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.

January 2019

UNIVERSIDADE D COIMBRA

Abstract

Nowadays, intelligent systems are ubiquitous, with most organisations and institutions either relying on Machine Learning algorithms to support their decision making systems or completely entrusting their decisions to automated processes. This widespread usage of intelligent systems raises several societal and legal concerns, mainly because of their application in scenarios which have a great impact on people's lives. Although these issues have been receiving attention from regulatory institutions for a long time, there is now a growing focus on them, fuelled by the recently implemented EU General Data Protection Regulation. Fairness is one of the properties systems require to be compliant with the legislation, and it is concerned with the introduction or perpetuation of bias against groups or individuals based on sensitive attributes such as gender or race. Organisations and institutions should be aware of the potential biases in their models at the design, implementation and deployment phases, and make regular fairness assessments.

The main objective of this work is to research and design methods to aid in the development of models and systems that can make fairer predictions. We started by collecting and analysing a set of fairness conditions and metrics widely used in the literature. Focusing on a subset of these metrics, we analysed baseline algorithms commonly used in decision making systems and modified versions which try to improve the models' fairness. The next steps include extending these experiments to more carefully assess the impact different data preparation and pre-processing techniques might have in the system's fairness. Furthermore, we will focus on Artificial Neural Networks and try to incorporate fairness in their architecture or learning process so as to make their predictions less biased. Finally, we will evaluate the effectiveness of the proposed approaches in mitigating discrimination, without neglecting the impact on the overall performance in the main predictive tasks.

Keywords

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

This page is intentionally left blank.

Resumo

Atualmente, os sistemas inteligentes são ubíquos, com a maioria das organizações e instituições a depender de algoritmos de Aprendizagem Computacional para suportar os seus sistemas de tomada de decisão ou a delegar completamente as suas decisões em processos automatizados. Esta utilização generalizada de sistemas inteligentes levanta diversas preocupações sociais e legais, sobretudo pela sua aplicação em cenários com grande impacto na vida das pessoas. Embora estas questões já recebam atenção por parte de instituições reguladoras há vários anos, são agora alvo de um foco acrescido, estimulado pela implementação do Regulamento Geral sobre a Proteção de Dados da UE. A *fairness* é uma das propriedades que os sistemas devem possuir para que cumpram esta legislação, e visa a introdução ou perpetuação de discriminação contra grupos ou indivíduos com base em atributos sensíveis, como sexo ou raça. As organizações e as instituições devem estar cientes de potenciais injustiças nas fases de design, implementação e produção dos modelos, devendo fazer avaliações regulares de *fairness*.

O objetivo principal deste trabalho é investigar e desenhar métodos que auxiliem o desenvolvimento de modelos e sistemas capazes de fazer previsões mais justas. Começámos por recolher e analisar um conjunto de condições e métricas de *fairness* existentes na literatura. Analisámos, com base num subconjunto destas métricas, algoritmos amplamente utilizados em sistemas de tomada de decisão e versões modificadas destes que tentam melhorar a *fairness* dos modelos. Os próximos passos incluem uma avaliação do impacto que diferentes técnicas de preparação e pré-processamento de dados podem ter na *fairness* dos sistemas. Focando em Redes Neuronais Artificiais, iremos tentar incorporar *fairness* na sua arquitetura ou no seu processo de aprendizagem, procurando tornar as suas previsões menos discriminatórias. Por fim, iremos avaliar a eficácia das abordagens propostas a mitigar discriminação, sem negligenciar o impacto no desempenho nas tarefas principais de previsão.

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

ANN Artificial Neural Network. 2, 5, 6, 13, 15, 20, 27

AUC Area Under the Curve. 8

COMPAS Correctional Offender Management Profiling for Alternative Sanctions. 1, 18

CS Computer Science. 3

EU European Union. 1

FN False Negative. 7

FP False Positive. 7

FPR False Positive Rate. 8, 11

GAN Generative Adversarial Network. 12

GDPR General Data Protection Regulation. 1

ID3 Iterative Dichotomiser 3. 5

ML Machine Learning. 1–4, 9, 12

NPI Normalized Prejudice Index. 11, 19, 21

ROC Receiver Operating Characteristic. 8

SVM Support Vector Machine. 13

TN True Negative. 7

TP True Positive. 7

TPR True Positive Rate. 7, 11

USA United States of America. 1

This page is intentionally left blank.

List of Figures

This page is intentionally left blank.

List of Tables

This page is intentionally left blank.

Chapter 1

Introduction

The ubiquity of intelligent systems, and their growing usage in scenarios which have a significant impact on people's lives, raises several legal and societal concerns. One potential problem is that the decisions taken by the algorithms may introduce or perpetuate bias against certain groups or individuals based on sensitive attributes such as race, gender and religion.

The range of scenarios in which automated decision making systems is being used is wide: from loan and mortgage approvals, to hiring and recruiting, or even criminal risk assessment [7, 23, 33, 36]. In particular, criminal recidivism risk scoring has received a lot of attention since the introduction of the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) system in the United States of America (USA) and after ProPublica published a study on the potential racial bias introduced by it [2]. They found that black individuals were mistakenly labelled as potential future criminals at an higher rate than white individuals, while the latter were mislabelled with a lower risk of re-offending at an higher rate than black defendants.

The application of intelligent systems, and in particular Machine Learning (ML), in real-world situations often lacks human supervision, resulting in automated decisions being used in a blind manner. While a clear advantage of shifting the decision to a system is that one might be able to eliminate personal biases from the process, new challenges also surface since these systems should be capable of providing reliable and fair decisions.

Although these issues have been receiving attention from regulatory institutions for a long time, the recently approved European Union (EU) General Data Protection Regulation (GDPR) fuelled the focus on them. GDPR demands organisations to handle personal data in a privacy-preserving, fair and transparent manner [12]. Furthermore, many organisations and governments have already acknowledged that bias introduced by the models might worsen the social imbalance found in our societies [7, 25, 33].

The development of techniques to assess fairness and build fairer models is of great help for organisations which intend to be GDPR compliant, but may lack resources or knowledge [3]. These organisations should be aware of the potential biases in their models at the design, implementation and deployment phases, but should also make regular fairness evaluations of their systems [1]. Moreover, the assessment approaches may be used to perform audits of non-compliant organisations, therefore providing valuable insight of how they are violating these fairness principles [3].

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 which may negatively impact important aspects of their daily lives.

Finally, by understanding how models work regarding fairness, we might get closer to providing an explanation of the procedures followed by the algorithms and the decisions they make.

The current approaches often present limitations in terms of the characteristics of the features or the range of models and fairness notions they support. Most of the proposals to improve fairness also fail to take individual fairness into account.

1.1 Objectives

The main goal of this work is to research and design methods to aid in the development of models which deliver fairer results and decisions. To accomplish this goal, representative datasets must be selected and the fairness concerns that they pose must be well understood.

To assess the degree of unfairness in both the datasets and the models, a set of conditions and metrics, based on different notions of fairness, must also be collected and analysed.

An analysis and evaluation of different data encodings and transformations should be performed to better understand how the preparation and pre-processing of data may impact the fairness of a system.

Due to the widespread application of Artificial Neural Networks (ANNs) and the challenges they pose in terms of interpretability, we plan to research and implement changes to these models so as to reduce the unfairness found in their decisions. We envision that these changes may take several forms, namely the incorporation of a regularization term in the loss function or the development of a new type of fairness-aware layer.

In terms of fairness assessment is concerned, a comparison should be made between the unfairness already observed in the datasets and the improvements, or potential deterioration, introduced by each approach. The evaluation should not only take fairness into consideration, but also the performance of the models. We will evaluate the impact that improvements in fairness might have in the performance of the models.

1.2 Structure

The remainder of this document is organised as follows:

- Chapter 2 provides an overview of ML concepts and algorithms, and reviews the related work on fairness of intelligent systems, both in terms of fairness metrics and proposed techniques to mitigate bias in a system;
- Chapter 3 details the research objectives of this work;
- Chapter 4 presents the experiments carried out during the first semester and includes the analysis of the obtained results;
- Chapter 5 describes the work plan for the second semester;
- Chapter 6 presents the main conclusions of the current work.

Chapter 2

Background and Related Work

This chapter provides an overview of Machine Learning, as well as a description of the most relevant algorithms and techniques for this work. It also encompasses an introduction to fairness notions and concerns in the scope of intelligent systems, followed by a review of previous work in the topic, namely in terms of proposed approaches to improve and assess the models' fairness.

2.1 Machine Learning

Machine Learning (ML) is a sub-field of Computer Science (CS) which comprises a set of algorithms and statistical models that enable computer systems to learn from data, without being explicitly programmed to perform some task, as described by Arthur Samuel, in 1959. Being an interdisciplinary field, it shares concepts and addresses problems also known to statistics, information theory, game theory, and optimisation [30].

Different problems may be categorized 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.

In **supervised learning**, the training examples include not only the features, but also the target outputs which are used to guide the learning process [5, 18]. A further distinction can be made based on the desired output: in **classification problems**, it corresponds to a discrete value within a range of possible categories, whereas in **regression** the output corresponds to one or more continuous variables.

In **unsupervised learning**, we do not have access to the target outputs and the aim is to describe associations and patterns within the data [5, 18]. **Clustering** is a typical example of an unsupervised learning task in which the goal is to find groupings of similar examples in the data.

By interacting with the environment, the aim of **reinforcement learning** problems is to learn mappings of situations to actions, while maximizing a reward. In contrast to supervised learning, the optimal actions are not given to the learning algorithm, but it instead 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 [5, 32].

A system which relies on ML usually follows a pipeline as shown in Figure 2.1: 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.

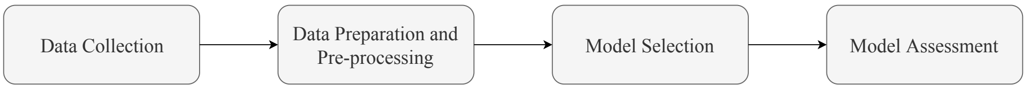


Figure 2.1: A typical Machine Learning pipeline.

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, discretization, feature normalization, feature selection and feature reduction techniques. Not only is the application of these techniques of pivotal importance for some models to deliver the expected results, but it also helps dealing with overfitting, a common problem in ML which is described in section 2.1.2.

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 [19]. **Model assessment** deals with evaluating the performance of the chosen model by estimating its generalization error on new unseen data [18, 19]. The methods used to address these phases of the pipeline are further discussed in section 2.1.2.

2.1.1 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 [5]. 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 [30]. 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 [27]. 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 [5, 36]. 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 $y_i \in \{0, 1\}$ [5]. To avoid overfitting, a penalty term (also called regularizer) can be added to the cost function. Common choices are L2 or L1 regularization.

- **Decision Trees** are classifiers which try to learn simple decision rules from the available features in the training data [27]. 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 [19]. Common choices for this criterion include the classification error rate, the Gini index and cross-entropy [19]. 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 [30]. To reduce the correlation between the trees, the candidates for splitting are randomly selected from the full set of input features before each split [18]. This randomization process also aims at reducing variance [19].

- **Artificial Neural Networks (ANNs)** are non-linear statistical models that take inspiration from the neural networks in the brain [30]. 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 [30]. In a feedforward neural network, as shown in Figure 2.2, the underlying graph is acyclic [30].

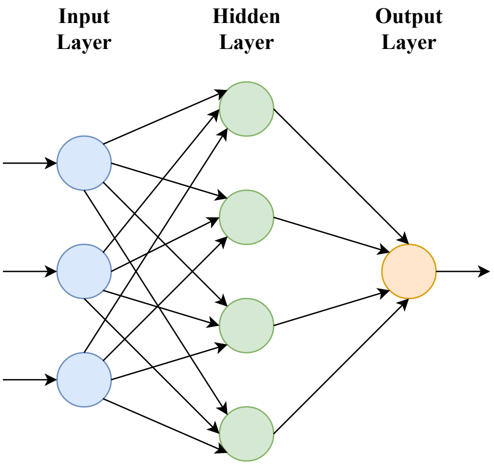


Figure 2.2: An example of a feedforward ANN with a single hidden layer.

We usually assume that the network is organized by layers, where each layer corresponds to a disjoint subset of nodes (neurons) [30]. 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 [30]. 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 [5].

The learning process of an artificial neural network consists of tuning the weights of the connections between the neurons in order to minimize 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 [5]. Several algorithms can be used to train a ANN, namely the backpropagation algorithm can 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 [5].

Several approaches have been suggested to help dealing with overfitting in an ANN, some of which include: early stopping [5], weight decay [5], and dropout layers.

2.1.2 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 [19], while variance measures the sensitivity of the model to the dataset used to fit it [5]. 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 between bias and variance is also closely related with 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 generalization performance, which is related to the model's capability of making accurate predictions given new unseen samples [18]. 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 generalization error of the chosen model [18].

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 [19].
- **K-fold cross-validation** randomly splits the available data into K groups (or folds) of approximately the same size [19]. 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 generalization 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 [18].

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** summarizes 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 [24].

- **Accuracy** is the most widely used metric to evaluate the performance of an algorithm and is given by the ratio between correctly classified samples and the total number of samples:
$$accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{2.4}$$
Despite its widespread usage, it might lead to misleading results in unbalanced scenarios and when incorrect classifications do not have the same cost.

- **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)**, is given by the fraction of positive samples that are correctly classified:
$$recall = \frac{TP}{TP + FN} \tag{2.6}$$

- **F1 score** is given by the harmonic mean of precision and recall:
$$F1\ score = 2 \times \frac{precision \times recall}{precision + recall} = \frac{2 \times TP}{2 \times TP + FN + FP} \tag{2.7}$$

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

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

- **Sensitivity** is a term more used in medical scenarios, which is actually defined in the same way as recall.
- **Specificity** is usually used in medical scenarios alongside sensitivity and is given by the fraction of negative samples that are correctly classified:

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

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 [18]. Figure 2.3 shows an example of a ROC curve.

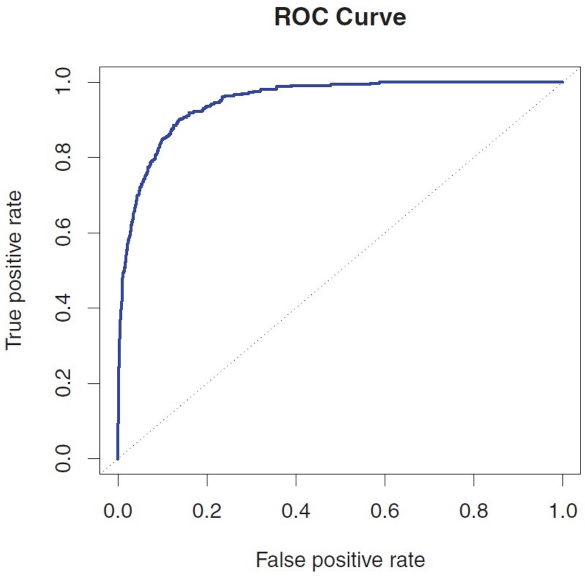


Figure 2.3: An example of a ROC curve, taken from [19].

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 [13]. Furthermore, they suffer from prevalence, bias, skew and cost ratio which might affect their usefulness and representativeness [28]. 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 ubiquity of intelligent systems has led to the outputs of ML models being used as a basis for real-world decisions. When these decisions impact the life of individuals, several legal and societal concerns arise, some of which related to discrimination and fairness [4]. Furthermore, discrimination by certain attributes of an individual, like race and gender, are often prohibited by law, as in the case of hiring and housing processes in the USA [9, 14]. We refer to these as *sensitive* or *protected attributes.*

Unfairness in intelligent systems may appear in many different forms and may have a variety of root causes. The available training data may be itself unfairly sampled or labelled [21], if certain groups are under-represented in the data or the labels in the training data are a result of biased decisions. This might be the case if a bank has been unfairly rejecting loans of people who belong to a certain minority group [21]. Moreover, even if the models are trained on historical data that contain biases against certain social-demographic variables, these biases should not be perpetuated by the algorithmic decisions.

A direct form of discrimination, referred to as *disparate treatment*, occurs when sensitive attributes are directly used to make decisions [34]. This type of discrimination can be avoided by removing the sensitive attributes from the data, prior to training the model [34]. However, due to the presence of features which are correlated with the sensitive attributes, the removal of the latter might be insufficient to prevent discriminatory decisions. This phenomenon is known as the *red-lining effect* and is linked to *disparate impact*, an indirect form of discrimination which occurs when protected groups are unfairly treated without this outcome resulting from an explicit utilization of sensitive attributes [14, 29, 34].

This notion of disparate impact actually resulted from a decision made in a US law case [26] which prohibited the requirement of a high-school diploma, which happens to be highly correlated with race, to make hiring decisions [14, 29]. When suspicions of disparate impact emerge, the defendant has to give objective and reasonable justifications to proof the absence of discrimination [29].

The notion of *individual fairness* was later introduced by [11] and is concerned with giving similar treatment to similar individuals. Several challenges arise with this notion, namely how to measure the similarity between individuals.

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 as samples or individuals): X are the non-sensitive attributes, S denotes a sensitive attribute, and Y is the target output. The variable predicted by a classifier is referred to as $\hat{Y}$.

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 the instances for which the value of the sensitive attribute is 1 is called the unprotected or privileged group.

Some of the most widely used metrics to assess fairness at a group level include statistical parity, equalized odds and disparate impact. Nevertheless, many others have been proposed, some of which also take fairness at an individual level into account. All these metrics are suitable for classification problems.

- **Statistical or demographic parity** requires that the output be independent of the sensitive attribute [38], 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 the decisions made by a binary classifier, statistical parity translates into the rate of favourable predictions being the same across all values of the sensitive attribute [38]:
  $$P(\hat{Y} = 1) = P(\hat{Y} = 1|S = s) \tag{2.10}$$

  For binary classification problems with a single binary attribute, a variation of statistical parity sometimes called **statistical parity difference**, or risk difference, considers the difference of the rate of favourable predictions between the protected and unprotected groups:
  $$P(\hat{Y} = 1|S = 1) - P(\hat{Y} = 1|S = 0) \tag{2.11}$$

  In [21], this variation is referred to as the Calders-Verwer score (CVS).

  Statistical parity can be applied not only to compute the discrimination in the predictions made by a classifier, but also in a labelled dataset.

  As pointed out by [11] and [17], 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 [17].

- **Disparate impact** can only be applied in a binary classification scenario and is given by the ratio of the rate of favourable predictions for the protected group to that of the unprotected group [8]:
  $$\frac{P(\hat{Y} = 1|S = 0)}{P(\hat{Y} = 1|S = 1)} \tag{2.12}$$

  This is often referred to as the p%-rule and for a classifier to be considered fair it should be no less than 80%, meaning that it does not have disparate impact [14, 36]. The 80 percent rule is actually advocated by the US Equal Employment Opportunity Commission [14]. This metric is often applied to the labelled dataset itself.

  The authors of [14] 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 [14].

- **Equalized odds** is satisfied if the predicted variable and the sensitive attribute are independent conditional on the true output [17]. This metric is not restricted to classification scenarios.

Considering the case where all three variables are binary, equalized odds requires that the TPRs and the FPRs are the same for both the protected and unprotected groups [17]:

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

where $y \in \{0,1\}$.

- **Equal opportunity** can only be directly applied in binary classification problems. It can be regarded as a relaxation of equalized 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 [17]:

$$P(\hat{Y}=1|S=0,Y=1)=P(\hat{Y}=1|S=1,Y=1) \tag{2.14}$$

- The **prejudice index** (PI) as defined by [21] corresponds to the mutual information between the target output and the sensitive attribute. The Normalized Prejudice Index (NPI) results from the application of a normalization technique for mutual information and is given by:

$$NPI=\frac{PI}{\sqrt{H(Y)H(S)}} \tag{2.15}$$

where $H(\cdot)$ is an information entropy function [21]. The NPI can be regarded as the geometrical mean of the ratio of information of the sensitive attribute used for predicting the target output, and the ratio of exposed information if a value of the target output is known [21]. This metric can be applied to the labelled dataset or the predictions made by a classifier.

- **Consistency** as defined by [37] is used as an individual fairness metric which measures the similarity between predictions for similar samples:

$$1-\frac{1}{n\times k}\sum_{i=1}^{n}|\hat{y}_i-\sum_{j\in kNN(\mathbf{x}_i)}\hat{y}_j| \tag{2.16}$$

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 **generalized entropy index** is used by the authors of [31] to measure the *overall individual-level unfairness* of an algorithm and is given by:

$$GE(\alpha)=\frac{1}{n\alpha(\alpha-1)}\sum_{i=1}^{n}\left[\left(\frac{b_i}{\mu}\right)^{\alpha}-1\right], \alpha\notin\{0,1\} \tag{2.17}$$

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. When $\alpha=2$, the value used throughout their work, we get half the squared coefficient of variation.

This benefit function 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 [31] .

The overall individual-level unfairness takes fairness at both an individual and group level into account and can be further decomposed into a between-group and a within-group component [31]. The between-group component is similar to other group level notions of fairness.

2.2.2 Fairness Improvement

Many approaches have been proposed to improve fairness or mitigate bias in ML models. These can be grouped in three different 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 regularization terms to the objective functions; and the **post-process approaches** directly change the predictions made by the models [34].

The methods proposed by [14] 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 [34]. 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 [14].

The authors of [37] 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 [37]. This proposal addresses fairness at both a group and an individual level.

FairGAN, the method proposed in [34], takes inspiration from Generative Adversarial Networks (GANs) [16] 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 [6], the authors propose three methods to modify Naive Bayes classifiers so as to remove discrimination from the predictions. 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 target output 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 [6]. This is accomplished by modifying the joint distribution of the sensitive attribute and the target variable so that the statistical parity difference approaches zero [21].

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 [6]. 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 2.4. From their experiments, this method seems to lead to the best results.

The third method considers the addition of a latent (or hidden) variable to the Bayesian model and uses expectation maximization to optimize the parameters for the likelihood function [6]. This latent variable aims at representing an unbiased target output. For more details regarding any of these three methods, refer to [6].

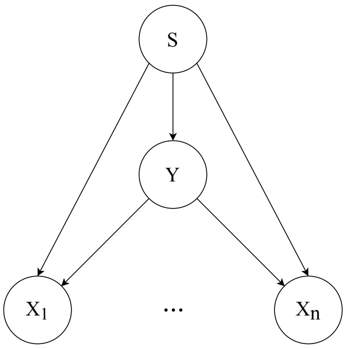


Figure 2.4: Graphical model for the two Naive Bayes method, adapted from [6].

The proposal of Kamishima et al. [21], which fits in the in-process category, is to add a regularizer to a logistic regression model so as to reduce the indirect discrimination during the learning process. The prejudice remover regularizer, $R_{PR}(\mathcal{D}, \mathbf{\Theta})$, is based on the prejudice index and is given by:

$$R_{PR}(\mathcal{D}, \mathbf{\Theta}) = \sum_{(\mathbf{x_i}, s_i) \in \mathcal{D}} \sum_{y \in \{0,1\}} p(y|\mathbf{x}_i, s_i, \mathbf{\Theta}) \ln \frac{\hat{p}(y|s_i)}{\hat{p}(y)} \tag{2.18}$$

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 regularizer takes large values when the predicted class is mainly determined by the sensitive feature.

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

$$-\sum_{i=1}^{n} \ln p(y_i|\mathbf{x}_i, s_i, \mathbf{\Theta}) + \eta R_{PR}(\mathcal{D}, \mathbf{\Theta}) + \frac{\lambda}{2} \sum_{s \in S} \|\mathbf{w}_s\|_2^2 \tag{2.19}$$

where η and λ are positive regularization parameters. For further details on the learning process followed by the authors, refer to [21].

The authors of [36] 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 [36]. Taking this measure as a basis, they derive two formulations of constrained optimization problems: one which maximizes accuracy subject to fairness constraints (ensuring compliance with a given p%-rule, if demanded by law) and another which maximizes fairness subject to accuracy constraints (ensuring that certain business needs are met, by allowing a relaxation on fairness) [36]. They show concrete instantiations of these problems using logistic regression classifiers and Support Vector Machines (SVMs).

In a more recent work, an adversary is added to an ANN with the goal of penalizing models for which it is possible to predict the sensitive attribute from the outcomes [33]. The authors take more than one fairness notion into consideration when designing their adversary. In [38], 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 proposals by [15] and [17] fit in the post-process category. In particular, the authors of [17] propose a post-processing step to achieve equalized odds and equal opportunity.

This page is intentionally left blank.

Chapter 3

Research Objectives

The main goal of this dissertation is to research and design methods to aid in the development of models which are able to make fairer predictions. To accomplish this goal, a set of intermediate objectives should be achieved:

- **Analysis of existing fairness conditions and metrics.**
  We need to 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. From this analysis, we should be able to select the most suitable metrics for the assessment of fairness in the scope of this work.

- **Analysis and evaluation of the impact of different data preparation and pre-processing techniques in the system's fairness.**
  We want to determine if different encodings and transformations of the features influence the ability of the algorithms in making fair predictions. Prior to this analysis, we must collect and understand the fairness concerns of a set of representative datasets.

- **Research and development of methods to mitigate unfairness in ANNs.**
  The application of ANNs in real-world scenarios has been growing, with these models delivering good results in a wide variety of tasks. We want to study these models while addressing fairness, an important property of systems that are used in contexts which have an impact on people's lives. Moreover, ANNs pose several challenges in terms of the interpretability of their inner workings and results.

A cross-cutting concern of this work is the evaluation of the **trade-off between the effectiveness of the proposed approaches in mitigating bias and the overall performance of the systems** in the main predictive 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 knowledge about these trade-offs and allow them to make informed decisions. Furthermore, we are always concerned with the improvements made with respect to the unfairness originally found in the datasets.

There are several possibilities when it comes to the third objective of developing fairness-aware ANNs. We want to focus on either incorporating fairness in the architecture or the learning process of these models. We want to further research the feasibility of introducing changes to the loss function or developing a new type of layer which performs some regularization-like operation regarding fairness. This new type of layer could be similar to a dropout layer, but instead of trying to reduce overfitting, it would try to mitigate the unfairness of the predictions produced by the model.

This page is intentionally left blank.

Chapter 4

Current Work and Preliminary Results

This chapter presents the work developed during the first semester, along with a discussion of the obtained results. Special focus was given to understanding the main concerns and concepts of fairness, followed by a state-of-art analysis of proposed approaches to assess and mitigate unfairness found in intelligent systems. We collected and analysed a set of datasets commonly used in the studied approaches. Furthermore, we reproduced and extended a work that fits in the category of in-process methods, as explained in 4.2.

4.1 Datasets

Three datasets are recurrently used in previous studies on fairness: the Adult Income dataset (also known as Census Income) [10], the German Credit Data dataset [10], and the COMPAS dataset [2]. This section provides a brief explanation of these datasets in terms of their main predictive task and the fairness concerns that they pose.

4.1.1 Adult Income dataset

The Adult or Census Income dataset is publicly available from the UCI Machine Learning Repository [10]. 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, gender, marital status, race, level of education, occupation, and working hours per week. There are 48842 instances in the dataset and a split into training (32561 instances) and test (16281 instances) sets is provided.

The main task is to predict whether a person earns over 50,000 dollars per year, therefore making a classification into high or low income. Such predictions might be used, for instance, to make decisions regarding the assignment of loans and might lead to legal actions against institutions if deemed unfair [6].

Most studies use gender as the sensitive attribute, since 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. Race as also been used in previous work as a sensitive attribute [31, 36].

4.1.2 German Credit Data dataset

The German Credit Data dataset is publicly available from the UCI Machine Learning Repository [10]. In our experiments, we used the original version of the dataset which contains financial information about 1000 individuals, described by a set of 20 categorical and numerical attributes. Some of these attributes include age, gender and marital status, existing checking account status, duration, credit amount, employment status, and type of housing (own, rented, or free).

The objective is to classify each person into one of two possible credit classes: good or bad credit. 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 gender and being a foreign worker are also considered as such in some previous work. As reported by [20], young people are less likely to be classified into the good credit class than aged people, which raises fairness concerns.

4.1.3 COMPAS dataset

The COMPAS dataset was compiled by ProPublica [2] 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 during 2013 and 2014. Besides the risk score given by COMPAS, the data includes demographic information (such as gender, age and race) and attributes related with the criminal history (such as the type of offence they were arrested for and the number of prior offences) of each individual [35].

The goal is to predict whether a criminal defendant will reoffend within two years or, in other words, to predict the 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 reoffending, or not, within two years after being given the score by the tool.

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 [2]. 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 deserved being given to the defendant [2].

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 black individuals with high risk at an higher rate than white individuals [22]. Additionally, they found white defendants were mislabelled with a lower risk of recidivism at an higher rate than black defendants [22]. This means black 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 reoffending than a man who receives the same score [22].

Further details about how the data was collected and analysed can be found in [22]. This dataset was not used during the experiments that we performed in the first semester.

4.2 Preliminary Experiments

From the different approaches found in the literature, we focused on the ones from the in-process category, as they appear to be the ones with the most room for improvement regarding fairness. Some preliminary experiments were performed to reproduce the work of Kamishima et al. [21], which fits in this category. Studying this work allows us to understand the feasibility of the approach in terms of modifying the objective function of a model and to better understand the strengths and limitations of the existing fairness metrics. Thus, it aligns with the goal of researching and developing fairness-aware models.

We extended the experiments in [21] by considering another dataset and analysing a wider range of baseline algorithms. We wanted to better understand how a set of baseline algorithms, ranging from white-box and black-box models, behaved regarding fairness. It is important to mention that our goal with these preliminary experiments was not to tune the parameters of the models to achieve the best possible predictive accuracy.

Aligning with the goal of understanding the impact of data preparation and pre-processing techniques on the fairness of a system, we always considered two versions of the datasets in our experiments.

From the analysed fairness metrics, we selected two which can be applied to both the datasets and the predictions made by the models: statistical parity difference and Normalized Prejudice Index (NPI). The choice of these metrics is supported by the fact that they were used by the authors whose work we aimed at reproducing. Furthermore, we wanted to compare the unfairness already found in the datasets to the unfairness resulting from the application of different algorithms.

Besides evaluating the performance of the models and the fairness found in both the datasets and the predictions, we also made an exploratory data analysis of the features so as to better understand the relationship between the models' decisions and the features used during the learning process. We computed the pairwise mutual information, as well as the pairwise Pearson correlation coefficient between attributes. Additionally, we computed these metrics between each attribute and the predictions made by the different models.

4.2.1 Experimental Setup

We started by reproducing the work of Kamishima et al. [21], in which the authors add a regularization term to the objective function of a logistic regression model with the goal of reducing the prejudice index during the learning process. The η parameter controls the degree of unfairness that is removed [21].

In the original experiments, the authors use the test set of the Adult Income dataset, taking gender as the sensitive attribute and `female` as the unprivileged group. They follow the feature discretization approach of Calders and Verwer [6]: the numerical attributes are discretized into 4 bins with the boundaries corresponding to the boundaries of the interquartile ranges; bins which correspond to low frequency counts (less than 50) are pooled together and the attribute values are replaced by the same `pool` value. They use this version of the dataset to train and test the Naive Bayes models.

Furthermore, they create another version of the dataset with all features using a one-hot (or 1-of-K) encoding scheme [5] after being discretized, meaning that they are represented by binary dummy variables. They use this version of the dataset to train and test the logistic regression models.

The authors make a comparison between a set of algorithms, always considering models trained with, and without, the sensitive attribute, except for their proposed approach and the two Naive Bayes approach by Calders and Verwer [6]. Thus, their study consists of a set of six methods: logistic regression with a sensitive attribute (LR), logistic regression without a sensitive attribute (LRns), their approach of logistic regression with a fairness regularizer (PR), Naive Bayes with a sensitive attribute (NB), Naive Bayes without a sensitive attribute (NBns), and the two Naive Bayes approach by Calders and Verwer (CV2NB).

In our experiments, we used their implementation of all methods excluding the baseline logistic regression, for which we used the implementation provided by the Scikit-learn Python package [27], with L2 regularization. The regularization parameter is fixed to 1. For further details about the implementations of the authors, refer to [21].

After reproducing their work, we used both versions of the Adult Income dataset with each of the six methods considered in the original paper. However, an additional step was required when dealing with the dataset where the features were not one-hot encoded: samples containing at least one missing value were dropped. It is not clear how the authors handled missing values, but this step was deemed necessary since some of the implementations we used from Scikit-learn do not support missing data.

We also conducted experiments with the German Credit Data dataset, taking age as the sensitive attribute and `young` as the unprivileged group. We transformed this attribute into a binary feature by considering a cut-off value of 25, a common threshold in other studies on the topic [14, 20]. We also transformed the `personal status and sex` attribute so as to only represent gender. We discretized numerical attributes with more than 4 different values into 4 bins, as done with the Adult Income dataset, but we did not pool low frequency counts.

Additionally, we considered decision trees (DTs), random forests (RFs), and Artificial Neural Networks (ANNs) in our experiments. Following the work we were reproducing, two variants of each of these methods were always trained and tested: one with the sensitive feature and another without the sensitive feature (we append the `ns` suffix to the name of this variant of the models). We used the Scikit-learn's implementations of these methods with default parameters, except for the hidden layer of the ANNs for which we set the number of neurons to be twice the number of features in the training set.

Furthermore, we performed experiments with Scikit's implementations of Naive Bayes, since we wanted to compare them to the implementation used in [21]. For the version of the datasets with one-hot encoded features, we used a Bernoulli Naive Bayes (BNB), while for the other version of the datasets we used a Gaussian Naive Bayes (GNB).

To evaluate the performance of the different methods in the main predictive task, we performed five-fold cross-validation, as done by [21], and computed their accuracy. Initially, we used the folds as defined by the authors of the original experiments, but afterwards performed cross-validation with the help of the methods provided by Scikit-learn [27].

4.2.2 Exploratory Data Analysis

An overview of the one-hot encoded version of the Adult Income dataset is shown in Table 4.1. The unprivileged group, `females`, only represents around 33.30% of the dataset. Furthermore, only around 15.34% of the favourable classifications correspond to `females`. For this version of the Adult Income dataset, the statistical parity difference is 0.191 and the NPI is 4.20×10^{-2}, when taking `gender` as the sensitive attribute.

		Sensitive Attribute	
		Male	Female
Target Output	High income	3256	590
	Low income	7604	4831

Table 4.1: Overview of the one-hot encoded version of the Adult Income dataset.

For the integer encoded version of the dataset, some samples are dropped due to the presence of missing values. Table 4.2 presents an overview of this version of the dataset. In this case, `females` represent around 32.62% of the dataset and only around 15.05% of the favourable classifications. The statistical parity difference slightly increases to 0.196 and the NPI also suffers a slight increase to 4.25×10^{-2}.

		Sensitive Attribute	
		Male	Female
Target Output	High income	3143	557
	Low income	7004	4356

Table 4.2: Overview of the integer encoded version of the Adult Income dataset.

These values of statistical parity difference reveal some 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 target output are not independent.

Table 4.3 presents an overview of the other dataset used in the experiments, the German Credit Data. In this case, `young` individuals constitute the unprivileged group and are represented by 14.90% of the dataset. Only 12.57% of the favourable classifications (`good credit`) are assigned to the unprivileged group.

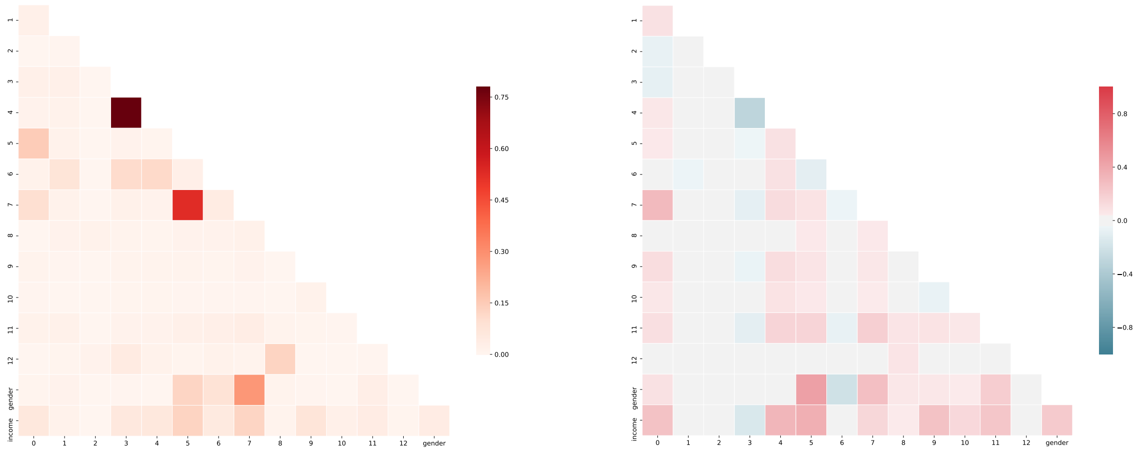
		Sensitive Attribute	
		Aged	Young
Target Output	Good credit	612	88
	Bad credit	239	61

Table 4.3: Overview of the German Credit Data dataset.

For both versions of the dataset, the statistical parity difference is 0.129 and the NPI is 9.39×10^{-3}, when taking `age` as the sensitive attribute. It is possible to draw conclusions similar to those regarding the previous dataset, since `young` individuals might be receiving unfair treatment.

Figures A.1 and A.2, found in Appendix A, present the pairwise mutual information and pairwise Pearson correlation between features, for the one-hot encoded version of the Adult Income dataset. Attributes 60, 41, 57 and 58 are correlated with the sensitive attribute. According to the results obtained for mutual information, these attributes are also dependent on the sensitive attribute. Special attention should be given to attributes 60 and 41, which are also correlated with the target output. Thus, it is likely that indirect discrimination emerges in the predictions made by the models.

For the integer encoded version of the Adult Income dataset, the pairwise mutual information and the pairwise Pearson correlation between features are shown in Figures 4.1a and 4.1b. Attributes 5 and 7 seem to share common information with the sensitive attribute. Furthermore, attribute 5 also seems to be correlated with the target output. Thus, removing the sensitive attribute from the models may not be sufficient to eliminate the unfairness found in the dataset.



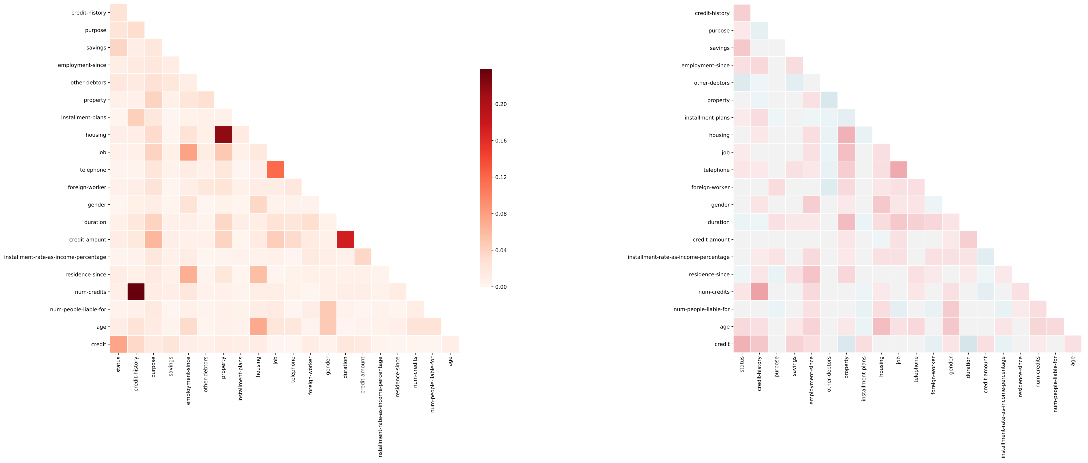
(a) Mutual information between features.

(b) Pearson correlation between features.

Figure 4.1: Integer encoded version of the Adult Income dataset.

Figures A.3 and A.4, found in Appendix A, present the pairwise mutual information and pairwise Pearson correlation between features, for the one-hot encoded version of the German Credit Data dataset. The sensitive attribute `age` has some correlation with the attribute `housing_A151`, which might lead to indirect discrimination. As far as mutual information is concerned, there seems to be no strong dependency between the sensitive attribute and the other attributes.

Figures 4.2a and 4.2b show the pairwise mutual information and pairwise Pearson correlation between features, for the integer encoded version of the German Credit Data dataset. The sensitive attribute `age` is correlated with some of the other attributes, namely `housing`, which is further corroborated by the mutual information results. This might potentially lead to discriminatory results, even if the sensitive attribute is removed from the dataset prior to training the models.



(a) Mutual information between features.

(b) Pearson correlation between features.

Figure 4.2: Integer encoded version of the German Credit Data dataset.

4.2.3 Further Results and Discussion

In this section, we present average statistics computed over all five folds. We present the accuracy (ACC) of the models, the statistical parity difference (CVS) in the predictions, and the normalized prejudice index (NPI) in the predictions. Furthermore, we present the ratio between the CVS in the predictions with respect to the CVS found in the dataset (CVS Ratio), as well as a similar ratio regarding NPI (NPI Ratio). These ratios give an indication of whether the algorithm reduced or augmented the unfairness found in the dataset, according to the underlying fairness metric. A value of 1 for these ratios means that the unfairness after the application of the algorithm remained the same, an absolute value greater than 1 means that this application increased the unfairness found in the dataset, and an absolute value less than 1 means the algorithm reduced the unfairness found in the dataset.

The obtained results using the one-hot encoded version of the Adult Income dataset are presented in Table 4.4, while the results for the integer encoded version of the same dataset are presented in Table 4.5.

We were able to compute the selected fairness metrics at both stages of the process, meaning that their usage to evaluate the impact of the algorithms with respect to the unfairness originally found in the datasets is a possibility. Nevertheless, caution must be taken when drawing conclusions, since these seem to be dependent on the chosen metric. In some cases, according to the CVS ratio, the methods seem to have been able to reduce the unfairness found in the original dataset, while according to the NPI ratio, the unfairness actually seems to have been increased. Moreover, if we were to sort the methods by one of these ratios, their relative ordering would also be dependent on the underlying metric.

In general, removing the sensitive attribute from the models leads, as expected, to improvements in fairness, regardless of the fairness metric considered in this evaluation. However, two exceptions occur: for the one-hot encoded version of the dataset and the random forest models, removing the sensitive attribute seems to lead to more discriminatory decisions; the other case is with the integer version of the dataset and decision trees, where removing the sensitive attribute seems to have an almost negligible effect. It is also interesting to notice that the decision trees can improve fairness from an unfair dataset, without the aid of any further mechanisms.

method	ACC		NPI		CVS		NPI Ratio	CVS Ratio
	mean	std dev	mean	std dev	mean	std dev		
LR	0.851	0.005	5.14E-02	1.78E-03	0.188	0.003	1.223	0.983
LRns	0.851	0.006	4.87E-02	2.65E-03	0.183	0.003	1.160	0.960
PR $\eta = 5$	0.851	0.005	4.99E-02	1.93E-03	0.186	0.002	1.189	0.974
PR $\eta = 15$	0.796	0.007	2.01E-02	3.20E-03	0.042	0.006	0.478	0.222
PR $\eta = 30$	0.766	0.007	9.60E-03	3.91E-03	0.004	0.003	0.229	0.021
NB	0.804	0.008	1.38E-01	7.81E-03	0.382	0.014	3.278	2.002
NBns	0.808	0.006	1.03E-01	7.54E-03	0.333	0.013	2.464	1.745
CV2NB	0.808	0.013	9.61E-04	1.30E-03	-0.003	0.033	0.023	-0.016
PR $\eta = 0$	0.851	0.005	4.99E-02	1.93E-03	0.186	0.002	1.189	0.974
PR $\eta = 10$	0.851	0.005	4.99E-02	1.93E-03	0.186	0.002	1.189	0.974
PR $\eta = 20$	0.813	0.032	3.26E-02	1.65E-02	0.095	0.083	0.778	0.498
BNB	0.804	0.008	1.38E-01	7.83E-03	0.383	0.014	3.281	2.003
BNBns	0.808	0.006	1.03E-01	7.74E-03	0.333	0.014	2.461	1.743
DT	0.793	0.005	3.03E-02	2.21E-03	0.156	0.005	0.721	0.815
DTns	0.791	0.001	2.97E-02	4.54E-03	0.154	0.010	0.706	0.807
RF	0.823	0.009	4.57E-02	3.42E-03	0.177	0.005	1.089	0.927
RFns	0.818	0.010	4.95E-02	6.56E-03	0.184	0.010	1.178	0.964
NN	0.827	0.011	4.54E-02	1.07E-02	0.187	0.033	1.082	0.980
NNns	0.825	0.006	4.38E-02	8.11E-03	0.183	0.024	1.043	0.957

Table 4.4: Results for the one-hot encoded version of the Adult Income dataset.

method	ACC		NPI		CVS		NPI Ratio	CVS Ratio
	mean	std dev	mean	std dev	mean	std dev		
LR	0.827	0.009	6.36E-02	7.81E-03	0.198	0.013	1.495	1.009
LRns	0.828	0.009	5.54E-02	9.80E-03	0.187	0.016	1.304	0.951
PR $\eta = 5$	0.827	0.008	5.28E-02	5.72E-03	0.185	0.011	1.241	0.942
PR $\eta = 15$	0.827	0.008	5.28E-02	5.72E-03	0.185	0.011	1.241	0.942
PR $\eta = 30$	0.827	0.008	5.28E-02	5.72E-03	0.185	0.011	1.241	0.942
NB	0.818	0.006	1.11E-01	9.14E-03	0.337	0.017	2.618	1.714
NBns	0.823	0.007	7.35E-02	9.59E-03	0.274	0.020	1.729	1.394
CV2NB	0.808	0.010	4.69E-04	4.24E-04	-0.004	0.024	0.011	-0.021
PR $\eta = 0$	0.827	0.008	5.28E-02	5.72E-03	0.185	0.011	1.241	0.942
PR $\eta = 10$	0.827	0.008	5.28E-02	5.72E-03	0.185	0.011	1.241	0.942
PR $\eta = 20$	0.827	0.008	5.28E-02	5.72E-03	0.185	0.011	1.241	0.942
GNB	0.796	0.009	8.27E-02	1.05E-02	0.279	0.017	1.946	1.419
GNBns	0.798	0.011	5.15E-02	1.06E-02	0.213	0.020	1.211	1.084
DT	0.786	0.010	2.91E-02	4.75E-03	0.158	0.012	0.684	0.805
DTns	0.786	0.009	2.91E-02	4.49E-03	0.158	0.012	0.685	0.804
RF	0.814	0.010	5.07E-02	9.65E-03	0.197	0.017	1.194	1.002
RFns	0.812	0.008	4.76E-02	9.89E-03	0.189	0.019	1.120	0.964
NN	0.833	0.007	5.04E-02	6.98E-03	0.186	0.019	1.186	0.948
NNns	0.833	0.012	4.59E-02	1.05E-02	0.175	0.027	1.081	0.886

Table 4.5: Results for the integer encoded version of the Adult Income dataset.

When it comes to the Naive Bayes methods, the implementation of the Bernoulli Naive Bayes from Scikit-learn seems to lead to similar results as those obtained with the implementation provided by the authors. The same does not happen with the Gaussian Naive Bayes implementation used for the integer encoded version of the dataset. This is not so surprising since the authors' implementation assumes a multinomial distribution for the likelihood of the features. Therefore, the implementations provided by the Scikit-learn package might need further adjustments to handle integer encoded features.

Another aspect that should be pointed out pertains with the results obtained with the method proposed by the authors (PR) using the integer encoded version of the dataset. In fact, the η parameter does not seem to have any impact on the performance or on the fairness of the models, as shown in Table 4.5. Furthermore, it seems to fail to remove the unfairness found in the dataset.

The obtained results for the one-hot encoded version of the German Credit Data dataset are presented in Table 4.6, while the results for the integer encoded version of this dataset are presented in Table 4.7.

method	**ACC**		**NPI**		**CVS**		**NPI Ratio**	**CVS Ratio**
	mean	**std dev**	**mean**	**std dev**	**mean**	**std dev**		
LR	0.760	0.044	3.78E-02	4.75E-02	0.176	0.182	4.019	1.368
LRns	0.757	0.045	2.77E-02	3.71E-02	0.144	0.160	2.953	1.124
PR $\eta = 5$	0.745	0.050	1.83E-02	1.79E-02	0.042	0.167	1.948	0.329
PR $\eta = 15$	0.744	0.044	1.29E-02	1.41E-02	0.020	0.145	1.370	0.156
PR $\eta = 30$	0.745	0.046	1.59E-02	1.88E-02	0.016	0.155	1.696	0.126
NB	0.739	0.048	2.63E-02	3.11E-02	0.180	0.140	2.801	1.400
NBns	0.736	0.048	1.52E-02	1.90E-02	0.120	0.129	1.622	0.937
CV2NB	0.730	0.048	1.82E-02	2.57E-02	0.004	0.186	1.943	0.028
PR $\eta = 0$	0.742	0.051	2.26E-02	2.59E-02	0.149	0.120	2.407	1.158
PR $\eta = 10$	0.740	0.049	1.14E-02	1.13E-02	0.036	0.136	1.217	0.282
PR $\eta = 20$	0.747	0.044	1.49E-02	1.94E-02	0.008	0.150	1.583	0.064
BNB	0.739	0.045	2.67E-02	3.19E-02	0.180	0.142	2.840	1.400
BNBns	0.733	0.052	1.68E-02	2.18E-02	0.127	0.136	1.787	0.991
DT	0.671	0.028	6.66E-03	8.82E-03	-0.002	0.121	0.709	-0.013
DTns	0.668	0.030	1.16E-02	2.45E-02	0.060	0.150	1.230	0.467
RF	0.717	0.034	2.27E-02	2.48E-02	0.159	0.118	2.419	1.235
RFns	0.718	0.041	5.19E-03	4.94E-03	0.081	0.045	0.552	0.633
NN	0.739	0.053	2.49E-02	3.50E-02	0.157	0.154	2.651	1.218
NNns	0.729	0.057	1.04E-02	1.21E-02	0.091	0.103	1.106	0.705

Table 4.6: Results for the one-hot encoded version of the German Credit Data dataset.

method	**ACC**		**NPI**		**CVS**		**NPI Ratio**	**CVS Ratio**
	mean	**std dev**	**mean**	**std dev**	**mean**	**std dev**		
LR	0.774	0.061	2.36E-02	2.62E-02	0.151	0.123	2.514	1.174
LRns	0.775	0.053	1.11E-02	9.95E-03	0.099	0.081	1.179	0.766
PR $\eta = 5$	0.760	0.046	2.68E-02	3.42E-02	0.111	0.187	2.854	0.867
PR $\eta = 15$	0.758	0.046	2.54E-02	3.53E-02	0.112	0.182	2.700	0.869
PR $\eta = 30$	0.758	0.046	3.02E-02	3.30E-02	0.086	0.218	3.218	0.669
NB	0.763	0.034	3.08E-02	3.43E-02	0.191	0.133	3.281	1.486
NBns	0.763	0.043	1.07E-02	1.48E-02	0.084	0.112	1.135	0.654
CV2NB	0.734	0.046	1.62E-02	2.16E-02	-0.014	0.171	1.722	0.112
PR $\eta = 0$	0.760	0.047	3.20E-02	3.82E-02	0.165	0.166	3.405	1.286
PR $\eta = 10$	0.759	0.046	2.53E-02	3.54E-02	0.113	0.180	2.691	0.878
PR $\eta = 20$	0.763	0.044	2.68E-02	3.43E-02	0.103	0.192	2.851	0.801
GNB	0.732	0.060	8.28E-02	2.79E-02	0.401	0.064	8.813	3.120
GNBns	0.724	0.053	1.27E-02	9.51E-03	0.146	0.071	1.353	1.134
DT	0.662	0.020	1.81E-02	1.88E-02	0.119	0.151	1.927	0.925
DTns	0.670	0.021	9.62E-03	3.70E-03	0.071	0.122	1.024	0.555
RF	0.700	0.066	2.52E-02	3.56E-02	0.149	0.152	2.680	1.159
RFns	0.723	0.053	1.33E-02	1.55E-02	0.069	0.141	1.414	0.537
NN	0.767	0.053	1.79E-02	1.69E-02	0.148	0.087	1.908	1.151
NNns	0.761	0.042	1.67E-02	1.93E-02	0.069	0.167	1.777	0.540

Table 4.7: Results for the integer encoded version of the German Credit Data dataset.

The remarks made about the fairness metrics for the Adult Income dataset are also applicable to the German Credit Data dataset. Additionally, if we were to consider the NPI ratio when analysing the one-hot encoded version of the dataset, we would conclude that only the decision tree model trained with the sensitive attribute and the random forest trained without the sensitive attribute could reduce the unfairness found in the dataset. However, when performing this analysis with the CVS ratio, we would conclude that, except for the logistic regression model, removing the sensitive attribute would be enough to reduce the unfairness originally found in the dataset.

As far as the integer encoded version of the dataset is concerned, the NPI ratio appears to indicate that the unfairness found in the predictions made by all models increased with respect to the unfairness found in the dataset. On the contrary, the CVS ratio seems to indicate that by simply removing the sensitive attribute from the models it was possible to reduce the unfairness found in the dataset, except for the Gaussian Naive Bayes model.

As was observed for the Adult Income dataset, removing the sensitive attribute from the models seems to lead, in general, to improvements in fairness, regardless of the fairness metric used for the evaluation. For the German Credit Data dataset, only one exception is observed: when using the one-hot encoded version of the dataset, removing the sensitive attribute from the decision tree model leads to more discriminatory predictions than when the model takes that feature into consideration. This may be due to the existence of indirect unfairness, which becomes more apparent when this attribute is removed. However, we still need to further explore this hypothesis, in order to draw stronger conclusions.

Regarding the implementations of the Naive Bayes models, similar conclusions as those drawn for the Adult Income dataset can be drawn, even if the differences observed between the Bernoulli Naive Bayes and the implementation of the authors seem to be more slightly more expressive when considering the German Credit Data dataset.

The η parameter seems to have a more erratic influence when the method proposed by [21] is applied to the German Credit Data dataset. However, contrary to what was observed with the Adult Income dataset, this parameter seems to have some marginal influence when the integer encoded version of the dataset is used.

Although we have already performed experiments with different versions of the datasets, the influence of the features' encoding in the fairness of ML models needs to be further investigated before any conclusions can be drawn.

The discrepancies found between the two fairness metrics that were considered in these experiments need to be further analysed and discussed.

We also want to better understand why removing the sensitive attribute prior to training the models sometimes leads to more discriminatory decisions than when considering that attribute. This was sometimes the case with decision trees and random forests. Since these are white-box models, we want to interpret the structure of the produced trees to try to find a possible explanation for this scenario.

Regarding the method proposed by [21], a more detailed analysis of the influence of the η parameter might also need to be performed. While for the integer encoded version of the Adult Income dataset it appeared to have no influence in the results, for both versions of the German Credit Data dataset it showed quite an unstable behaviour.

Chapter 5

Work Plan

This chapter details the scheduled plan for the second semester as a continuation to the work developed during the first semester. A high-level plan of the tasks to be performed is presented in Figure 5.1.

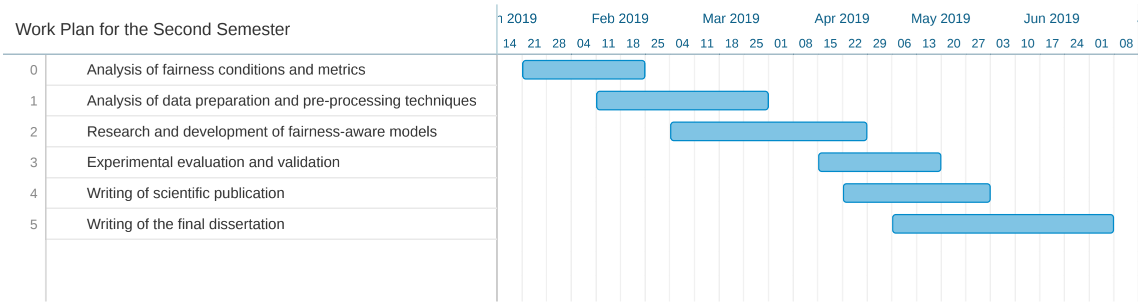


Figure 5.1: Scheduled plan for the second semester.

The first task will be to complete the analysis of the fairness conditions and metrics already started during the first semester. This includes a more in-depth analysis of the discrepancies found between the metrics used in the preliminary experiments.

We will then extend the analysis of the impact of different data preparation and pre-processing techniques on the system's fairness. We will consider different versions of the same dataset, resulting from the application of different encodings and transformations to the features. Further analysis on the effect of removing sensitive attributes from the datasets, prior to training the models, will also be performed.

The research and development of methods to mitigate unfairness in ANNs is estimated to be the task which requires the most effort to be completed. It includes studying the feasibility of the proposed methods, followed by their specification and implementation.

The experimental evaluation and validation of the proposed methods should not only take fairness into account, but also consider the trade-off with the predictive performance of the models. Furthermore, a comparison should be made with the unfairness originally found in the datasets.

The writing of a scientific publication and the final dissertation will take place in parallel with the previous tasks.

This page is intentionally left blank.

Chapter 6

Conclusion

The widespread usage of Machine Learning to help making decisions that have a significant impact on our lives raises several concerns related to discrimination and fairness. The main goal of this dissertation is to research and design methods to aid in the development of models and systems which can make fairer predictions.

To accomplish this goal, we started by collecting and analysing several fairness conditions and metrics used in previous works on the topic, some of which with a legal background for their definition. This analysis is crucial to understand their strengths and limitations and to select suitable fairness metrics for our evaluation. We also collected and analysed some representative datasets used in the literature.

Our preliminary experiments considered different versions of the datasets, based on the encodings used for the features. They revealed an unexpected behaviour of certain algorithms that we ought to further explore. Namely, we want to better understand why removing the sensitive attribute prior to training decision trees and random forests sometimes leads to more discriminatory results than if that attribute is used by the models. By analysing the structure of the resulting trees, we believe we might find an explanation.

Following the work of the first semester, we will analyse and evaluate how different techniques applied to data in the pre-processing stages of the pipeline influence the fairness of the systems. Additionally, we want to focus on the development of methods that can be incorporated in Artificial Neural Networks to improve the fairness in these models. We want to study the feasibility of introducing changes to the loss function, as done in the work we reproduced during the first semester, or developing a new type of layer with the goal of performing a regularization-like operation pertaining fairness.

In real-world scenarios some business objectives still need to be met, even if fairness concerns are taken into consideration. Therefore, we will always try to evaluate the trade-off between the fairness and performance of the models in the main predictive task. We should also evaluate the extent of the improvements introduced by the proposed methods with respect to the unfairness found in the datasets.

This page is intentionally left blank.

References

[1] ACM. Joint Statement on Algorithmic Transparency and Accountability by USACM and EUACM, May 25, 2017. Found at: https://www.acm.org/binaries/content/assets/public-policy/2017_joint_statement_algorithms.pdf.

[2] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner, ProPublica. (2016, May 26). Machine Bias. Taken from: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

[3] Nuno Antunes, Leandro Balby, Flavio Figueiredo, Nuno Lourenco, Wagner Meira, and Walter Santos. 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)*, pages 188–193, Luxembourg, 2018. IEEE.

[4] Reuben Binns. Fairness in machine learning: Lessons from political philosophy. In *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*, pages 149–159, 2018.

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

[6] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, Sep 2010.

[7] L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. *CoRR*, abs/1806.06055, 2018.

[8] IBM Corporation. AI Fairness 360 Open Source Toolkit – An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. Found at: http://aif360.mybluemix.net/.

[9] Hal Daumé III. A course in machine learning. *Publisher, ciml. info*, pages 5–73, 2012.

[10] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[11] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 214–226, 2012.

[12] European Parliament. General Data Protection Regulation, 2016.

[13] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[14] Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 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*, pages 259–268, 2015.

[15] 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.

[16] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[17] 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, December 5-10, 2016, Barcelona, Spain*, pages 3315–3323, 2016.

[18] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. *The elements of statistical learning: data mining, inference, and prediction, 2nd Edition.* Springer series in statistics. Springer, 2009.

[19] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: With Applications in R.* Springer Publishing Company, Incorporated, 2014.

[20] F. Kamiran and T. Calders. Classifying without discriminating. In *2009 2nd International Conference on Computer, Control and Communication*, pages 1–6, Feb 2009.

[21] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Machine Learning and Knowledge Discovery in Databases*, pages 35–50, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[22] Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. (2016, May 23). How We Analyzed the COMPAS Recidivism Algorithm. Taken from: https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm.

[23] Kristian Lum and James Johndrow. A statistical framework for fair predictive algorithms. *arXiv e-prints*, page arXiv:1610.08077, October 2016.

[24] Stephen Marsland. *Machine Learning: An Algorithmic Perspective, Second Edition.* Chapman & Hall/CRC, 2nd edition, 2014.

[25] Cecilia Muñoz, Megan Smith, and Dj Patil. United States – White House Office. Big Data: A Report on Algorithmic Systems, Opportunity, and Civil Rights, May 2016. Found at: https://www.hsdl.org/?view&did=792977.

[26] Supreme Court of the United States. Griggs v. Duke Power Co., 401 U.S. 424 (1971).

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau,

M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[28] David Martin Powers. Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[29] Andrea Romei and Salvatore Ruggieri. A multidisciplinary survey on discrimination analysis. *Knowledge Eng. Review*, 29(5):582–638, 2014.

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

[31] Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P. Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. 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*, KDD '18, pages 2239–2248, New York, NY, USA, 2018. ACM.

[32] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning - an introduction.* Adaptive computation and machine learning. MIT Press, 1998.

[33] Christina Wadsworth, Francesca Vera, and Chris Piech. Achieving fairness through adversarial learning: an application to recidivism prediction. *CoRR*, abs/1807.00199, 2018.

[34] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. FairGAN: Fairness-aware Generative Adversarial Networks. *CoRR*, abs/1805.11202, 2018.

[35] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. 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*, pages 1171–1180, 2017.

[36] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, pages 962–970, 2017.

[37] Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 325–333, 2013.

[38] 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 2018, New Orleans, LA, USA, February 02-03, 2018*, pages 335–340, 2018.

This page is intentionally left blank.

Appendices

This page is intentionally left blank.

Appendix A

Pairwise Mutual Information and Pearson Correlation

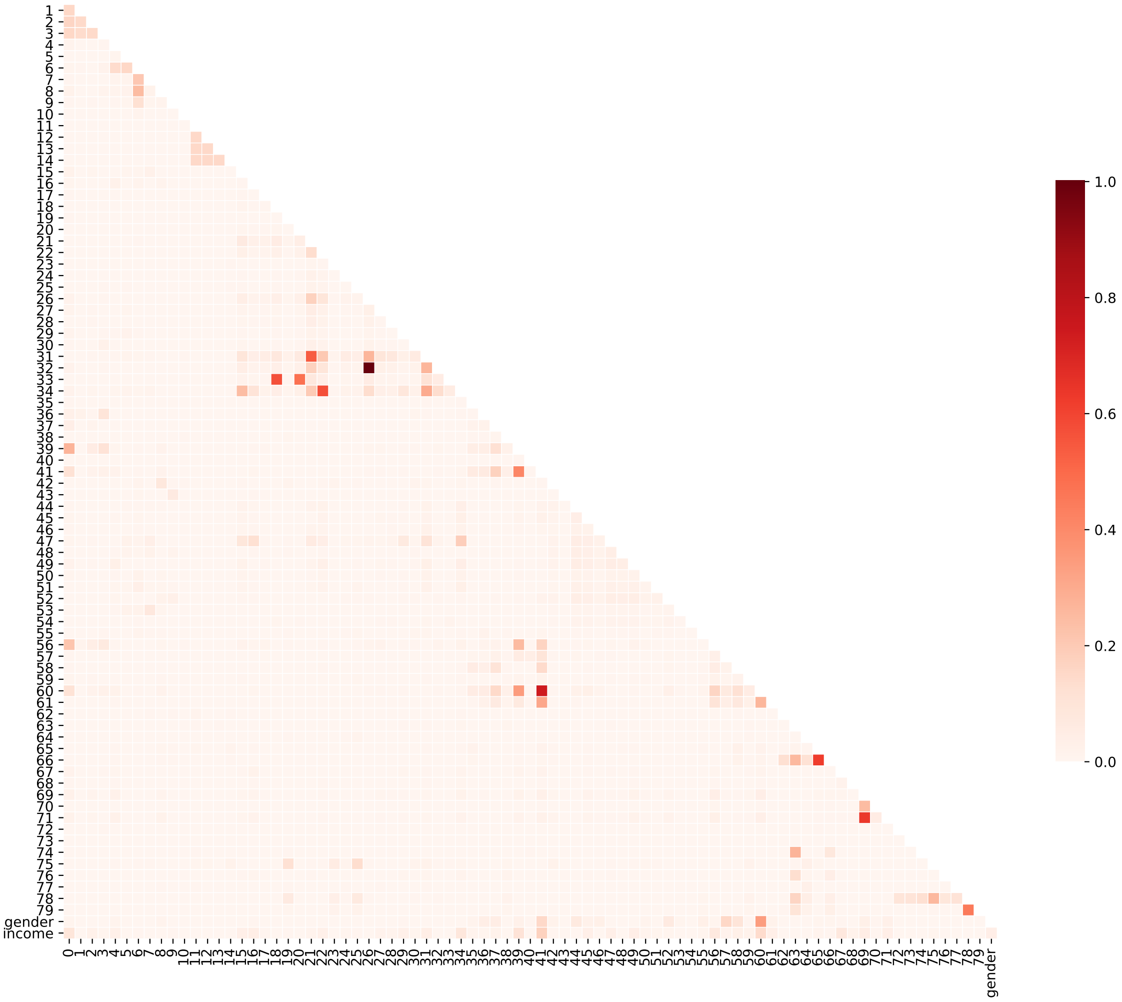


Figure A.1: One-hot encoded version of the Adult Income dataset - pairwise mutual information between features.

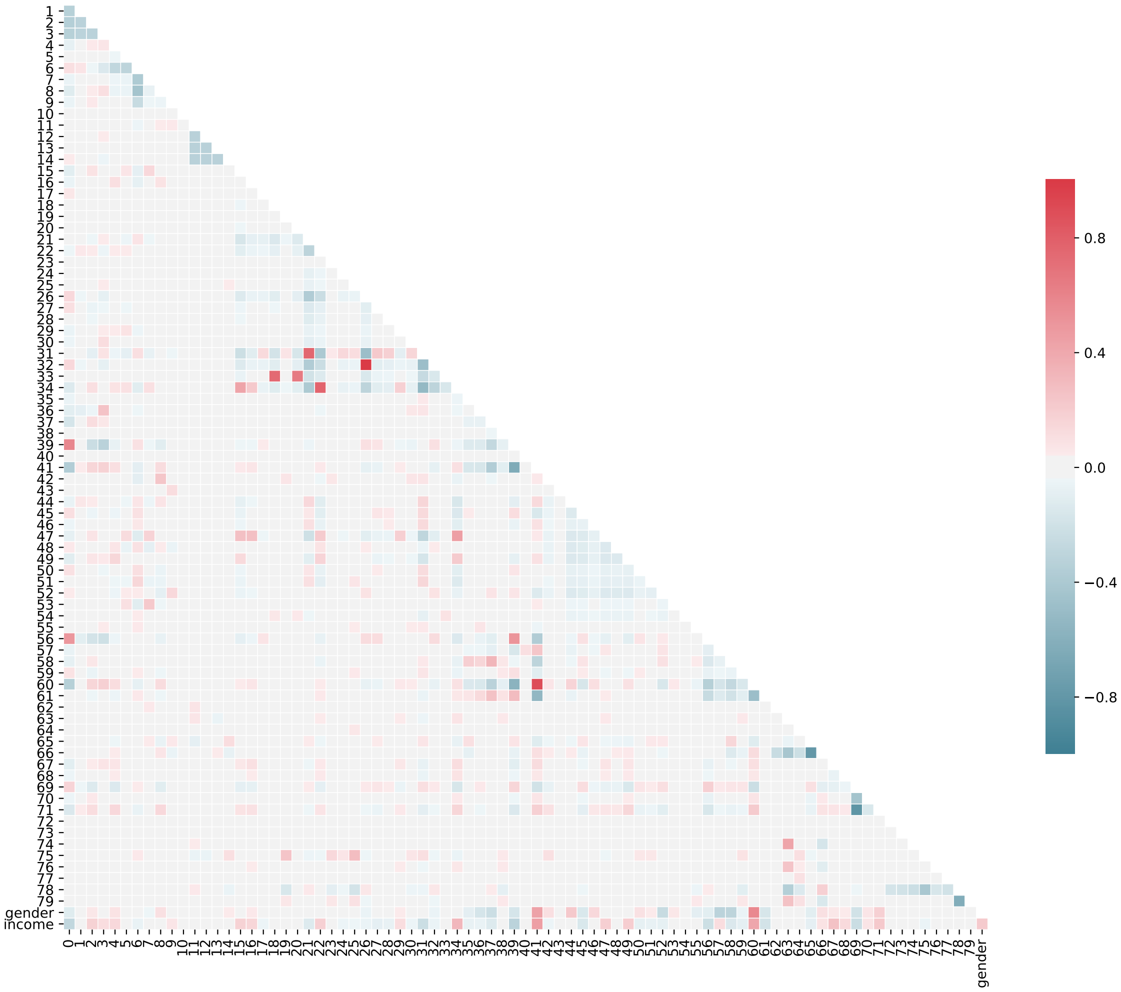


Figure A.2: One-hot encoded version of the Adult Income dataset - pairwise Pearson correlation between features.

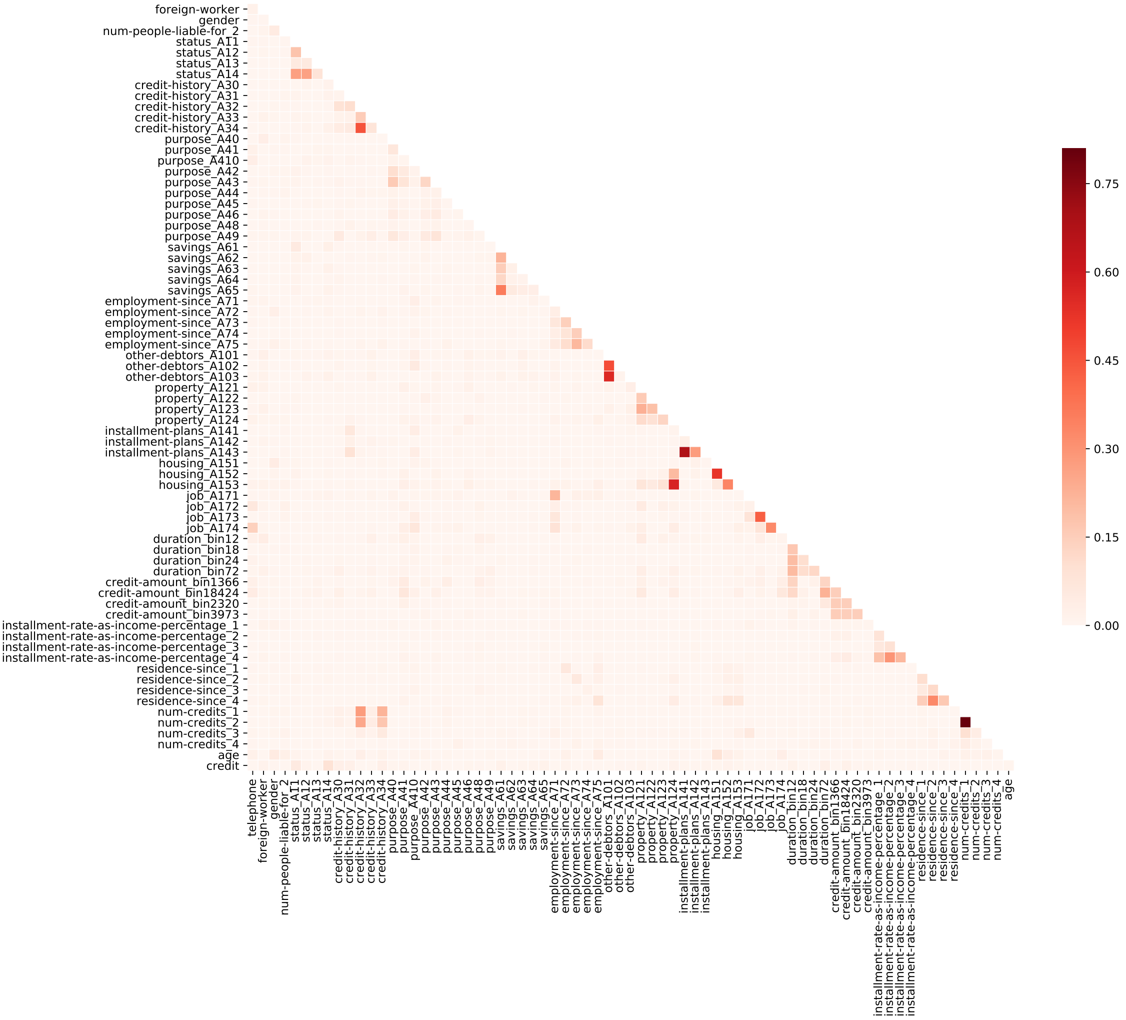


Figure A.3: One-hot encoded version of the German Credit Data dataset - pairwise mutual information between features.

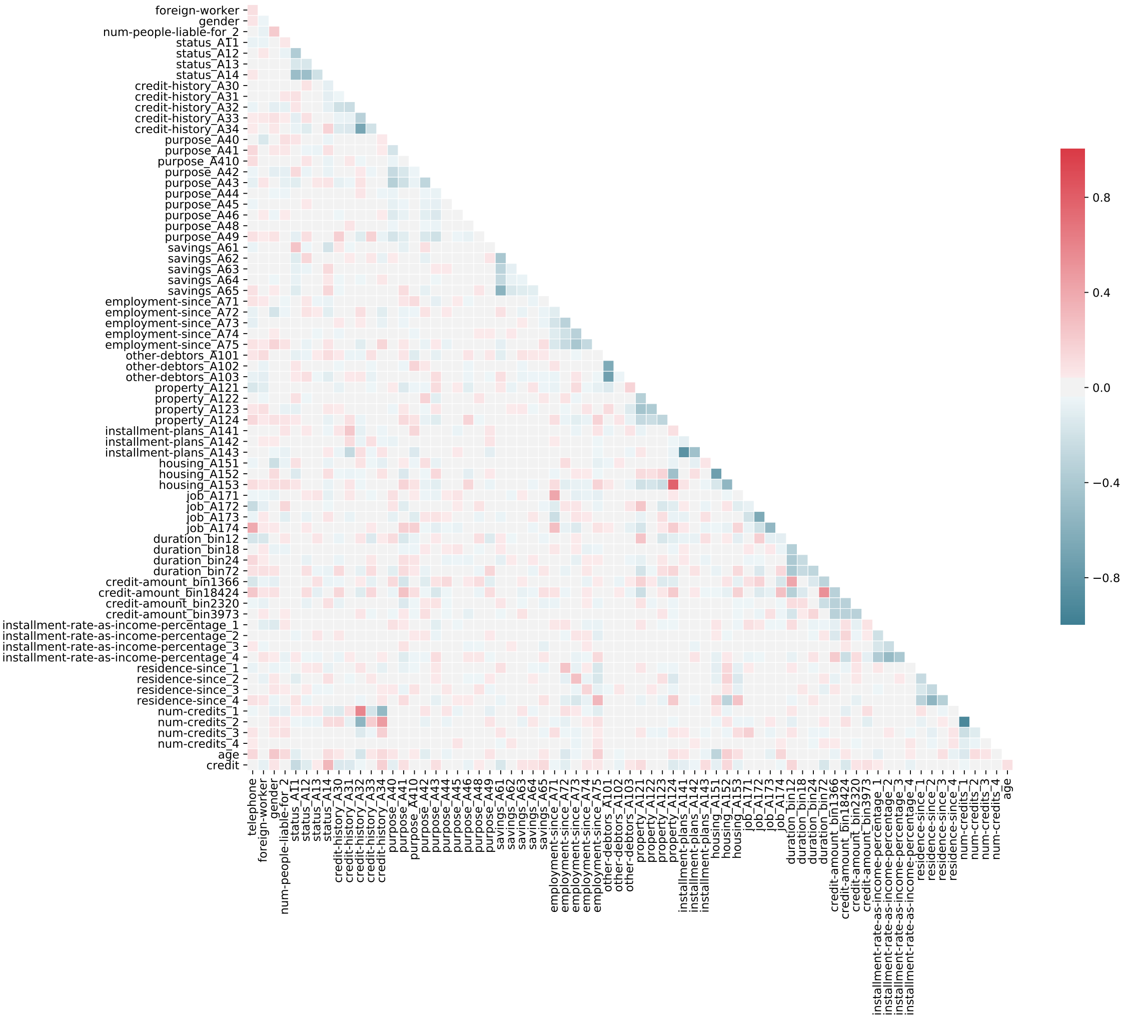


Figure A.4: One-hot encoded version of the German Credit Data dataset - pairwise Pearson correlation between features.