Faculty of Sciences and Technology

Department of Informatics Engineering

Assessing the Fairness of Intelligent Systems

Inês Filipa Rente Valentim

Dissertation in the context of the Master in Informatics Engineering, Specialisation in Intelligent Systems, advised by Prof. Dr. Nuno Antunes and Prof. Dr. Nuno Lourenço and presented to the Faculty of Sciences and Technology / Department of Informatics Engineering.

July 2019

UNIVERSIDADE D
COIMBRA

This work is within the intelligent systems specialisation area and was carried out in the Software and Systems Engineering (SSE) Group of the Centre for Informatics and Systems of the University of Coimbra (CISUC).

This work is partially supported by the project ATMOSPHERE, funded by the Brazilian Ministry of Science, Technology and Innovation (51119 - MCTI/RNP 4th Coordinated Call) and by the European Commission under the Cooperation Programme, H2020 grant agreement no `777154`.

It is also partially supported by the project METRICS (`POCI-01-0145-FEDER-032504`), co-funded by the Portuguese Foundation for Science and Technology (FCT) and by the *Fundo Europeu de Desenvolvimento Regional* (FEDER) through *Portugal 2020 - Programa Operacional Competitividade e Internacionalização (POCI)*.

This work has been supervised by Professor Nuno Manuel dos Santos Antunes and Professor Nuno António Marques Lourenço, Assistant Professors at the Department of Informatics Engineering of the Faculty of Sciences and Technology of the University of Coimbra.

This page is intentionally left blank.

Dedicated to my grandfather.

This page is intentionally left blank.

Acknowledgements

First of all, I would like to thank my advisors, Professor Nuno Antunes and Professor Nuno Lourenço, for their constant support and guidance throughout this last year. Thank you for always being ready to help me whenever I asked for your advice and for teaching me how to be a better person, both professionally and personally.

To João Campos and Noé Godinho, a special thank you for always being free to discuss anything related to this work and for offering your help willingly throughout this whole process. To Ana Duarte and Kevin, thank you for making me laugh and cheering me up, for having dinner with me whenever I felt down, and for bringing me food when I was the most stressed. The last few years of my academic life would have been much different without you.

Mariana Cunha, and especially Pedro Costa, you were there for me from the beginning and you are two of the most important people I had the opportunity to meet at the University. I had the chance to learn with you, thank you for inspiring me more than you know.

I would also like to thank all the people I had the chance to meet while working at the lab, especially José Pereira, Rui, and José Flora. You always managed to make me laugh even when everything around us appeared to be a complete chaos.

To Luís, thank you for being my main source of support during the last few months. Without you, I would have not been able to complete this journey. Thank you for keeping me company when you needed to rest, for encouraging me and cheering me up when I felt down and stressed, thank you for reviewing my work and believing in me.

Last but not the least, I want to thank my father and my grandmother. You have always been there for me when I needed the most, calming me down and giving me strength in the most stressful situations. I owe everything I have accomplished to you. Thank you for your patience and love.

This page is intentionally left blank.

Abstract

Nowadays, software systems based on Machine Learning models are ubiquitous, often being used in scenarios that directly affect people's lives. Consequently, societal and legal concerns arise, namely that decisions supported by the models' outputs may lead to the unfair treatment of individuals, based on attributes like race, age, or sex. In fact, fairness is one of the properties systems must have to be compliant with current legislation, namely the EU General Data Protection Regulation.

The main objective of this work is to assess the fairness of software systems based on Machine Learning models in classification scenarios. Data preparation and pre-processing are key on any Machine Learning pipeline, and their effect on fairness needed to be studied in detail. Thus, we assessed the impact of the encoding of the categorical features, the removal of the sensitive attribute from the training data, as well as sampling methods, such as random undersampling and random oversampling. The influence of the learning algorithm was also considered, with an initial evaluation of Decision Trees and Random Forests. Fairness was measured at different stages of the pipeline to understand the procedures with the most impact on it.

Our results show that performing sampling with respect to the true labels and opting for Random Forests over Decision Trees often has a negative effect on fairness. Although removing the sensitive attribute from the training data prevents incurring in direct discrimination, the models are often still able to explore associations between this attribute and the remaining features, with the resulting classifications sometimes even being more unfair than the data. As a result, organisations must be aware of and carefully assess the trade-off between classification performance and fairness.

Keywords

Decision Making, Discrimination, Fairness, Intelligent Systems, Machine Learning

This page is intentionally left blank.

Resumo

Atualmente, os sistemas de software baseados em modelos de Aprendizagem Computacional são ubíquos, sendo muitas vezes usados em cenários que afetam diretamente a vida das pessoas. Consequentemente, surgem diversas preocupações sociais e legais, nomeadamente que as decisões suportadas pelos resultados dos modelos possam levar ao tratamento menos favorável de alguns indivíduos, com base em atributos como raça, idade, ou sexo. Na realidade, a *fairness* é uma das propriedades que os sistemas devem possuir para que cumpram legislação atual, tal como o Regulamento Geral sobre a Proteção de Dados da UE.

O objetivo principal deste trabalho é avaliar a *fairness* de sistemas baseados em modelos de Aprendizagem Computacional, em problemas de classificação. A preparação e o pré-processamento de dados são fulcrais em qualquer *pipeline* de Aprendizagem Computacional, sendo que era necessário estudar o seu efeito em termos de *fairness*. Nesta perspetiva, avaliámos o impacto do *encoding* de atributos categóricos, a remoção do atributo sensível dos dados de treino, e mecanismos de amostragem, como *random undersampling* e *random oversampling*. A influência do algoritmo de aprendizagem foi também tida em conta, sendo avaliadas Árvores de Decisão e *Random Forests*. Medimos a fairness em diferentes etapas do *pipeline* para compreender os fatores com maior impacto nesta propriedade.

Os resultados mostram que fazer uma amostragem de acordo com o *output* esperado e optar por *Random Forests* em vez de Árvores de Decisão tende a ter efeitos negativos na *fairness*. Embora a remoção do atributo sensível dos dados de treino elimine a discriminação direta, os modelos são ainda assim capazes de explorar associações entre este atributo e os restantes, sendo que algumas vezes as classificações acabam mesmo por ser mais injustas que os próprios dados. Desta forma, é necessário que as organizações estejam cientes deste compromisso entre desempenho e fairness, avaliando-o de forma cuidada.

Palavras-Chave

Aprendizagem Computacional, Discriminação, *Fairness*, Sistemas Inteligentes, Tomada de Decisão

This page is intentionally left blank.

Contents

This page is intentionally left blank.

Acronyms

AI Artificial Intelligence. 5

ANN Artificial Neural Network. 9, 10, 58

AUC Area Under the Curve. 12

BCR Balanced Classification Rate. 12, 16, 26, 48, 50, 51, 53

BER Balanced Error Rate. 15

COMPAS Correctional Offender Management Profiling for Alternative Sanctions. 1, 2, 24, 25, 32–34

CVS Calders-Verwer Score. 15

DI Disparate Impact. 15, 16, 26, 35–42, 49, 50, 56

FN False Negative. 11

FP False Positive. 11

FPR False Positive Rate. 12, 13, 17

GAN Generative Adversarial Network. 18

GDPR General Data Protection Regulation. 2

GE Generalised Entropy Index. 17, 26, 39–42, 48, 49, 53, 56

HUD US Department of Housing and Urban Development. 1

LDA Linear Discriminant Analysis. 8

ML Machine Learning. 1–3, 5–8, 13, 18, 20, 21, 24, 29, 30, 32, 35, 54–58

NMI Normalised Mutual Information. 29, 32

NPI Normalised Prejudice Index. 15, 26, 35, 36, 38–42, 45, 47–49, 53

PCA Principal Component Analysis. 8

PI Prejudice Index. 15

RO Random Oversampling. 25

ROC Receiver Operating Characteristic. 12, 20

RU Random Undersampling. 25

SPD Statistical Parity Difference. 15, 16, 26, 35, 36, 38–42, 45, 47–49, 56

SVM Support Vector Machine. 9, 19

TN True Negative. 11

TNR True Negative Rate. 12

TP True Positive. 11

TPR True Positive Rate. 11, 16, 17

List of Figures

This page is intentionally left blank.

List of Tables

This page is intentionally left blank.

Chapter 1

Introduction

Software systems powered by Artificial Intelligence, especially by Machine Learning (ML) models, are being used at an increasingly higher rate on a multitude of scenarios that significantly impact people's lives. These scenarios range from loan and mortgage approvals, to hiring and recruiting, or even criminal risk assessment [Zaf+17b; WVP; Cel+; LJ]. Nevertheless, there are many risks associated with a careless adoption of such systems, particularly because one may tend to blindly trust their outcomes as they are made by a machine instead of a person. Wrong decisions in those scenarios may have irreversible and drastic consequences in someone's life, such as being incarcerated for committing minor crimes or not having access to a job opportunity.

In fact, the ubiquity of systems based on ML raises several societal and legal concerns, namely that the decisions supported by the models' outputs may introduce or perpetuate historical bias against certain groups or individuals, based on their intrinsic characteristics, such as race, sex or age. These are often referred to as protected or sensitive characteristics, and discriminating against someone based on these attributes is usually prohibited by law.

There is no denying that the gains associated with automated decision-making systems often make their usage appealing. Using the USA as an example, which has problems with the overpopulation of its prisons, it is important to make an informed decision, without bias, on who should be sent to prison and for how long [Ang+16]. A software system may be capable of providing faster and more accurate predictions about a defendant's risk of recidivism, and its adoption may even be perceived as a step forward in the eradication of personal bias from the process.

Continuing with the example of the USA's legal system, the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) tool is already being used in courtrooms around the country. Criminal recidivism risk scoring has been receiving a lot of attention since the introduction of this tool, partially due to the study published by ProPublica on the potential racial bias in the tool's predictions [Ang+16]. They found that African-American individuals were mistakenly labelled as potential recidivists at an higher rate than Caucasian individuals, while the latter were mislabelled with a lower risk of re-offending at an higher rate than African-American defendants [Ang+16].

Many examples can be provided in which the fairness of the underlying systems is being questioned. In March 2019, the US Department of Housing and Urban Development (HUD) announced that it was charging Facebook with violating the Fair Housing Act by restricting who can see housing ads on the platform, based on protected characteristics like race, sex, and nationality [HUD19; BTI19]. In [Ali+], the authors present further evidence on how

Facebook's ad delivery might be biased. Once again, systems based on ML models are in the spotlight.

In this context, fairness emerges as a key property in terms of the reliability and trustworthiness of software systems based on Machine Learning. Albeit not being new, these issues have been receiving increased attention from regulatory institutions, with the recently approved European Union General Data Protection Regulation (GDPR) demanding organisations to handle personal data in a privacy-preserving, fair and transparent manner [Eur16]. Furthermore, many organisations and governments have already acknowledged that bias introduced by ML models might worsen the social imbalance found in our societies [Cel+; WVP; MSP16].

The development of techniques to assess fairness and build models capable of providing fairer predictions is of great help to organisations which intend to be GDPR compliant, but may lack the needed resources or knowledge [Ant+18]. These organisations must be aware of the potential biases in their models at the design, implementation and deployment phases, and should make regular fairness evaluations of their systems [AA17]. Moreover, the assessment approaches may be used to audit non-compliant organisations, therefore providing valuable insight on violations of these fairness principles [Ant+18]. Individuals who rely on these organisations also benefit from the deployment of fairness-aware models and the adoption of such practices, since they provide an extra assurance that no personal data is being used in abusive ways that may negatively impact their daily lives.

One of the main challenges when working on the fairness of software systems is the lack of an universal and objective definition of this property. As a result, many fairness conditions and metrics have been proposed, each stemming from different notions of how a fair ML model should behave. As suggested by [Fri+19], a comprehensive comparison of different models, built for different datasets and evaluated under various fairness metrics, could give insightful information about the robustness of those metrics. One could also understand the algorithms' sensitivity to the choices made during the design of a system powered by ML. Previous work tends to focus on modifying the data prior to training the models or on proposing changes to the learning algorithms. However, to the best of our knowledge, an analysis of the impact of standard procedures, applied at different stages of the ML cycle, on the system's fairness is lacking in the literature.

Our main goal is to assess the fairness of software systems based on Machine Learning in classification scenarios. We first evaluate the impact of data preparation and data preprocessing techniques on the fairness of these systems. More precisely, we consider the removal of the sensitive attribute, the encoding of the categorical attributes, and instance selection methods, like sampling. Our analysis is complemented by an evaluation of the influence of the learning algorithm on fairness. From the many algorithms suitable for a supervised classification setting, we first focus on tree-based methods, like Decision Trees and Random Forests, partly due to the easier interpretability of the resulting models.

We designed a set of experiments using three widely used datasets, whose fairness concerns have been studied before: the Adult Income, the German Credit Data, and the COMPAS dataset. Several fairness metrics are considered when performing this evaluation. For fairness metrics that can be applied to the training data, we measure fairness not only at the models' outputs, but also at the data-level. By comparing both measurements, we aim at better understanding the behaviour of the learning algorithms when trained with data with different degrees of (un)fairness. Moreover, we also analyse the trade-off between fairness and the system's overall performance.

The obtained results suggest that caution must be taken when dealing with datasets which show an imbalance with respect to both the true labels and the sensitive attribute. Furthermore, standard sampling methods, such as random undersampling with respect to the true labels, may have undesired effects on fairness. Additionally, the results highlight the importance of adopting the standard legal practices to mitigate discrimination, namely the removal of the sensitive attribute prior to training. However, we found that this procedure might not always lead to the expected behaviour, with the models' classifications sometimes being more unfair than when the model has access to the sensitive attribute. We also report the drawbacks of using more complex learning algorithms, with Random Forests making more discriminatory classifications than Decision Trees.

1.1 Contributions

The main contributions of this work are summarised below:

- Proposal of an experimental methodology to evaluate fairness at different stages of a Machine Learning pipeline, according to different notions of fairness. The proposed methodology contemplates the measurement of fairness both at the data-level and when the classifications are known, allowing to pinpoint the steps which have more impact on fairness. By taking several fairness metrics into account, we can assess whether a certain configuration is likely to produce better results, regardless of the fairness notion from which the metric is derived.

- Analysis of data preparation and pre-processing techniques from a fairness point-of-view. Prior to this work, there were limited studies on the impact that these procedures may have on the fairness of a software system based on ML models.

- We show that standard Machine Learning procedures, namely sampling methods to deal with data imbalance, may have undesired effects on the fairness of a software system. Performing sampling with respect to the true labels may lead to more unfair classifications than performing no sampling at all.

- The complexity of the learning algorithm may significantly impact the fairness in the classifications made by a model. The obtained results suggest that Random Forests are likely to produce more unfair predictions than Decision Trees. However, the former also allow for the improvement of the classification performance. Organisations planning on deploying such models into production must consider this trade-off between classification performance and fairness.

1.2 Dissertation Structure

In the first chapter we introduced the problem addressed by this work and presented the main contributions. The remainder of this document is organised as follows.

Chapter 2 provides an overview of key Machine Learning concepts and algorithms, with a focus on classification problems. In this chapter, we also review the related work on fairness of intelligent systems based on ML models, in terms of fairness notions and metrics, as well as techniques which were proposed to mitigate bias in such systems.

Chapter 3 presents the research objectives and details the experimental methodology, including the datasets used in our experiments and the procedures whose impact on fairness we aim at evaluating.

In **Chapter 4**, we perform an exploratory data analysis of the three datasets used in our experiments, giving an emphasis to the imbalance and unfairness found on the training set of each of these datasets, and on the relations between features and true labels, which may play an important role in terms of the fairness of the classifications made by the models.

Chapter 5 presents and discusses the results of our experimental campaign. We break our analysis into different sections, namely we analyse the impact of the procedures we are evaluating on the fairness of the training data, we analyse their effect on the classifications, and compare the unfairness in the classifications made by the trained models to that in the training data. Finally, we analyse the trade-offs between fairness and performance. The discussion is made bearing in mind the research questions presented in Chapter 3.

Chapter 6 gathers the main conclusions and lessons learned of this work, and puts forward possible paths for future work.

Chapter 2

Background and Related Work

Research on the fairness of software systems has been increasing in the last few years, with companies like IBM, Microsoft, and Google investing in this area and recognising the new challenges posed by the usage of Machine Learning (ML) to make and support decisions that directly impact people's lives. In fact, there is a growing community of researchers whose focus is on the Fairness, Accountability, and Transparency of systems based on ML (FAT*), with several courses, conferences, and workshops on the topic being held annually.

This chapter provides an overview of Machine Learning in terms of the most relevant algorithms and techniques for this work. We focus on data preparation and pre-processing procedures, learning algorithms suitable for classification problems, as well as methods and metrics used to assess classification performance. Furthermore, the chapter also includes an introduction to fairness concepts and concerns in the scope of intelligent systems, such as the distinction between direct and indirect discrimination. A review of previous work in this topic is also included, with a special focus on fairness conditions and metrics, as well as approaches to mitigate the bias of the models' classifications.

2.1 Machine Learning

ML is a sub-area of Artificial Intelligence (AI) which comprises a set of methods that enable computer systems to learn from data, without being explicitly programmed to solve some task, as described by Arthur Samuel, in 1959. These systems should be able to improve their learning over time autonomously, without human intervention [Sky], and use this knowledge to make accurate predictions given new observations [Mur12]. Figure 2.1 shows the positioning of ML within Computer Science. Being an interdisciplinary field, it shares concepts and addresses problems also known to statistics, information theory, game theory, and optimisation [SB14].

Different problems may be categorised into different **types of learning**, namely: supervised, unsupervised, and reinforcement learning. The distinction between supervised and unsupervised learning is made depending on the available information in the training data, while reinforcement learning can be considered as an intermediate type of ML [SB14], but falls outside the scope of our work.

In **predictive** or **supervised learning**, the training examples include not only the features (or attributes), but also the target outputs which are used to guide the learning process [Bis06; HTF09]. A further distinction can be made based on the form of the

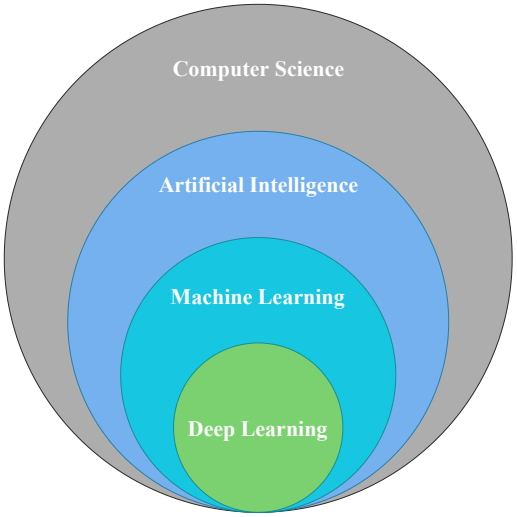


Figure 2.1: Relation of Machine Learning to other fields.

desired output (or response variable): in **classification problems**, it corresponds to a discrete value within a range of possible categories, whereas in **regression** the response variable is continuous [Mur12].

In **descriptive** or **unsupervised learning**, we do not have access to the outputs and the aim is to find associations and patterns within the data [Bis06; HTF09]. **Clustering** is a typical example of an unsupervised learning problem in which the goal is to find groupings of similar examples in the data [Bis06].

By interacting with the environment, the aim of **reinforcement learning** problems is to learn mappings of situations to actions, while maximising a reward. In contrast to supervised learning, the optimal actions are not given to the learning algorithm, but instead, it has to discover them through a trial-and-error process. A current action has an influence not only on the immediate reward, but also on all subsequent rewards. Furthermore, the actions taken by the model have consequences to its later inputs [Bis06; SB18].

A system which relies on ML usually follows a pipeline similar to the one shown in Figure 2.2: after the data is collected, it goes through a set of data preparation and pre-processing steps, followed by the model selection and assessment phases.

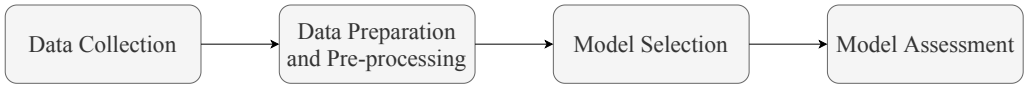


Figure 2.2: A typical Machine Learning pipeline.

The **data collection** phase includes gathering representative data for the problem we are trying to solve, as well as labelling the training examples when in the presence of a supervised learning task.

The **data preparation and pre-processing** steps may include handling missing data, encoding categorical features, discretisation, feature normalisation, feature selection and feature reduction techniques. Not only is the application of these techniques of pivotal importance for some models to deliver good results, but it also helps dealing with overfitting, a common problem in ML which is described in section 2.1.3.

Model selection deals with the process of selecting the most appropriate model for the problem we are trying to solve, taking the complexity and flexibility of the models into account [Jam+13b].

Model assessment deals with evaluating the performance of the chosen model by estimating its generalisation error on new unseen data [HTF09; Jam+13b]. The methods used to address these phases of the pipeline are further discussed in section 2.1.3.

2.1.1 Data Preparation and Pre-processing

Before a dataset can be used to train an ML model, it must go through a series of steps that not only help to deal with common problems in the data, but also allow it to be in the desired format. Such problems include features with mixed characteristics (e.g. categorical and real), and missing or invalid values. Furthermore, data pre-processing is crucial for some classifiers to deliver the expected results, while also playing an important role when it comes to avoiding overfitting.

It is often the case that the training data contains instances with **missing data**. There are several possibilities when it comes to overcoming this problem. The simplest approach is to simply discard instances containing at least one missing value. In scenarios in which we only have limited data available, we may not be able to afford just ignoring instances and potentially loosing valuable information for the task at hands. In such cases, missing data imputation procedures are more suitable and basically consist of filling the missing values [LR02]. These procedures include mean imputation, hot deck imputation, and regression imputation [LR02]. Imputation of the most common value of the feature is also a possibility. Additionally, there are model-based procedures to deal with missing data, in which a model is learned with the available data and inferences are made based on the distributions under the model [LR02]. Refer to [LR02] for more details on how missing data can be handled.

Feature normalisation is also an important step since some classifiers, such as linear regression, are sensitive to features with different scales [SB14]. There are many ways to perform feature scaling, such as centering (make a feature have zero mean), unit range (make the range of the feature be $[0, 1]$), and standardisation [SB14]. Standardisation (also known as Z-score normalisation) makes all features have zero mean and unit variance, by removing the mean value of each feature and then dividing all features by their standard deviations [SB14].

It might also be necessary to discretise continuous features to reduce the number of possible values they can take. Furthermore, this is an important step if one can only work with categorical features. After **discretisation**, one is able to **encode the categorical features** using different representations. In cases where the original feature is nominal, a possibility is to simply represent each possible value by an integer. Caution should be taken since nominal features might not be ordinal, and so, a permutation of the integer assigned to each possible category should not lead to changes when training a model. Another possibility is to apply one-hot encoding, which results in each feature being represented by dummy binary variables, with only one of these dummy variables being given a value of 1.

Feature selection is performed based on the principle that most datasets usually encompass irrelevant and highly correlated features. In order to perform feature selection, one should assess the discriminative capability of the features, as well as determine the existing associations between features. By ranking the features with a Kruskal Wallis test we can assess the features which are more discriminative. We can use the Pearson's correlation

coefficient or mutual information to determine which features share the most information with one another. Feature selection procedures do not alter the features themselves. On the other hand, **feature reduction** techniques project the original data to a lower dimensionality space [Jam+13a]. Linear combinations of the features and Principal Component Analysis (PCA) are common feature reduction techniques [Jam+13a], together with Linear Discriminant Analysis (LDA). These are important steps that help dealing with overfitting and the curse of dimensionality problem.

An **imbalanced dataset** has many more instances from some classes than others and poses challenges when trying to learn an ML model. In such scenarios, under-represented classes tend to be ignored in favour of the larger ones [CJK04]. One way to deal with this problem is **sampling**. Random undersampling randomly picks instances from the majority class to be discarded, while random oversampling duplicates instances from the minority classes with replacement. Both approaches have some drawbacks: with random undersampling we may be loosing important information, while random oversampling may lead to overfitting [CJK04]. More advanced techniques include SMOTE [Cha+02] and ADASYN [Hai+08], both being oversampling techniques. In [BPM04], the authors proposed combining undersampling and oversampling with the SMOTE+Tomek and SMOTE+ENN approaches.

2.1.2 Supervised Algorithms

There are several well-known algorithms when it comes to supervised learning tasks. We will focus on classification problems and detail some of the most relevant algorithms for this work. In this set we include both white-box and black-box algorithms, the latter raising more challenges when it comes to the interpretability of their inner workings and results.

- **Naive Bayes** are a set of probabilistic classifiers which apply the Bayes' theorem and simplify the structure of the model by making strong independence assumptions between features [Bis06]. More precisely, this set of classifiers assumes that each pair of features is independent, conditioned on the target output. Thus, the classification rule becomes:

$$\arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i|y) \tag{2.1}$$

  where $\mathbf{x} = (x_1, ..., x_n)$ is a feature vector and y is the class variable [SB14]. The difference between the classifiers derives from the assumptions made with respect to the likelihood of the features.

- **Logistic Regression** is a linear model for classification which uses the logistic function to model the probabilities of the possible outcomes [Ped+11]. For a binary classification problem we have:

$$p(y_i = 1|\mathbf{x}_i, \mathbf{w}) = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x}_i}} = \sigma(\mathbf{w}^\top \mathbf{x}_i) \tag{2.2}$$

  where $\sigma(\cdot)$ is the logistic function, $\mathbf{x}_i$ is a feature vector and the parameters $\mathbf{w}$ are estimated by solving a maximum likelihood problem using the available training data [Bis06; Zaf+17b]. Therefore, the cost function corresponds to:

$$-\sum_{i=1}^{N} \ln p(y_i|\mathbf{x}_i, \mathbf{w}) = -\sum_{i=1}^{N} y_i \ln \sigma(\mathbf{w}^\top \mathbf{x}_i) + (1 - y_i) \ln[1 - \sigma(\mathbf{w}^\top \mathbf{x}_i)] \tag{2.3}$$

where $y_i \in \{0, 1\}$ [Bis06]. To avoid overfitting, a penalty term (also called regulariser) can be added to the cost function. Common choices are L2 or L1 regularisation.

- **Support Vector Machines (SVMs)** map the data to a higher dimensionality space where the transformed data is linearly separable. By taking advantage of the kernel trick, we do not need to explicitly specify the feature space mappings [Bis06]. Being margin classifiers, SVMs try to maximise the distance between the separating hyperplane and the instances of any class. In general, the larger the margin, the lower the generalisation error of the model [Jam+13b]. Common choices for the kernel function include the linear kernel, the polynomial kernel of degree d, and the radial basis kernel [HTF09]. By introducing slack variables, soft margin SVMs allow some of the training examples to be misclassified, which helps avoid overfitting, contrary to hard margin SVMs, which require that all training points are correctly classified [Bis06]. The cost parameter C trades-off the penalty of misclassifications and the shape of the margin.

- **Decision Trees** are classifiers which try to learn simple decision rules from the available features in the training data [Ped+11]. These are white-box models which usually provide easily interpretable prediction results. A classification tree is built by following a recursive binary splitting process guided by a criterion which evaluates the quality of the splits [Jam+13b]. Common choices for this criterion include the classification error rate, the Gini index and cross-entropy [Jam+13b]. Tree pruning can be used to avoid overfitting. Some of the most well-known decision tree algorithms include Iterative Dichotomiser 3 (ID3) and C4.5.

- **Random Forests** are collections of decision trees where the final prediction is given by a majority vote over the predictions of all the trees in the ensemble [SB14]. To reduce the correlation between the trees, the candidates for splitting are randomly selected from the full set of input features before each split [HTF09]. This randomisation process also aims at reducing variance [Jam+13b].

- **Artificial Neural Networks (ANNs)** are non-linear statistical models that take inspiration from the neural networks in the brain [SB14]. The basic computing element of an ANN is called an artificial neuron. A network essentially consists of multiple of these neurons connected to one another, forming a directed weighted graph [SB14]. In a feedforward neural network, as shown in Figure 2.3, the underlying graph is acyclic [SB14].

  We usually assume that the network is organised by layers, where each layer corresponds to a disjoint subset of nodes (neurons) [SB14]. The input layer has no predecessors, the output layer has no successors, and the number of hidden layers is variable.

  The input of a neuron is given by the weighted sum of the outputs of the neurons connected to it [SB14]. This input then goes through a non-linear activation function to produce the neuron's output. Some of the most common activation functions include the logistic function, the hyperbolic tangent, and the rectifier function [Bis06].

  The learning process of an artificial neural network consists of tuning the weights of the connections between the neurons in order to minimise the chosen loss function. Two stages can be identified in this process: in the first stage, an evaluation of the derivatives of the loss function with respect to the weights is made, while in the second stage, the derivatives are used to compute the weight updates [Bis06]. Several algorithms can be used to train an ANN, namely the backpropagation algorithm can

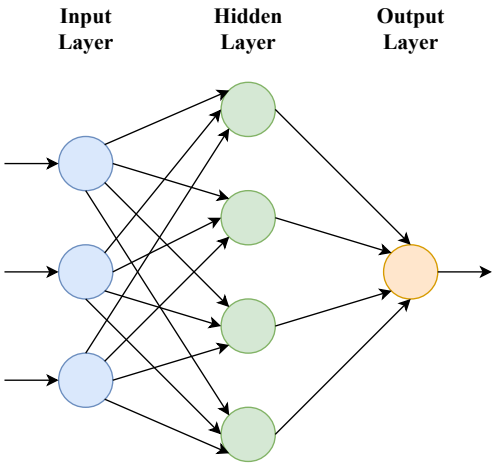


Figure 2.3: An example of a feedforward ANN with a single hidden layer.

be used in the first stage of the process to evaluate the derivatives of a feedforward network, and gradient descent can be used in the second stage of the process [Bis06].

Several approaches have been suggested to help deal with overfitting in an ANN, some of which include: early stopping [Bis06], weight decay [Bis06], and dropout layers.

2.1.3 Model Selection and Assessment

We have to deal with the inherent **bias-variance trade-off** when learning a statistical model. Bias represents the error that results from the inability of the model to represent the problem due to its simplicity [Jam+13b], while variance measures the sensitivity of the model to the dataset used to fit it [Bis06]. Complex models tend to have small bias but high variance, whereas simple models have large bias but small variance. Our goal is to find the model which best balances this trade-off between bias and variance.

This trade-off is also closely related to overfitting/underfitting. **Overfitting** is more likely to happen with models which exhibit high variance and can be defined as fitting the noise in addition to the data, while **underfitting** means that the model does not fit the data well.

To choose a model for the problem we are trying to solve, we need to be able to assess its generalisation performance, which is related to the model's capability of making accurate predictions given new unseen samples [HTF09]. If there is sufficient data, the best approach is to split the dataset into three different sets: a **training set**, used to fit the models; a **validation set**, used to estimate the performance of different models so as to choose the best one; and a **test set**, used to assess the generalisation error of the chosen model [HTF09].

However, it often happens that there is not enough data to split the dataset into these three different parts, making it impossible to set aside a validation set. Several methods have been proposed to address this situation:

- The **hold-out method** divides the available data into two sets. The training set is used to fit the model, which is then used to make predictions for the hold-out set [Jam+13b].

- **K-fold cross-validation** randomly splits the available data into K groups (or folds) of approximately the same size [Jam+13b]. The data of $K-1$ folds is used to fit the model, which is then used to make predictions on the remaining group. This procedure is repeated K times, so that each fold is used to estimate the prediction error exactly once. We then average the results to get an estimate of the generalisation error of the model. Typical values for K are 5 or 10.

- **Leave-one-out cross-validation** is a particular case of K-fold cross-validation where, at each round, only one sample is used for estimating the prediction error, while the remaining samples are used to train the model. This approach typically has low bias but high variance [HTF09].

Different metrics can be used to evaluate the performance of the models. In what follows, we will assume binary classification problems, meaning that the discrete output corresponds to one of two possible classes. Although not explicitly shown here, the performance metrics presented in this section can be adapted to multiclass problems.

A **confusion matrix** summarises the results of a classification problem in a tabular format, where rows correspond to the true class and columns correspond to the predicted class. Each sample may fall in one of four possible classification results: a True Positive (TP) is a positive sample correctly classified; a False Negative (FN) is a positive sample incorrectly classified; a False Positive (FP) is a negative sample incorrectly classified; and a True Negative (TN) is a negative sample correctly classified. A representation of a confusion matrix is shown in Table 2.1.

		Predicted Class	
		Positive	Negative
True Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Table 2.1: Confusion matrix for a binary classification problem.

From the confusion matrix and the four possible outcomes, we can define and compute several performance metrics, including: accuracy, precision and recall, sensitivity and specificity, and F1-score. The definitions of these metrics can be found in [Mar14].

- Even though **accuracy** is a widely used metric to evaluate the performance of an algorithm, it may lead to misleading results in imbalanced scenarios and when incorrect classifications have a different cost. It is given by the ratio between correctly classified instances and the total number of instances:

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{2.4}$$

- **Precision** is given by the fraction of samples classified as positive that are correctly classified:

$$precision = \frac{TP}{TP + FP} \tag{2.5}$$

- **Recall**, also known as **True Positive Rate (TPR)** or **sensitivity**, is given by the fraction of positive samples that are correctly classified:

$$recall = \frac{TP}{TP + FN} \tag{2.6}$$

- The **F1-score** is given by the harmonic mean of precision and recall:

$$F1\text{-}score = 2 \times \frac{precision \times recall}{precision + recall} = \frac{2 \times TP}{2 \times TP + FN + FP} \tag{2.7}$$

- **Specificity**, also called **True Negative Rate (TNR)**, is usually used in medical scenarios alongside sensitivity and is given by the fraction of negative samples that are correctly classified:

$$specificity = \frac{TN}{TN + FP} \tag{2.8}$$

- The **False Positive Rate (FPR)** is given by the fraction of the negative samples that are incorrectly classified:

$$FPR = \frac{FP}{TN + FP} = 1 - specificity \tag{2.9}$$

- The **Balanced Classification Rate (BCR)** is similar to accuracy, but is given by the mean between the TPR and the TNR:

$$BCR = \frac{1}{2} \times \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) = \frac{TPR + TNR}{2} \tag{2.10}$$

The **Receiver Operating Characteristic (ROC) curve** is also commonly used to evaluate the performance of an algorithm, depicting the trade-off between costs and benefits. It plots the recall against the FPR, as some threshold parameter of the classifier is varied. From the ROC curve it is possible to compute the **Area Under the Curve (AUC)**, which is a single quantitative summary of the performance of an algorithm [HTF09]. Figure 2.4 shows an example of a ROC curve.

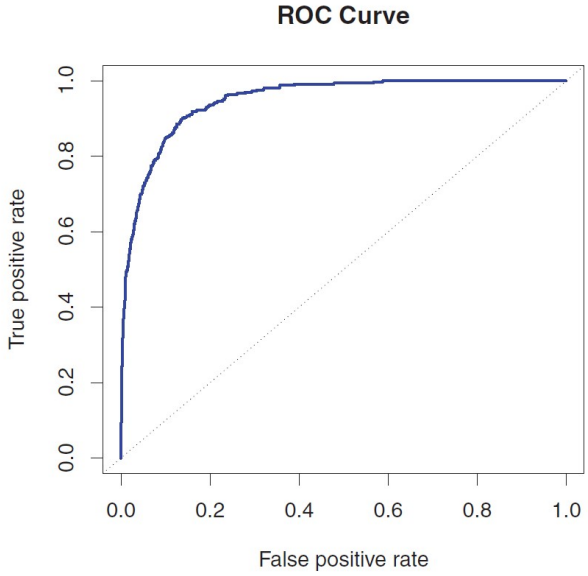


Figure 2.4: An example of a ROC curve, taken from [Jam+13b].

Some argue that metrics like precision, recall and F1-score are not representative of the overall performance of the algorithms, since they only focus on the positive samples [Faw06]. Furthermore, they suffer from prevalence, bias, skew and cost ratio which might affect their usefulness and representativeness [Pow11]. Thus, the choice of a performance metric is always dependent on the problem and the characteristics of the available data.

2.2 Fairness Concepts

The usage of Machine Learning models in real-world scenarios, to support decisions which have a significant impact on people's lives, raises several legal and societal concerns, some of which related to discrimination and fairness [Bin18]. Although this property is difficult to define due to its ambiguity and subjectivity, throughout this work we consider it to be the absence of bias or discrimination against people based on **protected** or **sensitive attributes**. These attributes are usually intrinsic characteristics of an individual, such as race, gender, or age, and discrimination based on them is often prohibited by law, as in the case of hiring and housing processes in the USA [Fel+15; Dau12]. Being a property of a system, it must be evaluated, and for that reason, fairness only materialises itself through certain conditions and metrics.

Unfairness in intelligent systems may appear in many different forms and may have a variety of root causes. The available training data may itself be unfairly sampled or labelled [Kam+12], if certain groups are under-represented in the data or the labels result from biased decisions. This might be the case if a bank has been unfairly rejecting loans of people who belong to a certain minority group [Kam+12]. However, even if the models are trained on historical data that contains bias against certain social-demographic groups, this bias should not be perpetuated by the algorithmic decisions. We are particularly interested in the potential unfairness of the outputs given by ML models in supervised classification scenarios, and its relation to the data used to train them, which in turn can be biased, as described above.

Disparate treatment, a direct form of discrimination, results from a deliberated use of the sensitive attribute during the decision-making process. This type of discrimination can be avoided by removing this attribute from the data, prior to training a model [Xu+].

The classifications of models trained without the sensitive attribute may still be discriminatory, leading to an unfair treatment of protected groups [CV10; Xu+]. This *red-lining effect* is due to the presence of features highly associated with the sensitive attribute, which can be used to identify the protected group [KC09; CV10; Xu+]. This effect is linked to **disparate impact**, an indirect form of discrimination that may results from an unintentional usage of sensitive attributes. As long as objective and reasonable justifications for it can be given by the defendant [RR14; Fel+15], indirect discrimination will not be considered illegal. The example given in [RR14] illustrates this scenario: for a transport company hiring truck drivers, it is legitimate to select applicants based on whether they possess a truck-driving licence, even if women are discriminated against for being less likely to have a truck-driving licence.

It is important to clarify that these notions of *direct* and *indirect discrimination* are contemplated in the legislation of many countries. Taking Portugal as an example, we can find references to these concepts in *Código do Trabalho* [Rep]. It is clearly stated in this document that indirect discrimination occurs when an apparently neutral disposition, criterion or practice puts an individual in a disadvantageous position in comparison to other individuals, by taking into account attributes correlated with a sensitive attribute [RR14]. In such cases, an objective justification for the effect of that disposition, criterion or practice must be given by the defendant. Similar to the US legislation, such a justification is accepted if "objectively justified by a legitimate aim and the means of achieving that aim are appropriate and necessary" [RR14].

Disparate mistreatment has also been proposed as a notion of unfairness which addresses differences in the misclassification rates, such as the FPRs, between the protected

and the unprotected groups [Zaf+17a]. The authors claim that this notion is particularly relevant when one has access to the ground-truth decisions [Zaf+17a].

All these notions seem to treat an individual as being part of a social-demographic group. However, in [Dwo+12] the notion of *individual fairness* was introduced, with the goal being to ensure that similar individuals are given similar treatment. Several challenges arise with this notion, namely how to measure the similarity between individuals.

For a more in-depth discussion of fairness and its relation to other fields, we recommend reading the survey by Romei and Ruggieri [RR14], as well as the paper by Binns [Bin18].

As far as the landscape of research on fairness is concerned, we can categorise related work based on its aim. On the one hand, there are those who are trying to find better conditions and metrics to measure fairness. On the other hand, we have those who try to improve the fairness of software systems by proposing ways in which the training data can be altered, so as to become discrimination-free, or by proposing modifications to learning algorithms to make them discrimination-aware. More recently, an effort is also being made to benchmark the different approaches that aim at improving fairness of software systems. There is also a body of work aiming at discovering unfairness in databases, sometimes referred to as discrimination discovery data mining [HBC16]. In the sections that follow, we try to overview the most relevant approaches in the scope of our work.

2.2.1 Fairness Assessment

In a supervised classification problem, we are given a labelled dataset $\mathcal{D} = \{X, S, Y\}$ of n instances (also called samples or individuals): X are the non-sensitive attributes, S denotes a sensitive attribute, and Y represents the true labels. The variable predicted by a classifier, i.e. the predicted classifications, is referred to as $\hat{Y}$. In a binary problem, the positive outcome or class is represented by 1 and the negative outcome is given by 0.

A binary sensitive attribute partitions the dataset into two disjoint subsets: the subset composed by the instances for which the value of the sensitive attribute is 0 is called the **protected** or **unprivileged group**, while the subset of instances for which the value of the sensitive attribute is 1 is called the unprotected or privileged group.

We divide the metrics used to assess fairness into two main categories: those which consider fairness at a group level and those which also focus on fairness at an individual level. The fairness metrics which measure fairness at a group level can be further decomposed into the ones that can only be applied when the classifications are known, and those which can be applied at the data-level or when the classifications made by a model are known. All the metrics presented here are suitable for classification problems, although there is also a body of work that focus on assessing fairness in rankings [YS17].

We start by presenting the metrics which can be applied to both measure the fairness in a labelled dataset and in the outputs given by a classifier. The following definitions can be applied to data if we use Y instead of $\hat{Y}$.

- **Statistical or demographic parity** requires that the output be independent of the sensitive attribute [ZLM18], meaning that the proportion of individuals from a certain group receiving any classification is the same as the proportion of individuals receiving that classification in the overall population.

Considering a binary classifier, statistical parity translates into the rate of favourable classifications being the same across all values of the sensitive attribute [ZLM18]:

$$P(\hat{Y} = 1) = P(\hat{Y} = 1|S = s) \tag{2.11}$$

For binary classification problems with a single binary sensitive attribute, a variation of statistical parity sometimes called **Statistical Parity Difference (SPD)** considers the difference of the rate of favourable classifications between the protected and the unprotected groups:

$$P(\hat{Y} = 1|S = 1) - P(\hat{Y} = 1|S = 0) \tag{2.12}$$

In [Kam+12], this variation is referred to as the Calders-Verwer Score (CVS). Measurements of this metric lie in $[-1, 1]$, with 0 being optimal fairness. The sign of a measurement indicates a skew in favour of either the protected or the unprotected group [Fri+19].

As pointed out by [Dwo+12] and [HPS16], this metric actually has some flaws and does not fully ensure fairness. For instance, undeserving individuals from the unprivileged group might receive favourable classifications as long as the proportions of individuals receiving that classification match across groups [HPS16].

- **Disparate Impact (DI)** is given by the ratio of the rate of favourable classifications for the protected group to that of the unprotected group [Bel+]:

$$\frac{P(\hat{Y} = 1|S = 0)}{P(\hat{Y} = 1|S = 1)} \tag{2.13}$$

This is often referred to as the p%-rule and for a classifier to be considered fair it should be greater than 80% but lower than 125%, meaning that it does not have disparate impact [Fel+15; Zaf+17b]. The 80% rule is advocated by the US Equal Employment Opportunity Commission [Fel+15], and can be found in the Uniform Guidelines on Employee Selection Procedures [Uni]. Measurements of this metric lie in $[0, \infty[$, with 1 being the optimal value. A measurement different from 1 indicates a skew in favour of one of the groups.

The authors of [Fel+15] link this metric to the Balanced Error Rate (BER) and define the notion of ϵ-fairness, where a dataset is considered to be ϵ-fair if the BER of any classifier trained on the dataset is bigger than ϵ. For further details on this notion, refer to [Fel+15].

- The **Prejudice Index (PI)**, as defined by [Kam+12], corresponds to the mutual information between the classifications given by a model and the sensitive attribute. The **Normalised Prejudice Index (NPI)** results from the application of a normalisation technique for mutual information and is given by:

$$NPI = \frac{PI}{\sqrt{H(\hat{Y})H(S)}} \tag{2.14}$$

where $H(\cdot)$ is an information entropy function [Kam+12]. The NPI can be regarded as the geometrical mean of the ratio of information of the sensitive attribute used for making the classifications, and the ratio of exposed information if a value of Y is known [Kam+12]. The NPI ranges between $[0, 1]$, with 0 being the optimal value.

In an attempt to overcome some of the problems that these simpler metrics raise, a new set of fairness metrics, based on the classification rates across the protected and the unprotected groups, have been proposed. These metrics can only be applied when the classifications made by the model under evaluation are known. In [Fri+19], the authors refer to these as *group-conditioned accuracy measures*. To get their definition, we take the metrics used to measure classification performance presented in Section 2.1.3 conditioned on the sensitive attribute. The definitions that follow apply to a binary classification problem with a single binary sensitive attribute, and so, $y \in \{0, 1\}$ and $s \in \{0, 1\}$. All the definitions can be found in [Fri+19].

- **S-ACC** uses accuracy as the underlying metric and, in a classification problem, is given by:
$$S\text{-}ACC = P(\hat{Y} = y|Y = y, S = s) \tag{2.15}$$
- **S-BCR** takes BCR as the underlying metric, being given by:
$$S\text{-}BCR = \frac{P(\hat{Y} = 1|Y = 1, S = s) + P(\hat{Y} = 0|Y = 0, S = s)}{2} \tag{2.16}$$
- **S-TPR** can be used when the focus is on the positive class:
$$S\text{-}TPR = P(\hat{Y} = 1|Y = 1, S = s) \tag{2.17}$$
This metric often appears in the form of a condition, under the name of **equal opportunity**, which can be regarded as a relaxation of equalised odds which requires fairness only within the group of positive samples. For a binary classifier to satisfy equal opportunity, the TPRs must be the same for all values of the sensitive attribute [HPS16]:
$$P(\hat{Y} = 1|S = 0, Y = 1) = P(\hat{Y} = 1|S = 1, Y = 1) \tag{2.18}$$
- **S-TNR** is a more suitable choice of fairness metric when the focus is on the negative class. It is given by:
$$S\text{-}TNR = P(\hat{Y} = 0|Y = 0, S = s) \tag{2.19}$$

There are various possibilities when it comes to aggregating the measurements for each group, for instance, by simply taking the mean. Measurements of such variant would lie in $[0, 1]$. One can also aggregate the measurements in a similar way to SPD, by taking the difference between the two groups, and the final value would lie in $[-1, 1]$. Similar to DI, we can also take the ratio between the protected and the unprotected groups, with the range of this metric being $[0, \infty[$ and fairer results being closer to 1.

Similar to the performance metrics in which these metrics are based, fairness measurements as given by group-conditioned accuracy metrics may also be misleading when there is an imbalance between the protected and the unprotected groups. In fact, most of the fairness metrics found in the literature seem to disregard this possible imbalance in the data, as pointed out by [Spe+18].

As with the case of equal opportunity, other group-conditioned accuracy metrics often appear in the form of a condition that must be satisfied for a classifier to be considered fair. For **equalised odds** to be satisfied, the classifications ($\hat{Y}$) and the sensitive attribute must be independent conditional on the true labels (Y). This fairness condition enforces

an equally high accuracy across all values of S, with models which only achieve high classification performance for the privileged group ending-up being penalised [HPS16]. This condition is not restricted to classification scenarios.

Considering the case where all three variables are binary, equalised odds requires that the TPRs and the FPRs are the same for both the protected and unprotected groups [HPS16]:

$$P(\hat{Y} = 1|Y = y, S = 0) = P(\hat{Y} = 1|Y = y, S = 1) \tag{2.20}$$

In other words, equalised odds corresponds to $S\text{-}TPR$ and $1 - S\text{-}TNR$ being equal for all values of S.

All the fairness metrics presented so far only take into consideration fairness at a group-level, i.e. unfairness arises when there are differences between the protected group and the unprotected group, with each of these groups being considered as a whole in the evaluation. The fairness metrics that follow aim at assessing fairness at an individual-level, i.e. unfairness arises when similar individuals are not treated equally.

- **Consistency**, as defined by [Zem+13], is used as an individual fairness metric which measures the similarity between classifications for similar samples:

$$1 - \frac{1}{n \times k} \sum_{i=1}^{n} |\hat{y}_i - \sum_{j \in kNN(\mathbf{x}_i)} \hat{y}_j| \tag{2.21}$$

  where $\mathbf{x}_i$ is the feature vector of individual i and the similarity between samples is given by a k-nearest neighbours function, $kNN(\mathbf{x})$.

- The **Generalised Entropy Index (GE)** is used by the authors of [Spe+18] to measure the *overall individual-level unfairness* of a classifier and is given by:

$$GE(\alpha) = \frac{1}{n\alpha(\alpha - 1)} \sum_{i=1}^{n} \left[\left(\frac{b_i}{\mu} \right)^{\alpha} - 1 \right], \alpha \notin \{0, 1\} \tag{2.22}$$

  where μ is the mean benefit and $b_i = \hat{y}_i - y_i + 1$ corresponds to the benefit of individual i according to the benefit function proposed by the authors. The authors' proposal is to use *inequality indices*, widely studied in economics and other fields, to measure (un)fairness [Spe+18]. They claim that these inequality indices satisfy many axioms also deemed relevant when it comes to the definition of a fairness metric. Such axioms include, among others, anonymity, population invariance, and subgroup decomposability. When $\alpha = 2$, the value used throughout their work, we get half the squared coefficient of variation.

  The benefit function as defined by [Spe+18] depicts individual fairness as the difference between the preference for the outcome an individual truly deserves and the preference for the outcome received from the learning algorithm [Spe+18]. There is a close relationship between this notion of fairness and classification accuracy, but the fairness optimal and the accuracy optimal classifiers will only be the same under certain strict conditions [Spe+18]. Optimal fairness is obtained when the received benefit is the same for every individual, in which case GE is zero. The range of possible values lies between $[0, \infty[$.

  The overall individual-level unfairness considers fairness at both an individual and at a group-level and can be further decomposed into a between-group component, similar to other group-level metrics, and a within-group component [Spe+18].

2.2.2 Fairness Improvement

The approaches that have been proposed to improve the fairness of ML models or mitigate bias in their predictions can be grouped into three categories: pre-process, in-process and post-process approaches. The **pre-process approaches** modify the training data to make it free of discrimination; the **in-process approaches** change the models by adding constraints and regularisation terms to the objective functions; and the **post-process approaches** directly change the predictions made by the models [Xu+].

Pre-process approaches align with our work in that they explore ways to manipulate the data before it is used to train the models. The Uniform Sampling and the Preferential Sampling methods proposed in [KC12] involve undersampling and oversampling instances from the four groups that result from the combination of a binary sensitive attribute and binary labels. A distinction between these two methods is made based on the criteria used to select the instances that are duplicated or discarded. In addition to these sampling methods, the authors also propose suppression (removal of the sensitive attribute and those highly correlated with it), massaging (modification of the labelling of the training examples) and reweighing (assignment of weights to the training instances).

The methods proposed by [Fel+15] aim at removing disparate impact from a dataset and fall in the category of pre-process approaches. These methods modify the distributions of the non-sensitive features so that the sensitive attribute cannot be predicted from them [Xu+]. The two approaches proposed by the authors allow for different *amounts of repair* through the introduction of a parameter which controls the trade-off between the ability to make accurate classifications and the fairness of the modified dataset [Fel+15].

The authors of [Zem+13] aim at learning a representation which satisfies statistical parity, while still encoding the data well and allowing for accurate classifications. This representation can be regarded as a set of prototypes and the model basically maps each individual, represented by a point in the input space, to a probability distribution in the space defined by the new representation [Zem+13]. This proposal addresses fairness at both a group and an individual level.

FairGAN, the method proposed in [Xu+], takes inspiration from Generative Adversarial Networks (GANs) [Goo+14] to generate fair data. This method allows for the generation of more data than other pre-process approaches and can be applied to both numerical and categorical features. The results show that the classifiers trained on the synthetic datasets generated by FairGAN can satisfy statistical parity in terms of their predictions, while still achieving high accuracy.

In [CV10], the authors propose three methods to modify Naive Bayes classifiers so as to remove discrimination from their classifications. They take statistical parity as their definition of fairness and aim at building classifiers where the target output is independent from the sensitive attribute. Both the response variable and the sensitive attribute are assumed to be binary.

The first method corresponds to a post-processing step where the probability of a positive decision is changed by modifying the probabilities in the Bayesian model [CV10]. This is accomplished by modifying the joint distribution of the sensitive attribute and the target variable so that the statistical parity difference approaches zero [Kam+12].

The second method, called the two Naive Bayes method, corresponds to training a classifier for each value of the sensitive attribute and then balancing the trained models [CV10]. The joint distribution of the sensitive attribute and the target variable is modified as in the

first method. The graphical model for this method is shown in Figure 2.5. From their experiments, this method seems to lead to the best results.

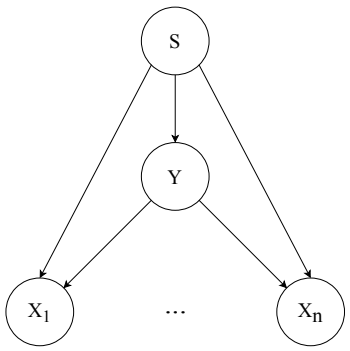


Figure 2.5: Graphical model for the two Naive Bayes method, adapted from [CV10].

The third method considers the addition of a latent (or hidden) variable to the Bayesian model and uses expectation maximisation to optimise the parameters for the likelihood function [CV10]. This latent variable aims at representing an unbiased target output. For more details regarding any of these three methods, refer to [CV10].

The proposal of Kamishima et al. [Kam+12], which fits in the in-process category, is to add a regulariser to a logistic regression model so as to reduce the indirect discrimination during the learning process. The prejudice remover regulariser, $R_{PR}(\mathcal{D}, \mathbf{\Theta})$, is based on the prejudice index and is given by:

$$R_{PR}(\mathcal{D}, \mathbf{\Theta}) = \sum_{(\mathbf{x_i}, s_i) \in \mathcal{D}} \sum_{y \in \{0,1\}} p(y|\mathbf{x}_i, s_i, \mathbf{\Theta}) \ln \frac{\hat{p}(y|s_i)}{\hat{p}(y)} \tag{2.23}$$

where $\mathbf{\Theta} = \{\mathbf{w}_s\}_{s \in S}$ are the logistic model parameters. The authors proposed specific approximations to $\hat{p}(y|s_i)$ and $\hat{p}(y)$. This regulariser takes large values when the predicted class is mainly determined by the sensitive feature.

Thus, the cost function, which also considers L2 regularisation, becomes:

$$-\sum_{i=1}^{n} \ln p(y_i|\mathbf{x}_i, s_i, \mathbf{\Theta}) + \eta R_{PR}(\mathcal{D}, \mathbf{\Theta}) + \frac{\lambda}{2} \sum_{s \in S} \|\mathbf{w}_s\|_2^2 \tag{2.24}$$

where η and λ are positive regularisation parameters. For further details on the learning process followed by the authors, refer to [Kam+12]. The authors compare models trained with and without the sensitive attribute and consider different data preparation procedures. However, no evaluation is performed with regards to using different versions of the same dataset to train the same learning algorithms.

The authors of [Zaf+17b] propose a measure of decision boundary (un)fairness based on the covariance between the individuals' sensitive attributes and the signed distance of their feature vectors to the decision boundary [Zaf+17b]. Taking this measure as a basis, they derive two formulations of constrained optimisation problems: one which maximizes accuracy subject to fairness constraints (ensuring compliance with a given p%-rule, if demanded by law) and another which maximises fairness subject to accuracy constraints (ensuring that certain business needs are met, by allowing a relaxation on fairness) [Zaf+17b]. They show concrete instantiations of these problems using logistic regression classifiers and SVMs.

In a more recent work, an adversary is added to an ANN with the goal of penalising models for which it is possible to predict the sensitive attribute from the outcomes [WVP].

The authors take more than one fairness notion into consideration when designing their adversary. In [ZLM18], the authors also use adversarial learning to ensure a set of fairness conditions. In this case, they add an adversary to a logistic regression model and also consider the removal of bias from word embeddings.

The approaches proposed by [KCP10] and [RSM18] focus on modifying tree-based methods so as to make them discrimination-aware and improve the fairness of the models' predictions. This goal is accomplished by changing the evaluation of the splitting criterion or by relabelling the leafs. The relabelling approach fits in the post-process category, since it can only be applied when a tree has already been learned.

The proposals by [FKL15] and [HPS16] also fit in the post-process category. In particular, the authors of [HPS16] propose a post-processing step to achieve equalised odds and equal opportunity by picking points in the ROC curves conditioned on the sensitive attribute.

More recently, [Fri+19] focused on defining a benchmark to evaluate fairness. A variety of fairness-enhancing methods are compared, and the relationship between different fairness metrics is investigated. This work also alerts to the need to carefully specify the data pre-processing techniques applied to the training data, as they may have a significant impact on the fairness evaluation of a system. IBM has also invested in an open source toolkit [Bel+] that comprises a large set of fairness metrics, as well as implementations of approaches proposed to mitigate bias in ML models from the three aforementioned categories. In the interactive demo of the toolkit, we can see that they are putting some effort on comparing fairness at different stages of the pipeline, more precisely before and after one of the supported approaches to mitigate bias is applied.

2.3 Conclusion

Several fairness metrics have been proposed in the literature that try to take different notions of fairness into account. However, not all can be applied to both a labelled dataset and afterwards when the classifications made by a model are known. Measuring fairness at different stages of the ML pipeline seems crucial to us, in order to be able to pinpoint the prevailing factors when changes in fairness are detected. Furthermore, a more in-depth analysis of the relationship between the data and the models' structure is needed, for instance, in terms of the weight each feature has on classifications made by the models.

As pointed out in [Fri+19], the procedures applied to the data before being used to train a Machine Learning model are often neglected. Nevertheless, it is known that they may have an impact on the fairness of the system. Moreover, there have been several proposals to enhance fairness, either by changing the training data, by modifying the learning algorithm, or by making alterations to its outputs, but little attention has been paid to assessing the impact that procedures which are commonly adopted by the community might have on fairness.

The fact that the datasets used in this context often present imbalance between privileged and unprivileged groups also demands further analysis, namely when it comes to the confidence in the estimates provided by the fairness metrics. Moreover, it is known that models trained with data in which one class is over-represented are likely to be biased towards that class. When analysing the trade-off between fairness and performance, accuracy is usually picked to evaluate performance, even when the original dataset is not balanced with respect to the true labels.

Chapter 3

Research Objectives and Approach

The main goal of this work is to assess the fairness of systems which use Machine Learning (ML) models in classification scenarios. To fulfil this goal, we aimed at understanding **how the different procedures applied to a dataset during data preparation and pre-processing impact the fairness of the training data, as well as the fairness of the outputs produced by those models**. Although several pre-processing approaches have been proposed to modify the data so as to make it free of discrimination before being used to train the models, little attention has been given to assessing the impact that more traditional procedures may have on the fairness of such models. Additionally, it is also important to understand how fairness changes throughout the pipeline, both at the data-level and when the final outcomes of the models are known. To achieve the main goal of this work, we formulated the following research questions:

RQ1. How does feature discretisation and the encoding of the categorical attributes impact the fairness of the classifications of an ML model?

RQ2. What is the impact of different instance selection techniques on the fairness of the classifications made by an ML model? We consider sampling methods as instance selection techniques.

RQ3. How does the removal of the sensitive attribute impact the fairness of the classifications made by an ML model?

RQ4. What is the relation between the unfairness in the data used to train ML models and the one in the classifications made by those models?

A cross-cutting concern of this work is the evaluation of the **trade-off between the fairness of the models' outputs and the overall performance of the system** in the main classification tasks. Intuitively, improving fairness in a system might lead to a degradation of its performance, but we plan to analyse this hypothesis so as to provide the system's developers and owners with the necessary knowledge about these trade-offs, allowing them to make informed decisions. Furthermore, we investigated the impact of the learning algorithm on the fairness of a system, with an initial focus on tree-based methods.

Figure 3.1 shows the approach that was followed to make this fairness evaluation. As depicted in this figure, we only used the training set of the datasets in our experiments. Our goal was to compare different configurations against one another from a fairness point-of-view and not to fine tune the parameters to maximise performance. For this reason, the test set of each dataset was not used in our study. The data represented by **(1c)** results from

the application of all the data preparation and pre-processing procedures whose impact on fairness we are evaluating. The steps represented by **(2a)**, **(2b)**, and **(2c)** are the main focus of our work, with dashed boxes, i.e. **(2b)** and **(2c)**, representing optional steps: for instance, under some of the tested configurations, the sensitive attribute is not removed prior to training. These steps are detailed in the remainder of this chapter.

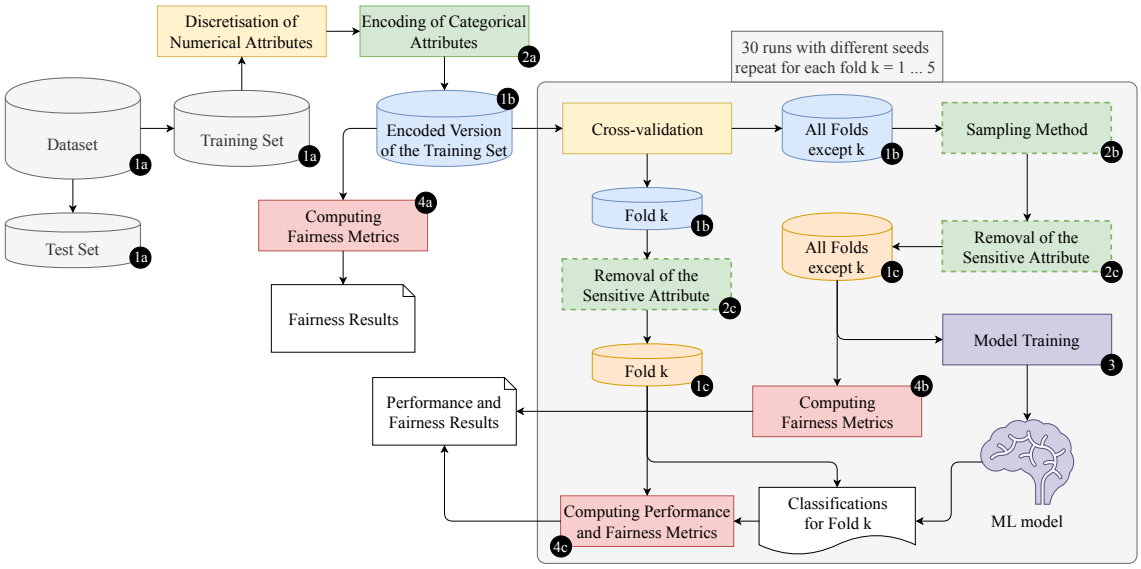


Figure 3.1: Overview of the proposed approach for fairness assessment.

Prior to this evaluation, we had to collect and understand the fairness concerns of a set of representative datasets, with different sizes and distributions of the sensitive attributes. An exploratory data analysis of the selected datasets, which were widely used in previous works on fairness, allowed us to get a better understanding of their characteristics, namely of the features that may be unfairly exploited and their relation to the classification labels.

We had to analyse and understand the underlying rationale for the definition of the various fairness conditions and metrics used in the literature, their limitations, and the scenarios in which they can be applied to. To better comprehend the impact of the data preparation and pre-processing techniques, we measure fairness at both the data-level, and regarding the classifications made by the trained models, as depicted by steps **(4a)**, **(4b)**, and **(4c)** in Figure 3.1. An analysis between fairness measurements from **(4a)** and **(4b)** is performed in Section 5.1.

3.1 Datasets

Three datasets are commonly used in the literature on fairness: the Adult Income dataset (also known as Census Income) [DG19], the German Credit Data dataset [DG19], and the COMPAS dataset [Ang+16]. This section provides an explanation of these datasets in terms of the main classification task and the fairness concerns that they pose. We perform a more in-depth analysis of each dataset in Chapter 4.

The **Adult** or **Census Income dataset** is publicly available from the UCI Machine Learning Repository [DG19]. It contains demographic data extracted from the 1994 US Census Bureau database, with each instance being described by 14 categorical and numerical attributes. Some of these attributes include age, sex, marital status, race, level of education,

occupation, and working hours per week.

There are 48,842 instances in the dataset and a split into training set (32,561 instances) and test set (16,281 instances) is provided with the original data. The main task is to predict whether a person earns over 50,000 dollars per year, therefore making a classification into `high income` (the favourable class) or `low income`. The models' classifications may be used to make decisions regarding the assignment of loans or to decide the salary of a new-hire [KC09], and might lead to legal actions against institutions if deemed unfair [CV10].

In our experiments, we followed previous work and used `sex` as the sensitive attribute with `female` being the unprivileged group. According to the dataset, women are more likely to be classified into the `low income` class than men. Furthermore, this decision has been historically biased in favour of men, with women tending to have lower salaries [KC09]. Race as also been used in previous work as a sensitive attribute [Zaf+17b; Spe+18].

The **German Credit Data dataset** is publicly available from the UCI Machine Learning Repository [DG19]. It contains financial information about 1,000 individuals, described by a set of 20 categorical and numerical attributes. Some of these attributes include age, marital status, existing checking account status, duration, credit amount, employment status, and type of housing (own, rented, or free). The attribute `sex` can be derived from `personal-status-sex` of the original dataset.

A pre-split into training and test data is not provided for German Credit Data. Thus, we performed a 70/30 stratified split and tried to maintain the distributions of the true labels and the sensitive attribute on each set. The objective is to classify each person into `good credit risk` (the favourable class) or `bad credit risk`. Credit risk assessment is a standard procedure in banks so as to protect them from risks associated with defaulting.

Most studies consider `age` as the sensitive attribute, although sex and being a foreign worker have also been considered as such in some previous work [Cel+; Fri+19; PRT08; YS17]. As reported by [KC09], young people are less likely to be classified into the good credit class compared to aged people, which raises fairness concerns. Thus, we considered `age` to be the sensitive attribute with `young` as the unprivileged group.

The **COMPAS dataset** was compiled by ProPublica [Ang+16] and contains records from all criminal defendants who were screened with the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) tool in Broward County, Florida, between 2013 and 2014. Besides the risk scores given by the tool, the data includes demographic information (such as sex, age and race) and attributes related to the criminal history of each individual (such as the type of offence each person was arrested for and the number of prior offences) [Zaf+17a].

The main task is to predict whether a criminal defendant will re-offend within two years or, in other words, to classify each defendant according to his/her risk of recidivism. The score given by the tool ranges from 1 to 10 and is decomposed into three classes: low, medium and high risk. From the criminal records, ProPublica tried to retrieve the ground-truth about each defendant re-offending, or not, within two years after being given the score by the tool [Lar+16]. Further details about how the data was collected and analysed by ProPublica can be found in [Lar+16].

We only performed experiments with respect to the *risk of recidivism*, therefore not taking into consideration data related to the *risk of violent recidivism* or to the *risk of failure to appear*. The dataset of interest can be found in a GitHub repository [Pro] as a CSV file named `compas-scores-two-years.csv`. We filtered the instances as described in the Notebook found in that same repository, which left us with a total of 6,172 entries. A

pre-split of the data is not given, and so, we performed a 70/30 stratified split with respect to both the true labels and the sensitive attribute.

We are interested in the ground-truth information about each defendant, and so, the features directly related to the application of the tool (e.g. date of the assessment) are not relevant to our work. We excluded attributes like `name` and `case_number`, as well as those directly associated with the application of the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) tool, since our goal is to build ML models that classify individuals according to their risk of recidivism, just like COMPAS. Therefore, we only keep the following columns of the original dataset: `id` (not a feature, just used to identify each instance), `sex`, `age`, `age_cat`, `race`, `juv_fel_count`, `juv_misd_count`, `juv_other_count`, `priors_count`, `c_charge_degree`, `c_charge_desc`, and `two_year_recid` (true labels).

Risk assessment algorithms like COMPAS are widely used in courtrooms across the USA to aid in deciding who can be set free at every stage of the legal process [Ang+16]. A mistake made by the system might lead to the release of a dangerous criminal or, on the other hand, to a more strict sentence than the one truly deserved by the defendant being applied [Ang+16].

Race is usually used as the sensitive attribute, although the analysis made by ProPublica also revealed unfairness with respect to gender. They found that the tool mistakenly labelled African-American individuals with high risk at an higher rate than Caucasian individuals [Lar+16]. Additionally, they found Caucasian defendants were mislabelled with a lower risk of recidivism at an higher rate than African-American defendants [Lar+16]. This means African-American defendants might be being unfairly punished by the system. Regarding gender bias, ProPublica reports that a woman who receives a high risk score has a much lower risk of re-offending than a man who receives the same score [Lar+16].

3.2 Mixed Features and Missing Values

So as to evaluate the impact that different encodings of the categorical features have on the fairness of the training data and the classifications, we followed the approaches of [CV10] and [Kam+12], and discretised the numerical attributes into 4 bins with the boundaries corresponding to those of the interquartile ranges. An additional transformation was performed for the Adult Income and the COMPAS datasets, with bins having low frequency counts (less than 50 instances) being pooled together and the new category being represented by the same `Pool` value. This additional transformation was only applied to originally categorical attributes. After the application of these procedures, the data only contains categorical features. For each feature of the transformed data, each possible category is then represented by an integer value, and we refer to this as the integer encoded version of a dataset.

We create another version of each dataset with all features using a one-hot (or 1-of-K) encoding scheme [Bis06] after being discretised, meaning that they are represented by binary dummy variables. We refer to this as the one-hot encoded version of a dataset.

Two exceptions occurred with German Credit Data. The `personal-status-sex` was removed from both versions of the dataset after deriving `sex`. The `age` attribute was discretised into two bins defined by a value greater than or equal to 25, a threshold set based on the findings reported by [KC09]. For the COMPAS dataset, the only exception relates to the `race` attribute, which is discretised into two bins: one with the instances that belong to the `caucasian` category, and another with the remaining instances.

Contrary to the German Credit Data dataset, Adult Income and COMPAS contain missing values. For the integer encoded version of these two datasets, all instances containing at least one missing value were dropped prior to training or testing the models. Regarding the one-hot encoded version of the datasets, all instances were kept regardless of the presence of missing values. In such cases, a missing value was represented by having all the corresponding dummy variables set to zero.

3.3 Imbalanced Data and Sampling Methods

The datasets have an imbalance with respect to not only the true labels, but also the sensitive attribute. In such scenarios, it is common to apply sampling methods, namely Random Undersampling (RU) and Random Oversampling (RO), so as to train the models with an equal number of instances from each class. Besides the typical scenario in which RU or RO is performed with respect to the true labels (`undersampling-label` or `oversampling-label`, respectively), we considered four additional configurations:

- RU applied with respect to the sensitive attribute (`undersampling-protected`);
- RO applied with respect to that same attribute (`oversampling-protected`);
- RU applied with respect to a variable which combines the true labels and the sensitive attribute (`undersampling-multivariate`);
- RO applied with respect to the multivariate variable resulting from the combination of the true labels and the sensitive attribute (`oversampling-multivariate`).

We compared these sampling strategies to a baseline scenario in which no sampling method is applied (`without-resampling`).

For each of the settings with RU, we determine which group has fewer instances and keep them, while randomly removing instances from the remaining groups, until their number equals that minimum. For the settings in which RO is applied, we determine which group has the most samples and keep them, while randomly sampling with replacement the instances from the remaining groups, until their number equals that maximum. By performing RU or RO in such a way, we ensure that we have a ratio of 1:1 between instances from each group. After applying `undersampling-multivariate` or `oversampling-multivariate` the training data can be considered perfectly fair under SPD, DI, and NPI.

We used the `imbalanced-learn` package [LNA17] to perform random undersampling and random oversampling.

3.4 Learning Algorithms and Removal of Sensitive Attributes

We performed our experiments with tree-based methods, more precisely Decision Trees and Random Forests. Bearing in mind that we were dealing with categorical attributes, we looked for implementations of these methods which offered support for such attributes. We opted for the implementations provided by the Python API of Apache Spark™.

We set the maximum depth of the trees to 30 (maximum supported by these implementations) and used the Gini index as the impurity criterion. Additionally, for the Random Forests, we considered ensembles of 10 trees and the squared root of the total number of

features when looking for the best split. Our goal was not to fine tune the parameters, but to understand how the different combinations of data preparation / pre-processing techniques and classifiers impacted the system from a fairness point-of-view. Therefore, we used the default values for most of the remaining parameters.

Since we wanted to analyse the influence of the removal of the sensitive attribute prior to training a model, we devised four possible scenarios: Decision Tree with and without the sensitive attribute (DT and DTns, respectively), and Random Forest with and without this attribute (RF and RFns, respectively).

3.5 Model Assessment

We performed five-fold cross-validation with the help of the methods provided by `Scikit-learn` [Ped+11]. Furthermore, each configuration was run with 30 different seeds for the random generators.

From the initial set of sixteen fairness metrics, we selected a final group of six, with at least one metric from each of the three categories presented in Section 2.2.1. This final set of metrics includes: Statistical Parity Difference (SPD), Disparate Impact (DI), the Normalised Prejudice Index (NPI), the Generalised Entropy Index (GE), the ratio variation of group-conditioned TPR (S-TPR-ratio), and the same variation of group-conditioned TNR (S-TNR-ratio).

We evaluated the classification performance of the models with the F1-score and Balanced Classification Rate (BCR), more suitable metrics when dealing with imbalanced datasets, since we wanted to consider in our analysis both the classifications given to instances from the positive class, as well as those given to instances from the negative class.

3.6 Conclusion

Table 3.1 presents a summary of the tested configurations. Taking all the possible combination of parameters into account, we get a total of 168 different configurations, each repeated 30 times with different seeds for the random generator and always performing five-fold cross-validation. For each configuration, we get a total of 150 repetitions.

Moreover, we initially considered three performance metrics and sixteen fairness metrics (some of which being the variations of the group-conditioned metrics) in our analysis, which gives us a total of 2,850 measurements for each of the 150 repetitions of each considered configuration. It would become unfeasible to present all these results, and so, we only report the results for two of these performance metrics and six of these fairness metrics, as explained in Chapter 5.

In the following chapter, we report the results of our exploratory data analysis of the three datasets, with a special emphasis on the distributions of true labels and sensitive attribute, as well as on the associations between features and true labels.

Parameter	**Tested Settings**
Dataset	Adult Income German Credit Data COMPAS
Encoding of Categorical Features	Integer encoding One-hot encoding
Sampling Method	undersampling-label undersampling-protected undersampling-multivariate oversampling-label oversampling-protected oversampling-multivariate without-resampling
Learning Algorithm and Removal of Sensitive Attributes	Decisions Tree with the sensitive attribute (DT) Decisions Tree without the sensitive attribute (DTns) Random Forest with the sensitive attribute (RF) Random Forest without the sensitive attribute (RFns)

Table 3.1: Summary of the experimental setup.

This page is intentionally left blank.

Chapter 4

Exploratory Data Analysis

In this chapter, we present the summary statistics about the features of each of the three datasets used in the experiments and give an overview of the distribution of the training data in terms of the sensitive attribute and the classification classes. By doing so, we show the imbalances found in the training set of each dataset. Additionally, we show the relation between features and classification classes through the computation of the Normalised Mutual Information (NMI) and the Cramer's V. By understanding these relations, we become aware of interactions that may lead to indirect discrimination when training Machine Learning (ML) models.

4.1 Adult Income dataset

In Table 4.1, we show the number of different categories for each feature of the integer encoded version of the Adult Income dataset. The one-hot encoded version of this dataset has 95 features, since features with only two categories in the integer encoded version can be represented by a single binary variable.

Feature Name	Number of Categories	Feature Name	Number of Categories
age	4	relationship	6
workclass	7	race	5
fnlwgt	4	capital-gain	2
education	16	capital-loss	2
education-num	4	hours-per-week	3
marital-status	7	native-country	22
occupation	14	sex	2

Table 4.1: Number of categories per feature - integer encoded version of Adult Income.

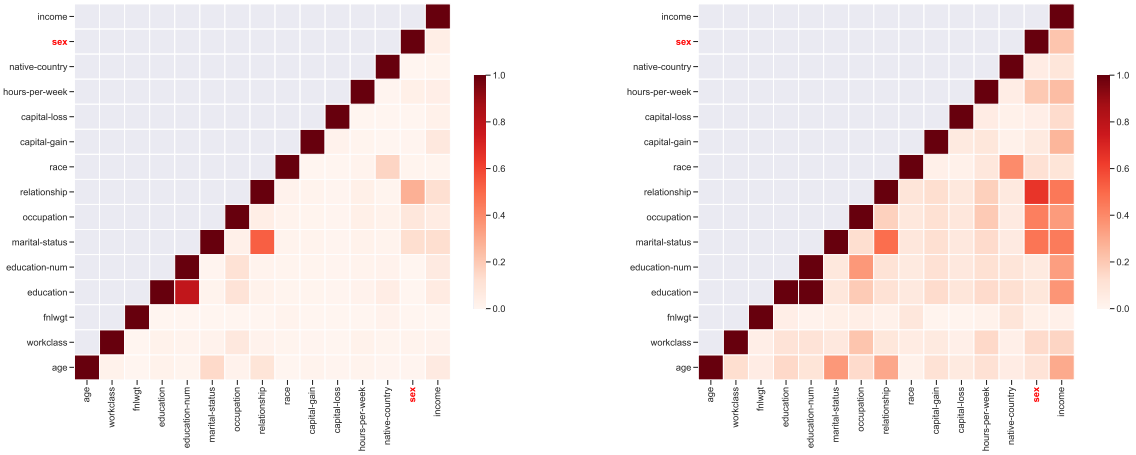
An overview of the Adult Income dataset is presented in Table 4.2, where the data for the integer encoded version is shown between parenthesis. The differences between the two versions of the dataset are due to the 2,399 instances with missing values, which are discarded in the integer encoded version. The favourable classifications (`high income`) represent 24.08% and 24.89% of the training data, for the one-hot and the integer encoded version of Adult Income, respectively.

		Sensitive Attribute	
		Male	Female
True Label	High income	6,662 (6,396)	1,179 (1,112)
	Low income	15,128 (13,984)	9,592 (8,670)

Table 4.2: Overview of the one-hot (integer) encoded version of the Adult Income dataset.

For the one-hot encoded version, the unprivileged group (`females`) represents around 33.08% of the dataset, with only around 15.04% of the favourable classifications being assigned to individuals from this group. In the integer encoded version of Adult Income, `females` represent around 32.43% of the training data and around 14.81% of the favourable classifications, after removing the missing values.

Figures 4.1a and 4.1b show the pairwise normalised mutual information and the pairwise Cramer's V between features and true labels, for the integer encoded version of the Adult Income dataset. Focusing on the last two columns of each figure, we can see that the features `relationship`, `marital-status`, and `occupation` are highly associated with the sensitive attribute `sex`. Not only do these features share information with the sensitive attribute, but also with the true labels. Removing the sensitive attribute before training the models may not be sufficient to eliminate the unfairness found in the dataset. In fact, it is likely that indirect discrimination emerges in the classifications made by the models.



(a) Normalised mutual information between features and response variable.

(b) Cramer's V between features and response variable.

Figure 4.1: Integer encoded version of Adult Income, with `sex` as the sensitive attribute.

Figure A.1 from Appendix A shows the pairwise NMI for the one-hot encoded version of Adult Income. According to the obtained results, `relationship_Husband`, `relationship_Wife`, `marital-status_Married-civ-spouse`, and `relationship_Unmarried` have the highest dependency with the sensitive attribute.

Figure A.2, also from Appendix A, shows the pairwise Cramer's V between features and true labels for the same version of this dataset. Special attention should be given to attributes `relationship_Husband` and `marital-status_Married-civ-spouse`, since they are highly associated with the sensitive attribute and the true labels. For this reason, anML model is still likely to pick up on these relations, leading to indirect discrimination. Although having weaker associations, `relationship_Wife`, `relationship_Unmarried`, `hours-per-week_bin40`, and `hours-per-week_bin99` should also be monitored.

Nevertheless, the last two of these features seem to be a reasonable choice considering the classification task at hand.

4.2 German Credit Data dataset

Table 4.3 shows the number of different categories per feature of the integer encoded version of German Credit Data. Features with two categories can be represented by a single binary variable, leading to a total of 71 features in the one-hot encoded version of this dataset.

Feature Name	Number of Categories	Feature Name	Number of Categories
status	4	property	4
duration	4	installment-plans	3
credit-history	5	housing	3
purpose	10	num-credits	4
credit-amount	4	job	4
savings	5	num-people-liable-for	2
employment-since	5	telephone	2
installment-rate-pct	4	foreign-worker	2
other-debtors	3	sex	2
residence-since	4	age	2

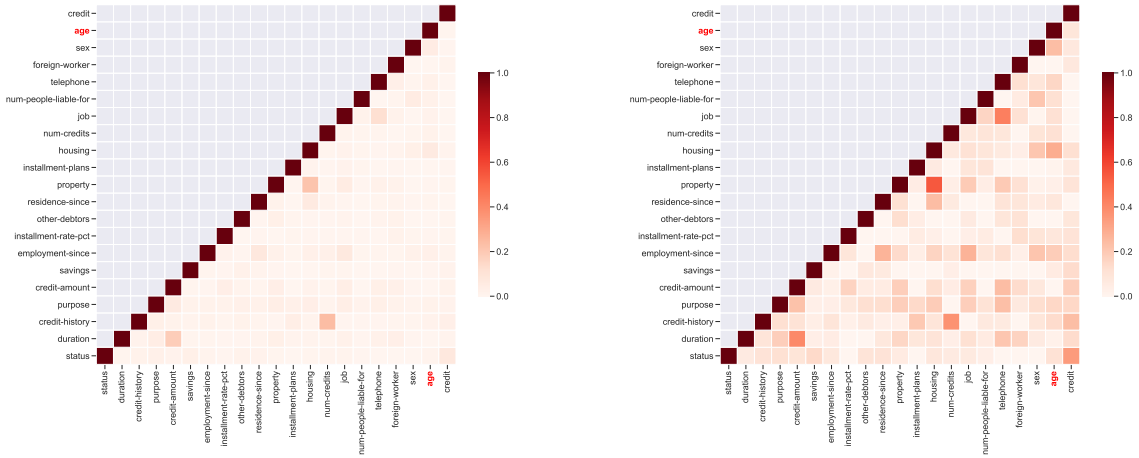
Table 4.3: Number of categories per feature - integer encoded version of German Credit.

Table 4.4 presents an overview of the German Credit Data dataset. In this case, `young` individuals, the unprivileged group, are represented by 15.00% of the dataset. The favourable classifications (`good credit`) represent 70.00% of the training data, with only 12.65% of them being assigned to the unprivileged group.

		Sensitive Attribute	
		Aged	Young
True Label	Good credit	428	62
	Bad credit	167	43

Table 4.4: Overview of the German Credit Data dataset.

For the integer encoded version of German Credit Data, the pairwise normalised mutual information and the pairwise Cramer's V between features and true labels are shown in Figures 4.2a and 4.2b, respectively. As far as mutual information is concerned, there seems to be no strong relation between the sensitive attribute `age` and any of the other features, with only a slightly higher value for the `housing` attribute. The relation between the sensitive attribute and `housing` is corroborated by the results with Cramer's V, which more clearly show the association between features and the response variable. Besides `housing`, we can see that attributes like `employment-since` and `sex` are associated with both the sensitive attribute and the true labels, meaning that there is room for indirect discrimination to occur. Although not being highly associated with the sensitive attribute, `status`, `credit-history`, and `purpose` have some association with the true labels, and so, caution should also be taken with regards to these features.



(a) Normalised mutual information between features and response variable.

(b) Cramer's V between features and response variable.

Figure 4.2: Integer encoded version of German Credit, with `age` as the sensitive attribute.

Figure A.3, which can be found in Appendix A, shows the pairwise NMI for the one-hot encoded version of the German Credit Data dataset. The feature which shares the most information with the sensitive attribute is `housing_A151`, but with a value of normalised mutual information which is below 0.10.

Figure A.4, also from Appendix A, shows the pairwise Cramer's V for the same version of German Credit Data. Special attention should be given to `housing_A151`, `sex`, and `housing_A152` which are the features with the strongest association with the sensitive attribute, while also being slightly associated with the true labels. Thus, there is a risk that the classifications made by an ML model trained with this data may suffer from indirect discrimination.

4.3 COMPAS dataset

The number of different categories for each feature of the integer encoded version of the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) dataset is shown in Table 4.5. Bearing in mind that binary features can be represented by a single dummy variable, we can conclude that the one-hot encoded version of the dataset is composed by 32 features.

Feature Name	Number of Categories	Feature Name	Number of Categories
sex	2	juv_other_count	2
age	4	priors_count	4
age_cat	3	c_charge_degree	2
juv_fel_count	2	c_charge_desc	15
juv_misd_count	2	race	2

Table 4.5: Number of categories per feature - integer encoded version of COMPAS.

An overview of the COMPAS dataset is shown in Table 4.6, where the data for the integer encoded version is shown between parenthesis. There are only four instances in the training

data which have missing values, and so, that is the only difference between the two versions of the dataset when it comes to the distributions of the true labels and the sensitive attribute. The favourable classifications (`no recidivist`) correspond to approximately 54.49% and 54.45% of the training data, for the one-hot and the integer encoded version of the dataset, respectively. Thus, the dataset does not have a significant imbalance with respect to the true labels.

		Sensitive Attribute	
		Caucasian	Other
True Label	No recidivist	897 (894)	1,457 (1,456)
	Recidivist	575 (575)	1,391 (1,391)

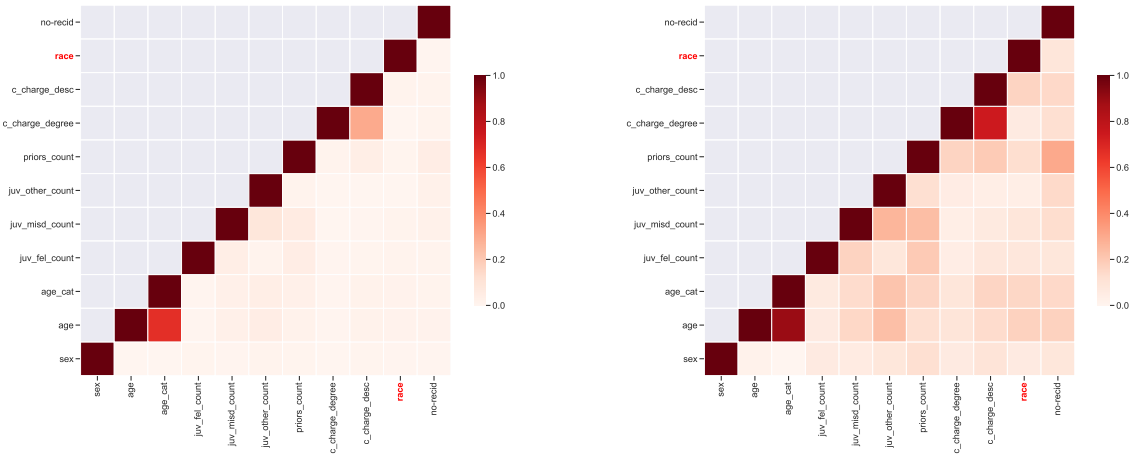
Table 4.6: Overview of the one-hot (integer) encoded version of the COMPAS dataset.

The unprivileged group (`other`) corresponds to around 65.93% and 65.96% of the training instances, for the one-hot encoded and the integer encoded version of the dataset, respectively. Since all instances with missing values correspond to `no recidivists`, the percentage of unfavourable classifications given to individuals from the unprivileged group is around 70.75% for both versions of the dataset.

Contrary to the other two datasets, in which there is an under-representation of the unprivileged group in terms of favourable classifications, in the COMPAS dataset there is an under-representation of the privileged group in terms of unfavourable classifications.

For the integer encoded version of the COMPAS dataset, the pairwise mutual information and the pairwise Cramer's V between attributes and true labels are shown in Figures 4.3a and 4.3b. According to Figure 4.3a, no feature seems to share a lot of information with the sensitive attribute `race`, nor with the true labels.

If we focus our attention on Figure 4.3b, we can observe that the `age` and the `age_cat` features have some association with the sensitive attribute and the true labels. Since this association is not strong, indirect discrimination does not seem likely to occur. It is also worth noting that `priors_count` is the feature with the highest association with the true labels, which seems fitting given the classification task for this dataset.



(a) Normalised mutual information between features and response variable.

(b) Cramer's V between features and response variable.

Figure 4.3: Integer encoded version of COMPAS, with `race` as the sensitive attribute.

Figures A.5 and A.6 from Appendix A present the pairwise NMI and Cramer's V for the one-hot encoded version of the COMPAS dataset. Similar to the integer encoded version of the dataset, an analysis based on mutual information does not show any strong relations between the dataset's features and the sensitive attribute or the true labels. However, this version of the dataset allows for a finer-grained analysis of the associations with the sensitive attribute and the true labels. We can see that the features `age_cat_Greater than 45` and `age_bin96` are the ones with a stronger association with the sensitive attribute, but the value of Cramer's V is lower than 0.20. When it comes to the true labels, there seems to be a stronger association with `priors_count_bin38` and `priors_count_bin0`.

4.4 Conclusion

We showed that the datasets used in our experiments have different characteristics that may lead to distinct fairness problems. While the Adult Income and the German Credit Data datasets are imbalanced with respect to true labels, as well as with respect to the sensitive attribute, the COMPAS dataset has no significant imbalance in this sense. Moreover, in the first two datasets there is an under-representation of individuals from the protected group with positive classifications, while the COMPAS dataset has an over-representation of individuals from that same group with negative classifications.

As far as associations between attributes are concerned, each dataset also poses different challenges. In comparison to the remaining two datasets, Adult Income has non-sensitive features with much more accentuated associations with the sensitive attribute. Besides these associations, the same non-sensitive features are also highly associated with the true labels. For this reason, the emergence of indirect discrimination seems much more likely with this dataset.

Chapter 5

Results and Discussion

We present and discuss the results of our study in this chapter. We measure fairness at different stages of the Machine Learning (ML) pipeline and, whenever possible, make a comparison between the fairness of the training data and that of the classifications.

Regarding the fairness of the training data, we evaluate the impact of the encoding of the categorical features and the sampling method. As far as the fairness of the classifications is concerned, we analyse these design choices, as well as the removal of the sensitive attribute and the learning algorithm. We are then able to compare the fairness of the training data to that of the classifications, and finally analyse the trade-off between fairness and classification performance.

5.1 Fairness of the Training Data

In this section, we start by measuring the fairness of the complete training set of each dataset, followed by the analysis of the fairness of the training data after a sampling method is applied, as depicted by **(4a)** and **(4b)** in Figure 3.1. We compare the obtained results with the baseline case in which no sampling method is used. This analysis is based on three fairness metrics that can be applied at the data-level: Statistical Parity Difference (SPD), the Normalised Prejudice Index (NPI), and Disparate Impact (DI). The fairness of the data is better for lower values of SPD and NPI, while higher values of DI correspond to fairer data.

Table 5.1 shows the fairness measurements for the one-hot encoded and the integer encoded versions of the three datasets used in the experiments, when considering the complete training set.

DATASET	ENCODING	SPD	NPI	DI
Adult Income	one-hot	0.19628	0.04353	0.35802
	integer	0.20016	0.04360	0.36222
German Credit Data	one-hot	0.12885	0.00947	0.82087
	integer	0.12885	0.00947	0.82087
COMPAS	one-hot	0.09779	0.00655	0.83953
	integer	0.09716	0.00647	0.84035

Table 5.1: Fairness measurements of the training set of each dataset.

After removing the missing values in the integer encoded version of Adult Income, the SPD and the NPI suffer an increase, suggesting that the data is less fair than in the one-hot encoded version. On the other hand, DI suggests a slight improvement on fairness when moving from the one-hot encoded version to the integer encoded version. However, none of these versions can be considered fair under the 80% rule, whose definition can be found in Section 2.2.1. These values of SPD and DI reveal discrimination against `females`, in the sense that they are less likely to be assigned a `high income` classification than `males`. There also seems to be room for indirect discrimination, as shown by the NPI values, since the sensitive attribute and the response variable are not independent.

The fairness measurements are the same for both versions of the German Credit Data dataset when taking `age` as the sensitive attribute. According to the 80% rule, the training set of this dataset can be considered fair, although being close to that legal threshold. Moreover, `young` individuals might still be receiving an unfair treatment, as expressed by SPD and NPI.

With the removal of the four instances with missing values in the integer encoded version of the COMPAS dataset, the SPD and the NPI both suffer a slight decrease and the DI also improves. Both versions of the dataset can be considered fair under the 80% rule. Similar to what was observed with German Credit Data, the SPD and the NPI show that individuals from the unprivileged group may still be unfairly treated.

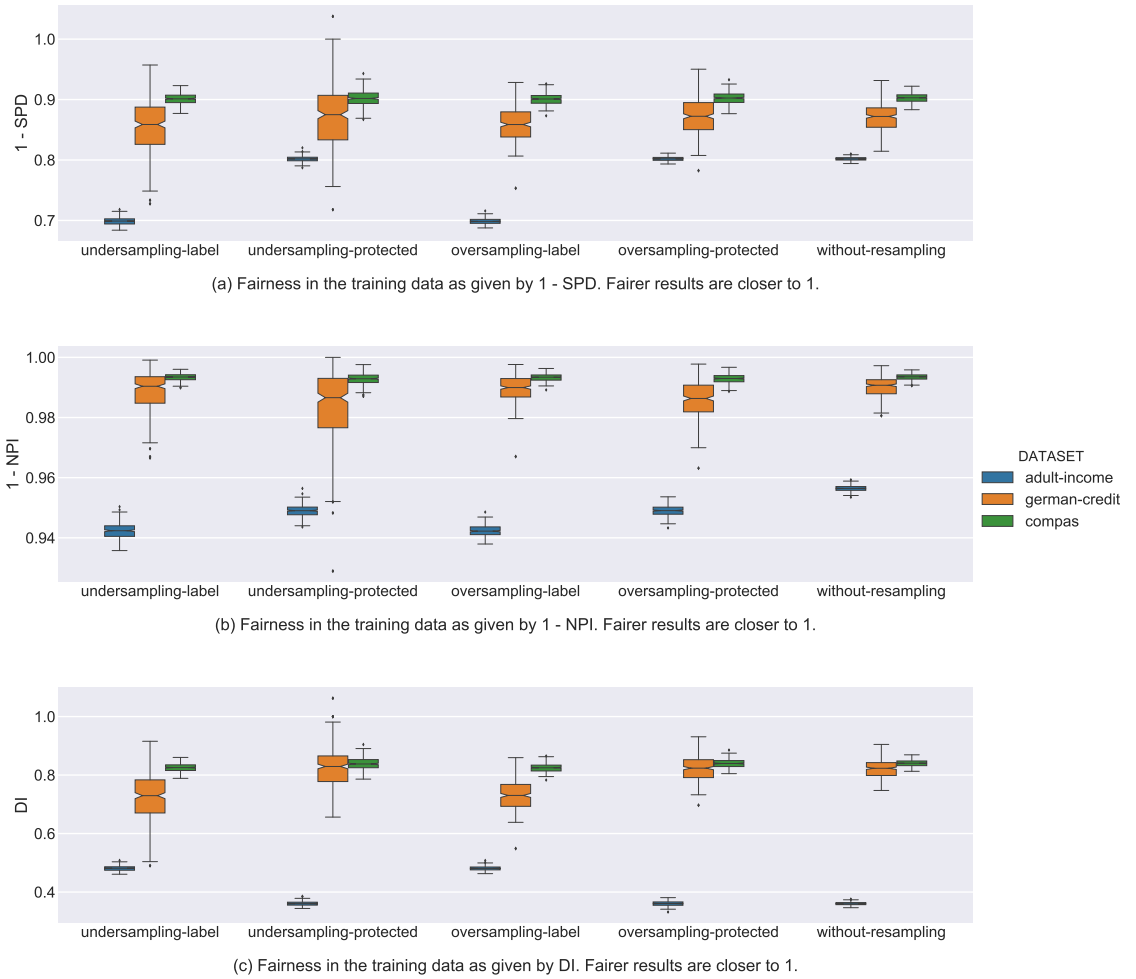
These results indicate that, for Adult Income, the one-hot encoded version of the dataset is fairer than the integer encoded version, while for the COMPAS dataset the opposite is verified. However, the number of instances containing missing values for COMPAS is far lower than that of Adult Income, with the difference of fairness between the two versions of COMPAS also being one order of magnitude lower for SPD and DI.

As far as the training data is concerned, the encoding of the categorical features does not seem to have a great impact on fairness under any of the three fairness metrics. This is true because both the sensitive attribute and the true labels are binary, and so, the encoding of the categorical features only affects the fairness of the training data when there are missing values. Even for the dataset with the largest number of instances containing missing values, the Adult Income dataset, their removal only seems to have a slight impact on the fairness of the complete training set.

For this reason, the analysis that follows does not differentiate between the two versions of the same dataset, and for each combination of dataset and sampling method we consider 300 measurements of each metric.

Figure 5.1 shows the distributions of the fairness measurements in the training data, after the application of a sampling method, as depicted by **(4b)** in Figure 3.1. We present the additive inverse of SPD and NPI so that the optimal fairness is 1 regardless of the metric. The results for `undersampling-multivariate` and `oversampling-multivariate` are not shown in the figure since their application makes the training data have optimal fairness, with zero variance. As far as NPI is concerned, the measurements are not always equal to zero, but due to their order of magnitude the observed differences are negligible.

The application of a sampling method does not change the relative degree of unfairness in the training data of each dataset, with the data of Adult Income being the most unfair and the one of the COMPAS dataset being the fairest. Furthermore, the impact of the sampling method on the fairness of the training data also seems to be dependent on the unfairness originally found in the complete set of training instances. The more unfair the original set of data, the more accentuated the effects of different sampling methods.



(a) Fairness in the training data as given by 1 - SPD. Fairer results are closer to 1.

(b) Fairness in the training data as given by 1 - NPI. Fairer results are closer to 1.

(c) Fairness in the training data as given by DI. Fairer results are closer to 1.

Figure 5.1: Fairness in the training data after applying a sampling method for the Adult Income, the German Credit Data, and the COMPAS datasets.

We would like to remind that, according to the 80% rule, a dataset can be considered fair if DI is above 80%. Using the median as a reference, we observe that the DI of the training data of Adult Income remains below this legal threshold after the application of any of the five sampling strategies shown in Figure 5.1, while the DI of the training data of the COMPAS dataset remains above it even after random undersampling or random oversampling is performed. For the German Credit Data dataset, performing `undersampling-label` or `oversampling-label` will cause the DI of the training data to fall below the legal threshold, while any of the other methods select instances so that the 80% rule still holds true.

However, random undersampling and random oversampling seem to have approximately the same behaviour when applied with respect to the same variable, independently of the metric used to measure fairness. In other words, there is no major difference in terms of the fairness of the training instances resulting from the application of `undersampling-label` and `oversampling-label`, or from the application of `undersampling-protected` and `oversampling-protected`. Nevertheless, there are differences between sampling methods when applied with respect to different variables, as explained in what follows.

Besides not being the same across datasets, the group of best sampling strategies in terms of fairness of the training data is dependent on the metric used to perform the analysis.

Only SPD leads to the same conclusions for all the datasets, suggesting that random undersampling and random oversampling with respect to the true labels (`undersampling-label` and `oversampling-label`, respectively) are the worst choices, i.e. lead to more unfair results. This is also the conclusion we reach when analysing German Credit Data and COMPAS with DI. However, NPI suggests that the worst sampling strategies for German Credit and COMPAS are `undersampling-protected` and `oversampling-protected`. For these two datasets, `without-resampling` can always be paired with the best sampling strategies after `undersampling-multivariate` and `oversampling-multivariate`.

For Adult Income, `undersampling-protected`, `oversampling-protected`, and `without-resampling` are better than `undersampling-label` and `oversampling-label` according to SPD and NPI, while being worse according to DI. Additionally, when using NPI with this dataset, `without-resampling` presents a clearer advantage over the two methods that can be paired with it in terms of fairness of the training data. These contradicting results, which are far more evident with Adult Income, are an indication that these metrics may be brittle, and demand for a further analysis.

The discrepancies found between fairness metrics are well-known in other fields, sharing similar problems to those posed by risk difference and relative risk. The results with SPD are more in an absolute sense and do not translate whether the measurements are good or bad. On the other hand, DI gives a more accurate idea of how the measurements between the protected and unprotected groups relate to one another. Contradictions between the two metrics occur when $P(Y = 1|S = 0)$ becomes smaller between two distinct configurations, evidencing the sensitivity of DI to lower values of the numerator. This behaviour is observed regardless of the measurements given by SPD. From a fairness point-of-view, this scenario occurs when the percentage of individuals from the protected group which are assigned a favourable classification becomes smaller. This also shows the importance of clearly understanding the metrics that are being used to support the analysis and decisions made based on the models' classifications.

There are also differences in the variability of the results. After `undersampling-multivariate` and `oversampling-multivariate`, the training instances selected by `without-resampling` have the smallest variance in terms of fairness. Performing random oversampling usually leads to smaller variances than performing random undersampling with respect to the same variable. Nevertheless, the sampling method leading to the largest variances varies between `undersampling-label` and `undersampling-protected`. As depicted in Figure 5.1, the variance is larger for German Credit Data, the smallest of the three datasets used in our study. This might be an indication that the size of the dataset is important to grasp the true values of (un)fairness.

5.2 Fairness of the Classifications

In this section, we focus on measuring the fairness of the classifications made by the different models. We aim at understanding the impact that the encoding of the categorical features, the sampling method, the removal of the sensitive attribute prior to training, and the learning algorithm have on fairness. For some metrics, we make a slight modification and consider the additive inverse so as to facilitate the analysis. By doing so, measurements closer to one are fairer under all the metrics used in this analysis. The results from the previous section seem to suggest that considering the ratio variant of group-conditioned accuracy metrics may be better than the difference variant, since we want to make relative comparisons between different configurations. Moreover, these variants have a similar

rationale to DI, with the 80% rule being an actual guideline adopted by governments.

Preliminary results suggest that the learning algorithm, together with the removal of the sensitive attribute, is one of the factors that influence the fairness in the classifications of an ML model the most. For this reason, we always show the results for all tested versions of the learning algorithms, even when analysing the impact of other variables.

We start by analysing the impact of the encoding, with the obtained results for all datasets being shown in Table 5.2. The average is taken across 30 runs of five-fold cross-validation, which means that we consider 1,050 measurements of each metric for each combination of dataset, classifier and encoding.

DATASET	CLASSIFIER	ENCODING	1 - SPD	1 - NPI	DI	1 - GE	S-TPR-ratio	S-TNR-ratio
Adult Income	DT	integer	0.80409	0.96179	0.43795	0.89675	0.87572	1.15600
		one-hot	0.80497	0.96042	0.42386	0.89943	0.86166	1.15232
	DTns	integer	0.81353	0.96594	0.45987	0.89686	0.89708	1.14323
		one-hot	0.81255	0.96400	0.44126	0.89945	0.87870	1.14224
	RF	integer	0.76622	0.94630	0.36831	0.91523	0.86774	1.18243
		one-hot	0.76950	0.94511	0.35367	0.91754	0.85414	1.17690
	RFns	integer	0.77378	0.95050	0.38226	0.91483	0.88468	1.17278
		one-hot	0.77346	0.94767	0.36211	0.91752	0.86630	1.17197
German Credit Data	DT	integer	0.89296	0.98471	0.84605	0.80745	0.85209	1.12389
		one-hot	0.90032	0.98582	0.85367	0.79874	0.86664	1.11021
	DTns	integer	0.93448	0.99026	0.90423	0.80632	0.91570	1.04990
		one-hot	0.93371	0.99010	0.90060	0.79688	0.91628	1.04735
	RF	integer	0.86928	0.97958	0.81649	0.84127	0.85003	1.21003
		one-hot	0.89321	0.98447	0.84516	0.83061	0.88028	1.13561
	RFns	integer	0.91532	0.98753	0.87793	0.83979	0.90653	1.08421
		one-hot	0.91847	0.98850	0.87906	0.82858	0.91576	1.06458
COMPAS	DT	integer	0.88746	0.98877	0.81307	0.79199	0.90664	1.25226
		one-hot	0.88496	0.98827	0.80918	0.79202	0.90603	1.26005
	DTns	integer	0.86897	0.98749	0.78121	0.79946	0.85362	1.22240
		one-hot	0.86828	0.98750	0.77912	0.79927	0.85264	1.22115
	RF	integer	0.84538	0.97945	0.77253	0.82568	0.88150	1.41733
		one-hot	0.84260	0.98002	0.76242	0.82382	0.87035	1.37756
	RFns	integer	0.84573	0.98261	0.76376	0.82868	0.86818	1.34453
		one-hot	0.84884	0.98352	0.76283	0.82240	0.86087	1.30607

Table 5.2: Average fairness results, grouped by dataset, classifier and encoding.

First of all, there is no clear winner across datasets, with the choice of better encoding also being dependent on the fairness metric we are using to perform the analysis. For the German Credit Data dataset, opting for one-hot encoding seems to be the safest choice when using DT, RF, and RFns, while DTns seem to be capable of making fairer classifications when trained with integer encoded data. As far as the fairness metrics are concerned, two clusters appear to emerge: SPD, NPI, and DI; and S-TPR-ratio together with S-TNR-ratio.

For the Adult Income and the COMPAS datasets, the choice of the best encoding seems harder to make. For Adult Income, the results with NPI, DI, and S-TPR-ratio suggest that one should opt for an integer encoding of the data. On the other hand, Generalised Entropy Index (GE) and S-TNR-ratio suggest that a one-hot encoding would be better. The choice according to SPD seems to be more dependent on whether the classifiers have access to the sensitive attribute: if so, one-hot encoding is better, otherwise the choice should be an integer encoding. In fact, each fairness metric agrees upon the best encoding for these two datasets if the determinant factor is the removal of the sensitive attribute prior to training the model.

Each fairness metric agrees upon the best encoding for DTns, RF, and RFns trained with data from the COMPAS dataset, except for SPD. Furthermore, two clusters of metrics are formed: NPI and S-TNR-ratio suggest that models trained with one-hot encoded data make fairer classifications, while DI, GE, and S-TPR-ratio suggest that integer encoded

data is better.

When it comes to models based on DT, integer encoding is the best option according to NPI, DI, S-TPR-ratio, and S-TNR-ratio, while GE suggest that one-hot encoding is better. Moreover, integer encoded data allows for any of the tested classifiers to make fairer predictions according to DI and S-TPR-ratio. According to SPD, integer encoding is better except for models based on RFns.

It is important to mention that the reported differences between encodings of the categorical features are, in general, of a lower order of magnitude than the differences that emerge when varying other aspects, like the classifier or the sampling method. For this reason, the analysis that follows will not differentiate between models trained with integer encoded data or one-hot encoded data.

The results shown in Table 5.3 correspond to the average fairness in the models' classifications, according to the selected set of fairness metrics, grouped by dataset and classifier. Results are averaged across 30 runs of five-fold cross-validation, which means that we consider 2,100 measurements of each metric for each combination of dataset and classifier.

DATASET	CLASSIFIER	1 - SPD	1 - NPI	DI	1 - GE	S-TPR-ratio	S-TNR-ratio
Adult Income	DT	0.80453	0.96110	0.43090	0.89809	0.86869	1.15416
	DTns	0.81304	0.96497	0.45056	0.89816	0.88789	1.14274
	RF	0.76786	0.94570	0.36099	0.91639	0.86094	1.17966
	RFns	0.77362	0.94909	0.37219	0.91618	0.87549	1.17237
German Credit Data	DT	0.89664	0.98527	0.84986	0.80310	0.85937	1.11705
	DTns	0.93409	0.99018	0.90242	0.80160	0.91599	1.04862
	RF	0.88125	0.98203	0.83082	0.83594	0.86516	1.17282
	RFns	0.91689	0.98802	0.87849	0.83418	0.91114	1.07440
COMPAS	DT	0.88621	0.98852	0.81113	0.79200	0.90633	1.25615
	DTns	0.86863	0.98749	0.78016	0.79937	0.85313	1.22178
	RF	0.84399	0.97973	0.76747	0.82475	0.87592	1.39745
	RFns	0.84729	0.98306	0.76329	0.82554	0.86453	1.32530

Table 5.3: Average fairness results, grouped by classifier, for the three datasets.

The obtained results for Adult Income and German Credit Data suggest that the removal of the sensitive attribute usually leads to an improvement of the fairness in the classifications made by a ML model, as expected. However, some exceptions occur when measuring fairness with GE. Under this fairness metric, the removal of the sensitive attribute always leads to more unfair classifications for German Credit data, the same happening for Adult Income but only between RF and RFns. For the COMPAS dataset, removing the sensitive attribute is also, in general, a good practice if one wants to improve the fairness of the classifications. Exceptions also occur for this dataset, this time when the analysis is performed with metrics other than GE. Most of these exceptions occur with models based on Decision Trees, specifically with SPD, NPI, DI, and S-TPR-ratio. The results with these last two metrics lead to the only exceptions with models based on Random Forests.

Regarding the learning algorithm, choosing Decision Trees is usually preferable to choosing Random Forests for all three datasets. Exceptions to this behaviour occur between DT and RF for the German Credit Data dataset, and between DTns and RFns for the COMPAS dataset, when fairness is given by S-TPR-ratio. Additionally, the preference of Decision Trees over Random Forests is not supported by the measurements with GE which always suggest that the best models are those based on Random Forests.

We now focus on the analysis of the impact of the sampling method on the classifications' fairness. Table 5.4 shows the obtained results for the Adult Income dataset, grouped by

classifier and sampling method, and averaged across 30 runs of five-fold cross-validation. For each combination of classifier and sampling method, we have 300 measurements of each metric.

CLASSIFIER	SAMPLING	1 - SPD	1 - NPI	DI	1 - GE	S-TPR-ratio	S-TNR-ratio
DT	oversampling-label	0.75225	0.94192	0.33977	0.90085	0.76095	1.21376
	oversampling-multivariate	0.78604	0.95491	0.38412	0.89866	0.80476	1.16576
	oversampling-protected	0.83567	0.96800	0.41400	0.89378	0.86527	1.10313
	undersampling-label	0.74293	0.94830	0.43080	0.90702	0.86012	1.25905
	undersampling-multivariate	0.85773	0.98456	0.65934	0.90008	1.11102	1.11316
	undersampling-protected	0.82432	0.96355	0.38673	0.89228	0.83629	1.11977
	without-resampling	0.83277	0.96648	0.40155	0.89394	0.84241	1.10448
DTns	oversampling-label	0.76527	0.94830	0.36790	0.90081	0.78926	1.19607
	oversampling-multivariate	0.79696	0.95979	0.41123	0.89847	0.83101	1.15223
	oversampling-protected	0.84599	0.97199	0.44273	0.89372	0.88999	1.09059
	undersampling-label	0.75933	0.95483	0.45930	0.90699	0.88418	1.23155
	undersampling-multivariate	0.84630	0.98203	0.63570	0.90055	1.09617	1.13042
	undersampling-protected	0.83497	0.96757	0.41004	0.89254	0.85684	1.10549
	without-resampling	0.84248	0.97026	0.42704	0.89402	0.86778	1.09280
RF	oversampling-label	0.68732	0.91635	0.29145	0.92086	0.78812	1.28991
	oversampling-multivariate	0.76153	0.94935	0.38804	0.91865	0.89999	1.17911
	oversampling-protected	0.81856	0.95634	0.32277	0.91132	0.81354	1.09268
	undersampling-label	0.67306	0.91756	0.33886	0.92387	0.84912	1.34687
	undersampling-multivariate	0.80475	0.97109	0.55764	0.91789	1.05141	1.15268
	undersampling-protected	0.81380	0.95419	0.31249	0.91041	0.81262	1.10093
	without-resampling	0.81601	0.95503	0.31567	0.91170	0.81176	1.09548
RFns	oversampling-label	0.70466	0.92576	0.32012	0.92054	0.82254	1.26570
	oversampling-multivariate	0.76633	0.95139	0.39646	0.91832	0.91197	1.17381
	oversampling-protected	0.82256	0.95836	0.33393	0.91120	0.83091	1.08931
	undersampling-label	0.69353	0.92789	0.37014	0.92348	0.87888	1.31302
	undersampling-multivariate	0.79048	0.96685	0.53365	0.91815	1.03266	1.17557
	undersampling-protected	0.81697	0.95583	0.32134	0.91025	0.82467	1.09812
	without-resampling	0.82084	0.95754	0.32967	0.91130	0.82679	1.09107

Table 5.4: Average fairness results, grouped by classifier and sampling, for Adult Income.

For the Adult Income dataset, combining `undersampling-multivariate` with DT or DTns is the best option according to all the fairness metrics, except GE and S-TNR-ratio. According to GE, applying `undersampling-label` seems to be the best option, while S-TNR-ratio suggests that the best option is `oversampling-protected`. For models based on RF or RFns, the conclusions are the same, except that `oversampling-protected` is also the best sampling method according to SPD.

The average fairness results for the German Credit Data dataset, grouped by classifier and sampling method, are shown in Table 5.5.

For the German Credit Data dataset, the best sampling method to train DT is `undersampling-multivariate` according to all fairness metrics, except GE, while `undersampling-protected` is the best method to train DTns, according to all metrics except GE and S-TNR-ratio. The best sampling method to train RF and RFns is `undersampling-multivariate` according to NPI and S-TNR-ratio, and `undersampling-protected` according to DI and S-TNR-ratio. According to SPD, `undersampling-protected` is also the best sampling method to train RFns. Performing random oversampling with respect to the sensitive attribute (`oversampling-protected`) allows for fairer classifications when training DTns, according to S-TNR-ratio. GE suggests that performing `oversampling-multivariate` is best to train DT and DTns, while `oversampling-protected` is better for RF and `without-resampling` is better for RFns.

CLASSIFIER	SAMPLING	1 - SPD	1 - NPI	DI	1 - GE	S-TPR-ratio	S-TNR-ratio
DT	oversampling-label	0.90437	0.98680	0.87064	0.83261	0.87410	1.09838
	oversampling-multivariate	0.88563	0.98300	0.84594	0.83316	0.85842	1.16801
	oversampling-protected	0.87554	0.98205	0.83020	0.83305	0.84954	1.19609
	undersampling-label	0.86230	0.98295	0.76296	0.74372	0.75669	1.08258
	undersampling-multivariate	0.94037	0.98940	0.90429	0.74012	0.92855	1.02585
	undersampling-protected	0.91655	0.98729	0.88505	0.81357	0.89687	1.12024
	without-resampling	0.89172	0.98539	0.84994	0.82545	0.85139	1.12821
DTns	oversampling-label	0.93228	0.98977	0.90962	0.83207	0.92704	1.07993
	oversampling-multivariate	0.93006	0.98979	0.90524	0.83309	0.91401	1.07637
	oversampling-protected	0.93908	0.99066	0.91677	0.83241	0.91709	1.00260
	undersampling-label	0.89965	0.98746	0.82599	0.74063	0.83622	1.05911
	undersampling-multivariate	0.94600	0.99133	0.90931	0.73957	0.93909	1.02423
	undersampling-protected	0.95290	0.99217	0.93366	0.80722	0.95167	1.04791
	without-resampling	0.93869	0.99007	0.91633	0.82621	0.92678	1.05023
RF	oversampling-label	0.86114	0.98111	0.81351	0.85567	0.86152	1.22327
	oversampling-multivariate	0.90721	0.98564	0.87795	0.85969	0.91006	1.12178
	oversampling-protected	0.87369	0.98014	0.84383	0.88111	0.87969	1.26916
	undersampling-label	0.81225	0.97347	0.67068	0.76032	0.70893	1.13574
	undersampling-multivariate	0.93139	0.99107	0.87809	0.74843	0.90313	1.00104
	undersampling-protected	0.90307	0.98263	0.88000	0.86529	0.91265	1.26326
	without-resampling	0.87998	0.98013	0.85172	0.88104	0.88012	1.19548
RFns	oversampling-label	0.89374	0.98583	0.85616	0.85543	0.88659	1.09980
	oversampling-multivariate	0.92270	0.98863	0.89778	0.85903	0.92927	1.07319
	oversampling-protected	0.93573	0.98920	0.92008	0.88012	0.94641	1.07994
	undersampling-label	0.87055	0.98434	0.77090	0.75783	0.82100	1.07755
	undersampling-multivariate	0.92421	0.99010	0.86735	0.74885	0.91145	1.04627
	undersampling-protected	0.94176	0.98876	0.92447	0.85777	0.95075	1.08716
	without-resampling	0.92955	0.98927	0.91271	0.88026	0.93252	1.05686

Table 5.5: Average fairness results, grouped by classifier and sampling, for German Credit.

Table 5.6 shows the average fairness results for the COMPAS dataset, grouped by classifier and sampling method. For this dataset, the best sampling method to train RF is `undersampling-multivariate`, according to all metrics except GE, which suggests that `without-resampling` is better. According to S-TNR-ratio, `undersampling-multivariate` is the best sampling method for RFns, while the remaining metrics suggest that the best one is `oversampling-protected`. The best sampling method to train DT and DTns according to SPD, NPI, and S-TNR-ratio is `undersampling-multivariate`. The remaining metrics all suggest a different sampling method to train DT, while `oversampling-protected` is suggested as the best sampling to train DTns according to DI and S-TPR-ratio.

In sum, all metrics suggest that, to deal with class imbalance in the training instances and get better fairness in the classifications, it is better to perform random undersampling / oversampling with respect to the sensitive attribute, or with respect to both this attribute and the true labels. This does not hold true when the analysis is made with GE. It is harder to find a pattern when considering this metric, except for the Adult Income dataset, for which training the models with `undersampling-label` seems to be the best option.

We would also like to point out the results obtained with S-TNR-ratio. According to this metric, the classifications are skewed in favour of the unprivileged group. The cost of a false positive and a false negative in contexts which pose fairness concerns should be further studied so as to ensure that fairness metrics also capture this aspect.

CLASSIFIER	SAMPLING	1 - SPD	1 - NPI	DI	1 - GE	S-TPR-ratio	S-TNR-ratio
DT	oversampling-label	0.86389	0.98640	0.76866	0.79007	0.87219	1.29263
	oversampling-multivariate	0.93152	0.99569	0.87714	0.79110	0.98044	1.16166
	oversampling-protected	0.84151	0.98160	0.75390	0.80881	0.85861	1.39875
	undersampling-label	0.86912	0.98725	0.76962	0.78286	0.87716	1.27230
	undersampling-multivariate	0.97718	0.99800	0.95825	0.76798	1.02026	1.03254
	undersampling-protected	0.85718	0.98458	0.77032	0.79811	0.85212	1.30524
	without-resampling	0.86306	0.98612	0.78000	0.80511	0.88354	1.32995
DTns	oversampling-label	0.85999	0.98605	0.76402	0.79802	0.84264	1.22812
	oversampling-multivariate	0.86760	0.98738	0.77071	0.78812	0.84071	1.20499
	oversampling-protected	0.87270	0.98805	0.80108	0.81698	0.87216	1.24661
	undersampling-label	0.85884	0.98581	0.75720	0.79285	0.84025	1.23001
	undersampling-multivariate	0.87764	0.98911	0.77560	0.77203	0.84056	1.17672
	undersampling-protected	0.87479	0.98851	0.80033	0.81047	0.86593	1.22494
	without-resampling	0.86884	0.98754	0.79219	0.81710	0.86964	1.24104
RF	oversampling-label	0.80887	0.97368	0.70943	0.81847	0.82654	1.45021
	oversampling-multivariate	0.91258	0.99373	0.85200	0.81515	0.95731	1.18351
	oversampling-protected	0.81285	0.97364	0.73719	0.83897	0.85193	1.54624
	undersampling-label	0.81202	0.97464	0.70940	0.81490	0.82898	1.43206
	undersampling-multivariate	0.93720	0.99584	0.89074	0.80853	0.97730	1.11515
	undersampling-protected	0.81189	0.97305	0.73678	0.83750	0.84010	1.52519
	without-resampling	0.81251	0.97355	0.73677	0.83973	0.84930	1.52977
RFns	oversampling-label	0.83698	0.98123	0.74046	0.81812	0.84972	1.32305
	oversampling-multivariate	0.84292	0.98255	0.74436	0.81165	0.84751	1.29513
	oversampling-protected	0.85718	0.98460	0.79284	0.84248	0.88900	1.35272
	undersampling-label	0.83687	0.98126	0.73896	0.81650	0.84898	1.32316
	undersampling-multivariate	0.84920	0.98377	0.75170	0.80779	0.84822	1.27269
	undersampling-protected	0.85680	0.98443	0.79268	0.84128	0.88393	1.34864
	without-resampling	0.85104	0.98357	0.78206	0.84096	0.88432	1.36174

Table 5.6: Average fairness results, grouped by classifier and sampling, for COMPAS.

5.3 Fairness Comparison between Data and Classifications

To understand whether the unfairness in the training data was increased or reduced under each configuration, we computed the ratio between the SPD in the classifications and the SPD in the data subset used to train the models (SPD Ratio), as well as a similar ratio regarding NPI (NPI Ratio). A value of 1 indicates that the unfairness in the classifications is the same as in the training data, an absolute value greater than 1 means that the unfairness in the classifications is greater, meaning that the models are able to find relationships in the data that increase the unfairness, and an absolute value lower than 1 means that the model makes fairer classifications than the procedure which produced the true labels of the training data. A DI Ratio was not computed since it would be difficult to interpret the results.

Caution must be taken when computing these ratios for models resulting from the application of `undersampling-multivariate` and `oversampling-multivariate`, since the subsets of data used to train them have a SPD or NPI equal to zero, as explained in Section 5.1. In such cases, the value represented in the boxplots corresponds to the SPD or the NPI in the classifications instead of the, otherwise invalid, value of the ratio. Furthermore, models trained with either of these two sampling methods always make classifications which are more unfair than the subset of training data used to train them, regardless of making this comparison using SPD or NPI.

The boxplots in Figure 5.2 represent the distributions of the SPD Ratio for the Adult Income dataset. The obtained results suggest that performing `undersampling-protected`, `oversampling-protected` or not performing random undersampling / oversampling at all (`without-resampling`) has similar effects on fairness, allowing for the creation of models likely to reduce the unfairness in the training data.

The opposite happens when models are trained with `undersampling-multivariate` or `oversampling-multivariate` which always increase it. Making a comparison with the fairness of the complete training set, we would notice that classifications made by DT and DTns are likely to reduce the unfairness in the data only if trained with `undersampling-multivariate`, while those made by RF and RFns are likely to increase it, especially if trained with `oversampling-multivariate`.

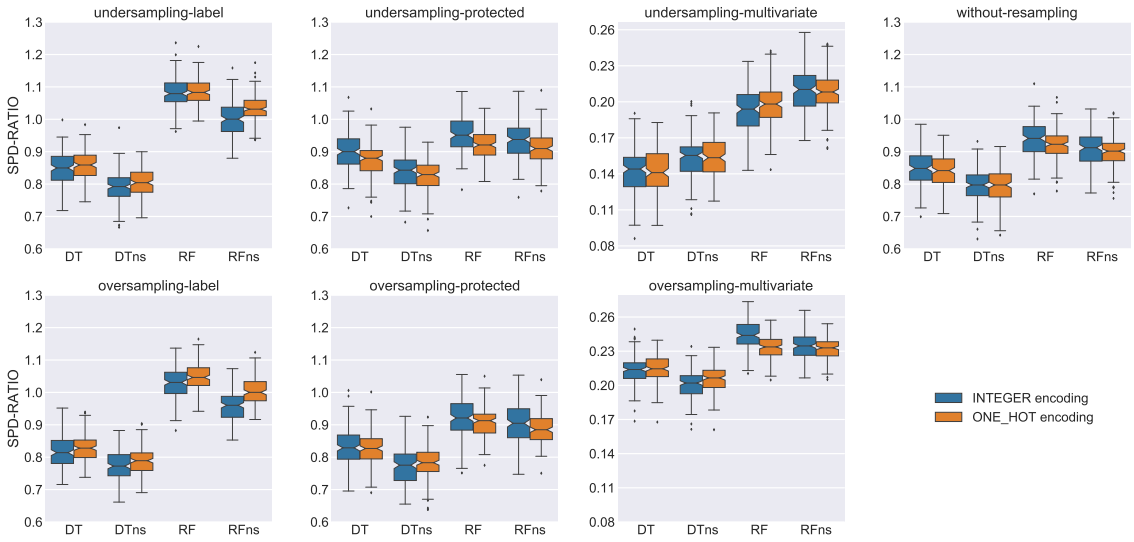


Figure 5.2: SPD Ratio for the Adult Income dataset.

When it comes to applying `undersampling-label` or `oversampling-label`, caution must be taken when choosing the learning algorithm. DT and DTns seem to reduce the unfairness in the data used to train them, while Random Forests trained with the sensitive attribute (RF) seem to increase it. The behaviour of Random Forests trained without the sensitive attribute (RFns) is also dependent on the encoding of the categorical features. The unfairness in the classifications made by RFns trained with integer encoded data and `undersampling-label`, as well as those of RFns trained with one-hot encoded data and `oversampling-label`, appears to be similar to the unfairness in the training data. RFns trained with one-hot encoded data and `undersampling-label` seem to make classifications with higher unfairness than the training data, while if trained with integer encoded data and `oversampling-label` their classifications appear to have lower unfairness.

These results suggest that when Random Forests are trained without having direct access to the sensitive attribute and with one-hot encoded data, indirect discrimination seems to emerge more prominently than when these models are trained under the same conditions but with integer encoded data.

The distributions of the NPI Ratio for Adult Income are represented by the boxplots in Figure 5.3. The NPI Ratio suggests that the unfairness in the classifications of models trained with `undersampling-protected` or with `oversampling-protected` is smaller than in the data used to train them. The unfairness in the classifications of models trained with `oversampling-multivariate` is likely to be higher than the one found in the complete set of training data, except when considering DTns. Although models trained with `undersampling-multivariate` make classifications more unfair than the instances used to train them, the classifications are fairer than the complete training set.

When it comes to `undersampling-label` and `without-resampling`, Decisions Trees tend to produce models capable of reducing the unfairness in the data used to train them.

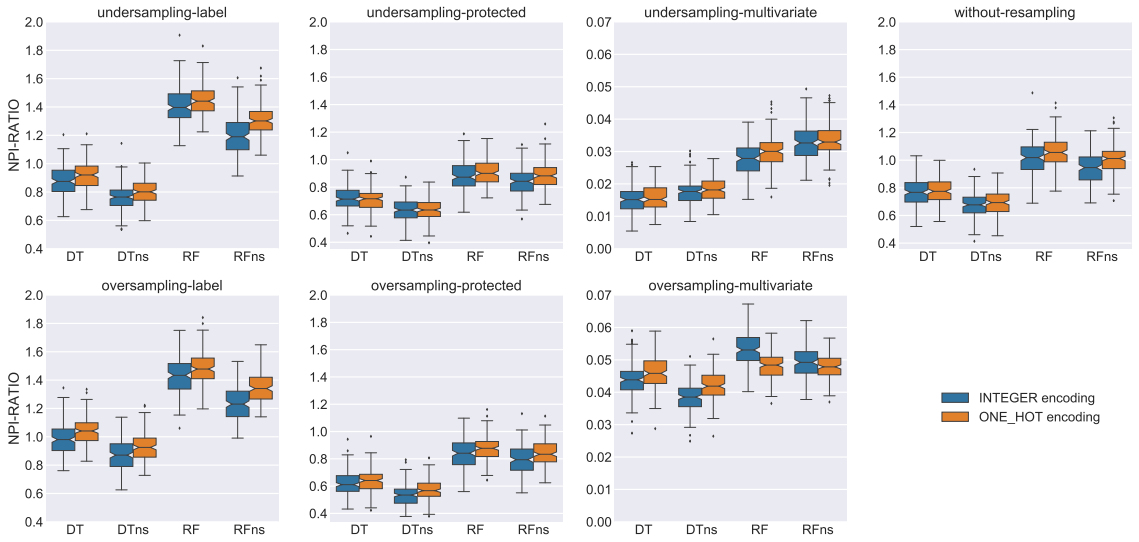


Figure 5.3: NPI Ratio for the Adult Income dataset.

On the other hand, the combination of `undersampling-label` or `oversampling-label` with Random Forests will likely produce models that tend to increase the unfairness in the training data, while the NPI of the classifications made by Random Forests trained `without-resampling` tends to be closer to the NPI of the data. Decision Trees trained with the sensitive attribute (DT) and with `oversampling-label` tend to make classifications with similar unfairness in comparison to the subset of data used to train them. If this data is one-hot encoded, the classifications are more likely to have a slightly higher unfairness than the data. Combining DTns and `oversampling-label` will create models whose unfairness is likely to be lower than that of the training instances.

The results with `undersampling-multivariate` and, in particular, with `oversampling-multivariate` suggest that even if the sensitive attribute is removed and the data used to train some model is completely fair under SPD or NPI, the model is still able to explore the remaining attributes and incorporate unfairness in its classifications. This is particularly relevant since it shows that removing the sensitive attribute and pre-processing the data by sampling is not enough to remove the unfairness in the data, highlighting the problem of having features that are highly associated with the sensitive attribute and the consequent emergence of indirect discrimination.

For the German Credit Data dataset, extreme outliers, mainly detected with the NPI Ratio, are not represented in the figures that follow to allow for a better visualisation of the distributions. The boxplots in Figure 5.4 represent the distributions of the SPD Ratio for this dataset.

The unfairness in the classifications of models trained with `undersampling-protected` tends to be lower than the one in the data used to train them. When it comes to `without-resampling`, the results for SPD Ratio suggest that models trained without the sensitive attribute (DTns and RFns) are likely to make classifications which are fairer than the training data. However, the same learning algorithms trained without the sensitive attribute are more likely to make classifications as fair as the training data, especially in the case of RF.

The behaviour of models trained with `oversampling-label` and `oversampling-protected` is similar to those trained `without-resampling`. However, RF combined with `over-`

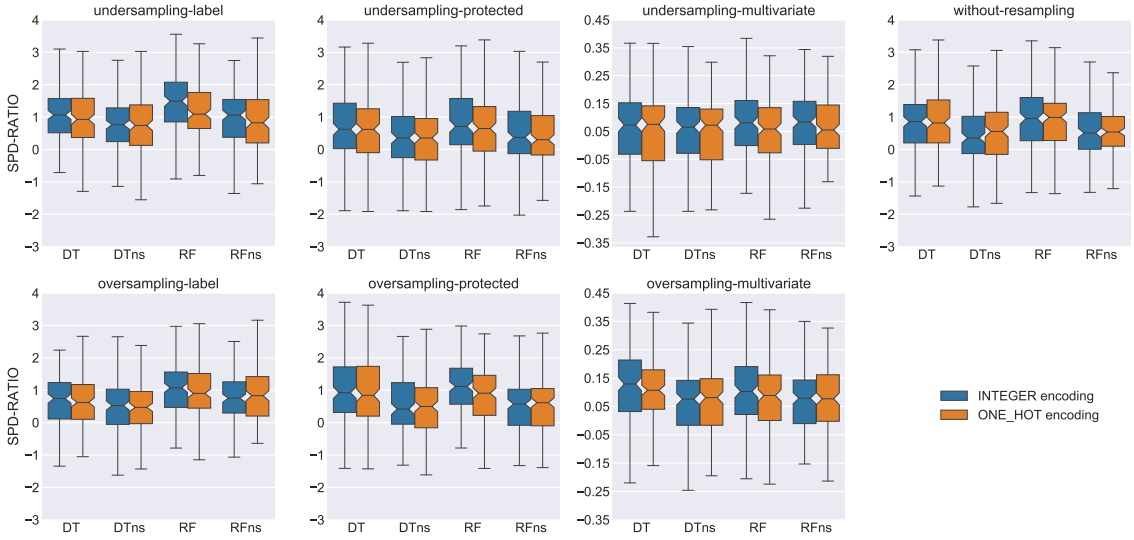


Figure 5.4: SPD Ratio for the German Credit Data dataset.

`sampling-protected` or with `oversampling-label` may make classifications less fair than the training data if trained with integer encoding, while being more likely to make fairer classifications than the data if one-hot encoding is used.

With `undersampling-label`, Random Forests trained with the sensitive attribute (RF) make classifications with an higher unfairness than that of the training data. Classifications made by DTns have a lower unfairness in comparison to the one found in the training data. Decision Trees trained with the sensitive attribute (DT) and RFns behave similarly: when trained with an integer encoding their classifications may increase the unfairness in the data, while if trained with one-hot encoded data they may reduce it.

The distributions of the NPI Ratio for German Credit Data are represented by the boxplots in Figure 5.5.

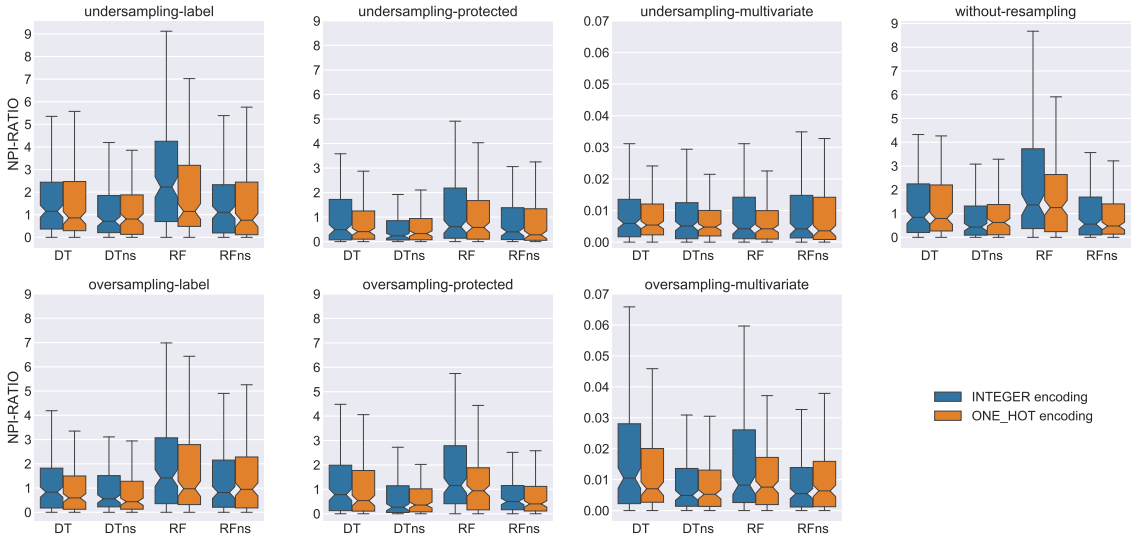


Figure 5.5: NPI Ratio for the German Credit Data dataset.

The results for the NPI Ratio suggest that the unfairness in the classifications made by models based on DTns is likely to be lower than the one in the data used to train them, except if `undersampling-multivariate` or `oversampling-multivariate` is applied.

With `undersampling-protected`, `oversampling-protected` or `without-resampling`, Decision Trees trained with the sensitive attribute (DT) and Random Forests trained without the sensitive attribute (RFns) are able to reduce the unfairness in the training data. However, Random Forests trained with the sensitive attribute (RF) and with `undersampling-protected` are likely to reduce the unfairness in the training data, while RF `without-resampling` seem likely to increase it. RF combined with `oversampling-protected` are likely to make classifications whose unfairness is close to the one in the one-hot encoded data used to train them, while if trained with integer encoded data the classifications may be more unfair.

With `undersampling-label`, the conclusions for RF are similar to those reached with SPD Ratio. DT and RFns show a similar behaviour, depending on the encoding: when trained with integer encoded data, the classifications might have higher unfairness than that of the training data, while with one-hot encoding the classifications tend to have lower unfairness.

As far as `oversampling-label` is concerned, RF tend to produce classifications with more unfairness than the data used to train them. Classifications made by RFns and DT are likely to be fairer than the training data. For both RF and RFns, if models are trained with one-hot encoded data, the unfairness is likely to be similar between classifications and training data.

For German Credit Data, the obtained results with SPD suggest that models trained with `undersampling-multivariate` or `oversampling-multivariate` are likely to make fairer classifications than the complete training set. Results with NPI, on the other hand, suggest that models trained with these sampling methods are likely to increase it, except for some exceptions with `undersampling-multivariate`.

The boxplots in Figure 5.6 represent the distributions of the SPD Ratio for the COMPAS dataset. For this dataset, applying any of the tested sampling methods leads to models whose classifications are more unfair than the data used to train them. In fact, the same happens with the baseline scenario of `without-resampling`.

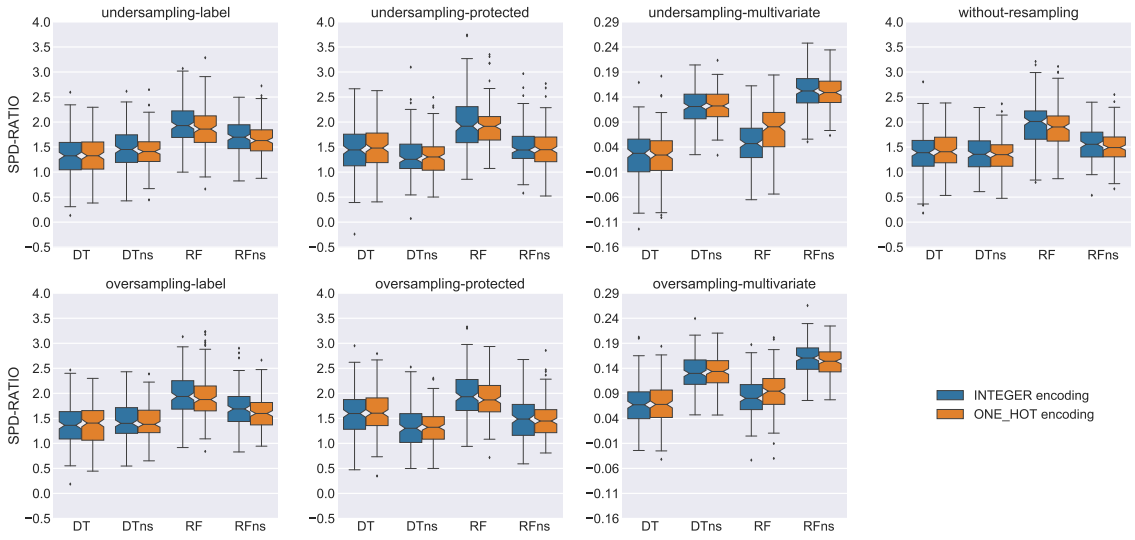


Figure 5.6: SPD Ratio for the COMPAS dataset.

The distributions of the NPI Ratio for COMPAS dataset are represented by the boxplots in Figure 5.7. Similar to the results with SPD Ratio, all tested configurations create models for which the unfairness in the classifications is higher than in the data used to train them. Apart from models trained with `undersampling-multivariate` and `oversampling-multivariate`, the same behaviour seems to emerge across the different configurations: models based on Random Forests show an higher increase in terms of unfairness when compared to models based on Decisions Trees. Moreover, this dataset has the lowest SPD and the lowest NPI in the training data after a sampling method is applied, and so, we should bear in mind that the SPD Ratio and the NPI Ratio only give a relative measurement of the unfairness, which in absolute terms might still be of a low order of magnitude.

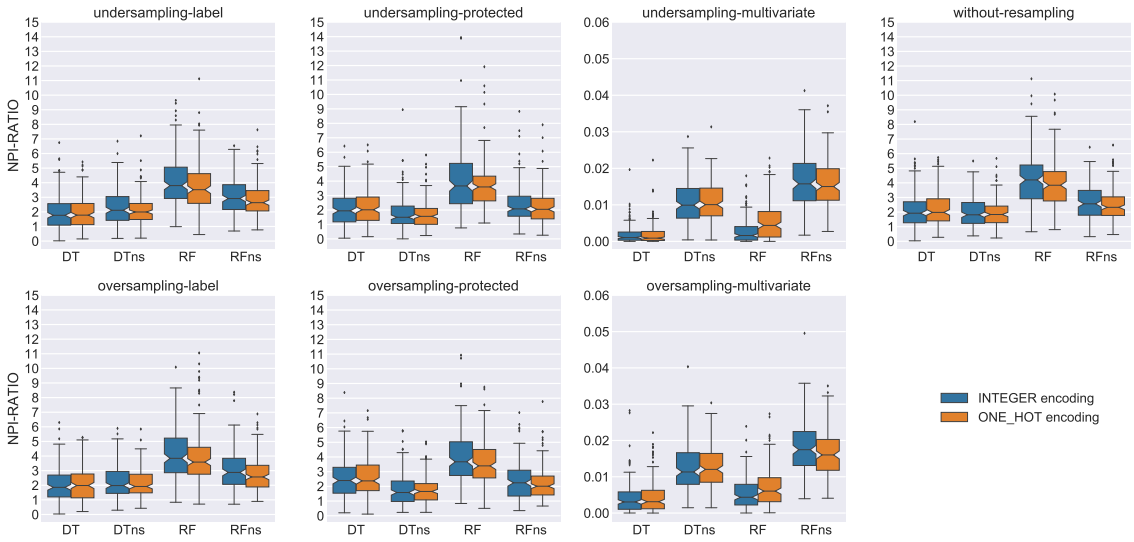


Figure 5.7: NPI Ratio for the COMPAS dataset.

Classifications made by DT and RF trained with `undersampling-multivariate` or `oversampling-multivariate` are likely to reduce the unfairness of the complete training data, while DTns and RFns are likely to increase it. For the COMPAS dataset, these models exhibit a somewhat unexpected behaviour, as discussed in Section 5.4.

5.4 Analysis of Fairness and Classification Performance

In this section, our analysis is focused on the trade-off between fairness and classification performance. As explained in Section 3.5, we use the F1-score and the Balanced Classification Rate (BCR) to measure classification performance, since using accuracy in imbalanced scenarios provides misleading estimates. We consider the additive inverse of SPD, NPI, and GE, this way ensuring that measurements closer to one are fairer under all fairness metrics that we consider in this analysis.

Figure 5.8 shows the average F1-score and fairness for the Adult Income dataset, grouped by sampling method and classifier. Caution must be taken when analysing the results with S-TNR-ratio: since all measurements are greater than one, fairer models are on the left side of the plot, while for most of the other plots fairer models are on the right side.

Performing random undersampling or random oversampling in the traditional way, i.e. with respect to the true labels (`undersampling-label` and `oversampling-label`, respectively),

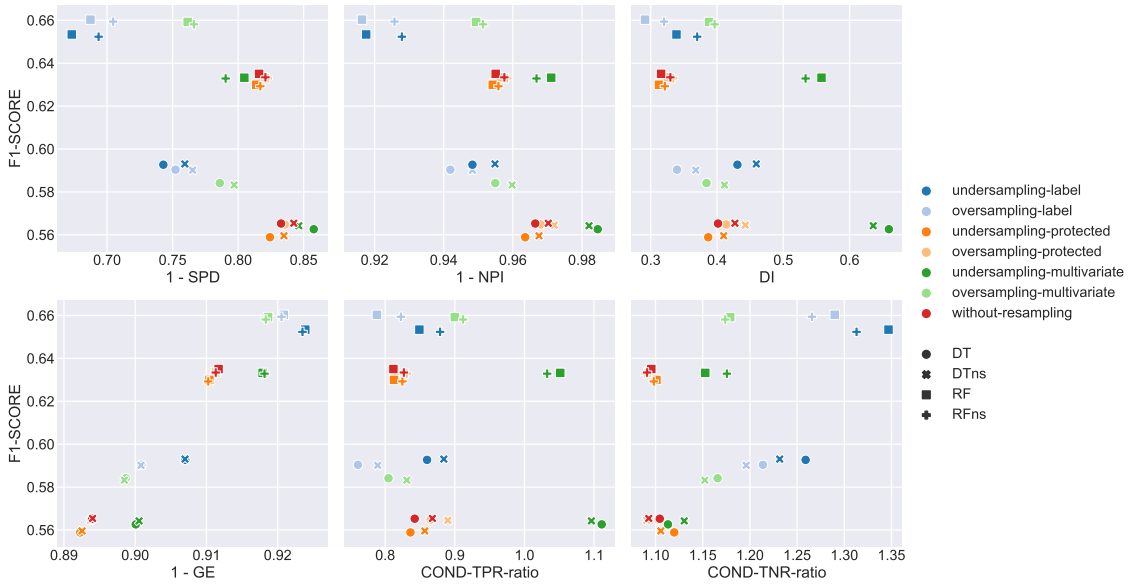


Figure 5.8: Trade-off between performance, given by the F1-score, and fairness for the Adult Income dataset. Fairer results are closer to 1.

and opting for Random Forests (RF or RFns) allows for the creation of models with the best F1-score from the tested configurations. However, these are also the models which consistently deliver the worst results regarding the fairness of the classifications, except when making this analysis based on GE. On the other side of the spectrum, i.e. with the worst performance but with fairer classifications, we usually have models based on Decision Trees (DT and DTns) trained with data selected by `undersampling-multivariate`.

The models which were trained with data sampled with respect to the sensitive attribute (`undersampling-protected` and `oversampling-protected`) or `without-resampling` exhibit a similar behaviour. When based on Random Forests, these models provide a better trade-off between performance and fairness in comparison to the models described above. This observation does not hold true when trying to improve fairness according to DI. However, none of the tested configurations make classifications which can be considered fair under the 80% rule. Models trained with any of these three sampling methods, together with those trained `undersampling-multivariate` have similar classification performance. Performance-wise, models trained with `undersampling-label`, `oversampling-label` and `oversampling-multivariate` can be grouped together.

The results with `undersampling-multivariate` and `oversampling-multivariate` should also be further analysed. Performing random oversampling with respect to both variables allows for models to achieve better F1-scores. However, when it comes to fairness, these models also seem to be more unfair than those trained with `undersampling-multivariate`. We recall that after applying any of these two sampling methods, the training data is perfectly fair under SPD, NPI and DI. The existence of indirect prejudice seems to be at the root of this behaviour, and the problem seems to be augmented when replicating instances from underrepresented groups. We hypothesise that this sampling procedure alters the distributions of the features that are associated with the sensitive attribute in such a way that the indirect prejudice in the training data is accentuated. These discrepancies are not so obvious when measuring fairness with GE, S-TPR-ratio or S-TNR-ratio. However, results with S-TPR-ratio also distance themselves from the other fairness metrics in the sense that RF and RFns trained with `undersampling-multivariate` and `oversampling-`

`multivariate` also seem to make fairer classifications than models trained with DT and DTns under the same conditions.

Figure 5.9 shows the average BCR and fairness for the Adult Income dataset, grouped by sampling method and classifier.

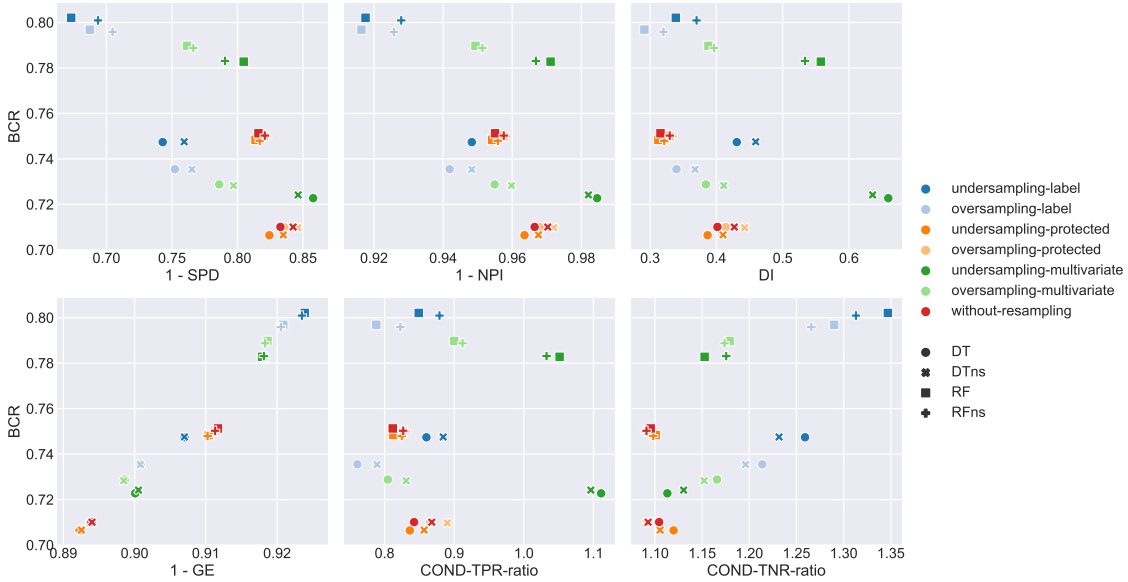


Figure 5.9: Trade-off between performance, given by the BCR, and fairness for the Adult Income dataset. Fairer results are closer to 1.

The main differences in comparison to the obtained results with F1-score are a clearer distinction between the classification performance of models trained with different sampling methods. We see that classification performance is mainly dictated by the variable with respect to which the sampling is performed, with more attenuated differences between random undersampling and random oversampling. Therefore, we have `undersampling-label` and `oversampling-label` as the best sampling methods, followed by `oversampling-multivariate` and `undersampling-label`. With worse classification performance we have `without-resampling` and, finally with the worst results, we have `undersampling-protected` and `oversampling-protected`.

Additionally, according to BCR, models resulting from the combination of DT with `undersampling-label` have a classification performance similar to those trained with `undersampling-protected`, `oversampling-protected`, or `without-resampling` and based on Random Forests. In fact, this last group of models is capable of making fairer predictions than the former, according to all fairness metrics except for DI and S-TPR-ratio.

For this dataset, the removal of the sensitive attribute usually has a minimal impact on the model's F1-score and BCR, as well as on the fairness of the classifications. We justify this by the emergence of indirect discrimination when this attribute is removed, as shall be later explained in this section.

Figure 5.10 shows the average F1-score and fairness for the German Credit Data dataset, grouped by sampling method and classifier.

We can detect some distinct trends with this dataset, in comparison to the obtained results for Adult Income. In this case, the worst performance is achieve when applying `undersampling-label` or `undersampling-multivariate`. This is probably due to the

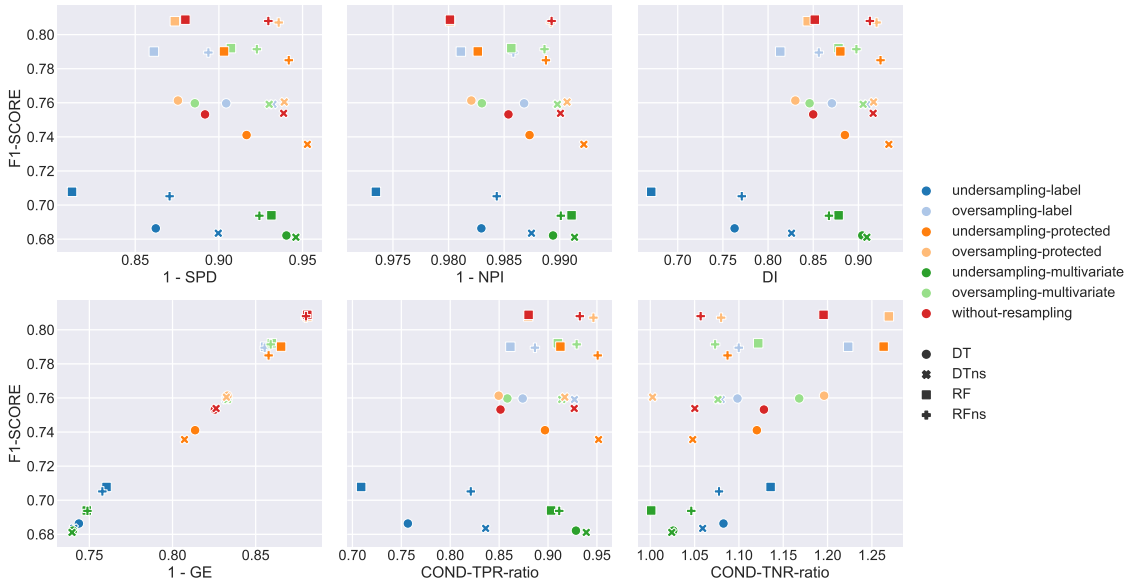


Figure 5.10: Trade-off between performance, given by the F1-score, and fairness for the German Credit Data dataset. Fairer results are closer to 1.

dataset's size and the limited number of training instances from the underrepresented groups. Thus, random oversampling naturally allows for the creation of models with better classification performance. The best sampling method in terms of classification performance also seems to be dependent on the classifier: `oversampling-protected` and `without-resampling` are better when training RF and RFns, while random oversampling seems more adequate when training DT and DTns.

Contrary to the Adult Income dataset, for which the sampling method and the learning algorithm were the key factors to improve fairness, the removal of the sensitive attribute seems to have a much more pivotal role when it comes to German Credit Data. Nevertheless, S-TNR-ratio suggests that it is better to choose models based on Decision Trees over models based on Random Forests. Moreover, we would like to point out a behaviour different from the norm when it comes to models trained with `oversampling-multivariate`. For most of the considered fairness metrics, models based on RF seem to be fairer than those based on DT.

Bearing the analysis from Chapter 4 in mind, we may find a possible explanation for the different impact of the removal of the sensitive attribute between the two datasets. In fact, some of the non-sensitive attributes of the Adult Income dataset are highly associated to the sensitive attribute. Although there are also some associations between non-sensitive and sensitive attributes in the German Credit Data dataset, these associations are much weaker. Thus, it is understandable that removing the sensitive attribute when training models for Adult Income has a lower impact on fairness, due to indirect discrimination in the data. Even if the sensitive attribute is removed, the models are still able to pick up the unfairness in the data from the remaining features used to train them.

Another difference from Adult Income is the fact that most models can be considered fair according to the 80% rule. The exceptions are DT, RF and RFns trained with `undersampling-label`.

Figure 5.11 shows the average BCR and fairness for the German Credit Data dataset, grouped by sampling method and classifier. An analysis based on this performance metric

yields completely different results than the one based on F1-score, but more in-line with what was reported for the Adult Income dataset. Models trained with `undersampling-label` and `oversampling-label` now have similar classification performances than those trained `without-resampling`, while models trained with `undersampling-protected` have the worst performance when compared to different sampling methods. Random Forests remain preferable to Decision Trees when it comes to better classifications from a performance point-of-view. For this reason, the trade-off between Decision Trees trained with `undersampling-protected` and Random Forests trained with `undersampling-label` emerges once again.

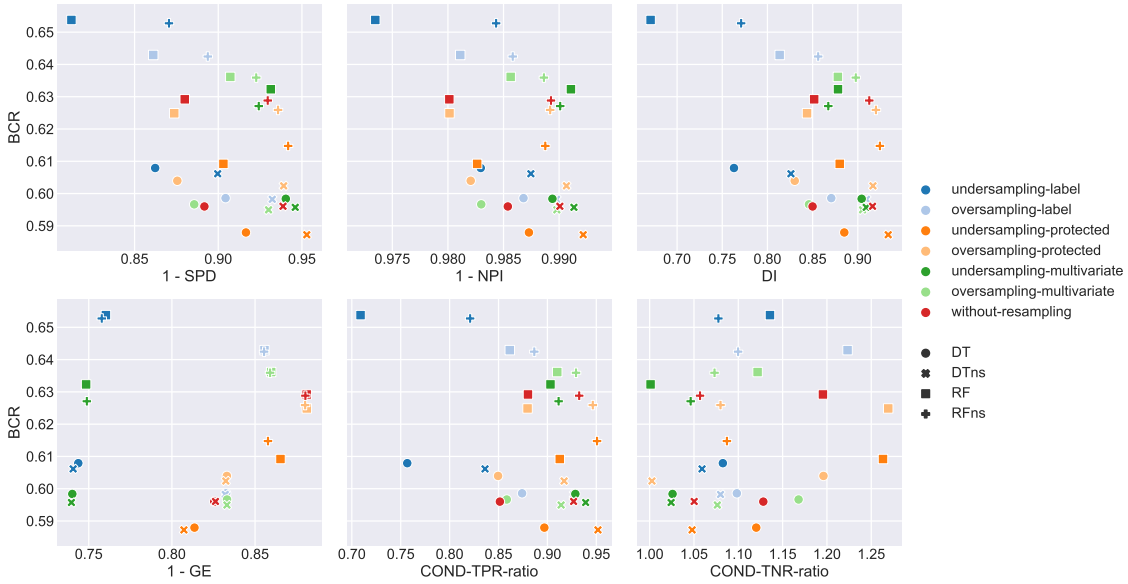


Figure 5.11: Trade-off between performance, given by the BCR, and fairness for the German Credit Data dataset. Fairer results are closer to 1.

Figure 5.12 shows the average F1-score and fairness for the COMPAS dataset, grouped by sampling method and classifier. Models based on Random Forests have better classification performance than those based on Decision Trees, as with the other two datasets. As far as the sampling method is concerned, the results with COMPAS are more similar to those of German Credit Data: within models based on the same learning algorithm, `without-resampling` together with `undersampling-protected` and `oversampling-protected` lead to the best results, followed by sampling with respect to the true labels (`undersampling-label` and `oversampling-label`). Models trained with `undersampling-multivariate` have the worst classification performance in comparison to the other sampling methods. Random oversampling is also preferable to random undersampling for this dataset, but sometimes leads to fairness losses, as in the case of `oversampling-multivariate`.

DT and RF trained with `undersampling-multivariate` and `oversampling-multivariate` produce the fairest classifications, being the only models that can be considered fair under the 80% rule. In fact, these models and RF and RFns trained with other sampling methods clearly distance themselves from the other models in opposite directions, but it seems harder to distinguish the other models from one another from a fairness point-of-view. When trained without the sensitive attribute and with `undersampling-protected`, `oversampling-protected` or `without-resampling`, the ML models all seem to lie close to this legal threshold of fairness. Surprisingly, most fairness metrics suggest that DTns and RFns trained with the same sampling methods produce classifications which are more

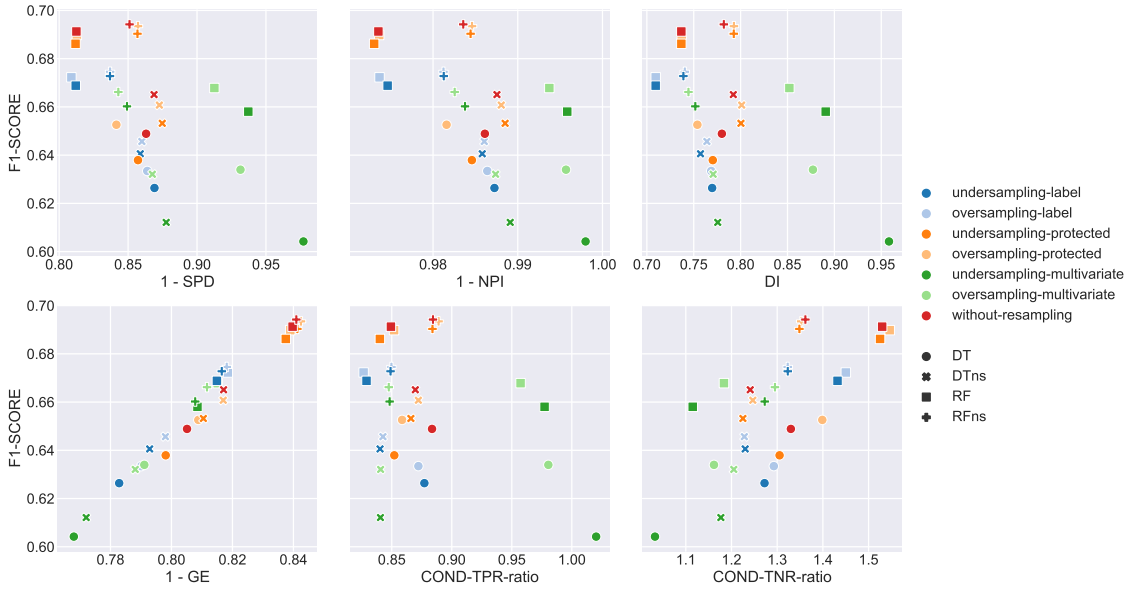


Figure 5.12: Trade-off between performance, given by the F1-score, and fairness for the COMPAS dataset. Fairer results are closer to 1.

unfair than when the same learning algorithm is trained with the sensitive attribute.

The impact of the removal of the sensitive attribute is more evident when analysing fairness with NPI or S-TNR-ratio, which may also be an indication that the problems posed by this dataset are somewhat different from those found in the remaining datasets. In fact, it seems that the fairness concerns are more connected to the over-representation of unprivileged individuals from the negative class, which may justify why the removal of the sensitive attribute has more visible effects when measuring fairness with S-TNR-ratio.

Figure 5.13 shows the average BCR and fairness for the COMPAS dataset, grouped by sampling method and classifier. These results highlight that differences in performance are mainly due to the underlying learning algorithm, with models based on Random Forests achieving higher classification performance than the ones based Decision Trees. It is hard to make a distinction between models solely based on the sampling method, which is understandable since this dataset can be considered balanced with respect to the true labels.

In Section 5.2, we had the opportunity to discuss the obtained results with S-TNR-ratio. It is also interesting to notice the relationship between fairness and performance when measuring fairness with GE. Contrary to the other fairness metrics, according to which classifiers with better classification performance tend to produce more unfair classifications, GE suggests that fairer models are also those with better performance. In fact, for all datasets, it is almost possible to draw a linear regression between fairness and any of the performance metrics used in the analysis. We recall that the authors who proposed using this metric to measure fairness took inspiration from inequality indices which are widely used in economics [Spe+18]. The metric is highly dependent on the underlying benefit function, and so, these results leave room for discussion on whether this is the way to go when it comes to measuring (un)fairness at an individual level. The similarity between individuals seems to be mainly dictated by the true labels, with the remaining features being, to a certain extent, ignored. However, we believe that the remaining features of each individual should be given more weight when determining "neighbour" instances.

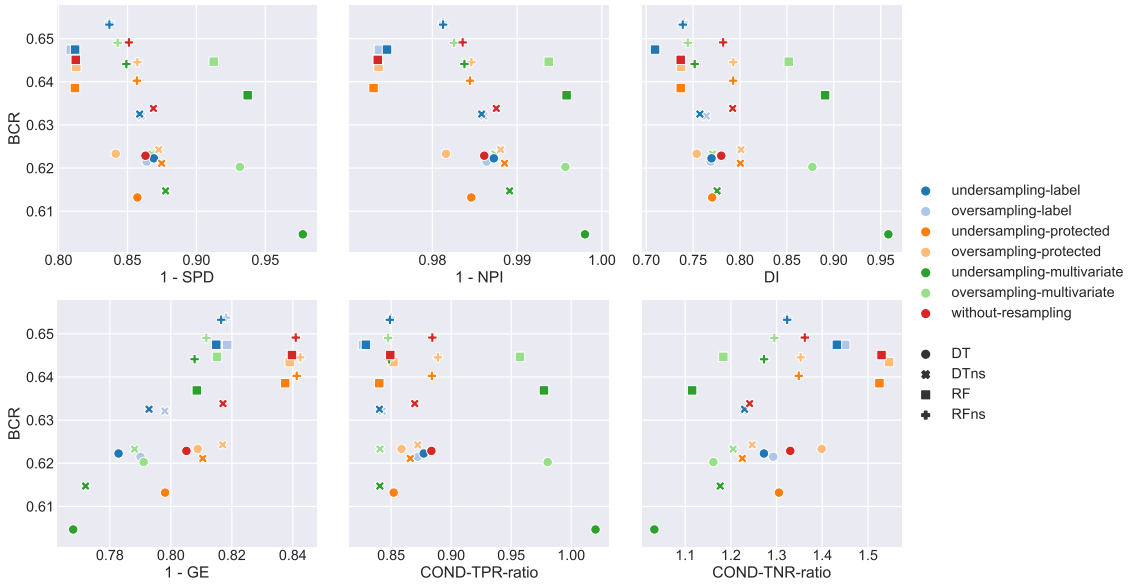


Figure 5.13: Trade-off between performance, given by the BCR, and fairness for the COMPAS dataset. Fairer results are closer to 1.

5.5 Conclusion

The removal of the sensitive attribute from the training data, the aspect under evaluation in **RQ3**, seems to have low impact on classification performance, and so, one needs not to jeopardise classification performance in order to improve fairness. Nevertheless, we warn that its impact on fairness depends on the characteristics of the dataset. In scenarios in which the non-sensitive attributes are highly associated with the sensitive attribute, the removal of the sensitive attribute may not be enough to create the necessary conditions for fairness improvements. This was mainly observed with the Adult Income dataset and is hypothesised to be caused by indirect discrimination, with models being capable of exploring the association with the sensitive attribute even when not having direct access to it.

We also observed some exceptions when performing random undersampling with respect to the both the true labels and the sensitive attribute (`undersampling-multivariate`), where removing the sensitive attribute prior to training the models lead to more unfair classifications. This is somewhat counter-intuitive in two aspects: not only are we removing the unfairness in the training data when we perform `undersampling-multivariate`, but we are also removing the main source of that unfairness, the sensitive attribute itself. This behaviour may be, one again, linked to indirect discrimination. This indirect prejudice might become more apparent after the removal of the sensitive attribute, an hypothesis that requires further investigation by looking at the structure of the resulting trees. Bearing all these considerations in mind, we still recommend that the sensitive be removed from the training data, not only because it is likely to have a minimal impact on classification performance, but also because the adoption of this legal procedure allows organisations to avoid incurring in direct discrimination.

As far as the encoding of categorical features is concerned, we found that the results vary greatly, being difficult to find a pattern that would hold true for all datasets or all fairness metrics. Nevertheless, a careful evaluation of its impact on the fairness of a system should be performed before some ML model is deployed into production, as different datasets

may behave differently depending on the representation of the categorical features. It is also worth mentioning that not all implementations of Decision Trees and Random Forests found in widely used ML frameworks can support categorical features. Thus, it is important to carefully understand the characteristics of the available data before using such frameworks in real-world scenarios. In sum, the results were inconclusive regarding **RQ1**, but this parameter is likely to have impact on the fairness of an ML model.

The choice of a sampling method seems to have a pivotal role on the fairness of a ML model (**RQ2**). In general, we can say that `undersampling-multivariate`, `oversampling-protected` and `undersampling-protected` seem to be the best options if one's aim is to build fairer models. Nevertheless, the obtained results also suggest that even if models are trained with completely fair data, as is the case when performing `undersampling-multivariate` and `oversampling-multivariate`, the classifiers are still able to explore the inherent unfairness of the original dataset, sometimes making classifications with a higher degree of unfairness than the complete set of training data. We attribute this behaviour to changes in the distributions of the non-sensitive features, especially of those associated with the sensitive attribute.

Some of these observations partially answer **RQ4**, although the relationship between the unfairness in the training data and in the classifications is quite dependent on the dataset. This observation is in line with what one would expect, since the fairness in the training data is mainly dictated by the distributions of the training instances with respect to the true labels and the sensitive attribute. With the COMPAS dataset, for instance, the one with the fairest training data among the three used in our experiments, all models seem to increase the unfairness in the data used to train them, which may result in damaging consequences for African-Americans.

In general, models based on Decisions Trees produce fairer classifications than those based on Random Forests, which means that model complexity may be a problem for fairness and needs to be further investigated. A more in-depth analysis of the resulting trees could allow for a better understanding of this behaviour, but we believe it may be due to the randomisation introduced by Random Forests during splitting. However, using Random Forests instead of Decisions Trees seems to be the decisive factor to achieve a better classification performance, which aligns with the expected behaviour of these algorithms.

These results are clear examples of the trade-offs an organisation may face when designing an ML model to be deployed into production. If classification performance is prioritised and fairness is completely neglected, such organisations might choose to build Random Forests while performing random undersampling in the traditional way, i.e. with respect to the true labels (`undersampling-label`). On the other hand, opting for Decision Trees and `undersampling-multivariate` might penalise classification performance in favour of fairer models. The behaviour resulting from the application of `undersampling-multivariate` appears to be quite unstable, but its negative impact on classification performance might be justified by the more drastic reduction in the number of instances used to train the models.

However, conclusions about the relative fairness of different models is dependent on the fairness metric used in the analysis. These discrepancies should be taken into consideration when choosing the metric, always bearing in mind its relation to the legal framework of the scenario in which the models are to be used.

Based on our observations, the best encoding of categorical attributes is data-dependent and different possibilities should be evaluated instead of choosing an encoding *a priori*. We would suggest opting for Decision Trees and for following the standard procedure of re-

moving the sensitive attribute to build fairer models. Even though performing multivariate random undersampling can lead to satisfactory fairness results, it might have a significant impact on the models' performance. Furthermore, combining `undersampling-label` with Random Forests should be avoided since other configurations are likely to offer a better trade-off between classification performance and fairness.

Caution must be taken with the choice of classification performance metric, especially when dealing with imbalanced datasets. When choosing this metric, one should also bear in mind that a false negative (for instance, some person being incorrectly classified as bad credit risk) is sometimes more costly than a false positive. When fairness concerns are also being considered, one should analyse not only the class imbalance with respect to the true labels, as typically done, but also the disproportion between privileged and unprivileged groups.

In fact, there seems to be room for improvement and progress when it comes to the definition of new fairness metrics specially when dealing with imbalanced scenarios. The results with the COMPAS dataset show precisely that, with S-TNR-ratio being one of the few metrics that give a bit more insight on how the model is making classifications between the privileged and the unprivileged group. We would also like to emphasise that when dealing with data imbalance, it is very unlikely to find a dataset with optimal NPI, since this metric is quite sensitive to small changes in class distributions. The sensitivity of this metric is also exacerbated by the emergence of extreme outliers, mainly in the obtained results for the smaller dataset.

Although being aware of the drawbacks of fairness metrics like statistical parity, as these have been widely discussed in the literature [HPS16; Bin18], we wanted to perform a not so common analysis of fairness that allowed us to compare the unfairness found in the classifications made by a ML model to that found in the data used to train such a model. Nevertheless, our experiments have shown the brittleness of these metrics, as even metrics which were expected to show similar behaviours, such as SPD and DI [Fri+19], sometimes presenting contradictory results.

It is also interesting to notice that an analysis based on DI, more precisely on the 80% rule, and a consequent decision on whether or not to consider a model trained under certain configurations to be fair is highly dependent on the dataset. We can illustrate this by comparing the results between the three datasets used in our experiments: for Adult Income, no configuration allowed for the creation of a fair model, while for German Credit Data most of the tested configurations originated fair models.

There is also room for improvement when it comes to the definition of individual fairness metrics. The challenge here is in finding a suitable measure of the similarity between individuals [Dwo+12]. It remains unclear if metrics like GE, based on benefit functions, solve the problem.

Chapter 6

Conclusion and Future Work

Systems based on Machine Learning (ML) are being used in scenarios that directly affect people, from loan approvals to hiring decisions, as well as criminal risk assessment and predictive policing. With this widespread usage of ML models, concerns arise that their outputs may be supporting decisions that result in the unfair treatment of individuals based on attributes like sex, race, age, or nationality.

In this context, it is of uttermost importance to assess the fairness of such systems. In this work, we focused on evaluating the impact that procedures applied to a dataset before being used to train an ML model potentially have on fairness. Fairness was measured both at the data-level and when the classifications made by the models were known, in this way allowing us to understand which procedures had the most influence on the obtained results.

We found that the method applied to select the training instances is one of the factors that influences the final fairness of the classifications the most. Also, removing the sensitive attribute, a standard legal practice, might not always lead to the expected improvement on the models' fairness, especially in cases where the non-sensitive features are highly associated with the sensitive attribute. Furthermore, models based on Random Forests make, in general, more unfair classifications than Decision Trees.

Based on our observations, we recommend a careful analysis of the dataset's characteristics before training an ML model intended to be deployed into production. Moreover, traditional practices, like applying a sampling method with respect to the true labels, may lead to undesired effects, especially if the data also shows an imbalance with respect to the sensitive attribute.

In real-world scenarios some business objectives still need to be met, and so, it may not be possible to choose the model which delivers the fairest classifications. However, organisations should evaluate the trade-off between the fairness and classification performance of the models, as there are usually better options that do not completely disregard fairness concerns in order to gain performance.

We argue that the procedures applied to a dataset during the data preparation and preprocessing phases of an ML pipeline are often neglected when making a fairness evaluation. Taking these procedures into consideration is of pivotal importance, especially because our findings suggest that standard procedures widely adopted in the field may have undesirable effects when fairness concerns are at stake.

Moreover, different datasets pose different fairness concerns, and so, what applies in a scenario may not hold true when the data does not have the same characteristics. We faced these challenges with the COMPAS dataset from the beginning. For instance, the imbalance in this dataset is mainly in the negative class, with the unprivileged group being over-represented. In the other two datasets, however, there is an under-representation of the unprivileged group in the positive classifications.

Our work constitutes a step forward in the development of a framework that enables the systematic analysis of the fairness of systems based on Machine Learning, allowing for the comparison of the obtained results at different stages of a typical Machine Learning cycle taking several fairness metrics, and to some degree their connection to current legislation, into consideration.

Future Work

There are several factors that influence the fairness of the classifications made by an ML model. We only focused on some of these factors, but a general recommendation, which aligns with the results reported in [Fri+19], is that all the procedures applied to a dataset before being used to train the model must be clearly specified.

As future work, we plan on considering numerical features besides the categorical features we studied in our experiments. By doing so, we broaden the range of sampling methods that can be tested under the proposed methodology. It would be interesting to investigate how the creation of synthetic instances by methods like SMOTE might impact the fairness of the training data and, consequently, the fairness of the classifications made by a model trained with that new synthetic instances.

We also intend to perform a more in-depth analysis of the consequences of having features highly associated with the sensitive attribute in the dataset. In fact, we are making a preliminary study of the effect of indirect discrimination on the structure of the learned models, especially when the sensitive attribute is removed from the training data.

After analysing the structure of tree-based methods, an important step would also be to consider more complex learning algorithms, namely Artificial Neural Network (ANN), always bearing in mind the associations between non-sensitive and sensitive attributes.

The fact that datasets that pose fairness concerns are often imbalanced with respect to privileged and unprivileged groups demands that fairness metrics should take this into consideration so as to deliver good estimates of the actual fairness in the data or in the classifications. The ability to compare different configurations and scenarios using these metrics should also be further investigated, in order to advance in terms of benchmarks of the fairness of different approaches proposed in the literature.

Bibliography

[AA17] ACM U.S. Public Policy Council and ACM Europe Policy Committee. *Joint Statement on Algorithmic Transparency and Accountability by USACM and EUACM*. 25th May 2017. URL: https://www.acm.org/articles/bulletins/2017/january/usacm-statement-algorithmic-accountability/.

[Ali+] Muhammad Ali et al. *Discrimination through optimization: How Facebook's ad delivery can lead to skewed outcomes*. arXiv: 1904.02095.

[Ang+16] Julia Angwin et al. *Machine Bias*. ProPublica. 23rd May 2016. URL: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

[Ant+18] Nuno Antunes et al. 'Fairness and Transparency of Machine Learning for Trustworthy Cloud Services'. In: *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. Luxembourg: IEEE, 2018, pp. 188–193.

[BPM04] Gustavo E. A. P. A. Batista, Ronaldo C. Prati and Maria Carolina Monard. 'A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data'. In: *SIGKDD Explorations Newsletter* 6.1 (June 2004), pp. 20–29. ISSN: 1931-0145. DOI: 10.1145/1007730.1007735.

[Bel+] Rachel K. E. Bellamy et al. *AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias*. arXiv: 1810.01943.

[BTI19] Katie Benner, Glenn Thrush and Mike Isaac. *Facebook Engages in Housing Discrimination With Its Ad Practices, U.S. Says*. The New York Times. 28th Mar. 2019. URL: https://www.nytimes.com/2019/03/28/us/politics/facebook-housing-discrimination.html.

[Bin18] Reuben Binns. 'Fairness in Machine Learning: Lessons from Political Philosophy'. In: *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*. Vol. 81. New York, NY, USA: PMLR, 2018, pp. 149–159. URL: http://proceedings.mlr.press/v81/binns18a.html.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.

[CV10] Toon Calders and Sicco Verwer. 'Three naive Bayes approaches for discrimination-free classification'. In: *Data Mining and Knowledge Discovery* 21.2 (Sept. 2010), pp. 277–292. ISSN: 1573-756X. DOI: 10.1007/s10618-010-0190-x.

[Cel+] L. Elisa Celis et al. *Classification with Fairness Constraints: A Meta-Algorithm with Provable Guarantees*. arXiv: 1806.06055.

[CJK04] Nitesh V. Chawla, Nathalie Japkowicz and Aleksander Kotcz. 'Editorial: Special Issue on Learning from Imbalanced Data Sets'. In: *SIGKDD Explorations Newsletter* 6.1 (June 2004), pp. 1–6. ISSN: 1931-0145. DOI: 10.1145/1007730.1007733.

[Cha+02] Nitesh V. Chawla et al. 'SMOTE: Synthetic Minority Over-sampling TEchnique'. In: *Journal of Artificial Intelligence Research* 16.1 (June 2002), pp. 321–357. ISSN: 1076-9757. URL: http://dl.acm.org/citation.cfm?id=1622407.1622416.

[Dau12] Hal Daumé III. 'A course in machine learning'. In: *Publisher, ciml. info* (2012), pp. 5–73.

[DG19] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*. 2019. URL: http://archive.ics.uci.edu/ml.

[Dwo+12] Cynthia Dwork et al. 'Fairness through awareness'. In: *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10*. 2012, pp. 214–226. DOI: 10.1145/2090236.2090255.

[Eur16] European Parliament. *General Data Protection Regulation*. 2016. URL: http://data.europa.eu/eli/reg/2016/679/oj/eng.

[Faw06] Tom Fawcett. 'An introduction to ROC analysis'. In: *Pattern Recognition Letters* 27.8 (2006), pp. 861–874. DOI: 10.1016/j.patrec.2005.10.010.

[Fel+15] Michael Feldman et al. 'Certifying and Removing Disparate Impact'. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13*. 2015, pp. 259–268. DOI: 10.1145/2783258.2783311.

[FKL15] Benjamin Fish, Jeremy Kun and Ádám Dániel Lelkes. 'Fair Boosting : a Case Study'. In: *Workshop on Fairness, Accountability, and Transparency in Machine Learning (FATML)*. 2015.

[Fri+19] Sorelle A. Friedler et al. 'A Comparative Study of Fairness-enhancing Interventions in Machine Learning'. In: *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19), Atlanta, GA, USA*. ACM, 2019, pp. 329–338. ISBN: 978-1-4503-6125-5. DOI: 10.1145/3287560.3287589.

[Goo+14] Ian J. Goodfellow et al. 'Generative Adversarial Nets'. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, Montreal, Quebec, Canada, December 8-13*. 2014, pp. 2672–2680. URL: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[Hai+08] Haibo He et al. 'ADASYN: Adaptive synthetic sampling approach for imbalanced learning'. In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. June 2008, pp. 1322–1328. DOI: 10.1109/IJCNN.2008.4633969.

[HBC16] Sara Hajian, Francesco Bonchi and Carlos Castillo. 'Algorithmic Bias: From Discrimination Discovery to Fairness-aware Data Mining'. In: *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. San Francisco, California, USA, 2016, pp. 2125–2126. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2945386.

[HPS16] Moritz Hardt, Eric Price and Nati Srebro. 'Equality of Opportunity in Supervised Learning'. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, Barcelona, Spain, December 5-10*. 2016, pp. 3315–3323. URL: http://papers.nips.cc/paper/6374-equality-of-opportunity-in-supervised-learning.

[HTF09] Trevor Hastie, Robert Tibshirani and Jerome H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd Edition. Springer, New York, NY, 2009. ISBN: 978-0-387-84857-0. DOI: 10.1007/978-0-387-84858-7.

[HUD19] HUD Public Affairs. *HUD charges Facebook with Housing Discrimination over Company's Targeted Advertising Practices*. The U.S. Department of Housing and Urban Development. 28th Mar. 2019. URL: https://www.hud.gov/press/press_releases_media_advisories/HUD_No_19_035.

[Jam+13a] Gareth James et al. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[Jam+13b] Gareth James et al. *An Introduction to Statistical Learning: With Applications in R*. Springer, New York, NY, 2013. ISBN: 978-1-4614-7137-0. DOI: 10.1007/978-1-4614-7138-7.

[KC09] F. Kamiran and T. Calders. 'Classifying without discriminating'. In: *2009 2nd International Conference on Computer, Control and Communication*. Feb. 2009, pp. 1–6. DOI: 10.1109/IC4.2009.4909197.

[KC12] Faisal Kamiran and Toon Calders. 'Data preprocessing techniques for classification without discrimination'. In: *Knowledge and Information Systems* 33.1 (Oct. 2012), pp. 1–33. ISSN: 0219-3116. DOI: 10.1007/s10115-011-0463-8.

[KCP10] Faisal Kamiran, Toon Calders and Mykola Pechenizkiy. 'Discrimination Aware Decision Tree Learning'. In: *2010 IEEE International Conference on Data Mining*. Dec. 2010, pp. 869–874. DOI: 10.1109/ICDM.2010.50.

[Kam+12] Toshihiro Kamishima et al. 'Fairness-Aware Classifier with Prejudice Remover Regularizer'. In: *Machine Learning and Knowledge Discovery in Databases*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 35–50. ISBN: 978-3-642-33486-3.

[Lar+16] Jeff Larson et al. *How We Analyzed the COMPAS Recidivism Algorithm*. ProPublica. 23rd May 2016. URL: https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm.

[LNA17] Guillaume Lemaître, Fernando Nogueira and Christos K. Aridas. 'Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning'. In: *Journal of Machine Learning Research* 18.17 (2017), pp. 1–5. URL: http://jmlr.org/papers/v18/16-365.html.

[LR02] Roderick J A Little and Donald B Rubin. *Statistical Analysis with Missing Data*. 2nd. John Wiley & Sons, Inc., 2002. ISBN: 978-0-471-18386-0.

[LJ] Kristian Lum and James Johndrow. *A statistical framework for fair predictive algorithms*. arXiv: 1610.08077 [stat.ML].

[Mar14] Stephen Marsland. *Machine Learning: An Algorithmic Perspective, Second Edition*. Chapman & Hall/CRC, 2014. ISBN: 9781466583283.

[MSP16] Cecilia Muñoz, Megan Smith and Dj Patil. *Big Data: A Report on Algorithmic Systems, Opportunity, and Civil Rights*. United States – White House Office, May 2016. URL: https://www.hsdl.org/?abstract%5C&did=792977.

[Mur12] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012. ISBN: 9780262018029.

[Ped+11] F. Pedregosa et al. 'Scikit-learn: Machine Learning in Python'. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[PRT08] Dino Pedreshi, Salvatore Ruggieri and Franco Turini. 'Discrimination-aware Data Mining'. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Las Vegas, Nevada, USA, 2008, pp. 560–568. ISBN: 978-1-60558-193-4. DOI: 10.1145/1401890.1401959.

[Pow11] David Martin Powers. 'Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation'. In: *Journal of Machine Learning Technologies* 2.1 (2011), pp. 37–63.

[Pro] ProPublica. *Data and Analysis for Machine Bias - GitHub repository*. URL: https://github.com/propublica/compas-analysis.

[RSM18] Edward Raff, Jared Sylvester and Steven Mills. 'Fair Forests: Regularized Tree Induction to Minimize Model Bias'. In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society (AIES'18)*. New Orleans, LA, USA, 2018, pp. 243–250. ISBN: 978-1-4503-6012-8. DOI: 10.1145/3278721.3278742.

[Rep] Assembleia da República. *Lei n.º 7/2009 de 12 de Janeiro. Diário da República n.º 30/2009, Série I*. Lisboa.

[RR14] Andrea Romei and Salvatore Ruggieri. 'A multidisciplinary survey on discrimination analysis'. In: *The Knowledge Engineering Review* 29.5 (2014), pp. 582–638. DOI: 10.1017/S0269888913000039.

[SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. New York, NY, USA: Cambridge University Press, 2014. DOI: 10.1017/CBO9781107298019.

[Sky] Skymind. *Artificial Intelligence (AI) vs. Machine Learning vs. Deep Learning*. URL: https://skymind.ai/wiki/ai-vs-machine-learning-vs-deep-learning (visited on 28/06/2019).

[Spe+18] Till Speicher et al. 'A Unified Approach to Quantifying Algorithmic Unfairness: Measuring Individual & Group Unfairness via Inequality Indices'. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 2239–2248. ISBN: 978-1-4503-5552-0. DOI: 10.1145/3219819.3220046.

[SB18] R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018. ISBN: 9780262039246.

[Uni] United States, Equal Employment Opportunity Commission. *Uniform Guidelines on Employment Selection Procedures, 29 C.F.R. § 1607.4(D) (2018)*.

[WVP] Christina Wadsworth, Francesca Vera and Chris Piech. *Achieving Fairness through Adversarial Learning: an Application to Recidivism Prediction*. arXiv: 1807.00199.

[Xu+] Depeng Xu et al. *FairGAN: Fairness-aware Generative Adversarial Networks*. arXiv: 1805.11202.

[YS17] Ke Yang and Julia Stoyanovich. 'Measuring Fairness in Ranked Outputs'. In: *Proceedings of the 29th International Conference on Scientific and Statistical Database Management (SSDBM '17)*. Chicago, IL, USA, 2017, 22:1–22:6. ISBN: 978-1-4503-5282-6. DOI: 10.1145/3085504.3085526.

[Zaf+17a] Muhammad Bilal Zafar et al. 'Fairness Beyond Disparate Treatment & Disparate Impact: Learning Classification without Disparate Mistreatment'. In: *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7*. 2017, pp. 1171–1180. DOI: 10.1145/3038912.3052660.

[Zaf+17b] Muhammad Bilal Zafar et al. 'Fairness Constraints: Mechanisms for Fair Classification'. In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April, Fort Lauderdale, FL, USA*. 2017, pp. 962–970. URL: http://proceedings.mlr.press/v54/zafar17a.html.

[Zem+13] Richard S. Zemel et al. 'Learning Fair Representations'. In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June*. 2013, pp. 325–333. URL: http://jmlr.org/proceedings/papers/v28/zemel13.html.

[ZLM18] Brian Hu Zhang, Blake Lemoine and Margaret Mitchell. 'Mitigating Unwanted Biases with Adversarial Learning'. In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society (AIES'18)*. New Orleans, LA, USA, 2018, pp. 335–340. ISBN: 978-1-4503-6012-8. DOI: 10.1145/3278721.3278779.

This page is intentionally left blank.

Appendices

This page is intentionally left blank.

Appendix A

Pairwise Mutual Information and Cramer's V

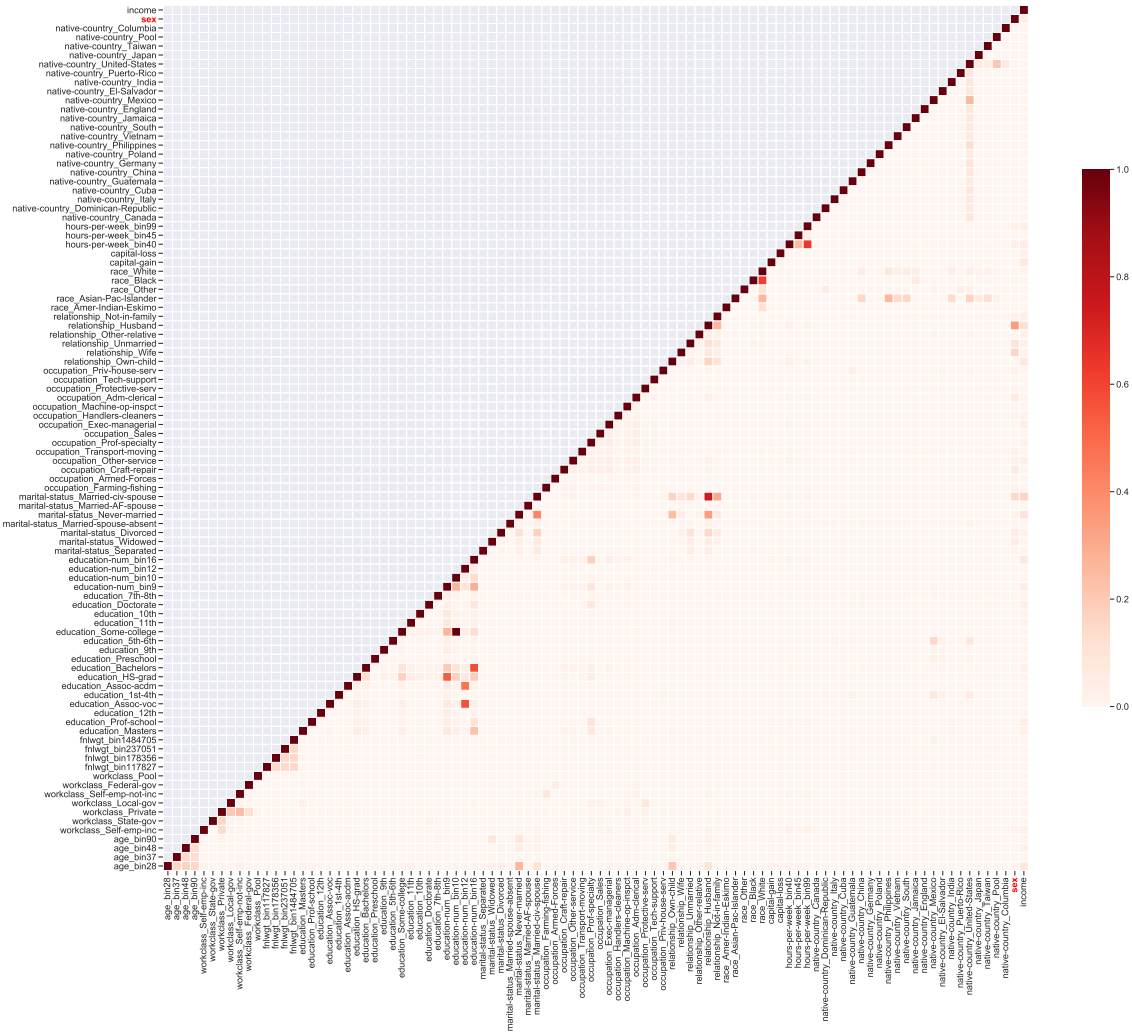


Figure A.1: One-hot encoded version of Adult Income - normalised mutual information.

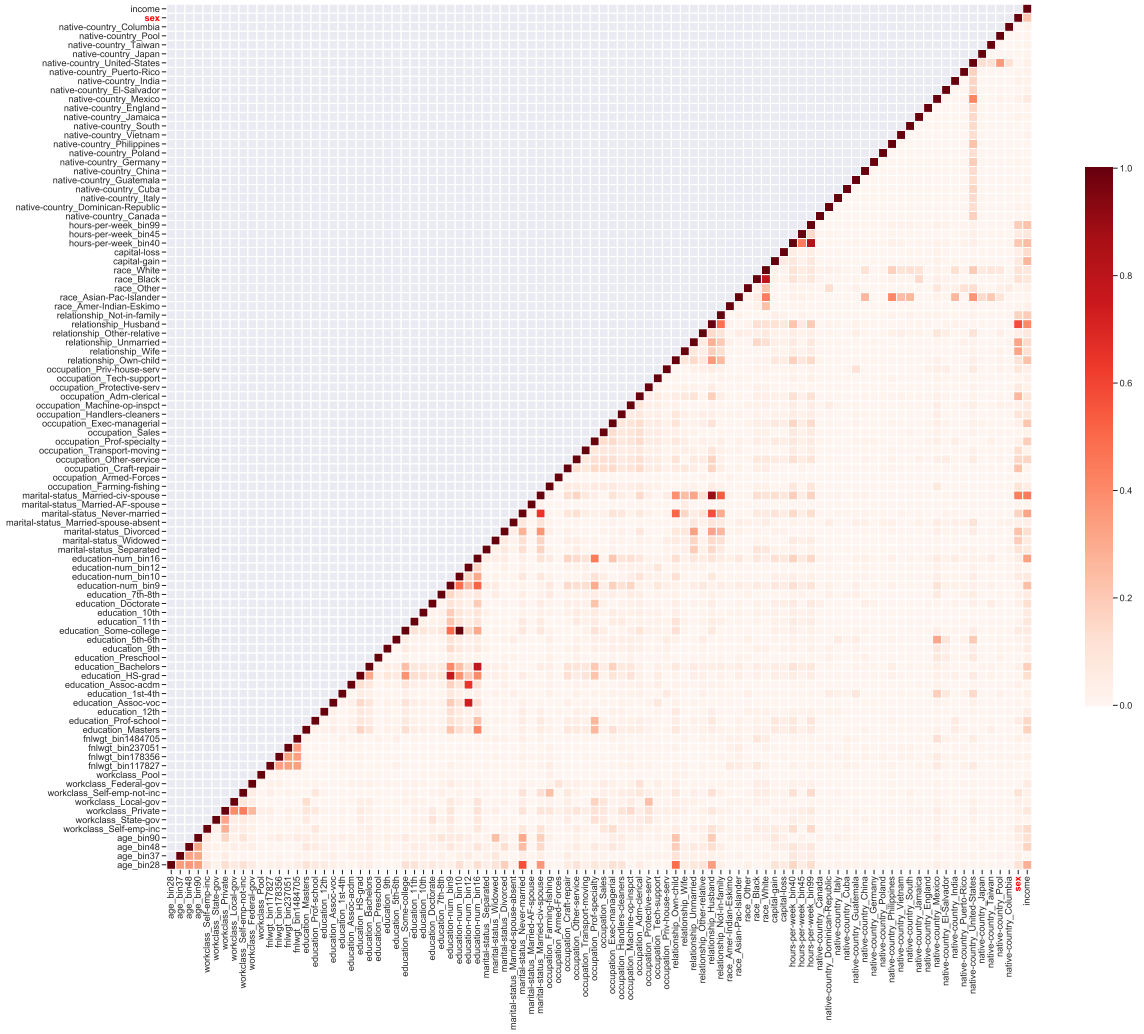


Figure A.2: One-hot encoded version of the Adult Income dataset - Cramer's V.

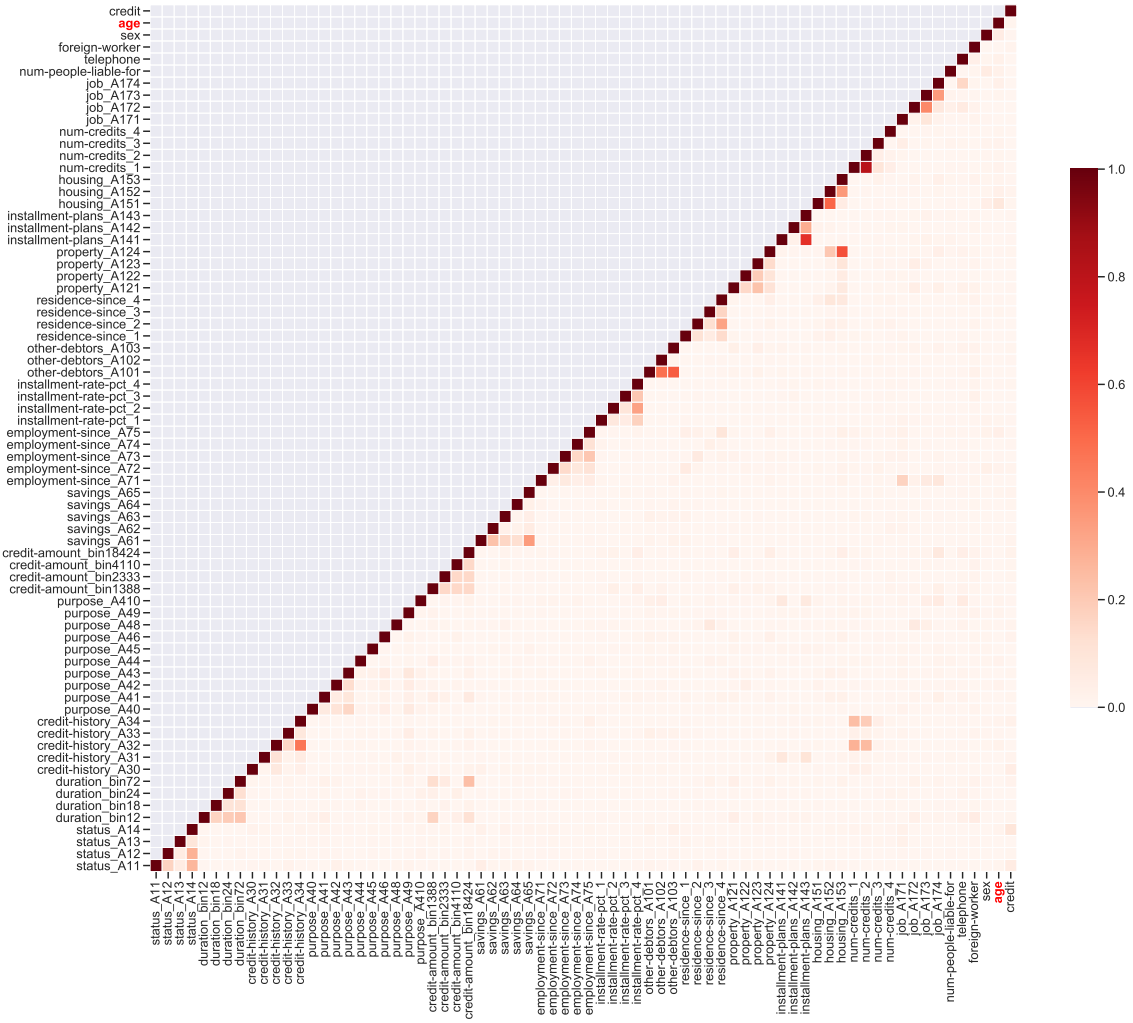


Figure A.3: One-hot encoded version of German Credit - normalised mutual information.

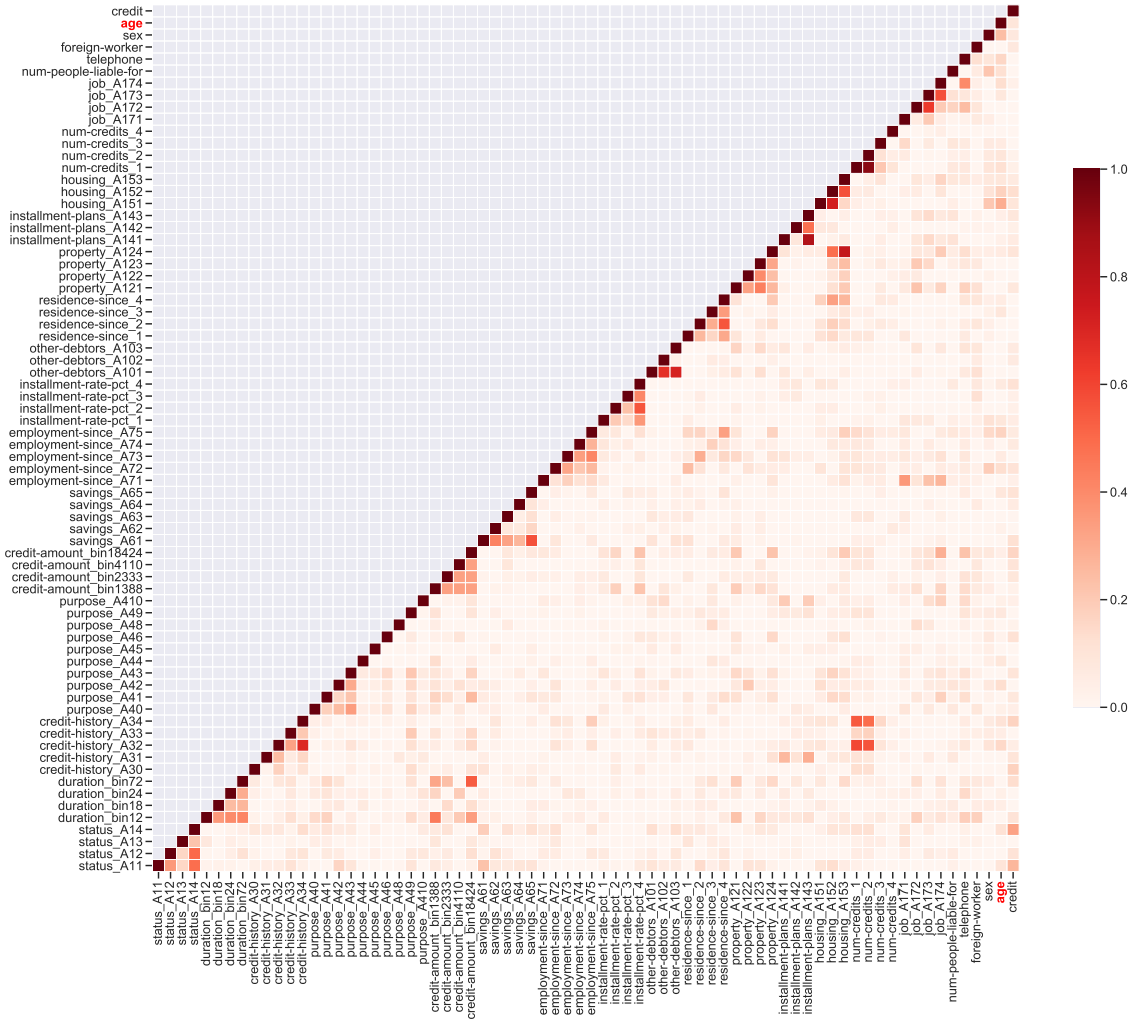


Figure A.4: One-hot encoded version of German Credit - Cramer's V.

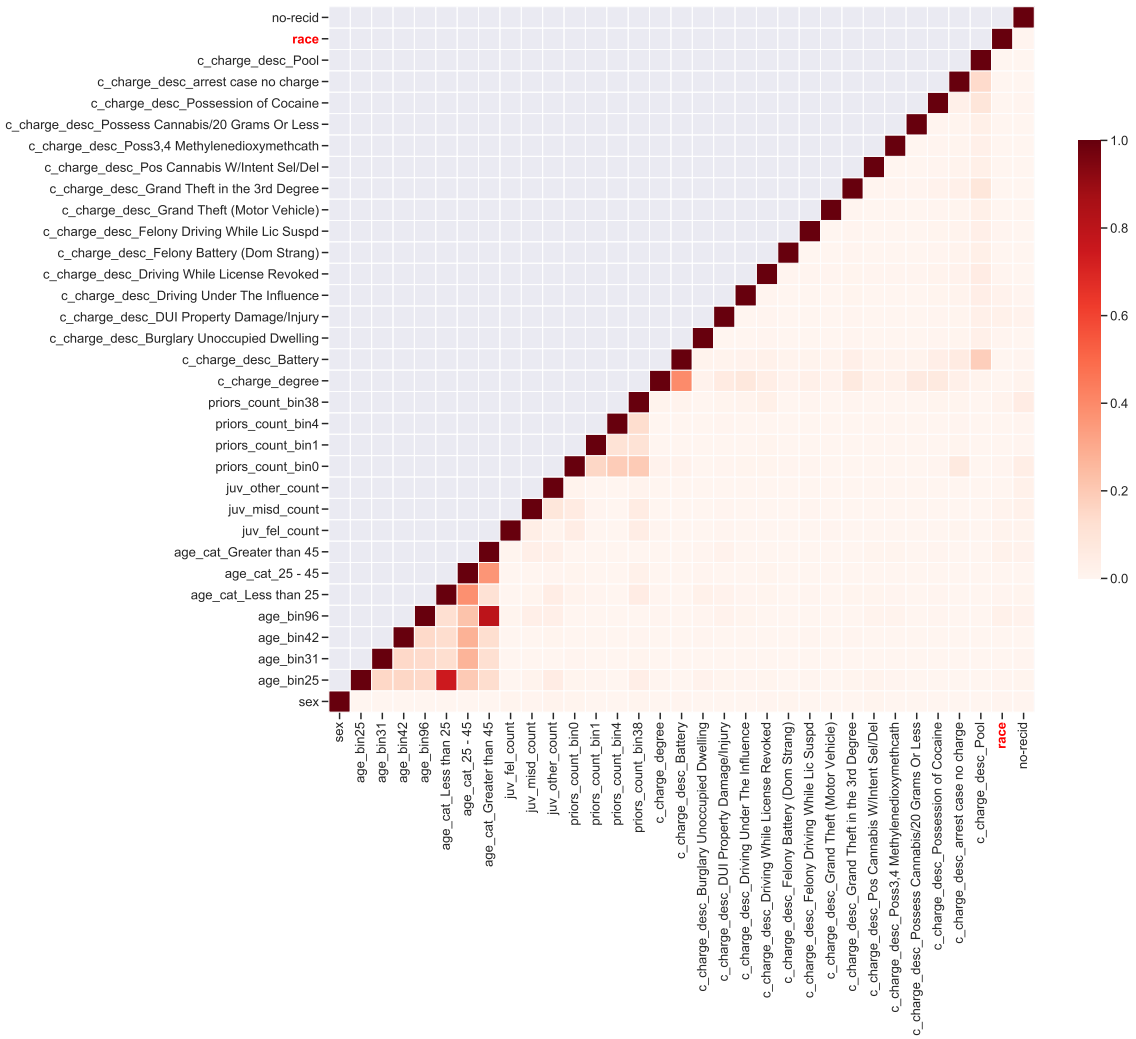


Figure A.5: One-hot encoded version of COMPAS - normalised mutual information.

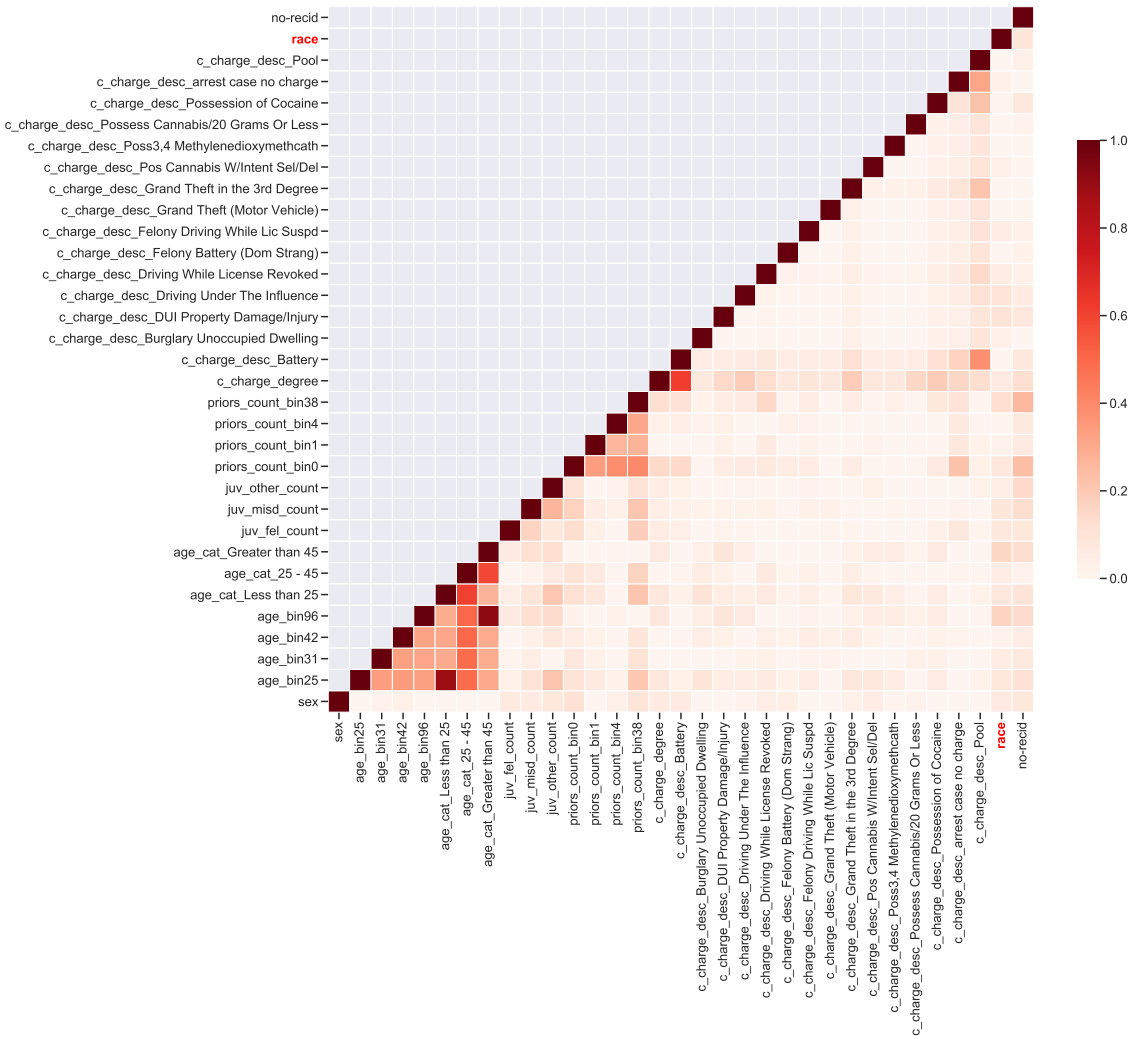


Figure A.6: One-hot encoded version of COMPAS - Cramer's V.