Politecnico di Torino

Master Degree in Data science and engineering
A.y. 2021/2022
Graduation period July 2022

Dynamic identification of risk thresholds for balance measures in machine learning

Supervisors

Prof. Antonio Vetrò

Dott. Mariachiara Mecati

Candidate

Andrea Adrignola

Abstract

Automated decision-making systems (ADM) may significantly affect our everyday life. They can assist us in a number of tasks when used as a reference, or even substitute humans entirely, and they are as much an opportunity to challenge our decision-making processes as they are a mean of reinforcing pre-existing biases. Because the data used to train the algorithms could (and usually do) encode social biases.

For this reason, one of the main approaches to mitigate bias in such a framework is to work on data quality. To assess how the quality of the data affects the outcome of a classification, we made use of two different set of indices, balance measures and fairness measures, relating to different stages of a machine learning pipeline. Balance measures assess the proportions of classes of a given sensitive attribute (training set), while fairness measures evaluate the fairness of the outcome, in our case a classification (test set), with respect to the same attribute. In our study we take into account both binary and multiclass attributes.

The aim of the study was to evaluate the feasability of thresholds for both balance and fairness measures, such that if the balance is over its threshold, then we can be assume that also the fairness is over its threshold (in our case we compute the unfairness, so we want it to be under a certain value). In other words, we want to anticipate an incoming bias, estimating the fairness of the classification looking at the balance of the data. To obtain a sufficient amount of instances, we created a large amount of synthetic versions of numerous datasets, with different levels of balance to see how it would affect the fairness of the outcome. To further generalize the study, we included different algorithms.

We created thresholds separately for each combination of balance-fairness-algorithm, taking into account not only the point of view on how balance and fairness should be evaluated (different measures encode different points of view), but also the specific way data are processed. To measure the goodness of the thesholds, we selected a variety of sensitivity measures.

Keywords: Automated decision making, Data ethics, Data quality, Data bias, Algorithm fairness

Table of Contents

1 Introduction 1

2 Background 3

3 Experimental design 5

3.1 Balance Measures 7

3.2 Unfairness Measures 9

3.3 Datasets 11

3.4 Algorithms 15

3.5 Mutation techniques 15

3.6 Sensitivity measures 16

4 Correlation analysis 19

4.1 Binary attributes 20

4.2 Multiclass attributes 22

5 Sensitivity analysis 24

5.1 Unfairness analysis 24

5.2 Method 25

5.3 Configurations 27

6 Results and discussion 30

6.1 Binary attributes 31

6.1.1 Sensitivity measures 31

6.1.2 Thresholds 32

6.1.3 Accuracy correlation 33

6.2 Multiclass attributes 36

6.2.1 Sensitivity measures 36

6.2.2 Thresholds 37

6.2.3 Accuracy correlation 38

7 Conclusions and future work 41

A Predictors and targets 43
A.1 Credit card default 43
A.2 Statlog 43
A.3 Student (Math and Portugese) 44
A.4 Drug consumption (Cannabis) 45
A.5 Drug consumption (Impulsive) 46
A.6 Heart disease 47

B Thresholds tables 48
B.1 Binary attributes 48
B.1.1 Gini 48
B.1.2 Shannon 49
B.1.3 Simpson 50
B.1.4 IR 51
B.2 Multiclass attributes 53
B.2.1 Gini 53
B.2.2 Shannon 54
B.2.3 Simpson 55
B.2.4 IR 56

Bibliography 58

Chapter 1

Introduction

The process of automating the decision process is becoming more and more prominent as algorithms are developed and data become increasingly available in our society [1]. From support to human decision-making to fully automated systems, today technology supposedly permits us to rationalize our thinking on the basis of the evidence we possess.

This is a great opportunity, because we can decide how to process data so that the decisional process is mostly clear and understandable, unlike the processes that happen in our minds that may lead to uncomprehensible decisions. Unfortunately, the data on which algorithms are trained could encode all kind of biases, leading to decisions no different than those of biased individuals [2]. However, if an automated decision system make biased decisions it could be even worse, because the use of an algorithm can lead to the illusion that everything is fair and square: from this understanding comes the focus on data quality, knowing that data should not mindlessly be taken as evidence but processed in such a way that assures a fair outcome, which is supposedly why we want to use algorithms in the first place.

From the perspective of data engineering, a biased dataset is an unbalanced dataset [3], i.e. a dataset with an unequal distribution of some attributes that are considered protected (gender, ethnicity, etc.). This is problematic because even if we don't use such attributes as predictors , an unbalanced dataset can still lead to significant discrimination. The reason is that if the sensitive attribute is even slightly correlated to a number of features (which is often the case), the compound effect is comparable to that of a highly correlated feature, making easy for the algorithm to infer the sensitive attribute even if it is removed. In conclusion, it is not possible to achieve fairness through unawareness but rather it is necessary to accurately evaluate the quality of the data we use.

In our study, we tried to determine thresholds that would help us predict the fairness of the classification starting from the balance of the data used to train the algorithms . This would be helpful in order to anticipate the bias, detecting it in the data we use to train our ADMs rather than searching for it in the outcomes of the decision-making process.

Chapter 2

Background

Here we outline the underlying concept supporting our approach: data imbalance as a risk factor for systematic discrimination caused by ADM systems. This approach stems from software quality and risk management ISO standards, which constitute the two guiding principles.

The first principle originates from the series of standards ISO/IEC 25000:2014 Software Engineering — Software Product Quality Requirements and Evaluation (SQuaRE) [4], which describes quality models and measurements of software products, data and services. In this family of standards, quality is composed of quantifiable characteristics and sub-characteristics. Especially, data quality is modeled in ISO/IEC 25012:2008 with 15 characteristics (e.g., accuracy, reliability, completeness), all quantifiable through measures defined in ISO/IEC 25024:2015.

Although data balance is not a characteristic of data quality in ISO/IEC 25012:2008 it can be seen as a possible extension, being a key element in the chain of effects and dependencies described in SQuaRE: according to it, data quality has an effect on the quality of the system in use and, consequently, on the users of the system. In the context of ADM systems, imbalanced datasets may lead to imbalanced software outputs, i.e. differentiation of products, information and services based on personal and protected characteristics, and thus discrimination. Then, data imbalance can be considered as a risk factor in all those ADM systems that rely on historical data and that automate decision on aspects that concern the exercise of rights and freedoms, such as the ones employed in the public sector services.

The second principle is derived from the ISO 31000:2018 standard on risk management [5]. This standard provides the guiding principles for risk management, a framework for integrating it into organizational contexts, and a process for managing risks at “strategic, operational, program or project levels”. Our assumption is that we can assess the risk of bias in the output of ADM systems by measuring the level of balance of protected attributes in the data processed by the algorithms [6].

Chapter 3

Experimental design

The goal of our study was to understand how the balance of protected attributes in the training data can be used to predict the impact of a classification, considering as impact un unfair treatment with respect to those protected attributes. In particular, we wanted to answer the following research question:

Is it possible to build two thresholds s and f, respectively for balance and unfairness measures, such that if the balance of the training dataset is over s, than the unfairness of the classification on the test set is assured to be under f?

As a starting point, we selected a number of datasets, algorithms and measures to use in the study. Specifically we got:

- 7 Datasets from domains in which the impact of biased ADMs can be particularly high. For each dataset, we identified two protected attributes, one binary and one multiclass;
- A set of indexes that are able to measure the balance of a dataset , and the unfairness of a classification task: 4 for the balance and 5 for the unfairness;
- 2 mutation techniques, one for the binary attributes and one for the multiclass attributes, used to create synthetic versions of the datasets;
- 4 classification algorithms to run a binary classification;
- 5 sensitivity measures to assess the goodness of the thresholds built

Then, to build the aforementioned thresholds, we adopted the following procedure, separately for the binary and multiclass case:

1. Using a specific mutation technique and given seeds, generate a large number of synthetic datasets with different levels of balance with respect to the protected attribute;

2. For each algorithm and for each synthetic dataset, perform a binary classification with a training-test split of 70%-30%, computing the the balance measures of the training set and the unfairness measures of the classification (see Figure 3.1), thus obtaining a collection of data in which each instance is a classification

3. Repeat step 1-2 with new seeds to obtain a new collection of data;

4. Using the first collection of data, build the thresholds; using the second collection of data, gauge their goodness with the chosen sensitivity measures.

Incidentally, we also examined the relationship between balance measures and fairness criteria, expecting to find a negative correaltion between them: in other words, that a lower level of balance leads to a higher level of unfairness.

The result of step 1-2 is a large collection in which we record, for each classification, the dataset processed, the algorithm used, the specific mutation that has occurred, the balance measures of the training set, and the unfairness measures of the classification. In other words, each row is one of the several thousands of classifications performed. When we implement the second run, the different seeds ensure that the results are different, and we obtain a second collection of other several thousands of classifications on which to test the thresholds built on the first collection.

The study was performed with the R software.

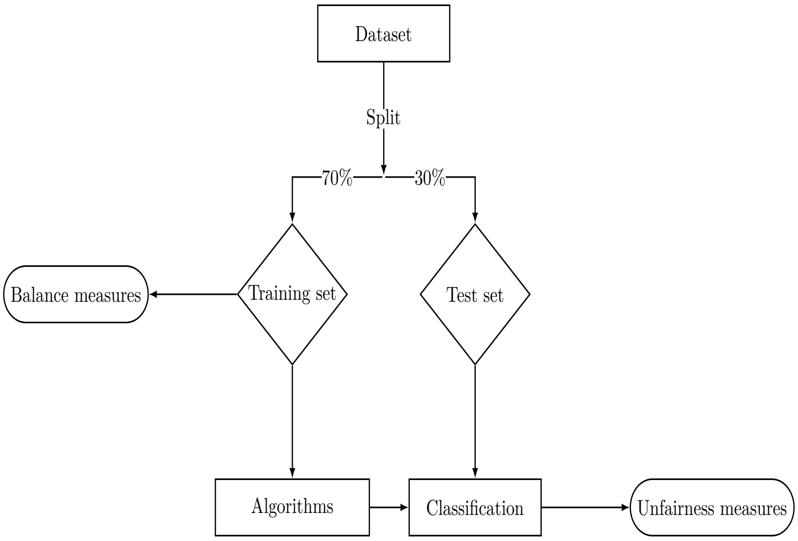


Figure 3.1: Balance and Unfairness measures.

3.1 Balance Measures

The first set of measures is known as balance measures; they are used to evaluate how balanced a dataset is with respect to a sensitive attribute. In particular, in our study, we limited our attention to categorical attributes and the measures respect the following conditions:

- values are in the range [0:100]
- the higher the balance of the data, the higher the value of the measure
- deal with empty classes, i.e. classes that exist (potentially there could be occurrences) but are not represented. The reason for this choice is that, in our view, a dataset that contains no instances of a given class is imbalanced

There are numerous ways to define the balance of a dataset and we selected four different criteria, namely Gini, Shannon, Simpson, and Imbalance Ratio.

Gini Index It is a measure of heterogeneity, which reflects how many types of a particular group are represented. It is used in a number of fields, such as political polarization or market competition, and often with different designations. In statistics, the heterogeneity of a discrete random variable which assumes m categories with frequency *fi* (with i = 1, ..., m) can vary between a degenerate case (minimum value of heterogeneity) and an equiprobable case (maximum value of heterogeneity, since categories are all equally represented). More similar frequencies equals a higher value of the index, which is computed as follows:

$$G = \frac{m}{m-1} \cdot (1 - \sum_{i=1}^{m} fi^2) \cdot 100$$

Where $\frac{m}{m-1}$ is the normalizing factor.

Shannon Index It is a diversity measure, useful to assess the balance of a community taking into consideration both its composition and the number of different species, a widely employed concept in biology and ecology. It is computed as follows:

$$G = -(\frac{1}{\ln m}) \sum_{i=1}^{m} fi \ln fi \cdot 100$$

Where $(\frac{1}{\ln m})$ is the normalizing factor. Since $\ln(0)$ tends to $-\infty$, when dealing with empty classes we resort to the notable limit $\lim_{x \to 0} x \ln x = 0$

Simpson Index It is another index of diversity: it measures the probability that two individuals randomly selected from a sample belong to the same species. It is employed in social and economic sciences for measuring wealth, uniformity and equity, as well as in ecology for measuring the diversity of living beings in a given location. It is computed as follows:

$$G = \frac{1}{m-1} \cdot (\frac{1}{\sum_{i=1}^{m} fi^2} - 1) \cdot 100$$

Where $\frac{1}{m-1}$ is the normalizing factor.

Imbalance Ratio It is a widely used measure made of the ratio between the highest and the lowest frequency (of classes). We take the inverse to normalize it in the chosen range. It is particularly sensitive to class imbalance, given that even if just one class is unrepresented (and the other are evenly distributed), the value goes to the minimum.

$$IR = \frac{\min\{f1...m\}}{\max\{f1...m\}} \cdot 100$$

3.2 Unfairness Measures

Unfairness measures are useful when evaluating the outcome of an automated decision system, to assure that it is fair with respect to the sensitive attribute. As for the balance measures, we will consider only categorical attributes, and in particular the unfairness measures respect the following conditions:

- values are in the range [0:100]
- the higher the fairness of the outcome, the lower the value of the measure (opposite behavior with respect to the balance measures)
- if the conditions for the specific criterion are not satisfied, we get an "NA"

Here we selected 3 different criteria, formalized in [7], each of them encoding a different point of view on what constitutes a fair classification. In general, to evaluate the fairness we consider a sensitive categorical attribute A, a target variable Y and the predicted class R: the fairness criteria proposed in this study are properties of the joint distribution (A,Y,R), and fall in one of three different categories: Independence, Separation, or Sufficiency, as summarized in the table below.

Independence	Separation	Sufficiency
R$\perp A$	R$\perp A\|Y$	Y$\perp A\|R$

Table 3.1: Non discrimination criteria.

Independence This criterion simply requires the sensitive characteristic to be statistically independent of the score. In the case of a binary classification, it simplifies to the condition:

$$\mathbb{P}\{R = 1|A = a\} = \mathbb{P}\{R = 1|A = b\}$$

In other words, it assumes that there is no correlation between the sensitive attribute and the target variable. If we think of a CV evaluation, it would mean that traits relevant for a job are indipendent of certain attributes. However, because this criterion does not take into account the accuracy of the classifier (Y is not considered) it can have undesirable properties. Imagine that the company hires people from both groups with the same probability and, among those who are hired, there are less qualified members in one of the two group: it would set a negative record for such a group. This can easily happen even if the two groups, in general, are equally qualified: for instance, the company does not have enough data on a given group, leading to higher error rates.

Separation The second criterion, instead, acknowledges that the sensitive attribute may be correlated with the target variable. For instance, a bank might argue that it is a business necessity to give different lending rates for two groups which have different default rates. Put it simply, the separation criterion allows correlation between the score and the sensitive attribute only to the extent that is justified by the target variable. Statistically speaking, this translates to R being statistically indipendent of A given Y, as illustrated in the graphical model in Figure 3.2.

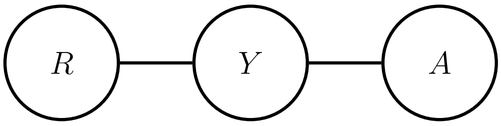


Figure 3.2: Graphical representation of the Separation criteria.

In the case where R is a binary classifier, separation is equivalent to requiring for all groups a, b the two constraints constraints:

$$\mathbb{P}\{R = 1|Y = 1, A = a\} = \mathbb{P}\{R = 1|Y = 1, A = b\}$$
$$\mathbb{P}\{R = 1|Y = 0, A = a\} = \mathbb{P}\{R = 1|Y = 0, A = b\}$$

$\mathbb{P}\{R = 1|Y = 1\}$ is called true positive rate, the rate at which the classifier recognizes positive instances, while $\mathbb{P}\{R = 1|Y = 0\}$ is called false positive rate, the rate at which the classifier mistakenly assign positive labels to negative instances. Thus, separation requires this two rates to be equal among the different classes, which can be called parity of odds.

Sufficiency The third criterion requires that individual of the same class are treated equally. It is a change of perspective, now accepting the score variable to be as correlated to the sensitive attribute as it needs to assure that the predictive quality is the same among the different classes. It can also be represented as a graphical model, in which the target variable is independent of the sensitive attribute given the score.

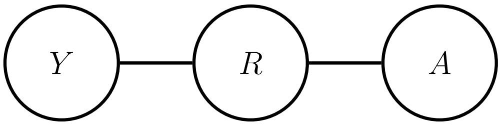


Figure 3.3: Graphical representation of the Sufficiency criterion.

When R has only two values we recognize this condition as requiring a parity of positive predictive values (PPV) and negative predictive values (NPV) across all groups:

$$\mathbb{P}\{Y = 1|R = 1, A = a\} = \mathbb{P}\{Y = 1|R = 1, A = b\}$$

$$\mathbb{P}\{Y = 1|R = 0, A = a\} = \mathbb{P}\{Y = 1|R = 0, A = b\}$$

Sufficiency is often satisfied by default as a consequence of standard machine learning practices. The flip side is that imposing sufficiency as a constraint on a classification system may not be much of an intervention.

In conclusion, we have 3 different criteria, but two of them have 2 condition, totaling 5 different conditions; each of them constitutes a measure in our study. From now on, for the purpose of disambiguation, we will refer to the conditions as measures, while Independence, Separation, and Sufficiency will be referred to as criteria.

3.3 Datasets

Now we examine the datasets processed, which are related to three different domains: Financial, Social and Health. We usually choose these fields when discussing the application of ADM systems because of the potential impact of unfair decisions, which could significantly affects people's lives. We selected 7 different datasets, but one of them was used in two different classification tasks.

All datasets were retrieved from the UCI machine learning repository, and their relevant features are summarized in Table 3.2. They contain informations about individuals: some variables are related to the given context and used for the classification, others are considered sensitive or protected, such as gender or age. Among these, we chose the sensitive attributes on which to base our analysis, one binary and one multiclass for each dataset. The list of predictors and target variable for each dataset are reported in Appendix A.

Dataset	Domain	Binary attribute	Multiclass attribute	Target
Credit card default	Financial	Sex	Education	Default payment next month
Statlog	Financial	Sex	Age	Creditworthiness
Student performance (Math)	Social	Sex	Father Education	Final grade
Student performance (Portuguese)	Social	Sex	Father Job	Final grade
Census income	Financial	Sex	Race	Income bracket
Drug consumption (Cannabis)	Social	Sex	Ethnicity	Cannabis consumption
Drug consumption (Impulsive)	Social	Sex	Ethnicity	Impulsiveness
Heart disease	Health	Sex	Age	Diagnosis

Table 3.2: Datasets relevant features.

Credit card default This dataset contains informations about default payments of credit card clients in Taiwan from April 2005 to September 2005 [8]. It includes credit data, history of payment, bill statements, together with demographic informations. The dataset is composed of 30000 instances with 25 variables, mostly categorical. For the purpose of having results within reasonable time with limited computing resources, we sampled 30% of the original dataset; even so, it has still a considerable amount of instances (9000), more than most of the dataset considered here.

We chose *Sex* and *Education* as sensitive attributes, and *default.payment.next.month* as target variable, which clarifies if the default has happened or not. The prediction of the default is based on the different credit data available, and could have a high impact on the individual, determining if the loan is granted or not.

Statlog This German credit dataset has been provided by the German professor Hans Hofmann as part of a collection of datasets from an European project called “Statlog” [9]. The data are a stratified sample of 1000 credits (700 good ones and 300 bad ones) and have been collected between 1973 and 1975 from a large regional bank in southern Germany, which had about 500 branches, both urban and rural ones. Bad credits have been heavily over-sampled, in order to acquire sufficient data for discriminating them from good ones. Specifically, the dataset contains 20 categorical attributes: each entry represents a person who takes a credit by a bank and is classified as good or bad credit risk.

We chose *Sex* and *Age* as sensitive attributes, and *costMatrix* as a target variable, which specifies the class of the customer (good or bad). The attribute *Age* ranges from 19 to 75, and we divided it into 5 ranges (classes) of 15 year. As for the previous dataset, a misclassification could result in excluding an individual from the loan service.

Student performance. This is a set of two datasets containing information on students achievements in secondary education of two Portuguese schools; they have been built by using school reports and questionnaires in 2014 [10]. There are a total of 624 instances, and the attributes include student grades, as well as demographic, social and school related features. The set of features are the same for the two dataset, including the target variable, which represents the final grade for Math or Portuegese (each dataset is about one of the two subjects); the final grade was divided into two classes by taking 9/20 as a threshold (<= 9, > 9).

We chose *Sex* and *Age* as sensitive attributes, and *G3_target* as target variable, which is nothing but the final grade for the given subject. Here, a biased evaluation of a student could significantly affect his academic and/or social life onward in unpredictable ways.

Census income. These data were extracted by Barry Becker from the 1994 Census database and is also known as “Census Income” dataset [11]; the associated prediction task is to determine whether a person makes over $50, 000 a year based on a set of reasonably clean records (the two classes to predict are then high or low income). It counts over 48000 instances and 15 variables: again, to avoid unnecessarily long training time, we took a sample of 30% of the original dataset.

We chose *Sex* and *Race* as sensitive attributes, and *test.income* as target variable, which can assume the two values <=50K or > 50K. If the income of a person is related to its demographics, it could be the reflection of some societal biases.

Drug consumption. It contains records for 1885 respondents; for each of them, personality measurements are known, together with some demographic data [12]. In addition, participants were questioned concerning their use of 18 legal and illegal drugs: for each drug they have to select one of the answers: never used, used it over a decade ago, or in the last decade, year, month, week, or day. There is also one fictitious drug (Semeron) which was introduced to identify over-claimers. All variables are quantified, with fixed values representing specific categories. This dataset was used for two different classification tasks:

1. Predict the consumption of a drug given the personality data. The problem was transformed to binary classification by union of part of classes into one new class: in particular, "Never Used", "Used over a Decade Ago" form class "Non-user" and all other classes form class "User". We chose *Cannabis*, but it could be done for any drug in the same manner. Over or underestimate the consumption of drugs of individuals could lead to worse treatments in a certain group.

2. Predict a personality trait given the consumption of drugs. The problem was transformed to a binary classification by dividing the personality trait into two classes, taking the mean value as a threshold. We chose *Impulsiveness*, but it could be done for any personality trait in the same manner. As for the consumption of drug, a biased judgment of the personality could induce an improper treatment.

For both the classification tasks, the sensitive attributes are *Sex* and *Ethnicity.*

Heart disease. This dataset describes a range of conditions that could affect the heart [13].These include blood vessel diseases, such as coronary artery disease, heart rhythm problems and congenital heart defects, as well as others. It consists of 303 instances and 14 variables, mostly clinical data. The original dataset contain 76 variables, but all published experiments refer to the subset considered in our study.

We chose *Sex* and *Age*, and *diagnosis_target* as target variable. The attribute *Age* ranges from 29 to 77, and we divided it into 5 ranges (classes) of 10 year (except for the last one). It is usually difficult to identify an heart disease, so it would be incredibly helpful to be able to predict an incoming disease thanks to the data. In the case the data were biased, the results would favor a group with respect to others in preventing the insurgence of the disease.

3.4 Algorithms

In our analysis, we used different algorithms, in order to process the data in a variety of ways. We wanted to verify if there are significant differences among them when establishing the thresholds, to better generalize our study, but we were not interested in their particular performances. For this reason, we did not perform hyper-parameters tuning, keeping the default parameters. These are the 4 algorithms used in the study:

- Logistic Regression - function *glm*, with attribute family=binomial(link="logit), from the package *stat* [14]
- Support Vector machine - function *svm* from the package *e1071* [15]
- Random Forest - function *randomForest* from the package *randomForest* [16]
- K-nearest neighbors - function *knn* from the package *class* [17]

3.5 Mutation techniques

We can distinguish between the binary and the multiclass case, for which we used two different mutation techniques.

For the binary attributes, we used the function *ovun.sample* from the package *ROSE* [18]. The parameter relevant for the mutations is *p*, which determines the probability to sample from the minority class. We selected 9 values for p, ranging from 0.01 (high imbalance) to 0.5 (perfect balance)

- p = {0.01, 0.025, 0.05, 0.075, 0.1, 0.2, 0.3, 0.4, 0.5}

If we factor in 8 datasets, and 50 seeds, we obtain $8 \times 50 \times 9 = 3600$ synthetic datasets. Considering that each dataset is processed by 4 differerent algorithms, we have a total of $3600 \times 4 = 14400$ classifications.

For the multiclass attributes, we created several distributions of occurrences for the classes of the protected attributes and then generated datasets reflecting those distributions. To this end, we used the function *SmoteClassif* from the package *UBL* [19]. The parameter relevant for the mutations is *c.perc*, a named list containing the percentages of under-sampling or/and over-sampling to apply to each class of the sensitive attributes. We examined five different configurations for the parameter C.perc: first, the default configuration “balance” (namely, the perfect uniform distribution, with all the occurrences equally distributed among the different classes) together with four additional configurations, corresponding to the exemplar distributions “Power2”, “HalfHigh”, “OneOff” and “QuasiBalance”.

- *Power 2*: occurrences are distributed according to a power-law with base 2, i.e., distributions among the classes increase like the powers of 2 (for instance, considering 3 classes, 1:7, 2;7, 4:7);

- *Half High*: occurrences are distributed mostly among half of the classes while the remaining ones have a very low frequency (in particular, a ratio of 1:9 has been chosen for the frequencies of the two halves);

- *One Off*: occurrences are distributed among all classes but one (which has 0 occurrences);

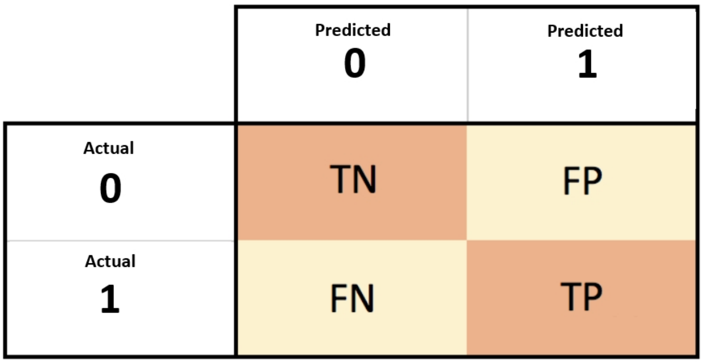
- *Quasi Balance*: half of the classes are 10% higher w.r.t. max balance and the other half is 10% lower.

In addition, for each exemplar distribution we considered 4 permutations of the percentages assigned to the different classes. For instance, in the *One Off* configurations the four different permutations have each a different class with zero occurrences. If we factor in 8 datasets , and 50 seeds, we obtain $8 \times 50 \times (4 \times 4 + 1) = 6800$ synthetic datasets. Considering that each dataset is processed by 4 different algorithms, we have a total of $6800 \times 4 = 27200$ classifications.

3.6 Sensitivity measures

Sensitivity measures are used to assess the reliability of the thresholds. As stated before, we have a threshold for the balance measure "s" and a threshold for the unfairness measure "f". When we the balance of the training set is over "s", we expect the unfairness of the classification to be under "f": if this actually happens, we have a positive instance, otherwise we have a negative instance.

In our two collections we have as many instances as classifications; in other words, the evaluation of the thresholds can be considered as a binary classification, verifying if a classification (instance of the new collection) respects the conditions (on the balance and unfairness measures) or not. The first collection can be seen as the training set, used to build the thresholds, the second as the test set, used to test them. Binary classifications performance measures are best introduced by showing a confusion matrix, in which each row represents the instances in the actual class while each column represents the instances in the predicted class:



	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

Figure 3.4: Confusion matrix [20].

- P: the number of actual positive instances in the data
- N: the number of actual negative instances in the data
- TP: the number of instances predicted as positive that belong to the positive class
- TN: the number of instances predicted as negative that belong to the negative class
- FP: the number of istanced predicted as positive but belong to the negative class
- FN: the number of istanced predicted as negative but belong to the positive class
- PPV: the fraction of positive instances correctly predicted to be in the positive class out of all predicted positive instances $= \frac{TP}{TP+FP}$
- TPR: the fraction of positive instances correctly predicted to be in the positive class out of all actual positive instances $= \frac{TP}{TP+FN}$

In our case we have:

- TP → balance $<$ s & unfairness $>$ f
- TN → balance $>$ s & unfairness $<$ f
- FP → balance $<$ s & unfairness $<$ f
- FN → balance $>$ s & unfairness $>$ f

And these are the 5 sensitivity measures used in our study, with their respective formulas:

1. Accuracy $=\frac{TP+TN}{P+N}$
2. Sensitivity $= \frac{TP}{TP+FN}$ = TPR
3. Specificity $= \frac{TN}{TN+FP}$ = TNR
4. Precision $= \frac{TP}{TP+FP}$
5. F1 $= \frac{PPV \times TPR}{PPV+TPR}$

Accuracy and precision are pretty straight-forward, evaluating how many instances are correctly classified: accuracy with respect to the total instances, precision with respect the positive predicted instances. Sensitivity, also called recall, examines instead how many of the actual positive instances are retrieved (or "recalled"), and specificity is its opposite, considering how many of the actual negative instances are retrieved. F1 is the harmonic mean between sensitivity and specificity.

Chapter 4

Correlation analysis

Once we obtained all the balance and unfairness measures from the thousands of classifications, we assessed the correlation between the two sets of measures, separately for the binary and the multiclass case. To illustrate the results, we report a smoothplot and a correlation table. The smoothplot shows the trend of the unfairness (y axes) as a function of balance (x axes), and each dataset is represented with a different color, so that we can follow the individual trends together with the general trend (the color legend is reported at the bottom of the figures). The correlation table just shows the value of the correlation for any given combination of balance-unfairness measure.

As previously mentioned, we have 3 distinct unfairness criteria, which then translates to 5 different conditions/measures, with the following meanings:

- Indipendence:
  - Diff.Ind
- Separation:
  - Diff.TP - Equality of True Positives
  - Diff.FP - Equality of False Positives
- Sufficiency:
  - Diff.PP - Equality of positive predictive values
  - Diff.PN - Equality of negative predictive values

Factoring in the 4 balance measures, we obtain 20 different combinations.

4.1 Binary attributes

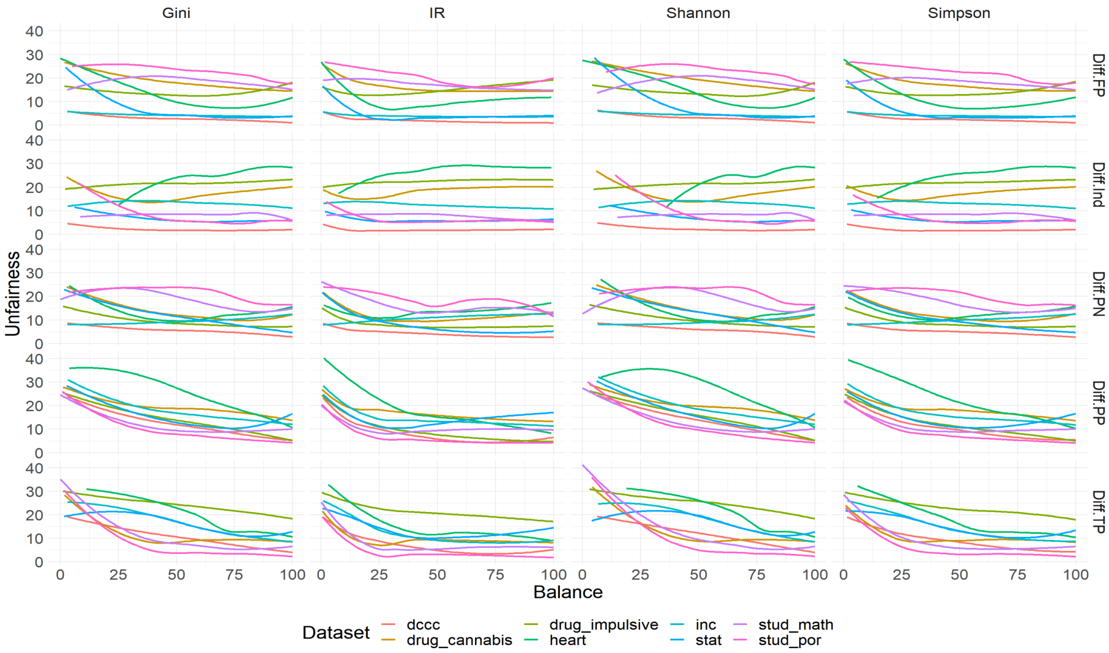
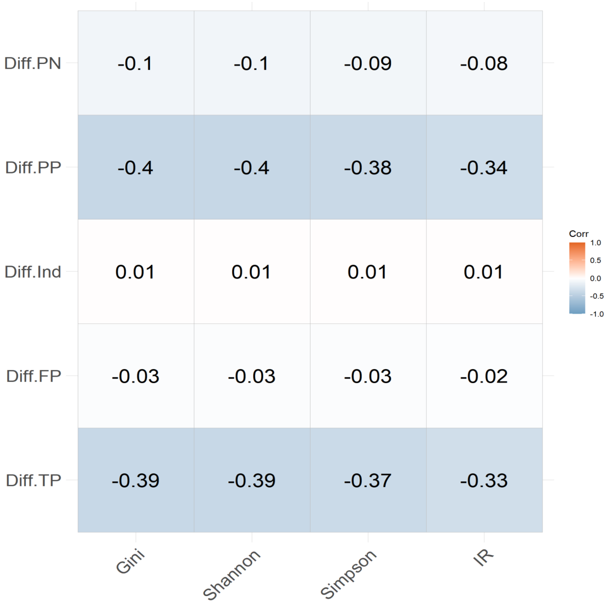


Figure 4.1: Smoothplot Balance-Unfairness, binary case.



	Gini	Shannon	Simpson	IR
Diff.PN	-0.1	-0.1	-0.09	-0.08
Diff.PP	-0.4	-0.4	-0.38	-0.34
Diff.Ind	0.01	0.01	0.01	0.01
Diff.FP	-0.03	-0.03	-0.03	-0.02
Diff.TP	-0.39	-0.39	-0.37	-0.33

Figure 4.2: Correlation table Balance-Unfairness, binary case.

Looking at the smoothplot, we notice that the trend isn't the same for every combination of balance-fairness, with some plots showing almost no variation for the unfairness as a function of the balance, while others show a clear decrease of the unfairness with the increase of the balance. There doesn't seem to be any significant difference in the trend among the different dataset, although we find different values at both ends of the spectrum.

Moving to the correlation table, we easily notice the difference among the various combinations, distinguishing between cases with almost no correlation (around 0) and cases with a moderate correlation (around 0.4). If we look at the specific combinations that return a negative correlation, we notice that it happens for 3 different unfairness conditions: Diff.PN, Diff.PP, and Diff.TP. Referring to the criteria, Diff.PN and Diff.PP together form the Sufficiency criterion, while Diff.TP is one of the two conditions for the Separation criterion; the other condition for Separation doesn't give significant correlation with the balance measures.

Apparently, the negative correlation seems to depend only on the unfairness criterion, rather than on the specific combination of balance-unfairness, because there's no significant difference in the correlation with the different balance measures, once the unfairness measure is fixed. In other words, unfairness conditions seem to correlate in the same way with every balance measure indiscriminately. This can be easily visualized in the correlation table observing that the values of the correlation are very similar along the horizontal axis.

4.2 Multiclass attributes

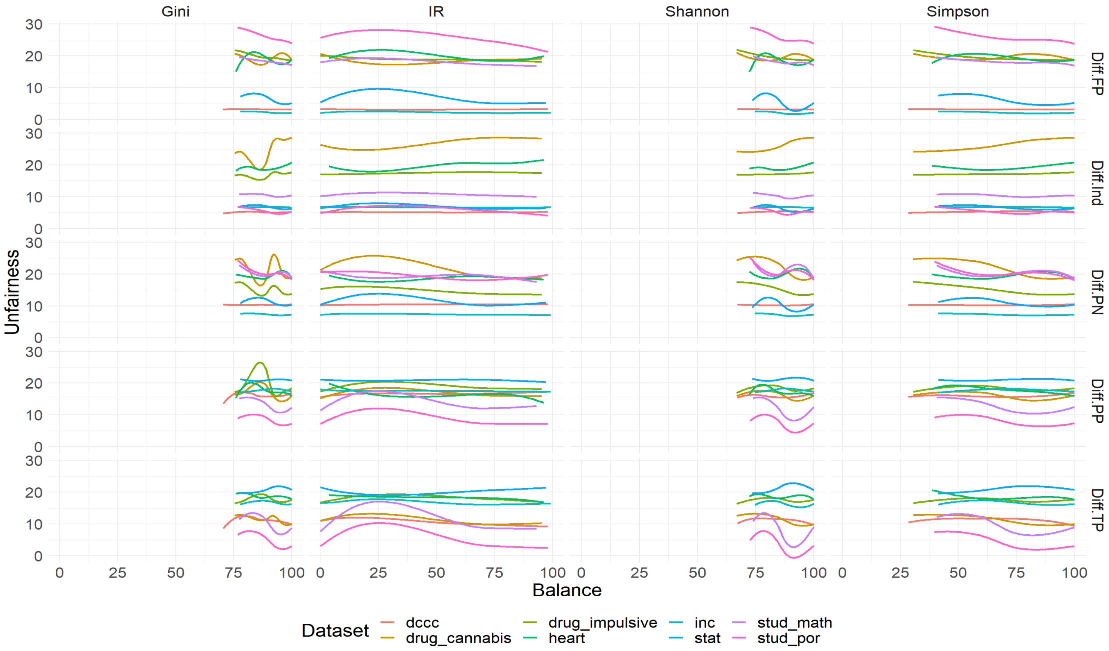
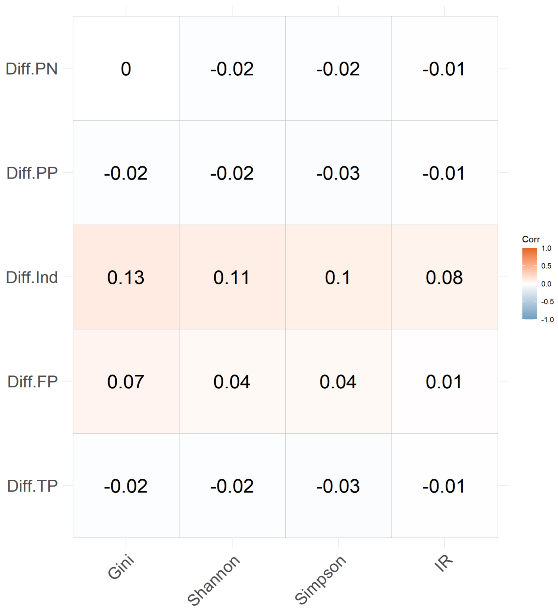


Figure 4.3: Smoothplot Balance-Unfairness, multiclass case.



	Gini	Shannon	Simpson	IR
Diff.PN	0	-0.02	-0.02	-0.01
Diff.PP	-0.02	-0.02	-0.03	-0.01
Diff.Ind	0.13	0.11	0.1	0.08
Diff.FP	0.07	0.04	0.04	0.01
Diff.TP	-0.02	-0.02	-0.03	-0.01

Figure 4.4: Correlation table Balance-Unfairness, multiclass case.

Here, the situation is quite different starting from the smoothplot, in which we see a lot of missing values, causing incomplete plots. The function used to compute the unfairness can return an "NA" when the conditions for the given measure are not respected, which in the multiclass case happens quite often. Specifically, this happens for low values of balance and especially for the balance measures Gini and Shannon.

Other than this, the trend is difficult to interpret even when we have a complete graph, with the correlation being close to 0 or even slightly positive in some cases, as visible in the correlation table. In conclusion, the negative correlation doesn't seem to hold in the multiclass case.

Chapter 5

Sensitivity analysis

We proceed to explain how we built the threshold on balance and unfairness measures. In our procedure, we first determine the threshold for the unfairness, and then retrieve the corresponding threshold for the balance, so as a starting point we looked at the distribution of the unfairness to see where the unfairness thresholds should be reasonably placed. We did it with violin plots, which are similar to boxplots, except that they show the probability density of the data at different values.

Then, we created different configurations of the threshold in a given range and selected the one that gave the best accuracy, separately for each combination of balance-fairness-algorithm. Each configuration consists in a different value for the fairness threshold, expressed in terms relative to the distribution of the fairness; the balance threshold is then retrieved accordingly.

5.1 Unfairness analysis

The violin plots for all the unfairness measures in both the binary and multiclass case are shown in Figure 5.1 and 5.2. We observe the same pattern for both the binary and the multiclass case: the values of the unfairness measures are relatively low, with the most density at the bottom of the distribution. This is true for every unfairness measure indiscriminately. From this finding we supposed that the thresholds should be also low, and took the first quartile of the distribution as a reference (we work with one distribution at a time, because we build the thresholds separately for each combination balance-unfairness-algorithm), building 5 configurations of thresholds around it.

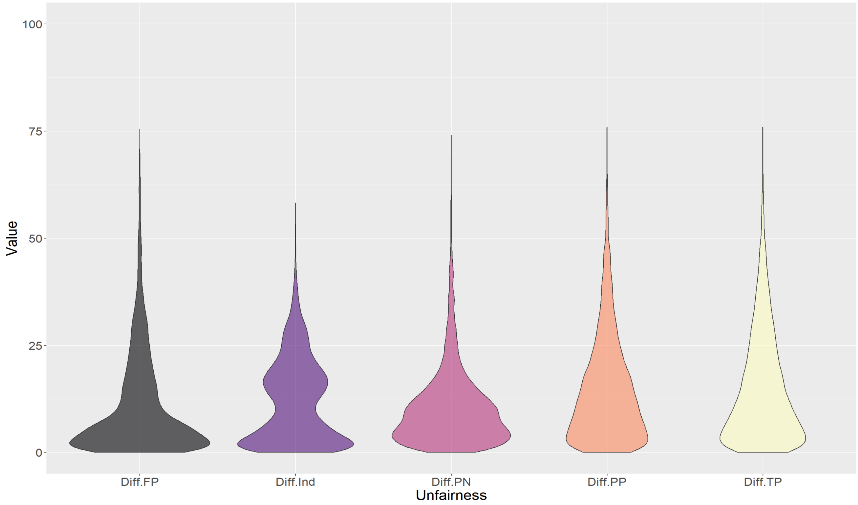


Figure 5.1: Unfairness measures distribution, binary case.

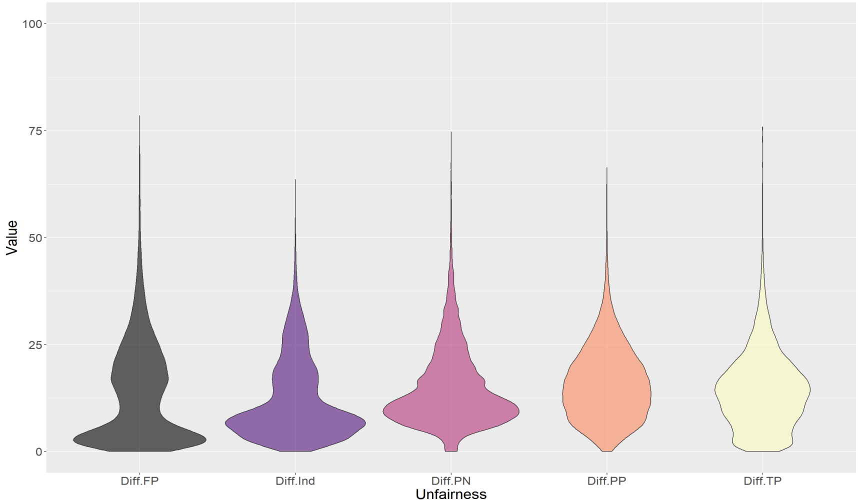


Figure 5.2: Unfairness measures distribution, multiclass case.

5.2 Method

We have two procedures, or better two variations: in the first case, the threshold for the unfairness is retrieved as an average between two values, so it's not possible to predict exactly where it falls, while in the second case it falls in a specific point relatively to the distribution. We created this two cases in order to distribute the thresholds for the unfairness evenly in the desired range.

Case A

1. Determine 2 values of the unfairness, f1_base and f2_base that conceptually create the following brackets:
   - $unfairness \leq f1\_base(low)$
   - $f1\_base < unfairness \leq f2\_base(medium)$
   - $unfairness > f2\_base(high)$
2. Identify in the first collection the values of unfairness nearest to f1_base and f2_base, and define them as f1 and f2
3. Retrieve in the first collection the two values of the balance corresponding to f1 and f2, i.e. the values found in correspondence of f1 and f2 in the collection, and define them as s1 and s2. If more than one balance value is found, we take their mean.
4. Define f as the mean between f1 and f2, s as the mean between s1 and s2

So, the first step is to identify the values of the unfairness that mark what should be considered a low/medium/high unfairness, and then put the threshold f in the middle. Then, we proceed to establish the corresponding threshold s for the balance. Here by collection we mean not the complete collection, but the collection filtered by the specific combination of balance-fairness-algorithm considered at the moment.

The unfairness thresholds could also be fixed arbitrarily, for instance in reference to case studies where we assume that an unfair treatment has occurred: if in some datasets that are knowingly biased the unfairness measures of a straight-forward classification are around 20, then we could assume 20 as threshold delimiting high values of unfairness. However, having a sufficient amount of data to generalize the results, we chose to refer to the actual distribution of the unfairness in the collection that we simulated, placing the unfairness thresholds in relative terms to the distributions.

Case B

1. Determine 1 value of the unfairness, f_base that create the following brackets:
   - $unfairness \leq f\_base(low)$
   - $unfairness > f\_base(high)$

2. Identify in the collection the value of unfairness nearest to f_base and define it as f

3. Retrieve the value of the balance corresponding to f, i.e. the values found in correspondence of f in the large collection, and define it as s. If more than one balance value corresponds to f we take their mean as s.

5.3 Configurations

Here, we introduce the 5 different configurations of thresholds created: for each combination of balance-unfairness-algorithm we select the one that returns the best accuracy. To show their placement, we take as a reference the distribution of Diff.TP in the binary case, coloring f1_base and f2_base with gray and f with red (or f_base, if we are following case B). The configurations have no specific reasoning behind them outside of placing the thresholds almost evenly in the desired range (around the first quartile); notice that configurations 1, 2 and 4 belong to Case A, whereas configurations 3 and 5 belong to Case B.

Configuration 1

- f1_base: 1st quartile
- f2_base: mean

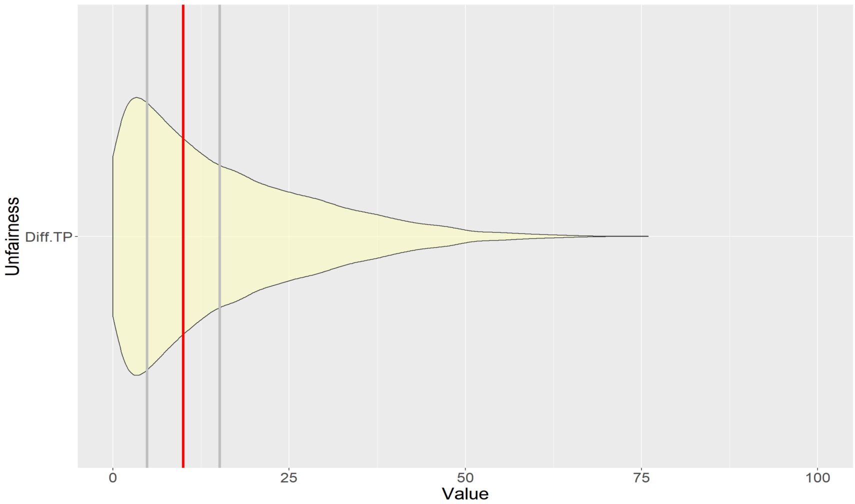


Figure 5.3: Thresholds configuration 1, unfairness values.

Configuration 2

- f1_base: mean (minimum, 1st quartile)
- f2_base: mean (1st quartile, mean)

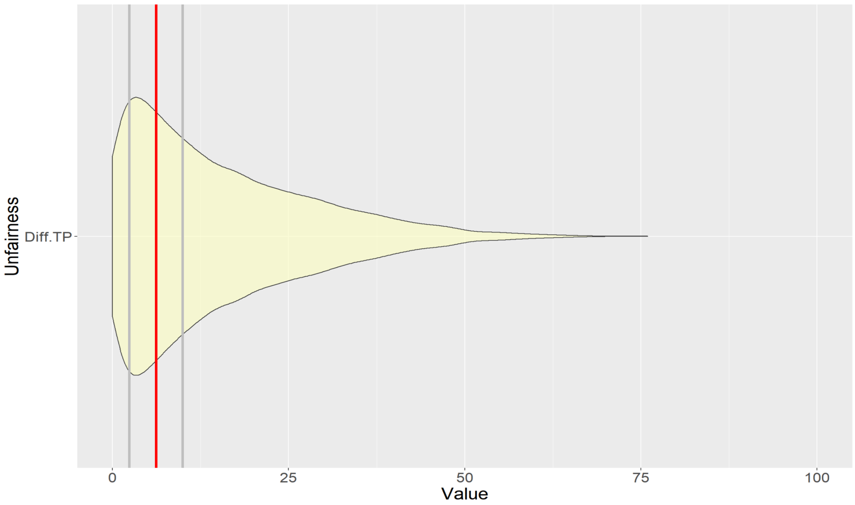


Figure 5.4: Thresholds configuration 2.

Configuration 3

- f_base: 1st quartile

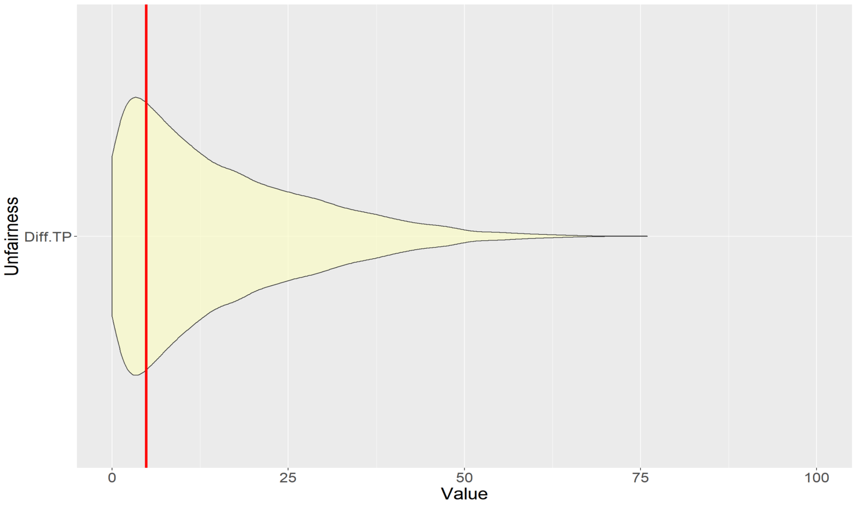


Figure 5.5: Thresholds configuration 3.

Configuration 4

- f1_base: mean (minimum, 1st quartile)
- f2_base: 1st quartile

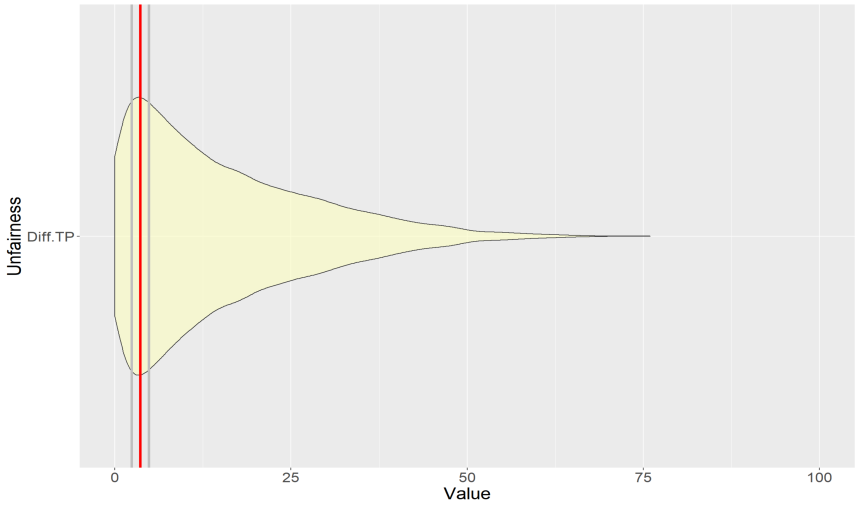


Figure 5.6: Thresholds configuration 4.

Configuration 5

- f_base: mean (minimum, 1st quartile)

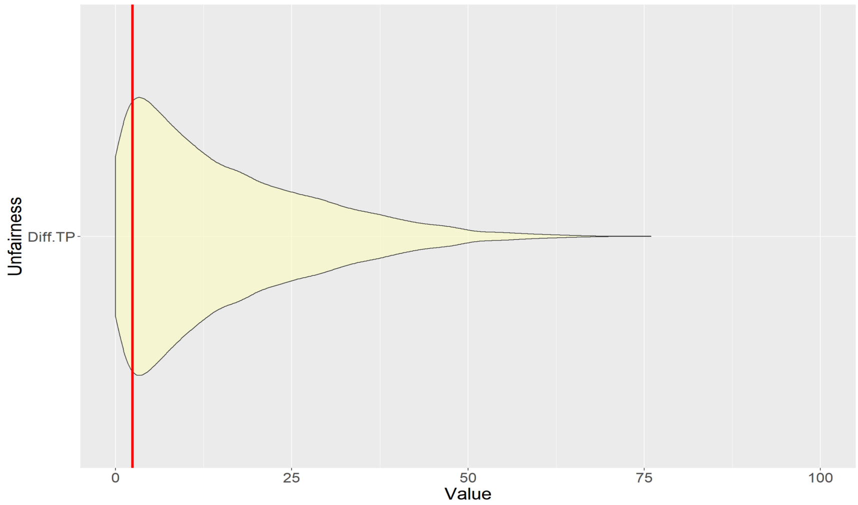


Figure 5.7: Thresholds configuration 5.

Chapter 6

Results and discussion

After establishing the thresholds, we obtain a dataset reporting, for each combination of balance-fairness-algorithm, the best thresholds selected by accuracy, the configuration they correspond to (among the 5 options enumerated in chapter 5.3), and all the sensitivity measures related to those thresholds (computed as described in chapter 3.6). The complete results are reported in appendix B as tables, ordered by balance measure (Gini, Shannon, Simpson, and IR), for both the binary and multiclass case. Here, instead, we show the aggregated results, the sensitivity measures and the thresholds, both displayed through boxplots.

Also, wanting to understand what factors could determine the goodness of the outcome, we investigated the correlation of the accuracy with respect to unfairness measures, balance measures, and algorithm, using again some boxplots.

Regarding the choice of the accuracy as a discriminant for the choice of the best configuration, other than it being the simplest measure, it also strongly correlates with the other measures, as clear from the following figures. Similar correlation values are found in both the binary and the multiclass case.

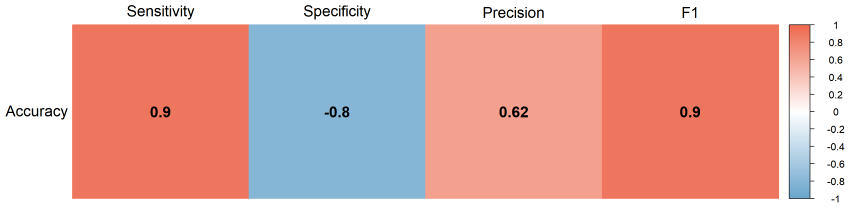


Figure 6.1: Correlation between accuracy and the other sensitivity measures, binary case.

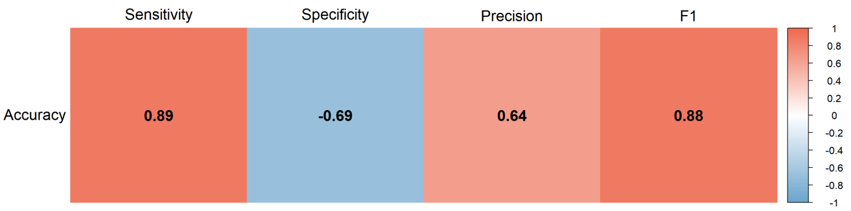


Figure 6.2: Correlation between accuracy and the other sensitivity measures, multiclass case.

6.1 Binary attributes

6.1.1 Sensitivity measures

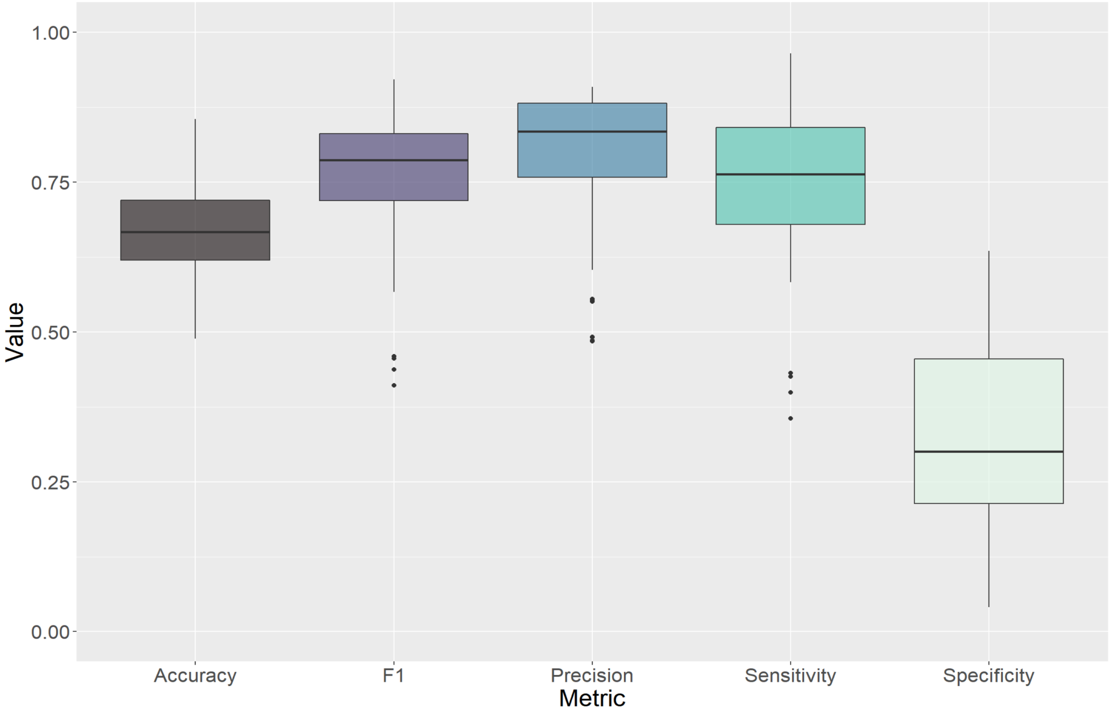


Figure 6.3: Sensitivity measures boxplot, binary case.

Almost all the sensitivity measures are high, except for specificity; being it equal to 1-sensitivity, if one of the two measure is high the other one is inevitably low. Sensitivity is also called recall, and it's the capacity of retrieving the positive instances, while specificity is the correspondent for negative instances. In our case, a positive instance is a classification with high impact (high unfairness); in other words, this thresholds are extremely responsive to risk (low balance), being able to anticipate the impact most of the times, but they also over-estimate it, causing all those false positives that reduce the specificity.

6.1.2 Thresholds

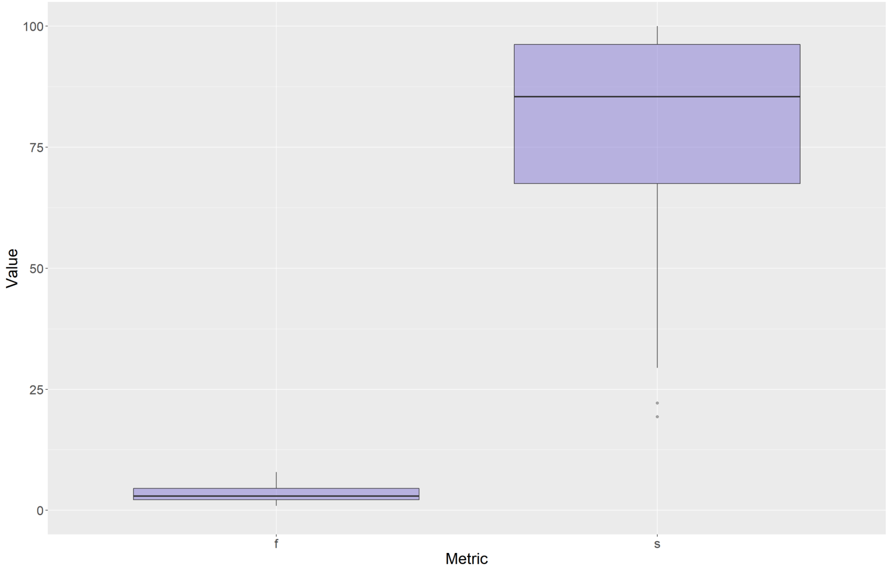


Figure 6.4: Thresholds distribution, binary case.

The thresholds for the balance are really high (around 80), explaining thus the high responsivity to risk, and the corresponding thresholds for unfairness are really low (around 5). The strict requirements on the balance probably lead to misjudge a lot of low-impact classification and reduce sensitivity, as already mentioned. Incidentally, in chapter4 we observed that the correlation between balance and unfairness isn't necessarily strong, having situations in which although the balance is low, the unfairness is still low (a sort of false alarm).

As for the configurations, shown in Figure 6.5, there is a clear tendency toward the ones positioned lower in the distribution, which correspond to the higher numbers (3 to 5). These are the configurations with a lower f (and usually an higher s) in accordance with the distribution of the thresholds just mentioned.

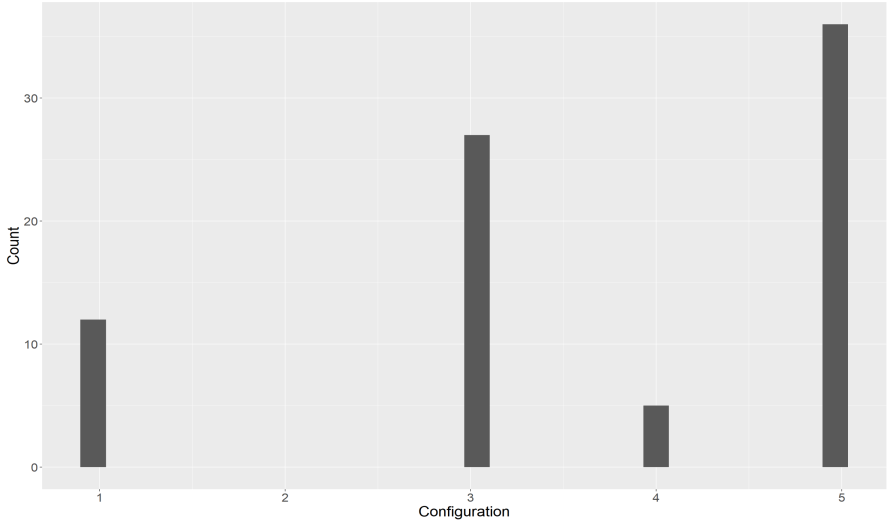


Figure 6.5: Configurations frequency, binary case.

6.1.3 Accuracy correlation

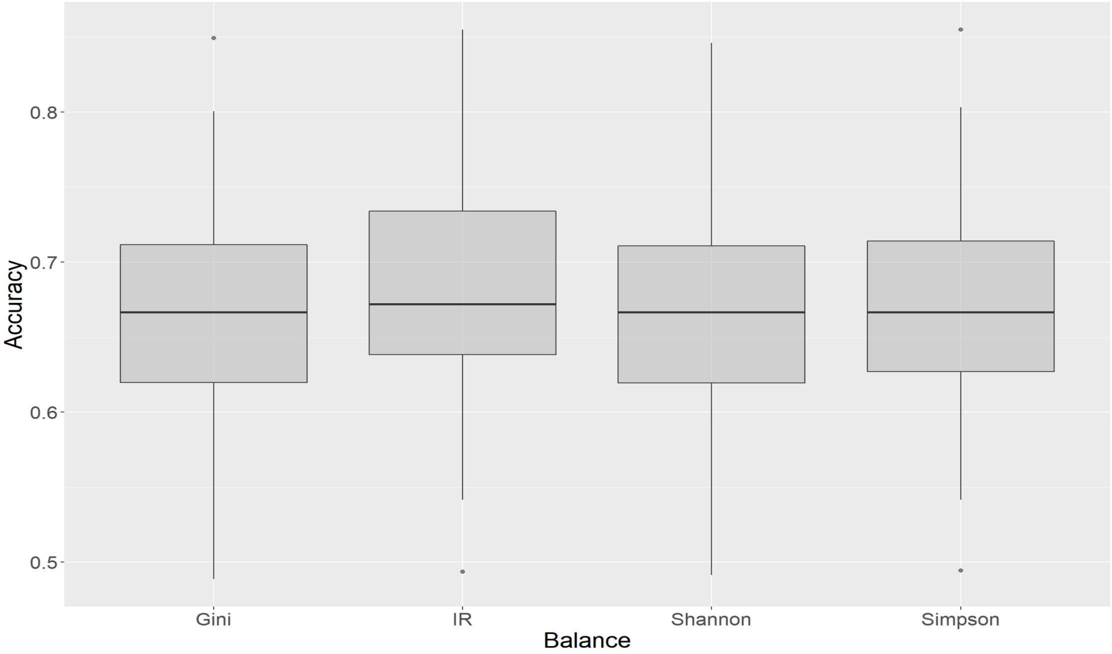


Figure 6.6: Balance measures - Accuracy boxplots, binary case.

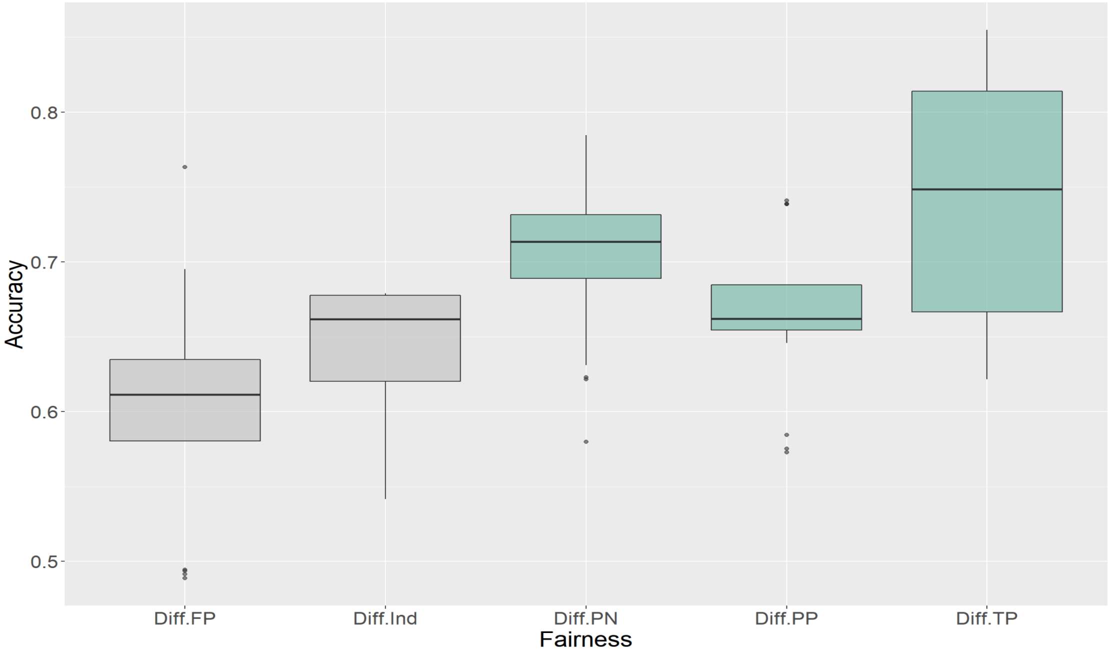


Figure 6.7: Unfairness measures - Accuracy boxplots, binary case.

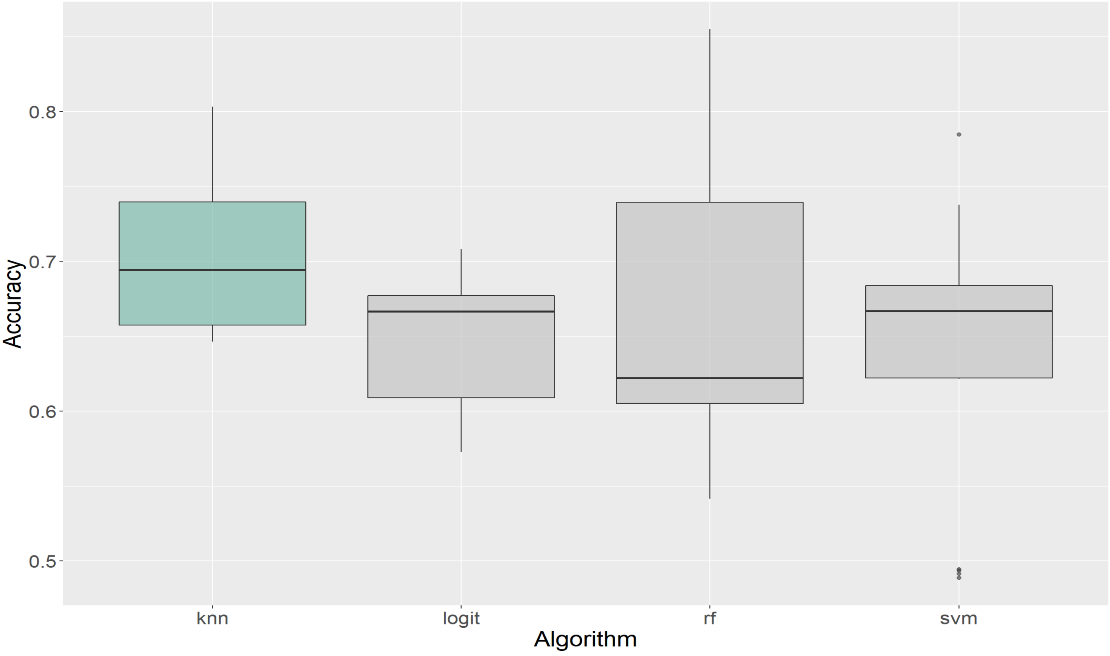


Figure 6.8: Algorithm - Accuracy boxplots, binary case.

Now we look at the possible correlation between the accuracy and the different factors involved in the process, namely the balance measures, the unfairness measure, and the algorithm, investigating the matter through some boxplots. This is an easy visual method to examine the correlation between a numerical and a categorical variable: if the correlation holds, we expect the boxplots relative to some categories to be different (higher or lower) with respect to the others. We comment on the results following the same order in which the boxplots are displayed.

Balance There is no significant correlation between the accuracy and the balance measures.

Unfairness There is a correlation between the accuracy and the unfairness measures involved. Specifically, the three measures on the right, Diff.PN, Diff.PP, and Diff.TP are at a higher level than the two on the left, Diff.FP and Diff.Ind: it means that the combinations that include those three measures return,on average, an higher accuracy. This is not entirely surprising; going back to chapter 4, we noticed that those three measures are the one that negatively correlates with the balance measures. Because our aim is to predict the values of the unfairness starting from the balance (in terms of ranges, not individual values), it is clear that this procedure becomes more accurate the more the two set of measures correlate.

Converting the measures to criteria, we can affirm that the thresholds are better when adopting the sufficiency criteria (Diff.PN and Diff.PP) and partly Separation (only for the true positives equality condition, while for the false positive equality condition we get the worst results). It is also evident that, among the three highlighted measures, Diff.TP has a much larger variance in terms of accuracy.

Algorithm There seems to be a slight correlation between the accuracy and the algorithm used, although not as high as with the unfairness measures. The algorithm most apt for establishing this thresholds is K-nearest neighbors by a slight margin. Random Forest also reaches high values but it has much more variance and its median is relatively low.

6.2 Multiclass attributes

6.2.1 Sensitivity measures

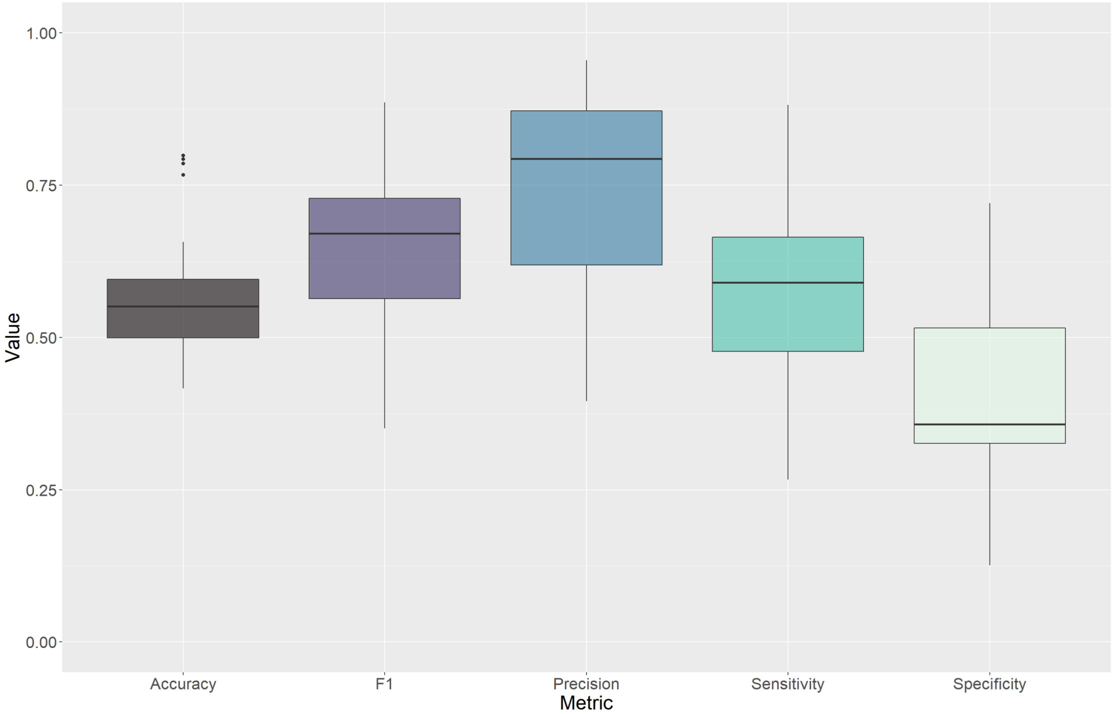


Figure 6.9: Sensitivity measures, multiclass case.

In the multiclass case the measures are quite lower then in binary case. We can interpret this results in light of the poor correlation between the balance measures and the unfairness measures in the multiclass case. Going back to chapter 4, we noticed that the negative correlation between balance measures and unfairness measures doesn't hold in the multiclass case; then, discussing the correlation between unfairness measures and accuracy in the binary case, we noticed that the higher the correlation of the unfairness measure with the balance measures, the higher the accuracy.

For the multiclass attributes there is not much correlation between the two set of measures, which then leads to the poor performances of the thresholds. Also, there is much more variability than in the binary case.

6.2.2 Thresholds

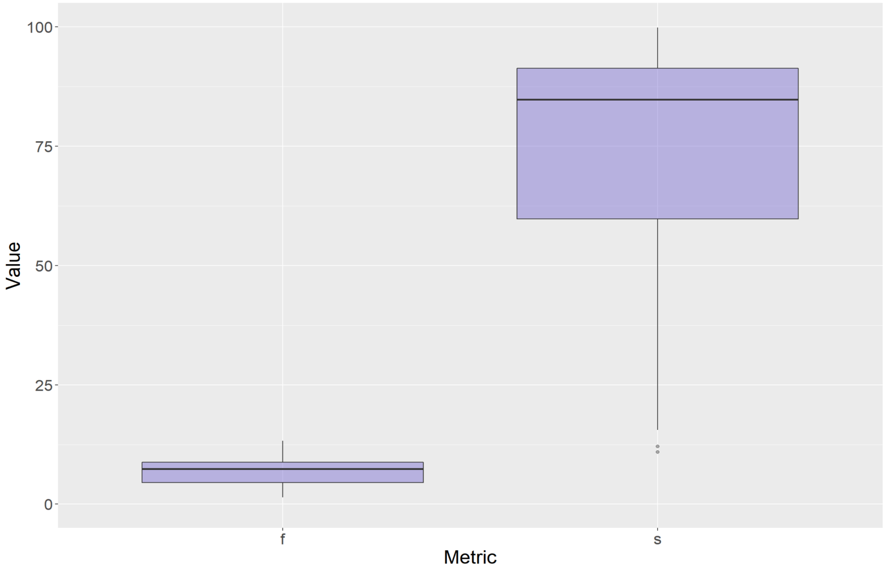


Figure 6.10: Thresholds distribution, multiclass case.

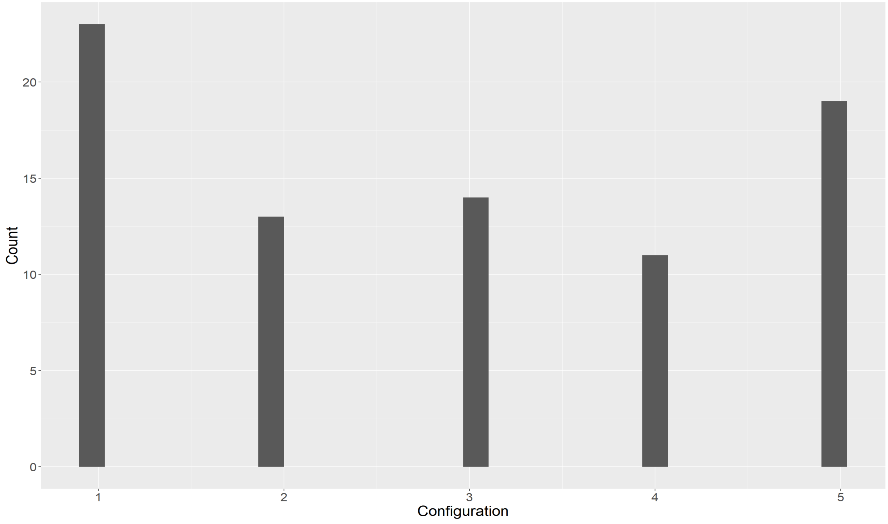


Figure 6.11: Configurations frequency, multiclass case.

As for the binary case, the thresholds for the balance tend to be very high and the thresholds for the unfairness very low, although not as extreme; the first ones are slightly lower and the second ones are slightly higher.

Looking at the configurations, we see that they are more evenly distributed in the range, suggesting that the multiclass case benefits from higher f thresholds (and lower s thresholds), even if the overall values are not so different.

6.2.3 Accuracy correlation

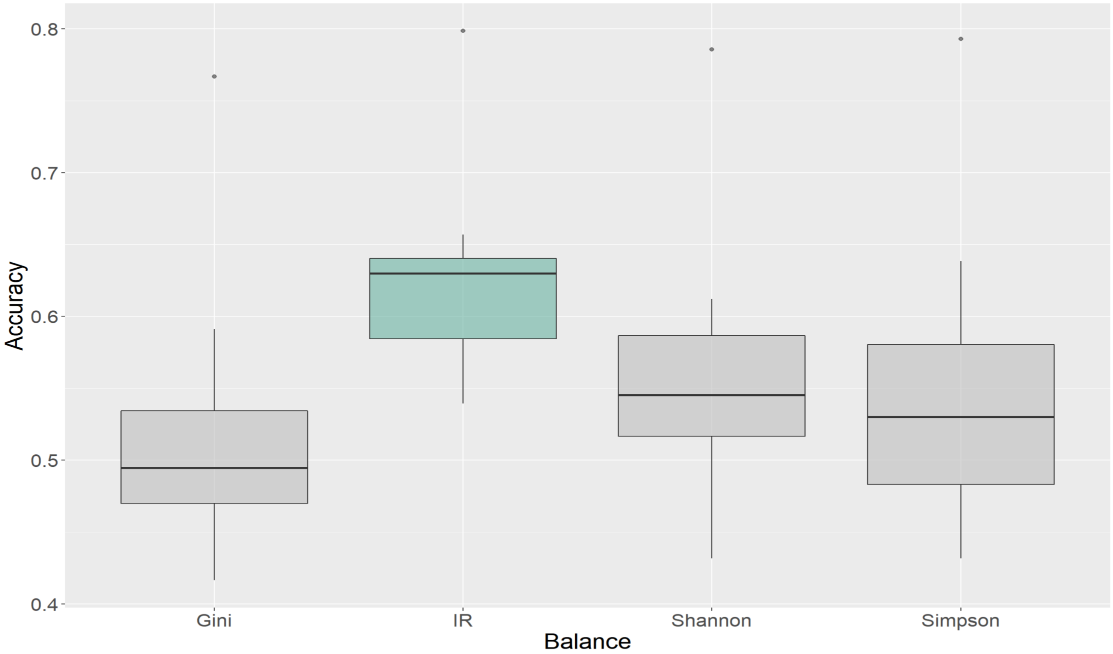


Figure 6.12: Balance measures - Accuracy boxplots, multiclass case.

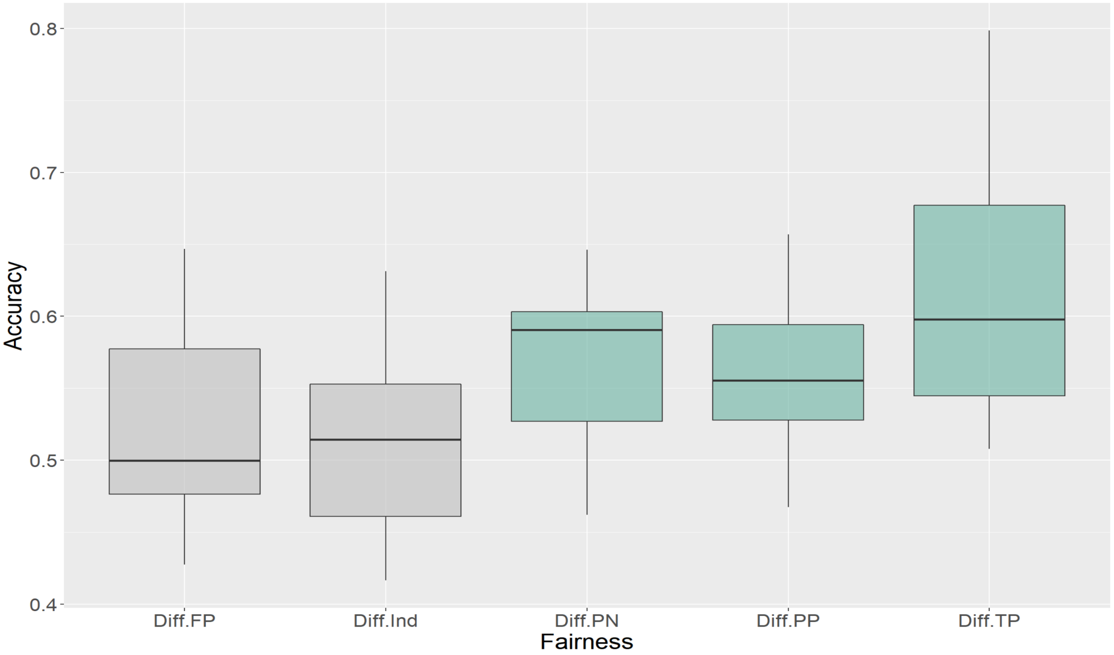


Figure 6.13: Unfairness measures - Accuracy boxplots , multiclass case.

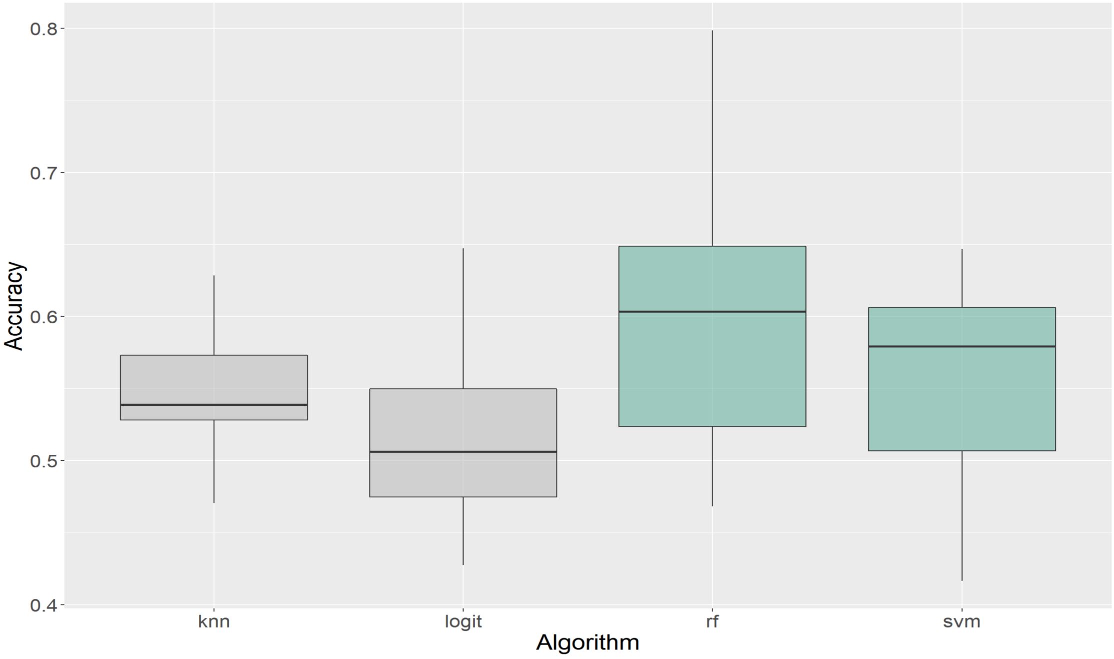


Figure 6.14: Algorithm - Accuracy boxplots, multiclass case.

Balance In this case the balance presents a correlation with the accuracy, with IR achieving better results. The reason could be that, in the multiclass case, IR is significantly more responsive to imbalance, thus leading to a better ability to predict the impact of the classification. The more classes are present, the less the impact of a single class being under-represented on the values of the other measures, because they consider a weighted sum over the classes, while the IR consider always two classes, the most represented and the less represented.

Unfairness We observe the same pattern found in the binary case, with the difference that now all measures present a rather high variance in terms of accuracy.

Algorithm Here we notice a difference with the binary case, with the best performing algorithms being Random Forest and Support Vector Machine.

Chapter 7

Conclusions and future work

In our study we evaluated imbalance in the data as a predictor for discriminatory output of ADM systems, combining aspects of data quality and risk management from the ISO standards. In order to do so, we selected four widely used indexes (Gini, Simpson, Shannon, and Imbalance Ratio), and three Fairness criteria (Independence, Separation, and Sufficiency), testing the reliability of thresholds used to infer the value of the fairness starting from the the value of the balance, i.e. anticipating the impact of a classification with respect to some sensitive attribute considering its distribution in the data fed to the algorithms.

Overall, the results indicate that the approach is suitable for the goal, but there is a significant variance between binary and multiclass cases; further work ought to be devoted to finding a procedure and/or measures that perform comparatively well in both cases. The range chosen for the thresholds (f around the first quartile of the distribution of a given unfairness measure) was effective in finding relatively high performing thresholds in both cases, even if in the binary case we notice a bias toward lower f thresholds (and higher s thresholds), while the multiclass case benefits from a more extensive range, with an even distribution of configurations. The accuracy as a criterion to choose the best configuration seems appropriate, also given the high correlation that it shows with a variety of commonly used sensitivity measures.

This leads to another consideration, namely that the values of the thresholds that produce the best results are rather strict. With respect to the full range of values (0-100) the f thresholds are all located in the range 0-15; as mentioned in chapter 5.2, values over the threshold f are to be considered medium/high unfairness. This is coherent with the values that unfairness measures reach in knowingly biased datasets when performing a straight-forward classification: for instance, the COMPAS dataset, well known in the scientific communities that study measures of algorithmic bias [21], presents values of different unfairness measures slightly above 20 (specifically, those measures correspond to Diff.Ind, Diff.TP, and Diff.FP in our study). The corresponding values of the balance thresholds show a much larger variance, but in general we could say that both thresholds present some extreme cases, with values near to 0 and 100, respectively for f and s.

Regarding the different factors involved, namely the specific balance measure, unfairness measure, and algorithm used, we found that there are significant differences when it comes to the performance of the thresholds. Regarding the balance, all measures are comparable in the binary case, while in the multiclass case IR gives the best performance; as mentioned in chapter 6.2.3, this could be related to the high responsivity to imbalance that distinguishes this measure from the others, especially when there a lot of classes. Considering the unfairness in both the binary and multiclass case, the criterion that, in its entirety, is clearly above the others is Sufficiency (Diff.PP, Diff.PN.); however, the condition on equality of true positives of the Separation criterion (Diff.TP) gets even better results (the condition on the equality of false positives has, on the contrary, very low performances). In an application in which the only concern is about the equality of true positive rate among different classes, then this criterion is even more suitable than Sufficiency. Finally, the algorithms show a slightly different behavior, with K-nearest neighbors outperforming the others in the binary case, and Random Forest and Support Vector Machine being better in the multiclass case. Further work could be done in testing other algorithms and/or trying a different approach by fine-tuning them and seeing if there are any noticeable differences as the algorithm best fits the data.

We hope that these findings will lead researchers and policy-makers toward assessing the risk of discrimination by measuring the imbalance of the protected attributes in the training set: we deem that the adoption and improvement of this approach would help in developing socially sustainable ADM systems.

Appendix A

Predictors and targets

A.1 Credit card default

Predictors:

- LIMIT_BAL: Amount of given credit
- PAY_0: Repayment status in September, 2005 payment delay for eight months, 9=payment delay for nine months and above)
- BILL_ATM1: Amount of bill statement in September, 2005
- PAY_AMT1: Amount of previous payment in September, 2005

Target:

- default.payment.next.month_f: Default payment (1=yes, 0=no)

A.2 Statlog

Predictors:

- Purpose: purpose of the credit (car, furniture, business, etc.)
- Duration: duration in month
- Credit_history: credit history
- Credit_amount: credit amount
- Savings: Savings account/bonds
- Employment_since: present employment since

- Installment_rate: installment rate in percentage of disposable income
- Other_Debtors_Guarantors: other debtors / guarantors
- Property: type of properties
- Housing: type of housing
- Residence_since: present residence since
- Other_installment_plans
- Existing_credits: number of existing credits in this bank
- Job: type of job
- People_liable_to_provide_maintenance_for: number of people being liable to provide maintenance for
- Telephone: present or not

Target:

- costMatrix (good customer:0 , bad customer:1)

A.3 Student (Math and Portugese)

Predictors:

- school:student's school
- address: student's home address type
- famsize: family size
- Pstatus: parent's cohabitation status
- reason: reason to choose this school
- nursery: attended nursery school
- internet: Internet access at home
- studytime: weekly study time
- failures: number of past class failures
- paid: extra paid classes within the course subject

- activities: extra-curricular activities
- nursery: attended nursery school
- higher: wants to take higher education
- freetime: free time after school
- goout: going out with friends
- Dalc: workday alcohol consumption
- Walc: weekend alcohol consumption
- health: current health status
- absences: number of school absences
- guardian: student's guardian
- traveltime: home to school travel time
- famsup: family educational support
- romantic: with a romantic relationship
- famrel: quality of family relationships

Target:

- G3_target: final grade (Less than or equal to 9:0, higher than 9:1)

A.4 Drug consumption (Cannabis)

Predictors:

- Nscore: NEO-FFI-R Neuroticism
- Escore: NEO-FFI-R Extraversion
- Oscore: NEO-FFI-R Openness to experience
- Ascore: NEO-FFI-R Agreeableness
- Cscore: NEO-FFI-R Conscientiousness
- SS: sensation seeing measured by ImpSS

Target:

- Cannabis_target: cannabis consumption (Never Used/Used over a decade ago:0 , Used less than a decade ago:1)

A.5 Drug consumption (Impulsive)

Predictors:

- Alcohol: consumption of alcohol
- Amphet: consumption of amphetamines
- Amyl: consumption of amyl nitrite
- Benzos: consumption of benzodiazepine
- Caff: consumption of caffeine
- Cannabis: consumption of cannabis
- Choc: consumption of chocolate
- Coke: consumption of cocaine
- Crank: consumption of crack
- Ecstasy: consumption of ecstasy
- Heroin: consumption of heroine
- Ketamine: nconsumption of ketamine
- Legalh: consumption of legal highs
- LSD: consumption of lsd
- Meth: consumption of methadone
- Mushrooms: consumption of magic mushrooms
- Nicotine: consumption of nicotine
- VSA: consumption of volatile substances
- Semer: consumption of the fictious drug "Semeron"

Target:

- Impulsive: impulsivess rating (Less than or equal to average:0 , Higher:1)

A.6 Heart disease

Predictors:

- cp: chest pain type
- trestbps: resting blood pressure
- chol: serum cholestoral
- fbs: fasting blood sugar
- restecg: relieved after rest
- thalach: maximum heart rate achieved
- exang: exercise induced angina
- oldpeak: ST depression induced by exercise relative to rest
- slope: the slope of the peak exercise ST segment
- ca: number of major vessels colored by flourosopy
- thal: state of blood disorder called thalassemia

Target:

- Diagnosis: (absence of disease:0 presence:1)

Appendix B

Thresholds tables

B.1 Binary attributes

B.1.1 Gini

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	Gini	logit	3	3,2	97,62	0,68	0,84	0,21	0,75	0,79
Diff.Ind	Gini	svm	5	1,66	95,49	0,68	0,75	0,30	0,86	0,80
Diff.Ind	Gini	rf	3	4,01	80,59	0,54	0,58	0,41	0,76	0,66
Diff.Ind	Gini	knn	3	2,33	96,18	0,65	0,78	0,21	0,77	0,77

Table B.1: Thresholds and sensitivity measures for the couple Gini-Diff.Ind, Binary case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	Gini	logit	3	4,72	94,55	0,67	0,76	0,33	0,80	0,78
Diff.TP	Gini	svm	3	3,99	79,29	0,63	0,64	0,56	0,86	0,73
Diff.TP	Gini	rf	5	3,31	99,99	0,85	0,96	0,07	0,88	0,92
Diff.TP	Gini	knn	5	2,06	99,96	0,80	0,93	0,11	0,85	0,89

Table B.2: Thresholds and sensitivity measures for the couple Gini-Diff.TP, Binary case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	Gini	logit	3	2,89	84,68	0,61	0,68	0,42	0,76	0,72
Diff.FP	Gini	svm	1	6,77	38,44	0,49	0,40	0,58	0,48	0,44
Diff.FP	Gini	rf	1	7,875	54,58	0,61	0,59	0,64	0,55	0,57
Diff.FP	Gini	knn	5	0,91	91,5	0,69	0,74	0,35	0,90	0,81

Table B.3: Thresholds and sensitivity measures for the couple Gini-Diff.FP, Binary case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	Gini	logit	5	2,5	72,02	0,58	0,59	0,50	0,89	0,71
Diff.PP	Gini	svm	5	2,16	85	0,67	0,70	0,46	0,91	0,79
Diff.PP	Gini	rf	4	4,46	96,53	0,74	0,86	0,29	0,82	0,84
Diff.PP	Gini	knn	3	6,1	96,27	0,66	0,85	0,26	0,71	0,77

Table B.4: Thresholds and sensitivity measures for the couple Gini-Diff.PP, Binary case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	Gini	logit	5	2,17	95,77	0,71	0,78	0,21	0,88	0,82
Diff.PN	Gini	svm	5	1,92	95,80	0,72	0,79	0,25	0,88	0,83
Diff.PN	Gini	rf	1	7,835	67,60	0,62	0,71	0,53	0,60	0,65
Diff.PN	Gini	knn	5	2,47	96,41	0,73	0,82	0,25	0,85	0,84

Table B.5: Thresholds and sensitivity measures for the couple Gini-Diff.PN, Binary case.

B.1.2 Shannon

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	Shannon	logit	3	3,2	98,28	0,68	0,84	0,21	0,75	0,79
Diff.Ind	Shannon	svm	5	1,66	96,72	0,68	0,75	0,30	0,86	0,80
Diff.Ind	Shannon	rf	3	4,01	85,51	0,54	0,58	0,41	0,76	0,66
Diff.Ind	Shannon	knn	3	2,33	97,23	0,65	0,78	0,21	0,77	0,77

Table B.6: Thresholds and sensitivity measures for the couple Shannon-Diff.Ind, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	Shannon	logit	3	4,72	96,03	0,67	0,76	0,33	0,80	0,78
Diff.TP	Shannon	svm	3	3,99	83,03	0,62	0,63	0,58	0,87	0,73
Diff.TP	Shannon	rf	5	3,31	99,99	0,85	0,95	0,08	0,88	0,92
Diff.TP	Shannon	knn	5	2,06	99,97	0,80	0,93	0,12	0,85	0,89

Table B.7: Thresholds and sensitivity measures for the couple Shannon-Diff.TP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	Shannon	logit	3	2,89	88,65	0,61	0,68	0,41	0,76	0,72
Diff.FP	Shannon	svm	1	6,77	46,58	0,49	0,36	0,63	0,49	0,41
Diff.FP	Shannon	rf	1	7,88	63,19	0,61	0,58	0,64	0,55	0,57
Diff.FP	Shannon	knn	5	0,91	93,68	0,69	0,74	0,35	0,90	0,81

Table B.8: Thresholds and sensitivity measures for the couple Shannon-Diff.FP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	Shannon	logit	5	2,50	77,06	0,57	0,58	0,50	0,89	0,70
Diff.PP	Shannon	svm	5	2,16	88,89	0,67	0,69	0,45	0,91	0,79
Diff.PP	Shannon	rf	4	4,46	97,48	0,74	0,86	0,29	0,82	0,84
Diff.PP	Shannon	knn	3	6,10	97,30	0,66	0,84	0,26	0,71	0,77

Table B.9: Thresholds and sensitivity measures for the couple Shannon-Diff.PP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	Shannon	logit	5	2,17	96,93	0,71	0,78	0,21	0,88	0,82
Diff.PN	Shannon	svm	5	1,92	96,90	0,72	0,78	0,25	0,88	0,83
Diff.PN	Shannon	rf	1	7,84	73,07	0,62	0,68	0,56	0,61	0,64
Diff.PN	Shannon	knn	5	2,47	97,39	0,73	0,82	0,25	0,85	0,84

Table B.10: Thresholds and sensitivity measures for the couple Shannon-Diff.PN, Binary case

B.1.3 Simpson

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	Simpson	logit	3	3,20	95,35	0,68	0,84	0,21	0,75	0,79
Diff.Ind	Simpson	svm	5	1,66	91,37	0,68	0,75	0,30	0,86	0,80
Diff.Ind	Simpson	rf	3	4,01	67,50	0,54	0,58	0,41	0,76	0,66
Diff.Ind	Simpson	knn	3	2,33	92,65	0,65	0,78	0,21	0,77	0,77

Table B.11: Thresholds and sensitivity measures for the couple Simpson-Diff.Ind, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	Simpson	logit	3	4,72	89,66	0,67	0,76	0,33	0,80	0,78
Diff.TP	Simpson	svm	3	3,99	73,11	0,65	0,70	0,47	0,85	0,77
Diff.TP	Simpson	rf	5	3,31	99,99	0,85	0,96	0,04	0,88	0,92
Diff.TP	Simpson	knn	5	2,06	99,93	0,80	0,93	0,11	0,85	0,89

Table B.12: Thresholds and sensitivity measures for the couple Simpson-Diff.TP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	Simpson	logit	3	2,89	73,43	0,61	0,68	0,41	0,76	0,72
Diff.FP	Simpson	svm	1	6,77	29,43	0,49	0,43	0,56	0,49	0,46
Diff.FP	Simpson	rf	1	7,88	40,32	0,61	0,59	0,63	0,55	0,57
Diff.FP	Simpson	knn	5	0,91	85,47	0,70	0,74	0,35	0,90	0,81

Table B.13: Thresholds and sensitivity measures for the couple Simpson-Diff.FP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	Simpson	logit	5	2,50	63,82	0,58	0,60	0,48	0,89	0,72
Diff.PP	Simpson	svm	5	2,16	73,91	0,67	0,69	0,45	0,91	0,79
Diff.PP	Simpson	rf	4	4,46	93,36	0,74	0,86	0,29	0,82	0,84
Diff.PP	Simpson	knn	3	6,10	92,82	0,66	0,84	0,26	0,71	0,77

Table B.14: Thresholds and sensitivity measures for the couple Simpson-Diff.PP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	Simpson	logit	5	2,17	91,88	0,71	0,78	0,21	0,88	0,82
Diff.PN	Simpson	svm	5	1,92	92,54	0,74	0,81	0,23	0,88	0,84
Diff.PN	Simpson	rf	1	7,84	60,61	0,63	0,74	0,52	0,61	0,67
Diff.PN	Simpson	knn	5	2,47	93,07	0,73	0,82	0,25	0,85	0,84

Table B.15: Thresholds and sensitivity measures for the couple Simpson-Diff.PN, Binary case

B.1.4 IR

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	IR	logit	3	3,20	73,27	0,68	0,84	0,21	0,75	0,79
Diff.Ind	IR	svm	5	1,66	64,96	0,68	0,75	0,30	0,86	0,80
Diff.Ind	IR	rf	3	4,01	38,84	0,54	0,58	0,41	0,76	0,66
Diff.Ind	IR	knn	3	2,33	67,32	0,65	0,78	0,21	0,77	0,77

Table B.16: Thresholds and sensitivity measures for the couple IR-Diff.Ind, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	IR	logit	3	4,72	62,14	0,67	0,76	0,33	0,80	0,78
Diff.TP	IR	svm	4	3,00	51,20	0,70	0,74	0,43	0,89	0,81
Diff.TP	IR	rf	5	3,31	98,29	0,85	0,96	0,04	0,88	0,92
Diff.TP	IR	knn	5	2,06	96,28	0,80	0,93	0,11	0,85	0,89

Table B.17: Thresholds and sensitivity measures for the couple IR-Diff.TP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	IR	logit	3	2,89	43,74	0,61	0,68	0,41	0,76	0,72
Diff.FP	IR	svm	1	6,77	19,31	0,49	0,43	0,56	0,49	0,46
Diff.FP	IR	rf	1	7,88	22,13	0,62	0,60	0,63	0,55	0,57
Diff.FP	IR	knn	5	0,91	70,24	0,76	0,84	0,19	0,89	0,86

Table B.18: Thresholds and sensitivity measures for the couple IR-Diff.FP, Binary case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	IR	logit	5	2,50	43,87	0,65	0,68	0,43	0,89	0,77
Diff.PP	IR	svm	5	2,16	44,16	0,67	0,69	0,45	0,91	0,79
Diff.PP	IR	rf	4	4,46	70,25	0,74	0,87	0,26	0,82	0,84
Diff.PP	IR	knn	3	6,10	67,64	0,66	0,84	0,26	0,71	0,77

Table B.19: Thresholds and sensitivity measures for the couple IR-Diff.PP, Binary Case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	IR	logit	5	2,17	65,88	0,71	0,78	0,21	0,88	0,82
Diff.PN	IR	svm	5	1,92	82,40	0,78	0,87	0,16	0,88	0,88
Diff.PN	IR	rf	1	7,84	52,13	0,58	0,81	0,35	0,55	0,66
Diff.PN	IR	knn	5	2,47	68,13	0,73	0,82	0,25	0,85	0,84

Table B.20: Thresholds and sensitivity measures for the couple IR-Diff.PN, Binary case

B.2 Multiclass attributes

B.2.1 Gini

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	Gini	logit	1	8,45	88,54	0,44	0,33	0,55	0,43	0,38
Diff.Ind	Gini	svm	1	10,08	93,73	0,42	0,47	0,37	0,40	0,43
Diff.Ind	Gini	rf	5	3,61	91,09	0,47	0,47	0,50	0,92	0,62
Diff.Ind	Gini	knn	5	1,69	93,71	0,52	0,55	0,38	0,86	0,67

Table B.21: Thresholds and sensitivity measures for the couple Gini-Diff.Ind, Multiclass case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	Gini	logit	1	10,04	92,23	0,51	0,49	0,54	0,65	0,56
Diff.TP	Gini	svm	2	8,21	94,64	0,56	0,59	0,44	0,80	0,68
Diff.TP	Gini	rf	5	4,55	99,83	0,77	0,84	0,14	0,89	0,87
Diff.TP	Gini	knn	1	9,98	93,50	0,53	0,52	0,55	0,64	0,57

Table B.22: Thresholds and sensitivity measures for the couple Gini-Diff.TP, Multiclass case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	Gini	logit	1,00	8,32	87,58	0,43	0,34	0,56	0,53	0,41
Diff.FP	Gini	svm	5,00	1,46	92,02	0,48	0,48	0,51	0,96	0,63
Diff.FP	Gini	rf	1,00	9,55	91,96	0,48	0,47	0,50	0,55	0,51
Diff.FP	Gini	knn	1,00	8,79	94,10	0,47	0,58	0,35	0,49	0,53

Table B.23: Thresholds and sensitivity measures for the couple Gini-Diff.FP, Multiclass case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	Gini	logit	1	13,29	87,96	0,48	0,40	0,59	0,56	0,46
Diff.PP	Gini	svm	1	11,99	89,61	0,47	0,41	0,56	0,57	0,47
Diff.PP	Gini	rf	5	4,85	95,16	0,59	0,61	0,34	0,93	0,73
Diff.PP	Gini	knn	1	12,88	86,83	0,55	0,39	0,72	0,60	0,47

Table B.24: Thresholds and sensitivity measures for the couple Gini-Diff.PP, Multiclass case

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	Gini	logit	1	12,04	86,67	0,48	0,30	0,69	0,53	0,38
Diff.PN	Gini	svm	3	7,60	98,92	0,59	0,66	0,31	0,80	0,72
Diff.PN	Gini	rf	2	7,72	94,68	0,52	0,57	0,35	0,76	0,65
Diff.PN	Gini	knn	2	8,58	93,05	0,53	0,51	0,62	0,87	0,64

Table B.25: Thresholds and sensitivity measures for the couple Gini-Diff.PN, Multiclass case

B.2.2 Shannon

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	Shannon	logit	3	5,19	84,89	0,48	0,46	0,54	0,76	0,57
Diff.Ind	Shannon	svm	3	6,54	85,52	0,43	0,44	0,42	0,68	0,53
Diff.Ind	Shannon	rf	5	3,61	87,27	0,55	0,58	0,27	0,90	0,70
Diff.Ind	Shannon	knn	5	1,69	86,10	0,53	0,55	0,36	0,86	0,67

Table B.26: Thresholds and sensitivity measures for the couple Shannon-Diff.Ind, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	Shannon	logit	4	5,14	87,50	0,57	0,60	0,41	0,82	0,69
Diff.TP	Shannon	svm	2	8,21	93,80	0,60	0,66	0,35	0,79	0,72
Diff.TP	Shannon	rf	5	4,55	99,79	0,79	0,87	0,14	0,89	0,88
Diff.TP	Shannon	knn	1	9,98	92,80	0,54	0,67	0,33	0,61	0,64

Table B.27: Thresholds and sensitivity measures for the couple Shannon-Diff.TP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	Shannon	logit	1	8,32	83,97	0,47	0,44	0,51	0,57	0,49
Diff.FP	Shannon	svm	5	1,46	89,82	0,58	0,60	0,23	0,95	0,73
Diff.FP	Shannon	rf	3	4,36	86,44	0,52	0,58	0,33	0,74	0,65
Diff.FP	Shannon	knn	3	4,23	87,54	0,52	0,59	0,31	0,71	0,64

Table B.28: Thresholds and sensitivity measures for the couple Shannon-Diff.FP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	Shannon	logit	1	13,29	84,93	0,53	0,50	0,57	0,61	0,55
Diff.PP	Shannon	svm	2	8,20	84,67	0,52	0,50	0,58	0,81	0,62
Diff.PP	Shannon	rf	2	8,72	93,24	0,61	0,68	0,34	0,80	0,74
Diff.PP	Shannon	knn	1	12,88	82,46	0,56	0,48	0,65	0,60	0,53

Table B.29: Thresholds and sensitivity measures for the couple Shannon-Diff.PP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	Shannon	logit	1	12,04	81,04	0,46	0,31	0,64	0,50	0,38
Diff.PN	Shannon	svm	3	7,60	97,16	0,59	0,66	0,31	0,80	0,72
Diff.PN	Shannon	rf	2	7,72	94,02	0,59	0,66	0,34	0,79	0,72
Diff.PN	Shannon	knn	2	8,58	89,19	0,59	0,61	0,46	0,85	0,71

Table B.30: Thresholds and sensitivity measures for the couple Shannon-Diff.PN, Multiclass case.

B.2.3 Simpson

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	Simpson	logit	1	8,45	62,19	0,50	0,47	0,54	0,51	0,49
Diff.Ind	Simpson	svm	3	6,54	66,97	0,43	0,44	0,42	0,68	0,53
Diff.Ind	Simpson	rf	4	5,42	66,00	0,47	0,47	0,50	0,85	0,60
Diff.Ind	Simpson	knn	5	1,69	74,88	0,53	0,55	0,36	0,86	0,67

Table B.31: Thresholds and sensitivity measures for the couple Simpson-Diff.Ind, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	Simpson	logit	3	6,85	77,24	0,55	0,60	0,39	0,76	0,67
Diff.TP	Simpson	svm	2	8,21	84,92	0,60	0,66	0,35	0,79	0,72
Diff.TP	Simpson	rf	5	4,55	99,14	0,79	0,87	0,15	0,89	0,88
Diff.TP	Simpson	knn	1	9,98	83,05	0,54	0,63	0,40	0,62	0,62

Table B.32: Thresholds and sensitivity measures for the couple Simpson-Diff.TP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	Shannon	logit	1	8,32	83,97	0,47	0,44	0,51	0,57	0,49
Diff.FP	Shannon	svm	5	1,46	89,82	0,58	0,60	0,23	0,95	0,73
Diff.FP	Shannon	rf	3	4,36	86,44	0,52	0,58	0,33	0,74	0,65
Diff.FP	Shannon	knn	3	4,23	87,54	0,52	0,59	0,31	0,71	0,64

Table B.33: Thresholds and sensitivity measures for the couple Simpson-Diff.FP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	Simpson	logit	1	13,29	64,35	0,53	0,51	0,57	0,61	0,55
Diff.PP	Simpson	svm	3	8,84	66,23	0,52	0,50	0,57	0,78	0,61
Diff.PP	Simpson	rf	2	8,72	87,44	0,61	0,68	0,34	0,80	0,74
Diff.PP	Simpson	knn	1	12,88	60,97	0,56	0,55	0,57	0,58	0,56

Table B.34: Thresholds and sensitivity measures for the couple Simpson-Diff.PP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	Simpson	logit	1	12,04	53,71	0,47	0,27	0,71	0,51	0,35
Diff.PN	Simpson	svm	4	5,70	84,02	0,64	0,66	0,28	0,94	0,77
Diff.PN	Simpson	rf	2	7,72	85,30	0,59	0,66	0,34	0,79	0,72
Diff.PN	Simpson	knn	3	9,54	68,87	0,53	0,51	0,59	0,81	0,63

Table B.35: Thresholds and sensitivity measures for the couple Simpson-Diff.PN, Multiclass case.

B.2.4 IR

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.Ind	IR	logit	3	5,19	18,61	0,59	0,67	0,35	0,76	0,71
Diff.Ind	IR	svm	4	4,91	10,92	0,56	0,61	0,27	0,83	0,70
Diff.Ind	IR	rf	5	3,61	32,69	0,63	0,66	0,27	0,91	0,77
Diff.Ind	IR	knn	4	2,54	12,08	0,57	0,64	0,30	0,77	0,70

Table B.36: Thresholds and sensitivity measures for the couple IR-Diff.Ind, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.TP	IR	logit	5	3,42	24,12	0,65	0,68	0,41	0,89	0,77
Diff.TP	IR	svm	5	4,44	42,71	0,64	0,66	0,36	0,91	0,77
Diff.TP	IR	rf	5	4,55	80,85	0,80	0,88	0,13	0,89	0,89
Diff.TP	IR	knn	4	4,86	22,47	0,61	0,67	0,34	0,82	0,74

Table B.37: Thresholds and sensitivity measures for the couple IR-Diff.TP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.FP	IR	logit	2	4,83	16,07	0,56	0,67	0,32	0,68	0,67
Diff.FP	IR	svm	5	1,46	56,18	0,65	0,67	0,23	0,95	0,78
Diff.FP	IR	rf	4	3,27	20,81	0,64	0,68	0,34	0,88	0,77
Diff.FP	IR	knn	4	3,17	18,33	0,59	0,66	0,28	0,79	0,72

Table B.38: Thresholds and sensitivity measures for the couple IR-Diff.FP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PP	IR	logit	1	13,29	30,65	0,54	0,68	0,35	0,58	0,63
Diff.PP	IR	svm	4	6,63	15,57	0,63	0,68	0,35	0,86	0,76
Diff.PP	IR	rf	5	4,85	18,36	0,66	0,68	0,34	0,94	0,79
Diff.PP	IR	knn	3	9,40	16,39	0,58	0,68	0,34	0,72	0,70

Table B.39: Thresholds and sensitivity measures for the couple IR-Diff.PP, Multiclass case.

Fairness	Balance	algorithm	configuration	f	s	Accuracy	Sensitivity	Specificity	Precision	F1
Diff.PN	IR	logit	2	8,11	17,71	0,59	0,66	0,30	0,79	0,72
Diff.PN	IR	svm	4	5,70	25,90	0,64	0,66	0,28	0,94	0,77
Diff.PN	IR	rf	5	4,11	46,74	0,65	0,66	0,31	0,95	0,78
Diff.PN	IR	knn	4	7,16	19,22	0,63	0,67	0,35	0,88	0,76

Table B.40: Thresholds and sensitivity measures for the couple IR-Diff.PN, Multiclass case.

Bibliography

[1] F. Chiusi, S. Fischer, N. Kayser-Bril, and M. Spielkamp. *Automating Society Report 2020*. `https://automatingsociety.algorithmwatch.org` (cit. on p. 1).

[2] S. Barocas and A. D. Selbst. *Big Data's Disparate Impact*. `https://papers.ssrn.com/abstract=2477899` (cit. on p. 1).

[3] G. Ristanoski, W. Liu, and J. Bailey. *Discrimination aware classification for imbalanced datasets*. `https://dl.acm.org/doi/10.1145/2505515.2507836` (cit. on p. 1).

[4] *International Organization for Standardization, ISO/IEC 25000:2014 - Systems and software engineering Systems and software Quality Requirements and Evaluation (SQuaRE)*. `https://www.iso.org/standard/64764.html` (cit. on p. 3).

[5] *International Organization for Standardization, ISO 31000:2018 - Risk management Guidelines*. `https://www.iso.org/standard/65694.html` (cit. on p. 3).

[6] A.Vetrò, M.Torchiano, and M.Mecati. *Imbalanced data as risk factor of discriminating automated decisions: a measurement-based approach*. `https://doi.org/10.1016/j.giq.2021.101619` (cit. on p. 4).

[7] S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning*. `http://www.fairmlbook.org` (cit. on p. 9).

[8] *Default of credit card clients Data Set*. `https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients` (cit. on p. 12).

[9] *Statlog (German Credit Data) Data Set*. `https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)` (cit. on p. 12).

[10] *Student Performance Data Set*. `https://archive.ics.uci.edu/ml/datasets/student+performance` (cit. on p. 13).

[11] *Census Income Data Set*. `https://archive.ics.uci.edu/ml/datasets/census+income` (cit. on p. 13).

[12] *Drug consumption (quantified) Data Set.* https://archive.ics.uci.edu/ml/datasets/Drug+consumption+%28quantified%29 (cit. on p. 13).

[13] *Heart Disease Data Set.* https://archive.ics.uci.edu/ml/datasets/heart+disease (cit. on p. 14).

[14] *R stats package.* https://www.rdocumentation.org/packages/stats/versions/3.6.2 (cit. on p. 15).

[15] *R e1071 package.* https://www.rdocumentation.org/packages/e1071/versions/1.7-9 (cit. on p. 15).

[16] *R randomForest package.* https://www.rdocumentation.org/packages/randomForest/versions/4.7-1.1/topics/randomForest (cit. on p. 15).

[17] *R class package.* https://www.rdocumentation.org/packages/class/versions/7.3-20 (cit. on p. 15).

[18] *R ROSE package.* https://www.rdocumentation.org/packages/ROSE/versions/0.0-4 (cit. on p. 15).

[19] *R UBL package.* https://www.rdocumentation.org/packages/UBL/versions/0.0.6 (cit. on p. 15).

[20] *Confusion Matrix.* https://subscription.packtpub.com/book/data/9781838555078/6/ch06lvl1sec34/confusion-matrix (cit. on p. 17).

[21] *Machine Bias - Risk assessment in criminal sentencing.* https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing (cit. on p. 42).