**Table of contents**

[1. Literature Review on Credit Scoring 5](#_Toc106060813)

[**1.1 History of credit scoring** 5](#_Toc106060814)

[**1.2 What is credit scoring** 7](#_Toc106060815)

[**1.3 What are scorecards** 8](#_Toc106060816)

[**1.4 Credit scoring modelling characteristics** 10](#_Toc106060817)

[**1.5** **Best practices in Credit Scoring** 11](#_Toc106060818)

[**1.6** **Classical credit scoring approaches** 13](#_Toc106060819)

[**1.7** **Obstacles of classical scoring approach** 14](#_Toc106060820)

[**1.8 Economics behind credit scoring** 15](#_Toc106060821)

[2. Machine learning in credit scoring 17](#_Toc106060822)

[**2.1. Parametric models used in credit scoring** 17](#_Toc106060823)

[**2.1.1. Linear discriminant analysis** 17](#_Toc106060824)

[**2.1.2. Logistic regression** 20](#_Toc106060825)

[2.1.2.1 Origins of logistic regression 20](#_Toc106060826)

[2.1.2.2. The logic of logistic regression 21](#_Toc106060827)

[2.1.2.3. Estimation of coefficients and goodness of fit 25](#_Toc106060828)

[2.1.2.4. interpretation of logistic regression coefficients 28](#_Toc106060829)

[**2.1.3. Regularisation methods** 30](#_Toc106060830)

[2.1.3.1 Ridge Regression 31](#_Toc106060831)

[2.1.3.2 Lasso Regression 33](#_Toc106060832)

[2.1.3.3 Elastic net 34](#_Toc106060833)

[**2.2 Non-Parametric models used in credit scoring** 35](#_Toc106060834)

[**2.2.1 Decision Trees** 35](#_Toc106060835)

[**2.2.2 Random Forest** 41](#_Toc106060836)

[**2.2.3 Extreme Gradient Boosting** 43](#_Toc106060837)

[**2.3 Problem of multicollinearity** 46](#_Toc106060838)

[**2.4 Omitted variables problem** 55](#_Toc106060839)

[3. Important remarks in credit scoring modelling 58](#_Toc106060840)

[**3.1 Regulatory environment** 65](#_Toc106060841)

[**3.2 Regulations for capital requirements** 68](#_Toc106060842)

[**3.3 Definition of default** 70](#_Toc106060843)

[**3.4 IRB and IFRS9 modelling approaches** 73](#_Toc106060844)

[4. Algorithmic fairness 78](#_Toc106060845)

[**4.1 Fairness in a machine learning context** 78](#_Toc106060846)

[**4.2 Unwanted bias in machine learning** 80](#_Toc106060847)

[**4.3 Important fairness metrics** 83](#_Toc106060848)

[**4.4 Unwanted algorithmic bias mitigation** 88](#_Toc106060849)

[**4.4.1 Pre-processing bias mitigation approaches** 89](#_Toc106060850)

[4.4.1.1 Reweighing 89](#_Toc106060851)

[4.4.1.2 Disparate impact remover 90](#_Toc106060852)

[**4.4.2 Post-processing bias mitigation approaches** 91](#_Toc106060853)

[4.4.2.1 Equalised odds 91](#_Toc106060854)

[4.4.2.2 Reject option classification 91](#_Toc106060855)

[5. Methodology and Data 93](#_Toc106060856)

[**5.1 Methodology** 93](#_Toc106060857)

[**5.1.1. Bias detection** 94](#_Toc106060858)

[**5.1.2. Scorecard development** 95](#_Toc106060859)

[**5.1.3. Bias mitigation** 95](#_Toc106060860)

[**5.2 Data** 97](#_Toc106060861)

[6. Results and discussion 97](#_Toc106060862)

[7. Conclusions 97](#_Toc106060863)

[References 98](#_Toc106060864)

**List of tables**

[Table 1. Example of impact of classed variables on a credit score 8](#_Toc106060231)

[Table 2. Example of an application scorecard 9](#_Toc106060232)

[Table 3. Impact of credit scoring on businesses 16](#_Toc106060233)

[Table 4. Relationship between probabilities, odds, and logged odds 23](#_Toc106060234)

[Table 5. ANOVA results on generated and 48](#_Toc106060235)

[Table 6. Meaning of segments in a Venn diagram for multiple regression 50](#_Toc106060236)

[Table 7. Bank’s healthy balance sheet - Example 60](#_Toc106060237)

[Table 8. Bank’s balance sheet with defaulted loans- Example 61](#_Toc106060238)

[Table 9. Bank’s balance sheet with defaulted loans- Example 62](#_Toc106060239)

[Table 10. Definition of default example 72](#_Toc106060240)

[Table 11. Risk weight per exposure category – Standard IRB 74](#_Toc106060241)

[Table 12. Comparison between IRB and IFRS 9 77](#_Toc106060242)

**List of figures**

[Figure 1. Example of customer classification using LDA 18](#_Toc106060097)

[Figure 2. Customer classification projected on a one-dimensional graph 18](#_Toc106060098)

[Figure 3. Comparison of a linear and logistic curve for a binary classification problem 22](#_Toc106060099)

[Figure 4. Comparison of the odds and logged odds against the probability 24](#_Toc106060100)

[Figure 5. Linear regression for logged-odds against a single regressor 25](#_Toc106060101)

[Figure 6. Ridge regression contour curves with restriction 32](#_Toc106060102)

[Figure 7. Lasso regression contour curves with restriction 33](#_Toc106060103)

[Figure 8. Elastic net contour curves with restriction 34](#_Toc106060104)

[Figure 9. Components and sample space view of a binary decision tree 36](#_Toc106060105)

[Figure 9. Pairwise relationships plot for the generated dataset 47](#_Toc106060106)

[Figure 10. Size of variation of and using the sum of squares 48](#_Toc106060107)

[Figure 11. Shared covariance between generated and 48](#_Toc106060108)

[Figure 12. Shared covariance between generated , and 49](#_Toc106060109)

[Figure 13. Effects of multicollinearity on multiple regression 50](#_Toc106060110)

[Figure 14. Effects of multicollinearity on parameters estimates 53](#_Toc106060111)

[Figure 15. Effects of multicollinearity on parameters variance 53](#_Toc106060112)

[Figure 16. Effects of multicollinearity on the standard errors 54](#_Toc106060113)

[Figure 17. Effects of multicollinearity on the standard errors 54](#_Toc106060114)

[Figure 18. Effects of multicollinearity on the standard errors 55](#_Toc106060115)

[Figure 19. Contributions of sectors to the net financial wealth of the euro area 59](#_Toc106060116)

[Figure 20. How capital keeps banks safe 63](#_Toc106060117)

[Figure 21. Loss distribution – expected and unexpected losses 64](#_Toc106060118)

[Figure 22. Three Pillars of Basel II 67](#_Toc106060119)

[Figure 23. Regulatory Financial Bodies in the EU 68](#_Toc106060120)

[Figure 24. Timeline of most recent regulations 70](#_Toc106060121)

[Figure 25. DPD +90 trigger example 71](#_Toc106060122)

[Figure 26. Forward-looking stress tests IFRS 9 76](#_Toc106060123)

[Figure 27. Scorecard development flowchart 81](#_Toc106060124)

[Figure 28. Confusion matrix metrics 84](#_Toc106060125)

[Figure 29. Effect of Disparate Impact Remover algorithm 90](#_Toc106060126)

[Figure 30. Fairness pipeline using AIF360 96](#_Toc106060127)

# **1. Literature Review on Credit Scoring**

This section introduces and explores the concept of credit scoring from a theoretical and practical point of view, firstly understanding the origins and the development of this practice over the recent decades until nowadays. This chapter aims to present a comprehensive definition of credit scoring, covering its characteristics and main statistical aspects in the process of credit score modelling, as well as the importance of credit scoring on a daily basis.

## **1.1 History of credit scoring**

Credit has been present in human history since the beginning of civilisation, and it can be dated back to 5000 years ago (Thomas et al., 2017). During the ancient Mediterranean period, the economy was entirely based on agriculture, and it was the primary source of wealth. Agricultural economies were characterised by having many ups and down due to seasonal changes, affecting the time of payments of buyers. Some people had a surplus of crops, while others needed this surplus but not the means to pay for it. The earliest references to banking and credit arrangements are exposed on stone tables in 2000 BC (Lewis, 1992) and in the Code of Hammurabi around 1726 BC. It is fully extended until the end of the roman empire in which money was created, and credit started to be known as today (MacDonald & Gastmann, 2017).

Creditworthiness scoring has been among the oldest methods both in data analytics and risk management, and it is essential towards evaluating credit applications (Thomas et al., 2017). Credit scoring as we know it today dates to the 1950s. Back then, there was no such thing as a credit score. For people to get loans, they had to hold extended interviews with credit officers at the banks in which a decision was made based on individual judgement. Regarding this old system, there were two big problems. First, the loan decision depended on a subjective assessment which was not an accurate way of determining whether a borrower would pay back a loan or not. Second, the discriminatory bias regarding racial and gender characteristics of the applicants (Abdou & Pointon, 2011).

Although credit scoring is relatively new today, the literature points out the existence of credit scoring attempts in the USA in the late 19th century during the creation of the first credit bureau on the east coast. They collected information about people to sell to landlords, retailers, and anyone interested. Most of the information collected was related to the consumption habits, debts, gambling and drinking issues of the person in question (Lauer, 2017). However, the information was not statistically analysed during the decision-making process but was more subjective. Later, through the development of technology and the ability of companies and banks of gathering data, some first quantitative approaches arose to credit scoring in the 1930s. Some retailers, banks and financial institutions introduced a point-based system for identifying who would and would not pay back a credit based on an application form that included information on the applicant's occupation, age, race, marital status, income, the neighbourhood of residence, etc. (Lauer, 2017). Also, the first records of statistical analysis on credit data were done by David Durant, which used the discriminant analysis techniques to analyse instalment loan data and determine whether some loans were good or bad (Anderson, 2007).

In the 1950s, Bill Fair and Earl Isaac created the Fair, Isaac and Company (FICO), intending to develop an impartial and standardised scoring system. During the launch of the credit scoring system, only the American Investment Company decided to use the scoring method based on statistical analysis. However, there was considerable resistance to banks' use of such a system since there was no means to apply it out of the papers. However, after the 1960s, companies started to computerise data from the customers, and the FICO scoring system was widespread throughout the USA. Some standardisations of defaults were brought into action, for example, the ones to mark late payments as 30, 60 or 90 days behind. By this time, most credits were granted to companies, and the share of granted credits to individuals was too small.

During the 1960s and the arrival of credit cards, banks realised the importance of credit scoring; the number of new applicants for credit cards made it impossible for banks to have credit scoring as a non-automatised lending decision. The growth of computing power made the credit scoring adoption widespread by the 1980s. It used the system for credit cards and other financial products, such as mortgages, personal loans, etc. Also, this time, new statistical techniques were introduced, specifically logistic regression and linear programming. Those techniques are still being used; however, as computers and technology develop, newer and more precise techniques are put into action nowadays that provide higher quality results in credit scoring, for example, artificial intelligence (AI) and machine learning (ML) based algorithms.

## **1.2 What is credit scoring**

Credit scoring can be defined as a set of statistical techniques in the form of a group of automatised decision models that aim to determine the probability of default of a customer, helping the lender assess whether to grant a loan. These techniques are used to establish how much of the loan credit a borrower should get and magnify the lenders' profitability (Thomas et al., 2017). Being creditworthy is not an attribute of a particular person; however, it is an assessment done by the lender when evaluating the borrower's profile using credit scoring. In other words, credit scoring determines the probability that a borrower will be "good" or "bad" (creditworthy or uncreditworthy) based on the profile of the borrower, as well as the economic scenario, potential losses, churns, and approval rates. Together, these factors are relevant in the credit adjudication process (Siddiqi, 2013).

Businesses' mission is to create value and maximise profits (Handy, 2002). Identifying those "good customers" is very important in the financial industry. To this, banks and risk managers constantly use credit scoring techniques to select those "right" customers (low-risk customers) and implement a marketing strategy to offer financial products and get some profit. It is important to remark that one of the primary sources of revenue of banks is lending money and charging an interest rate to the borrower based on the probability of default of the loan (risk), this interest rate is known as the price of money, therefore the higher the risk of the loan the higher the interest rate related to it (Lee & Hogarth, 2018). For banks to maximise profits, they need to lend as much as possible to low-risk customers or reduce costs related to the lending process by limiting granting loans to customers with a high probability of default (high-risk). The problem is that sometimes these low-risk customers do not need loans or financial products, and they reject the banks offer regardless of the effort put in the marketing campaign (Siddiqi, 2017), and that's why banks and financial institutions make a massive effort in identifying those high-risk customers that may default a loan. Banks and financial institutions have a vast history of their customers' data, along with the application forms when applying for a loan. The information gathered by the banks is presented generally as a form of a scorecard where the characteristics of the profile of a given customer (borrower) have a score, and the sum of these scores determines whether a loan will be granted or not to a customer (Thomas et al., 2017). Credit scoring is ultimately used to facilitate business decision-making, although it is often associated with the statistical techniques and processes used in the scorecard's development.

## **1.3 What are scorecards**

As explained before, credit scoring uses predictive statistical methods to classify borrowers by their probability of being a "good" or "bad" in the future, based on the lender's past experiences and the profile of the borrower. Credit scoring models are presented in different forms, but most are presented in the form of regression (Anderson, 2007).

|  |  |
| --- | --- |
|  | (1) |

Where is the dependent variable associated with the outcome that refers to the probability of default, being 0 for "bad" and 1 for "good" customer, it may also come as a form of "logit" or "probit" depending on the model specifications. Being is the set of independent variables as a form of an original, transformed, or dummy variable. stands for the regression coefficient by which the independent variables are weighted, indicating the relative importance, and is the regression error that cannot be captured by the model.

Regression coefficients are used to present a model that attempts to explain the relationship between the independent variables and the independent variable . Traditional scorecards use classed variables in which scores are given if a specific condition holds true based on the model coefficients, for example:

|  |  |  |
| --- | --- | --- |
| **Condition** | **Action** | **Score** |
| Age < 25 | Deduct | 20 |
| Age > 40 | Add | 25 |
| If owning a house | Add | 40 |
| If renting a house | Deduct | 15 |
| If married | Add | 20 |
| If single | Deduct | 20 |
| etc | … | … |

Table 1. Example of impact of classed variables on a credit score

Source: own work

The scorecards can also be presented in a tabular format as shown in the following table in which the characteristics are compressed, and attributes are columns assessed on the score (points) inspired on a FICO scorecard:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Characteristic | Attributes | | | | | Points |
| Years at address | <3 years  30 | 3– 6 years  36 | >6 years  38 |  | Blank  20 | 30 |
| Years at  employer | <2 years  30 | 2-8 years  39 | 9-20 years  43 | >20 years  69 | Blank | 43 |
| Accommodation status | Own  41 | Rent  32 | Parents  35 |  | Other  36 | 32 |
| Civil status | Married  40 | Single  22 | Divorced  32 |  | Blank  32 | 32 |
| Past experience | None  3 | New  15 | Updated  30 | Past due  -5 | Write-off  Reject | 15 |
|  |  |  |  |  | Final Score | 152 |

Table 2. Example of an application scorecard

Source: based on Anderson, 2007.

The scorecard characteristics are usually obtained from different sources of data to which the lender has access during the application process. For building a scorecard, it is considered behavioural, financial, socio-economic, and demographic data (Martin & Evzen, 2006). Regarding the sources of this data, most of the data used for evaluating the application comes internally, from the application form, credit bureaus, public, commercial registers, financial statements, tax statements, integrators of financial information, and national statistical offices (Kaszyński et al., 2021). Naturally, the variables to consider may vary if the loan applicant is an individual or a company. To this, Kaszyński et al. (2021) define a list of potential variables to include in the traditional scorecard depending on the type of applicant. For lenders, it is essential to know the primary source of income, disposable income, alternative sources of income, type of activity, stability of fixed expenses, assets, saving rate, credit history, delinquencies, and geolocation from a customer. However, for companies, a lender would also be interested in knowing the nature of business, business history, management board, balance sheet, income statement, cash flow, suppliers, clients, delinquencies, and exposure to cross border risks.

Based on the score that a borrower gets from the application form, a lender may decide to grant a loan or reject it. In particular, the total score is a measure of the risk associated with the loan for an application, and the lender (depending on the risk appetite) may grant high-risk loans that comply with a particular score cut off to which a higher interest rate will be charged, or reject it.

For example, based on the previous table, the lender may decide to reject all loans that score less than 150 points, those between 150 – 160 points may be charged a higher interest rate, and those who will get a score higher than 160 will be granted a loan automatically. For those high-risk applicants, the lender does not necessarily need to charge a higher interest rate but may request additional conditions such as assigning a lower loan/credit, charging a higher premium on insurance, asking for default insurance, request for extra documentation on assets, etc. The "due diligence" policy may depend on the expected approval rate and revenue or profit potential at each risk level (Siddiqi, 2017).

The attributes in the scorecard are assigned points based on statistical techniques. There are different ways to calculate the weights of the attributes in a scorecard depending on the kind of credit scoring model used, and these models provide the predictive strength of the characteristics. The literature defines two kinds of predictive modelling techniques in credit scoring: parametric modelling that makes assumptions on the data and non-parametric modelling that does not make any assumptions. (Abdou & Pointon, 2011; Anderson, 2007; Lauer, 2017). Also, other factors are considered when calculating the attribute scores, such as the correlation between them and operational factors (Siddiqi, 2013). Lastly, the total score is calculated by summing all scores of the applicant's attributes.

## **1.4 Credit scoring modelling characteristics**

Different statistical methods have arisen over the years. At the very beginning, it was mentioned that essential statistical tools were used for credit scoring, and more sophisticated statistical methods were taking over. Most of the techniques used in credit scoring techniques are within the field of predictive modelling, and these can be classified into two sections:

**Parametric modelling techniques**: The main feature of this type of modelling technique is that it makes assumptions about the data, such as Linear Discriminant Analysis and Logistic Regression. The main benefit of parametric models is that they are simple and easy to interpret and understand. These models are fast computationally speaking, and require less data for training purposes. On the other hand, parametric models are highly constrained, they have a limited complexity, and it is essential to mention that the literature points out that these methods many times fail to match very complex underlying processes (Puertas et al., 2003)

**Non-parametric modelling techniques**: These models do not make assumptions on the underlying process and have more flexibility in adapting any functional form from the training data. Some of these models are Pondered Regression, Regression Trees, Algorithm C4.5, Multivariate Adaptative Regression Splines, Random Forest, Extreme Gradient Boosting, Support Vector Machines, and Neural Artificial Nets (Kennedy, 2013; Knutson, 2020; Puertas et al., 2003)

## **Best practices in Credit Scoring**

Credit scoring heavily depends on data, and during the scorecard development, there are many data considerations. The lender can build a credit scoring model that predicts future default events using statistical methods with the collected data. When building a credit scoring model, some factors must be taken into account so the reliability of the final model will not be compromised. In this sense, the literature highlights 4 important factors to consider (Abdou & Pointon, 2011; Anderson, 2007; Martin & Evzen, 2006; Onay & Öztürk, 2018; Siddiqi, 2017):

* Transparency: The information can be used for assessment,
* Structure: The data is easy to analyse data considering its form,
* Data quantity: There are enough observations for model development,
* Data quality: consistency and accuracy of the data.

Anderson (2007) defines the concept of best practices as *"…processes, techniques, methodologies, and the use of technology, equipment, and resources that have a proven record of success at providing the desired result*". In order to ensure the quality of the credit scoring model, the data must be pre-processed and analysed in advanced so that we can make sure of the precision of the estimations. Some practices and factors are industry-specific. In credit scoring modelling, some general best practices in credit scoring modelling are explained by Kaszyński et al. (2021) as follows:

Firstly, for building a reliable credit scoring model, we need extensive datasets to ensure that our population sample for training a model can converge to the entire population, and therefore we can get more accurate estimates (Martin & Evzen, 2006). For this, it is essential to understand the target population, meaning that when building a scoring model, the scorecards should not be separated based on different products (credit card, cash loans, etc.) or segmentate the scorecards based on the characteristics of the type of applicant (size of the company, balance sheet, etc.). This can lead to a more complex architecture, and it lowers the number of observations in each of the specific models, and therefore it affects the model stability. However, the target population should be defined considering the available information avoiding the loss of observations due to segmentation. The idea is to treat certain atypical observations (clients) in the training sample of the model as "outcome exclusions" and minimise the number of observations that may distort the model.

Secondly, we must understand the data and assess the kind of information we get for training a scoring model. The financial and banking sector, in general, is vastly regulated, and certain rules apply for processing and modelling the data (Knutson, 2020; World Bank Group, 2019). It is important that the training model has as many relevant explanatory features that help to correctly assess the probability of default of a customer. Although the intuition is to include as many variables as possible, we must pay extra care when handling the data and avoid violating the privacy of the customer. European authorities warrant customers' right to explanation and acknowledge of data processing in the latest data protection regulation (*EU General Data Protection Regulation (GDPR): Regulation (EU) 2016/679*, 2016). To this, customers must be informed that their data is being handled and assessed. Although this principle applies in the European Union, some countries are applying similar legislation; therefore, we need to make sure that the data in our training sample complies with the current legislation (Demajo et al., 2020). If some "sensitive" information is detected in the training sample that does not comply with the legislation, then it should be removed.

Thirdly, we must keep in mind the data feed process and understand how the data is updated in the database. The training data generally contains a combination of internal data as well as data from external sources (as explained above), and this external data, in some cases, can be not updated. Therefore, the training dataset must be prepared in a way that these time differences are spotted and reflected. As a rule of thumb, credit scoring models must be trained with the same data feed in which they will be used in production. Furthermore, it is important to keep track of the data feed and ensure the inflow of up-to-date data and avoid situations when credit bureaus stop data feed to subscribers that breach the reciprocity agreement (Anderson, 2007).

Fourthly, we must remove visible rubbish from the data set in the sense that it must be included for training purposes those variables that are strictly relevant for modelling issues and regulatory standards. A few examples of variables that are irrelevant are those with variable coverage in time, optional variables that may come from the application form, dummy variables that indicate missing data, categorical variables with many non-hierarchical categories, customer identification data, unprocessed transaction data, and data compromising the privacy of the customer. A good practice is to create summary variables for each unprocessed transaction data that contains counts and frequencies.

Fifthly, there must be tight control on the quality of the outcome variable. Some of the good practices addressed by Kaszyński et al. (2021) in this issue are the following: get a clear definition of delinquency, mind that a restructured loan is usually a bad loan, First-in-first out (FIFO) is a better definition for a bad day's past due than Last-in-first-out (LIFO), there are alternative indicators for bad behaviour according to the European Bank Authority guidelines for products with compromised days past due, better to use a less precise "bad" definition than a more precise one but less consistent, the outcome variable must be defined taking into account all products on the customer level, and lastly, inactive clients should be marked as outcome exclusions.

Finally, it is important to emphasise the importance of Basel compliance, the Internal Rating-Based Approach, and International Financial Reporting Standards (IFRS) 9. These regulatory modelling approaches will be further discussed in section 3.1.3.

## **Classical credit scoring approaches**

Around the 1950's most of the credit scoring decisions were based on subjective analysis, and due to the lack of technology, many processes couldn't be automatised. As explained in section 1.1, some attempts were made to use statistical techniques to classify customers based on their previous experience; however, the usage of statistical techniques became more popular around the 1980s. The adoption of parametric statistical models in credit scoring started at an earlier stage with the linear discriminant analysis and later the widespread adoption of the logistic regression. These parametric models will be further explained in section 2.1.

The classic credit scoring modelling approaches consider some ideal requirements that made easy the adoption in the industry in the early 1980s. Mainly, these models were suitable for the task at hand, which is a classification problem; also, these methods sometimes are fast to learn from the data and quick to develop in production. Also, these methods can accommodate problems that are specific to a particular development, and lastly, the output of these models is very transparent, making them easy to interpret and explain, meaning that there is not a black box behind the model explaining the underlying process (Anderson, 2007). The advantages can be summarised in high interpretability, fast implementation and therefore, easy auditability.

As mentioned before, classic scoring models are based on parametric statistical models that rely on assumptions on the training data. Manipulating the data to comply with these assumptions is a process that must be done manually compared to more advanced statistical learning techniques.

## **Obstacles of classical scoring approach**

In the context of classical credit scoring models, there are many assumptions on the data and the interactions between the features that are methodologically important for ensuring quality estimates and for creating top-notch scorecards. As mentioned in previous sections, the to-go statistical model in credit scoring is the logistic regression. However, when modelling a binary response variable (default or not default) using the logistic regression, the data must satisfy independence of errors, absence of multicollinearity, a linear relation between the covariates and the logit of the response, no missing data and lack of outliers (Stoltzfus, 2011).

Modelling a response variable is far from being a way to input data to an algorithm and retrieve estimations. Under the classical modelling approach, there is a manual and time-consuming process behind the scenes that aims to prepare and maintain the data in order to have the expected quality according to the assumptions of the model so that it can be served to the algorithm and get the best estimates. Also, during the modelling phase, the modeller must dedicate some effort to selecting the best model and validating it accordingly. Kaszyński et al. (2021) explain that in the case of a bank, this manual process of data quality and maintenance can be multiplied for each of the existing scoring models per product or client and a whole team or department are dedicated to this process in full. Moreover, classical models fail to capture non-linearities between the covariates and the response variable; instead, the features must be transformed in a way that makes the interpretation difficult. In the case of the logistic regression, it belongs to the family of the Generalised Linear Models (GLM) that, with the help of a link function, allows to model non-linear relationships, meaning that the relationship is linearly modelled in terms of the expected response of the link function and the explanatory variables (DeMaris et al., 1990). Most of these manual steps described above are eliminated when using non-parametric modelling approaches.

## **1.8 Economics behind credit scoring**

As explained in previous sections, back in the 1950s, there was not a clear scoring method for credit applications. In fact, credit scoring, as known nowadays, became possible due to the development of computers that brought the industrial revolution in the financial sector due to the automatisation of many processes, especially the accounting and billing systems. Credit scoring is a way to automatise a process based on statistical techniques, which became the standard method in volume-driven retail credit markets.

Those early lenders that adopted credit scoring models as a way to filter out credit applications managed to shift those customers with higher risk to default to other lenders less able to detect those bad customers, increasing the chances of adverse selection, leading to higher default rates and therefore the success in the market for those early adopters of credit scoring. Also, those early adopters experienced a transformation in the whole structure of the organisation, changing old practices and teams; especially, the wide adoption of computers and access to online electronic banking modified the skill requirements into a more tech-focused type of employees.

Moreover, the credit market has experienced tremendous growth since the adoption of credit scoring models; Furletti (2002) explains that decision automation lowered the costs of extending credit, this led to an improvement of the lender’s capacity to grant smaller loans while at the same time increased service level generally.

When analysing the effects from the business point of view on the usage of credit scoring automation process Anderson (2007) defines the following table:

|  |  |  |
| --- | --- | --- |
|  | Positive effects | Negative effects |
| Operations | - Fast  - Consistent  - Objective and defendable  - Comprehensive  - Grater reach | - Complexity data dependent |
| Finance | - Cost savings  - Reduced bad debts  - Reduced security costs | - Capital intensive  - Lost information rents |
| Strategy | - Controllable  - Adaptable | - Backward-looking |
| HR | - Improved human capital allocation | - Skills sensitive  - Staff acceptance |
| Customer | - Lower transaction costs  - Improved credit access  - More choices and channels  - Mobility between lenders | - Impersonal demands  - Lack of direct communication |

Table 3. Impact of credit scoring on businesses

Source: Compiled from Anderson (2007)

From the above table, we can appreciate that positive effects impact more on the decision of adoption of credit scoring as an automation process on the business side. In the operational structure of the business, we can highlight the speed, consistency, and comprehensiveness of the decisions, also allowing larger geographical reach, but at the cost of complexity. From the financial point of view, reduced bad debts less collateral management while improving cost savings at the expense of higher capital investment and lost information rents. The strategic structure improved control over strategies allowing monitoring at a very detailed level that can be adapted to changing circumstances, at the cost that these models are backwards-looking. Similarly, the human resource structure changed into a more productive staff allocation with serious problems of acceptance, at the cost of scarcity of the new required skills. Lastly, the customer improved its access to credit at a very lower cost, also access to different lenders but at the expense of losing the traditional relationship with the bank. Credit scoring as an automation decision process reduces the adverse selection because of improved decision-making. Moreover, credit decisions can be defended, reducing the possibility of unfair discrimination, and models can be trained in a way that avoids unfair discrimination from behavioural training data. Finally, lenders can get analytics from KPIs to check the performance of the business providing enough information to forecast price for risk, portfolio values, and in some cases, trade debt with other counterparties

# **2. Machine learning in credit scoring**

As explained in previous sections, machine learning techniques are widely used in credit scoring due to the simplification of complex scoring decisions, aiming to generate real-time predictions and reduce credit risks. Formally speaking, machine learning is defined as the capacity given to computers to learn without being explicitly programmed for a particular task (James et al., 2021). The learning process of a machine learning algorithm starts with training from a dataset. The algorithm will capture and learn all unobserved relationships between the dataset's attributes. In this chapter, we will dig further into the most common modelling techniques used nowadays in credit scoring and the main problems that may arise in this process.

## **2.1. Parametric models used in credit scoring**

Nowadays, credit-scoring systems are based on statistical and operational research methodologies that allow automatisation of the scoring process, increasing productivity and predicting defaults in a faster way. They are among the most effective and lucrative uses of statistical theory (Langdon et al., 1992). In previous sections, it was mentioned two different approaches used in credit scoring modelling. While parametric modelling techniques make several assumptions about the underlying data, non-parametric techniques make few (Anderson, 2007). The parametric models, including linear discriminant analysis, logistic regression, and regularisation methods such as ridge regression, lasso regression and elastic net, are discussed below.

### **2.1.1. Linear discriminant analysis**

Linear Discriminant Analysis (LDA) is a widely used technique for dimensionality reduction and classification. LDA enables class separability by establishing a decision region between the various classes by maximising the distance between the means of the classes and by minimising the between-class variance and the within-class variance (Mohanty et al., 2013). Sir Fisher R. A. (1936) developed the linear discriminant function, trying to identify the set of factors that best divided two groups using accessible attributes, and LDA came later as a simple generalisation of this function. In credit scoring, the two categories are those categorised as good and bad customers by the lender and the characteristics are the details in the application form and credit bureau information.

Chart, scatter chart

Description automatically generated

Figure 1. Example of customer classification using LDA

Source: own work

Figure 1 is an example of credit scoring in which the classes (good and bad customers) are separated by the dashed line defined by LDA. These data points are projected into the dashed line reducing this two-dimensional graph into a one-dimensional graph. That's why this method can be considered as a dimensionality reduction algorithm. However, when the mean of the distributions is the same (or shared), this algorithm fails to convert the n-dimensional data into a single one-dimensional graph that makes both classes separable such as:

Shape, rectangle

Description automatically generated

Figure 2. Customer classification projected on a one-dimensional graph

Source: own work

From a mathematical point of view, the representation of LDA consists of statistical properties of the data, which are calculated per class, more specifically, the mean and the variance of each class. LDA assumes that the data follows a normal distribution and that each attribute has the same variance meaning that the values of the variables are centred around the mean by the same amount on average. Keeping that in mind, it is advisable to standardise the data prior to using LDA.

Let our target Y denote a random binary variable with values 1 for reference to the class "default" and value 0 for reference to the class "non-default", and X a single continuous random variable normally distributed with mean around 0 and standard deviation of 1. With the assumptions for LDA satisfied, the mean value of the X variable for each class is calculated as follows:

|  |  |
| --- | --- |
|  | (2) |

Note that k stands for each of the potential classes of the dependent variable, which in this case, there are two classes since the dependent variable is binary. The mean is calculated by dividing the sum of values by the total number of values in the corresponding class (1 or 2). The variance is calculated across all classes using the following formula:

|  |  |
| --- | --- |
|  | (3) |

The variance is calculated as the squared difference of each value from the mean within the class groups while averaging that difference across all class groups. LDA uses the Bayes theorem to estimate probabilities of a value to belong to a particular class k.

|  |  |
| --- | --- |
|  | (4) |

Being the probability of a given class, denoted as the ratio of this observations with this class in the training dataset. Since the independent variable follows a normal gaussian distribution, we can estimate which is the estimated probability of the value belonging to the class k. The denominator normalises across for each class . When inserting the Gaussian into equation (4) and simplifying, we obtain the discriminate function for class k.

|  |  |
| --- | --- |
|  | (5) |

The discriminate function is calculated for each class, and the highest value of this function for a particular class will define the output classification (Brownlee, 2016).

Based on that, we can highlight that linear discriminant analysis (LDA) uses the training observations to determine the location of a boundary between the response classes. The boundary location is determined by treating the observations of each class as samples. Theoretically, we could fit an (n) dimensional normal distribution from a multidimensional normal distribution to the observations in each class. Calculating this involves calculating the mean vector and covariance matrix for each class, as these determine the centre and shape of the distribution, respectively (Johnson & Wichern, 2002; Thomas et al., 2017). Having fitted the distributions, we could draw a boundary between the classes by selecting the set of points where the probabilities are equal. Observations on one side of the boundary would be classified as one class, and observations on the other side as the other class. We can do all this theoretically, resulting in an equation for the boundary that depends on the parameters of the fitted distribution. This means that we don't need to go through the whole process to perform discriminant analysis; we need to calculate the means and covariances and apply the formula for the boundary.

### **2.1.2. Logistic regression**

Logistic regression is by far the most widely used algorithm in credit scoring. Its usage is preferred due to its simplicity and ease of interpretation. Therefore, a huge section of this dissertation is dedicated to covering the logistic regression model.

#### **2.1.2.1 Origins of logistic regression**

The logistic function was developed in the nineteenth century to describe population growth and the behaviour of autocatalytic chemical processes. That being said, the growth rate of a quantity *W* (*t*) over time is given by:

|  |  |
| --- | --- |
|  | (6) |

Assuming that *Ẇ* (*t*) is proportional to *W* (*t*), we get:

|  |  |
| --- | --- |
|  | (7) |

Where *b* refers to the constant rate of growth, thus leading to the exponential growth model

|  |  |
| --- | --- |
|  | (8) |

Where the initial value *W* (0) may replace *A*, however, the exponential growth model is devoid of any upper limit. This problem was approached by the Belgian astronomer and statistician Alphonse Quetelet (1795-1874) and his student Pierre-Francois Verhulst (1804-1849) through the inclusion of an additional term, representing the increasing resistance to further growth, in equation (7) as follows:

|  |  |
| --- | --- |
|  | (9) |

Experimentation with varied forms of *f* led to the following model when *f* is a quadratic function:

|  |  |
| --- | --- |
|  | (10) |

Where W refers to the upper limit or saturation level of *W*.

In the above model, is proportional to both and . Expressing as a proportion of W, or, which results in the following differential equation:

|  |  |
| --- | --- |
|  | (11) |

Solving the above differential equation, we get the so-called logistic function:

|  |  |
| --- | --- |
|  | (12) |

Equation (12) was named as the ***logistic*** function by Verhulst. In regression analysis, a and *b*  may be interpreted as the intercept and the regression coefficient (or the slope of the regression line), respectively. The logistic function was rediscovered in 1920 by Raymond Pearl (1879-1940), the Director of the Department of Biometry and Vital Statistics at John Hopkins University, and his deputy Lowell J. Reed (1886-1966) while studying the United States' population increase (Cramer, 2002).

#### **2.1.2.2. The logic of logistic regression**

Discrete or qualitative rather than continuous or quantitative events are common in many social phenomena; for example, an event happens or does not happen, a person's life can be changed in a variety of ways that involve a characteristic, event, or choice, or large social entities such as groups, organisations and nations can arise or disintegrate, become insolvent, confront, revolt, and so on. A dichotomous indicator or dummy variable is the most common way to represent discrete binary phenomena (Pampel, 2000).

On the surface, a binary qualitative dependent variable seems to be appropriate for use in multiple linear regression. The dependent variable generally assumes only two values of zero and one. However, the estimated values for regression are in the form of mean proportions or probabilities conditional on the values of the independent variables. The regression coefficients may be interpreted as the increase or reduction in the estimated probability of possessing a characteristic or experiencing an event due to a unit change in the independent variables. However, such linear regression faces the following problems:

**Probability ceilings:** Probabilities and proportions, by definition, cannot exceed one (ceiling) or fall below zero (floor). Nonetheless, the linear regression line may stretch upward into positive infinity (or extend downward towards negative infinity) as the values of the independent variables increase (or decrease). A model may provide illogical and useless predicted values of the dependent variable greater than one and less than zero based on the slope of the regression line and the observed values of the independent variables. This can be observed in figure 3.

**Additivity assumption:** Typically, linear regression assumes additivity, which states that the influence of one independent variable on the dependent variable remains constant regardless of the values of the other independent variables. When the value of one independent variable reaches a level sufficient to push the probability of the dependent variable close to one (or close to zero), the impacts of other variables have little effect. Thus, the ceiling and floor impose an intrinsic, non-additive and interactive nature on the impact of all independent variables (Long & Freese, 2014; Pampel, 2000).

**Normality and homoscedasticity assumption:** Linear regression with dichotomous dependent variable violates the normality assumption (each value of the independent variables in the population is associated with a normal distribution of error terms around the predicted value of the dependent variable) and the homoscedasticity assumption (the dispersion of the error terms for each value of the independent variables is similar) (Pampel, 2000).

The relationship between a binary dependent variable and an independent variable is inherently non-linear. A constantly changing curve, such as the S-shaped curve, represents the relationship more smoothly and adequately than a straight line. Although other non-linear functions may depict the S-shaped curve, the logistic or logit transformation has gained popularity due to its desirable characteristics and relative simplicity since it allows to model a fitting curve between 0 and 1, as shown in figure 3.

Chart, line chart

Description automatically generated

Figure 3. Comparison of a linear and logistic curve for a binary classification problem

Source: Own work

Let *Yi* be the observations of a dichotomous dependent variable representing some event and assuming the values of 1 (occurrence of the event) and 0 (non-occurrence of the event) only. Keeping that in mind, given the probability of the occurrence of the event as *,* the probability of the non-occurrence of the event will be . In his paper, Pampel (2000) explains that the odds of the probability of the occurrence of the event relative to the probability of the non-occurrence of the event is then given by:

|  |  |
| --- | --- |
|  | (13) |

Odds are generally expressed implicitly as a ratio to one or as a single number. For instance, if the probability of an event equals 0.3, the odds are (0.3 / 0.7) or 0.428, indicating that the event occurs 0.428 times for each time it does not occur, or 428 occurrences per 1000 non-occurrences. Even though both probabilities and odds have a lower bound (floor) of zero, denoting the increasing likelihood of an event with increasingly large positive numbers, odds have no upper bound (ceiling), unlike probabilities. The transformation of probabilities into odds eliminates the probabilities' ceiling value of one. Now, the logit or logged odds are formed by taking the natural logarithm of the odds to eliminate the odds and hence probabilities' floor value of zero, as:

|  |  |
| --- | --- |
|  | (14) |

Based on the above statements, we can present the odds and the log of odds (logit) for a probability range from 0 to 1:

|  |  |  |
| --- | --- | --- |
| Probability | Odds | Log (Odds) |
| 0 | 0 | NEG INFINITY |
| 0.1 | 0.11 | -2.2 |
| 0.2 | 0.25 | -1.39 |
| 0.3 | 0.43 | -0.85 |
| 0.4 | 0.67 | -0.41 |
| 0.5 | 1 | 0 |
| 0.6 | 1.5 | 0.41 |
| 0.7 | 2.33 | 0.85 |
| 0.8 | 4 | 1.39 |
| 0.9 | 9 | 2.2 |
| 1 | INFINITY | INFINITY |

Table 4. Relationship between probabilities, odds, and logged odds

*Source:* Own work.

When plotting the probabilities against the odds, we realise that it reaches 1 when the probability is 0.5; after this point, the odds tend to infinity. However, the range of log of odds (logit) with respect to the probabilities has a range between negative and positive infinity. Also, the data is squished due to the effects of the logarithms.

Diagram, histogram

Description automatically generated

Figure 4. Comparison of the odds and logged odds against the probability

Source: Own work

Considering the initial formulation of the binary variable with values 0 and 1. From the above explanation, we know that logit of the probabilities of 0 and 1 is negative and positive infinity, which is a key part of the logistic regression process. The non-linear relationship between the independent variable (*X*) and the probability of the occurrence of the dichotomous dependent variable (*Y*) conditional on the observed values of *X* (*Xi*) expressed as *P* (*Xi*) = [ *P* (*Yi* = 1 | *X* = *Xi* )],may be transformed into a linear regression function based on the logit of *P* (*Xi*) from equation (14) as:

|  |  |
| --- | --- |
|  | (15) |

Where *a* and *b* denote the intercept and the regression coefficient (or the slope of the linear regression line), which are estimated using the maximum likelihood method, and the predictions are in the form of logit.

Graphically speaking, this algorithm transforms the values of the dependent variable into the logit (that is, positive and negative infinity) and fits a straight line (as in linear regression) to understand where these points would fit on this line. Then the projected values on the line are extrapolated to the Y-axis, providing the predicted logit for these data points of the variable X, having the actual and predicted logit based on the regression line, as shown in figure 5.

Chart, line chart

Description automatically generated

Figure 5. Linear regression for logged-odds against a single regressor

Source: Own work

In order to convert the predicted logit back to probabilities so that it can be plotted as in figure 3, this algorithm uses the sigmoid function from equation (12) which is the inverse of the logit, and substitutes from equation (15) to get the logit predictions in the form of probabilities as follows:

|  |  |
| --- | --- |
|  | (16) |

In that manner, the predicted logit can be transformed into predicted probabilities. However, it is important to remark once again that the interpretation of the coefficients is based on the change to the predicted logit, not on the probabilities.

#### **2.1.2.3. Estimation of coefficients and goodness of fit**

In the case of a binary logistic regression model, we can estimate the coefficients using different methods. The most popular in the field of credit scoring and machine learning is the Maximum Likelihood Estimation (MLE). Due to the binary nature of the dependent variable, the error term does neither have a normal distribution nor equal variances for the values of the independent variables. As a result, the estimating approach derived from the Ordinary Least Squares (OLS) criterion, which involves minimising the sum of the squared deviations between the observed and predicted values of the dependent variables, fails to produce efficient estimates. Therefore, rather than using OLS, logistic regression depends on maximum likelihood algorithms to estimate the coefficients.

For logistic regression, the estimation of the regression coefficients begins with an expression for the likelihood of observing the pattern of occurrences (*Y* = 1) and non-occurrences (*Y* = 0) of an event or characteristic in each sample. This expression, termed the likelihood function, depends on unknown logistic regression parameters. Maximum Likelihood Estimation (MLE) finds the model parameters that provide the maximum value for the likelihood function, thereby identifying the estimates for model parameters that are most likely to give rise to the pattern of observations in the sample data. In order to estimate the parameter of the logistic regression function, first, we need to write down the log-likelihood function, then we need to optimise this function to find the values of the coefficients that maximise the log-likelihood function.

Since our dependent variable is binary, we use the corresponding density function of a Bernoulli distribution that allows us to write the probability of one data point.

|  |  |
| --- | --- |
|  | (17) |

Based on the above density function, we can write the likelihood function as follows:

|  |  |
| --- | --- |
|  | (18) |

From equation (18), we can combine the products and take the log of both sides of the equation to get the log-likelihood function. Due to the properties of logarithms, the products are transformed into summation as follows.

|  |  |
| --- | --- |
|  | (19) |

The above equation is the cost function to optimise in order to find the set of parameters that maximises log-likelihood. This maximisation problem could be solved by calculating the derivative of the log-likelihood function and setting it to zero. However, there is not a closed-form for the maximum of this function. Therefore, we must use an optimisation algorithm to compute the coefficients (Pampel, 2000). Since the log-likelihood equation is usually non-linear, general-purpose optimisation algorithms, such as Gradient Ascent, Newton-Raphson and Fisher Scoring, are also used for finding the global maxima of the function and estimating the coefficients. It is important to emphasise that logistic regression using local optimisation algorithms such as gradient descent yields in optimising the function globally.

In order to optimise equation (19) with the gradient ascent algorithm, we define the partial derivative of the log-likelihood with respect to each parameter:

|  |  |
| --- | --- |
|  | (20) |

The idea of the gradient ascent algorithm is to take the direction of the gradients (partial derivates), pointing to where the function is maximised and continuously taking small steps in that direction until the function is maximised. This is an iterative process that updates the coefficients based on the values of the previous ones as follows:

|  |  |
| --- | --- |
|  | (21) |

Where is the step size that controls how fast the algorithm converges to the maximum. The higher the value of it is more likely to diverge from the optimal solution, whereas if is too small, then it will take more time to converge (computationally expensive). In this case, is a tuning parameter that can be set by the modeller, and mathematically speaking, it only makes sense for to take values between (0, 1). According to Monroe et al. (2017), this algorithm can be simplified with the following pseudocode:

initialise: = [0, 0, ..., 0] (m elements)

repeat many times:

gradient = [0, 0, ..., 0] (m elements)

for each training example ( , ):

for j = 0 to m:

gradient[j] +=

for j = 0 to m:

[j] += \* gradient[j]

return

Once our parameters are estimated, there are various approaches to test the Goodness of Fit (GoF) of a logistic regression model. There are some tests to assess whether the probabilities produced by the model reflect the observed data. In statistical lingo, we can distinguish between the saturated model, which reflects the ideal fit, meaning that the probabilities estimated by the model are equal to observed probabilities for each sample, and the estimated model:

**Deviance and Pearson :** These goodness-of-fit tests are a way to assess how different are the current model and the full model. In this approach, the data represents the fit of the ideal model possible, the saturated model having a separate parameter for each observation. It is tested whether allparameters that are in the saturated model but not in the estimated model equal zero.

|  |  |
| --- | --- |
|  | (22) |

Where are the observed counts, are the counts estimated in the model. Also, the Deviance and Pearson χ2 follow a chi-square distribution. However, the Deviance and Pearson χ2 are not recommended goodness-of-fit statistics in a model with many predictors, especially when those are continuous. Also, it is not recommended in a model with categorical predictors with a considerable number of categories. When one of these tests is rejected, we can be certain that there is a statistically significant lack of fit. Otherwise, there is no lack of fit. In other words, the null hypothesis is that the model has a good fit against the alternative hypothesis that the model has a bad fit.

**Hosmer-Lemeshow**: By contrast to the previous tests, the Hosmer-Lemeshow test compares the observed and expected frequencies of events and non-events, then it compares how the model fits the data (Hosmer et al., 1989). More specifically, it helps to determine whether the predicted probabilities deviate from the observed probabilities. First, it estimates the individual probabilities sorting within response and dividing the set into ten groups with a similar number of observations from 1 to 10. Lastly, it uses the groups to calculate expected counts and compare with observed counts using Pearson statistic with 10-2=8 degrees of freedom

|  |  |
| --- | --- |
|  | (23) |

If the p-value is lower than the significance level, then we reject the null hypothesis that the model fits the data nicely, meaning that there is a lack of fit (Hosmer et al., 1989).

#### **2.1.2.4. interpretation of logistic regression coefficients**

The sign of *b* in equation (16) determines whether the logit is increasing or decreasing as *X* increases. The rate of ascent or descent increases as the magnitude of *b* that is | *b* |, increases; as *b* approaches zero, the curve flattens to a horizontal straight line. When *b* equals zero, *Y* is independent of *X*. For quantitative *X* with *b* greater than zero, the curve for *P* (*X*)has the shape of the cumulative distribution function of the logistic distribution, given by:

|  |  |
| --- | --- |
|  | (24) |

Where *m* and *t* are the mean and standard deviation respectively of the logistic distribution. Due to the symmetrical distribution of the logistic density, *P* (*X*)approaches one at the same rate that it approaches zero (Agresti, 2002). Multiple interpretations exist for *b* in terms of logged odds, odds and probabilities and the nature of the independent variable being continuous or dummy (Pampel, 2000). We can focus on the following ways to interpret logistic regression coefficients:

**Logged odds:** In the case of continuous independent variables, the logistic regression coefficients indicate the change in the projected logged odds of the occurrence of an event for one unit change in the independent variables. In contrast, an implicit comparison of the indicator group with the reference or excluded group is made by a one-unit change in the case of dummy independent variables (DeMaris, 1993). Browne, (1997), for example, uses logistic regression to forecast labour force participation for 922 female heads of home between the ages of 18 and 54 in 1989. For the continuous independent variable 'Years employed', the logistic regression coefficient of 0.13 indicates an increase in the logged odds of the dependent variable' Labor Force Participation' by 0.13 for an additional year of employment. The author also compares the 'Labor Force Participation' with two dummy variables, namely 'High school dropout' and 'High school graduate', to that of the reference group consisting of women with some college education. These two dummy variables have coefficients of -1.29 and -0.68, respectively, indicating that the logged odds of being in the labour force are 1.29 lower for high school dropouts than for those with some college and 0.68 lower for high school graduates than for those with some college education (Pampel, 2000).

**Odds ratio:** Analysing the coefficients in terms of the logged odds is not very intuitive and straightforward in order to draw insights due to the logarithms. However, we can express and interpret the coefficients in terms of Odds, which is the most common way. For this, let us consider the following binary logistic regression model with two independent variables, *X*1 and *X*2 :

|  |  |
| --- | --- |
|  | (25) |

Where ais the intercept term, and *b*1 and *b*2 are the regression coefficients. Exponentiating both sides we get:

|  |  |
| --- | --- |
|  | (26) |

The equation (26) determining the odds is multiplicative, where the predicted value of the dependent variable does not change when multiplied by a coefficient of one. Therefore, the effect of each independent variable on the odds can be measured by taking the antilog of the regression coefficients (DeMaris, 1993). The percentage change in the odds for a one-unit change in the independent variable (with regression coefficient *b*) which is given by .

Now it presents a more meaningful interpretation of the regression coefficients. Concerning Browne's study mentioned above, the exponentiated coefficient for the continuous variable 'Years employed', *e*0.13 or 1.14 indicates that a one-year increase in employment multiplies the odds of the dependent variable' Labour Force Participation' by 1.14 or increases the odds by a factor of 1.14 or 14%. (Pampel, 2000). To summarise, the log odds refer to the logit's additive effect. It tells us how much the logit increases by 1 unit change of the dependent variable, while the odds ratio gives the multiplicative effect on the odds, it tell us how high the odds of y = 1 by one unit change of the dependent variable.

**Probabilities**: This approach consists in calculating at any single point of the logistic curve in Figure 3 the linear slope of the tangent of the non-linear curve. This slope is defined by the partial derivative of the non-linear equation and describes the linear change in the probability for a one-unit change in the independent variable (Pampel, 2000). Note that it just defines the change in the tangent line due to a unit change in the independent variable, but it does not define the change in the logistic regression curve. This is the so-called marginal effect. Following Browne's example, for the variable years employed, we have a logistic regression coefficient of 0.13, the expected value, the probability of an event for the dependent variable is 0.83 and no event 0.17. By multiplying all three values, we get a value of 0.018. Therefore, 1 year increase of employment increases the probability of event by roughly 2% at the mean.

### **2.1.3. Regularisation methods**

In statistical learning modelling, there is a trade-off between minimising the variance and bias of a model. High model bias and low model variance oversimplify the underlying process by increasing the difference between the average prediction and the observed values, making it hard to capture the real pattern of the data, this type of model make huge generalisations resulting in a simple model that does not learn from the training data very well, this is called underfitting in machine learning lingo. On the other hand, high model variance and low bias is a very common situation where the model learns very well from the training data but doesn't generalise enough to unseen data; therefore, there is a high error rate on the test data (Briscoe & Feldman, 2011).

In credit scoring modelling, the training sets have many dimensions, and the models are exposed to the multicollinearity problem (which will be explained in further sections), also to overfitting due to the large complexity of the data (Kaszyński et al., 2021). Having very less complex data to learn from may result in an underfitted model. On the contrary, a very complex training set may result in an overfitted model. in this respect, regularisation techniques are used in statistical learning modelling to add bias to a model. In some cases, they are useful for selecting explanatory variables; therefore, they aim to minimise the variance due to overfitting during the training process.

#### **2.1.3.1 Ridge Regression**

Ridge regression is a regularisation technique that adds some bias to the model by imposing a penalisation term to the cost function of the model. It is usually used to estimate model parameters when there is a high risk of multicollinearity that, as a result, could lead to highly unstable parameter estimates that are hard to interpret and to rely on (Marquardt & Snee, 1975). By imposing a penalisation term on our estimates, there is a reduction in the variance of the model at the cost of biased estimates. In the case of using the binary cross-entropy (also known as log loss) as our cost function to regularise using ridge regression, we can observe the role of the penalisation term in the following expression:

|  |  |
| --- | --- |
|  | (27) |

Where is the set of explanatory variables . In the above equation, works as a tuning parameter that weights the sum of the squared betas. Thus, it controls the amount of bias imposed on the model. In this way, the idea is to tune the parameter in a way that it won't bias too many estimates, but at the same time, it is efficient enough to considerably reduce variance.

Diagram

Description automatically generated

Figure 6. Ridge regression contour curves with restriction

Source: Kaszyński et al. (2021)

On the top right-hand side of figure 6, we find the maximum likelihood estimator, which is the point from the coordinate (0.48, 3.2), the contour curves around this point are the projection of the likelihood function. The aim is to maximise the likelihood function subject to a constraint (instead of getting the maximum likelihood estimate) which is defined as the circles around the point in coordinate (0, 0). Let the first contour curve be our desired restriction around the coordinate (0, 0). We can observe that it has a value of 0.15, which mean that the restriction will be the following:

|  |  |
| --- | --- |
|  | (28) |

The restriction represented by equation (28) is equivalent to adding the penalisation term to the loss function in equation (27), meaning that the solution of the optimisation problem should be within the first circle around the point of coordinate (0, 0). More specifically, the point circled in red from figure 6.

This regularisation method is helpful to deal with when multicollinearity occurs. Under these circumstances, the estimates are unbiased, but their variances are large (far from the true values). By adding more bias to the estimated using ridge regression, we reduce the standard errors making the estimates more stable and reliable and, therefore, reducing the variance of the model. Thus, the bias imposed by will shrink the collinear estimates towards zero, but it will not be able to exclude them from the model unless get a very large value.

#### **2.1.3.2 Lasso Regression**

The Least Absolute Shrinkage and Selection Operator (lasso) regression. This method allows the exclusion of estimates from the model and works as a variable selection method by adding a penalty term that makes the potential solutions of the likelihood function to be restricted to a certain area in which the estimates are equal to zero when maximised the function. (Yang & Wen, 2018). This penalty can be observed as follows:

|  |  |
| --- | --- |
|  | (29) |

In this case, the penalisation term is small than in ridge regression, note that ridge regression imposes the sum of squared betas, while lasso considers the absolute value of betas. Graphically speaking, we can observe in figure 7 that equals zero when is between 0 and 0.28

Diagram, engineering drawing

Description automatically generated

Figure 7. Lasso regression contour curves with restriction

Source: Kaszyński et al. (2021)

The diamond-shaped restriction is due to the absolute value from the penalisation term in the cost function. In this case, the aim of the lasso regression is to work as a variable selection method that reduces model complexity by decreasing the dimensionality of the model. Therefore, the interpretation is simpler than in ridge regression which in contrast shrinks the estimates to zero, and even sometimes the parameter can change sign. In summary, lasso also trades variance for bias but also has embedded a variable selection method.

#### **2.1.3.3 Elastic net**

The elastic netregularisation was introduced by (Zou & Hastie, 2005), which linearly combines the penalties terms used in the previously explained methods. This method combines both properties from the lasso and ridge regression. In this regularisation method, we can find two tuning parameters that can be observed in the following formula:

|  |  |
| --- | --- |
|  | (30) |

It can be noted that is still a tuning parameter as in ridge and lasso, and in addition, we find the parameter that weights the amount of penalty that comes from ridge and lasso. On this detail, when equals 0 ridge penalty is applied, when is lower than 0.5 the elastic net accounts for more penalty weight from the ridge, contrary to that, when is higher than 0.5, there is heavier lasso penalty, and lastly, when equals 1 only lasso penalisation takes place.

In figure 8, we can observe that the contour curves for the restriction are not diamond-shaped nor circular-shaped, instead, it is a combination of both. The restriction has smooth sides and sharped edges, allowing to remove estimates as in lasso, but depending on the tuning parameter, it could only add bias to the estimates.

Diagram

Description automatically generated

Figure 8. Elastic net contour curves with restriction

Source: Kaszyński et al. (2021)

The elastic net has an embedded grouping effect that allows correlated features to be discarded or included in the model. Also, (Zou & Hastie, 2005) proved that contrary to lasso, the elastic net method is useful when the number of variables is larger than the number of rows, this situation can be sometimes frequent in credit scoring, especially with behavioural data (Kaszyński et al., 2021).

It is important to note that (Zhou et al., 2014) demonstrated that the elastic net could be reduced to a support vector machine (SVM) which can be parallelised in a Graphics Processing Unit (GPU) for faster SVM solvers.

## **2.2 Non-Parametric models used in credit scoring**

Prediction techniques are typically referred to as supervised learning in the machine learning scenario. They seek to identify the link between input characteristics (independent variables) and a target attribute (dependent variable). The relationship is expressed in a structure known as a Model. Models often describe and explain phenomena hidden in datasets and may be used to predict the target attribute’s value when the input attributes’ values are known. Supervised models may be categorised into (a) regression models and (b) classification models or classifiers. Regression models convert the input space into a domain with actual values. A regressor, for example, may forecast demand for a particular product based on its attributes. Classifiers divide the input space into predefined categories. For example, classifiers may be used to categorise mortgage customers as good (full mortgage repayment on time) or bad (delayed payback) (Rokach & Maimon, 2015). In this section we will covering non-parametric models which require a few or no assumptions concerning the relationship between a dependent variable and explanatory variables, and the distribution of the variables or the errors. Among these we can find decision trees, random forest, and extreme gradient boosting (XGBoost) discussed below.

### **2.2.1 Decision Trees**

A decision tree is a hierarchical model for supervised learning in which a series of recursive splits determine the local area through decision nodes with a test function. A decision tree is likewise a nonparametric model because no parametric form for the class density is assumed (Kantardzic, 2019). When a decision tree is used to do classification tasks, it is referred to as a classification tree. When used for regression tasks, it is referred to as a regression tree (Rokach & Maimon, 2015). More specifically, when the dependent variable is continuous, regression trees are employed, and when the dependent variable is categorical, classification trees are used (Sullivan, 2017).

A decision tree model’s major components are nodes and branches. Thus, nodes are classified into three categories (Song & Lu, 2015).

* A root node or decision node represents a decision that divides all records into two or more mutually exclusive subsets.
* Internal nodes, or chance nodes, represent one of the alternatives available in the tree structure; the node’s top edge is linked to its parent node, while the node’s bottom edge is related to its child nodes or leaf nodes.
* Leaf nodes, or end nodes, describe the outcome of a series of choices or occurrences.

On the other hand, branches are random events or occurrences arising from the root and internal nodes. A decision tree model is built using a branching hierarchy. Each route from the root node to the leaf node represents a categorisation determination rule. These decision tree paths are often known as ‘if-then’ rules; for example, “*If condition one and condition two and condition...and condition p exist, then result q happens*” (Song & Lu, 2015).

The following figure depicts a basic decision tree model with a single binary target variable Y (0 or 1) and two continuous variables ranging from 0 to 1. As observed, decision trees may also be represented as segmented space. Being exhaustive and mutually exclusive, each segment corresponds to a leaf node in the sample space. Each record is assigned to a unique segment (leaf node). The goal of decision tree analysis is to find the optimal model for segmenting all data (Song & Lu, 2015).

|  |  |
| --- | --- |
|  |  |

Figure 9. Components and sample space view of a binary decision tree

Source: Song & Lu (2015)

Regarding the phases in the development of a decision tree model, the most significant phases are binning, splitting, stopping and pruning which are being discussed as follows:

Binning is a technique of data pre-processing that involves discretising a continuous variable by assigning its values to a series of bins. This strategy helps control the variable by bucketing outliers in the lowest or highest intervals of the range with less extreme values. Using this strategy, outliers become similar to other values in the distribution’s tail. Binning may also help alleviate problems created by the variable’s high degree of skewness. Binning may be unsupervised (bins are constructed purely based on the distribution of the variable) or supervised (bins are produced using external information, e.g. target variable). One of the supervised binning methods is to fit a decision tree to forecast target variables, such as default rates, using just the variable the analyst intends to discretise. The goal is to find a split that minimises the Gini coefficient or entropy (Kaszyński et al., 2020).

Splitting happens to input variables relevant to the target variable that are utilised to divide parent nodes into target variables’ purer child nodes (Song & Lu, 2015). Choosing the best test for partitioning the set requires determining which feature to test and which test results are available. Each node in the same tree may have access to a different test set. Furthermore, tests performed at different nodes may be based on the same variable yet provide different sets of results (Kaszyński et al., 2020). The criteria relating to the degree of ‘purity’ of the generated child nodes, which are used to choose amongst several possible input variables, include, among others, entropy, information gain, Gini index, and Twoing criterion (Patel et al., 2012). This splitting process is repeated until certain homogeneity or stopping conditions are reached. The decision tree model will not require all potential input variables in most circumstances. In other cases, different levels of the decision tree may require a single input variable several times (Song & Lu, 2015).

The stopping criteria indicate if the tree-building process should be stopped. It is important to consider complexity and resilience as two opposing model characteristics that must be assessed simultaneously while designing a statistical model. The more intricate a model is when used to forecast future data, the less trustworthy it is. Building a ubiquitous decision tree model that extends far enough to ensure that the data in each leaf node are completely pure is an extreme situation. Each leaf of such a decision tree would contain few records and would be inappropriate for present facts, thus preventing it from forecasting future occurrences and having little generalisability (Song & Lu, 2015). The stopping criteria avoids overfitting which occurs when a tree is overexpanded to the training set. Consequently, its quality on the test set is often much worse (Kaszyński et al., 2020).

Pruning can be understood as a way to reduce variance in the model similar to lasso regulatisation, since in some situations, stopping rules do not work well. An alternative way to build a decision tree model is to develop a large tree first and then prune it to optimal size by removing nodes that provide less additional information (Hastie et al., 2009). A conventional approach to choosing the best potential sub-tree from numerous possibilities is considering the fraction of records with the incorrect prediction (i.e., the proportion of inaccuracy in the target’s expected occurrence). A validation dataset or cross-validation are two other ways of finding the best choice. Pre-pruning or forward pruning and post-pruning or backward pruning are the two pruning methods. Pre-pruning uses Chi-square tests or multiple-comparison adjustment methods to avoid the formation of insignificant branches. After creating a complete decision tree, post-pruning is used to delete branches to enhance the overall classification accuracy when applied to the validation dataset (Song & Lu, 2015).

There are several statistical algorithms widely used in the industry for building decision trees, including CART (Classification and Regression Trees) by Breiman et al., (1984), C4.5 algorithm by Quinlan, (1994), Chi-Squared Automatic Interaction Detection (CHAID) by Kass, (1980), and Quick, Unbiased, Efficient, Statistical Tree (QUEST) by Loh & Shih, (1997). Table 1 briefly compares the four most widely used decision tree methods (Bhukya & Ramachandram, 2010; Song & Lu, 2015). The most commonly used decision tree algorithm in the industry is the CART. Since the scope of this dissertation is on credit scoring, which is a classification problem, we will focus on explaining classification trees.

A classification tree is akin to a regression tree, with the exception that it is used to predict qualitative rather than quantitative responses. In the case of a regression tree, the average response of the training observations associated with the same terminal node determines an observation’s projected response. In the case of a classification tree, however, it is anticipated that each observation belongs to the training class occurring most often in the area where the observation occurs. In understanding the findings of a classification tree, the interest lies not only in the class prediction associated with a specific terminal node area but also in the class proportions of the training observations that fall into that region (James et al., 2021)

If the target is a classiﬁcation outcome assuming values 1,2,...,M, the only necessary modifications to the tree method are the criteria for splitting nodes and pruning the tree. In a node *n*, representing a region *Rn* with *Kn* observations, let the proportion of class *m* observations in node *n* be:

|  |  |
| --- | --- |
|  | (31) |

The observations in node *n* are classified to class *m*(*n*) = arg max*m* the majority class in node *n*. The following are the various measures of node impurity :

Misclassification error:

|  |  |
| --- | --- |
|  | (32) |

Gini index, a measure of total variance across the *m* classes

|  |  |
| --- | --- |
|  | (33) |

Cross-entropy or deviance

|  |  |
| --- | --- |
|  | (34) |

For two classes, let *p* be the proportion in the second class. Then:

|  |  |
| --- | --- |
|  | (35) |

All three measures are comparable, but the cross-entropy and Gini index are differentiable and more suitable for numerical optimisation. Moreover, the Gini index and cross-entropy are more susceptible to changes in node probabilities than the misclassification rate. For this reason, while expanding the tree, either the Gini index or cross-entropy should be used. Any three indicators may guide cost-complexity pruning, but the misclassification rate is the most common. There are two fascinating ways to analyse the Gini index. Instead of classifying observations with the majority class in the node, they may be classified to class *m* with probability . The expected training error rate in the node will then be or the Gini index. Similarly, if each observation is coded as 1 for class *m* and zero otherwise, the variance over the node of this 0-1 response is . Summation over classes *m* results in the Gini index (Hastie et al., 2009) again.

The hyperparameters (HPs) make machine learning algorithms highly configurable. However, they often significantly impact the learner’s complexity, behaviour, and speed. So their values must be carefully chosen to obtain optimum performance (Bischl et al., 2021). The CART algorithm is mainly influenced by the following:

* splitting criteria, such as the Gini index, are used to determine the optimal feature and split point for each internal node
* stopping criteria for splitting, which regularise the tree by determining its size
* procedure for assigning a projected value to every leaf node (Bischl et al., 2021)

Some other hyperparameters of the CART algorithm include the following which are also embedded in most programming languages:

* complexity parameter, which prevents splits that decrease the overall loss by a fraction less than this
* minimum number of observations in a node for a split to be attempted
* minimum number of observations in a leaf node
* maximum depth of any node of the final tree
* number of surrogate splits retained in the output
* means of using surrogates in the splitting process
* number of cross-validations (applicable when training using this method)

Most of these hyperparameters aim at avoiding overfitting which is one of the most important drawbacks on tree-based algorithms. Overfitting occurs when the induction algorithm develops a classifier that precisely fits the training data but has lost the capacity to generalise to examples not given during training. In other words, the classifier just memorises the training cases rather than learning. Overfitting in decision trees happens when the tree has too many nodes compared to the quantity of training data provided. The training error often lowers as the number of nodes increases, but the generalisation error worsens at some point (Rokach & Maimon, 2015). Overfitting has been discovered to reduce prediction accuracy in decision trees in the face of significant noise or when the input qualities are unrelated to the classification task (Schaffer, 1991). Therefore, it is crucial to assess when more training will not improve generalisation to minimise overfitting. There are two methods in decision trees that assist in minimising overfitting. The first is to prevent dividing the tree if the split is ineffective, for example, by only approving statistically. The second approach is to prune the unnecessary nodes after growing the tree (Rokach & Maimon, 2015).

Decision trees can be ensembled using the Bootstrap and Bagging Aggregation techniques. Bootstrap aggregation or bagging, and boosting are general-purpose ensemble methodologies used to lower the variance of a statistical learning approach. They are beneficial and are widely employed in decision trees (James et al., 2013). The fundamental concept of an ensemble technique is to combine several models, each of which solves the same original problem, to generate a superior composite global model with more accurate and trustworthy estimates or choices than can be achieved with a single model (Rokach & Maimon, 2015).

Using the bootstrap, or sampling with replacement, bagging produces numerous copies of a training set. First, these datasets are used to train distinct models. Then, by averaging (in the case of regression) or voting (in the case of classification), the outputs of the models are merged to form a single result (Kantardzic, 2019). The fundamental concept behind boosting approaches is to employ a weak learning algorithm to construct a strong learner or an accurate PAC (probably approximately correct)-learning algorithm. Furthermore, boosting strategies employ an ensemble method: They integrate many basic classifiers supplied by a weak learner to produce a more precise prediction.

Bagging aggregation entails producing several copies of the original training data set using the bootstrap, fitting a distinct decision tree to each copy, and integrating all trees to produce a single prediction model. Notably, each tree is independently constructed using a bootstrap data set. Boosting operates similarly, except that trees are created progressively, with each tree using information from previously grown trees. In addition, bootstrap sampling is not used in boosting; instead, each tree is fitted on a modified version of the original data set (James et al., 2013).

Bagging generates replicate training sets by sampling with replacements from the training instances. Boosting utilises all instances at each repetition but keeps a weight for each instance in the training set representing its relevance; altering the weights enables the learner to concentrate on various instances, resulting in different classifiers. In each scenario, multiple classifiers are then combined by voting to produce a composite classifier. In the case of bagging, each component classifier receives the same vote, while boosting assigns component classifiers varied voting strengths based on their accuracy (Quinlan, 2006).

### **2.2.2 Random Forest**

Random Forest, proposed by Breiman (2001), is a collection of Decision Trees derived from a random sample of data. The term “Random Forest” is derived from the random selection of a subset of data and the presence of several decision trees, therefore, a forest (Hartshorn, 2016). Random forest methodology is a considerable modification of bagging. A considerable number of de-correlated trees are constructed and then averaged. Random forests perform comparably to boosting, and they are easier to train and modify (Hastie et al., 2009). A random forest is an ensemble of classifiers that use many unpruned individual decision trees. It employs a modified tree learning algorithm that picks a random subset of the features at each candidate split in the learning process. This method is also referred to as “feature bagging”. This is done because of the correlation between the trees in a typical bootstrap sample. Suppose one or a few features are accurate predictors of the output class. In that case, these features will be picked from several trees, leading them to become linked. The random forest approach introduces additional randomisation into the model during tree growth. Instead of looking for the optimal feature when splitting a node, this algorithm seeks the optimal feature among a random group of features. This procedure generates a great deal of variety, resulting in a superior model. (Kantardzic, 2019).

Hastie et al. (2009) outline the random forest algorithm for the construction of regression and classification trees as follows:

(1) For *b* = 1 to *B*:

(a) A bootstrap sample **Z\*** of size *N* is drawn from the training data.

(b) A random-forest tree *Tb* is then grown to the bootstrapped data, by recursively repeating the following procedures for each of the tree’s terminal nodes before attaining the minimum node size *nmin* :

(i) Randomly selecting *m* variables from the *q* input variables.

(ii) Picking the best variable or split-point among the *m* variables.

(iii) Splitting the node into two child nodes.

The default value for *m* and the minimum node size have been proposed as:

(q/3) and five respectively for regression trees

(√q) and one respectively for classification trees

(2) The output of the ensemble of trees is given by

(3) The random forest predictor at a new point *x* is given for:

(a) Regression trees:

(b) Classification trees:

Where is the class prediction of the *b*th random forest tree. An average of B independent and identically distributed random variables, each with variance , has variance . If the variables are simply identically distributed, but not necessarily independent with positive pairwise correlation , the variance of the average is given by:

As B rises, the second component vanishes. However, the first term remains, limiting the benefits of averaging to the magnitude of the correlation of pairs of bagged trees. The random forests approach designed to enhance bagging variance reduction by minimising correlation across trees without excessively raising variance. This is accomplished throughout the tree-growing process by randomly selecting the input variables (Hastie et al., 2009).

An essential feature of random forests is its use of out-of-bag (OOB) samples. For each observation its random forest predictor is constructed by averaging only those trees that correspond to bootstrap samples where did not occur. The OBB error estimates are almost equivalent to N-fold error estimates. As a result, unlike many other non-linear estimators, random forests may be fit in a single sequence, including cross-validation along the way. The training may be stopped after the OBB error has stabilised (Hastie et al., 2009).

Like individual classification trees, random forests enable one to evaluate the importance of variables in model construction. In general, Mean Decrease Gini is often used to determine the importance of variables (Louppe et al., 2013) This statistic displays the average decline in the Gini index throughout the set due to the application of a given variable. The most critical variables in a set contribute the most to decreasing the Gini index. Shapley values (Molnar, 2022) and Variable Importance Measure (VIM) computations (Breiman, 2001) are two other methods for evaluating the significance of variables in a random forest (Kaszyński et al., 2020).

Random forests are likely to perform poorly with small *m* when the number of variables is large. However, the proportion of relevant variables is small. There is a slight possibility that the relevant variables will be chosen at each split. The performance of random forests is remarkably resistant to an increase in the number of noise factors as the number of important variables grows. It is often said in the literature that random forests “cannot overfit” the data. Although, it is possible for Random Forest to overfit in case of having wrong tuning of hyperparameters.

### **2.2.3 Extreme Gradient Boosting**

Extreme Gradient Boosting (XGBoost) is a scalable end-to-end tree boosting system initially developed by Tianqi Chen based on the gradient boosting framework proposed by Friedman (2001) and extensively described in the literature. Data scientists commonly utilise it to produce cutting-edge outcomes on various machine learning challenges. It is available as an open-source package. The most crucial component in XGBoost’s success lies in its scalability in all settings. The system operates more than ten times quicker than current popular solutions. Furthermore, it is scalable to billions of samples in distributed or memory-limited scenarios on a single computer. XGBoost’s scalability is due to numerous major system and algorithmic enhancements. Among these developments are a unique tree learning technique for dealing with sparse data; and a theoretically justified weighted quantile sketch process for dealing with instance weights in approximation tree learning. In addition, parallel and distributed computing speeds up learning, allowing for faster model exploration. More crucially, XGBoost uses out-of-core processing, allowing data scientists to analyse hundreds of millions of instances on a single desktop. Finally, combining these strategies to create an end-to-end system that expands to even more extensive data sets while using the fewest cluster resources is fascinating (T. Chen & Guestrin, 2016).

The XGBoost algorithm is explained below, following Chen & Guestrin (2016) and Chen et al., (2019).

Let a data set be composed of *n* samples with *k* features. Also, let be defined as the predicted value given by:

|  |  |
| --- | --- |
|  | (36) |

Where denoting an independent regression tree is the prediction score given by the *m* -th tree to the *i*-th sample.

The set of functions in the regression tree model can be learned by minimisation of the objective function:

|  |  |
| --- | --- |
|  | (37) |

Where denotes the training loss function (a differentiable convex function) measuring the difference between the prediction and the target . The term penalises the complexity of the model and helps to smooth the ﬁnal learnt weights to avoid over-ﬁtting:

|  |  |
| --- | --- |
|  | (38) |

Where and represent the degrees of regularisation, and denote the numbers of leaves and the scores on each leaf, respectively.

The tree ensemble model in equation (37) is trained in an additive manner. It includes functions as parameters and so cannot be optimised utilising traditional optimisation methods in the Euclidean space. Letting be the prediction of the *i*-th instance at the *t*-the iteration, needs to be added greedily to minimise the following objective:

|  |  |
| --- | --- |
|  | (39) |

Using second-order Taylor expansion and removing the constant terms, equation (39) may be simplified as:

|  |  |
| --- | --- |
|  | (40) |

Where and are the first and second-order gradient statistics on the loss function. Defining the instance set of leaf *j* as , equation (40) may be rewritten by expanding as:

|  |  |
| --- | --- |
|  | (41) |

Given a fixed tree structure , the optimal weight of leaf *j* and the corresponding optimal value of the objective function may be computed as:

|  |  |
| --- | --- |
|  | (42) |

Where , .

Regarding , from equation (42), can be used as a scoring function to measure the quality of a tree structure *q*. This score is similar to the impurity score for analysing decision trees, except it is calculated for a wider variety of objective functions. Usually, it is challenging to list all potential tree structures *q*. Instead, a greedy method is utilised, which begins with a single leaf and iteratively adds branches to the tree. Let and be the instance sets of the left and right nodes after the split and . Then the loss reduction or the gain after the split may be expressed as:

|  |  |
| --- | --- |
|  | (43) |

The XGBoost model generates many simple trees used to score a leaf node during splitting. The first, second, and third terms of equation (43) represent the score on the left, right, and original leaf, respectively. Moreover, the term representing the regularisation on the extra leaf will be used throughout the training process (Chen et al., 2019).

## **2.3 Problem of multicollinearity**

Prior to digging into the topic of multicollinearity, first, we introduce how the variance of a dependent variable is determined by a set of independent variables in a graphical way using Venn diagrams introduced by (Venn, 1880). The earliest example found for explaining statistical concepts, and regression variance using Venn diagrams is by (Cohen et al., 1975), and this method of visualisation has gained popularity over the past years, especially in the academic field due to the adoption of this visualisation method by Agresti & Finlay (1997) and Ip (2017).

For this example, we generate correlated data using the Cholesky decomposition on a desired correlation matrix. Let assume that is a non-singular symmetric correlation matrix; then it can be expressed as the product of the matrix and as follows:

From the above matrix definition, is a lower triangular matrix and is the transpose of . When multiplying by its transpose, produces the original symmetric matrix .

The generated dataset will consist of 3 variables: ; with mean 10, 9, 9 respectively and 10000 observations. These variables correlate each other with the following correlation matrix:

From the above 3x3 correlation matrix, we make and be correlated at 0.8, and at 0.5, and and correlated at 0.3. The idea is to draw samples from a standard normal distribution of uncorrelated samples creating a 3x10000 matrix with random normally distributed data. Then we multiply the lower triangular matrix of our correlation matrix resulting from the Cholesky decomposition by the generated matrix , then we add the vector of the desired mean; to the resulting matrix we calculate the transpose and get our 3x10000 correlated dataset as follows:

Generally, in regression diagrams, the size of the Venn diagrams of each variable is sized according to its variance. However, Ip (2017) suggests using the sum of squares as a sizing factor. Thus, we calculate the sum of squares of and to calculate de size of the Venn diagrams as follows:

A picture containing chart

Description automatically generated

Figure 10. Size of variation of and using the sum of squares

Source: own work

From the above graph, we can observe that and have similar variation, although it is worth mentioning that has slightly more variation than . These variables are related to each other and have some covariance. This shared covariance can be calculated using analysis of variance (ANOVA) with the *anova\_lm* function from the *statsmodels* package in python.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Sum of squares | df | F | PR(sF) |
| X1 | 6374.42 | 1 | 17762.94 | 0 |
| Residual | 3587.88 | 9998 | NaN | NaN |

Table 5. ANOVA results on generated and

Source: own work

The above table indicates that 6374.42 sum of squares units are shared between the two variables, with 3587.88 not shared (or residual). Graphically speaking this relationship is represented as follows:

Diagram

Description automatically generated with medium confidence

Figure 11. Shared covariance between generated and

Source: own work

The shared area C in red represents the amount of variation in explained by the variable while A is the unexplained area or, in other words, the error term in regression analysis, the above model (relationship) can be mathematically expressed as follows:

|  |  |
| --- | --- |
|  | (44) |

From figure 11, we can also observe that area C also represents the coefficient of determination which is a goodness of fit measure that indicates the portion of the variance in the dependent variable explained by the independent variable. In other words, it is the shared area C over the size of the circle Y:

|  |  |
| --- | --- |
|  | (45) |

Based on figure 11, we can infer that explains 64% of the variation in . Alternatively, we can say that graphically, the area C is 64% of the size of the circle . Consequently, we can represent the relationship between and in the same fashion.

In multiple regression analysis, the same principle applies. In the following Venn diagram, we plot the association of , and being the sizes of the circles, respectively, and the overlapping areas are the shared variance. Here, the combination of G and D (or ) is the covariance previously calculated in table 6, and F is the explained variance in explained by .

Diagram, venn diagram

Description automatically generated

Figure 12. Shared covariance between generated , and

Source: own work

Based on the above graph, we can calculate the coefficient of determination as the shared red area between A, B and C.

|  |  |
| --- | --- |
|  | (46) |

This also helps to understand the intuition of increasing when adding extra variables to a model. Intuitively, this can be explained due to the fact of adding an extra circle to the diagram that absorbs more of the variation of the dependent variable. To better understand the meaning of each of the overlapping areas (segments), we can refer to the following table:

|  |  |
| --- | --- |
| **Segment** | **Meaning** |
| A + D + F + G | Total variation in dependent variable |
| B + D + E + G | Total variation in the first regressor |
| C + E + F + G | Total variation in the second regressor |
| A | Unexplained variation in after accounting for and |
| B | Unexplained variation in after accounting for and |
| C | Unexplained variation in after accounting for and X1 |
| D + G | Variance shared by and |
| F + G | Variance shared by and |
| E + G | Variance shared by and |
| D | Variance only between and , without influence from |
| F | Variance only between and , without influence from |
| E | Variance only between and , without influence from |
| G | Variance shared by , and |

Table 6. Meaning of segments in a Venn diagram for multiple regression

Source: Compiled from Heiss (2021)

Using the previous diagram, we can observe the effect of multicollinearity in the model. Multicollinearity is a common issue that occurs due to the correlation between the independent variables in a multiple regression model. When 2 variables correlate with each other, they automatically share a common variance. The multicollinearity phenomenon refers to the degree of shared variance between independent variables, meaning that two or more regressors can predict each other with some accuracy.

Diagram

Description automatically generated with low confidence

Figure 13. Effects of multicollinearity on multiple regression

Source: own work

In the above Venn diagram, the segments D and F are the unique areas from and respectively that explain the variation of the dependent variable . However, the segment G is overlapped by and making it difficult for the model to determine which variable explains that part of the variation of . Likewise, segment E represents the shared variation between and , consequently, it is hard to know which is the unique segment. In that sense, G + E is considered the total multicollinearity in our model.

In multivariate linear regression, in order to ensure unbiased estimates using the ordinary least squared method (OLS), some assumptions must be fulfilled, more specifically, randomly sampled observations, homoscedasticity of the error term, linearity of parameters, zero conditional mean, normally distributed error terms, and lastly, no multicollinearity (Puntanen & Styan, 1989; Rao, 1976). Analytically speaking, when perfect multicollinearity is present, in the case of the OLS method, the coefficients of the regression are estimated by solving the equation in which is irreversible due to the linear combination of correlated variables (Kaszyński et al., 2021). In the context of credit scoring, as explained in section 1.7, logistic regression also assumes the absence of multicollinearity.

The problem with near multicollinearity is that it leads to strange coefficient estimates, and the variance of parameters estimates in the regression is inflated (Carroll et al., 1997). However, the predictive power of the model is not affected (Heiss, 2021; Kaszyński et al., 2021). In credit scoring, it is essential to have reliable estimates so they can be interpreted appropriately and comply with regulations and business requirements for developing accurate scorecards. If the estimates' variance is inflated, then these parameters are not reliable and hence, not interpretable.

In the monograph presented by Kaszyński et al. (2021) is indicated the two main factors causing multicollinearity:

1. **The input data**: Usage of highly linearly correlated independent variables in the model additionally, poorly designed data collection method, few samples, the inclusion of many independent variables, etc.
2. **Incorrect model specifications**: one-hot encoding including as many binary variables as categories, the addition of independent variables resulting from the calculation on other independent variables

In both linear and logistic regression, the estimates' variance is calculated straight from the diagonal of the variance-covariance matrix of the estimates. In the case of logistic regression, the variance-covariance matrix is given by the inverse of the Fisher information matrix. The Fisher information matrix consists of the second partial derivatives of the log-likelihood function (equation 19) towards each parameter (Widyaningsih et al., 2017).

Also, for linear regression using the OLS method, O’Brien (2007) presents a formula that helps to calculate the variance of the estimates and, therefore, assess the effect of multicollinearity on the variance of estimators.

|  |  |
| --- | --- |
|  | (47) |

From the above formula, is the number of observations, stands for the number of parameters, represents the coefficient of determination of the model (explained before), is the coefficient of determination when regressing the independent variable in question with respect to the other explanatory variables (a measure of multicollinearity), is the sum of squared deviations of the regressor in question, and stands for the sum of squared deviations of the dependent variable.

From equation 47*,* we can observe that the variance of estimates is decreased when increasing the number of observations , decreasing the number of parameters , including relevant variables that increase the value of , not including variables that relate to each other which increase the value of and causes multicollinearity. Lastly, reducing the sum of squared deviations from the dependent and explanatory variable in question.

The main problem of high variance of estimates is that these may exhibit changes in the sign of the parameter, from negative to positive, and the statistical significance also is affected. Before the modelling phase, it is important to check and assess the correlation matrix of explanatory variables and consider dropping or using ridge regression as indicated in section 2.1.3.1. Also, during the modelling phase is advisable to calculate the Condition Index (CI) and the Variance Inflation Factor (VIF) as a measure of multicollinearity defined as:

|  |  |
| --- | --- |
|  | (48) |

Where is the coefficient of determination of the variable in question with respect to the rest of the explanatory variables. Although the literature is not conclusive about a standard threshold level of VIF that indicates multicollinearity, O’Brien (2007) indicates that the most common threshold values are 4, 5 and 10.

Using the same method explained at the beginning of this section for generating correlated data with the Cholesky decomposition, we replicate the same experiment as Kaszyński et al. (2021) for multicollinearity effect on parameter estimates. On a normally distributed dataset, with three explanatory variables, and 100 observations, we create an OLS regression model for different levels of correlation between two explanatory variables.

A picture containing chart

Description automatically generated

Figure 14. Effects of multicollinearity on parameters estimates

Source: own work

From the above graph, we can observe that different levels of correlation affect the coefficient estimates in the regression, leading to a change in the sign of the coefficient value of Estimator 2 from positive to negative. Also, the confident intervals of the parameters get wider when augmenting the level of correlation between the features, showing a sign of increasing variance due to the presence of multicollinearity. As mentioned earlier, there are 2 methods for calculating the variance of a parameter estimate using OLS, the first approach is to use the diagonal of the variance-covariance matrix, and the second approach is the formula from equation 34 by O’Brien (2007).

Graphical user interface

Description automatically generated with low confidence

Figure 15. Effects of multicollinearity on parameters variance

Source: own work

Our results are consistent and similar to those obtained by Kaszyński et al. (2021). We can observe a inflation on the variance of the parameters when increasing the level of correlation between the variables when using both methods. Moreover, we obtain the standard errors which is calculated as the squared root of diagonal of the variance-covariance matrix. We calculate the standard errors to better visualise the impact of multicollinearity on the parameters average distance of the observed values from the regression line (variance).

A picture containing shape

Description automatically generated

Figure 16. Effects of multicollinearity on the standard errors

Source: own work

The standard error is the standard deviation of the coefficient estimates, and it tells us how the coefficients vary across cases, in other words, it is a measure of precision of the estimate. The standard errors are expected to increase in the case of high variance and both have the same trend. This result is consistent with the literature and confirms the effect of multicollinearity of parameters variance. Apart from inconsistent parameter values and inflated variance, multicollinearity also affects the statistical significance of the estimated parameters. In regression analysis, the t-statistic is used to determine the p-values of the estimates and it is obtained by dividing the coefficient value by its standard error.

Graphical user interface, application

Description automatically generated

Figure 17. Effects of multicollinearity on the standard errors

Source: own work

In order to determine the statistical significance of a parameter, we must get the p-value associated. The p-value helps test the null hypothesis that our parameter estimates are linearly related to the dependent variable and, therefore, extrapolate this insight to a population level. From the above graphs, we observe that for a 99% confidence level when increasing the level of correlation, also decreases the significance of the estimates. We can also appreciate the increasing level of multicollinearity measured by the VIF in the following graph:

Graphical user interface

Description automatically generated with medium confidence

Figure 18. Effects of multicollinearity on the standard errors

Source: own work

## **2.4 Omitted variables problem**

In section 2.3, it was presented that in many cases is not possible for an independent variable to explain the whole variance of the output variable in a model. Generally, the part that is not explained is captured by the model's error term or the area A from figure 11. In regression analysis, in order to ensure unbiased estimates, we assume that the error term is not correlated with the regressors of the model; nonetheless, sometimes, our model specification excludes a critical variable that also determines the variance of the output and at the same time correlates to another regressor. In this case, our error term confounds this omitted variable and automatically correlates to the variable/s in our regression; therefore, this assumption is violated (Hanck et al., 2021).

When omitting a relevant variable for explaining the variance of the output, it induces an estimation bias to the rest of the regressors. In the ordinary least squared method, this bias affects in a way that the sampling distribution of the OLS estimator does not equal the true mean. In other words, omitting a relevant variable prevents the regressors from converging to the true parameter value. On this matter, the strength and direction of the bias are determined by the correlation level from the following equation between the regressor and the error term. It means that depending on the degree and sign of correlation between the error term and the regressors; the regression parameters can get over or under-estimated (biased).

|  |  |
| --- | --- |
|  | (49) |

Also, Hanck et al. (2021) defined the conditions that must be fulfilled in order to omitted variable bias to occur when estimating coefficients using the OLS method

1. The omitted variable correlates with the regressor/s
2. The omitted variable is a determinant of the variance of the dependent variable.

The omitted variables bias is, in nature, a problem generated due to a poor model specification and design. Nevertheless, at the same time, not all models are perfectly designed, and to some extent is hard to know a priori all variables influencing the output making the omitted variable bias an inevitable common problem in all econometric models. Thus, the standard remedy to omitted variable bias is finding instrument variables (proxies) for omitted variables; however, this approach makes assumptions rarely met in practice (Leightner & Inoue, 2012). When adding instrument variables to the model, we can test if the coefficient of the original regression change; if so, these instrument (proxy) variables are omitted variables. It is plausible to consider when modelling any theoretical foundation to select instrument variables and to posit a correct and credible interpretation of results. Additionally, depending on the nature of the study is advisable to use cross-tabulation, run a randomised controlled experiment, or include the omitted variable back to the model in case it was deliberately excluded (Skjelbred, 2015).

The logistic regression is also likely to suffer the same problem of omitted variable bias. Similarly to linear regression, the effect of the omitted variable is captured by the regressor to which it is correlated and biases the result. On the other hand, in logistic regression, the residual variance increases when a relevant variable is omitted. In her research paper, Mood (2010) states that the main effect of omitted variables in logistic regression is that the logged-odds and odds ratios are affected in 2 ways:

1. The odds ratios are biased (downwards or upwards) due to a factor determined by the correlation between the regressor/s and the omitted variable and the correlation with the dependent variable.
2. Downwards biased by a factor determined when calculating the difference in the residual variance between a model omitting the variable and a model including it.

In this regard, the log odds and odds ratios cannot be interpreted since they reflect the level of unobserved heterogeneity in the model. Moreover, the logged-odds and odds ratio cannot be straightforwardly compared for similar models across groups, samples or time.

Contrary to linear regression, the main difference in logistic regression is that the omitted variable does not need to be correlated with the rest of the independent variables in the regression, and the direction of such bias depends on the geometric properties of the sigmoid function, which leads the bias towards zero (Neuhaus & Jewell, 1993). The literature points that we can test omitted variables in logistic regression by conducting goodness of fit tests such as the wald test, likelihood ratio test or score test (Fox, 1997).

In the context of fair lending in credit scoring, regulators consider some customer data “protected” variables, limiting their inclusion in a credit scoring model. In the United States of America, according to the Equal Credit Opportunity Act (ECOA), regulation B, the variables race, ethnicity, religion, and in some cases, age and marital status are considered as protected (Consumer Financial Protection Bureau, 2022), these variables, were found in the literature relevant for explaining the probability of default. We may find in some cases the inclusion of proxy variables to account for protected variables; however, Gillis (2021) argues about this and highlights the orthogonalisation of inputs, excluding the proxies variables. This method attempts to transform the inputs and, therefore, reduce or eliminate the bias.

Orthogonalisation of inputs is very useful when an input variable correlates with a protected variable. This input variable plays two roles. First, it absorbs part of the explanation of the output variable. Second, it may be used as a proxy variable for a protected characteristic (Dobbie et al., 2018). Let us suppose that we attempt to predict the variable “default” , and our model includes a protected variable, “race” , which is correlated with , and a dependent variable that does not correlate with any other variable .

|  |  |
| --- | --- |
|  | (50) |

This statistical method separates the training and testing phases of a model. During the training phase, the model is trained with a protected variable, say race , producing a parameter estimate (weight) for that variable in the prediction for default. However, during the testing phase, race is substituted by the mean race making the algorithm unaware of the race groups of the applicants, and it automatically subtracts the effect of “race” on the prediction. This approach satisfies the requirement of not discriminating on the basis of race and allows the exclusion of proxy variables. This method is handy in classical modelling (see section 2.1); however, it is not much reliable in the context of machine learning (Gillis, 2021).

# **3. Important remarks in credit scoring modelling**

Financial stability is a topic of great concern worldwide. In consequence, governments are afraid of the consequences of a potential financial crisis to the economy, and this is primarily due to the economic cost of a financial crisis. Mathematically speaking, the economic cost of a financial crisis can be quantified as the multiplication of the probability of the event of a financial crisis by the losses incurred in case this event realises. Currently, the financial sector in the European Union represents 36% of the gross domestic product (GDP). This thriving sector employs more than 45.000 people and is a capital-intensive sector that generates excellent benefits for society. Therefore, there is much at stake when it comes to the financial sector's stability.

Regulations in the financial sector fundamentally are laws that aim to reduce the probability of the event of a financial crisis by keeping the financial sector stable and providing a sound monetary environment for the performance of financial institutions. Regulations also ensure fair competition, protect customers, and prevent financial crimes (Kumar, 2014). By setting the right lending goals, regulators can increase the barrier to access to credit and reduce the potential losses in case of default of many customers, decreasing the potential social cost of a financial crisis. It is worth mentioning that the financial system plays a key role in the economy since it facilitates the flow of money between savers and borrowers, provides numerous payment mechanisms, and motivates the formation of capital in the economy by lending money that can be invested and, therefore, enhances growth in the economy.

For example, in figure 19, it can be observed that financial corporations positively impact net savings in the EU and provide the infrastructure for this component to exist. On average, from 2009 to 2020, the share of net savings in the EU by financial corporations was 28.2%.

Figure 19. Contributions of sectors to the net financial wealth of the euro area

Source: Eurostat, 2022

The financial crisis in 2007 is attributed to the collapse of the financial system that originated from the lack of financial regulation and supervision (Jickling, 2009). During the last financial crisis, banks did not have the solvency to pay all their debts in the future nor the liquidity to honour the deposit of the savers. Politicians and financial regulators learnt from that situation and started to reinforce and improve the regulations to avoid bank insolvency. Simply speaking, banks are insolvent when they cannot pay their due debts, and it generally happens when their liabilities are higher than their assets; in other words, banks owe more than what is possessed. On the other hand, a bank may become insolvent if there is a lack of liquidity to honour the upcoming debts.

As mentioned before, the financial sector is highly regulated due to its importance in the society and economy. It is widely known that most of the money in the economy is created by banks, and this process starts when savers put their money in the banks, the bank keeps a required fraction as a deposit and lends the rest of the money. Thus, this is a complex system based on trust that is repeated until the initial amount deposit extinguishes or gets repaid. At the Euro Zone level, the European Central Bank requires credit institutions to hold deposits on accounts with their national central bank, and these deposits are called minimum required reserves (MRR). The legal framework for the minimum reserve system is set out in the regulation (EC) No. 1745/2003 of the European Central Bank of 12 September 2003 on the application of minimum reserves (European Central Bank, 2003).

Bank insolvency may occur due to loan defaults, negative expectations leading to panic, and downturn situations. In the context of credit risk, a desirable situation is when the bank’s assets are higher than the liabilities, and the bank can honour its short, middle and long term obligations. The following simplified table shows the case of a bank in a financially healthy position according to its balance sheet.

|  |  |
| --- | --- |
| **Balance sheet of a bank** | |
| **Assets** | **Liabilities** |
| |  | | --- | | Liquid assets: cash, bonds, reserves in the central bank, etc. | | Non-liquid assets: Loans made by the bank to its customers (account receivables) | | |  | | --- | | Equity: Money given by shareholders of the bank (investors) | | Liabilities: Deposits of savers | |

Table 7. Bank’s healthy balance sheet - Example

Source: Own work.

The bank's equity is the difference between the assets and liabilities, as observed in table 8. Supposing that some of the loans given to customers start to default, these losses can be absorbed by the shareholder’s equity, meaning that depositors are not affected. In the case that more customers keep defaulting their loans, then the bank can classify the loans as “performing” and “non-performing” as follows:

|  |  |
| --- | --- |
| **Balance sheet of a bank** | |
| **Assets** | **Liabilities** |
| |  | | --- | | Liquid assets: cash, bonds, reserves in the central bank, etc. | | Non-liquid assets: Performing loans | | Non-liquid assets: Non-performing loans | | |  | | --- | | Equity: Money given by shareholders of the bank (investors) | | Liabilities: Deposits of savers | |

Table 8. Bank’s balance sheet with defaulted loans- Example

Source: Own work.

In light of this situation, the bank will start a recovery process that may take months or, in some cases, it may take years. This debt collection process may end in a friendly way in which the amount of the debt is recovered by arranging a repayment agreement with the borrower. Nonetheless, the recovery process may end in an unfriendly way in which legal authorities can be involved and grant the bank the legal possession of collateral (asset declared by the customer to be used as a buffer in case of default). In case of bad quality loans with a low recovery likelihood, the bank may decide to keep the recovery process open for a certain amount of years, and in some cases, the bank may sell the receivable for part of its nominal value (aka securitisation). In the worst-case scenario, the bank simply writes off the debt and deletes these receivables from the balance sheet, as shown in the following table:

|  |  |
| --- | --- |
| **Balance sheet of a bank** | |
| **Assets** | **Liabilities** |
| |  | | --- | | Liquid assets: cash, bonds, reserves in the central bank, etc. | | Non-liquid assets: Performing loans | | |  | | --- | | Liabilities: Deposits of savers | |

Table 9. Bank’s balance sheet with defaulted loans- Example

Source: Own work.

The case shown in table 10 is an example of an insolvent bank. In this case, the bank’s liabilities are larger than the assets, and therefore, the bank will not be able to repay all its depositors, even if it sells all its assets. Apart from ensuring the proper estimates for the probability of default of a particular customer, the bank also needs to calculate the amount of capital it should have on its balance sheet to absorb the potential losses that may occur in case of default of many loans in order to avoid a situation of insolvency.

From the bank's perspective, capital is the money raised from investors and profits that have not been shared among shareholders. In order words, capital is a way a bank has to raise funds. Also, a bank may raise funds via debt in the form of debt securities, loans taken by the bank, and deposits from customers. Banks may suffer significant losses if the risks associated with the lending activity get to materialise. Consequently, capital is an important element for sound banks since they absorb the losses and allow the bank to keep operating in good and bad times while ensuring customer deposits. Thus, banks must constantly check the risks they are exposed to and jointly with banking supervisors, they can assess the financial health and capital levels.

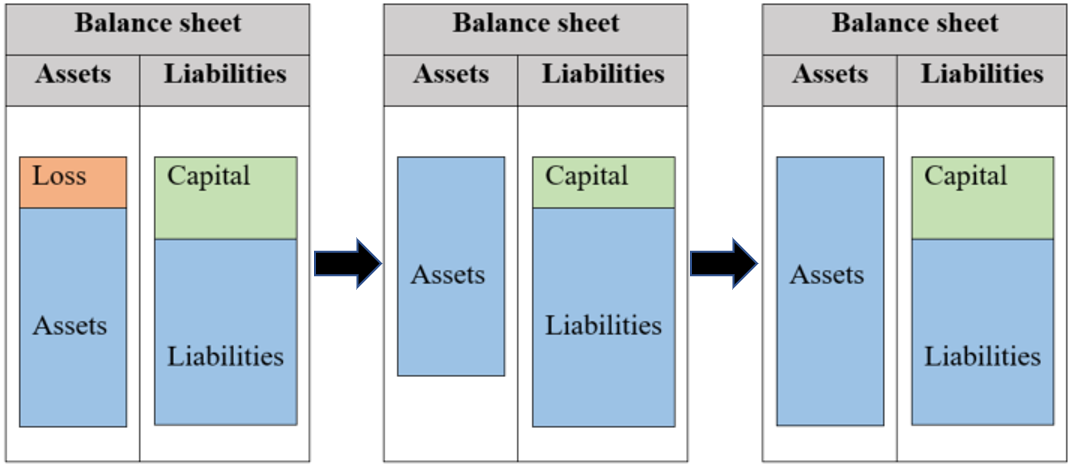


Figure 20. How capital keeps banks safe

Source: own work

As discussed earlier, capital serves as a buffer in case of losses. As depicted in the figure above, having sufficient capital, the losses can be absorbed easily, and still the bank can operate. Then the bank can have some time to rebuild its capital to prepare for possible extra future losses. The bank can increase its capital by issuing more shares and retaining profits that are supposed to be shared among shareholders.

A key element in determining the amount of capital a bank must have is the analysis of the distribution of credit losses in a portfolio over a given period (generally in a year). The loss distribution assumes that certain customers may become insolvent, and the exposure will be rarely recovered (Bellini, 2017). An example of the distribution of losses in a portfolio is presented in figure 21. The curve of the loss distribution highlights an asymmetrical behaviour of the losses, meaning that small losses have greater changes to occur than large losses. Figure 21 also points out the areas for expected and unexpected losses. In this regard, the expected loss is the average credit loss in the portfolio that a bank may expect, and it is understood as the cost of the business. Since these losses are part of the operation of the business, a bank designates a budget for that, which in accountancy terms is known as provisions. However, a drawback of the expected losses estimation is that the losses volume can be easily exceeded by fluctuations in the market and macroeconomic situation. On the other hand, we can also spot the unexpected losses that are defined as the volatility of the losses and are calculated as the standard deviation from the mean of the distribution, taking as reference a confidence level. In this respect, the unexpected losses are calculated as the difference between a high quantile and the expected loss; this quantile is generally 99.9% according to the pilar I of the Basel accord. The unexpected losses are covered by the regulatory capital mentioned in this section and further depicted in figure 20.

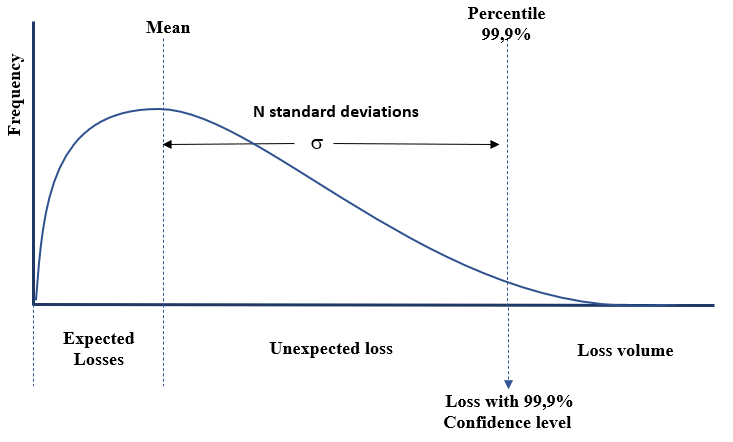


Figure 21. Loss distribution – expected and unexpected losses

Source: own work

The Expected losses are calculated by summing the product of the probability of default (PD), Exposure at Default (EAD) and the Loss Given Default (LGD), as shown in the following formula.

|  |  |
| --- | --- |
|  | (51) |

The components of the expected losses formula are described as follows:

* **The probability of default (PD):** the probability that a counterparty is unable to meet its contractual obligation at a specific timeframe.
* **Exposure at Default (EAD):** Total value a financial entity is exposed to at the time of default (volume of risk exposed at the time of default).
* **Loss Given Default (LGD):** Is the percentage of the exposure that is lost at the time of default (percentage not recovered).

For estimating the expected losses, banks use internal models following accountancy standards determined by the International Accounting Standards Board (IFRS9), allowing the calculation of provisions. For calculating the unexpected losses, banks also use internal models determined by the Internal Rating Based (IRB) approach allowing the calculation of regulatory capital. These concepts will be further explained in the next sections.

## **3.1 Regulatory environment**

Many financial authorities play a fundamental role in the regulation of credit issuance and management since they create the legal framework for a sound and stable financial sector. In Europe, the most important and relevant financial authorities are:

* **Bank for International Settlements (BIS):** The main mission of the BIS is to serve central banks in their pursuit of monetary and financial stability, foster international cooperation in those areas and act as a bank for central banks (Bank for International Settlements, 2005). The Basel Committee Charter is designated in the BIS, and the main objective is to enhance understanding of key supervisory issues and improve the quality of banking supervision worldwide.
* **European Bank Authority (EBA)**: Works as an independent EU authority which contributes, through the adoption of Regulatory Technical Standards and Guidelines, to the creation of the European Single Rulebook. This Rulebook aims at providing a single set of harmonized prudential rules for financial institutions throughout the EU (European Banking Authority, 2022).
* **European Central Bank (ECB)**: It is the single supervisory mechanism in the EU. The ECB Safeguards the value of the euro and maintains price stability while ensuring the safety and soundness of the European banking system. In addition, it performs the prudential supervision of the Single Supervisory Mechanism (SSM) members (European Central Bank, 2015).
* **Local Regulator**: The Local regulator transposes into local regulation while being responsible for the supervision of banks, the constitution of corporates, credit unions, insurers and major investment firms. An example of a local regulator: Banco de España (Spain), Komisja Nadzoru Finansowego (Poland), Bundesanstalt für Finanzdienstleistungsaufsicht (Germany), among others.

Most of the regulations impacting credit scoring are based on the Basel accord by the Basel Committee on Bank Supervision (BCBS) from the Bank for International Settlements. The Basel accord establishes international standards in relation to supervisory activity and capital adequacy so that banks have enough capital to absorb unexpected losses in cases of a business downturn worldwide. The Basel Committee on Bank Supervision (BCBS) has issued three accords named Basel I, Basel II, and Basel III, and soon it is expected to come into force the Basel IV in January 1st, 2023.

* **Basel I** accord established the basic principles on which the banking activity should be based, such as regulatory capital, permanence requirement, loss absorption capacity and bankruptcy protection. The regulatory capital must be sufficient to face credit, market and exchange rate risks. The accord also determined the minimum capital of a bank should be 8% of the total risk-weighted assets. This framework contributed to improving the solvency of the international financial system by establishing common minimum capital rules.
* **Basel II** accord consisted of three pillars: minimum capital requirements, supervisory review process, and market discipline. The Basel II further developed the calculation of risk-weighted assets (RWA) and allowed banks to apply risk ratings based on their internal models, as long as these models were previously approved by the supervisor. Basically, it promoted banks to develop internal models to allow them to have more adjustable requirements for their risk profile.
* **Basel III** accord was created as a response to the previous financial crisis in 2007. This accord takes into account the exposure of a large part of banks around the world to “toxic assets” in bank balance sheets and in derivatives circulating in the market; therefore, it adds an extra layer of common equity (aka capital conservation buffer) for banks. As for the new recommendations, we can highlight the increase of the minimum solvency ratio and quality of equity and the adoption of new capital buffers (conservation, countercyclical and systemic).

The three pillars were implemented first time in Basel II, and they aim to establish a common framework for the calculation of the amount of capital needed to cover the financial and operational risk of the lending activity.

Figure 22. Three Pillars of Basel II

Source: own work

According to the new capital accord (Basel I and II) Pilar 1 defines the mechanism for calculating the necessary capital that protects the functioning of a bank against the risks it is exposed from the lending activity. It also establishes three types of risks that affect capital requirement (credit, market, and operational risk) and the principles to estimate these risks. Since the focus of this dissertation is on credit scoring, we will focus only on credit risk. Pilar 2 on the other hand establishes the process of the supervisory review that the methods used for capital calculation will be subject to and introduces the requirement to maintain additional capital to cover all material risks to which the bank is exposed. Lastly, Pillar 3 focuses on a framework to increase market discipline and promote transparency and security in the operations of the business. This pilar obliges banks to disclose information on the risk profile and capitalisation level.

Regarding the modelling approaches for calculating the regulatory capital referred to in Pillar 1 of Basel II and III we can highlight three types of approaches.

Under the standard approach, banks use external ratings from rating agencies to determine the risk weights of the debtors. These risk weights are determined by the type of category of the borrower ranging from 0 to 150%. This is a simplified approach that targets banks desiring a simplified capital framework and with a small number of transactions (low volume of data). In this method, some credit risk mitigants (CRM) are recognised; in this sense, the minimum capital requirement is calculated as

The foundation IRB (F-IRB) approach considers five types of exposures (corporates, sovereigns, retail, banks & equity) and banks are able to estimate their own probability of default (PD) models. In contrast to the standard method, the supervisors provide the estimates for the loss given default, exposure at default, and maturity. In this approach, more credit risk mitigants (CRM) are recognised.

Lastly, the advanced IRB (A-IRB) still considered the same five types of exposure categories as in F-IRB. However, the main difference is that banks can calculate their own estimates for all credit risk parameters: the probability of default (PD), Loss Given Default (LGD), Exposure at Default (EAD) and Maturity (M). These models, however, are subject to internal thorough validation and approval from regulators; therefore, these models are more appropriate for large banks with many transactions (rich datasets).

## **3.2 Regulations for capital requirements**

Regulations in the EU are the result of general standards guidelines issued mainly by the Basel Committee on Banking Supervision (BCBS), then the prudential regulators in the EU such as the European Commission, European Parliament, and the Council of the EU may issue a regulation that is applicable to all credit entities among the EU Member States, at individual or consolidated level, as appropriate. The supervisory authorities, both at the EU and national level, ensure that financial entities are compliant with regulations. This flow is presented in the following figure.

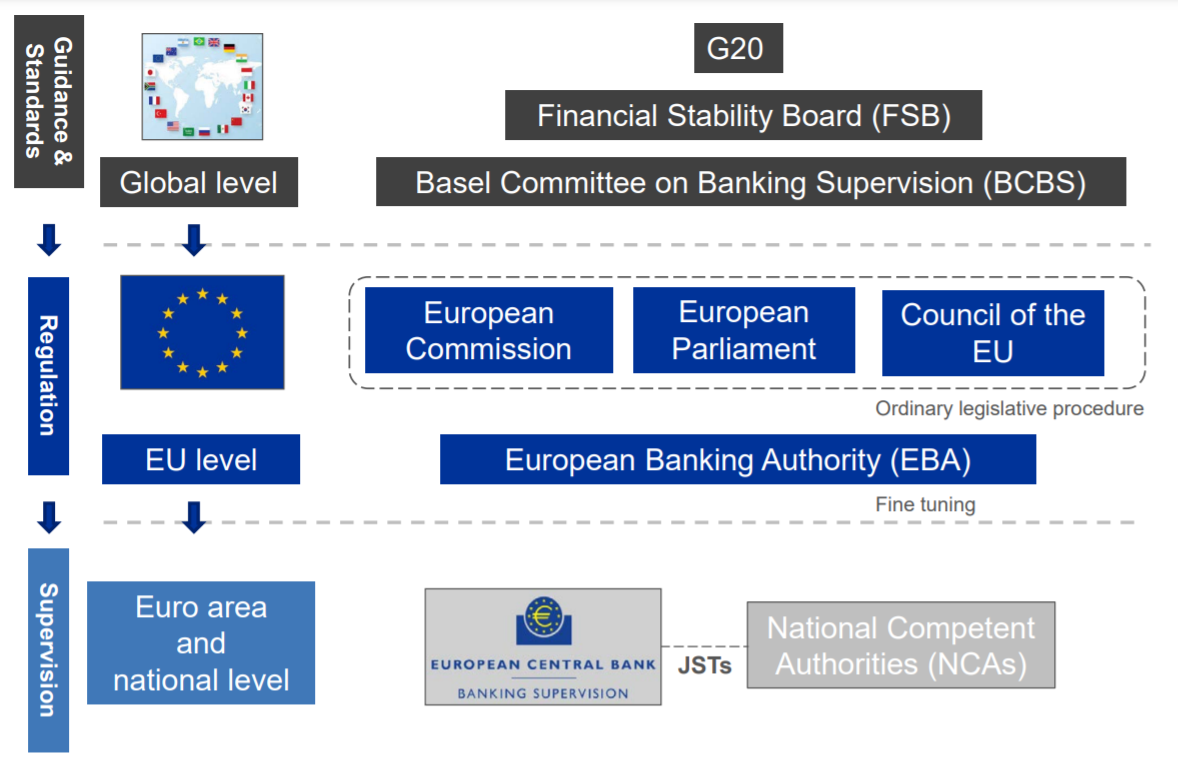


Figure 23. Regulatory Financial Bodies in the EU

Source: Lautenschläger, S (2017)

Although the banking sector is highly supervised and controlled by general and local regulators, it is important to highlight the most relevant regulations for banks when it comes to credit scoring in Europe.

First, and the most important regulation is the Capital Requirements Regulation II (CRR II), defined in the regulation (EU) No 575/2013 of the European Council of 26 June 2013 on prudential requirements for credit institutions and investment firms and amending Regulation (EU) No 648/2012 (European Parliament, 2013a). According to article 1, the CRR II applies to all credit institutions in Europe and defines, among other things, capital requirements, liquidity, large exposures, supervisory reporting, and disclosure. The CRR II sets uniform rules concerning the own funds and eligible liabilities. Allocates responsibilities to member states for the designation of the competent authority responsible for assessing the adequacy of regulatory LGD thresholds. The CRR II focuses primarily on pillars I and III of the Basel accord.

The second most relevant regulation in the EU is the Capital Requirements Directive IV (CRD IV), defined in the directive 2013/36/EU of the European Parliament and of the Council of 26 June 2013 on access to the activity of credit institutions and the prudential supervision of credit institutions and investment firms, amending Directive 2002/87/EC and repealing Directives 2006/48/EC and 2006/49/EC (European Parliament, 2013b). The CRD IV introduced a supervisory framework in the EU, which reflects the Basel II and Basel III rules on capital measurement and capital standards. Basically, the CRD IV contains the requirements for pilar II of the Basel accord regarding the supervisory review and the buffer framework (Looney, 2013).

The EBA guidelines define the third most relevant set of regulations, which are specific to the A-IRB approach. They provide clarification on the estimation of risk parameters, with a focus on PD and LGD. In addition, the EBA also define the Regulatory Technical Standards for some specific items for IRB parameters. Also, the EBA launched in 2016 the Targeted Review of Internal Models (TRIM) project, which focuses on the reduction of unwarranted variability in Risk-Weighted Assets (RWA). In this project, the National Competent Authorities (NCAs) that are competent in the EBA participated in assessing whether Basel pillar I models within the Single Supervisory Mechanism (SSM) are appropriate based on the requirements from the aforementioned regulations and determine whether the results of these models are reliable and comparable.

As a response to the previous financial crisis, the industry has experienced frequent changes to the listed regulations in figure 23.

Figure 24. Timeline of most recent regulations

Source: Compilation by the author

## **3.3 Definition of default**

Since the event of default is the target variable of interest in credit risk modelling, for the calculation of PD models, banks keep track of the behaviour of the customers, and an alert is triggered whenever there is a high risk of delinquency. For capital requirements modelling purposes, from the regulation side, the definition of default is described in article 178 of the Capital Requirement Regulation (CRR II) by the European Parliament, (2013a), which should be taken into consideration for the purpose of the IRB approach for PD model estimation:

*“a default shall be considered to have occurred with regard to a particular obligor when either or both of the following have taken place:*

1. *the institution considers that the obligor is unlikely to pay its credit obligations to the institution, […]in full, without recourse by the institution to actions such as realising security;*
2. *The obligor is past due more than 90 days on any material credit obligations to the institution […]”*

Essentially, the default flag triggers once a customer is more than 90 days past due (as indicated in figure 25), and the financial institution considers that the debtor is unable to repay back its loan. The 90 days past due (DPD) is the commonly used delinquency flag based on historical data and empirical evidence. To comply with the new definition of default, article 178 of CRR II requires banks to set and assess an absolute and relative materiality threshold which serves as an indicator that reflects the level of risk that the bank considers reasonable; consequently, whenever these thresholds are breached, the DPD count starts.

Figure 25. DPD +90 trigger example

Source: Own work

The materiality thresholds are detailed in the Draft Regulatory Technical Standards on the materiality threshold for credit obligations past due under Article 178 of Regulation (EU) No 575/2013 (European Parliament, 2013a). These RTS defines that the absolute threshold is the sum of the past due amount related to the credit obligation. Similarly, the relative threshold is defined as percentage of a credit obligation past due with respect to the total exposure amount. When these thresholds are breached for more than 90 days the debtor is considered as defaulted.

For better understanding of the definition of default, suppose that a bank sets the materiality threshold of the absolute limit to 200 EUR and for the relative limit > 1.5% of the on-balance exposure (due amount divided by exposure). Keeping this in mind, a bank decides to grant a loan of 100000 EUR to customer X and sets a monthly instalment of 1000 EUR due every first day of the month. However, this customer stops paying the monthly instalments in month 3. The following table illustrates how are the default triggers are activated.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **End of the month** | **Paid Amount** | **Past due amount** | **Exposure amount** | **Absolute threshold (Past due amount > 200 EUR)** | **Relative Threshold (Due amount / Exposure > 1.5%)** | **DPD Counter** | **Default** |
| 1 | 1000 | 0 | 99000 | No | No | 0 | No |
| 2 | 1000 | 0 | 98000 | No | No | 0 | No |
| 3 | 0 | 1000 | 98000 | Yes | No | 0 | No |
| 4 | 0 | 2000 | 98000 | Yes | Yes | 30 | No |
| 5 | 0 | 3000 | 98000 | Yes | Yes | 60 | No |
| 6 | 0 | 4000 | 98000 | Yes | Yes | 90 | Yes |
| 7 | 0 | 5000 | 98000 | Yes | Yes | 120 | Yes |
| 8 | 0 | 6000 | 98000 | Yes | Yes | 150 | Yes |
| 9 | 0 | 7000 | 98000 | Yes | Yes | 180 | Yes |
| 10 | 0 | 8000 | 98000 | Yes | Yes | 210 | Yes |

Table 10. Definition of default example

Source: Own work

As observed in the previous table, the absolute threshold trigger is activated in month 3 when the due amount exceeds 200 EUR, in month 4 the ratio between the due amount and the exposure is higher than 1.5% meaning that the relative threshold trigger is activated. When both thresholds are activated the DPD count starts and finally the customer is considered as defaulted in month 6 after 90 days past due since the activation of the triggers.

## **3.4 IRB and IFRS9 modelling approaches**

In previous sections, it was introduced the idea of capital requirements and provisions estimation, the importance of these estimates for the business, and the influence of regulators in the sector around these estimates. Although the focus of this dissertation is not on regulatory modelling, we discuss the most important modelling approaches for the estimation of expected and unexpected losses under the IRB and IFRS9 modelling framework.

When it comes to IRB models it is important to understand the concept of risk weights and risk-weighted assets. The risk weight is a function of the bank’s perception of the risk associated with the credit exposures in the portfolio which is fundamental for the calculation of risk-weighted assets. Risk-weighted assets (RWA) are calculated as follows:

|  |  |
| --- | --- |
|  | (52) |

From the above formula we can observe that risk-weighted assets are the off-balance-sheet exposures, weighted according to the risk. This sort of asset calculation is used in determining the capital requirement for a bank. In section 3.1 it was mentioned that according to the Basel accords, the amount of capital that a bank must have should be at least 8% of the total risk-weighted assets.

|  |  |
| --- | --- |
|  | (53) |

The estimation of capital is of much importance for a bank since it helps to keep the business alive, but also is a way to employ these funds on investments that make the business grow, ensuring an optimal rate of return on investment (profitability) of the bank. Under the standard IRB approach the risk weights are provided by the regulator. These weights are specified in the following articles of the CRR II from the EBA guidelines, an example of these weights is shown in the following table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of exposures** | **Article** | **Important Remarks** | **Risk Weight** |
| Exposures to central governments or central banks | Article 114 |  | 0% |
| Retail exposures | Article 123 |  | 75% |
| Exposures to corporates | Article 122 |  | 100% |
| Exposures secured by mortgages on immovable property | Article 124 |  | 100% |
| Exposures in default | Article 127 |  | 150% |
| Specific credit risk adjustments are no less than 20% | 100% |
| Residential property according to article 125 | 100 |

Table 11. Risk weight per exposure category – Standard IRB

Source: Own work

According to article 147 CRR II, depending on the class of exposure a risk weight is assigned which is then used to calculate the risk-weighted asset from formula 39. Under this approach, the data for the EAD, LGD and Maturity is fully provided by the regulator.

Contrary to the standard approach, the advanced IRB (A-IRB) allows the development of empirical models (PD, LGD, EAD) to quantify capital for credit risk based on the bank’s particular portfolio specifications. Under this approach, the risk weights are not provided and must be calculated depending on the class of exposure. The calculation of these risk weights is described in articles 153 to article 157 of CRR II (European Parliament, 2013a).

The A-IRB approach uses the through-the-cycle rating method in which debtors’ riskiness is assessed taking into consideration the worst-case scenario in the economic cycle, therefore ratings are measured over a time horizon long enough to do not account for the effects of the economic cycle, which by convention is 5 or more years (Aguais et al., 2004). The customer-specific characteristics used for the calculation of the ratings slowly vary or stay constant in time (Casellina et al., 2021).

The usage of historical data for the calculation of the estimates is an important characteristic of IRB models. This approach in particular models the probability of default in one year's time, for that, the bank checks if a customer has defaulted at least once in 12 months even if by the time of observation, the customer already paid back its obligations and has a “cured” credit status.

Additionally, A-IRB models must consider macroeconomic information, especially for the through-the-cycle calculation. Regulators expects adjustments in the parameter estimates in IRB models by removing certain constraints like downturn adjustments and floors. On top of that, the EBA guidelines on PD estimation, LGD estimation and treatment of defaulted assets, establish that from January 1st 2021 banks should add additional corrections to the parameter estimates that arise from the following deficiencies (European Bank Authority, 2017):

* Data and methodological deficiencies
* Changes to underwriting standards, risk appetite, collection and recovery policies, or any other uncertainty
* General estimation error

Such correction in the estimates is called Margin of Conservatism (MoC), although there is not a clear way to determine the margin of conservatism, some research papers propose a framework to quantify the identified uncertainties for the purpose of IRB risk parameter estimation (Liu, 2019).

Contrary to IRB approach for regulatory capital requirements, the International Accounting Standards Board (IASB) as well as other accounting standards regulators set out the standards for the recognition and provision for credit losses for financial statement related purposes. Back in 2014 it was issued the International Financial Reporting Standard 9 (IFRS 9) that came into force from January 2018 and introduced the expected credit loss (ECL) framework regarding the recognition of impairment.

As mentioned in previous sections, debt instruments (loans) granted by the bank are receivables that are registered as assets in the balance sheet. A financial asset, in this case a loan, is said to be impaired due to credit risk if an event (or many) has an adverse effect on the estimated profit or cash flows of this financial asset. An impaired loan is related to a reduction on the creditworthiness of a client which is observed with historical data when a customer has financial difficulties, or in the case there is a breach of the contract (a default event), basically any event that increases the likelihood of bankruptcy of the customer is accounted accordingly.

Under IFRS 9 there are specific standards for the recognition of impairment, for that, banks should consider past events, current conditions and make some forecasts so the expected credit losses (ECL) can be updated and reflect the changes in the risk of the assets (loans).

The IFRS 9 standard is known to have a “forward-looking” modelling approach since it demands the use of information of different macroeconomic scenarios with the probabilities of events occurring, here stress test models use forecasts of the main macroeconomic variables, with which forecast average PDs and LGDs for every scenario as shown in figure 26. Nonetheless, the estimates of PDs are based on the point-in-time approach where estimates are not consistent and may vary depending on the time of estimation that is being affected by the economic cycle, this is a short-term modelling approach (Aguais et al., 2004). The provisions calculation based on impairment models aligns the regulatory capital calculation (proposed by Basel Committee) with the expected losses model based on the distribution of bank’s losses explained in figure 21.

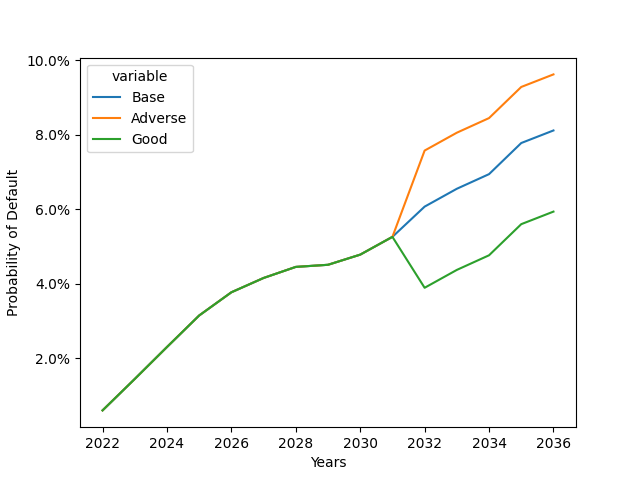


Figure 26. Forward-looking stress tests IFRS 9

Source: Own work

Under the IFRS 9 we can distinguish three “stages” with the aim to recognise credit deterioration (impairment) of financial assets at an early stage. Each stage has its own approach to calculate the ECL, and these resulting ECL should be unbiased, and probability weighted. These stages are based on changes in credit quality since the initial recognition of the loan, and are defined as follows:

Stage 1: This stage classifies loans that are newly originated or purchased, and those existing facilities whose risk (probability of default) have not significantly increased since they were recognised in the portfolio. For stage 1 credit facilities, a 12 months ECL is calculated. In this case, banks assume what the loss will be in the event of a loss that has occurred or will occur in the next year.

Stage 2: Includes credit exposures for which there was a considerable increase in the credit quality deterioration since the first time they were recognised in the portfolio. For stage 2 credit facilities, a lifetime ECL is calculated.

Stage 3: Credit exposures whose risk increased to the point in which the credit is considered as “impaired”, in this category we can identify defaulted credit facilities. A lifetime ECL is calculated as in stage 2.

There is clearly a difference between 12 month and lifetime ECL, however, there are linked. 12-month ECL is the portion of lifetime ECL associated with the possibility of default in 12 months. It shouldn’t be confused with the credit losses expected in the next 12 months, but rather the effect of a credit loss on a loan over its lifetime weighted by the probability of event in the next 12 months. For stage 2, the lifetime ECL covers the potential default events over the remaining maturity, and for stage 3, the ECL takes into account the fact that the asset is in default. For lifetime ECL calculation the time value of money concept is incorporated, meaning that ECLs must be discounted to present value using the average effective interest rate (EIR) of the portfolio. In consequence, the formula for calculate the expected losses for each credit exposure is the following:

|  |  |
| --- | --- |
|  | (54) |

Where ttm is the time to maturity, which is 1 for stage 1 credit exposures.

The following table pinpoints the main differences between IRB and IFRS 9 modelling approaches based on the topics discussed in this section:

|  |  |  |
| --- | --- | --- |
| **Aspect** | **IRB** | **IFRS 9** |
| Time Horizon | 12 months | 12 months and lifetime (depending on the stage) |
| Sensitivity to macroeconomic scenarios | Represents long-run average (Through-the-cycle, TtC) over full economic cycle | Reflects the current state of the economy (point-in-time, PiT) |
| Extra adjustments | Floors and downturn adjustments | No needed |
| Level of conservatism | MoC added to the estimates | Represents fair view (accounting approach) |
| Information | Historical | Forward looking |

Table 12. Comparison between IRB and IFRS 9

Source: Own work

# **4. Algorithmic fairness**

Fairness is becoming an essential factor in machine learning automated decisions such as credit scoring. As introduced in chapter 3, the banking industry is highly regulated, and lending decisions are expected to be fair from a regulatory point of view. Therefore, these fair decisions must comply with a quantitative parity concerning the protected attributes of individuals. In this regard, there is a considerable concern nowadays in the machine learning industry towards making models fairer. The problem is that current models are being trained with historical data that embed historical biases from which algorithms can learn. In most cases, these biases can be amplified by ML algorithms. In this chapter, we present an overview of algorithmic fairness, discuss some essential fairness metrics, and incorporate the idea of bias mitigation techniques.

## **4.1 Fairness in a machine learning context**

Fairness is a multidisciplinary concept widely used in philosophy, politics, law, etc. In general, fairness is linked to human rights and non-discriminatory decisions and the idea is built around things that are equal, impartial, and decisions without favouritism. From a regulatory point of view, the High-Level Expert Group on Artificial Intelligence of the (European Commission, 2019) on Artificial Intelligence defines two dimensions of fairness that must be taken into consideration when building fair machine learning models:

* **Substantive dimension**: Guarantee that individuals and groups are free from unfair bias, stigmatisation and discrimination by ensuring equal distribution of benefits and costs.
* **Procedural dimension**: This involves using effective governance frameworks and controls to find effective remedies for decisions made by AI systems (and the people who manage/operate them).

Defining and incorporating fairness in a quantitative setting has been a considerable challenge for researchers. In general, the literature on fairness in machine learning relies on the theory of justice by Rawls (1971), who presents the idea of justice as fairness and proposes a set of principles of justice that aims to identify favoured and deprived groups. The theory of justice by Rawls set the playground for modern research in algorithmic fairness; for example, studies by Joseph et al., (2016) and Shah et al., (2021) are solely based on the idea of Rawlsian concept of fairness. Considering Rawlsian justice theory, discrimination should not be related to irrelevant and intrinsic factors of individuals. Although fairness and justice are likely the same (according to Rawls), these are subjective concepts, meaning they can be interpreted differently among individuals, and differences in interpretations can sometimes be controversial.

Rawls also introduces the criterion of moral arbitrariness by which moral decisions are judged. In this respect, factors unrelated to moral considerations should not be included when deciding on moral dilemmas. We may distinguish situations in which specific attributes of individuals, such as nationality, are not consequences of moral considerations but rather a matter of randomness. Therefore, decisions unrelated to induvial-specific characteristics should not consider other factors specific to a person (say, nationality) that would make a decision biased. For instance, access to credit should not be subject to morally arbitrary factors, such as nationality, but rather to other factors.

We can spot two types of fairness from the literature: group and individual fairness. Bellamy et al., (2018) define group fairness as a subset of a population (population subgroup) defined by protected variables such as nationality, religion, gender, ethnicity, etc. This deprived subgroup seeks to be treated equally across all other subgroups. Contrary to that, individual fairness relates to equal treatment across similar individuals. This section will focus merely on a group fairness perspective.

There are three points of intervention in a machine learning context to control unwanted bias arising from protected variables: the training data, the underlying learning process, and the output predictions. Likewise, there are three types of bias mitigation techniques: pre-processing, in-processing and post-processing. The post-processing bias mitigation techniques are suitable for all kinds of models since they work on predictions and do not alter the model's internal structure. Basically, a post-processing bias mitigation technique takes a subset of samples and changes the prediction category label to meet the desired fairness metric. Note that a fairness metric quantifies the unwanted bias in the model, and the post-processing algorithm reduces this bias (Lohia et al., 2018).

In order to understand group fairness from a mathematical point of view, we need to follow the standard mathematical notation in algorithmic fairness, as in Lohia et al., (2018). Consider a binary classification problem with set of features (input vector), categorical protected attributes (such as gender, race, nationality, etc.), and a target output . Based on a set of provided samples we train a classifier such that . On this point, the value corresponds to the favoured subgroup and to the deprived subgroup. Likewise, stands for a favourable outcome and otherwise. The literature points out a standard measure to assess group fairness: the disparate impact.

Disparate impact is the adverse effect of decisions on a protected group of people compared to others. From a legal perspective, in order to avoid discrimination against protected groups according to anti-discrimination laws such as in the Title VII of the Civil Rights Act, (1964) the Code of Federal Regulations in the US sets a rule of thumb for considering a selection rate benchmark of 80% for any protected attribute (race, sex, nationality, etc.), also known as the “four-fifths rule”. In algorithmic fairness, this benchmark is taken into consideration, and a level of deviation is set, making the range of values for accepted fairness between 80 and 125%. Expressing group fairness through the disparate impact, we get the following formula:

|  |  |
| --- | --- |
|  | (55) |

From the above formula, the disparate impact is calculated as the outcome of the classifier for a sample with a protected attribute , divided by the outcome of a sample with a favoured attribute . According to the literature, we can consider group fairness if this ratio is between [0.8 – 1.25].

In the context of credit scoring, the risk presented in the lending activity is the counterparty default. Therefore, a lending institution must find a reliable way to assess and quantify this risk so it can be mitigated and benefit the business. Generally, when determining the risk of default, a lending entity assigns different ratings to the applicants and based on the score obtained, a lending decision takes place. However, the risk of default is financial, and according to the fairness definition, individual-intrinsic (protected features) characteristics should not be considered when assessing a loan application.

## **4.2 Unwanted bias in machine learning**

Unwanted bias sets privileged groups at a systematic advantage with respect to the unprivileged ones. These systematic disadvantages are unknown to the lending institution and result from the data or models used. Since machine learning models learn from data, generally, the cause of the unintentional bias is the data; however, this bias may also come from the model (algorithm) or the deployment decisions that reflect these cognitive biases against particular groups. It is important to note that bias may enter in different phases during the scorecard development. In this case, Makhlouf et al., (2022) define the sources of bias that enter in the steps of a model, say a scorecard development pipeline:

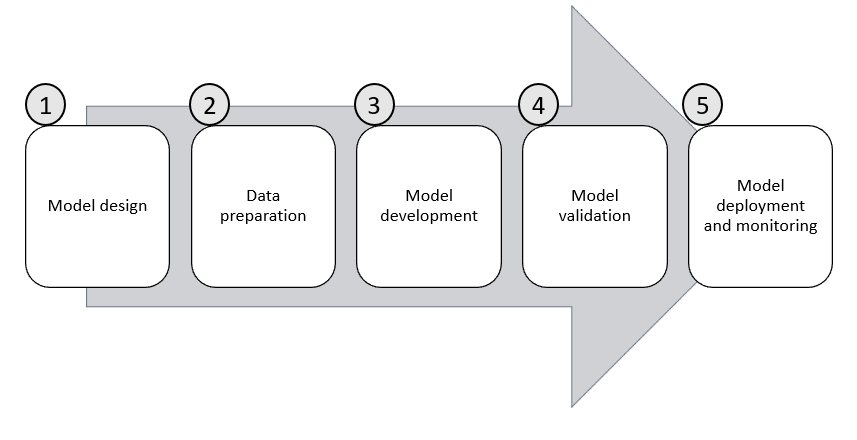


Figure 27. Scorecard development flowchart

Source: Own work

The first phase of model design is crucial since it defines the model's main characteristics, structure, and objectives; it should be detailed enough and cover the business needs. Whenever the model's design is not complete enough to cover different edge cases, we can encounter the exclusion bias that arises when certain groups are excluded from the model and consequently from the development dataset.

The second phrase regarding data preparation is the most susceptible step to unwanted bias. The most common type of data bias that enters in this phase are:

* Representation bias: occurs to the under-representation of certain groups in the dataset. Some protected groups cannot be representative enough when scoring a loan application from different groups. For example, the distribution of female applicants can be lower than male applicants, causing the model to be hampered (Caeyers & Marcel, 2020).
* Measurement bias: when a systematic error in data collection affects the input and target variables in a dataset, creating erroneous data and, in some cases, information asymmetry (Page & Henderson, 2008).
* Proxy bias: Arises when the input or target variable is calculated using a proxy variable that differs from the variable of interest. For example, using a zip code variable as a proxy for ethnicity may incorporate bias to the model if the zip code variable discriminates.
* Pre-processing bias: Occurs during the data pre-processing, and it is related to how missing data, outliers, and feature engineering are handled. An incorrect data pre-processing can contribute to systematic bias in the dataset.
* Historical bias: Arises due to historical disparities between groups that are present in the dataset. For example, the historical income gap between males and females. To some extent, historical bias can be linked to representation bias.

During the model development phase, our model can be exposed to aggregation bias that arises when a single model is used for groups with different risk distributions. The idea is to make a model for a particular use case and business need in order to ensure correct predictions (Makhlouf et al., 2022).

The model validation phase is susceptible to evaluation bias that occurs when external populations do not equally represent the various parts of the target user population. When testing a model, the validation dataset should be consistent with the data used for training, meaning that population sub-groups should follow a similar distribution.

Lastly, the model deployment and monitoring phase is sensitive to two types of unwanted bias:

* Deployment bias: Occurs when there is a discrepancy between the issue a model is expected to solve and the way by which the model is really utilised in practice.
* Drift bias: This is more common for monitoring-related tasks, and it is a type of bias that occurs over time post-implementation of the model. This bias occurs primarily due to data drift when the population’s main characteristics changes over time and the model remains the same. For example: due to a downturn in the economy, the payment patterns may change; therefore, the model should be re-calibrated to account for these new changes in the population. On the other hand, drift bias can also occur due to concept drift when the statistical properties of the target variable change because of risk trends, different behaviour patterns, technological improvements, and moral hazard related issues. For example, the risk assessment of unemployed people can differ from the employed population as both have different behaviours.

Ideally, during the scorecard development, there should be specific methods and tests in each step for identifying, quantifying, and assessing the unwanted bias, making it easier to correct. However, since this bias is unintentional and, in some cases, hard to detect, the resulting systematic bias can be handled with bias mitigation techniques, as explained in the previous section. However, it could be argued that trying to address these biases may increase the risk in the model and the business. When correcting the bias and ensuring parity among sub-groups, the bank may face a situation where the entry barrier to credit should be lowered, translating into a risk of higher default rates. Also, it may happen that by correcting the bias and increasing the chances of a favourable event for the deprived group, the favoured group can be worse off.

Historically disadvantaged groups have never enjoyed the benefits of a loan due to the limited access to credit from the existing historical bias. Consequently, we can theorise that the favourable outcome, in this case, getting a loan, leverages the advantaged group’s wealth since the loan can be invested and compounded easily. Here we can identify a spiralling effect that enlarges the differences between the rich and poor.

## **4.3 Important fairness metrics**

Let’s consider the scenario of a machine learning model that aims at predicting probabilities of default for clients applying for a loan in a bank; based on the obtained probability of default, the bank can decide whether to accept or reject the application. Such a model is trained on a dataset that includes protected attributes such as gender. The model's predicted probabilities are transformed into labels by setting a cut-off point. Probabilities that breach the benchmark are classified as a defaulted client and non-defaulted otherwise. This example will be considered for explaining the fairness metrics later in this section.

To be consistent with the notation, we consider the same mathematical notation defined in section 4.1. Once our classifier is trained and predictions are obtained, we can use a confusion matrix to observe the combinations of the 4 possible predicted and actual values. A confusion matrix creates a frequency table that summarises how the model performed in terms of classified observations and gives an overview of the strengths and weaknesses of the model (see the following figure). In the context of algorithmic fairness, we can use a confusion matrix to obtain the performance metrics of the model and compare across various protected subgroups to assess how the model discriminates between them and quantify the fairness in the model’s predictions.

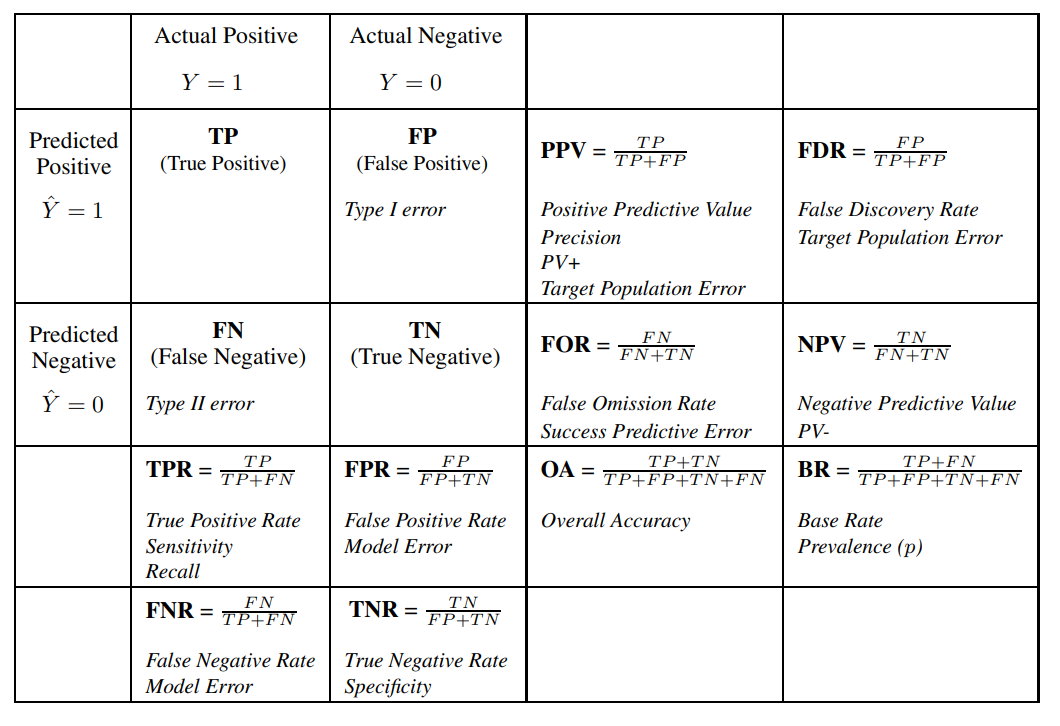


Figure 28. Confusion matrix metrics

Source: (Makhlouf et al., 2022).

The literature on algorithmic fairness has rapidly evolved over the past years, and new metrics are being developed to assess unfair systems. Due to regulations, companies and institutions are catching much attention towards implementing fair systems. From the literature, we can highlight the following most used fairness metrics:

* Statistical Parity: The notion of statistical parity requires predictions to be independent of the protected attributes. This group fairness metric ensures that the acceptance rate between subgroups is the same In this case, the confusion matrix can be used for formally defining the equality of acceptance rate between protected subgroups as follows: . A model with statistical parity will ensure the same proportion of males and females in the predictions of accepting or rejecting a loan application. This metric is also known as demographic parity or independence (Kusner et al., 2017).
* Predictive Parity: This metric requires the precision (positive predictive value) to be the same for the protected subgroups. Using this metric makes sense when no intrinsic differences between the groups are assumed. In the case there are intrinsic differences, then we should correct them. Otherwise, we are essentially doing reverse discrimination and thereby reducing model accuracy. Note that precision or PPV is the probability that individuals predicted to belong to the default class belong to this class. Mathematically speaking, predictive parity requires: . We can use the confusion matrix to illustrate the equality of precision for each of the protected sub-groups: . Some authors like Mayson, (2019) define a more constrained notion of the predictive parity in which the negative predictive value should also be the same between sub-groups, mathematically speaking: . This notion is also known as the overall predictive parity.
* Equalised Odds: The equalised odds, also known as the error balance rate, this notion considers the predicted and true labels. Given the actual outcome, the prediction is assumed to be independent of the protected attributed. It requires both sub-groups to have the same true positive rate (TPR) and false positive rate (FPR), mathematically speaking: . In order words, the probability of a customer whose loan application is accepted to be predicted accepted, and the probability of a customer who is not accepted to be incorrectly predicted accepted should be the same among protected sub-groups. Using the confusion matrix as an illustrative example: ^ . This metric can be understood as a notion of statistical parity, but it takes into account the errors instead of considering outcomes.
* Overall Accuracy Parity: As the name refers, this is achieved when the overall accuracy is the same for protected sub-groups. Using the mathematical notation, this implies that . Taking the confusion matrix as an example, we can illustrate the case when true negatives and true positives are equally considered and predicted . Thus, in other words, the probability of qualified applicants being correctly accepted for the loan and the probability of non-well qualified applicants being correctly rejected is the same for both subgroups (Berk et al., 2017).

Another set of important fairness metrics used in the literature and business cases is embedded in the Artificial Intelligence Fairness 360 (AIF360) python package, a toolkit for detecting, understanding, and mitigating unwanted bias by (Bellamy et al., (2018) for IBM Research. The list of available fairness metrics can be found in the documentation of **aif360.metrics** in the AIF360 python package, a module for computing fairness metrics from the raw dataset used for modelling purposes and fairness metrics for predicted labels (classification metrics). In this section, we will only focus on the fairness metrics that will be used in our numerical experiment.

To assess the bias presented in the dataset and answer questions related to the different treatment of protected subgroups, we could run a t-test and f-test for all sub-groups and test the null hypothesis of equality between sub-groups. Additionally, we could run fairness metrics that compare the true label for protected sub-groups, and in the case we find some biases and correct them using bias mitigation techniques, we could run the fairness metrics again on the predicted label and compare the differences between the biased truth labels and the fair predicted labels. Among the fairness metrics in the AIF360 python package, we can find:

* Base rate: The base rate of a group is defined as the ratio of people in the particular group that belongs to the positive class concerning the total number of people in that group. Basically, it tells us the probability of the positive class. The base rate in our example would be the default rate for the whole portfolio.

|  |  |
| --- | --- |
|  | (56) |

* Mean difference: describes the differences in the acceptance rates. This metric is also known as statistical parity difference. It should be noted that the following formula is similar to the disparate impact, but in this case, instead of a ratio, we have the difference. We have a statistical parity difference when the value equals zero.

|  |  |
| --- | --- |
|  | (57) |

* Smoothed empirical difference fairness: This fairness metric focuses on the formulated intersectional definition of fairness by Foulds et al., (2019). According to them, this definition states that “*regardless of the combination of protected attributes, the probabilities of the outcomes will be similar, as measured by the ratio versus other possible values of those variables, for small values of* ”. In this case, the value of is estimated using a symmetrical Dirichlet-multinomial model via the posterior predictive distribution, and this value is consistent with the 80% rule, meaning that . For this fairness definition, the following formula should hold for the dataset:

|  |  |
| --- | --- |
|  | (58) |

Following our example, and Are the number of observations granted a loan for different protected subgroups (genders) and the number of observations with a sensitive attribute, respectively.is the concentration parameter of the posterior predictive distribution, being the range of outcomes. Fairness is considered when the values of this formula are between the range [0.8, 1.25]

In addition, in the AIF360, we can also find fairness metrics that apply to the outcomes of the machine learning model. These metrics also provide a sense of model performance; therefore, the notation following is consistent with the confusion matrix presented in figure 28.

|  |  |
| --- | --- |
|  | (59) |

* Average odds difference: Measures the average of the difference in the model error (FPR) and sensitivity (TPR) for unprivileged and privileged groups
* Average absolute odds difference: Measures the average absolute difference in the model error (FPR) and sensitivity (TPR) for unprivileged and privileged groups. When the outcome of this metric is close to zero, we can assume fairness.

|  |  |
| --- | --- |
|  | (60) |

* Equal opportunity difference: the EOD, also known as the true positive rate (TPR) difference. This metric uses the TPR, indicating whether the model correctly classifies a customer that will not default. The TPR is calculated for each of the subgroups, and then the difference is calculated

|  |  |
| --- | --- |
|  | (61) |

* Error rate of the model: Also known as misclassification rate, this metric measures the percentage of misclassified observations to the total sample size. This metric can be calculated for each of the sub-groups, and then we can calculate the difference of this metric between the deprived and favoured sub-groups. We assume fairness for values of the difference close to zero. Alternatively, we could also calculate the ratio of this metric, being values close to one considered as fairness.

|  |  |
| --- | --- |
|  | (62) |

* False discovery rate: This metric describes the share of positively classified observations which were falsely identified as such. In order words, this metric allows us to calculate the type I error and compare between sub-groups. We can also calculate the difference or ratio of this metric between protected subgroups to assess fairness.

|  |  |
| --- | --- |
|  | (63) |

* False omission rate: Contrary to the false discovery rate, the FOR describes the share of negatively classified observations which were falsely identified as such. This metric is the type II error which can be compared between sub-groups to which we can calculate the difference or ratio as in previous metrics.

|  |  |
| --- | --- |
|  | (64) |

If bias is detected using the metrics mentioned above, the next step is to set a strategy to mitigate and remove this bias. Note that there are separate metrics for identifying bias in the dataset and bias in the model's predictions.

## **4.4 Unwanted algorithmic bias mitigation**

Potential bias mitigation actions can be diverse depending on the use case for our machine learning model and will depend on the outcomes of the fairness assessment conducted to detect bias. In section 4.2, we discussed that most of the bias in our system occurs during the data preparation phase. Bias mitigation in this sense may include interventions in this phase by improving how the data is collected and processed. However, the bias mitigation techniques referred to in this section focus on those that are implemented during the model development phase. During the model development phase, we can encounter three points of intervention for which bias can be mitigated:

* Pre-processing: This takes place before developing the model. These techniques aim to change the dataset's structure to remove the underlying bias from the data before training the model.
* In-processing: This takes place during the training phase of the model. These techniques aim to modify the underlying learning process of machine learning.
* Post-processing: This takes place on the predictions of the model. These techniques aim to modify the labels to achieve a fairness objective.

It should be noted that in order to achieve fairness, there is a trade-off with model performance. In most cases, the literature emphasises the loss in accuracy after bias mitigation. Later in this section, we will discuss some commonly used techniques in the literature and by practitioners, with a particular focus on pre and post-processing bias mitigation techniques in order to narrow down the scope of the study.

### **4.4.1 Pre-processing bias mitigation approaches**

Pre-processing bias mitigation techniques are plausible when modifying the training data is possible. The techniques used in this phase attempt to reduce bias before training the model. The idea is to achieve independence between the protected variable and other variables in the dataset, and this could be quite dangerous since reducing independence will come at the cost of model performance. Note that a model learns from the training data, and most likely, the predictions would not be much accurate. These bias mitigation techniques would not be much recommended in a credit scoring context since regulators emphasise the importance of model interpretability, and interpreting models with a changed structure may be quite difficult. The supported algorithms for pre-processing bias mitigation in the AIF360 python package include: Disparate Impact Remover, Learning Fair Representations, Optimised Preprocessing, and Reweighing (Bellamy et al., 2018). In this dissertation, we will focus on Disparate Impact Remover and Reweighing.

#### **4.4.1.1 Reweighing**

Kamiran & Calders, (2011) proposed this algorithm that generates a set of weights for the dataset to make it less discriminatory and bias-free. Essentially, this algorithm creates a flag variable that indicates if the corresponding observation needs to be pulled up or down using weights, this class value has positive and negative labels. An object is set between the positive and negative label group if the respective object has a score higher or lower than the average, indicating that those objects with a lower score will be pulled up (higher weight) and otherwise pulled down (lower weight), these scores are calculated from a machine learning model without modifying the input data.

The weights are calculated on two sub-groups separately. The first group contains observations by which the protected variable contributes to positively affecting the score of a particular observation (aka favoured group). The second group contains observations from the deprived group. The main assumption in this algorithm is that both sub-groups have the same class probability (score). Weights are calculated after proving the main assumption, meaning that observations with extremely high values (outliers) will be given a smaller weight than observations with lower values. The weights are calculated using the following formula (Kamiran & Calders, 2012)

|  |  |
| --- | --- |
|  | (65) |

Where is the weight for a particular observation X, is the expected probability in the non-discriminatory dataset, and is the observed probability in the actual dataset. In order words, weights are calculated as the expected probability with protected attribute and class given independence divided by the observed probability. These weights are used for training a classifier later on.

#### **4.4.1.2 Disparate impact remover**

The Disparate impact remover algorithm proposed by (Feldman et al., 2014) changes the values of the attributes of an object to aim to increase group fairness and, at the same time, preserve rank-ordering within groups. In previous sections, we have already introduced and discussed the concept of disparate impact for evaluating bias. It is known that some observations have attributes that place them into a favoured or deprived group. When modifying the values of the sensitive attributes, the information about to which group an observation belongs cannot be inferred, therefore increasing bias.

To illustrate this concept, we will use the example provided by Ronaghan, S. (2019) in figure 29, which uses a dataset with a single continuous variable with values between 0 and 10, a sensitive binary variable, and a target variable.

|  |
| --- |
|  |

Figure 29. Effect of Disparate Impact Remover algorithm

Source: Ronaghan, S. (2019)

From the above figure, it can be noted that initially, the distributions for each sub-groups were distant and easily identifiable; however, after applying the disparate impact remover pre-processing technique, both distributions were equalised (overlapped), making it hard to distinguish between the protected sub-groups. The distributions can be overlapped according to a parameter known as *repair\_level*. If this parameter is set to 0.8, then 80% of the observations are equalised. The above example uses a *repair\_level* of 1.

### **4.4.2 Post-processing bias mitigation approaches**

Post-processing bias mitigation techniques take place after training the model and getting predictions, and these techniques can be applied to any algorithm and are recommended for treating bias from already trained classifiers. The general approach of these algorithms is to take a subset of observations and change the predicted label to comply with a fairness metric requirement; therefore, the model becomes fair-aware by the time of predictions. In the context of credit scoring, these types of techniques are more plausible from a regulatory point of view.

#### **4.4.2.1 Equalised odds**

This algorithm proposed by Hardt et al. (2016) relies on the criterion that the average odds are achieved when the predicted score for observation and the protected variable is independent and conditional to the actual outcome. This algorithm modifies the positive class probability by relying on the concept of equalised odds to provide a new unbiased, positive class. In a nutshell, according to Bellamy et al., (2018), the equalised odds post-processing technique “*solves a linear program with which to find probabilities to change output labels to optimize equalized odds*”.

This algorithm creates a new class probability using a loss function () that describes the loss incurred when predicting the positive class probability given a different outcome aiming at optimising for a linear function to transform the positive class probabilities into the equal odds positive class probability. This optimisation problem resulted in a trade-off between the positive class probability and equalised odds probability.

#### **4.4.2.2 Reject option classification**

This algorithm proposed by Kamiran et al. (2012) uses the positive class probability to relabel and remove bias from the predictions of the model. According to Bellamy et al., (2018), this algorithm “*gives favorable outcomes to unprivileged groups and unfavorable outcomes to privileged groups in a confidence band around the decision boundary with the highest uncertainty*”. The intuition of this algorithm is based on the idea that observations with a positive class probability close to 1 or 0 will report a higher degree of certainty. Scores close to the minimum or the maximum of the distribution will remain in their corresponding groups; however, observations close to the boundary decision cut-off of 0.5 will present a more uncertain outcome.

Observations around the boundary decision cut-off are known to belong to a “critical region” called *D*. Essentially, observations in this region are more likely to be influenced by bias. For calculating the range in space for this critical range, let's assume a parameter that controls the size of the critical region in order to eliminate bias. For constructing the critical region, we take the absolute distance of the predicted class probability with respect to the upper or lower boundary, being the critical region defined as the

Observations that fall in the critical will have a value of lower or equal to . These observations in the critical region will be incorporated into the negative class label if they belong to the favoured group; likewise, observations be incorporated into the positive label class if they belong to the deprived group. On the other hand, objects not lying in the critical region will be labelled similarly to the reweighing method explained in section 4.4.1.1, where observations are placed into the positive or negative group if the score of this object is higher or lower than the average score. Since there is a trade-off between fairness and model performance, Kamiran et al. (2012) express that this method results in the lowest loss in accuracy due to how the critical region is constructed.

# **5. Methodology and Data**

This section is devoted to explaining the data used for the experiments and the approach used for modelling, specifying each of the steps followed to obtain the final results presented in chapter 6.

## **5.1 Methodology**

This study is primarily motivated by the increasing research in algorithmic fairness and the different techniques available to quantify, assess and mitigate unwanted bias from protected variables. In particular, we take as reference the studies conducted by Kozodoi et al. (2021) and Verkoulen et al. (2022), which from a credit scoring perspective, tackled the problem of unwanted bias in different phases of the modelling pipeline. These studies compared the performance of various bias mitigation techniques and the performance of the models accordingly to check for the potential losses in accuracy as per the existing fairness-accuracy trade-off documented in the literature.

In this study, we focus on the effectiveness of bias mitigation in credit scoring using a parametric model such as logistic regression (LR), a common modelling approach in credit scoring, and non-parametric models inspired by tree-based algorithms such as random forest (RF) and extreme gradient boosting (XGBoost). In this sense, this research aims at answering the following questions:

1. *How are bias mitigation techniques in the context of credit scoring effective in reducing unwanted bias that arises from the inclusion of protected variables?*
2. *Which bias mitigation technique is more effective at reducing existing biases?*

In previous sections of this dissertation, we discussed different machine learning algorithms used in credit scoring. In this sense, we are interested in assessing unwanted bias and the impact generated by each of these algorithms. On this matter, we set our research hypothesis:

*Considering that non-parametric models amplify existing biases. Therefore, bias mitigation techniques are more effective for non-parametric models, particularly for random forests.*

Many research studies are inspired by the framework for mitigating bias in the Artificial Intelligence Fairness 360 (AIF360) toolkit developed by IBM research. This python package contains techniques to detect and mitigate bias, with over 70 fairness metrics and 10 bias mitigation algorithms. These techniques are designed to be applied in different phases of the modelling pipeline: pre, in, and post-processing.

This study only focuses on pre and post-processing bias mitigation techniques. Due to the usage of non-parametric models such as random forest and extreme gradient boosting the learning process, to some point, is more complex than a simple logistic regression. Since regulators emphasize the importance of model interpretability, adding an extra layer of complexity to our algorithms may not be plausible in the context of machine learning in credit scoring.

This research is split into three phases. Firstly, we calculate and assess the bias from the raw dataset that includes protected variables. Secondly, we mitigate the bias using the embedded techniques in the AIF360 python package. Lastly, we compare different metrics and results to draw conclusions. It is important to highlight that the code created for this study is fully conducted in the python programming language due to the availability of different libraries for data manipulation, visualisation, machine learning, and fairness quantification.

To achieve our results, we first train a machine learning model where the protected variables are included in the training dataset. Then we train a second model that is blind to those protected variables, meaning that the protected attributes are not included in the training dataset; this is known in the literature as debiasing by unawareness. Lastly, we apply the fairness pipeline proposed by Bellamy et al. (2018) that incorporates all bias mitigation techniques (see figure 30). It is important to highlight that all machine learning models used in the pipeline are trained by unawareness, meaning that the protected attributes are not included in the training process of the models. This process is repeated for each scoped machine learning model (logistic regression, random forest and extreme gradient boosting), and the results are compared accordingly.

### **5.1.1. Bias detection**

As mentioned before, firstly, we test for bias in the dataset, which corresponds to the first stage of this research study. The dataset used is fully described and summarised in the data section of this chapter, in section 5.2. To identify and quantify bias in the dataset, we run a t-test and f-test to test for different outcomes between the protected sub-groups in the sample for a particular confidence level. In this case, the protected variables in the dataset are race, ethnicity, and gender. On top of that, we will address the bias quantification issue by calculating some fairness metrics already discussed in this dissertation and included in the AIF360 python package, more specifically, the base rate, disparate impact, mean difference (also known as statistical parity difference), and the smoothed empirical difference fairness. We conclude in this section that our dataset is biased towards individuals of different ethnicities and genders.

### **5.1.2. Scorecard development**

Regarding the modelling approach, we decided to use logistic regression, random forest, and extreme gradient boosting as models to predict the probability of being rejected for each of the customers in our dataset. At the same time, we assess if these models are impacted differently by the various bias mitigation techniques employed and previously mentioned.

Initially, we rely on the weight of evidence (WoE) transformation of our dataset to prepare the data for training purposes. Basically, the idea is that for each protected sub-group in the raw data, we calculate the log ratio of the distribution of accepted and rejected customers (event and no event). This special binning method is very common in credit scoring modelling and has been widely used over the last 4 decades (Siddiqi, 2013). We use WoE variables to determine the predominant distribution of events for each of the classes. More importantly, WoE has important benefits such as handling outliers and missing values and can be used to calculate the information value (IV). The information value describes the predictive power of each class in a variable and is helpful for variable selection. That being said, we rely on WoE transformed variables for training and IV for variable selection.

After performing the WoE transformation on our dataset and selecting the most relevant variables, we train the aforementioned models without mitigating any potential bias. In this case, we find that the outcomes of these models are biased by running some fairness metrics on the models' predictions. We expect to observe amplification of the dataset's bias on the biased models' predictions. Likewise, we will calculate the same metrics as section 5.1.1 on the resulting dataset with predicted labels. Additionally, we will calculate some metrics to assess the fairness of the predictions, making a particular focus on the metrics mentioned in section 4.3: The average odds difference, Average absolute odds difference, equal opportunity difference, error rate, error rate ratio, false discovery rate, false discovery ratio, false omission rate, and false omission ratio.

It is important to highlight that we perform hyperparameter tuning on tree-based algorithms (Random Forest and XGBoost) using stochastic search and cross-validation. Also, considering the fact that some of our input variables are slightly correlated, we perform ridge regression (L2 regularisation) with a very small penalty. Furthermore, among the protected features found in the dataset (race, ethnicity, and gender), we decided to narrow down the scope of the analysis and conduct our study only considering the variable “race” since it has the largest information value with respect the target variable, as observed in section 5.2 (further details in table 18).

### **5.1.3. Bias mitigation**

The literature on algorithmic fairness for bias mitigation follows a general workflow, and this workflow is primarily incorporated in the AIF360 toolkit, which is presented in figure 30. The general pre-processing bias mitigation technique approach starts by loading and transforming the data into a fair dataset using a pre-processing algorithm. Then the transformed dataset is split into the training and test sets containing 80% and 20% of the observations. Subsequently, a model is trained on the fair dataset and predictions are obtained from the model. Later the probability scores are transformed into labels which are used to assess the fairness of the predictions. We use the reweighing and disparate impact remover techniques for pre-processing bias mitigation techniques.

Following the example of a fairness pipeline as in figure 30, we focus on the equalised odds and reject option classification in the case of bias post-processing techniques. Both techniques differ from each other; the equalised odds classification aims to reach an equalised odds of 1 (equality of opportunity), whereas the reject option classification aims to obtain statistical parity for all groups with protected variables. To implement these techniques in the fairness pipeline, we will load the data that is then split into the training and test sets containing 80% and 20% of the observations, respectively. Then we train a ML model from which we obtain predicted probabilities that are consequently converted into predicted labels by setting a cut-off threshold. Ultimately, the predicted labels are passed through a fair post-processor to obtain fair predictions.

Fair predictions from pre and post-processing techniques are also tested against the test dataset for calculating model performance and fairness metrics provided in chapter 6. It is expected to see model performance decrease while achieving fairness. Lastly, for comparative purposes, we apply the fairness framework for each scoped machine learning model (LR, RF, XGBoost) and follow the example of the fairness pipeline.

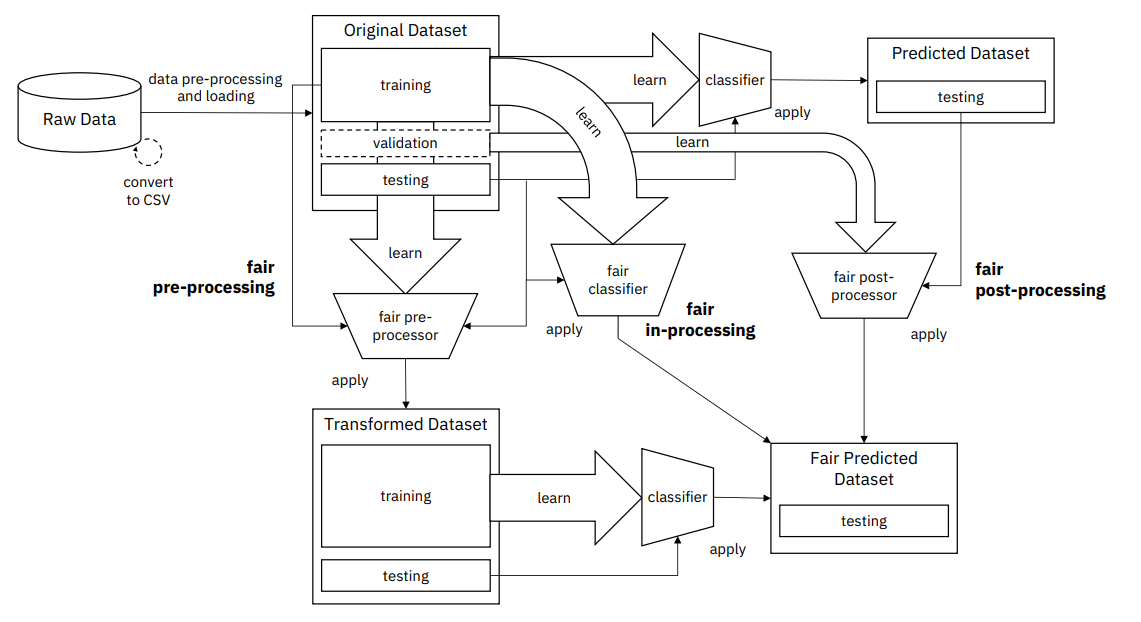


Figure 30. Fairness pipeline using AIF360

Source: Bellamy et al. (2018)

## **5.2 Data**

For this research, we use the Home Mortgage Disclosure Act (HMDA) database to retrieve data from the Consumer Financial Protection Bureau for 2017. The dataset contains 14,285,496 observations and 78 features with general information about mortgage applications in the United States of America. It contains features about the applicant information, property location, loan information, purchaser and denial information, census information and other data. We filter out variables concerning the customer's application and relevant information that happened before the approval or denial of a loan having a final dataset with 11 features. For computation simplicity, we extract a representative random sample of 600,000 observations without replacement from the dataset that will be used for further calculations and results.

The final dataset contains 3 protected variables: race, ethnicity, and gender. However, as mentioned in section 5.1.2, the calculations and modelling analysis will be conducted on the race variable. Also, we found that the variables loan amount and applicant income have 0.03% and 5.6% of missing observations, respectively. The missing observations were imputed with the median since the distributions of these continuous variables are skewed according to table 13. The rest of the variables have no missing values. Our target variable is the action taken, whether a loan has been approved or rejected.

This section presents a description of the dataset and features used for modelling. First, we present in table 13 descriptive statistics for continuous variables from the final dataset.

|  |  |  |
| --- | --- | --- |
| **Description** | **Loan Amount (thousands)** | **Applicant Income (thousands)** |
| Number of observations | 600000 | 600000 |
| Mean | 247.13 | 112.14 |
| Standard Deviation | 1064.67 | 775.82 |
| Min amount | 1 | 1 |
| 25% | 104 | 52 |
| 50% | 180 | 78 |
| 75% | 288 | 119 |
| Max amount | 350000 | 260000 |
| Number of Outliers | 913 | 454 |

Table 13. General statistics: numerical variables

Source: Own work

The above table indicates that in 2017, the mean income of an applicant was 112,140 USD, and the mean loan amount was 247,130 USD. Since the maximum values in the distributions of loan amount and applicant income vary considerably respecting the 50 and 70-ith percentile, we have strong evidence of very skewed distributions with the presence of outliers. The number of outliers is calculated following the Three-Sigma Rule, which considers observation as an outlier if the value is 3 standard deviations from the mean. The presence of outliers can be visible in the following boxplots for the corresponding continuous variables in the dataset.

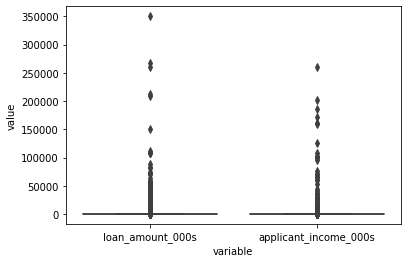


Figure 31. Boxplots of continuous variables in the dataset

Source: Own work

No treatment for outliers is conducted since we pretend to perform the weight of evidence (WoE) transformation on the input variables later in this section, since it nicely handles outliers, among other positive properties (Siddiqi, 2013). Regarding the analysis of categorical variables, we present descriptive statistics of these variables in table 14.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variables** | **Num of observations** | **Unique values** | **Top class** | **Frequency** |
| Loan Type | 600000 | [1, 2, 3, 4] | 1 | 73.8% |
| Property Type | 600000 | [1, 2, 3] | 1 | 95.5% |
| Loan Purpose | 600000 | [1, 3, 2] | 1 | 52.6% |
| Owner Occupancy | 600000 | [1, 2, 3] | 1 | 89.4% |
| Region | 600000 | [0, 1, 2, 3, 4, 5] | 3 | 35.7% |
| Ethnicity | 600000 | [1, 2, 3, 4] | 2 | 76.6% |
| Race | 600000 | [1, 2, 3, 4, 5, 6, 7] | 5 | 72.3% |
| Gender | 600000 | [1, 2, 3, 4] | 1 | 62.1% |
| Target (Approved) | 600000 | [0, 1] | 0 | 79.5% |

Table 14. General statistics: categorical variables

Source: Own work

From the above table, we can observe that some classes have a high frequency of observations, suggesting class imbalance in the dataset in many variables. This is not the case for the variable “Region”, where 1 class holds around 36% of observations. The following bar chart represents the distribution of the target variable regarding approved and rejected loan applications. From figure 32, 79.5% of the total applications are approved customers, a common scenario in credit scoring where negative events (rejects) are less represented in the sample.

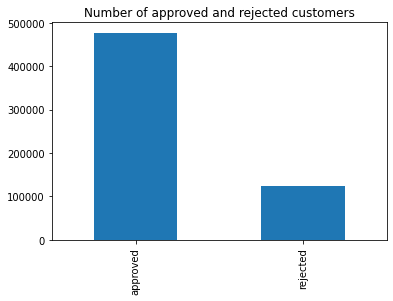


Figure 32. Barplot target variable – frequency of observations

Source: Own work

When analysing the distribution of the protected variables in the dataset and the proportion of rejected and approved events for each of the protected sub-groups, we find out that for the “Race” variable, white applicants are predominant in the distribution and have the majority of approvals, followed by black applicants which are considered as a protected sub-group.

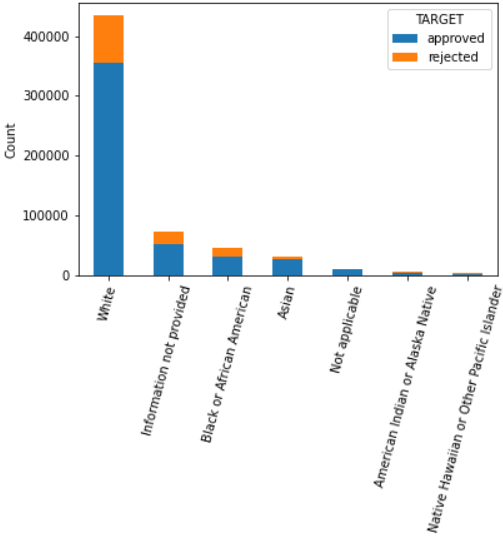


Figure 33. Barplot race variable – frequency of observations

Source: Own work

Similarly, for the ethnicity variable, the “Not Hispanic or Latino” class holds the majority of frequency of observations, and the size compared to the protected sub-group “Hispanic or Latino” is considerably large. Also, the proportion of approved applicants in the favoured sub-group is larger than in the protected sub-group.

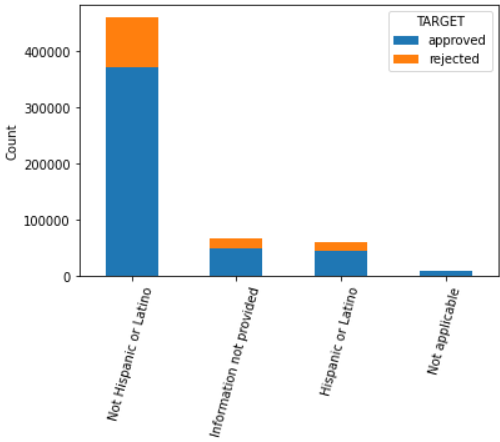


Figure 34. Barplot ethnicity variable– frequency of observations

Source: Own work

Regarding the last protected variable, “gender”, we find that male applicants are more frequent in the sample than females. Likewise, as in previously presented protected variables, favoured sub-groups are over-represented. Also, the proportion of positive events for the favoured sub-groups is larger than for deprived ones. These class imbalances clearly indicate biases in the dataset that may be translated into further predictions.

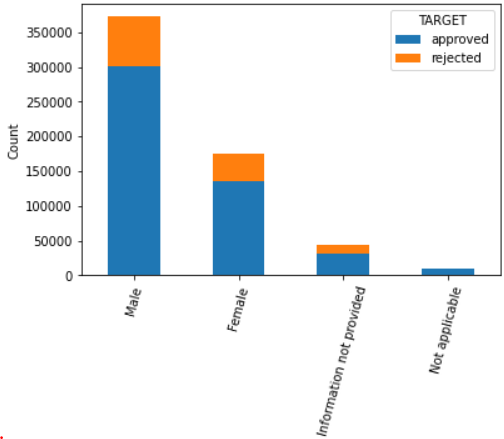


Figure 35 Barplot gender variable – frequency of observations

Source: Own work

The demographic characteristics of the applications suggest that most of the applications are from California, Texas and Florida; hence, most of the approved and rejected applications are around these states. However, the south region of the USA concentrated the majority of mortgage applications in 2017. Contrary to that, Alaska and North Dakota are at the tail of the distribution with few loan applications.

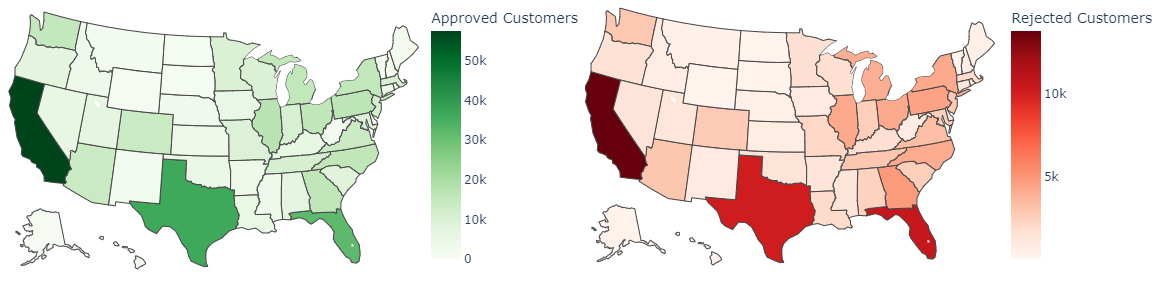


Figure 36. Approved and rejected customers per state

Source: Own work

Multicollinearity is an important issue to be aware of in credit scoring modelling. For that, we check the level of correlation across variables.

Regarding continuous variables, applicant income and loan amount, we find a weak Pearson correlation of 0.135. For categorical variables, we calculate the Cramer’s V coefficients shown in table 15 and represent the results as a heatmap.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Loan Type** | **Property Type** | **Loan Purpose** | **Owner Occupancy** | **Region** | **Ethnicity** | **Race** | **Gender** | **TARGET** |
| **Loan Type** | 1 | 0 | 0.02 | 0.02 | 0 | 0.01 | 0.01 | 0.01 | 0 |
| **Property Type** | 0 | 1 | 0.01 | 0.29 | 0.01 | 0.09 | 0.1 | 0.1 | 0.03 |
| **Loan Purpose** | 0.02 | 0.01 | 1 | 0 | 0.01 | 0 | 0.01 | 0 | 0.04 |
| **Owner Occupancy** | 0.02 | 0.29 | 0 | 1 | 0 | 0.11 | 0.11 | 0.11 | 0 |
| **Region** | 0 | 0.01 | 0.01 | 0 | 1 | 0.01 | 0.01 | 0 | 0.01 |
| **Ethnicity** | 0.01 | 0.09 | 0 | 0.11 | 0.01 | 1 | 0.56 | 0.51 | 0.01 |
| **Race** | 0.01 | 0.1 | 0.01 | 0.11 | 0.01 | 0.56 | 1 | 0.5 | 0.02 |
| **Gender** | 0.01 | 0.1 | 0 | 0.11 | 0 | 0.51 | 0.5 | 1 | 0.01 |
| **TARGET** | 0 | 0.03 | 0.04 | 0 | 0.01 | 0.01 | 0.02 | 0.01 | 1 |

Table 15. Association matrix heatmap - Cramer’s V for categorical variables

Source: Own work

The above heatmap shows that the race variable has a moderate correlation with other protected variables such as ethnicity and gender. However, the rest of the variables have a low correlation. Therefore, we do not find any strong alarming correlation in our dataset that must be treated. Note that we apply the L2 regularisation with a small penalty for modelling purposes due to these moderate correlations between the protected variables.

As discussed in the previous sections of this dissertation, we can apply tests of the central tendency of the distribution for each sub-group and compare between samples. In particular, we will use the t-test and f-test. For gender, we test male against female; for race, we test white against blacks and others; and for ethnicity, we test others against Latino. Then we perform an f-test to compare the variable's variance across the protected sub-groups. The results are summarised in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Variables** | **Statistical tests** | **Statistic** | **p-value** |
| Gender | T-test: Male and Female | 31.88 | < 0.01 |
| F-test: Male, Female and Others | 800.85 | < 0.01 |
| Race | T-test: White and Others | -70.77 | < 0.01 |
| T-test: White and Black | -79.78 | < 0.01 |
| F-test: White, Asian, Black, and Others | 2773.12 | < 0.01 |
| F-test: White, Asian, and Black | 3273.19 | < 0.01 |
| Ethnicity | T-test: Others and Latino | 37.09 | < 0.01 |
| F-test: Others and Latino | 1217.87 | < 0.01 |

Table 16. Central tendency tests for protected subgroups

Source: Own work

The significance of the above tests suggests a difference in the distribution, meaning that the central tendency of the distribution of protected sub-groups is different from favoured groups. This is a key indicator when assessing biases in the dataset. However, the leverage on the fairness metrics proposed in previous sections and available in the AIF360 package.

|  |  |  |  |
| --- | --- | --- | --- |
| **Fairness Metrics** | **Race** | **Ethnicity** | **Gender** |
| Base Rate | 0.2052 | 0.2052 | 0.2052 |
| Disparate Impact | 0.5746 | 0.7903 | 0.8676 |
| Mean Difference | -0.1439 | -0.0530 | -0.0300 |
| Smoothed Empirical Differential Fairness | 0.5541 | 0.2353 | 0.1420 |

Table 17. Fairness metrics on the raw dataset

Source: Own work

The fairness metrics used are calculated for all protected variables (race, ethnicity, and gender) on the raw data. We find that all variables have the same base rate, which indicates that the probability of being accepted is 20.5%. Regarding the disparate impact, we can observe that the value for Gender is between the range of fairness indicated by the fourth-fifth rule (0.8 – 1.25); however, this is not the case for ethnicity and race. In this case, there is clear discrimination in loan acceptance with respect to the race and ethnicity of applicants. Note that disparate impact for gender is close to the boundary of the fairness range, and still, there is a significant difference in the treatment of female and male applicants.

The existing bias is exacerbated in the case of the race of applicants. In fact, the mean difference confirms this situation. For the race variable, it indicates that there is a 14.4% difference in the probability of being accepted between white and black applicants. In the case of ethnicity, this difference between protected and favoured sub-groups is 5%, and for gender, this value falls to 3%, which is an acceptable value in the fairness literature. Another metric to consider when assessing the unfair treatment among applicants is the Smoothed Empirical Differential Fairness (SEDF), values close to zero indicate the same acceptance rate; in other words, the same probability of being accepted, we can observe that this value is higher than zero for all variables and particularly high for the race variable.

After analysing the fairness in our raw dataset, we proceed to perform the weight of evidence transformation in each of the features (except the target) with the aim of handling outliers and getting a more appropriate dataset for training purposes. After implementing the WoE transformation, we calculate the information value (IV) for each feature as the difference in the percentage of event and non-event weighted by the WoE of the corresponding class. We can use the values of the IV to filter out irrelevant variables. The convention in the literature is to exclude variables whose IV is lower than 0.02. Variables with an IV between 0.02 and 0.1 have a weak relationship with the odds ratio of the event, those with an IV between 0.1 and 0.3 have a moderate relationship, and between 0.3 and 0.5 and strong relationship (Siddiqi, 2013).

|  |  |
| --- | --- |
| **Variable** | **IV** |
| Loan Purpose | 0.235 |
| Applicant Income | 0.217 |
| Loan Amount | 0.216 |
| Property Type | 0.143 |
| Race | 0.112 |
| Ethnicity | 0.063 |
| Gender | 0.050 |
| Region | 0.030 |
| Owner Occupancy | 0.004 |
| Loan Type | 0.002 |

Table 18. Information Value (IV) per variable

Source: Own work

From the above table, we exclude from the training the variables owner occupancy and loan type since they are not useful for modelling. Loan purpose, applicant income, loan amount, property type and race have a medium relationship with the odds ratio of accepted loan applications.

# **6. Results and discussion**

This section presents and discusses the main findings obtained after implementing the fairness pipeline presented in figure 30 for bias mitigation. The following results aim at responding to the research questions and hypothesis exposed in chapter 5. In order to do that, the algorithmic fairness analysis is divided into three sections per the type of credit scoring algorithm used.

## **6.1 Logistic regression**

The fairness pipeline is applied in order to mitigate the bias detected in section 5.2 regarding the variable race. In this section, we consider two scenarios, the first one in which the protected attribute (race) is included in the training process of the logistic regression model (hereinafter, the discriminative model) and the second one, in which the protected variable is excluded from the training (hereinafter, the blind model. For the bias mitigation techniques, we use a blind model. The results are summarised in Table 19.

Recalling the disparate impact value from the raw dataset in table 17, the value for the race was variable 0.5746, meaning that the distance to the fairness range was 0.225. In the results from table 19, we can observe the disparate impact for the discriminative and blind model is 1.963 and 1.329, respectively, meaning that the distance to the closest fairness threshold of the fourth-fifth rule is 0.713 and 0.079, respectively. This result indicates that debiasing a model by unawareness does not necessarily mean that the model will be fair and does not solve the problem at hand, even though our fairness metrics improve. Given the values of the disparate impact metric, we can ensure that predictions from the discriminative model are less fair than predictions from the blind model.

When debiasing a model using pre-processing techniques, we find similar results as debiasing by unawareness (blind model). Pre-processing techniques achieve identical fairness metrics in terms of disparate impact, mean difference, average odds difference, and equality of opportunity difference. The values are really close as debiasing by unawareness. This situation is similar to the Reject Option Classification from the post-processing bias mitigation methods. In fact, some of the extra fairness metrics regarding the model performance towards protected subgroups are also similar. This, however, is not the case for the Equalised Odds, which outperforms the rest of the bias mitigation techniques and provides fair labels. In general, we can say that bias is amplified when developing a discriminative model.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Fairness Metrics** | **Discriminative model** | **Fair model** | **Pre-processing** | | **Post-processing** | |
| **Initial model** | **Blind model** | **Disparate Impact** | **Reweighting** | **Equalised Odds** | **Reject Option Classification** |
| Base rate | 0.428 | 0.419 | 0.427 | 0.411 | 0.352 | 0.419 |
| Disparate impact | 1.963 | 1.329 | 1.328 | 1.326 | 1.104 | 1.329 |
| Mean difference | 0.361 | 0.129 | 0.134 | 0.125 | 0.046 | 0.129 |
| SEDF | 4.93 | 4.419 | 3.608 | 3.918 | 9.957 | 4.419 |
| AOD | 0.297 | 0.08 | 0.084 | 0.077 | -0.001 | 0.08 |
| AAOD | 0.297 | 0.08 | 0.084 | 0.077 | 0.002 | 0.08 |
| EOD | 0.252 | 0.08 | 0.082 | 0.079 | -0.002 | 0.08 |
| Error rate | 0.35 | 0.347 | 0.351 | 0.342 | 0.346 | 0.347 |
| Error rate ratio | 1.454 | 1.084 | 1.076 | 1.082 | 0.979 | 1.084 |
| False Discovery Rate | 0.669 | 0.669 | 0.671 | 0.666 | 0.699 | 0.669 |
| False Discovery Ratio | 0.855 | 0.753 | 0.758 | 0.748 | 0.734 | 0.753 |
| False Omission Rate | 0.111 | 0.114 | 0.113 | 0.116 | 0.153 | 0.114 |
| False Omission Ratio | 1.404 | 1.829 | 1.819 | 1.823 | 2.041 | 1.829 |

Table 19. Results fairness pipeline – Logistic Regression

Source: Own work

In general, we can observe how bias is incremented by the discriminative model in figure 37. The gaps in the model performance between the sub-groups (white and black) represent the unwanted and amplified bias existing in the predictions.

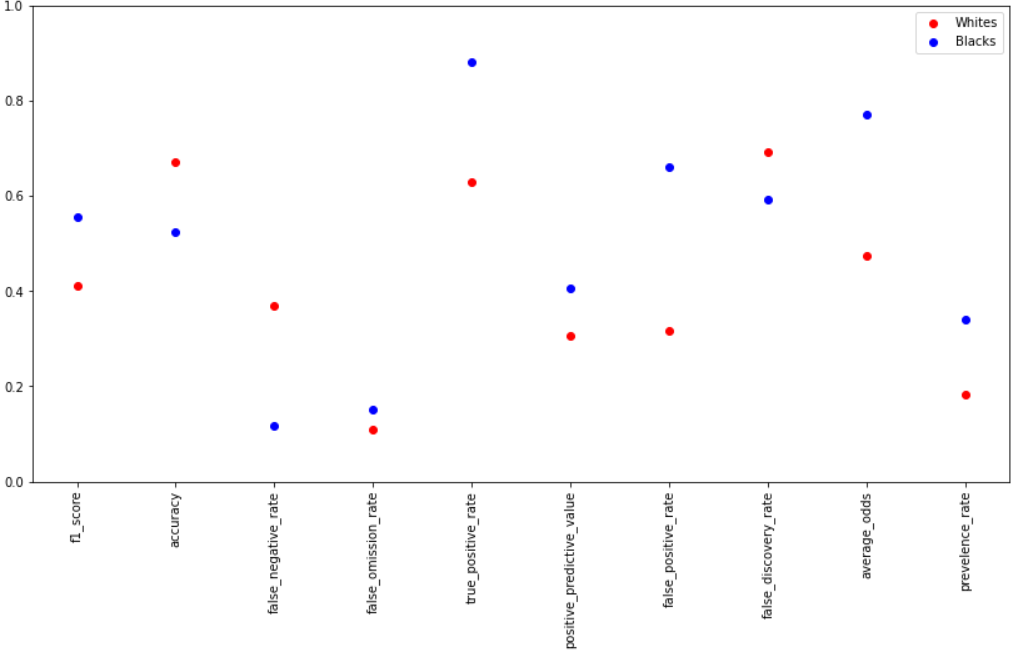


Figure 37. Discriminative model fairness – logistic regression

Source: Own work

However, when debiasing the logistic regression model using the equalised odds technique, we can observe most of the gaps are eliminated and, in some cases, reduced considerably, producing fair predictions. Further details of these results are available in table 23.

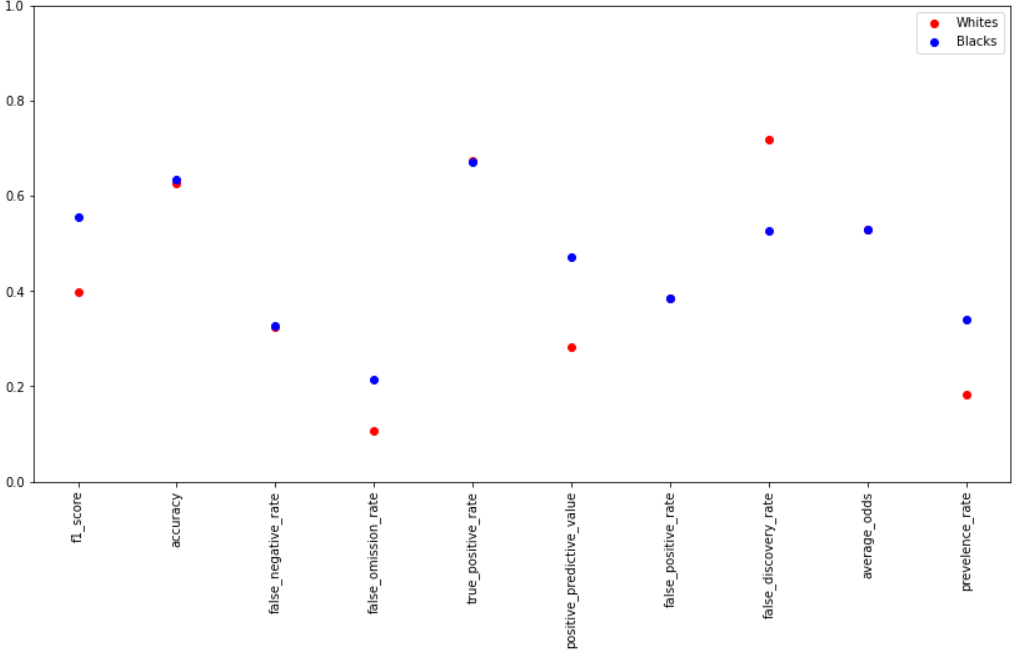


Figure 38. Equalised odds model fairness – logistic regression

Source: Own work

## **6.2 Random Forest**

Similar to logistic regression, we debias our blind random forest model using pre- and post-processing techniques, for which the results of this section are summarised in table 20. The discriminative random forest model presents unfair predictions that are confirmed by a disparate impact higher than 2.1. However, the blind model reduces the bias considerably but still does not reach the fairness threshold since the disparate impact is 1.33. This is a similar outcome as in the logistic regression model.

The distance in the value of the disparate impact of the discriminative model is 0.86, meaning that it is almost 3 times higher than the disparate impact analysed in the raw dataset. Clearly, we can observe that machine learning models amplify the disparate impact and increase unfairness. In terms of disparate impact, we can observe that reweighting and equalised odds bias mitigation techniques provide values that are between the range of fairness; nonetheless, the value of equalised odds is closer to 1, meaning that it provides even more fair predictions. In this case, we find that equalised odds technique is more efficient than reweighting to mitigate unwanted bias in the case of a random forest model and, in general, is superior to other bias mitigation techniques analysed in the scope of this research.

Although the equalised odds technique outperforms comparing to other bias mitigation techniques, it has the highest false omission rate and false discovery rate. Although these two metrics are analogous to some point and can be closely related, we can observe that equalised odds achieves fairness at the cost of higher highest false omission rate and false discovery rate. The values for the average odds difference, Average absolute odds difference, and equal opportunity difference of the unbiased model with equalised odds technique are close to zero, which is a technique that ratifies the existence of fair predicted labels.

Later in figure 39, it can be observed that the discriminative random forest model increases the gaps between black and white applicants, and some of these gaps quite are similar in size compared to logistic regression as per tables 23 and 24 in the appendices. Likewise, when using the equalised odds technique, the gaps in the model performance are closed in some of the fairness metrics, as shown in figure 40.

Regarding the model performance, we found out that equalised odds technique has a lower loss in accuracy towards achieving fairness. Contrary to that, the pre-processing disparate impact bias mitigation technique has the highest loss in accuracy. However, the predictions do not enter the fairness area.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Fairness Metrics** | **Discriminative model** | **Fair model** | **Pre-processing** | | **Post-processing** | |
| **Initial model** | **Blind model** | **Disparate Impact** | **Reweighting** | **Equalised Odds** | **Reject Option Classification** |
| Base rate | 0.358 | 0.368 | 0.378 | 0.372 | 0.302 | 0.371 |
| Disparate impact | 2.110 | 1.334 | 1.329 | 1.256 | 1.128 | 1.329 |
| Mean difference | 0.334 | 0.114 | 0.118 | 0.090 | 0.048 | 0.113 |
| SEDF | 2.972 | 2.942 | 2.788 | 2.948 | 9.820 | 2.679 |
| AOD | 0.277 | 0.066 | 0.070 | 0.042 | 0.000 | 0.065 |
| AAOD | 0.277 | 0.066 | 0.070 | 0.042 | 0.001 | 0.065 |
| EOD | 0.250 | 0.072 | 0.077 | 0.050 | -0.001 | 0.070 |
| Error rate | 0.310 | 0.317 | 0.321 | 0.319 | 0.312 | 0.318 |
| Error rate ratio | 1.524 | 1.110 | 1.090 | 1.069 | 1.031 | 1.107 |
| False Discovery Rate | 0.647 | 0.651 | 0.654 | 0.654 | 0.676 | 0.652 |
| False Discovery Ratio | 0.858 | 0.735 | 0.735 | 0.719 | 0.718 | 0.735 |
| False Omission Rate | 0.123 | 0.122 | 0.120 | 0.122 | 0.154 | 0.121 |
| False Omission Ratio | 1.562 | 1.848 | 1.819 | 1.882 | 2.019 | 1.852 |

Table 20. Results fairness pipeline – Random Forest

Source: Own work

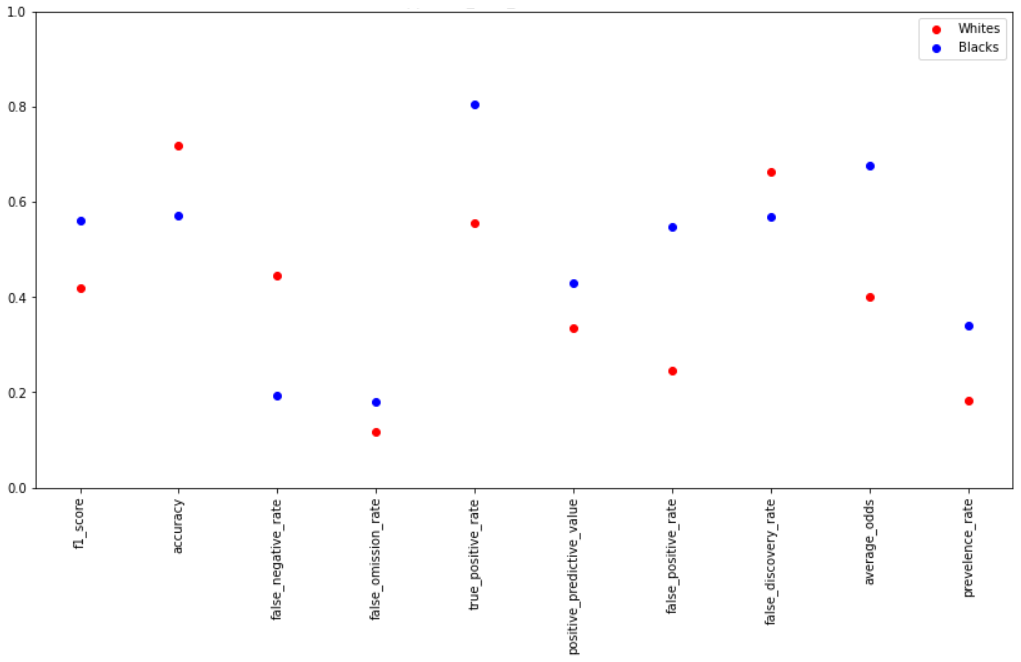


Figure 39. Discriminative model fairness – Random Forest

Source: Own work

In the following graph, it can be observed that in order to close the gaps between the sub-groups in accuracy, equalised odds increases the accuracy for black applicants while reducing the accuracy for white applicants at the same time. In this case, whites are being worse off in order to achieve fairness. This issue is highly discussed by Martinez et al. (2020) and Diana et al. (2020), where new algorithms are being proposed based on a new definition of fairness that relies on the minimax algorithm and Pareto optimality, where the idea is to minimise the maximum loss across all groups, instead of equalising the losses of the groups as in figure 40.

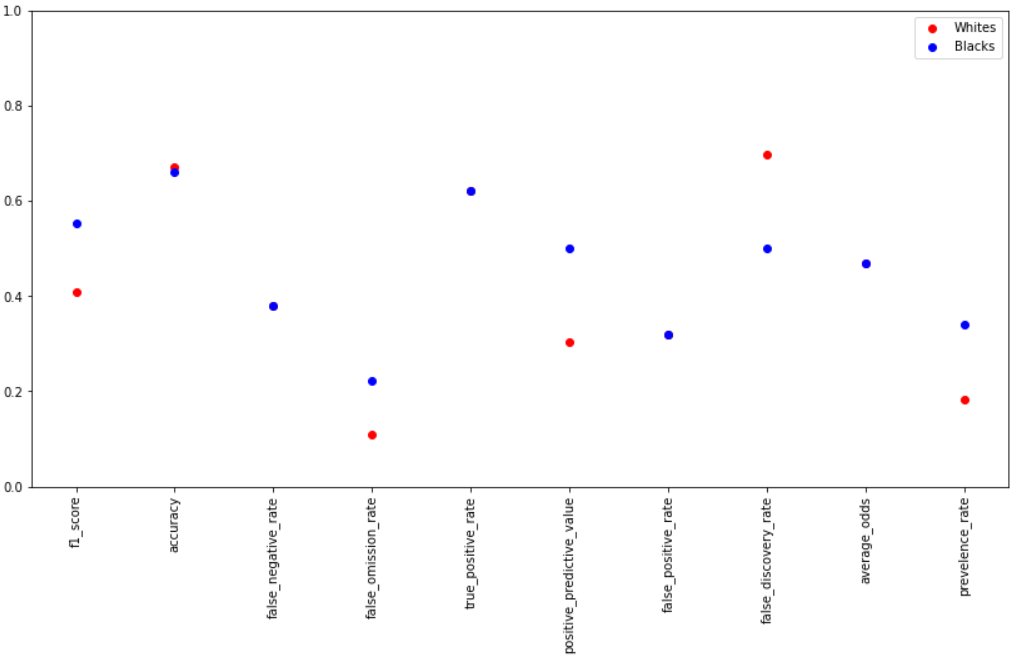


Figure 40. Discriminative model fairness – Random Forest

Source: Own work

Another important remark is that the order of feature importance of the variables changes with respect to the information value calculated in table 18. The Scikit-learn package in python calculates the feature importance of tree-based algorithms as the mean and standard deviation of the impurity accumulation decrease for each tree used in the algorithm. In figure 41, we can observe that race has lower feature importance compared to the rank of the information value calculated in table 18. For the equalised odds, the property type becomes the most relevant feature, rather than the income of the applicant according to the discriminative model.

|  |  |
| --- | --- |
| Discriminative model | Equalised Odds |
|  |  |

Figure 41. Variable importance model comparison – Random Forest

Source: Own work

Lastly, by comparing the confusion matrices as a heatmap between the predictions of the discriminative model and equalised odds predicted labels, we find out that equalised odds predictions increased type II error and improved the predictions of accepted loan applications.

|  |  |
| --- | --- |
| Discriminative model | Equalised Odds |
|  |  |

Figure 42. Confusion matrix model comparison – Random Forest

Source: Own work

## **6.3 Extreme Gradient Boosting**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Fairness Metrics** | **Discriminative model** | **Fair model** | **Pre-processing** | | **Post-processing** | |
| **Initial model** | **Blind model** | **Disparate Impact** | **Reweighting** | **Equalised Odds** | **Reject Option Classification** |
| Base rate | 0.364 | 0.374 | 0.381 | 0.365 | 0.298 | 0.374 |
| Disparate impact | 2.146 | 1.317 | 1.333 | 1.271 | 1.134 | 1.317 |
| Mean difference | 0.349 | 0.110 | 0.119 | 0.093 | 0.049 | 0.110 |
| SEDF | 3.295 | 3.181 | 3.105 | 3.134 | 9.814 | 3.181 |
| AOD | 0.289 | 0.062 | 0.070 | 0.045 | 0.000 | 0.062 |
| AAOD | 0.289 | 0.062 | 0.070 | 0.045 | 0.000 | 0.062 |
| EOD | 0.263 | 0.071 | 0.076 | 0.055 | -0.001 | 0.071 |
| Error rate | 0.308 | 0.316 | 0.321 | 0.311 | 0.307 | 0.316 |
| Error rate ratio | 1.532 | 1.086 | 1.092 | 1.071 | 1.031 | 1.086 |
| False Discovery Rate | 0.641 | 0.649 | 0.652 | 0.645 | 0.670 | 0.649 |
| False Discovery Ratio | 0.861 | 0.726 | 0.736 | 0.713 | 0.713 | 0.726 |
| False Omission Rate | 0.117 | 0.118 | 0.117 | 0.119 | 0.152 | 0.118 |
| False Omission Ratio | 1.455 | 1.830 | 1.824 | 1.864 | 2.023 | 1.830 |

Table 21. Results fairness pipeline – XGBoost

Source: Own work

## **6.4 Discussion of results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **AUC train** | **AUC test** | **Overfit?** | **Model Quality** |
| Logistic Regression Initial | 0.664 | 0.665 | FALSE | Poor |
| Logistic Regression Unawareness | 0.662 | 0.662 | FALSE | Poor |
| LR Disparate Impact | 0.661 | 0.662 | FALSE | Poor |
| LR Reweighting | 0.662 | 0.662 | FALSE | Poor |
| LR Equalised Odds | 0.662 | 0.662 | FALSE | Poor |
| LR Reject Option Classification | 0.662 | 0.662 | FALSE | Poor |
| Random Forest Initial | 0.704 | 0.662 | FALSE | Fair |
| Random Forest Unawareness | 0.687 | 0.662 | FALSE | Poor |
| RF Disparate Impact | 0.679 | 0.663 | FALSE | Poor |
| RF Reweighting | 0.686 | 0.661 | FALSE | Poor |
| RF Equalised Odds | 0.687 | 0.662 | FALSE | Poor |
| RF Reject Option Classification | 0.687 | 0.662 | FALSE | Poor |
| XGBoost Initial | 0.688 | 0.672 | FALSE | Poor |
| XGBoost Unawareness | 0.679 | 0.668 | FALSE | Poor |
| XGBoost Disparate Impact | 0.676 | 0.667 | FALSE | Poor |
| XGBoost Reweighting | 0.679 | 0.667 | FALSE | Poor |
| XGBoost Equalised Odds | 0.679 | 0.668 | FALSE | Poor |
| XGBoost Reject Option Classification | 0.679 | 0.668 | FALSE | Poor |

Table 22. Comparison of model performance

Source: Own work

# **Conclusions**

* Modelling for a larger time frame using the available data
* Taking into consideration other protect variables : race and ethnicity

# **References**

Abdou, H. A., & Pointon, J. (2011). Credit Scoring, Statistical Techniques and Evaluation Criteria: A Review of the Literature. *Intelligent Systems in Accounting, Finance and Management*, *18*(2–3), 59–88. https://doi.org/10.1002/ISAF.325

Agresti, A. (2002). *Categorical Data Analysis | Wiley Series in Probability and Statistics* (Second Edition). John Wiley & Sons, Inc. https://onlinelibrary.wiley.com/doi/book/10.1002/0471249688

Agresti, A., & Finlay, Barbaracoaut. (1997). *Statistical methods for the social sciences*. Pearson Prentice Hall.

Aguais, S. D., Forest, L. R., Wong, E. Y. L., & Diaz-Ledezma, D. (2004). Point-in-Time versus Through-the-Cycle Ratings. *Z-Risk Engine*, 29. https://www.z-riskengine.com/media/ylrpyd2z/point-in-time-versus-through-the-cycle-ratings.pdf

Anderson, Raymond. (2007). *The credit scoring toolkit : theory and practice for retail credit risk management and decision automation*. Oxford University Press.

Bank for International Settlements. (2005, January 1). *About BIS - overview*. https://www.bis.org/about/index.htm

Bellamy, R. K. E., Dey, K., Hind, M., Hoffman, S. C., Houde, S., Kannan, K., Lohia, P., Martino, J., Mehta, S., Mojsilovic, A., Nagar, S., Ramamurthy, K. N., Richards, J., Saha, D., Sattigeri, P., Singh, M., Varshney, K. R., & Zhang, Y. (2018). AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. *ArXiv*, *63*(4), 1–15. https://doi.org/10.48550/arxiv.1810.01943

Bellini, T. (2017). *Stress testing and risk integration in banks : a statistical framework and practical software guide (in MATLAB and R)*. Academic Press. https://www.elsevier.com/books/stress-testing-and-risk-integration-in-banks/bellini/978-0-12-803590-0

Berk, R., Heidari, H., Jabbari, S., Kearns, M., & Roth, A. (2017). Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociological Methods and Research*, *50*(1), 3–44. https://doi.org/10.48550/arxiv.1703.09207

Bhukya, D. P., & Ramachandram, S. (2010). Decision Tree Induction: An Approach for Data Classification Using AVL-Tree. *International Journal of Computer and Electrical Engineering*, 660–665. https://doi.org/10.7763/IJCEE.2010.V2.208

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A.-L., Deng, D., & Lindauer, M. (2021). *Hyperparameter Optimization: Foundations, Algorithms, Best Practices and Open Challenges*. https://doi.org/10.48550/arxiv.2107.05847

Breiman, L. (2001). Random Forests. *Machine Learning 2001 45:1*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and regression trees. *Classification and Regression Trees*, 1–358. https://doi.org/10.1201/9781315139470/CLASSIFICATION-REGRESSION-TREES-LEO-BREIMAN-JEROME-FRIEDMAN-RICHARD-OLSHEN-CHARLES-STONE

Briscoe, E., & Feldman, J. (2011). Conceptual complexity and the bias/variance tradeoff. *Cognition*, *118*(1), 2–16. https://doi.org/10.1016/J.COGNITION.2010.10.004

Browne, I. (1997). Explaining the Black-White gap in labor force participation among women heading households. *American Sociological Review*, *62*(2), 236–252. https://doi.org/10.2307/2657302

Brownlee, J. (2016). *Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch* (J. Brownlee, Ed.; Edition, v1.12). https://machinelearningmastery.com/master-machine-learning-algorithms/

Caeyers, B., & Marcel, F. (2020). Exclusion Bias in the Estimation of Peer Effects. In *NBER working papers* (NBER Working Paper). https://www.nber.org/system/files/working\_papers/w22565/w22565.pdf

Carroll, J. Douglas., Green, P. E., & Chaturvedi, A. (1997). *Mathematical tools for applied multivariate analysis*. Academic Press. https://www.elsevier.com/books/mathematical-tools-for-applied-multivariate-analysis/chaturvedi/978-0-08-091723-8

Casellina, S., Landini, S., & Uberti, M. (2021). The estimation risk and the IRB supervisory formula. In *EBA STAFF PAPER SERIES*. 10.2853/032235

Chen, M., Liu, Q., Chen, S., Liu, Y., Zhang, C. H., & Liu, R. (2019). XGBoost-Based Algorithm Interpretation and Application on Post-Fault Transient Stability Status Prediction of Power System. *IEEE Access*, *7*, 13149–13158. https://doi.org/10.1109/ACCESS.2019.2893448

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. https://doi.org/10.1145/2939672

Cohen, J., Cohen, P., West, S., & Aiken, L. (1975). Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences. In *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. Routledge. https://doi.org/10.4324/9780203774441

Consumer Financial Protection Bureau. (2022). *12 CFR Part 1002 (Regulation B) | 1002.6 Rules concerning evaluation of applications.* . https://www.consumerfinance.gov/rules-policy/regulations/1002/6/

Cramer, J. S. (2002). The Origins of Logistic Regression. *SSRN Electronic Journal*, *2002-119/4*, 0–16. https://doi.org/10.2139/SSRN.360300

Demajo, L. M., Vella, V., & Dingli, A. (2020). *Explainable AI for Interpretable Credit Scoring*. 185–203. https://doi.org/10.5121/csit.2020.101516

DeMaris, A. (1993). Odds versus Probabilities in Logit Equations: A Reply to Roncek. *Social Forces*, *71*(4), 1057. https://doi.org/10.2307/2580130

DeMaris, A., Teachman, J., & Morgan, S. P. (1990). Interpreting Logistic Regression Results: A Critical Commentary. *Journal of Marriage and the Family*, *52*(1), 271. https://doi.org/10.2307/352857

Diana, E., Gill, W., Kearns, M., Kenthapadi, K., & Roth, A. (2020). Minimax Group Fairness: Algorithms and Experiments. *AIES 2021 - Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 66–76. https://doi.org/10.48550/arxiv.2011.03108

Dobbie, W., Liberman, A., Paravisini, D., Pathania, V., Battaglia, E., Gandre, N., Grogan, J., Palmer, B., & Reeves, J. (2018). Measuring Bias in Consumer Lending. *National Bureau of Economic Research*. https://doi.org/10.3386/W24953

Guidelines on PD estimation, LGD estimation and the treatment of defaulted exposures, Pub. L. No. EBA/GL/2017/16 (2017). https://www.eba.europa.eu/-/eba-sets-out-roadmap-for-the-implementation-of-the-regulatory-review-of-internal-

European Banking Authority. (2022, January 25). *The Single Rulebook | European Banking Authority*. https://www.eba.europa.eu/regulation-and-policy/single-rulebook

European Central Bank. (2003). *Regulation (EC) No 1745/2003 of the European Central Bank of 12 September 2003 on the application of minimum reserves (ECB/2003/9)*. https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:32003R1745

European Central Bank. (2015, June 19). *ECB mission*. https://www.ecb.europa.eu/ecb/orga/escb/ecb-mission/html/index.en.html

European Commission. (2019). *Ethics guidelines for trustworthy AI*. Directorate-General for Communications Networks, Content and Technology. https://data.europa.eu/doi/10.2759/346720

European Parliament. (2013a). *Regulation (EU) No 575/2013 of the European Parliament and of the Council of 26 June 2013 on prudential requirements for credit institutions and investment firms and amending Regulation (EU) No 648/2012*. https://eur-lex.europa.eu/legal-content/EN/LSU/?uri=celex:32013R0575

European Parliament. (2013b, June 27). *Directive 2013/36/EU of the European Parliament and of the Council of 26 June 2013 on access to the activity of credit institutions and the prudential supervision of credit institutions and investment firms, amending Directive 2002/87/EC and repealing Directives 2006/48/EC and 2006/49/EC*. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32013L0036

*EU General Data Protection Regulation (GDPR): Regulation (EU) 2016/679*, (2016) (testimony of European Parliament). https://gdpr-info.eu/

Eurostat. (2022). *Contributions of each sector - Institutional sector accounts* . https://ec.europa.eu/eurostat/web/sector-accounts/detailed-charts/contributions-sectors

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., & Venkatasubramanian, S. (2014). Certifying and removing disparate impact. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *2015-August*, 259–268. https://doi.org/10.48550/arxiv.1412.3756

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*(2), 179–188. https://doi.org/10.1111/J.1469-1809.1936.TB02137.X

Foulds, J. R., Islam, R., Keya, K. N., & Pan, S. (2019). An Intersectional Definition of Fairness. *Proceedings - International Conference on Data Engineering*, *2020-April*, 1918–1921. https://doi.org/10.48550/arxiv.1807.08362

Fox, J. (1997). *Applied regression analysis, linear models, and related methods.* Sage Publications, Inc. https://psycnet.apa.org/record/1997-08857-000

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Https://Doi.Org/10.1214/Aos/1013203451*, *29*(5), 1189–1232. https://doi.org/10.1214/AOS/1013203451

Furletti, M. (2002). An Overview and History of Credit Reporting. *Payment Cards Center of the FRB of Philadelphia*.

Gillis, T. B. (2021). The Input Fallacy. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.3571266

Hanck, C., Arnold, M., Gerber, A., & Schmelzer, M. (2021). *Introduction to Econometrics with R*. https://www.econometrics-with-r.org/

Handy, C. (2002). What is a Business for? *Harvard Business Review*, 9. https://ssrn.com/abstract=932676

Hartshorn, S. (2016). *Machine Learning With Random Forests And Decision Trees: A Visual Guide For Beginners*. https://www.amazon.es/Machine-Learning-Random-Forests-Decision-ebook/dp/B01JBL8YVK

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer New York. https://doi.org/10.1007/978-0-387-84858-7

Heiss, A. (2021, August 21). *Exploring R2 and regression variance with Euler/Venn diagrams*. https://www.andrewheiss.com/blog/2021/08/21/r2-euler/#ref-CohenCohen:1975

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (1989). Applied Logistic Regression. In *Wiley Series in Probability and Statistics* (3rd Edition). Wiley. https://www.wiley.com/en-ca/Applied+Logistic+Regression%2C+3rd+Edition-p-9780470582473

Ip, E. H. S. (2017). Visualizing Multiple Regression. *Https://Doi.Org/10.1080/10691898.2001.11910646*, *9*(1). https://doi.org/10.1080/10691898.2001.11910646

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning* (2nd ed.). Springer US. https://doi.org/10.1007/978-1-0716-1418-1

Jickling, M. (2009). Causes of the Financial Crisis. In *Congressional Research Service*. https://ecommons.cornell.edu/handle/1813/77536

Johnson, R. A., & Wichern, D. W. (2002). *Applied multivariate statistical analysis* (5th ed.). Prentice Hall.

Joseph, M., Kearns, M., Morgenstern, J. H., Neel, S., & Roth, A. (2016). *Rawlsian Fairness for Machine Learning*. 27. https://www.semanticscholar.org/paper/Rawlsian-Fairness-for-Machine-Learning-Joseph-Kearns/2d55ff4542eaae18dcd10c2cd74199396e260402

Kamiran, F., & Calders, T. (2011). Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems 2011 33:1*, *33*(1), 1–33. https://doi.org/10.1007/S10115-011-0463-8

Kantardzic, M. (2019). *Data mining : concepts, models, methods, and algorithms* (3rd ed.). Wiley-IEEE Press.

Kass, G. v. (1980). An Exploratory Technique for Investigating Large Quantities of Categorical Data. *Applied Statistics*, *29*(2), 119. https://doi.org/10.2307/2986296

Kaszyński, D., Kamiński, B., Szapiro, T., Kwiatkowski, M., Przanowski, K., Zając, S., Opiński, Ł., Wrzosek, M., Siuta, K., Cerazy, K., Chlebus, M., Kłosok, M., Biecek, P., Kraiński, Ł., & Nosarzewski, A. (2021). *Credit Scoring in Context of Interpretable Machine Learning* (D. Kaszyński, B. Kamiński, & T. Szapiro, Eds.). SGH Publishing House. https://doi.org/10.5281/ZENODO.4692106

Kennedy, K. (2013). Credit Scoring Using Machine Learning. *Doctoral*. https://doi.org/https://doi.org/10.21427/D7NC7J

Knutson, M. (2020). *Credit Scoring Approaches Guidelines*. The World Bank Group. https://thedocs.worldbank.org/en/doc/935891585869698451-0130022020/CREDIT-SCORING-APPROACHES-GUIDELINES-FINAL-WEB

Kozodoi, N., Jacob, J., & Lessmann, S. (2021). Fairness in Credit Scoring: Assessment, Implementation and Profit Implications. *European Journal of Operational Research*, *297*(3), 1083–1094. https://doi.org/10.1016/j.ejor.2021.06.023

Kumar, R. (2014). Regulatory Environment of Financial Institutions. *Strategies of Banks and Other Financial Institutions*, 31–60. https://doi.org/10.1016/B978-0-12-416997-5.00002-6

Kusner, M., Loftus, J., Russell, C., & Silva, R. (2017). Counterfactual Fairness. *NeurIPS Proceedings*, 11. https://proceedings.neurips.cc/paper/2017/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf

Langdon, R. J., Yousefi, P. D., Relton, C. L., & Suderman, M. J. (1992). The Degradation of the Scorecard over the Business Cycle. *IMA Journal of Mathematics Applied in Business and Industry*, *4*, 497–509. https://doi.org/10.2/JQUERY.MIN.JS

Lauer, J. (2017). *Creditworthy : a history of consumer surveillance and financial identity in America*. Columbia University Press.

Lautenschläger, S. (2017). *European banking supervision, global cooperation and challenges for banks*. https://www.bankingsupervision.europa.eu/press/speeches/date/2017/html/ssm.sp170518\_slides.en.pdf

Lee, J., & Hogarth, J. M. (2018). The Price of Money: Consumers’ Understanding of APRs and Contract Interest Rates: *Https://Doi.Org/10.1177/074391569901800108*, *18*(1), 66–76. https://doi.org/10.1177/074391569901800108

Leightner, J. E., & Inoue, T. (2012). Solving the omitted variables problem of regression analysis using the relative vertical position of observations. *Advances in Decision Sciences*, *2012*. https://doi.org/10.1155/2012/728980

Lewis, E. (1992). *An Introduction to Credit Scoring* (Reprint edition). Athena Press.

Liu, Y. (2019). Margin of Conservatism Framework for IRB PD, LGD and CCF. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.3258825

Loh, W.-Y., & Shih, Y.-S. (1997). Split Selection Methods For Classification Trees. *Statistica Sinica*, *7*, 815–840.

Lohia, P. K., Natesan Ramamurthy, K., Bhide, M., Saha, D., Varshney, K. R., & Puri, R. (2018). Bias Mitigation Post-processing for Individual and Group Fairness. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, *2019-May*, 2847–2851. https://doi.org/10.48550/arxiv.1812.06135

Long, J. S., & Freese, J. (2014). *Regression Models for Categorical Dependent Variables Using Stata* (Third Edition). https://www.stata.com/bookstore/regression-models-categorical-dependent-variables/

Looney, M. (2013). Capital Requirements Directive (CRD) IV. *CFA Institute*. https://www.cfainstitute.org/-/media/documents/issue-brief/crd-iv-issue-brief-final.pdf

Louppe, G., Wehenkel, L., Sutera, A., & Geurts, P. (2013). *Understanding variable importances in Forests of randomized trees*. https://www.researchgate.net/publication/264046801\_Understanding\_variable\_importances\_in\_Forests\_of\_randomized\_trees

MacDonald, S. B., & Gastmann, A. L. (2017). A History of Credit & Power in the Western World. *A History of Credit and Power in the Western World*, 1–314. https://doi.org/10.4324/9781315083513

Makhlouf, K., Zhioua, S., & Palamidessi, C. (2022). Machine learning fairness notions: Bridging the gap with real-world applications. *Information Processing and Management*, *58*(5). https://doi.org/10.1016/j.ipm.2021.102642

Marquardt, D. W., & Snee, R. D. (1975). Ridge regression in practice. *American Statistician*, *29*(1), 3–20. https://doi.org/10.1080/00031305.1975.10479105

Martin, V., & Evzen, K. (2006). Credit-Scoring Methods. *Czech Journal of Economics and Finance (Finance a Uver)*, *56*(3–4), 152–167. https://EconPapers.repec.org/RePEc:fau:fauart:v:56:y:2006:i:3-4:p:152-167

Martinez, N., Bertran, M., & Sapiro, G. (2020). Minimax Pareto Fairness: A Multi Objective Perspective. *37th International Conference on Machine Learning, ICML 2020*, *PartF168147-9*, 6711–6720. https://doi.org/10.48550/arxiv.2011.01821

Mayson, S. (2019). Yale Law Journal - Bias In, Bias Out. *The Yale Law Journal*, *128*(8). https://www.yalelawjournal.org/article/bias-in-bias-out

Mohanty, N., John, A. L. S., Manmatha, R., & Rath, T. M. (2013). Shape-Based Image Classification and Retrieval. *Handbook of Statistics*, *31*, 249–267. https://doi.org/10.1016/B978-0-444-53859-8.00010-2

Molnar, C. (2022, March 29). *Interpretable Machine Learning*. https://christophm.github.io/interpretable-ml-book/index.html

Mood, C. (2010). Logistic Regression: Why We Cannot Do What We Think We Can Do, and What We Can Do About It. *European Sociological Review*, *26*(1), 67–82. https://doi.org/10.1093/ESR/JCP006

Neuhaus, J. M., & Jewell, N. P. (1993). A Geometric Approach to Assess Bias Due to Omitted Covariates in Generalized Linear Models. *Biometrika*, *80*(4), 807. https://doi.org/10.2307/2336872

O’Brien, R. M. (2007). A Caution Regarding Rules of Thumb for Variance Inflation Factors. *Quality & Quantity 2007 41:5*, *41*(5), 673–690. https://doi.org/10.1007/S11135-006-9018-6

Onay, C., & Öztürk, E. (2018). A review of credit scoring research in the age of Big Data. *Journal of Financial Regulation and Compliance*, *26*(3), 382–405. https://doi.org/10.1108/JFRC-06-2017-0054

Page, L. A., & Henderson, M. (2008). Appraising the evidence: what is measurement bias? *Evidence-Based Mental Health*, *11*(2), 36–37. https://doi.org/10.1136/EBMH.11.2.36

Pampel, F. C. (2000). *Logistic regression : A primer*. Sage Publications.

Patel, N., Patel, S. P. B., & Prof, A. (2012). Study of Various Decision Tree Pruning Methods with their Empirical Comparison in WEKA Saurabh Upadhyay. *International Journal of Computer Applications*, *60*(12), 975–8887.

Puertas, R., Olmeda, I., & Bonilla, M. (2003). *Parametric and non-parametric models in the credit scoring problems*. https://www.researchgate.net/publication/289978389\_Parametric\_and\_non-parametric\_models\_in\_the\_credit\_scoring\_problems

Puntanen, S., & Styan, G. P. H. (1989). The equality of the ordinary least squares estimator and the best linear unbiased estimator. *American Statistician*, *43*(3), 153–161. https://doi.org/10.1080/00031305.1989.10475644

Quinlan, J. R. (1994). C4.5: Programs for Machine Learning. *Machine Learning*, *16*(3), 235–240. https://doi.org/10.1007/BF00993309

Quinlan, J. R. (2006). Bagging, Boosting, and C4.5. *University of Sydney*.

Rao, C. R. (1976). Estimation of Parameters in a Linear Model. *Https://Doi.Org/10.1214/Aos/1176343639*, *4*(6), 1023–1037. https://doi.org/10.1214/AOS/1176343639

Rawls, J. (1971). A Theory of Justice. In *Harvard University Press*. Harvard University Press. https://doi.org/10.2307/J.CTVJF9Z6V

Rokach, L., & Maimon, O. (2015). *Data Mining with Decision Trees: Theory and Applications* (2nd ed.). World Scientific Publishing Co. Pte. Ltd. http://www.worldscientific.com/series/smpai

Ronaghan, S. (2019, April 22). *AI Fairness — Explanation of Disparate Impact Remover*. Towards Data Science. https://towardsdatascience.com/ai-fairness-explanation-of-disparate-impact-remover-ce0da59451f1

Schaffer, C. (1991). When does overfitting decrease prediction accuracy in induced decision trees and rule sets? *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *482 LNAI*, 192–204. https://doi.org/10.1007/BFB0017014/COVER/

Shah, K., Gupta, P., Deshpande, A., & Bhattacharyya, C. (2021). *Rawlsian Fair Adaptation of Deep Learning Classifiers*. https://arxiv.org/pdf/2105.14890.pdf

Siddiqi, N. (2017). Intelligent Credit Scoring. *Intelligent Credit Scoring*. https://doi.org/10.1002/9781119282396

Siddiqi, Naeem. (2013). *Credit risk scorecards : developing and implementing intelligent credit scoring*. https://books.google.com/books/about/Credit\_Risk\_Scorecards.html?hl=fr&id=SEbCeN3-kEUC

Skjelbred, S.-E. (2015). Lecture 7&8 March 9 Multiple regression model. *Class Presentation: ECON3150/4150 Spring 2015*, 60. https://www.uio.no/studier/emner/sv/oekonomi/ECON4150/v15/lectures/lec-7-v2.pdf

Song, Y. Y., & Lu, Y. (2015). Decision tree methods: applications for classification and prediction. *Shanghai Archives of Psychiatry*, *27*(2), 130. https://doi.org/10.11919/J.ISSN.1002-0829.215044

Stoltzfus, J. C. (2011). Logistic regression: a brief primer. *Academic Emergency Medicine : Official Journal of the Society for Academic Emergency Medicine*, *18*(10), 1099–1104. https://doi.org/10.1111/J.1553-2712.2011.01185.X

Sullivan, W. (2017). Machine Learning For Beginners Guide Algorithms. In *Healthy Pragmatic Solut-ions Inc*. Healthy Pragmatic Solutions Inc. https://books.google.com.co/books?id=v6saxAEACAAJ&dq=Machine+Learning+For+Beginners+Guide+Algorithms+Decision+Tree+%26+Random+Forest+Introduction&hl=es&sa=X&ved=2ahUKEwjr3JLeiN\_qAhUQUa0KHStSAKoQ6AEwAHoECAEQAg

Thomas, L., Crook, J., & Edelman, D. (2017). Credit Scoring and Its Applications, Second Edition. *Credit Scoring and Its Applications, Second Edition*. https://doi.org/10.1137/1.9781611974560

Civil Rights Act, Pub. L. No. 42 U.S.C., Title VII (1964). https://www.eeoc.gov/statutes/title-vii-civil-rights-act-1964

Venn, J. (1880). On the diagrammatic and mechanical representation of propositions and reasonings. *Https://Doi.Org/10.1080/14786448008626877*, *10*(59), 1–18. https://doi.org/10.1080/14786448008626877

Verkoulen, C., Tromp, S., & Cordes, D. (2022). *Quantifying Bias and Fairness in Credit Scoring Algorithms*.

Widyaningsih, P., Sari Saputro, D. R., & Putri, A. N. (2017). Fisher Scoring Method for Parameter Estimation of Geographically Weighted Ordinal Logistic Regression (GWOLR) Model. *Journal of Physics: Conference Series*, *855*(1). https://doi.org/10.1088/1742-6596/855/1/012060

World Bank Group. (2019). Disruptive Technologies in the Credit Information Sharing Industry. *Disruptive Technologies in the Credit Information Sharing Industry*. https://doi.org/10.1596/31714

Yang, X., & Wen, W. (2018). Ridge and Lasso Regression Models for Cross-Version Defect Prediction. *IEEE Transactions on Reliability*, *67*(3), 885–896. https://doi.org/10.1109/TR.2018.2847353

Zhou, Q., Chen, W., Song, S., Gardner, J. R., Weinberger, K. Q., & Chen, Y. (2014). A Reduction of the Elastic Net to Support Vector Machines with an Application to GPU Computing. *Proceedings of the National Conference on Artificial Intelligence*, *4*, 3210–3216. https://arxiv.org/abs/1409.1976v1

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320. https://doi.org/10.1111/J.1467-9868.2005.00503.X

# **Appendices**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Fairness Metrics** | **Discriminative model** | **Fair model** | **Pre-processing** | | **Post-processing** | |
|  | **Initial model** | **Blind model** | **Disparate Impact** | **Reweighting** | **Equalised Odds** | **Reject Option Classification** |
|  | Accuracy overall | 0.650 | 0.653 | 0.649 | 0.658 | 0.654 | 0.653 |
|  | f1-score overall | 0.447 | 0.444 | 0.444 | 0.445 | 0.380 | 0.444 |
| **Favoured group** | f1-score | 0.412 | 0.411 | 0.410 | 0.412 | 0.397 | 0.411 |
| Accuracy | 0.673 | 0.662 | 0.651 | 0.667 | 0.626 | 0.662 |
| False negative rate | 0.370 | 0.352 | 0.334 | 0.359 | 0.325 | 0.352 |
| False omission rate | 0.108 | 0.105 | 0.103 | 0.106 | 0.105 | 0.105 |
| True positive rate | 0.630 | 0.648 | 0.666 | 0.641 | 0.675 | 0.648 |
| Positive predictive value | 0.306 | 0.301 | 0.297 | 0.304 | 0.281 | 0.301 |
| False positive rate | 0.318 | 0.335 | 0.352 | 0.327 | 0.384 | 0.335 |
| False discovery rate | 0.694 | 0.699 | 0.703 | 0.696 | 0.719 | 0.699 |
| Average odds | 0.474 | 0.492 | 0.509 | 0.484 | 0.530 | 0.492 |
| Prevelence rate | 0.182 | 0.182 | 0.182 | 0.182 | 0.182 | 0.182 |
| **Deprived group** | f1-score | 0.557 | 0.574 | 0.575 | 0.576 | 0.555 | 0.574 |
| Accuracy | 0.524 | 0.634 | 0.625 | 0.640 | 0.634 | 0.634 |
| False negative rate | 0.118 | 0.272 | 0.252 | 0.280 | 0.327 | 0.272 |
| False omission rate | 0.152 | 0.193 | 0.187 | 0.194 | 0.215 | 0.193 |
| True positive rate | 0.882 | 0.728 | 0.748 | 0.720 | 0.673 | 0.728 |
| Positive predictive value | 0.407 | 0.474 | 0.467 | 0.479 | 0.473 | 0.474 |
| False positive rate | 0.660 | 0.415 | 0.439 | 0.402 | 0.386 | 0.415 |
| False discovery rate | 0.593 | 0.526 | 0.533 | 0.521 | 0.527 | 0.526 |
| Average odds | 0.771 | 0.571 | 0.593 | 0.561 | 0.529 | 0.571 |
| Prevelence rate | 0.339 | 0.339 | 0.339 | 0.339 | 0.339 | 0.339 |

Table 23. Model performance (sub-groups) – Logistic Regression

Source: Own work

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Fairness Metrics** | **Discriminative model** | **Fair model** | **Pre-processing** | | **Post-processing** | |
|  | **Initial model** | **Blind model** | **Disparate Impact** | **Reweighting** | **Equalised Odds** | **Reject Option Classification** |
|  | Accuracy overall | 0.690 | 0.683 | 0.679 | 0.681 | 0.688 | 0.682 |
|  | f1-score overall | 0.448 | 0.448 | 0.449 | 0.447 | 0.385 | 0.447 |
| **Favoured group** | f1-score | 0.419 | 0.418 | 0.418 | 0.417 | 0.407 | 0.418 |
| Accuracy | 0.719 | 0.695 | 0.686 | 0.690 | 0.671 | 0.693 |
| False negative rate | 0.445 | 0.398 | 0.382 | 0.391 | 0.379 | 0.396 |
| False omission rate | 0.116 | 0.110 | 0.108 | 0.110 | 0.110 | 0.110 |
| True positive rate | 0.555 | 0.602 | 0.618 | 0.609 | 0.621 | 0.604 |
| Positive predictive value | 0.336 | 0.321 | 0.315 | 0.317 | 0.303 | 0.320 |
| False positive rate | 0.245 | 0.284 | 0.299 | 0.292 | 0.318 | 0.287 |
| False discovery rate | 0.664 | 0.679 | 0.685 | 0.683 | 0.697 | 0.680 |
| Average odds | 0.400 | 0.443 | 0.458 | 0.450 | 0.470 | 0.445 |
| Prevelence rate | 0.182 | 0.182 | 0.182 | 0.182 | 0.182 | 0.182 |
| **Deprived group** | f1-score | 0.561 | 0.575 | 0.580 | 0.574 | 0.554 | 0.574 |
| Accuracy | 0.572 | 0.661 | 0.658 | 0.669 | 0.661 | 0.660 |
| False negative rate | 0.195 | 0.326 | 0.305 | 0.341 | 0.380 | 0.326 |
| False omission rate | 0.181 | 0.204 | 0.197 | 0.207 | 0.223 | 0.204 |
| True positive rate | 0.805 | 0.674 | 0.695 | 0.659 | 0.620 | 0.674 |
| Positive predictive value | 0.430 | 0.501 | 0.497 | 0.509 | 0.500 | 0.500 |
| False positive rate | 0.549 | 0.345 | 0.361 | 0.326 | 0.319 | 0.347 |
| False discovery rate | 0.570 | 0.499 | 0.503 | 0.491 | 0.500 | 0.500 |
| Average odds | 0.677 | 0.509 | 0.528 | 0.493 | 0.470 | 0.510 |
| Prevelence rate | 0.339 | 0.339 | 0.339 | 0.339 | 0.339 | 0.339 |

Table 24. Model performance (sub-groups) – Random Forest

Source: Own work

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Fairness Metrics** | **Discriminative model** | **Fair model** | **Pre-processing** | | **Post-processing** | |
|  | **Initial model** | **Blind model** | **Disparate Impact** | **Reweighting** | **Equalised Odds** | **Reject Option Classification** |
|  | Accuracy overall | 0.692 | 0.684 | 0.679 | 0.689 | 0.693 | 0.684 |
|  | f1-score overall | 0.459 | 0.454 | 0.453 | 0.454 | 0.391 | 0.454 |
| **Favoured group** | f1-score | 0.426 | 0.422 | 0.420 | 0.423 | 0.414 | 0.422 |
| Accuracy | 0.721 | 0.694 | 0.687 | 0.698 | 0.677 | 0.694 |
| False negative rate | 0.431 | 0.387 | 0.377 | 0.392 | 0.374 | 0.387 |
| False omission rate | 0.113 | 0.108 | 0.107 | 0.109 | 0.108 | 0.108 |
| True positive rate | 0.569 | 0.613 | 0.623 | 0.608 | 0.626 | 0.613 |
| Positive predictive value | 0.341 | 0.322 | 0.317 | 0.325 | 0.309 | 0.322 |
| False positive rate | 0.245 | 0.288 | 0.299 | 0.281 | 0.312 | 0.288 |
| False discovery rate | 0.659 | 0.678 | 0.683 | 0.675 | 0.691 | 0.678 |
| Average odds | 0.407 | 0.450 | 0.461 | 0.445 | 0.469 | 0.450 |
| Prevelence rate | 0.182 | 0.182 | 0.182 | 0.182 | 0.182 | 0.182 |
| **Deprived group** | f1-score | 0.569 | 0.583 | 0.581 | 0.582 | 0.560 | 0.583 |
| Accuracy | 0.572 | 0.668 | 0.658 | 0.677 | 0.667 | 0.668 |
| False negative rate | 0.168 | 0.316 | 0.301 | 0.338 | 0.374 | 0.316 |
| False omission rate | 0.164 | 0.198 | 0.195 | 0.202 | 0.218 | 0.198 |
| True positive rate | 0.832 | 0.684 | 0.699 | 0.662 | 0.626 | 0.684 |
| Positive predictive value | 0.432 | 0.508 | 0.497 | 0.519 | 0.508 | 0.508 |
| False positive rate | 0.561 | 0.340 | 0.363 | 0.316 | 0.312 | 0.340 |
| False discovery rate | 0.568 | 0.492 | 0.503 | 0.481 | 0.492 | 0.492 |
| Average odds | 0.696 | 0.512 | 0.531 | 0.489 | 0.469 | 0.512 |
| Prevelence rate | 0.339 | 0.339 | 0.339 | 0.339 | 0.339 | 0.339 |

Table 25. Model performance (sub-groups) – XGBoost

Source: Own work