VRIJE
UNIVERSITEIT
BRUSSEL

Graduation thesis submitted in partial fulfilment of the requirements for the degree of Bachelor of Science in de Computerwetenschappen

# A GENERAL PURPOSE FRAMEWORK FOR FAIRNESS IN JOB HIRING

Sam Vanspringel

June 2023

Promotors: prof. dr. Pieter Libin, prof. dr. Ann Nowé
Advisor: Ioana Alexandra Cimpean

**Sciences and Bioengineering Sciences**

VRIJE
UNIVERSITEIT
BRUSSEL

Proefschrift ingediend met het oog op het behalen van de graad van Bachelor of Science in de Computerwetenschappen

# EEN ALGEMEEN GESCHIKT FRAMEWORK VOOR FAIRNESS IN TEWERKSTELLING

Sam Vanspringel

Juni 2023

Promotors: prof. dr. Pieter Libin, prof. dr. Ann Nowé
Advisor: Ioana Alexandra Cimpean

**Wetenschappen en Bio-ingenieurswetenschappen**

**Abstract**

Due to the increasing popularity of machine learning algorithms to automate decision-making, attentiveness with regard to fairness implications is crucial. Therefore, we require a methodology to detect and mitigate bias and unfairness in algorithms. We survey existing fairness notions and their applicability in the context of job hiring scenarios. Further, we investigate a selection of pre- and post-processing techniques to mitigate potential bias. We propose a framework capable of dealing with multiple fairness and bias requirements based on distinct problem settings. Using our framework, we investigate a series of job hiring scenarios based on realistic populations and highlight unfairness for different biases.

# 1 Introduction

Whether algorithms are able to treat people fairly and objectively becomes increasingly important each day. As algorithms play a critical role when making decisions that affect people. To avoid discriminating against minorities or even specific individuals, it is necessary to be able to detect and mitigate potential bias. To this end, we build a framework that can quantify bias using fairness notions and mitigate it.

The term fairness cannot be used unambiguously. Different scenarios require different fairness notions, depending on distinct criteria [16]. Fairness notions show how fair groups or individuals were treated. A criterion for deciding the right fairness notion is the context of the problem setting. E.g., the fairness of rejecting a loan applicant should be computed differently than the fairness for criminal risk assessment. Therefore, selecting an appropriate fairness notion requires insights from a domain-expert. Another criterion for using a certain fairness notion can be the availability of the ground truth. This means the availability of an objectively right answer. When hiring people for a job, there is no ground truth available for rejected applicants, while the ground truth for qualified applicants may be biased. Because the decision is inferred from human decisions, it is subjective in nature.

In job hiring, it is important to be objective and fair to all candidates to determine who is the best suited for the job. As machine learning-based decision-making systems (MLDMs) are becoming popular, we need a way to assess these systems on their fairness. Machine learning is a way to let an algorithm learn from a dataset and infer patterns [19]. Compared to humans, it can perform certain tasks better and faster. An MLDM is able to process a large amount of data, which makes it able to base its decision in a more objective manner than humans can. Additionally, it can process this data in a shorter timespan than a group of people could. This makes it a useful tool in scenarios where making quick and informed decisions is crucial.

When finalising this learning step, the algorithm is able to make predictions for a new dataset based on the task it has learned [19]. However, by learning from past data, the algorithm might pick up on patterns that are biased and should therefore not be used. For example, if the dataset that the model is learning from contains a pattern of discrimination, the model might lead to undesired results. Because their goal is to make an as accurate decision as possible, machine learning models will try to recognise a pattern whether it is discriminating or not. Consequently, it is likely to apply the same discriminatory reasoning on future examples and thus make unfair judgements. Imagine a dataset from 1964 with men and women who are applicants for a high-level job. At this time, it was more common for men to be higher educated than women [10, 30]. If a machine learning model learns from this data, it is likely to pick up on the pattern that more men are chosen than women and starts using this in its own decision process. However, if this model is used for making the same choice in a dataset where men and women are equally educated, it might still discriminate by continuing to pick men over women.

Fairness has been previously studied in machine learning [35, 9, 8]. Moreover, connections to biased decisions and discrimination have been considered [17, 2]. In data mining, techniques focused on bias mitigation enforce fairness notions at different times during the model development. If unfairness is present in the training dataset, pre-processing techniques aim to mitigate the bias before training starts by removing bias from the dataset itself [7, 4]. A possible solution is re-weighing the samples in the dataset to reduce potential bias effects [11]. However, this means that the data must be available. In contrast, in-processing techniques attempt to constrain the

algorithm itself to conform to fairness requirements. Typically, the learning process is adapted to include fairness constraints such as regularization terms [15, 32]. At last, post-processing techniques provide a solution when neither the dataset nor the algorithms can be modified, as they focus on changing the model's predictions to satisfy fairness constraints [13, 31].

# 2 Problem definition

We explore algorithms in the context of the real-world problem of job hiring, in which treating each applicant objectively is crucial. As hiring processes are repeated numerous times in a company's lifetime, this results in plenty of candidate evaluations that machine learning algorithms can learn from. However, using this data could have undesired effects. Encoded bias in these datasets could cause the machine learning algorithm to reproduce this judgment bias on new candidates. Unnecessarily filtering out potentially valuable candidates early on hurts the company and is unfair towards the applicants, considering hiring is a process of multiple steps. This leads to the objective of designing a clear and intuitive application framework for detecting bias and unfairness when using machine learning. This way the effect of bias on the decision-making of machine learning algorithms can be illustrated. Furthermore, such a framework shows how this bias and unfairness can be mitigated, to ensure a fair and objective decision algorithm. We investigate existing fairness notions and selection criteria to explore the applicability of fairness notions in job hiring. Based on the results, we explore mitigating methods such as pre- and post-processing techniques to maximize fairness. These insights will provide a guideline on how to make the application more intuitive and clear to users, to detect bias present in the dataset or algorithm, as well as to provide suggestions on how to mitigate this discrimination and improve the model. This results in a framework that applies several fairness notions to any given dataset and mitigates potential bias in these datasets. This framework can be used in real life scenarios to illustrate the consequences of bias when using machine learning algorithms in decision-making. Moreover, the framework provides the necessary means to use such algorithms in a safer and more objective way. The application framework ensures that the resulting decisions are fair towards groups as well as individuals and can be calibrated appropriately.

# 3 Background

We consider a job hiring scenario in which machine learning models run the risk of learning discriminative behaviour. Specifically, we investigate a series of applicants who are deemed qualified for the job based on certain features. The behavioural difference of the model is then examined and analysed for groups or individuals. The distinction between groups will be decided on *sensitive features* [16]. Sensitive features are characteristics that should not influence the decision whether the applicant is considered (qualified or not), but are historically susceptible to discrimination. For example, women were historically more disadvantaged compared to men when it came to the outcome of their education [10]. In this section, we provide some background on additional concepts used in the framework.

## 3.1 Machine Learning

We discuss the machine learning algorithms used in our framework. These algorithms will construct a model or learn using the majority of the candidates, which is referred to as the training set. When the model has learned this part, it can be used to make predictions for new candidates.

To evaluate the performance of the model, we use the remainder of the candidates (i.e., the test set).

### 3.1.1 Decision Tree

Decision trees are suited for classifying instances based on a fixed set of attributes [19, 24]. Decision trees are popular as they are easily interpretable. A decision tree is inferred from the training dataset. First, a statistical test is used to determine which attribute should be used in the root node. For each possible value of this root node attribute, a descendant is created. The process is then repeated for each subset of samples that have the descendants value for the root attribute. This is repeated until a sample can be classified as an output value. When the decision tree is constructed, it is possible that *overfitting* happens. This means that the decision tree is able to perfectly predict instances of the training set, but is unable to generalize to unseen samples. To avoid overfitting the tree on the training set and improve performance on the test set, *pruning* can be applied [23, 18]. This is a common technique to remove branches from the tree.

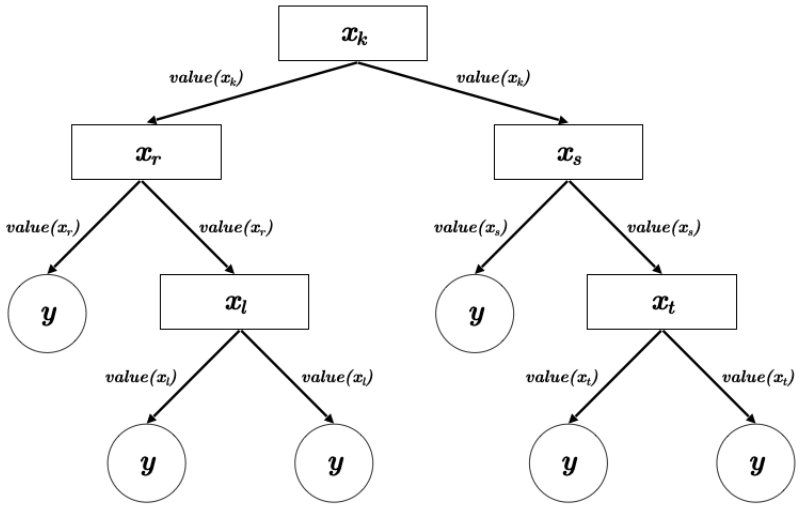



Figure 1: Decision tree structure

Figure 1 depicts a possible decision tree model. The decision nodes (rectangle) are the ones who decide in which leaf node (circle) an instance will end up. They represent a certain attribute $x_j$ from a sample $x$ of the dataset. The leaf nodes represent the outcome $y$ of the corresponding samples for each of the leaf nodes. When complete, the decision tree can classify new samples in a dataset. Classifying an instance starts with checking the value of the root node attribute. The instance is then passed on according to that value. This process repeats itself until the instance reaches a leaf node. The value of this leaf node corresponds to the prediction $\hat{y}$. The ID3 method for constructing decision trees top-down is provided in Algorithm 1 [19].

**Algorithm 1** ID3(*Examples*, *Target*, *Features*)

Create a *Root* node for the tree

**if** All *Examples* are positive **then**
    **Return** single-node tree *Root* with a positive label

**else if** All *Examples* are negative **then**
    **Return** single-node tree *Root* with a negative label

**else if** *Features* is empty **then**
    **Return** single-node tree *Root* with the most common label of *Target*

**else**
    $F \leftarrow$ feature from *Features* that best classifies *Examples*
    The decision attribute for *Root* $\leftarrow F$
    **for** All possible values $v_i$ for F **do**
        Add new branch below *Root* for test $F = v_i$
        $Examples_{vi}$ is the subset of *Examples* with $F = v_i$
        **if** $Examples_{vi}$ is empty **then**
            Add leaf node below branch with the most common value of
            *Target* in *Examples* as label
        **else**
            Add subtree ID3($Examples_{vi}$, *Target*, $Features - \{A\}$)
        **end if**
    **end for**
**end if**

**Return** *Root*

Our framework uses the default DecisionTreeClassifier from the scikit-learn library as an implementation of a decision tree [21].

#### 3.1.2 *k*-Nearest Neighbour learning

$k$-Nearest neighbour learning is an instance-based learning method [19]. This means that it does not construct a general internal model. Instead, it considers all instances of a training set part of the $n$-dimensional space $\mathbb{R}^n$. If two arbitrary instances $\vec{x}$ and $\vec{z}$ of a dataset are described by

$$(x_1, x_2, ..., x_n) \text{ and } (z_1, z_2, ..., z_n)$$

then the distance $d(\vec{x}, \vec{z})$ between instance $\vec{x}$ and instance $\vec{z}$ is calculated to determine their similarity in order to find their nearest neighbours. Note that the chosen distance metric has an impact on how the nearest neighbours are determined. In this work, we assume the Euclidean distance:

$$d(\vec{x}, \vec{z}) = \sqrt{\sum_{k=1}^{n} (|x_k| - |z_k|)^2}$$

The $k$-Nearest neighbour classifies an instance by calculating the distance to the other instances. Therefore, a new instance gets the outcome that is the most common among the closest $k$-neighbours. The chosen value of $k$ is data-dependent. The classification of instances is shown

in Algorithm 2[19]. In our framework, we use the KNeighborsClassifier from scikit-learn that considers the three closest neighbours [21].

**Algorithm 2** KNN-classification($Training\_instances$)

Instance $x_q$ to be classified

$(x_1, f(x_1)), ..., (x_k, f(x_k))$ denote the k instances from
$Training\_instances$ with their outcomes.

**Return** prediction $\hat{f}(x_q) \leftarrow \text{argmax}_{v \in V}(\sum_{i=1}^{k} \delta(v, f(x_i))$

With $\delta(a, b) = 1$ if $a = b$ and $\delta(a, b) = 0$ otherwise

### 3.1.3 Confusion Matrix

Machine learning algorithms optimise on the training set. This means that the test set is unknown territory for the model. When a new instance has no comparable one in the training set, the algorithm may not be able to correctly classify it, resulting in incorrect predictions. Assuming we know the true outcome of the testing samples and $y$ represents a positive or a negative prediction, we can illustrate these mistakes using a confusion matrix. In our job hiring scenario, a positive outcome represents that the applicant is suitable for hiring or qualified, while a negative means the applicant should be rejected. Table 1 depicts the confusion matrix regarding binary outcomes and predictions.

| Actual | Negative | TN | FP |
|---|---|---|---|
| | Positive | FN | TP |
| | | Negative | Positive |
| | | Predicted | |

Table 1: Confusion matrix

We highlight the different quadrants of the confusion matrix and present them in the context of our job hiring scenario.

- **TN:** The *true negatives*. These instances were predicted negative when the actual value was negative as well. In job hiring, this means that an applicant is predicted rejected, and the true value indicates the same.

$$y : \text{Not qualified} \wedge \hat{y} : \text{Not qualified}$$

- **FP:** These instances are mistakes called *false positives*. Which means the actual value was negative, but the machine learning model classified them as positive. This corresponds to an applicant being predicted qualified when they should have been classified as not qualified.

$$y : \text{Not qualified} \wedge \hat{y} : \text{Qualified}$$

- **FN:** Just as the false positives, these are mistakes as well. *False negatives* are instances classified as negative when the actual value was positive. In the case of job hiring, the applicant is predicted qualified when the true value indicates they should not be qualified.

$$y : \text{Qualified} \wedge \hat{y} : \text{Not qualified}$$

- **TP:** The *true positives*. True positives are instances that were classified as true and predicted true as well. In job hiring context, the applicant is predicted qualified when the true value indicates they should be qualified as well.

$$y : \text{Qualified} \wedge \hat{y} : \text{Qualified}$$

These elements of the confusion matrix can be used to compute additional properties of predictions:

- **Accuracy:** The accuracy denotes the proportion of instances that were predicted correctly by the model.

$$\text{Accuracy} = (TN + TP)/(TN + FP + FN + TP)$$

- **Precision:** The precision is the proportion of correctly predicted positive instances to all the predicted positive instances.

$$\text{Precision} = TP/(TP + FP)$$

- **Recall:** Recall is the ratio of correctly predicted positive instances to all actual positive instances. The recall can also be referred to as the true positive rate (TPR).

$$\text{Recall} = TP/(TP + FN)$$

## 3.2 Fairness

Fairness in job hiring can be described as the degree in which a machine learning algorithm can judge applicants or groups of applicants objectively. As it is possible that a model judges subjectively and tries to avoid hiring people with certain sensitive features, we use fairness notions to quantify the ability of the model to make fair predictions [16]. The applicability and suitability of fairness notions is dependent on the context [9]. We identify fairness notions based on two main categories: group fairness and individual fairness. The goal of group fairness notions is to make sure groups of people who share one or more sensitive features are treated equally. In contrast, notions targeting individuals check that an individual is not discriminated against based on one or more sensitive features.

The drawback of group fairness notions is that they only focus on a particular feature or set of features when computing the fairness. Even when a group fairness notion is satisfied, it is possible that an individual of that group was discriminated against [5]. Imagine a dataset of applicants for a job, where a group fairness notion measures roughly the same quantity for women as for men. This means that the groups of men and women were treated equally in this decision. However, it is possible that discrimination against a woman with dark skin went unnoticed, because the fairness notion did not take a combination of features into account but instead focused on the larger group of all women only. For this reason, individual fairness notions are necessary.

Achieving total fairness is difficult, possibly intractable, as it requires that a combination of fairness notions are simultaneously satisfied [3]. Moreover, the fact that a certain fairness notion is satisfied does not mean another one is. This will be illustrated in Section 5.

### 3.2.1 Selection criteria

This section describes the most important criteria to decide which fairness notion should be used in a specific situation [16]. We discuss them in context of the job hiring scenario.

**Ground truth:** When the ground truth is available, the outcomes for a dataset are objectively true. When it is not available, it means that the outcomes of the dataset are unknown or were decided subjectively. In the job hiring scenario, this would mean that the applicants' outcome of being qualified is decided by a human.

**Unreliable outcome:** This criterion requires that the outcomes of the dataset are inferred by a subjective source. When no ground truth is available, the outcomes of the dataset should be considered unreliable. This means that a dataset with unreliable outcome should use a fairness notion that does not consider the actual values corresponding to the ground truth. In the job hiring scenario, we consider datasets that score applicants in an objective way as ground truth. When datasets contain bias, they should not be used as ground truth.

**Emphasis on recall:** The emphasis on the positive instances of the prediction can be either on precision or recall (Section 3.1.3). When the emphasis is on precision, it is more important how precise a positive prediction is. In a scenario where the outcome concerns which employee should be fired, the emphasis would be on precision. Consequently, fairness notions would be more sensitive to incorrectly firing an employee [16]. In the job hiring scenario, the emphasis is on recall. Concretely, we are more interested in maximising the amount of qualified people who are correctly predicted qualified.

**Fixed or floating threshold:** When deciding whether someone meets the requirements to be labelled positive, a threshold is used. A fixed threshold is one that does not change according to circumstances. For example, in loan granting, a floating threshold may depend on the current economic state of the country and can therefore fluctuate over time. In the job hiring scenario, we assume a fixed threshold. All applicants that receive an outcome higher than the threshold are classified as qualified, otherwise they are considered not qualified.

**Likelihood of intersectionality:** This criterion is most important in individual fairness. When intersectionality occurs, it means that someone can be discriminated based on multiple features at once. When this is more likely to happen, an individual fairness notion should be used. An example of this is a foreign woman, who could be discriminated against gender and nationality.

### 3.2.2 Group fairness notions

Group fairness notions can be used to see if two or more groups of people are judged equally. In the context of our job hiring setting, we group applicants together based on their gender. Additionally, we discuss a selection of group fairness notions that are suitable to be used in a job hiring scenario [16]. To compute these notions, we use the different elements of the confusion matrix from Table 1.

**Statistical parity:** When the statistical parity of two groups is equal, it means that both groups have an equal acceptance rate [16]. The following equation defines a satisfaction of statistical parity:

$$P(\hat{Y} = 1|G = 0) = P(\hat{Y} = 1|G = 1)$$

The formula denotes that the prediction of $\hat{Y}$ is statistically independent of the values for sensitive attributes $G$. In job hiring, this means that the probability of being predicted qualified should be the same for both women and men. Statistical parity can also be computed from the confusion matrix by requiring the following formula is equal for both groups:

$$(TP + FP)/(TP + FP + FN + TN)$$

This fairness notion is suitable for job hiring because the positive outcome (i.e., qualified) is preferred to the negative (i.e., not qualified). However, statistical parity is unsuitable when the ground truth is available. When satisfied, it ensures that the same proportion of people are predicted qualified from each group. Therefore, it does not consider the base rates or the percentage of people who are considered qualified in the actual outcomes. When statistical parity is satisfied, but the base rates are different, one group is still discriminated against. For example, it is possible that more women than men are objectively qualified for a job. When statistical parity is then satisfied, the women are still discriminated against because the base rate of the women is higher.

**Equal opportunity:** This fairness notion is a relaxation of the fairness notion equalized odds. Equalized odds requires the sensitive attribute to be conditionally independent of the outcome [16]. However, satisfying equalized odds is difficult to explain. Therefore, we consider two relaxed versions. The first relaxation is equal opportunity. Let $Y$ and $\hat{Y}$ be variables that represent, respectively, the actual and predicted outcome with 1 a positive and 0 a negative. Variables $G = 0$ and $G = 1$ indicate that samples belong to group 0 or 1 respectively, where the groups are defined based on one or more sensitive features. Equal opportunity is defined by the following equation:

$$P(\hat{Y} = 1|Y = 1, G = 0) = P(\hat{Y} = 1|Y = 1, G = 1)$$

In equal opportunity, the true positive rates (TPR) or sensitivity recall should be equal for both groups.

$$TPR = TP/(TP + FN)$$

This formula takes into account the false negatives of a prediction. Which means that in our job hiring scenario, it focusses more on the rejecting of qualified applicants as opposed to the hiring of unqualified candidates.

The suitability of equal opportunity for the job hiring scenario is decided by the focus on fairness of this work. It clearly is not desirable for a company to hire people who are unqualified. However, for a company, fairness with regard to the amount of qualified applicants who are qualified may be more important to consider instead of the amount of mistakenly qualified unsuitable individuals. This is under the assumption that the proportion of false negatives is the same for both groups. In job hiring, it is more critical for fairness that for both groups the same proportion of qualified people are rejected. For this reason, the applicability of equal opportunity depends on the meaning of a positive outcome. Rejecting more qualified applicants for a group would be discriminatory.

**Predictive equality:** The second relaxation of equalized odds that does take into account the false positives is predictive equality [16]. When false positives are important to the fairness of decisions, predictive equality is a better option than equal opportunity. Predictive equality is defined by the following formula:

$$P(\hat{Y} = 1|Y = 0, A = 0) = P(\hat{Y} = 1|Y = 0, A = 1)$$

Predictive equality demands the false positive rates (FPR) of two groups to be the same.

$$FPR = FP/(FP + TN)$$

In predictive equality, the focus is on the applicants who were predicted qualified and whose actual value was negative. In other words, this notion focuses on the hiring of unqualified applicants.

In job hiring, it is often the case that fairness regarding hiring unqualified people, is less critical than rejecting qualified applicants. Predictive equality is more appropriate for a scenario of firing employees. The reason for that is the negative interpretation of a positive outcome, i.e., getting fired. In such scenario, it is critical that the proportion of false positives are the same for both groups. Firing more well-performing people of one group would be discriminatory.

### 3.2.3 Individual fairness notions

This section discusses applicable fairness notions regarding individuals. As group fairness notions mostly look at one sensitive feature that people can be discriminated on, they ignore all other features. This way, discrimination on some other features or combinations of features could go unnoticed.

**Causal discrimination:** When intersectionality is possible, satisfying causal discrimination ensures that individuals are not discriminated against. In the case of two individuals having the exact same features apart from the sensitive one, causal discrimination expects them to have the same outcome [16]. In job hiring this would mean a man and woman who have the same feature values $X$, apart from the sensitive feature for gender $G$, get the same prediction.

$$X_{woman} = X_{man} \wedge G_{woman} \neq G_{man} \rightarrow \hat{y}_{woman} = \hat{y}_{man}$$

Causal discrimination is well-suited in scenarios where people often share the same features with other people. In our framework, causal discrimination is not suitable, as the occurrences of candidates sharing the exact same features apart from one sensitive feature are rare.

**Fairness through unawareness:** This is an individual fairness notion that is a generalization of causal discrimination. It states that similar individuals should have a similar prediction [16, 6]. We consider candidate $k$ and candidate $l$ with their feature vectors $v_k$ and, $v_l$ respectively. These vectors are associated with a probability distribution $M(v_k)$ and $M(v_l)$ over their outcome. Let $d(v_k, v_l)$ be the similarity distance between the candidates and $D(M(v_k), M(v_l))$ the distance between their outcome distributions. Fairness through unawareness is achieved when the following equation is satisfied for each couple of candidates $k$ and $l$:

$$D(M(v_k), M(v_l)) \leq d(v_k, v_l)$$

**Consistency score:** The consistency score is computed by checking the prediction of the $k$-nearest neighbours of a candidate (Section 3.1.2). For each candidate $i \in n$, the percentages of the nearest neighbours $N$ that have the same prediction is calculated. Taking the average of these values over all candidates results in the consistency score [33]. In our application, the complement of this score is used to illustrate the inconsistency in the dataset:

$$Inconsistency = \frac{1}{n} \sum_{i=1}^{n} |\hat{y}_i - \frac{1}{|N|} \sum_{j \in N} \hat{y}_j|$$

In this work, we use the consistency score as a measure of fairness through unawareness. We consider the 20 nearest neighbours for each candidate.

## 3.3 Mitigation

Upon detecting discrimination towards individuals or groups, mitigation methods can be applied. More specifically, pre-, in- and post-processing techniques can be used to alter the dataset or prediction, to mitigate potential bias. Pre- and post-processors work either before or after making a prediction. Using in-processing techniques to mitigate bias, a classifier learns from the training dataset while making sure that the predictions that it makes ensure fairness. In this section, we discuss a collection of mitigating methods to reduce the effects of bias on the decision-making of the algorithm.

### 3.3.1 Pre-processing

This subsection discusses pre-processing. Pre-processing changes the dataset before training a machine learning algorithm. It mitigates bias beforehand to make sure all candidates are treated fairly.

The pre-processing technique used in our framework is sample reweighing[12]. With sample reweighing, the dataset will be altered to minimise the effects of bias on the decision-making of the algorithm. In this approach, the outputs will not change, but each candidate in the dataset is assigned a weight. Now we discuss how the weights are computed. If our dataset contains no bias, the sensitive features and the outcome of a candidate should be conditionally independent. Encoded bias in the dataset results in a lower probability of having sensitive features with a disadvantaged value and being considered qualified. The goal of reweighing is compensating for this bias by assigning lower weights to candidates that were favoured or avoided. By adjusting these weights, the dataset can be transformed into being less biased.

### 3.3.2 Post-processing

Using in-processing techniques to mitigate bias, a classifier learns from the training dataset while making sure that the predictions that it makes ensure fairness. However, datasets and algorithms may be inaccessible to modify in machine learning. In this case, post-processing techniques can be used. In post-processing approaches, the machine learning algorithm predicts outcomes for the candidates and modifies the predictions afterwards to improve fairness.

**Calibrated equalized odds:** This post-processing technique tries to mitigate the equalized odds of the prediction. In this framework, calibrated equalized odds is focussed on equalising the false positive rates of the groups. This should result in a reduction of the difference in predictive equality [22]. Calibrated equalized odds will change the outcomes of candidates to ensure these conditions.

**Reject Option Classification:** This mitigation technique works by changing the outcomes of discriminated groups to positive and the outcomes of favoured groups to negative[14]. In our framework, the reject option classifier is focussed on lowering the difference in statistical parity between the privileged and unprivileged groups.

# 4 Related work

In this work, we focus on explicit bias that could be encoded in datasets. This means that the bias is based directly on sensitive features. We use suitable fairness notions to quantify this bias according to our specific scenario. However, explicit bias can be quantified in other ways such as fuzzy-rough uncertainty (FRU) as discussed by Nápoles and Koutsoviti Koumeri [20]. FRU is based on the change in decision-making when a sensitive feature is removed from the dataset. When an algorithm made fair decisions, this should cause very little changes.

Satisfying individual fairness is increasingly more important with the rising importance of diversity in companies. iFlipper provides a pre-processing technique that optimises the minimal flipping of labels in a dataset to improve individual fairness [34]. The algorithm works by reducing the amount of similar candidates with different labels to a given limit.

In the context of machine learning, AIF360 is an open-source library used to compute fairness notions and mitigate bias [1]. The library contains a variety of metrics to detect the amount of bias present in a dataset. Our framework uses the implementation of the AIF360 library for the aforementioned pre- and post-processing techniques.

Besides considering a job hiring scenario in supervised learning, it can also be studied in reinforcement learning. Schumann et al. discuss the importance of fairness when using machine learning in a modern hiring process [26]. The challenges of fair treatment of marginalised groups are investigated and incorporated in systems for modern day hiring. Schumann et al. had previously discussed an algorithm for allocating machine learning and in-person hiring steps in university admissions [25], while incorporating diversity as a focus point.

# 5 Experiments

We consider distinct scenarios for our job hiring setting, where we generate applicants in a simulator based on the Belgian population [28, 29, 27]. In these generated populations, a bias or different distribution can be implemented to suit our different scenarios. Each candidate receives a score based on their years of experience, age and the amount of degrees. Using this score, each applicant is classified as qualified or not qualified. The focus of these experiments is on discrimination based on gender, but we will also consider a scenario that takes the combination of gender and nationality into account.

We perform experiments for three different scenarios. First, a baseline scenario is implemented, where there is no bias and the distribution of men and women is based on the actual distribution of the Belgian population [28, 29, 27]. Next, we define a scenario with an uneven gender distribution, where the population consists of 70% men and 30% women. Finally, we study two cases of bias. In the first case, there is a bias implemented that favours men over women. In the second case, a bias towards Belgian men is added, such that we can discuss the individual fairness of the candidates. The feature distribution is again based on a real dataset of the Belgian government

[28, 29, 27].

To run an experiment, the simulator generates a dataset of 20.000 applicants for a job. When each candidate is labelled as qualified or not qualified, a machine learning model is trained on 90% of this dataset. The other 10% of the candidates will serve as a testing dataset to evaluate the model performance. In the first two scenarios, we can consider these simulated evaluations as a ground truth. When the dataset is biased, the samples cannot be regarded as ground truth. To compare the predictions of the trained models to the ground truth, we calculate the confusion matrix and additional properties (Section 3.1.3). Furthermore, we look at the proportions of the candidates predicted qualified, categorised on their sensitive features. Finally, for each scenario, the previously mentioned fairness notions are computed to see whether the models discriminate based on the sensitive features (Sections 3.2.2 & 3.2.3). To ensure the validity of these experiments, we repeat this process 100 times. For each computed measure, we plot the mean and the standard deviation.

## 5.1 Simulator

We make use of a job hiring simulator to sample job applicants based on the real Belgian population.[1] The simulator generates a number of samples who represent applicants with several features, as discussed below. The probabilities of a feature value are each time computed by taking a subset of the real dataset and calculating the actual probability of the value in this subset.

**gender (G):** This attribute represents the gender of the applicant. The possible values are male (0) and female (1).

**age (A):** The age of the applicant. The age is generated ranging from age 18 to 65 years.

**degree (D):** Binary attribute indicating if the applicant has obtained a relevant degree for the job.

**extra degree (E):** This is also a binary attribute like degree. In this case, the applicant can only have an extra degree if it has a degree to begin with.

**experience (X):** The experience is defined based on the age and obtained degrees, such that an applicant's experience X ranges from 0 to $A - 18 - 3D - 2E$, with a linearly increasing probability.

**goodness:** We define an objective goodness score $S \in [0, 10]$ based on the features above, to indicate how qualified an applicant is for the job. It takes into account the age, degree, extra degree and the experience of the applicant. Before assigning the goodness to the applicant, the simulator adds Gaussian noise. We assume noise on the goodness score, as two applicants with identical features may have different performance once hired, despite both being considered qualified. The noise is sampled from a Gaussian distribution with mean 0 and a standard deviation of 0.5. Based on this goodness score, the applicant is labelled as **qualified** or **not qualified**. Each applicant that scores higher than a given threshold is assigned value 1 (i.e., qualified) and the rest is assigned 0 (i.e., not qualified). We use a fixed threshold of 5 in these

[1] The job hiring simulator is an internal artefact of the VUB AI lab by Ioana Alexandra Cimpean.

experiments. Equation 1 is the formula we assume to compute the goodness of a candidate based on their features.

$$S = (A_{\max} - A) * w_A + (D * w_D) + (E * w_E) + (X * w_X) \\ + (D * X * w_X) + (E * X * w_X) + D * E * (w_D * w_E * w_X) + \mathcal{N}(0, 0.5) \tag{1}$$

The value of $A_{\max}$ is the maximum age (65 years). The weights $w_I$ refer to scaling factors applied for the corresponding feature $I$. The values of the weights can be found in Table 2.

| | $w_A$ | $w_D$ | $w_E$ | $w_X$ |
|---|---|---|---|---|
| **Value** | 0.15 | 1 | 1.5 | 0.25 |

Table 2: The corresponding weights for the features

## 5.2 Baseline

First, we have a look at a baseline scenario. Where, all applicants are generated following realistic distributions of men and women. In this scenario, we assume our unbiased, objective goodness score (Equation 1).

### 5.2.1 Ground truth

For the baseline scenario, we can consider the goodness score of the simulated applicants as the ground truth. First, we consider the mean of the proportions of applicants evaluated qualified for each gender in Figure 2. We can see that there is a negligible difference in the evaluation of men and women, with a small standard deviation over 100 iterations. Considering there is no bias in the dataset, this outcome is expected. The small discrepancy could be due to the high level of education for women in Belgium [27, 29].

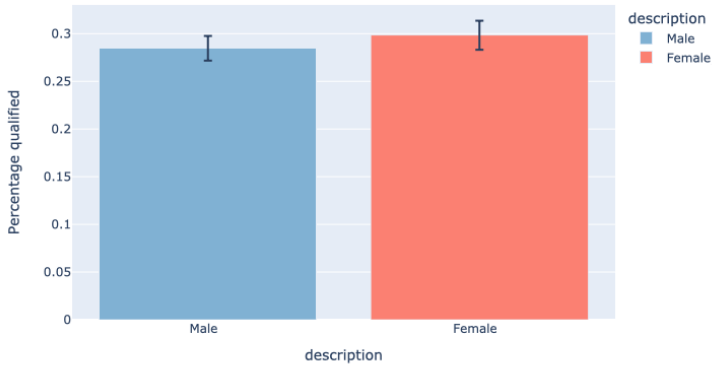



Figure 2: Proportions of men (blue) and women (red) evaluated qualified in the baseline scenario for the ground truth. The bars show the mean and standard deviations across 100 runs.

In Table 3, we observe that all fairness notions have a value of zero or close to zero. For predictive equality and equal opportunity, this is due to not having any false positives and false negatives.

Fairness notions, that look at errors between outcomes, are not adequate when there are no errors. For statistical parity and the inconsistency score, we do get a result as a consequence of not considering the ground truth. Getting a result of zero for a fairness notion means that the outcome is completely fair. The higher the result, the higher the unfairness in the outcome. As achieving a fairness of zero is difficult in practice, a small enough value typically suffices. We observe the inconsistency is rather high compared to the other fairness notions. This can be due to the noise added to the goodness score. The most similar candidates can still get a different goodness score, potentially resulting in different classifications.

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.018 \pm 0.017$ | $0 \pm 0$ | $0 \pm 0$ | $0.127 \pm 0.005$ |

Table 3: The fairness notions of the prediction of the ground truth in the baseline scenario. The bars show the mean and standard deviations across 100 runs.

### 5.2.2 Decision tree

We consider again the means of the proportions qualified per gender. This time for the constructed decision tree (Section 3.1.1).

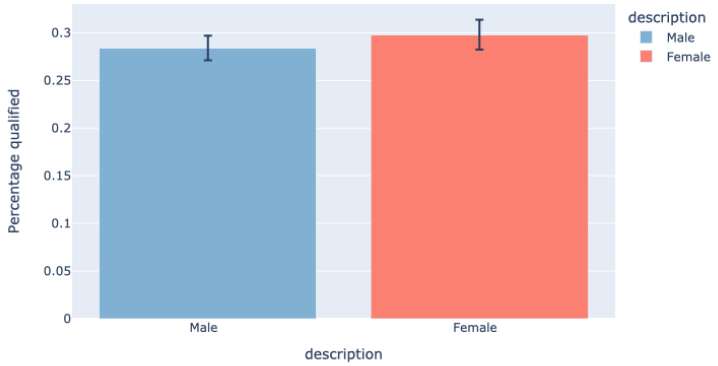
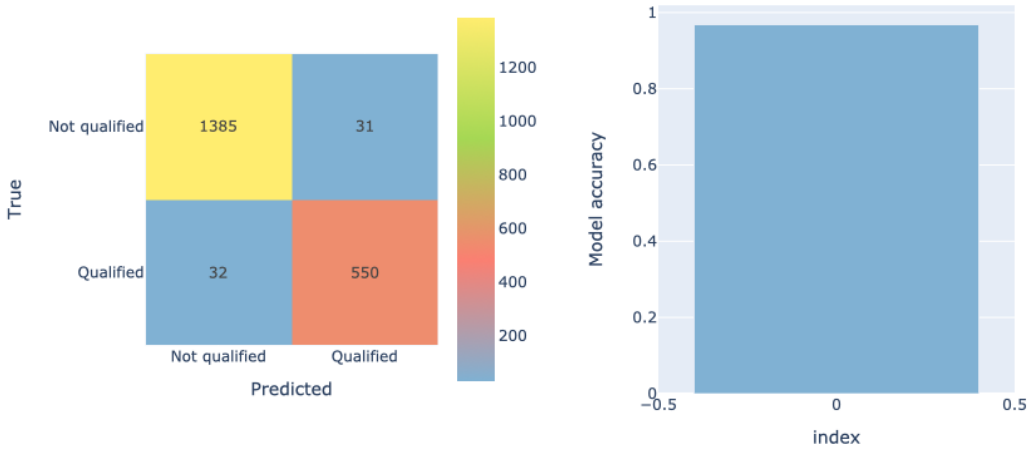



Figure 3: Proportions of men (blue) and women (red) evaluated qualified in the baseline scenario by the decision tree. The bars show the mean and standard deviations across 100 runs.

In Figure 3, we can see that the decision tree learned the pattern of the ground truth. For both men and women, the proportion of applicants who are considered qualified is more or less the same. Because the decision tree made its own prediction, we can now have a look at the confusion matrix and the accuracy of the model.




(a) Average confusion matrix (100 runs) (b) Average accuracy (100 runs)

Figure 4: Confusion matrix (a) and model accuracy (b) for a decision tree in the baseline scenario.

The confusion matrix in Figure 4a and the accuracy in figure 4b show that the decision tree learned the dataset rather well. It predicts the outcome of the applicants for both genders with an average accuracy close to 100%. Next, we observe the fairness notions computed for the prediction.

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.019 \pm 0.017$ | $0.022 \pm 0.004$ | $0.014 \pm 0.011$ | $0.105 \pm 0.004$ |

Table 4: The fairness notions of the prediction by the decision tree in the baseline scenario. The bars show the mean and standard deviations across 100 runs.

In Table 4, we can see that the results of the fairness notions are close to zero. The inconsistency in the classification of candidates is slightly reduced. This can be due to the absence of noise in the decision process of the decision tree, which is present in the training data due to the goodness score (Equation 1). We conclude that the decision tree had an unbiased prediction.

### 5.2.3 k-Nearest Neighbours

We discuss the results computed by the k-nearest neighbour algorithm. We look at the proportions of qualified candidates per group in Figure 5.

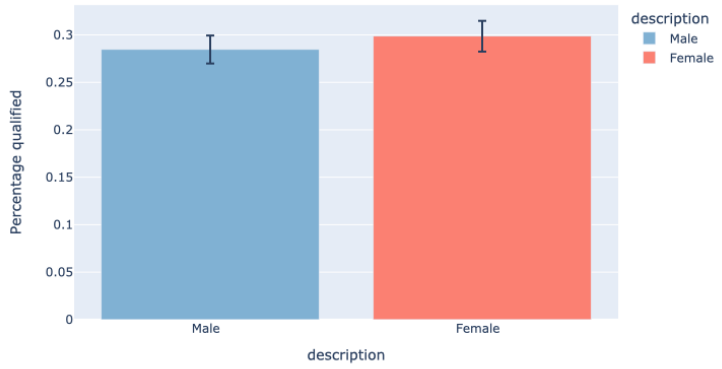
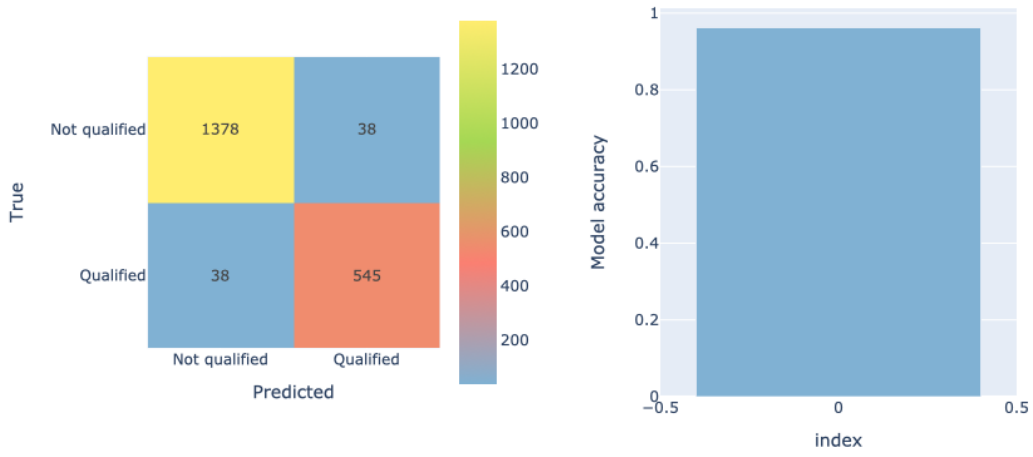



Figure 5: Proportions of men (blue) and women (red) evaluated qualified in the baseline scenario by k-nearest neighbours. The bars show the mean and standard deviations across 100 runs.

We observe similar results compared to the decision tree. This can again be explained by looking at the confusion matrix in Figure 6a and the accuracy of the model in Figure 6b. We can see that the model classified the candidates similarly to the decision tree. Hence, the high accuracy and the few false positives and false negatives in the confusion matrix.




(a) Average confusion matrix (100 runs)

(b) Average accuracy (100 runs)

Figure 6: Confusion matrix (a) and model accuracy (b) for k-nearest neighbours in the baseline scenario.

Finally, we consider the results for the average fairness notions and standard deviations computed for the k-nearest neighbour algorithm.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.021 \pm 0.017$ | $0.027 \pm 0.005$ | $0.021 \pm 0.016$ | $0.112 \pm 0.005$ |

Table 5: The fairness notions of the prediction by k-nearest neighbours in the baseline scenario. The bars show the mean and standard deviations across 100 runs.

In Table 5, we observe again small values for the fairness notions. We can conclude that k-nearest neighbours also produced an unbiased classification for the candidates.

## 5.3 Uneven distribution

In this scenario, the distribution of men and women is changed. We consider a case where the men make up approximately 70% of the dataset. Now we investigate the impact of available data from both groups on the performance of the model.

### 5.3.1 Ground truth

First, the proportions of qualified applicants of both genders are plotted in Figure 7.

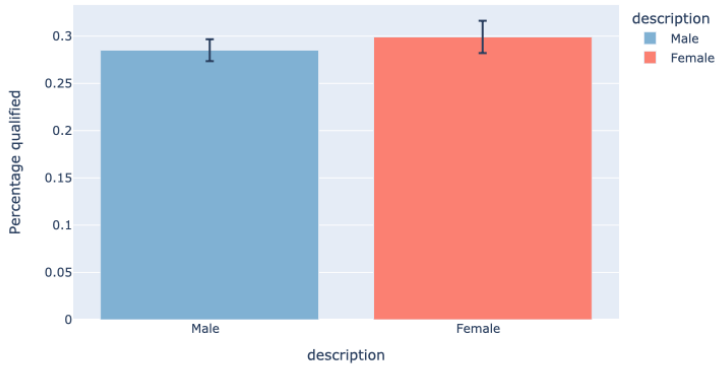



Figure 7: Proportions of men (blue) and women (red) evaluated qualified in the uneven distribution scenario for the ground truth. The bars show the mean and standard deviations across 100 runs.

Here, we see that the proportion of the applicants qualified from their respective groups are again approximately the same, with a small standard deviation. This is the expected result for a dataset with no bias. For the fairness notions the same applies here as for the baseline scenario as the dataset can be considered the ground truth. As statistical parity and the inconsistency do not take the ground truth into account, they again detect unfairness with regard to the groups when there is not necessarily any. The inconsistency of the ground truth is again higher due to the noise in the goodness score. In contrast, predictive equality and equal opportunity do take the ground truth into account and predict no unfairness with regard to the perfect prediction, which is shown in Table 6.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.019 \pm 0.017$ | $0 \pm 0$ | $0 \pm 0$ | $0.123 \pm 0.004$ |

Table 6: The fairness notions of the ground truth in the uneven distribution scenario. The bars show the mean and standard deviations across 100 runs.

### 5.3.2 Decision tree

Now we look at the impact of the different distributions on the decision tree. If the algorithm learned the dataset well enough, the proportions should be more or less the same. The results are plotted in Figure 8.

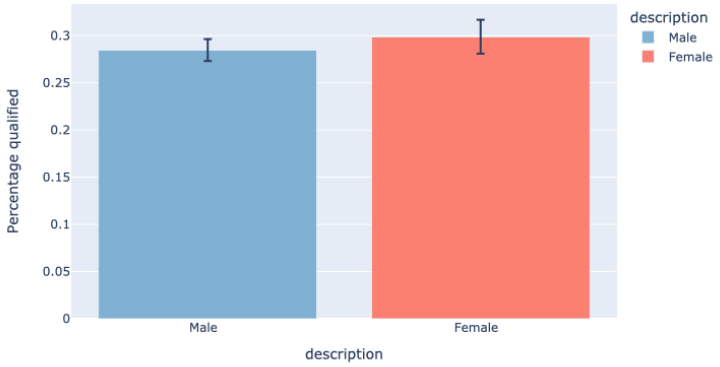
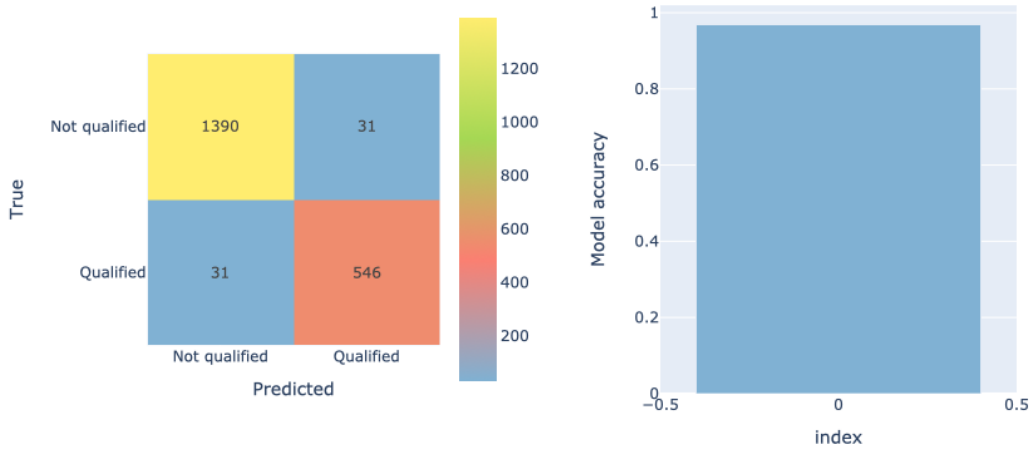



Figure 8: Proportions of men (blue) and women (red) evaluated qualified in the uneven distribution scenario by the decision tree. The bars show the mean and standard deviations across 100 runs.

As shown in Figure 8, we can see that the proportions are indeed not subject to bias. This can be supported by the results of the confusion matrix and accuracy, Figures 9a and 9b, respectively.




(a) Average confusion matrix (100 runs) (b) Average accuracy (100 runs)

Figure 9: Confusion matrix (a) and model accuracy (b) for decision tree in different distribution scenario.

The confusion matrix and the accuracy show that the decision tree was not impacted by the discrepancy between the amount of women and men in the dataset. Even though the amount of men considered qualified is higher, proportionally the women are treated equally. We can see that the decision tree made mistakes on predicting outcomes for candidates, but it did not discriminate. Next, we consider the results of the fairness notions in Table 7.

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.019 \pm 0.017$ | $0.007 \pm 0.006$ | $0.015 \pm 0.011$ | $0.102 \pm 0.004$ |

Table 7: The fairness notions of the prediction by the decision tree in the uneven distribution scenario. The bars show the mean and standard deviations across 100 runs.

We observe similar results for the fairness notions in Table 7. Again, there is a slight decrease in the inconsistency due to the lack of noise when classifying. The difference in the other fairness notions is small and is negligible on this scale.

### 5.3.3 k-Nearest Neighbours

We investigate whether the results of k-nearest neighbour algorithm differ from the decision tree results. Figure 10 depicts the average proportions classified as qualified by the k-nearest neighbour algorithm. In this graph, we can observe again a similar qualification rate for both genders. We can explain this by looking at Figure 11a and 11b, which tell us that the model classified the candidates similarly to the ground truth.

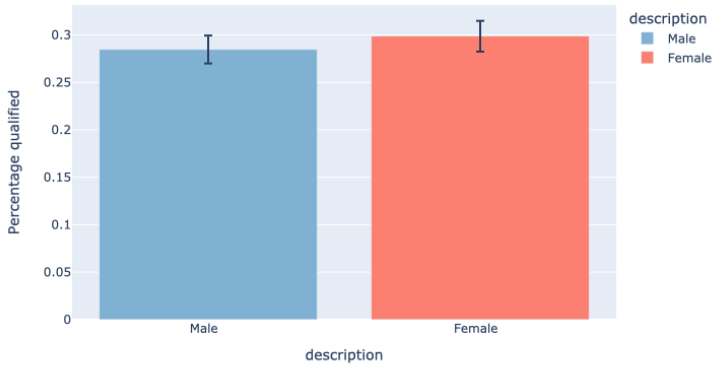
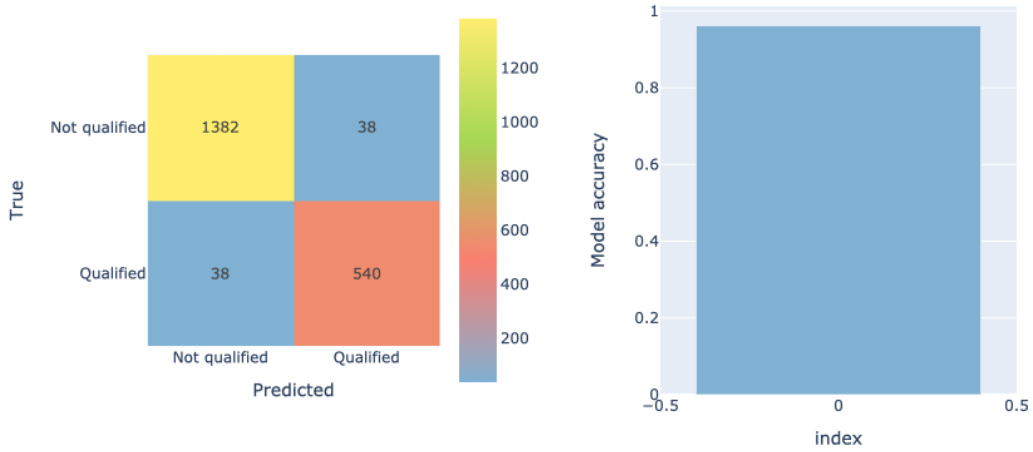



Figure 10: Proportions of men (blue) and women (red) evaluated qualified in the uneven distribution scenario by k-nearest neighbours. The bars show the mean and standard deviations across 100 runs.




(a) Average confusion matrix (100 runs) (b) Average accuracy (100 runs)

Figure 11: Confusion matrix (a) and model accuracy (b) for k-nearest neighbours in different distribution scenario.

Next, we observe the average computations of the fairness notions with their standard deviations in Table 8. The table shows values close to zero for all group fairness notions, and again a slightly smaller value for the inconsistency. We can confirm that the k-nearest neighbours algorithm classified the candidates in a fair and objective way.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.023 \pm 0.018$ | $0.027 \pm 0.006$ | $0.027 \pm 0.019$ | $0.109 \pm 0.005$ |

Table 8: The fairness notions of the prediction by k-nearest neighbours in the uneven distribution scenario. The bars show the mean and standard deviations across 100 runs.

## 5.4 Bias

Next, we consider two bias scenarios. In the first case, men are considered better applicants than women. Each of their goodness scores is incremented by 2, and consequently they have a higher likelihood of being evaluated as qualified. Secondly, we discuss a scenario in which men and candidates with a Belgian nationality are considered better applicants. Again, a surplus of 2 is added to the goodness score depending on nationality. This means that Belgian men get 4 added to their goodness score. Following the selection criterion of unreliable outcome, fairness notions that rely on the ground truth are unsuited for this scenario (Section 3.2.1).

### 5.4.1 Gender-based bias

In this case, we only look at gender as a sensitive feature. First, we discuss the original dataset, followed by the predictions of the machine learning models. We can no longer call the initial dataset the ground truth. After all, the ground truth are results inferred in an objective manner. This dataset was created using a bias judgement, and thus can no longer be referred to as the truth. We consider the proportions of qualified applicants for both genders in Figure 12.

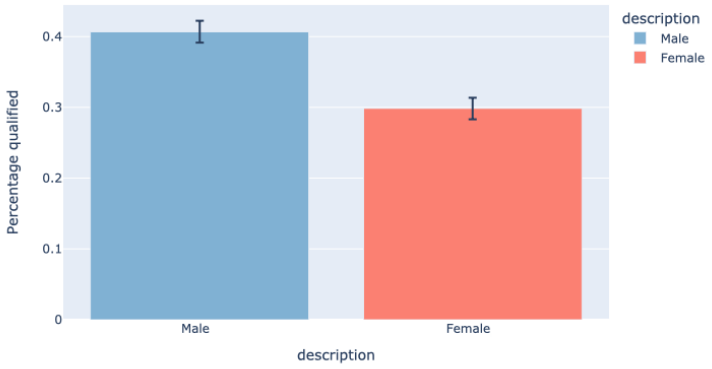



Figure 12: Proportions of men (blue) and women (red) evaluated qualified in the gender-based bias scenario for the original dataset. The bars show the mean and standard deviations across 100 runs.

This time, the proportions of qualified applicants differ for the considered genders. As expected, the amount of qualified men is higher than the amount of qualified women. We have a look at the computations for the fairness notions.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.109 \pm 0.022$ | $0 \pm 0$ | $0 \pm 0$ | $0.152 \pm 0.005$ |

Table 9: The fairness notions of the original dataset in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

It is visible why statistical parity is a fairness notion that is used in scenarios where the ground truth is not available or unreliable. Statistical parity can quantify discrimination in datasets where no there are no actual values at hand. The other fairness notions are unsuitable because the used ground truth is unreliable. Therefore, this produces a deceiving result, as seen in Table 9. Next, we look at the impact of the encoded bias on a decision tree.

#### 5.4.1.1 Decision tree

We examine whether decision trees adopt the discriminative behaviour of the original dataset, and to what extent. If the decision tree learns the dataset well enough, we should see similar results. We look at the proportions of qualified applicants per gender in Figure 13.

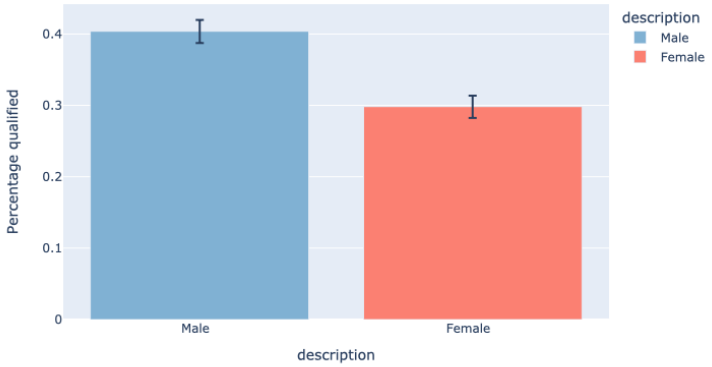
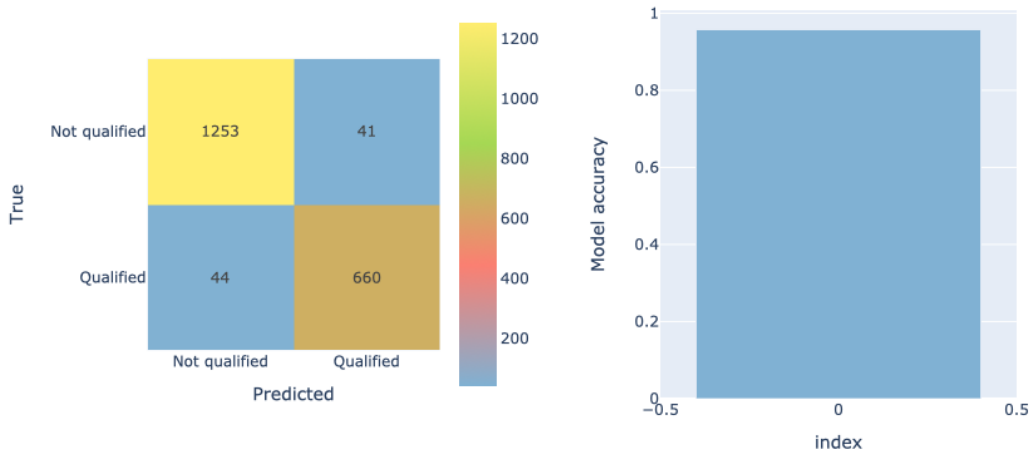



Figure 13: Proportions of men (blue) and women (red) evaluated qualified in the gender-based bias scenario by the decision tree. The bars show the mean and standard deviations across 100 runs.

We can see that the algorithm indeed copied the discriminative behaviour of the original dataset. This result can be explained when looking at the confusion matrix in Figure 14a and the accuracy in 14b.




(a) Average confusion matrix (100 runs)

(b) Average accuracy (100 runs)

Figure 14: Biased confusion matrix (a) and model accuracy (b) for decision tree in bias scenario

The decision tree learned the training set with a high accuracy, as for previous experiments. This means that patterns from the dataset are likely to be observable in the prediction of the decision tree. Next, we look at the results of the fairness notions in Table 10.

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.106 \pm 0.023$ | $0.032 \pm 0.005$ | $0.021 \pm 0.015$ | $0.129 \pm 0.005$ |

Table 10: The fairness notions of the prediction by the decision tree in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

We can see that all fairness notions pick up on the discriminative behaviour of the decision tree.

**Post-processing** Next, we will use the reject option classifier discussed in Section 3.3.2 to mitigate the bias in the prediction. In Figure 15 and Table 11 we can see the effects of this mitigation technique.

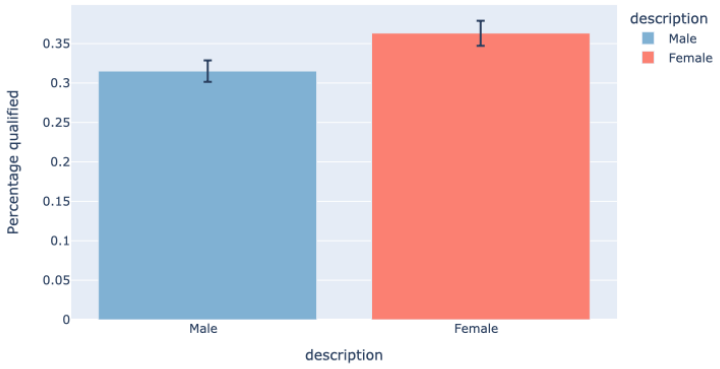



Figure 15: Proportions of men (blue) and women (red) evaluated qualified in the mitigated prediction by the decision tree in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

Figure 15 shows a difference in the average proportions of the groups classified as qualified. The post-processing technique altered the outcomes drastically, resulting in more women being considered qualified than men. We observe the effect on the fairness notions in Table 11

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.048 \pm 0.021$ | $0.052 \pm 0.008$ | $0.225 \pm 0.02$ | $0.118 \pm 0.005$ |

Table 11: The fairness notions of the mitigated prediction by the decision tree in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

We can see that the post-processing technique worked. The difference in statistical parity between the groups has been decreased significantly. The inconsistency has been decreased by a small margin. Both predictive equality and equal opportunity have increased. This is due to the reject option classifier not considering true or false positive rates when altering outcomes. This illustrates the trade-off that has to be made when trying to achieve total fairness [3]. When decreasing one fairness notion, it may be possible that a different fairness notion is being increased.

#### 5.4.1.2 k-Nearest Neighbours

We investigate the influence of the biased classification in the original dataset on the k-nearest neighbours algorithm. Figure 16 shows the impact of the bias on the algorithm. Proportionally, more men are considered qualified than women. We can assume that the model classified the candidates similarly to the original dataset. Figures 17a and 17b confirm this assumption. The k-nearest neighbours made few mistakes and performed with an accuracy close to 100%.

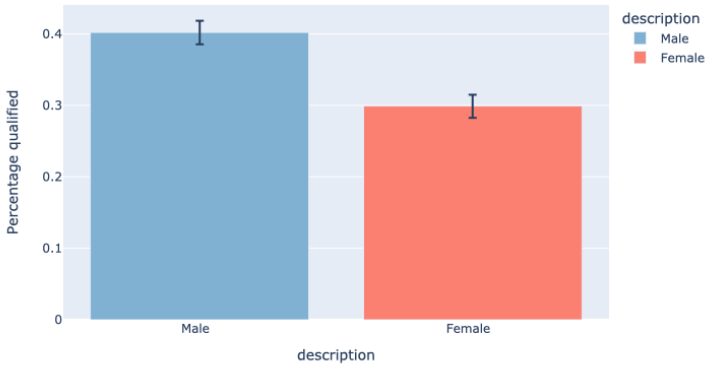
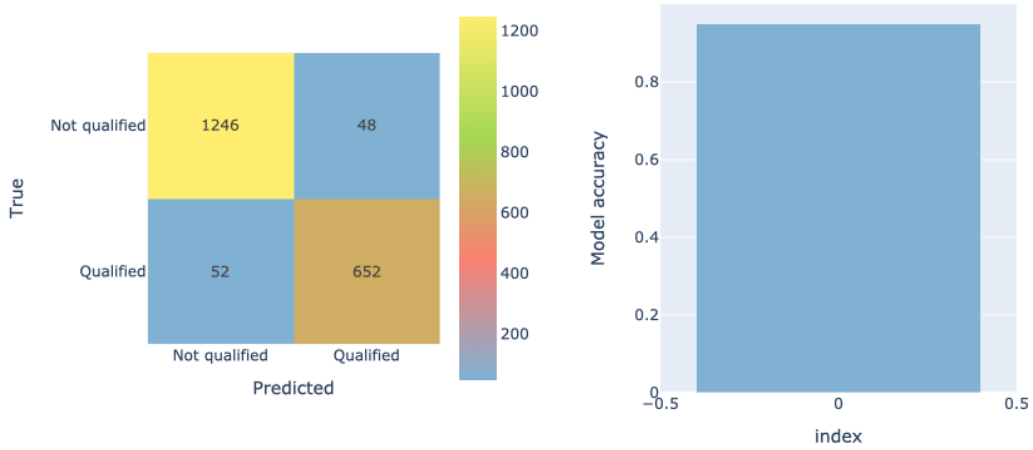



Figure 16: Proportions of men (blue) and women (red) evaluated qualified in the gender-based bias scenario by k-nearest neighbours. The bars show the mean and standard deviations across 100 runs.




(a) Average confusion matrix (100 runs) (b) Average accuracy (100 runs)

Figure 17: Confusion matrix (a) and model accuracy (b) for k-nearest neighbours in bias scenario

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.103 \pm 0.023$ | $0.037 \pm 0.007$ | $0.025 \pm 0.02$ | $0.135 \pm 0.006$ |

Table 12: The fairness notions of the prediction by k-nearest neighbours in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

We look at the average results of the fairness notions in Table 12. We observe again that statistical parity is the most suitable fairness notion in absence of a ground truth. The values for the fairness notions that make use of the errors made, are close to zero, making it seem no bias is encoded in the dataset.

**Post-processing** We again use the reject option classifier to mitigate the prediction made by the k-nearest neighbour algorithm. We investigate the results of the mitigation.

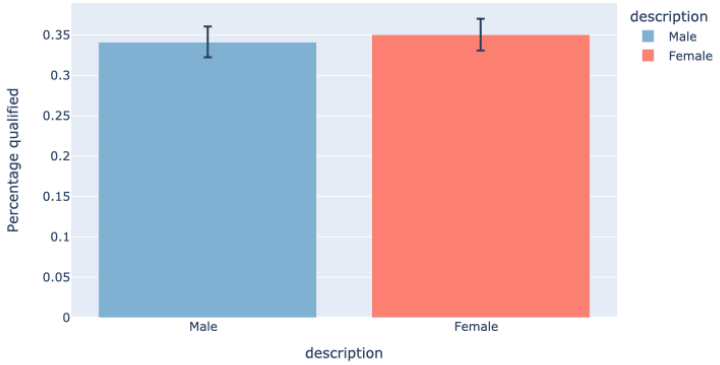



Figure 18: Proportions of men (blue) and women (red) evaluated qualified in mitigated prediction of k-nearest neighbours in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.017 \pm 0.015$ | $0.054 \pm 0.012$ | $0.161 \pm 0.029$ | $0.129 \pm 0.008$ |

Table 13: The fairness notions of the mitigated prediction of k-nearest neighbours in the gender-based bias scenario. The bars show the mean and standard deviations across 100 runs.

Figure 18 and Table 13 show that the mitigation worked. The proportions of the genders classified as qualified are now more similar, as opposed to the mitigation of the decision tree. In the fairness notions, there is again an increase in predictive equality and equal opportunity. However, the mitigation decreased the statistical parity difference.

### 5.4.2 Bias based on intersectionality

In this section, we discuss a case with two sensitive features, where the candidates can be discriminated against based on gender and nationality. Considering we introduce the likelihood of intersectionality, we focus on individual discrimination (Section 3.2.1). We first look at the average proportions of candidates deemed qualified. We do this for all possible groups based on the combination of the sensitive features gender and nationality.

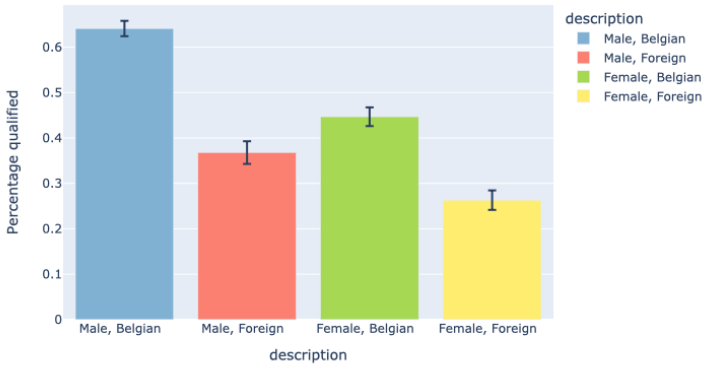



Figure 19: Proportions of all different groups evaluated qualified in the intersectionality-based bias scenario for the original dataset. The bars show the mean and standard deviations across 100 runs.

In Figure 19 we can see that Belgian men are indeed the most favoured candidates amongst the groups. The Belgian women are the second most favoured, followed by the foreign men and women. We look at the fairness notions to quantify this discriminative behaviour.

| **Fairness notions** | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.266 \pm 0.02$ | $0 \pm 0$ | $0 \pm 0$ | $0.184 \pm 0.006$ |

Table 14: The fairness notions of the original dataset in the intersectionality-based bias scenario. The bars show the mean and standard deviations across 100 runs.

Table 14 shows a high result for statistical parity and inconsistency. This is the result of having a higher discrepancy in goodness scores. Belgian men get a significantly higher score compared to other candidates, which makes them almost immediately considered qualified.

#### 5.4.2.1 Decision tree

Now we look at the results of the constructed decision tree to illustrate the influence of this bias. Figure 20 depicts the proportions in qualified predictions by the decision tree.

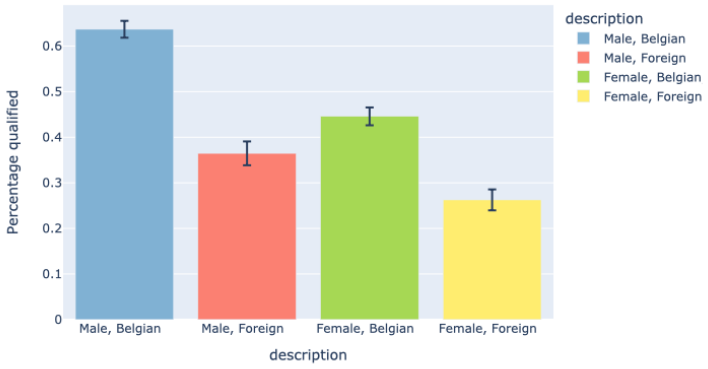
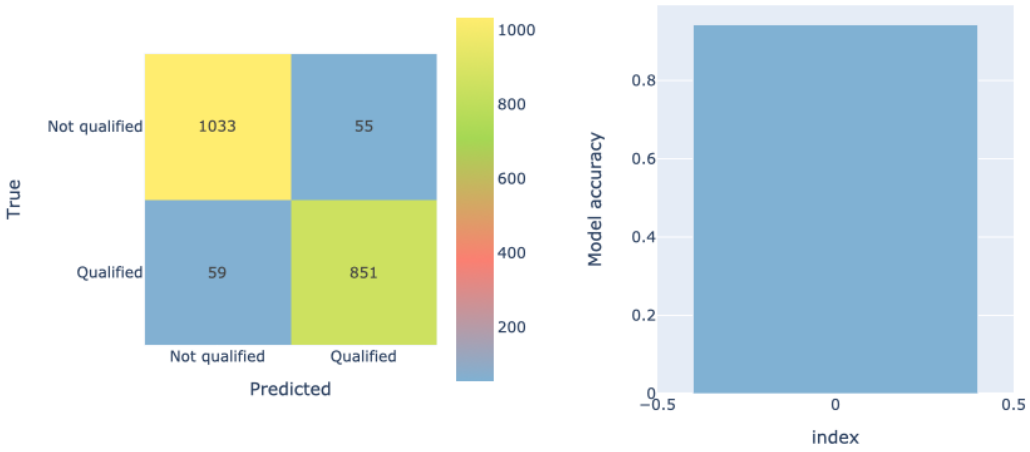



Figure 20: Proportions of all different groups evaluated qualified in the intersectionality-based bias scenario by the decision tree. The bars show the mean and standard deviations across 100 runs.

We can see that the model produced results similar to those of the original dataset. This can again be explained by the confusion matrix and the average accuracy in Figure 21a and 21b.




(a) Average confusion matrix (100 runs)

(b) Average accuracy (100 runs)

Figure 21: Biased confusion matrix (a) and model accuracy (b) for decision tree in bias scenario

Next, we look at the average fairness notions computed for the 100 iterations in Table 15. The computations reflect the bias influence in the decision-making of the decision tree. The inconsistency is again slightly lower due to the noise in the goodness score.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.264 \pm 0.021$ | $0.051 \pm 0.007$ | $0.014 \pm 0.01$ | $0.157 \pm 0.006$ |

Table 15: The fairness notions of the prediction by the decision tree in the intersectionality-based bias scenario. The bars show the mean and standard deviations across 100 runs.

**Post-processing** We will again use the reject option classifier to mitigate the discriminative behaviour of the decision tree. In Figure 22 we observe this time only a minor improvement on first sight.

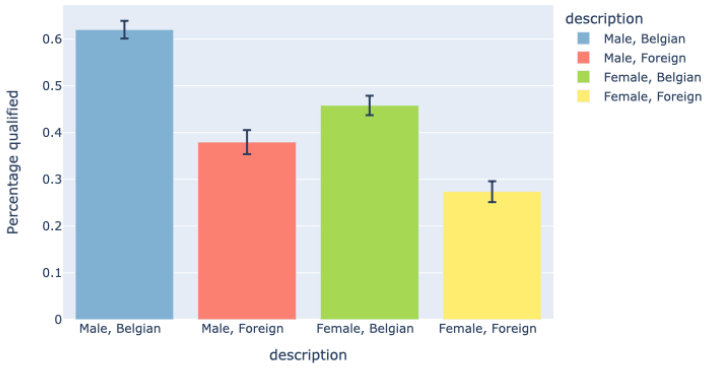



Figure 22: Proportions of all different groups evaluated qualified in mitigated prediction of the decision tree in the bias scenario. The bars show the mean and standard deviations across 100 runs.

We investigate the fairness notions in Table 16 for other signs of reduced bias.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.234 \pm 0.023$ | $0.05 \pm 0.008$ | $0.029 \pm 0.017$ | $0.15 \pm 0.005$ |

Table 16: The fairness notions of the mitigated prediction of the decision tree in the intersectionality-based bias scenario. The bars show the mean and standard deviations across 100 runs.

Table 16 confirms the small effect of the mitigation technique. Only the difference in statistical parity has been reduced by some amount. This explains the small reduction of Belgian men considered qualified. The mitigation technique only changed a few of the classifications of the Belgian men to not qualified and the other groups to qualified.

#### 5.4.2.2 k-Nearest Neighbours

At last, we look at the influence of the intersectionality based bias on the k-nearest neighbours algorithm. Figure 23 shows a similar result as seen previously. Foreign women are least favoured

amongst the candidates, while Belgian men are considered the best.

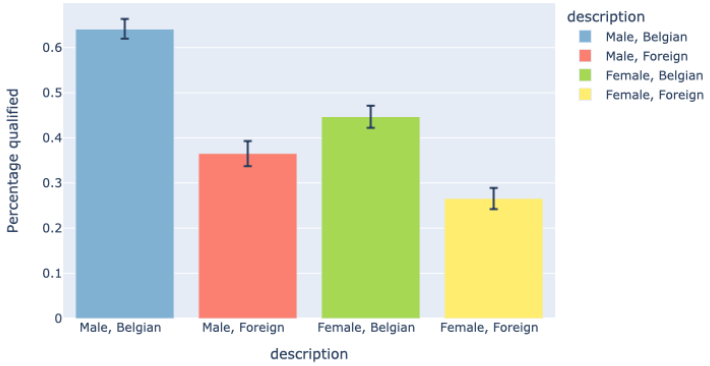
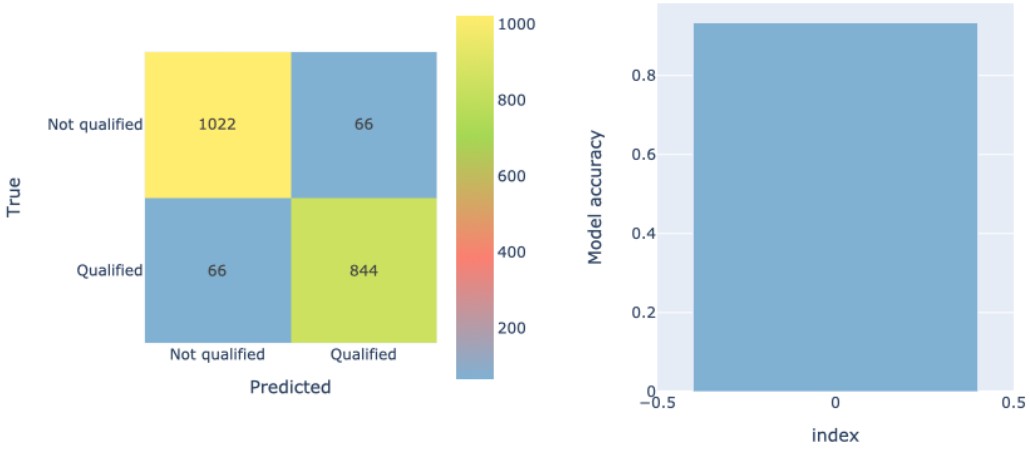



Figure 23: Proportions of all different groups evaluated qualified in the intersectionality-based bias scenario by k-nearest neighbours. The bars show the mean and standard deviations across 100 runs.

The similar results are reflected in the results of the confusion matrix and accuracy in Figures 24a and 24b.




(a) Average confusion matrix (100 runs)

(b) Average accuracy (100 runs)

Figure 24: Biased confusion matrix (a) and model accuracy (b) for k-nearest neighbours in bias scenario

We look at the average fairness notions in Table 17 to prove the discriminative behaviour. We can see the difference in evaluation of candidates reflected in the statistical parity difference and

the inconsistency. The k-nearest neighbours algorithm applied the same discriminative behaviour observed in the original dataset.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.267 \pm 0.027$ | $0.061 \pm 0.009$ | $0.021 \pm 0.015$ | $0.161 \pm 0.007$ |

Table 17: The fairness notions of the prediction by k-nearest neighbours in the intersectionality-based bias scenario. The bars show the mean and standard deviations across 100 runs.

**Post-processing** We use the reject option classifier to post-process the prediction made by the algorithm. Figure 25 shows the difference in evaluation between the groups.

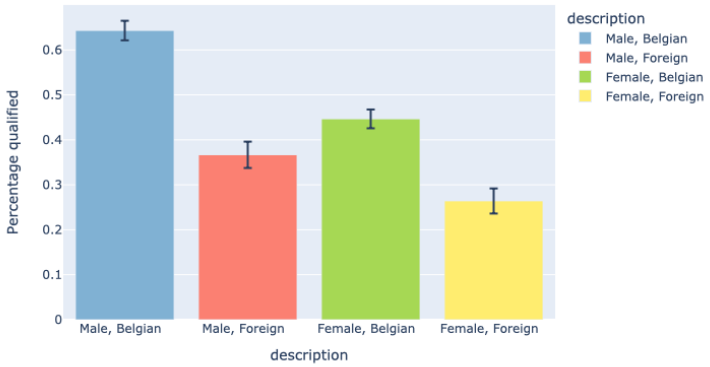



Figure 25: Proportions of all different groups evaluated qualified in mitigated prediction of k-nearest neighbours in the intersectionality-based bias scenario. The bars show the mean and standard deviations across 100 runs.

Again, the mitigation only had little to no effect on the proportions of candidates considered qualified. All discriminated groups increase a small percentage, and the proportion of Belgian men is reduced slightly. We look at the average mitigated fairness notions in Table 18.

| Fairness notions | | | |
|---|---|---|---|
| Statistical parity | Predictive equality | Equal opportunity | Inconsistency |
| $0.269 \pm 0.023$ | $0.057 \pm 0.011$ | $0.018 \pm 0.014$ | $0.161 \pm 0.007$ |

Table 18: The fairness notions of the mitigated prediction of k-nearest neighbours in the intersectionality-based bias scenario. The bars show the mean and standard deviations across 100 runs.

This time, the negligible effect of the mitigation is reflected in the computations of the fairness notions. Not a single fairness notion shows a significant improvement in fairness after post-processing. We can conclude that the reject option classifier did not improve the fairness of both the prediction of the decision tree and k-nearest neighbours in bias based on intersectionality.

# 6 Framework

In this section, we will describe the constructed framework application. The user is guided through several steps to detect potential bias in a personal dataset. This bias can then be mitigated using a chosen pre- or post-processing technique[2]. Images illustrating each step be found in Appendix 8.

## 6.1 Upload

In the first step, the user can upload a dataset consisting of candidates for a job. These candidates should all have certain features and should be paired with a label. This label denotes whether the candidate is considered qualified. Figure 26 shows the upload step.

## 6.2 Setup

Upon successfully uploading a dataset, the user is shown a sample of it. Next, the user selects the sensitive features important for their problem context and the machine learning algorithm they wish to train. The sensitive features are the features that determine how the candidates are divided into a privileged group and an unprivileged group. The fairness notions will be computed as the difference between the results for both groups. The choice of algorithm decides whether an algorithm will construct a model based on the dataset (decision tree) or whether an algorithm classifies candidates based on similar candidates (k-nearest neighbours). Additionally, the user can choose to just use the dataset for the further steps. This means that the uploaded dataset itself be tested on fairness. The different options are shown in Figure 27.

## 6.3 Train model

On the next screen, the machine learning model is trained on the dataset. A confusion matrix and a graph that denotes the accuracy of the model is shown. The user is given insight on how the model was trained and how to interpret the results. In Figure 28, we can see a possible outcome of this step.

## 6.4 Fairness

In this step, the prediction is evaluated on fairness. First, the results of the prediction are illustrated by a sunburst plot and a proportional bar graph. The sunburst plot illustrates the amount of candidates classified as qualified per group. This graph provides a clear view of the distribution of qualified candidates, but may be misleading. When a certain group is represented more, it is expected to classify more candidates from this group as qualified. The proportional bar graph shows a more objective view of this distribution. For each group, the proportions of candidates considered qualified are computed. This removes the influence of base rates on the illustration of bias in the dataset. Figures 29 and 30 show an example of the illustration of fairness in this step.

Below the sunburst plot and proportional bar graph, the computations of the supported fairness notions are plotted in another bar graph. The amount of bias in the prediction is highlighted in a clear and intuitive way. The user is given insight in the suitability and the computability of the fairness notions. It gives the user the ability to investigate the results with a critical view.

[2]Our framework can be found at `https://github.com/samvanspringel/Fairness-framework`.

### 6.5 Mitigation

This step explains a selection of pre- and post-processing techniques. The user is provided some background on the techniques, giving them insight on which to use. Using the chosen pre- or post-processing technique, the bias in the prediction will be mitigated. Additionally, the user can choose to save the testing part of the dataset, the model prediction and the mitigated prediction. This provides the ability to see which candidates should have been classified differently according to the mitigation. The different options and their explanation is shown in Figure 31.

### 6.6 Comparison

In the last step, the user is shown a summary. A new sunburst plot is shown to compare the amount of qualified candidates per group. This illustrates the amount of individual candidates that were classified differently after mitigation. Below these plots, the same comparison is made for the proportions per group, showing the effect of the mitigation. Figures 32 and 33 depict an example of a comparison in the last step in our framework.

Finally, the user is shown a comparison of the fairness notions. This comparison illustrates which fairness notions have been decreased or potentially increased.

## 7 Conclusion

We explore multiple existing fairness notions to quantify the fairness of an algorithm's decision [8]. Whether a fairness notion is applicable to a certain scenario is dependent on distinct criteria [16]. This makes deciding appropriate fairness notions for the job hiring scenario a non-trivial task. Using various fairness notions, we are able to investigate whether an algorithm makes objective decisions. Using this insight in the potential discriminative behaviour of a machine learning model, we explore mitigation techniques to increase the fairness.

In our framework, we show that machine learning algorithms indeed adopt discriminative behaviour if it is encoded in the dataset that they learn from. It is important to quantify this discrimination, as the decisions of an algorithm, may impact communities and individuals profoundly. It elucidates different fairness notions and their suitability by applying them [16]. When a biased judgement of a machine learning model is identified, mitigation techniques can be applied to reduce the discriminative behaviour [12, 14, 22, 7, 13, 4, 31]. Using certain pre- and post-processing techniques, we mitigate the bias judgement of the model. The framework shows the difference between the classification before and after mitigation, illustrating the effect of the pre- and post-processing techniques. Exploratory experiments show that different scenarios and machine learning models, produce different results. We illustrate the need for mitigating techniques that influence the machine learning model on their functionality. Additionally, we observe the trade-off between satisfying multiple fairness notions, proving the difficulty of achieving total fairness [3].

# 8 Future Work

In future work, we envision that our framework can be used to conduct more elaborate experiments and explore the topic of fairness further. The framework can be extended to support other scenarios, and their context-specific fairness requirements, in which machine learning could be beneficial. Consequently, the importance of fairness can be illustrated in a more generalized way. Other scenarios allow us to focus on different fairness notions, resulting in broader insights for improving our framework. Additional mitigation techniques such as in-processing techniques and fairness notions can also be added to the framework.

# A Images

## A.1 Upload


Figure 26: Uploading a dataset

## A.2 Setup

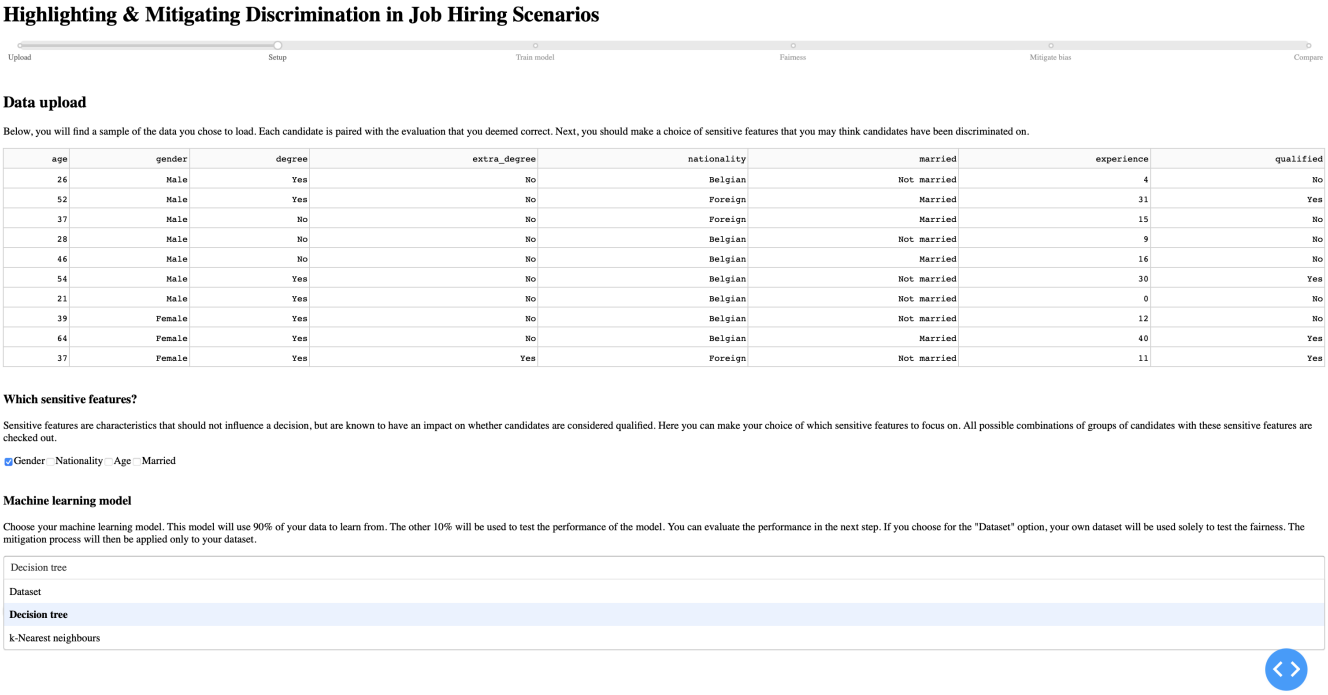



Figure 27: Setup of the application

## A.3 Train model

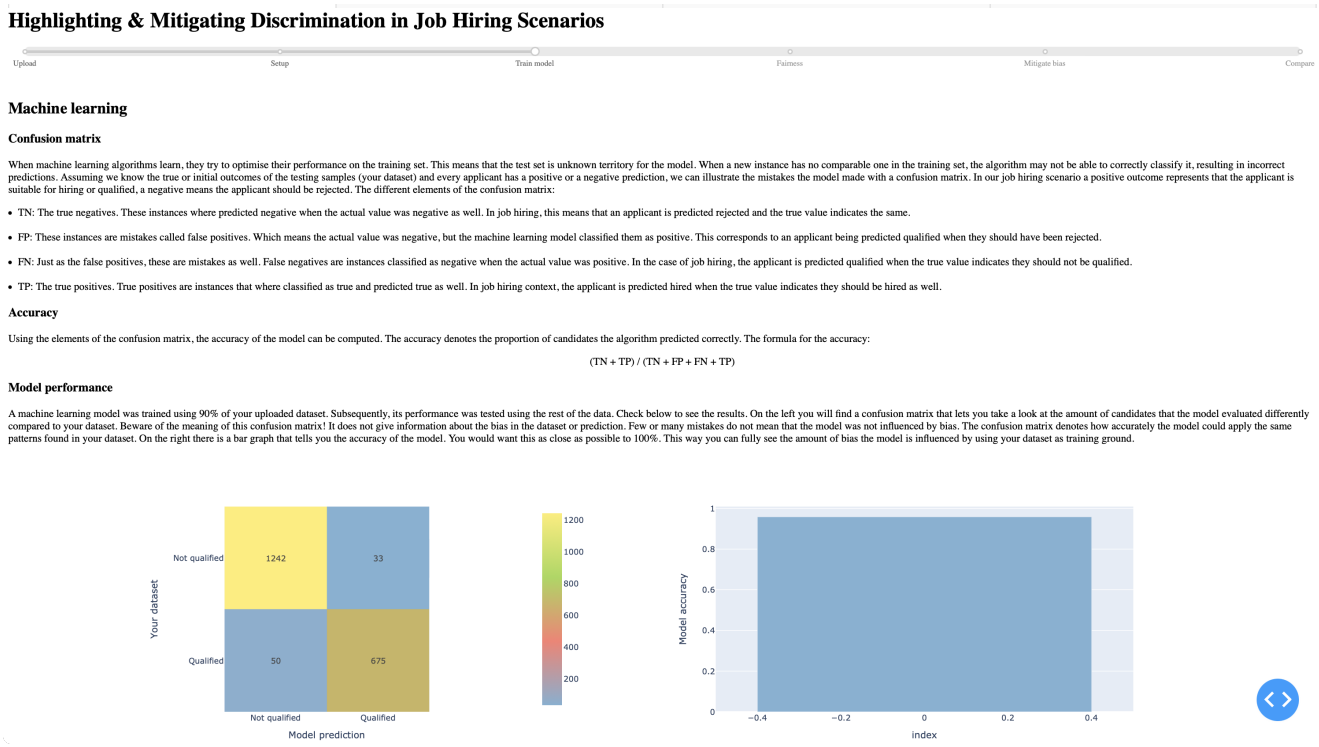



Figure 28: Model trained

## A.4 Fairness


Figure 29: Sunburst plot and proportional bar graph

**Fairness notions**

Below you will find certain fairness notions that quantify the amount of bias the model was influenced by. Each fairness notion focuses on a different aspect of the data. The group fairness notions calculate the difference in certain rates between the groups made on sensitive features. If you find that these values are too high, you will find certain methods to mitigate this bias. Ultimately, you would want these values as close as possible to zero.

**Group fairness notions**

- Statistical parity: When the statistical parity of two groups is equal, it means that both groups have an equal acceptance rate. To satisfy statistical parity, the probability of being predicted qualified should be the same for all groups. This fairness notion only looks at the prediction of the model. This way it can detect bias in datasets without having to compare it to a prediction. Because it doesn't use a comparison between datasets, it is suitable when the outcomes of the initial dataset are unreliable. If the outcomes of the candidates were decided by a subjective source, this fairness notion will give you the most clear view of the bias. Statistical parity is computed by this formula:

$$(TP + FP)/(TP + FP + FN + TN)$$

- Equal opportunity: This fairness notion is a relaxation of the fairness notion equalized odds. Equalized odds requires the sensitive attribute to be conditionally independent from the outcome. However, satisfying equalized odds is difficult to implement and interpret in practice. Therefore, we consider its two relaxed versions. The first relaxation is equal opportunity. In equal opportunity the true positive rates (TPR) or sensitivity recall should be equal for all groups. The formula for the true positive rates is the following:

$$TPR = TP/(TP + FN)$$

- Predictive equality: The other relaxation of equalized odds that takes into account the false positives is predictive equality. When false positives are important to the fairness of decisions, predictive equality is a better option than equal opportunity. Predictive equality demands the false positive rates (FPR) of two groups to be the same.

$$FPR = FP/(FP + TN)$$

**Individual fairness notions**

- Consistency score: The consistency score is computed by checking the prediction of the nearest neighbors or the most similar other candidates of a candidate. For each candidate the percentages of the nearest neighbours that have the same prediction is calculated. Taking the average of these values over all candidates results in the consistency score. In the application the complement of this score is used to illustrate the inconsistency in the dataset:


Figure 30: Results fairness notions

## A.5 Mitigation

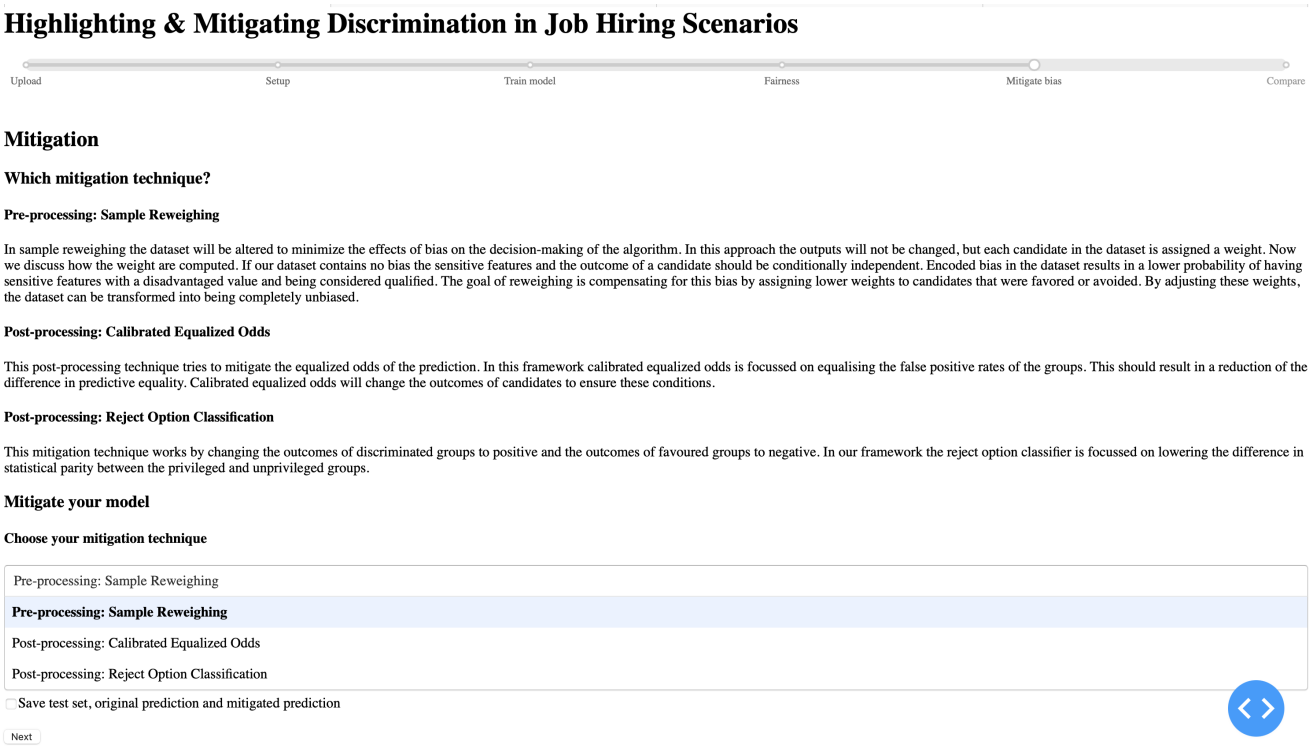


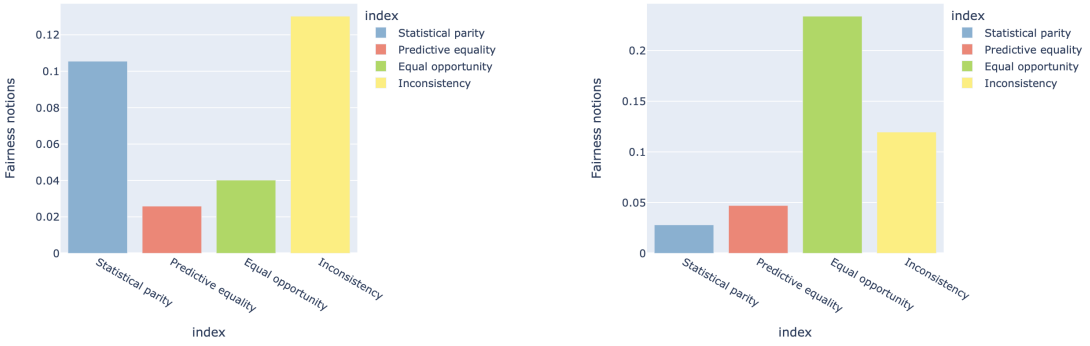

Figure 31: The user chooses a mitigation technique

## A.6 Comparison


Figure 32: Comparison of the qualified candidates

**Fairness notions**

Below you will find a comparison of the computations of the fairness notions from before and after mitigation. This will illustrate the best whether the mitigation worked. The fairness notion that your chosen technique focussed on, should have been reduced. Depending on the results, you may also see that certain fairness notions have been increased. This is due to the difficulty of satisfying multiple fairness notions at once. When decreasing one, you may be increasing a different fairness notion.




Next

Figure 33: Comparison of the fairness notions

# References


[1] IBM Research Trusted AI. AI Fairness 360. URL `https://aif360.mybluemix.net`. `https://aif360.mybluemix.net`.

[2] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019. `http://www.fairmlbook.org`.

[3] Richard A. Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 50:3 – 44, 2018.

[4] Flavio P. Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R. Varshney. Optimized pre-processing for discrimination prevention. In *31st International Conference on NIPS*, page 3995–4004, 2017. ISBN 9781510860964.

[5] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *3rd Innovations in Theoretical Computer Science Conference*, page 214–226, 2012. ISBN 9781450311151. doi: 10.1145/2090236.2090255.

[6] Cynthia Dwork, Christina Ilvento, Guy N. Rothblum, and Pragya Sur. Abstracting Fairness: Oracles, Metrics, and Interpretability. In *1st Symposium on Foundations of Responsible Computing*, 2020.

[7] Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *21st International Conference on Knowledge Discovery and Data Mining*, KDD '15, page 259–268, 2015. ISBN 9781450336642. doi: 10.1145/2783258.2783311.

[8] Sorelle A. Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. "On the (im)possibility of fairness", 2016.

[9] Nina Grgić-Hlača, Muhammad Bilal Zafar, Krishna P. Gummadi, and Adrian Weller. Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 51–60, 2018.

[10] Jerry A. Jacobs. Gender inequality and higher education. *Annual Review of Sociology*, 22: 153–185, 1996. ISSN 03600572, 15452115. URL `http://www.jstor.org/stable/2083428`.

[11] Faisal Kamiran and Toon Calders. Classifying without discriminating. In *2nd International Conference on Computer, Control and Communication*, pages 1–6, 2009. doi: 10.1109/IC4.2009.4909197.

[12] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012. doi: 10.1007/s10115-011-0463-8. URL `https://doi.org/10.1007/s10115-011-0463-8`.

[13] Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. Discrimination aware decision tree learning. In *2010 IEEE International Conference on Data Mining*, pages 869–874, 2010. doi: 10.1109/ICDM.2010.50.

[14] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929, 2012. doi: 10.1109/ICDM.2012.45.

[15] 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. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-33486-3.

[16] Karima Makhlouf, Sami Zhioua, and Catuscia Palamidessi. "On the applicability of ML fairness notions", 2020.

[17] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *arXiv*, 2019. ISSN 23318422.

[18] John Mingers. An Empirical Comparison of Pruning Methods for Decision Tree Induction. *Machine Learning*, 4(2):227–243, 1989. doi: 10.1023/A:1022604100933.

[19] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, 1 edition, 1997. ISBN 978-0070428072.

[20] Gonzalo Nápoles and Lisa Koutsoviti Koumeri. A fuzzy-rough uncertainty measure to discover bias encoded explicitly or implicitly in features of structured pattern classification datasets. *Pattern Recognition Letters*, 154, 01 2022. doi: 10.1016/j.patrec.2022.01.005.

[21] 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.

[22] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/b8b9c74ac526fffbeb2d39ab038d1cd7-Paper.pdf`.

[23] J. R. Quinlan. Simplifying decision trees. *International Journal of Man-Machine Studies*, 27(3):221–234, 1987. ISSN 00207373. doi: 10.1016/S0020-7373(87)80053-6.

[24] Stuart Russel and Peter Norvig. *Artificial Intelligence : A Modern Approach*. Pearson Education, 3 edition, 2010. ISBN 9780136042594.

[25] Candice Schumann, Samsara N. Counts, Jeffrey S. Foster, and John P. Dickerson. The Diverse Cohort Selection Problem. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 2:601–609, 2019. ISSN 15582914.

[26] Candice Schumann, Jeffrey S. Foster, Nicholas Mattei, and John P. Dickerson. We need fairness and explainability in algorithmic hiring. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 2020-May(Aamas): 1716–1720, 2020. ISSN 15582914.

[27] STATBEL. Employment and unemployment, 2023. URL `https://statbel.fgov.be/en/themes/work-training/labour-market/employment-and-unemployment#figures`. `https://statbel.fgov.be/en/themes/work-training/labour-market/employment-and-unemployment#figures`.

[28] STATBEL. Volwasseneneducatie, 2023. URL https://statbel.fgov.be/nl/themas/werk-opleiding/opleidingen-en-onderwijs/volwasseneneducatie#figures. https://statbel.fgov.be/nl/themas/werk-opleiding/opleidingen-en-onderwijs/volwasseneneducatie#figures.

[29] STATBEL. Transitions on the labour market, 2023. URL https://statbel.fgov.be/en/themes/work-training/labour-market/transitions-labour-market#figures. https://statbel.fgov.be/en/themes/work-training/labour-market/transitions-labour-market#figures.

[30] Hannah Van Borm and Stijn Baert. Diving in the Minds of Recruiters: What Triggers Gender Stereotypes in Hiring? *SSRN Electronic Journal*, (15261), 2022. doi: 10.2139/ssrn.4114837.

[31] Blake Woodworth, Suriya Gunasekar, Mesrob I. Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. In *Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pages 1920–1953. PMLR, 2017.

[32] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P. Gummadi. Fairness Constraints: Mechanisms for Fair Classification. In *20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970. PMLR, 2017.

[33] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 325–333, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL https://proceedings.mlr.press/v28/zemel13.html.

[34] Hantian Zhang, Ki Hyun Tae, Jaeyoung Park, Xu Chu, and Steven Euijong Whang. iflipper: Label flipping for individual fairness, 2022.

[35] Junzhe Zhang and Elias Bareinboim. Fairness in decision-making the causal explanation formula. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 2037–2045, 2018.