method used to select the most suitable features for the model has been the SHAP (SHapley Additive exPlanation) value. As described in [15] Shapley values are based on cooperative game theory, and provide one possible answer to the following problem: a coalition of players cooperates and obtains a certain payout from the cooperation; however, some players may contribute more to the total payout than others; how to fairly distribute the payout among the players in any particular game?

Suppose we have a cooperative game where a set of players each collaborate to create some value. If we can measure the total payoff of the game, Shapley values capture the marginal contribution of each player to the end result, after having tried all the possible combinations of features.

The problem of fairly finding features importance in the prediction of a machine learning model can be addressed from this perspective. We consider a $V$-players game where each feature $j \in 1, ..., V$ is a player and we want to value their contribution. There are $2^V$ possible coalitions and each coalition S is associated with a characteristic function:

$$v : 2^V \longrightarrow S$$

The Shapley value of each player $j$ is:

$$\phi_j(v) = \sum_{S \subseteq \{1,...,d\} \setminus \{j\}} \frac{|S|!(V - |S| - 1)!}{V!} [v(S \cup j) - v(S)] \tag{4}$$

The idea is that if player $f$ plays much better than the other, then $v(S \cup j)$is consistently higher than $v(S)$ and therefore $\phi_j(v) >> 0$.

The general idea shared by several explanation methods is to locally approximate the original model with an "explanation model," which is simpler to interpret. SHAP is a framework unifying some of these interpretability methods in the class of "additive feature attribution methods" and providing feature importance measures, based on a solid theory. The global importance of variable $v$ is given by the sum of the absolute Shapley values for all observations in the data:

$$I_j = \sum_{j=1}^{n} |\phi_j(v)| \tag{5}$$

The result of this framework applied on the Italian dataset have reflected the expected results of EBM overall importance showing a strong consensus on which determinants are the most important, as shown in Figure 19 .

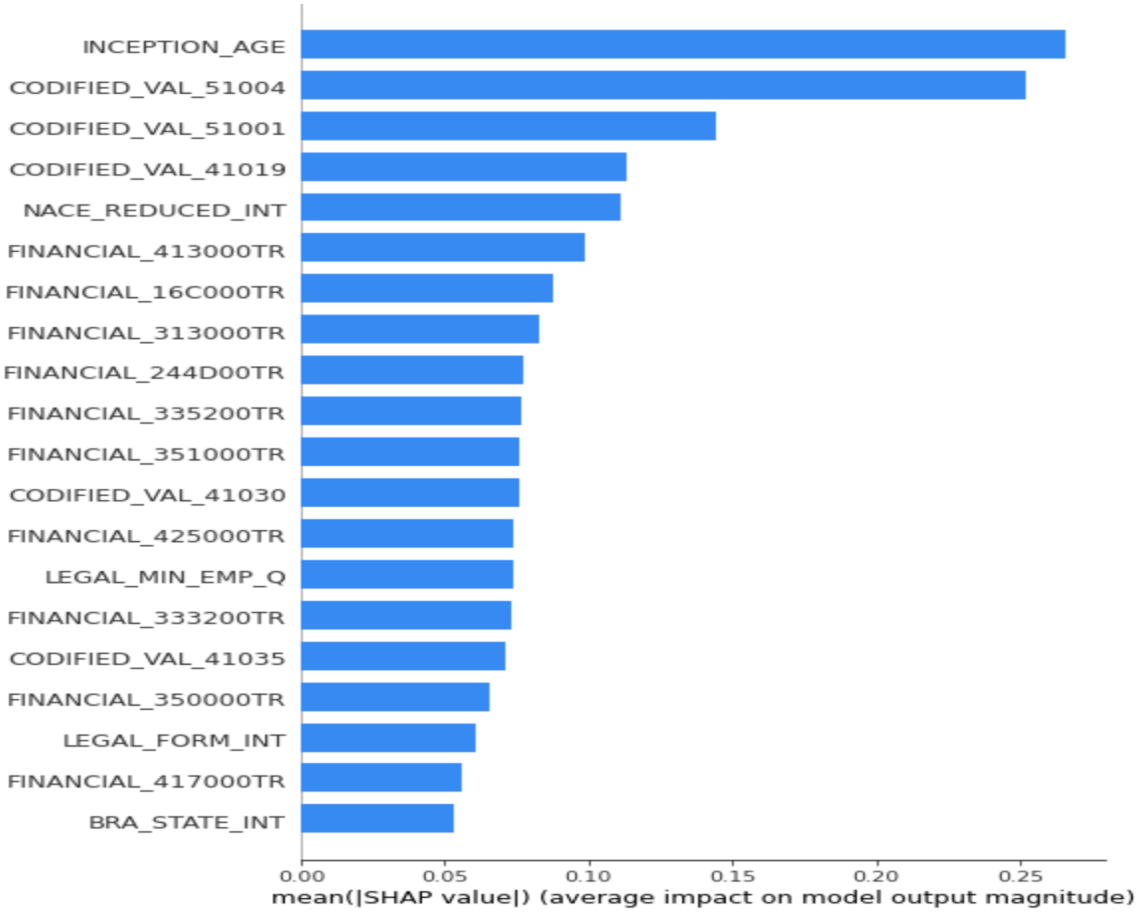


Figure 19: Top 15 variables from SHAP Values Summary (Italy model)

## 4.6 Model Selection Criteria

The Explainable Boosting Machine Classifier has been created with the variables provided by the two frameworks in 4.5. Different models have been trained with different number of features starting from 60 and reducing them until 20. Also, the model has been instructed with and without interactions to understand effectively if those could create a better results in terms of accuracy. After many tests, the most suitable model has resulted the one with 40 variables and zero interactions, maintaining a good balance between accuracy and interpretability. More over, the model reflected the pre-imposed business rules for the model production. The evaluation of these models has been done using the *ROC-AUC Score*. The area under a receiver operating characteristic *(ROC)* curve, abbreviated as *AUC*, is a single scalar value that measures the overall performance of a binary classifier (Hanley and McNeil 1984). The AUC value is within the range [0.5–1.0], where the minimum value represents the performance of a random classifier and the maximum value would correspond to a perfect classifier (e.g., with a classification error rate equivalent to zero). The AUC is a robust overall measure to evaluate the performance of score classifiers because its calculation relies on the complete ROC curve and thus involves all possible classification thresholds. The ROC curve is based on two indices which represent the true positive rate and the

true negative rate, both of them can be easily calculated from a confusion matrix . The true positive rate (TPR) also known as sensitivity or recall is a measure of how well the machine learning model can detects positive instances and can be calculated as follow:

$$TPR = \frac{TP}{TP + FN} \tag{6}$$

where TP is the number of true positive cases and FN the number of false negatives cases. The true negative rate (TNR) also known as specificity is a measure of how well the model can classifies negative instances, but the *ROC-AUC* considers the false positive rate (FPR) given by (1-specificity) and that is the probability that a true negative will test positive:

$$FPR = \frac{FP}{TN + FP} \tag{7}$$

Where FP is the number of false positive cases while TN represents the number of true negative cases. These two values determine the different thresholds that create the *ROC* Curve and where it is possible to calculate the area underneath, in order to show which model has the higher separability power.

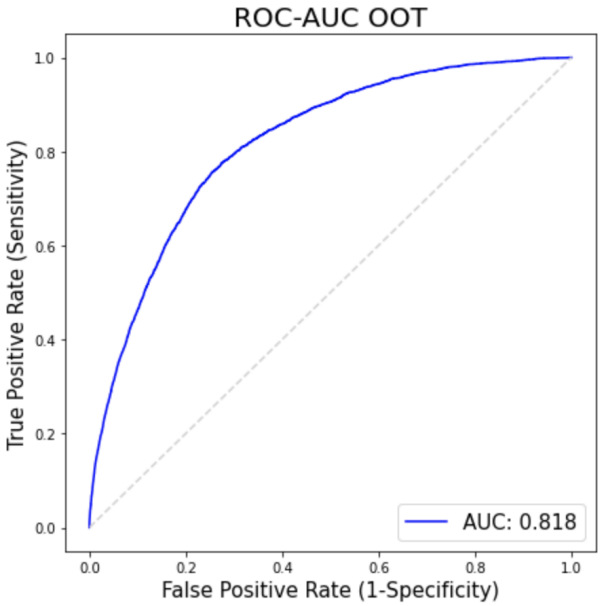


Figure 20: Example of ROC-AUC Curve

In the figure above is possible to observe the curve of the model chosen for the comparison. In insurance terms the number of true positive rate on the y-axis represents the number of Defaulted buyers while on the x-axis the false positive rate is the number of Non-Defaulted buyers. The results in Table 2 show the out-of-time *ROC-AUC Score* of the top 5 models tested for this empirical application.

| Model | ROC-AUC Score |
|---|---|
| EBM (40 var- 0 int) | 0.818 |
| EBM (40 var- 10 int) | 0.819 |
| EBM (20 var- 0 int) | 0.808 |
| EBM (35 var- 20 int) | 0.821 |
| EBM (50 var- 0 int) | 0.817 |

Table 2: Top 5 EBM models ROC-AUC Scores

After some evaluation with the managers, the candidate model used for the comparison with the other frameworks was the one with 40 variables and 0 interactions. It is worth noting that the EBM model selected for this project was not the one with the highest accuracy, but rather the one that balances accuracy with interpretability. The selected model achieved a very similar *ROC-AUC Score* to the best-performing model, but it is much more interpretable. This is because no interactions were added to the model, meaning that the relationships between the features and the target variable are modeled as simple, additive functions. This makes the model easier to understand and explain, which is crucial for the credit insurance field where interpretability is a top priority. Nowadays, Shamrock model is in production and it is based on *XGBoost*, considered the most powerful ML model on the market. But the aim of the analysis is to compare Explainable Boosting Machine performances with other benchmark models such as: *Logistic Regression (LR)*, *Random Forest (RF)*, *Light Gradient Boosting Machine (LGBM)* and *Extreme Gradient Boosting (XGBoost)*. Here again, the selection criteria was the *ROC-AUC Score*. In Table 3 is possible to observe the Out-of-Time *ROC-AUC Score* of the benchmark models compared for the analysis.

| Model | ROC-AUC Score |
|---|---|
| EBM | 0.818 |
| XGB | 0.822 |
| LGBM | 0.821 |
| RF | 0.807 |
| LR | 0.645 |

Table 3: Benchmark Models ROC-AUC Scores

The reason beyond the use of only one type of evaluation metric is due to the fact that the regulators use the *ROC-AUC Score* as the default metric for the back testing. For so I was not able to show other metrics that could be helpful for the purposes of the company.

## 4.7 Isotonic Regression

The machine learning models presented before can give outstanding predictions but it is crucial to understand how they work especially in high-stakes settings (e.g. Medicine, Insurance, Banking and Criminal Justice). It is always recommended to compare the model with the given knowledge because if the model uses expected patterns to make predictions, business and people will feel more confident to deploy it to solve real problems. Sometimes, ML interpretability can uncover hidden relationships in the data helping people gain insights about the problems they want to tackle. Within the GDA team a function to edit Generalized Additive Models including EBM has been built. The function is inspired to GAM Changer [5] which is a model editor with interactive visualizations. The function used is able to change and adapt the shape of the function produced by the model making it monotone (increasing or decreasing). To do so the function apply an isotonic regression or monotonic regression and that is the technique of fitting a free-form line to a sequence of observations such that the fitted line is non-decreasing (or non-increasing) everywhere, and lies as close to the observations as possible. It solves the following problem:

$$\text{minimize} \sum_i w_i \left(y_i - \hat{y}_i\right)^2 \tag{8}$$

Subject to $\hat{y}_i \leq \hat{y}_j$, whenever $X_i \leq X_j$. Where $\hat{y}_i$ is the fitted value and $y_i$ the true observation. Where the weights $w_i$ are strictly positive, and both X and y are arbitrary real quantities. The increasing parameter changes the constraint to $\hat{y}_i \geq \hat{y}_j$ whenever $X_i \geq X_j$ . Setting it to automatic will automatically choose the constraint based on Spearman's rank correlation coefficient. The great feature of this function is that is applicable only where the plot does not makes sense in reality or overfitting is detected. In order to do so it is needed to set the places in where the function should apply and the starting and ending index are based on the additive terms position of the bins that are kept in consideration. Figure 21 shows an example to understand what the function does and the visualization output.

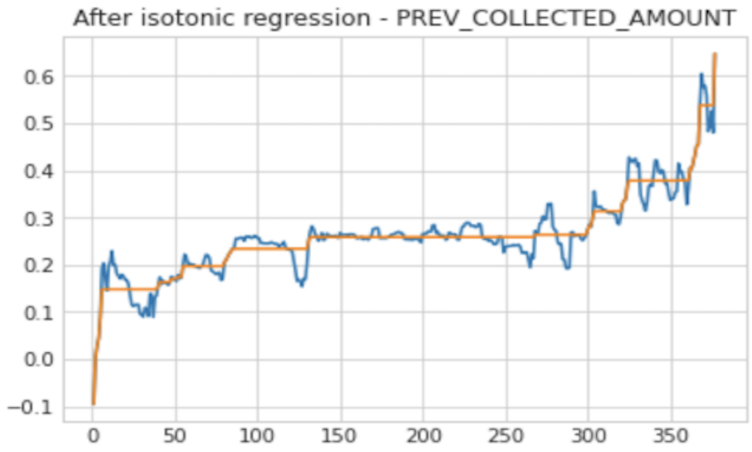


Figure 21: Isotonic Regression Example

[5] https://github.com/interpretml/gam-changer

In the previous figure it is possible to observe the original shape of the function in blue and in orange the function after the application of the isotonic regression. As shown, the function collapse right after zero and then increases tremendously. Where the function is more jumpy is representing an abnormal behaviour and it is probably due to overfitting or some errors in the data, but this is quite normal in large dataset. For this reason the function has been forced to be monotonically increasing after the first highest peak where the score is close to 0.2. This technique leads to a more stable model reducing overfitting and unexpected function behaviours, but on the other hand it reduces the accuracy of the model since the function is smoothed. For so, the manual check of each feature has been a crucial step in the analysis and represented the hardest part of the internship, because included both technical evaluation and business acumen. In Table 4 is shown the comparison of the previous model with the "smoothed" EBM.

| Model | ROC-AUC Score |
|---|---|
| Smoothed EBM | 0.812 |
| XGB | 0.822 |
| LGBM | 0.821 |
| RF | 0.807 |
| LR | 0.645 |

Table 4: Benchmark Models ROC-AUC Scores with modified EBM

## 4.8 Robustness Test

To ensure that the assumptions previously declared in 2 is appropriate to check the robustness of the model after its modification. I created a simple function to compare the robustness of the models before and after transforming them. A model is considered robust when react well after a value perturbation and continues to generalize well without loosing too much accuracy power. This test is central in the credit insurance field since the variables of the buyers can change quite often depending on what happen in the trade market. This has represented the unofficial pre-testing part of the official stress test used for Shamrock model. The function created, simply adds noise to the tested dataset with a value extracted from a normal distribution where the parameter $\mu$ can be modified manually and they will represent a range of the amount of jitters added. Then the function calculate the *ROC-AUC Score* for each value of the range and plot it in a graph. An example is shown in Figure 22.

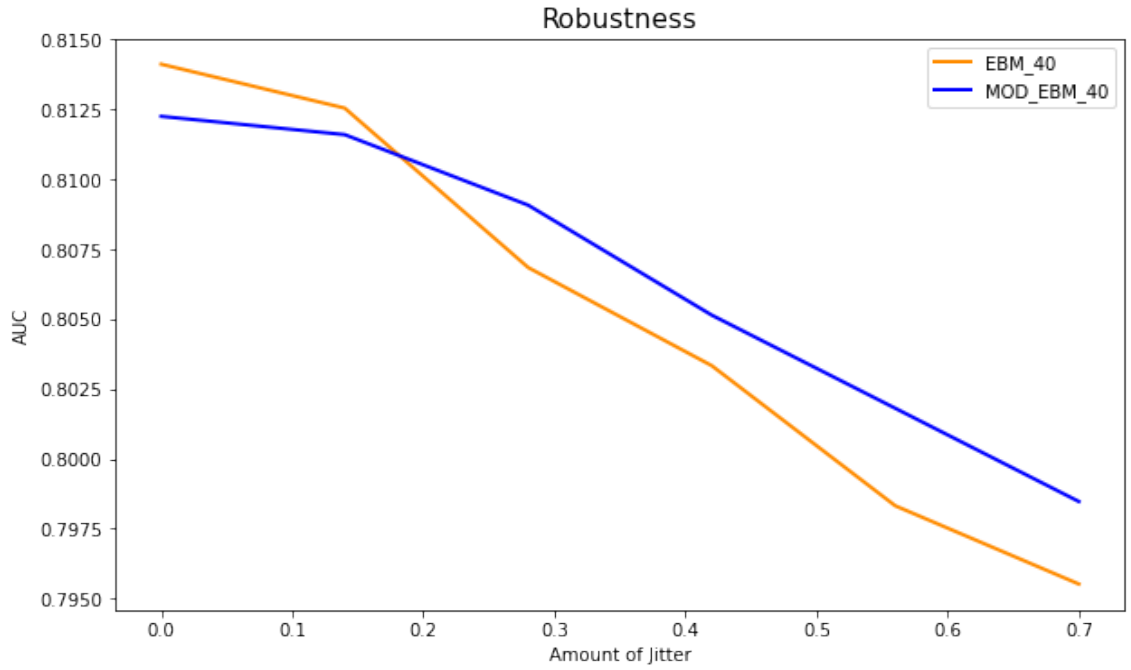


Figure 22: Robustness test for EBM

In the figure above it is observable how the smoothed EBM perform better than the standard EBM model. In fact the reduction of overfitting after the isotonic regression allowed the smoothed model to generalize better on noisy data and gain in performances when the amount of jitter increased making it much more robust.

# 5 Conclusion

In conclusion, the analysis performed in this study has led to promising results in terms of both predictive performance and interpretability. Concerning the results obtained, has been documented that EBM can offer performance comparable with XGBoost and LightGBM and stronger results than simpler methods, like Logistic Regression. These results are very important since strict regulations are approaching the credit insurance industry in the midterm. For so, building up an approach that would outperform competitors while satisfying regulatory and business constraints is a must. Regarding the interpretability, with EBM we reached an outstanding result in terms of *ROC AUC Score* maintaining at the same time a high level of intelligibility, while avoiding the mistake of taking a simplistic approach that would negate all of the research and development investments that the company has made in data science. Nowadays, ADT++ model is the only one built with Explainable Boosting Machine and is live in four countries, but since the business is very confident in EBM, a migration will be conducted progressively on the other models from 2023. So far, the main candidates are Shamrock in Italy and United Kingdom given them superlative results. For so, this approach will not be considered anymore a plan B but a concrete solution for the core business.

# References


[1] Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining; 2016. p. 785-94.

[2] A Trade. Poster shamrock - business part: Unpublished internal company document; 2020.

[3] A Trade. Prism: Unpublished internal company document; 2022.

[4] Marcinkevičs R, Vogt J. Interpretability and explainability: A machine learning zoo mini-tour. arXiv preprint arXiv:201201805. 2020.

[5] Doshi-Velez F, Kim B. Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:170208608. 2017.

[6] Rudin C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence. 2019;1(5):206-15.

[7] Wright RE. Logistic regression. 1995.

[8] Hastie T, Tibshirani R. Generalized additive models: some applications. Journal of the American Statistical Association. 1987;82(398):371-86.

[9] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. nature. 1986;323(6088):533-6.

[10] Kop M. EU Artificial Intelligence Act: The European Approach to AI; 2021. .

[11] Lou Y, Caruana R, G J, Hooker G. Accurate intelligible models with pairwise interactions. In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining; 2013. p. 623-31.

[12] A Trade. IRP : Unpublished internal company document; 2018.

[13] A Trade. Shamrock Implementation: Unpublished internal company document; 2020.

[14] Developers I. Explainable Boosting Machine; 2021. `https://interpret.ml/docs/ebm.html`.

[15] Bastos JA, Matos SM. Explainable models of credit losses. European Journal of Operational Research. 2022;301(1):386-94.

[16] McNeil BJ, Hanley JA. Statistical approaches to the analysis of receiver operating characteristic (ROC) curves. Medical decision making. 1984;4(2):137-50.

# Appendices

## Appendix A

| Classification Performances (ROC-AUC) | | | | |
|---|---|---|---|---|
| Model | Shamrock US (grading) (160746,45) | Shamrock IT (grading) (315394,49) | Shamrock PL (grading) (61339,48) | Shamrock IE (grading) (130562,36) |
| EBM | 0.823 | 0.815 | 0.785 | 0.809 |
| XGBoost | **0.829** | 0.820 | **0.792** | **0.813** |
| LightGBM | 0.827 | **0.821** | 0.789 | 0.810 |
| Random Forest | 0.806 | 0.807 | 0.750 | 0.782 |
| Logistic Regression | 0.635 | 0.643 | 0.707 | 0.620 |

Figure 23: Shamrock Tests

| Classification Performances (ROC-AUC) | | | | |
|---|---|---|---|---|
| Model | Sherlock UK (fraud) (27340,42) | Sherlock FR (fraud) (35478,30) | Sherlock DE (fraud) (58292,30) | Easy-Collect UK (collection) (12183, 55) |
| EBM | 0.956 | 0.966 | 0.981 | 0.763 |
| XGBoost | **0.959** | **0.973** | **0.984** | **0.770** |
| LightGBM | 0.955 | 0.965 | 0.981 | 0.770 |
| Random Forest | 0.916 | 0.965 | 0.979 | 0.761 |
| Logistic Regression | 0.814 | 0.910 | 0.965 | 0.694 |

Figure 24: Other models Tests