ΕΛΛΗΝΙΚΗ
ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΜΑΚΕΔΟΝΙΑΣ

Business Analytics
and Data Science
Πρόγραμμα Μεταπτυχιακών Σπουδών στην
ΑΝΑΛΥΤΙΚΗ ΤΩΝ ΕΠΙΧΕΙΡΗΣΕΩΝ ΚΑΙ ΕΠΙΣΤΗΜΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ
Τμήμα Οργάνωσης και Διοίκησης Επιχειρήσεων

Πρόγραμμα Μεταπτυχιακών Σπουδών στην Αναλυτική των Επιχειρήσεων και Επιστήμη των Δεδομένων, Τμήμα Οργάνωσης και Διοίκησης Επιχειρήσεων

Διπλωματική Εργασία:

«Fairness in Predictive Analytics: Integrating Bias Detection, Mitigation, and Explainability in Machine Learning Models» ,
Αθηνά Μούσια

Επιβλέπων Καθηγητής: Ταραμπάνης Κωνσταντίνος

Υποβλήθηκε ως απαιτούμενο για την απόκτηση του μεταπτυχιακού διπλώματος στην Αναλυτική των Επιχειρήσεων και Επιστήμη των Δεδομένων

Θεσσαλονίκη, Ιανουάριος 2024

Abstract

This thesis presents a critical analysis of machine learning algorithms within the realm of educational predictive analytics, with a particular emphasis on detecting and mitigating socio-economic biases. In an era where machine learning profoundly impacts decision-making processes, this study highlights the imperative of developing fair and unbiased models. The research employs an analytical framework comprising Bias Detection techniques to identify inherent biases in algorithms or datasets, Bias Mitigation models to adjust these elements and reduce socio-economic disparities, and Explainability methods, particularly LIME and SHAP, to elucidate the decision-making mechanisms of the algorithms. These methodologies are pivotal in recognizing biases and assessing the effectiveness of mitigation strategies.

The study emphasises the need for ethical considerations in the application of machine learning techniques. It advocates for the continuous development and refinement of predictive models to uphold ethical standards and foster equity. The findings lay the groundwork for future explorations into more advanced methods for ensuring fairness and transparency in machine learning across different domains.

Content

List of Figures

1. Introduction

In the field of data science, this thesis embarks on a critical exploration of biases in machine learning models, a subject of paramount importance in our increasingly automated world. The real-world impact of these biases is both far-reaching and profound, affecting key sectors like finance, healthcare, and employment. These biases, often subtle and unnoticed, can lead to unfair, unethical outcomes, posing significant challenges to both individuals and societies.

The central aim of this research is to investigate the nature and extent of biases present in machine learning models. This involves a deep dive into the mechanisms through which biases are introduced and perpetuated in these models. The study is driven by pivotal questions: How can biases within machine learning models be reliably detected? What are the most effective strategies to mitigate these biases, ensuring the models' integrity and accuracy? These questions are crucial in navigating the complexities of ethical AI.

The significance of this research extends beyond the academic sphere, touching on the crucial aspect of ethical responsibility in technology. By addressing the issue of bias in machine learning, this thesis aims to contribute to the development of fairer, more equitable technological solutions, a step forward in responsible AI. This research holds the potential to influence policy-making, shape ethical guidelines, and drive innovation in the field, ensuring that technological advancements are aligned with societal values.

Structured in a comprehensive and systematic manner, the thesis begins by identifying the various forms of biases in machine learning models, followed by an exploration of methodologies and strategies to detect and mitigate these biases. Subsequently, the thesis evaluates the effectiveness of these strategies in real-world scenarios, providing an understanding of the challenges and potential solutions in this domain. This structured approach ensures a holistic understanding of the issue, offering insightful perspectives on the ethical implications and practical solutions for bias in machine learning, ultimately guiding the field towards a more equitable future.

2. Exploratory Data Analysis

The dataset that is used through the study, is supported by program SATDAP - Capacitação da Administração Pública under grant POCI-05-5762-FSE-000191, Portugal.It is sourced from a higher education institution and is compiled from various separate databases. It encompasses student records from diverse undergraduate programs such as agronomy, design, education, nursing, journalism, management, social service, and technologies.

The dataset includes information available at the point of student enrollment, including academic trajectory, demographic details, and socio-economic factors. Additionally, it incorporates data concerning students' academic performance at the conclusion of their first and second semesters.

The dataset consists of 4.424 rows and 37 variables. The variables of the dataset are described in Table 1.

In preparing the dataset for analysis, several crucial preprocessing steps were undertaken. These steps are outlined below.

Firstly, the target variable has three distinct categories: 'Dropout,' 'Graduate,' and 'Enrolled.' However, for the purpose of this analysis, which aims to predict whether a student will graduate or dropout from the university and identify biases across various attributes, only 'Dropout' and 'Graduate' values were retained, with the 'Enrolled' value being excluded. Additionally, these categorical values are replaced with numerical ones, where 'Dropout' is denoted as 0 and 'Graduate' as 1.

Additionally, every variable in the dataset had its punctuation changed, using underscores (_). This was done to make sure they work well with the functions of the libraries it is used.

Furthermore, the 'Admission grade' variable was discretized into five bins to facilitate analysis. These bins were defined as follows: 1 for grades ranging from 0.0 to 114.0, 2 for grades from 114.0 to 133.0, 3 for grades from 133.0 to 152.0, 4 for grades from 152.0 to 171.0, and 5 for grades exceeding 171.0. The purpose of this binning was to investigate potential biases among students in terms of graduation or dropout based on their admission grades.

Likewise, the variable 'Age at enrollment' was discretized into five bins to aid in the analysis. The bin boundaries were set as follows: 1 for ages ranging from 0 to 21, 2 for ages from 21 to 30, 3 for ages from 30 to 45, 4 for ages from 45 to 60, and 5 for ages exceeding 60. This binning was essential to examine potential biases in students' likelihood of graduating or dropping out based on their age.

The exploration of the distribution of key variables provides valuable insights into the characteristics of the student population, their academic background, and the socio-economic context.

In the analysis of numerical variables as shown in Figure 1, Previous Qualification (Grade) and Admission Grade both exhibit a right-skewed distribution, indicating a majority of students with lower to moderate grades and a minority achieving very high grades, reflecting the diverse academic capabilities and the institution's admission standards. Additionally, Age at Enrollment is also right-skewed, typical of higher education populations predominantly comprising younger students transitioning from secondary schooling. Uniquely, the Unemployment Rate shows a multimodal distribution, hinting at fluctuating economic conditions over time which could impact the student community.

The examination of categorical variables reveals distinct patterns. Marital Status is predominantly unmarried or single students, common in younger demographics. Gender distribution within the dataset is fairly balanced, showcasing a gender-diverse student body essential for an inclusive educational environment. Regarding financial support, a larger portion of students are without scholarships, shedding light on the socio-economic backgrounds and financial support structures available to them.

Lastly, the Target variable, categorizing student outcomes as "Graduate" or "Dropout," presents a relatively even split. This balanced distribution offers insight into the varied academic success rates, emphasizing the need to understand and address the diverse factors contributing to different student outcomes in the educational journey.

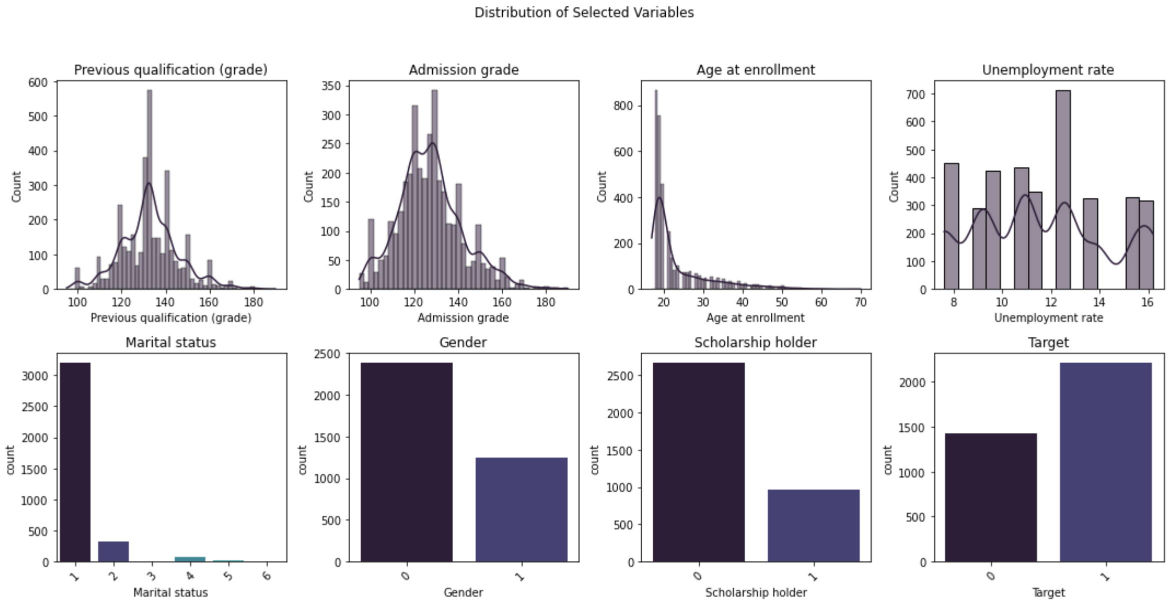


Figure 1: Distribution of selected continuous and categorical variables

3. Metrics

In the realm of data science, the quantification and assessment of algorithmic performance and ethical considerations are pivotal. This section delves into two critical dimensions of this quantification: Evaluation Metrics and Fairness Metrics. Each of these facets plays a fundamental role in not only guiding the development of machine learning models but also in ensuring their alignment with ethical standards and societal needs.

This section captures the dual objectives of achieving high performance in machine learning models while ensuring that they operate within an ethical framework. This balance is crucial for the advancement of data science as a discipline that not only excels in technical proficiency but also in social responsibility.

3.1 Evaluation Metrics

Evaluation Metrics are a set of performance measures used to assess the overall quality and effectiveness of machine learning models, particularly in the context of binary classification tasks. These metrics help quantify how well a model performs at distinguishing between two classes, typically a positive class and a negative class. Evaluation metrics provide valuable insights into various aspects of a model's performance, allowing data scientists and machine learning practitioners to make informed decisions about model selection, fine-tuning, and deployment.

These metrics are fundamentally grounded in the classification of instances into primary categories, based on their actual and predicted values.

These categories are:

- **P (Positive Cases)**: This denotes the number of instances where the outcome of interest (Y=1) is present. These are the 'positive' cases that the model aims to predict, such as the occurrence of a disease in medical diagnostics or a successful outcome in other predictive scenarios.
- **N (Negative Cases)**: Contrary to P, N represents the count of 'negative' cases where the outcome of interest (Y=0) is absent. These instances are those where the event or condition the model is designed to detect does not occur.
- **Total Number of Instances (P + N)**: This metric is the summation of all positive (P) and negative (N) instances within the dataset, providing the total count of instances under evaluation.
- **True Positives (TP)**: This category comprises instances that are correctly identified as positive by the model, meaning that they are actual positive cases (Y=1) and the model successfully predicts them as such.
- **True Negatives (TN)**: Analogously, TN represents instances that are both actually and predictably negative. These are the cases where the outcome is absent (Y=0), and the model accurately classifies them as negative.
- **False Positives (FP)**: Often referred to as Type I errors, these are instances where the model incorrectly predicts positive outcomes. In other words, these are cases that are actually negative (Y=0) but are erroneously classified as positive.

- **False Negatives (FN)**: Known as Type II errors, these instances are those where the model fails to identify positive cases. They represent actual positive cases (Y=1) that the model mistakenly categorises as negative.

3.1.1 Base rate

The base rate is the prior probability of the event occurring without any additional information or predictive factors. In this context, the base rate refers to the probability of the event Y=1 happening in the absence of any specific data or predictive model.

In many real-world applications, understanding and accounting for the base rate is important, as it provides a baseline or starting point for evaluating the significance of predictive models or diagnostic tests. It helps you assess whether a model or test is adding meaningful information beyond what can be inferred from the base rate alone. If the base rate is very low, even a highly accurate model may not be very useful, as the event is rare to begin with.

The equation of base rate is $Pr(Y = 1) = \frac{P}{P+N}$.

$Pr(Y = 1)$ represents the probability that a particular event or condition Y is true, specifically Y being equal to 1. In many contexts, Y=1 is used to represent the presence or occurrence of an event or outcome, while Y=0 would represent the absence or non-occurrence of that event.

3.1.2 True negative rate

The True Negative Rate (TNR), also known as specificity, is a binary classification performance metric that measures the proportion of actual negative instances (Y=0) that were correctly predicted as negative (true negatives) by a machine learning model.

The formula for the True Negative Rate is: $TNR = \frac{TN}{N}$.

The True Negative Rate quantifies the model's ability to correctly identify negative instances. It measures how effective the model is at avoiding false alarms or false positive predictions.

A high TNR indicates that the model is effective at correctly identifying most of the negative instances, meaning it has high specificity. A low TNR suggests that the model is failing to correctly identify many of the negative instances, leading to a significant number of false positive predictions.

3.1.3 True positive rate

The True Positive Rate (TPR), also known as sensitivity, is a binary classification performance metric that measures the proportion of actual positive instances (Y=1) that were correctly predicted as positive (true positives) by a machine learning model.

The formula for the True Positive Rate is: $TPR = \frac{TP}{P}$.

The True Positive Rate quantifies the model's ability to correctly identify positive instances. It measures the model's sensitivity in detecting the positive cases in the dataset.

A high TPR indicates that the model is effective at correctly identifying most of the positive instances, meaning it has high sensitivity or recall. A low TPR suggests that the model is failing to correctly identify many of the positive instances, leading to a significant number of false negatives.

3.1.4 False negative rate

The False Negative Rate (FNR) is a binary classification performance metric that measures the proportion of actual positive instances (Y=1) that were incorrectly predicted as negative (false negatives) by a machine learning model.

The formula for FNR is: $FNR = \frac{FN}{P}$.

The False Negative Rate is a measure of how effective the model is at identifying positive instances. It quantifies the rate at which the model fails to correctly identify instances that are actually positive.

A low FNR indicates that the model is effective at correctly identifying most of the positive instances. A high FNR suggests that the model is missing a significant portion of the positive instances, leading to a substantial number of false negatives. This may be undesirable in situations where correctly identifying all positive instances is crucial.

FNR is particularly relevant in applications where missing positive instances can have significant implications, such as in medical diagnostics, where failing to detect a disease can be life-threatening, or in fraud detection, where failing to identify fraudulent transactions can result in financial losses.

3.1.5 False positive rate

The False Positive Rate (FPR) is a binary classification performance metric that measures the proportion of actual negative instances (Y=0) that were incorrectly predicted as positive (false positives) by a machine learning model.

The formula for the False Positive Rate is: $FPR = \frac{FP}{N}$.

The False Positive Rate quantifies the rate at which the model incorrectly identifies instances as positive when they are actually negative.

A low FPR indicates that the model is effective at correctly identifying most of the negative instances, meaning it has high specificity. A high FPR suggests that the model is incorrectly classifying a significant portion of negative instances as positive, leading to a substantial number of false positives. This may be undesirable in situations where false positives have significant consequences.

FPR is particularly relevant in applications where avoiding false positives is critical. For example, in spam email detection, a high FPR means that legitimate emails are incorrectly classified as spam, leading to important emails being missed. Reducing FPR is often a priority in such cases to improve the model's specificity and reduce the occurrence of false alarms.

3.1.6 Error rate

The error rate (ERR) is a binary classification performance metric that measures the overall rate at which a machine learning model makes errors in its predictions. It is also known as the classification error rate. The formula for error rate is as follows:

$$ERR = \frac{False\ Positives + False\ Negatives}{Total\ Number\ of\ Instances}$$

The error rate quantifies the overall proportion of instances that were misclassified by the model. It takes into account both false positives and false negatives and provides a general measure of prediction accuracy.

A low error rate indicates that the model is making correct predictions for the majority of instances, meaning it has high accuracy.A high error rate suggests that the model is making a significant number of incorrect predictions, which could be due to various factors such as imbalanced data, model complexity, or the choice of the threshold for classification.

The error rate is a straightforward metric for understanding the overall quality of a binary classification model. However, it may not always be the most informative metric, especially when dealing with imbalanced datasets, where one class is much more prevalent than the other. In such cases, other metrics like precision, recall, F1-score, or the area under the ROC curve (AUC-ROC) may provide a more nuanced evaluation of model performance.

3.1.7 Accuracy

Accuracy is a measure of how many of the total instances were correctly classified by the model. The formula for accuracy, in general, is:

$$Accuracy = \frac{(True\ Positives + True\ Negatives)}{(Total\ Number\ of\ Instances)}.$$

A high accuracy value indicates that the model is making correct predictions for a large portion of the dataset, while a low accuracy suggests that the model is making more incorrect predictions.

3.1.8 Balanced Accuracy

Balanced Accuracy is a performance metric used in binary classification tasks, particularly when dealing with imbalanced datasets where one class significantly outweighs the other in terms of the number of instances. It provides a balanced assessment of a model's ability to correctly classify both the positive and negative classes.

The Balanced Accuracy is calculated as the average of two important metrics: **Sensitivity** (True Positive Rate) and **Specificity** (True Negative Rate). These metrics are crucial in understanding how well a model performs across different classes and help account for class imbalances.

Here's a breakdown of Balanced Accuracy and its components:

Sensitivity (True Positive Rate) measures the proportion of actual positive instances that were correctly predicted as positive by the model. The formula for Sensitivity is: $Sensitivity \ = \frac{TP}{TP + FN}$.

Specificity (True Negative Rate) measures the proportion of actual negative instances that were correctly predicted as negative by the model. The formula for Sensitivity is: $Sensitivity \ = \frac{TN}{TN + FP}$.

Balanced Accuracy is then calculated as the average of Sensitivity and Specificity: $Balanced\ Accuracy \ = \frac{(Sensitivity + Specificity)}{2}$.

The advantage of Balanced Accuracy is that it provides a fair and balanced assessment of a model's performance, especially in cases where one class is rare, and the model might be heavily biassed towards the majority class. It helps prevent overly optimistic evaluations in imbalanced datasets.

A high Balanced Accuracy indicates that the model is effective at classifying both positive and negative instances, while a low Balanced Accuracy suggests that the model may struggle with one or both classes. It is a valuable metric when you want a comprehensive view of classification performance that considers both the detection of positive cases and the accurate identification of negative cases, regardless of class imbalances.

3.1.9 Precision

Precision is a binary classification metric that evaluates the accuracy of a model's positive predictions, specifically focusing on how many of the instances it predicted as positive were actually correct. In other words, it measures the proportion of true positive predictions (correctly identified positive instances) out of all the positive predictions made by the model.

The precision metric is especially relevant when minimising false positives (Type I errors) is a priority, such as in applications where the cost of making incorrect positive predictions are high.

Here's the formula for precision: $Precision = \frac{TP}{TP+FP}$.

Precision answers the question: "Of all the instances the model predicted as positive, how many were truly positive?" A high precision value means that the model is making accurate positive predictions, with fewer false alarms. In other words, it has a low rate of false positives.

A low precision value suggests that the model makes many positive predictions, but a significant portion of those predictions are incorrect. This could indicate a high rate of false positives, which may be undesirable in scenarios where false alarms have serious consequences.

3.1.10 Recall

Recall, also known as Sensitivity or True Positive Rate, is a binary classification metric that assesses a model's ability to correctly identify all actual positive instances

(Y=1) out of the total number of actual positives. In other words, recall quantifies the model's ability to find and "recall" as many positive instances as possible.
Here's the formula for recall: $Recall = \frac{TP}{TP+FN}$.

Recall answers the question: "Of all the actual positive instances, how many did the model correctly identify?" A high recall value means that the model is effective at capturing most of the positive instances, minimising false negatives. In other words, it has a low rate of missing actual positives.

A low recall value suggests that the model is missing a significant portion of the actual positive instances, leading to a high rate of false negatives. This may be undesirable in scenarios where correctly identifying all positive instances is crucial, such as in medical diagnostics or security applications.

3.1.11 F1-score

The F1-Score is a binary classification metric that provides a balanced assessment of a model's performance by combining precision and recall into a single value. These two metrics focus on different aspects of classification performance: precision emphasises the accuracy of positive predictions, while recall emphasises the model's ability to capture positive instances.

The F1-Score is calculated using the harmonic mean of precision and recall, as per the formula $F1\ score = 2 \cdot (Precision \cdot Recall) / (Precision + Recall)$. The harmonic mean is utilised because it gives more weight to the lower of the two values, encouraging a balance between precision and recall. This means that if either precision or recall is significantly lower than the other, the F1-Score will reflect this lower value, promoting a balanced approach in which both false positives and false negatives are minimised.

The F1-Score is particularly valuable when you need to find a trade-off between the accuracy of positive predictions and the model's ability to capture as many positive instances as possible. It is often used in situations where the consequences of false positives and false negatives are approximately equal, such as in medical diagnostics, where both missing a critical diagnosis and incorrectly diagnosing a healthy patient have serious implications. By considering both precision and recall, the F1-Score helps you evaluate the model's performance from a more comprehensive and balanced perspective, striking a harmony between these two vital aspects of binary classification performance.

3.1.12 ROC-AUC score

The ROC-AUC score quantifies the overall quality of the model's predictions across different classification thresholds, focusing on the trade-off between the True Positive rate and the False Positive rate.

To calculate the ROC-AUC score, you first construct the ROC curve. The ROC curve is a graphical representation that illustrates the model's performance at various classification thresholds. It plots the True Positive Rate (TPR) on the y-axis against the False Positive Rate (FPR) on the x-axis as the threshold for classifying positive instances is varied.

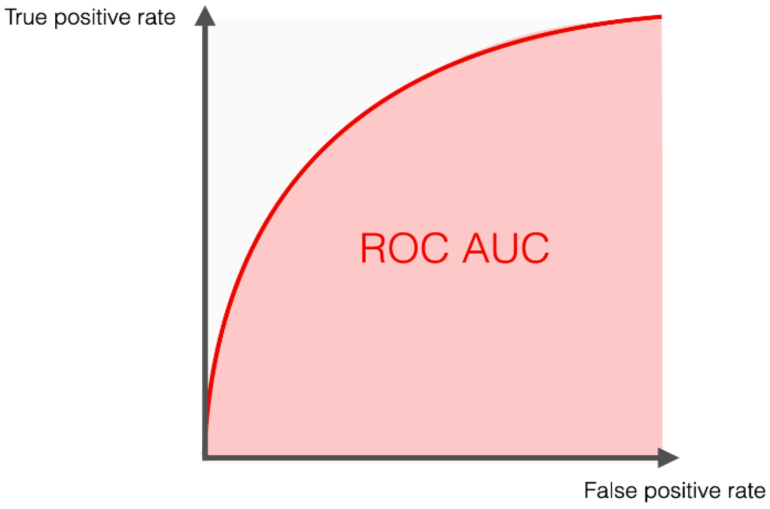


Figure 2: ROC AUC curve representation

The ROC-AUC score is calculated by measuring the area under the ROC curve. It quantifies the model's ability to distinguish between the two classes. The higher the

ROC-AUC score, the better the model's discriminatory power. A score of 0.5 indicates that the model's predictions are no better than random chance. A score of 1.0 indicates a perfect model that can perfectly separate the two classes.

The ROC-AUC score is a valuable metric because it provides an aggregated measure of the model's ability to rank positive instances higher than negative instances across various classification thresholds. It is particularly useful in situations where the class distribution is imbalanced or when you want to evaluate a model's overall discrimination capability without specifying a single classification threshold.

3.2 Fairness Metrics

The Fairness Metrics shifts the focus towards the ethical aspects of data science. Here, the discussion centres around the development and implementation of metrics that ensure models do not perpetuate biases and inequalities present in data. This part explores various fairness concepts such as demographic parity, equality of opportunity, and individual fairness, among others.In summary, this section emphasises the importance of fairness in model development, illustrating how data science can be leveraged for socially responsible outcomes.

The annotated terms provide a comprehensive framework for understanding the differential impacts of a predictive model on various groups, typically delineated by privileged and unprivileged statuses based on certain protected attributes.

- $Pr(Y = 1| D = unprivileged)$: This metric indicates the probability of a positive outcome (Y=1) for the unprivileged group. In this context, 'unprivileged' refers to a group that may be at a systemic disadvantage or underrepresented in the dataset. This measure helps in understanding the model's propensity to predict positive outcomes for those who are typically marginalised.
- $Pr(Y = 1| D = privileged)$: Conversely, this measures the probability of a positive outcome for the privileged group. Here, 'privileged' refers to individuals

or groups who, due to systemic advantages or overrepresentation, might have a higher likelihood of receiving positive predictions from the model.

- $FPR_D = unprivileged$: The False Positive Rate for the unprivileged group indicates the proportion of instances where the model incorrectly predicts positive outcomes (false positives) for cases that are actually negative, specifically for the unprivileged group. This metric is crucial for assessing the model's bias in overestimating positive outcomes for this group.
- $FPR_D = privileged$: This is the False Positive Rate for the privileged group, representing the proportion of false positives within this group. A comparison between the FPR of privileged and unprivileged groups can highlight disparities in model performance.
- $TPR_D = unprivileged$: This metric, the True Positive Rate for the unprivileged group, quantifies the proportion of correct positive predictions (true positives) within this group. It is a measure of the model's ability to accurately identify positive outcomes for the unprivileged group.
- $TPR_D = privileged$: Similarly, this represents the True Positive Rate for the privileged group, indicating the proportion of true positives within this group. This metric can be used to assess whether the model favors the privileged group in correctly identifying positive cases.
- $FNR_D = unprivileged$: This metric represents the False Negative Rate for the unprivileged group, showing the proportion of actual positive instances incorrectly predicted as negative. This rate is crucial for understanding the model's tendency to overlook positive outcomes in the unprivileged group.
- $FNR_D = privileged$: Lastly, this is the False Negative Rate for the privileged group. It quantifies the proportion of false negatives within this group, helping to assess whether the model disproportionately misses positive outcomes for the privileged group.
- $ERR_D = unprivileged$: This denotes the overall error rate for the unprivileged group, calculated as the proportion of all misclassified instances (both false positives and false negatives) in this group. This metric is vital for understanding the model's overall accuracy in predicting outcomes for the unprivileged group.

- $ERR_D = privileged$: The error rate for the privileged group, similarly, measures the proportion of misclassified instances in this group. Discrepancies in error rates between privileged and unprivileged groups are indicative of model bias.

3.2.1 Consistency

Consistency measures how similar the labels are for similar instances. The equation of consistency is given below:

$$1 - \frac{1}{n \cdot n\_neighbors} \cdot \sum_{i=1}^{n} \left| \widehat{y_i} - \sum_{j \in N_{n\_neighbors(x_i)}} \widehat{y_j} \right|$$

- i is an index ranging from 1 to n, where n is the number of data points in your dataset.
- y_i represents the predicted value for the i_{th} data point.
- j is an index ranging over the nearest neighbours for each data point.
- x_i represents the features of the i_{th} data point.
- $N_{n\_neighbors(x_i)}$ represents the set of nearest neighbours for the i_{th} data point, typically determined using a distance metric like Euclidean distance.
- $\widehat{y_j}$ represents the predicted value for the j-th nearest neighbour of the i_{th} data point.

Consistency calculates the absolute difference between the predicted value for a data point and the average predicted value of its nearest neighbours. This is done for all data points in the dataset, and the results are summed up.

The expression as a whole computes a consistency measure for the K-NN regression model. It quantifies how similar the predicted values for each data point are to the average of the predicted values of its nearest neighbours. If the predictions are highly consistent (i.e., they are similar to the averages of their neighbours), this measure will be closer to 1. If the predictions are inconsistent or widely scattered, the measure will be closer to 0.

In practice, a higher consistency measure indicates that the model is making predictions that are more coherent and conform to the local structure of the data, which is generally desirable in K-NN regression. This measure can be useful for evaluating the quality of K-NN models and for selecting an appropriate number of neighbours (*n_neighbors*) to achieve the desired level of consistency in predictions.

3.2.2 Disparate Impact

Disparate impact measures whether a particular decision or prediction has a different impact on different groups, especially with regard to protected attributes such as gender, race, or age.

The equation of disparate impact is given below: $\frac{Pr(Y=1 | D=unprivileged)}{Pr(Y=1 | D=privileged)}$

In other words, disparate impact refers to the ratio of the rate of a positive outcome for the disfavored group to the rate of a positive outcome for the favored group.

If disparate impact is equal to 1, it indicates equal treatment or no disparate impact. If ratios are greater than 1, it signifies a higher likelihood of positive outcomes for unprivileged group compared to privileged group, which could indicate potential bias in the decision-making process. If ratios are less than 1, it signifies a higher likelihood of positive outcomes for privileged group compared to unprivileged group, which could indicate potential bias in the decision-making process.

3.2.3 Statistical Parity Difference

The Statistical Parity Difference is a fairness metric used to assess and quantify potential bias in the predictions of a binary classification model with respect to a protected attribute (denoted as D).

The formula for the Statistical Parity Difference is:

$$Pr(\widehat{Y} = 1|D = unprivileged) - Pr(\widehat{Y} = 1|D = privileged)$$

The Statistical Parity Difference quantifies the difference in predicted positive outcomes between the unprivileged group and the privileged group. It tells you whether there is a disparity in how the model's predictions are made based on the protected attribute.

If the Statistical Parity Difference is close to 0, it suggests that the model's predictions are fairly consistent between the unprivileged and privileged groups. In other words, the likelihood of a positive outcome does not significantly differ based on the protected attribute. If the Statistical Parity Difference is significantly greater than 0, it indicates that the unprivileged group is less likely to receive positive predictions compared to the privileged group. This suggests potential bias that favours the privileged group. If the Statistical Parity Difference is significantly less than 0, it indicates that the unprivileged group is more likely to receive positive predictions compared to the privileged group. This, too, suggests potential bias, but in favour of the unprivileged group.

3.2.4 Average Odds Difference

This metric evaluates how the model's false positive and true positive rates differ between two groups, typically based on a protected attribute.

The formula of Average Odds Difference is:

$$\frac{1}{2}[(FPR_D = unprivileged - FPR_D = privileged) +$$
$$(TPR_D = unprivileged - TPR_D = privileged))]$$

The formula calculates the average of the differences between the false positive rates and the true positive rates for the unprivileged and privileged groups. The metric measures model's performance in terms of both false positives and true positives differs between the two groups.

If the value is close to zero, it suggests that there is relatively little difference in false positive rates and true positive rates between the unprivileged and privileged groups. In other words, the model is making predictions that are fairly consistent across both groups in terms of false positives and true positives. If the value is significantly greater than zero, it indicates that the unprivileged group is experiencing higher false positive rates and lower true positive rates compared to the privileged group, which could be indicative of bias in model predictions. If the value is significantly less than zero, it suggests that the unprivileged group has lower false positive rates and higher true positive rates compared to the privileged group, which could also indicate a form of bias.

3.2.5 Average Absolute Odds Difference

This metric measures the absolute differences in False Positive rate and True Positive rate between two groups based on a protected attribute.

The formula of Absolute Odds Difference is:

$$\frac{1}{2}[|FPR_D = unprivileged - FPR_D = privileged| + |TPR_D = unprivileged - TPR_D = privileged|]$$

If the value is close to zero, it suggests that there is relatively little difference in false positive rates and true positive rates between the unprivileged and privileged groups, with any differences being relatively balanced. If the value is significantly greater than zero, it indicates that there are substantial differences in false positive rates and true positive rates between the two groups, and these differences are of significant magnitude. This could be indicative of bias in model predictions.

3.2.6 Error rate difference

The error rate difference (ERR_D) is a fairness metric used to assess disparities in the error rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute.

The formula for error rate difference is:

$ERR_D = ERR_D = unprivileged\ -\ ERR_D = privileged.$

The error rate difference quantifies the difference in error rates between the unprivileged and privileged groups. It helps evaluate whether there is a disparate impact in terms of misclassification rates based on the protected attribute.

If the ERR_D value is close to zero, it suggests that there is relatively little difference in error rates between the unprivileged and privileged groups, indicating that the model's prediction errors are fairly consistent across both groups. If the ERR_D value is significantly greater than zero, it indicates that the unprivileged group has a higher error rate compared to the privileged group, suggesting potential bias or disparities in model performance. If the ERR_D value is significantly less than zero, it suggests that the unprivileged group has a lower error rate compared to the privileged group, which could also indicate disparities in a different direction.

3.2.7 Error rate ratio

The ratio of error rates (ERR_D) is a fairness metric used to assess disparities in the error rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute.

The formula for the ratio of error rates is: $ERR_D = \frac{ERR_D=unprivileged}{ERR_D=privileged}$.

The ERR_Dmetric calculates the ratio of the error rate for the unprivileged group to the error rate for the privileged group. It measures the extent to which error rates differ between these two groups based on the protected attribute.

If the ERR_D value is close to 1, it suggests that the error rates are roughly the same for both the unprivileged and privileged groups, indicating that the model's prediction errors are fairly consistent between the two groups. If the ERR_D value is significantly greater than 1, it indicates that the unprivileged group has a higher error rate compared to the privileged group, suggesting potential bias in model performance. In this case, a value of 2, for example, means that the unprivileged group has an error rate twice as high as the privileged group. If the ERR_D value is significantly less than 1, it suggests that the unprivileged group has a lower error rate compared to the privileged group. This, too, could indicate disparities in a different direction.

This metric is particularly useful in cases where you want to understand the relative magnitude of error rate differences between different groups.

3.2.8 False negative rate difference

The False Negative Rate Difference (FNR_D) is a fairness metric used to assess disparities in the false negative rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute.
The formula for the False Negative Rate Difference is:

$FNR_D = FNR_D = unprivileged - FNR_D = privileged$.

The FNR_D metric calculates the difference in false negative rates between the unprivileged and privileged groups. It helps evaluate whether there is a disparate impact in terms of false negatives based on the protected attribute.

If the FNR_D value is close to zero, it suggests that there is relatively little difference in false negative rates between the unprivileged and privileged groups, indicating that the model's ability to correctly identify positive instances is fairly consistent between the two groups.If the FNR_D value is significantly greater than zero, it indicates that the unprivileged group experiences a higher rate of false negatives compared to the privileged group. This suggests potential bias or disparities in the model's ability to identify positive instances.If the FNR_D value is significantly less than zero, it suggests that the unprivileged group has a lower rate of false negatives compared to the privileged group, which could also indicate disparities, albeit in a different direction.

3.2.9 False negative rate ratio

The False Negative Rate Ratio (FNR_D) is a fairness metric used to assess disparities in the false negative rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute.

The formula for the False Negative Rate Ratio is:

$$FNR_D = \frac{FNR_D=unprivileged}{FNR_D=privileged} \ .$$

The FNR_D metric calculates the ratio of the false negative rate for the unprivileged group to the false negative rate for the privileged group. It measures the extent to which false negative rates differ between these two groups based on the protected attribute.

If the FNR_D value is close to 1, it suggests that the false negative rates are roughly the same for both the unprivileged and privileged groups, indicating that the model's ability to correctly identify positive instances (low false negatives) is fairly consistent between the two groups. If the FNR_D value is significantly greater than 1, it indicates that the unprivileged group has a higher false negative rate compared to the privileged group, suggesting potential bias or disparities in the model's ability to identify positive instances. In this case, a value of 2, for example, means that the unprivileged group has a false negative rate twice as high as the privileged group.

If the FNR_D value is significantly less than 1, it suggests that the differences in false negative rates between the unprivileged and privileged groups are substantial, and the unprivileged group has a lower false negative rate compared to the privileged group.

3.2.10 False positive rate difference

The False Positive Rate Difference (FPR_D) is a fairness metric used to assess disparities in the false positive rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute such as gender, race, or age. The formula for the False Positive Rate Difference is:

$FPR_D = FPR_D = unprivileged - FPR_D = privileged\,.$

The FPR_D metric calculates the difference in false positive rates between the unprivileged and privileged groups. It helps evaluate whether there is a disparate impact in terms of false positives based on the protected attribute.

If the FPR_D value is close to zero, it suggests that there is relatively little difference in false positive rates between the unprivileged and privileged groups, indicating that the model's ability to correctly identify negative instances (low false positives) is fairly consistent between the two groups. If the FPR_D value is significantly greater than zero, it indicates that the unprivileged group experiences a higher rate of false positives compared to the privileged group, suggesting potential bias or disparities in the model's ability to identify negative instances. If the FPR_D value is significantly less than zero, it suggests that the differences in false positive rates between the unprivileged and privileged groups are substantial, and the unprivileged group has a lower false positive rate compared to the privileged group, which could also indicate disparities in a different direction.

3.2.11 False positive rate ratio

The False Positive Rate Ratio (FPR_D) is a fairness metric used to assess disparities in the false positive rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute such as gender, race, or age.

The formula for the False Positive Rate Ratio is:

$$FPR_D = \frac{FPR_D = unprivileged}{FPR_D = privileged}$$

The FPR_Dmetric calculates the ratio of the false positive rate for the unprivileged group to the false positive rate for the privileged group. It measures the extent to which false positive rates differ between these two groups based on the protected attribute.

If the FPR_Dvalue is close to 1, it suggests that the false positive rates are roughly the same for both the unprivileged and privileged groups, indicating that the model's ability to correctly identify negative instances (low false positives) is fairly consistent between the two groups.If the FPR_Dvalue is significantly greater than 1, it indicates that the unprivileged group has a higher false positive rate compared to the privileged group, suggesting potential bias or disparities in the model's ability to identify negative instances. In this case, a value of 2, for example, means that the unprivileged group has a false positive rate twice as high as the privileged group. If the FPR_Dvalue is significantly less than 1, it suggests that the differences in false positive rates between the unprivileged and privileged groups are substantial, and the unprivileged group has a lower false positive rate compared to the privileged group.

3.2.12 Equal Opportunity Difference or True positive rate difference

The True Positive Rate Difference (TPR_D) is a fairness metric used to assess disparities in the true positive rates of a binary classification model between unprivileged and privileged groups, typically based on a protected attribute.
The formula for the True Positive Rate Difference is:

$TPR_D = TPR_D = unprivileged \ - \ TPR_D = privileged\,.$

The TPR_D metric calculates the difference in true positive rates between the unprivileged and privileged groups. It helps evaluate whether there is a disparate impact in terms of true positives based on the protected attribute.

If the TPR_D value is close to zero, it suggests that there is relatively little difference in true positive rates between the unprivileged and privileged groups, indicating that the model's ability to correctly identify positive instances (high true positives) is fairly consistent between the two groups. If the TPR_D value is significantly greater than zero, it indicates that the unprivileged group has a lower true positive rate compared to the privileged group, suggesting potential bias in the model's ability to identify positive instances. If the TPR_D value is significantly less than zero, it suggests that the unprivileged group has a higher true positive rate compared to the privileged group, which could also indicate disparities in a different direction.

4. Bias Detection

Bias detection is the process of identifying and assessing biases that may exist within a dataset, algorithm, or decision-making system. Bias, in this context, refers to systematic and consistent deviations in the data or the processes applied to it, which can lead to unfair or skewed outcomes. Detecting bias is crucial for ensuring fairness, equity, and transparency in various domains, particularly in fields related to machine learning, artificial intelligence, data analysis, and decision-making.

Bias often originates from the data itself. It can manifest in various forms, such as underrepresentation or overrepresentation of certain groups, inaccuracies, or misleading information. Detecting biased data involves analysing the dataset to identify patterns or discrepancies that could potentially lead to biased outcomes.

In the context of bias detection, we are referring to two types of bias detection, pre-modeling and post-modeling bias detection. The difference between these two types is that pre-modelling bias detection involves identifying bias before a model is built. It primarily focuses on the data and how it's prepared for the model. On the other hand, post-modeling bias detection involves evaluating and correcting for biases after the model has been developed. Both pre-modeling and post-modeling bias detection require a multifaceted approach, involving technical, ethical, and social considerations. They are essential for building models that are not only effective but also equitable.

4.1 Pre-Modeling Bias Detection

In the realm of pre-modeling bias detection in data analysis, various statistical methods are employed to uncover and understand the underlying structure and potential biases within datasets. These methods are critical in ensuring the reliability and fairness of subsequent machine learning models. Understanding and addressing biases at this stage helps in developing models that make fair and accurate predictions. Among these methods, correlation analysis, Kernel Density Estimation (KDE) or hypothesis tests are particularly noteworthy.

Firstly, Spearman correlation coefficient is a statistical method used to measure the strength and direction of the relationship between two ranked variables. Unlike the Pearson correlation coefficient, which requires the data to be normally distributed and the relationship between variables to be linear, the Spearman correlation is non-parametric and does not depend on these assumptions. This makes it particularly useful for analysing relationships between variables when these conditions are not met.

In the Spearman correlation method, data values are first converted to ranks. When the data is sorted, each value is replaced by its rank, and in cases of tied values, the average rank is assigned. This ranking approach is crucial for the Spearman method as it compares the monotonic relationship between the variables based on these ranks. The calculation of the Spearman correlation coefficient (ρ) involves assessing the differences in the ranks of corresponding values in the two variables. The coefficient can range from -1 to +1. A coefficient of +1 signifies a perfect positive correlation, indicating that an increase in one variable exactly predicts an increase in the other. Conversely, a coefficient of -1 indicates a perfect negative correlation, meaning that an increase in one variable predicts a decrease in the other. A coefficient of 0 implies no correlation, indicating that the movement in one variable does not predict the movement in the other in any specific way.

The formula for calculating Spearman's correlation coefficient (ρ) is centred on the sum of the squared differences in ranks between the corresponding values of the two variables, adjusted for the number of observations.

Accompanying the Spearman correlation coefficient is a p-value, which is crucial for understanding the statistical significance of the correlation. This p-value represents the probability of observing the given data if the null hypothesis (which states that there is no association between the two variables) were true. A low p-value (typically less than 0.05) suggests that the correlation observed is unlikely to have occurred by random chance, thereby indicating a statistically significant correlation. In contrast, a high p-value suggests that the observed correlation could be due to random variation, and there is insufficient evidence to conclude a significant association between the variables.

In the analysis of the correlation coefficients with the target variable in Figure 3, several key variables emerge as particularly correlated and thus merit attention for potential bias. These variables include Debtor (correlation coefficient of -0.27), Gender (-0.25), Tuition Fees Up to Date (0.44), Previous qualification (-0.15) and Marital Status (-0.12). The selection of these variables is predicated not only on their statistical correlation with the target variable but also on their societally significant dimensions. For instance, age and gender are often regarded as critical attributes in bias detection analyses due to their potential role as protected characteristics, which may reveal underlying discrimination in the initial dataset.

The presence of a star symbol on the graph is a notable element, signifying those correlations that reach statistical significance, where the p-value is less than 0.05. This threshold is conventionally used in statistical analyses to denote a low likelihood (less than 5%) of the observed correlation being a product of random chance.

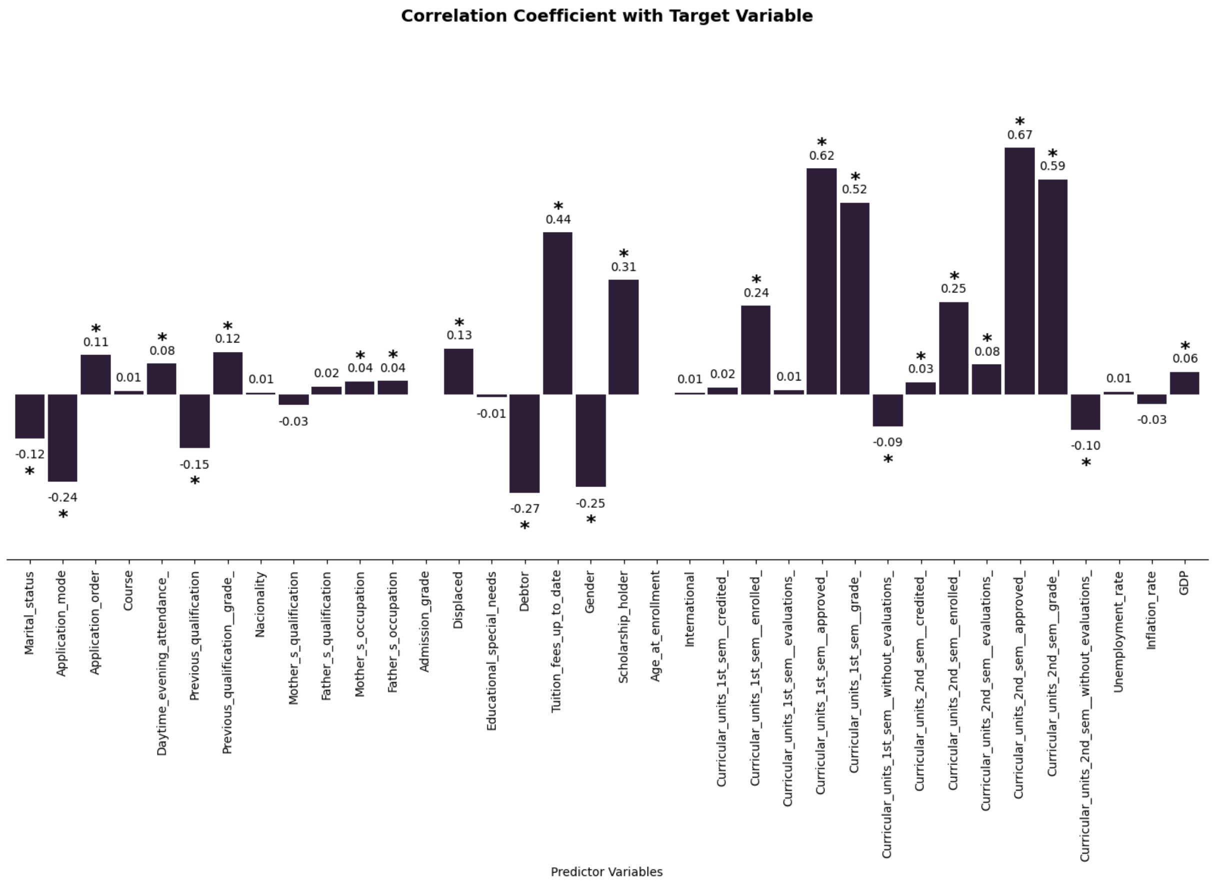


Figure 3: Correlation coefficient of each variable with the Target variable

Additionally, Kernel Density Estimation (KDE) is a non-parametric technique used to estimate the probability density function of a random variable. It provides a smoothed representation of the data's distribution, allowing for a visual assessment of the underlying density.

The Kernel Density Estimation (KDE) graph in Figure 4 provides an insightful visualisation of the distribution differences between a given variable and the target variable. The analysis highlights binary variables such as Debtor, Tuition Fees Up to

Date, and Gender and categorical variables such as Age at enrollment, Marital Status and Admission grade. This graphical representation indicates a higher likelihood of graduation among students who are not debtors to the university, identify as female, or have their tuition fees paid up to date. Additionally, it indicates a higher likelihood of graduation among students who are at an age under 21, are single and their admission grade is between 133 to 152.

Several factors might contribute to this observations, ranging from societal discrimination to potential sampling errors in the initial dataset. It is crucial to note that the primary objective of this analysis is to enable fair prediction across various groups, irrespective of the underlying causes of the observed bias.

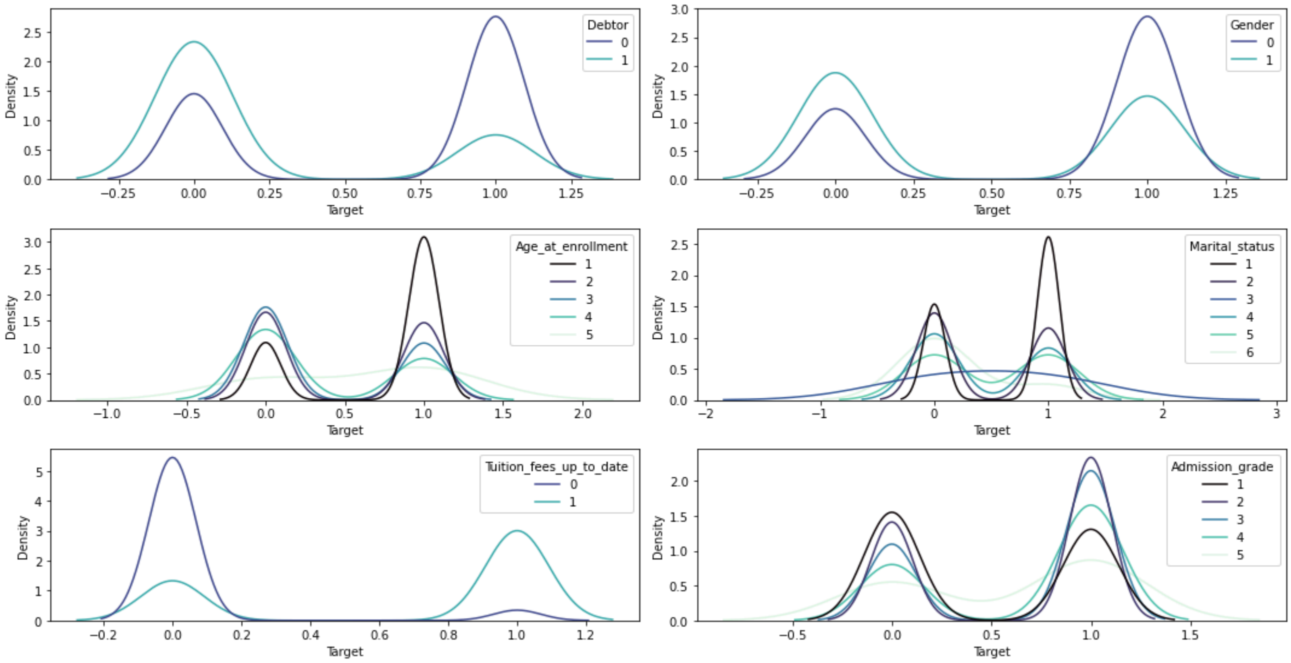


Figure 4: Kernel Density Estimation (KDE) plot of selected variables

4.2 Post-Modeling Bias Detection

Post-modeling bias detection is a critical phase in the model evaluation process, focusing on identifying and addressing biases that may have been introduced by the predictive model. This step is essential to ensure fairness and equity in the model's decisions, particularly when these decisions impact diverse groups of individuals. Several fairness metrics are commonly used in this context including Statistical Parity Difference, Disparate Impact Ratio, Equal Opportunity Difference and Average Odds Difference.

Statistical Parity Difference measures the difference in the probability of positive outcomes between the privileged and unprivileged groups. A value of zero indicates perfect parity, meaning both groups have equal chances of receiving positive outcomes. Values deviating from zero suggest potential bias, with positive or negative values indicating favouritism towards the privileged or unprivileged group, respectively.

Disparate Impact Ratio compares the proportion of positive outcomes between the unprivileged and privileged groups. A value of 1 implies no disparate impact, indicating that both groups are equally likely to receive positive outcomes. Values less than 1 indicate a bias against the unprivileged group and values more than 1 indicate bias against the privileged group.

Equal Opportunity Difference focuses specifically on the true positive rate, measuring the difference in this rate between the unprivileged and privileged groups. It aims to ensure that both groups have equal chances of being correctly identified for a positive outcome. A value close to zero suggests fairness in terms of equal opportunity.

Average Odds Difference averages the differences in the false positive rates and true positive rates between the unprivileged and privileged groups. It provides a more comprehensive view of bias by considering both types of errors (false positives and false negatives). Similar to other metrics, a value near zero indicates fairness.

By employing these metrics, post-modeling bias detection endeavours to quantify and rectify any disparities in model outcomes across different demographic groups, thereby upholding the principles of fairness and equity in predictive modelling. During the analysis, it is used XGBoost classifier .XGBoost, short for Extreme Gradient Boosting, is a popular and efficient implementation of gradient boosting machines, a type of machine learning algorithm. It is known for its performance and speed in classification tasks. XGBoost is particularly favoured for its ability to handle large datasets and its effectiveness in dealing with a wide range of predictive modelling problems.

The output metrics in Figure 5 suggest that the classification model is performing exceptionally well, almost to an ideal level in certain aspects. With True Negatives (TN) at 359 and True Positives (TP) at 628, the model shows a high capability in correctly classifying both negative and positive cases. The False Positives (FP) and False Negatives (FN) are relatively low at 73 and 29, respectively, indicating that the model makes mistakes but they are not excessive.

The 100% Balanced Accuracy is particularly notable. This implies that the model is equally proficient at identifying both classes, which is quite rare in real-world scenarios and often points to an extremely well-tuned model or, in some cases, could suggest an overfitting issue or an imbalance in the dataset.

The F1-score, at 92%, reflects a strong balance between precision and recall, meaning the model is reliable in its predictions and misses very few positive cases. This is further supported by the ROC-AUC score of 95%, indicating a superior ability of the model to distinguish between the positive and negative classes. Such a high ROC-AUC score usually signifies that the model has a very good rate of true positive predictions while maintaining a low rate of false positives.

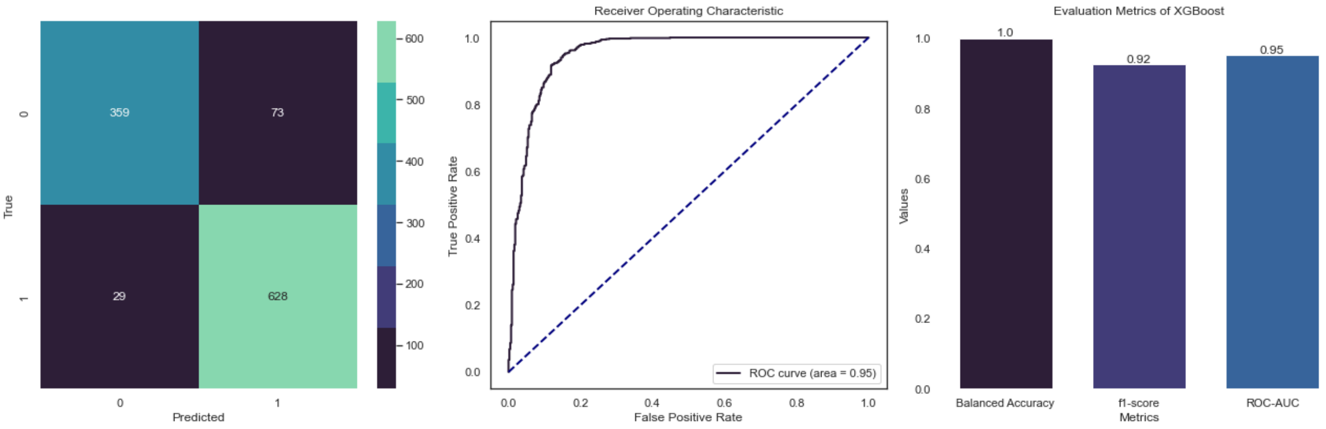


Figure 5: Confusion matrix, ROC-AUC curve and Evaluation Metrics results of XGBoost model

Table 2 provides details of the outcomes for each variable across several key fairness metrics, such as statistical parity difference, disparate impact ratio, equal opportunity difference, and average odds difference. Notably, the variables 'Previous_qualification', 'Debtor', and 'Tuition_fees_up_to_date' exhibit the most substantial biases according to these metrics.

For 'Previous_qualification', the statistical parity of 0.537 and disparate impact ratio of 5.8333 both suggest a significant bias, indicating that individuals in this group are more likely to receive positive outcomes compared to others. The equal opportunity difference of 0.4573 and average odds difference of 0.3145 reinforce this, showing a substantial disparity in favour of this group, especially in terms of true positive rates.

In the case of 'Debtor', the statistical parity of 0.4667 and disparate impact ratio of 2.9835 again indicate bias, with this group being more favourably treated. The equal opportunity difference of 0.2344 and average odds difference of 0.1513, though lower than in the 'previous_qualification' case, still suggest moderate bias towards debtors in terms of correctly identifying positive outcomes.

For 'Tuition_fees_up_to_date', the scenario is reversed. The negative statistical parity (-0.6869) and low disparate impact ratio (0.0657) point towards significant bias against this group. The negative equal opportunity difference (-0.4601) and average odds difference (-0.3334) further highlight this bias, indicating that individuals who are up to date with tuition fees are less likely to receive positive outcomes and face a higher disparity in false positive and true positive rates.

Overall, these results indicate varying levels of bias in model's predictions based on the attributes analysed. 'Previous_qualification' and 'Debtor' groups seem to receive more favourable outcomes, while 'tuition_fees_up_to_date' is disadvantaged.

5. Explainability Methods

Explainability methods in machine learning are essential tools that enable a deeper understanding of the decisions made by machine learning models. These techniques are especially important for complex models, often referred to as "black boxes", like deep learning networks or sophisticated ensemble methods.

The primary purpose of these methods is to render the outcomes of machine learning models transparent and comprehensible to humans. This clarity is vital for several key reasons. Firstly, it builds trust and confidence among users. When people understand how a model makes its decisions, they are more likely to trust and adopt these advanced technologies. Secondly, explainability is integral to ensuring ethical decision-making. It aids in identifying and addressing any biases present in the model, promoting fairness and the ethical application of artificial intelligence.

Moreover, explainability methods serve as a valuable tool for model improvement. By unraveling how a model arrives at its decisions, data scientists can pinpoint weaknesses or areas for refinement, which might include enhancing the model, choosing more effective features, or reevaluating the problem being addressed. In industries governed by strict regulations, like finance and healthcare, there's also a practical necessity for explainability. Regulations often mandate that decisions made by automated systems must be interpretable, ensuring accountability and transparency. Debugging and troubleshooting are additional areas where explainability proves invaluable. It can reveal whether a model's decisions are based on valid patterns in the data or on misleading correlations.

Among the various explainability methods, LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) stand out. LIME provides local explanations for individual predictions, shedding light on how a model arrived at a specific decision. SHAP, drawing from Shapley values in game theory, offers a comprehensive means to elucidate the output of any machine learning model, delivering insights into the importance of features both locally and globally.

In summary, explainability methods like LIME and SHAP are indispensable in the contemporary landscape of machine learning. They not only foster a deeper understanding and trust in AI systems but also ensure that these systems are used responsibly, ethically, and in compliance with regulatory standards. These methods bridge the gap between advanced machine learning techniques and human-centric decision-making, ensuring that AI advancements align with societal values and ethical principles.

5.1 LIME

LIME (Local Interpretable Model-agnostic Explanations) technique works by creating explanations for individual predictions, allowing users to understand how a model arrived at its decision for a specific instance.

The primary mechanism of LIME involves approximating the complex model locally. It starts by generating a new dataset composed of perturbed samples around the instance in question, and then it feeds these samples into the complex model to obtain predictions. The key here is that these perturbed samples are created to be close to the original instance, ensuring that the explanations are locally faithful.

Once the new dataset is prepared, LIME trains a simple, interpretable model, such as a linear regression or a decision tree, on this dataset. The simplicity of this model is crucial, as it needs to be easily understandable to humans. The model then provides insights into which features were most influential in the complex model’s prediction for the specific instance.

One of the key strengths of LIME is its model-agnostic nature, meaning it can be applied to any machine learning model. This universality makes it a highly versatile tool for model interpretation. However, it's important to note that LIME's explanations are local. They are valid for the particular instance being examined and do not necessarily represent the overall behaviour of the model.

While LIME has been immensely valuable in enhancing the transparency of AI systems, particularly in critical applications where understanding model decisions is essential, it is not without limitations. The technique's reliance on local approximations means that the explanations may not capture the model's behaviour in a broader context. Additionally, the approach used to create perturbations and the choice of the interpretable model can significantly influence the explanations, potentially leading to varying degrees of reliability.

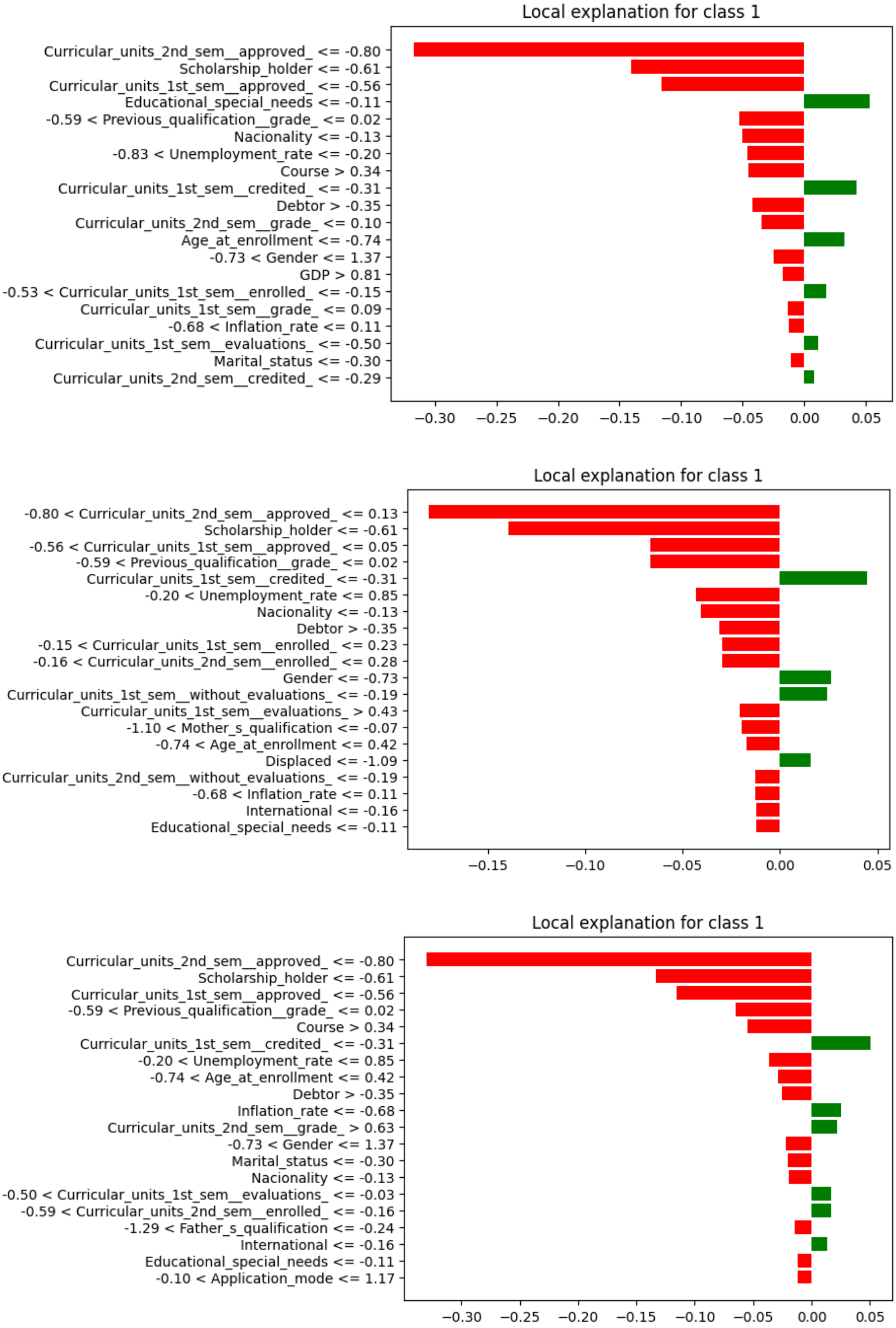
Below, the results are presented for the XGBoost algorithm, which were derived using the LIME methodology. The plots show the top contributing features to the model's prediction for the selected instance.Each bar represents a feature, and the length and direction of the bar indicate the feature's impact on the prediction. Positive contributions (pushing the prediction towards the positive class) are usually shown in one colour (e.g., green), and negative contributions (pushing towards the negative class) in another (e.g., red).The x-axis represents the weight of each feature, showing how much each feature pushes the model's output higher or lower.

For instance, in all three plots in Figure 6 we can see the attribute "Debtor" with a negative coefficient (red bars), suggesting that being a debtor is a negative contributor to the prediction of class 1. The magnitude of this feature's coefficient is fairly consistent across the three plots, indicating that its importance in the prediction of class 1 is stable across these instances.

The consistency in the direction and magnitude of the feature importance for the "Debtor" attribute implies that regardless of the other features and their values in different instances, having a debt to the university tends to uniformly decrease the likelihood of an instance being classified as class 1 by the model. This uniform behaviour suggests that the "Debtor" feature has a reliably negative influence on the model's decision for class 1, which can be an interesting insight for those analysing the model's behaviour.

Additionally, the negative coefficients of features like curricular units approved and having educational special needs suggest that these attributes may compound the challenges faced by debtors. Similarly, the scholarship holder attribute also negatively influences the prediction, which could reflect a nuanced relationship between financial aid and indebtedness on student outcomes. The unemployment rate's varied direction across the plots hints at complex external economic influences that might affect students, potentially making the situation worse for those with financial struggles.

In summary, the persistent negative influence of the "Debtor" attribute across multiple features suggests a significant weighting of financial status within the model's predictions, which may mirror larger systemic patterns affecting academic performance and student welfare. This uniformity in the data verifies that the "Debtor" status could be a potential axis of bias, systematically impacting certain student groups.


Local explanation for class 1

Local explanation for class 1

Local explanation for class 1

Figure 6: LIME explanation of students with debtor

5.2 SHAP

SHAP (SHapley Additive exPlanations) is an advanced method in machine learning designed to provide clear explanations for the predictions of any machine learning model. Rooted in cooperative game theory, specifically the Shapley values concept, SHAP delivers a unified measure of feature importance that is consistent and locally precise. This method is pivotal in interpreting complex models, offering insights into how individual features contribute to each prediction.

The fundamental process of SHAP involves assessing the impact of each feature in a model by computing its contribution to the prediction. This is done by evaluating every possible combination of features. The Shapley value, central to this approach, calculates the average impact of a feature across all these combinations. In other words, for a given prediction, a SHAP value tells you how much each feature in the dataset contributed to pushing the model's output from the baseline (average) prediction. If the SHAP value is high, it means that the feature significantly influenced the prediction. A positive SHAP value indicates that the feature pushed the model's prediction higher, while a negative value indicates it pushed the prediction lower.

SHAP stands out for its consistency — a key property ensuring that the importance attributed to a feature reflects its impact on the model's output. This aspect of SHAP sets it apart from other interpretability methods, which might not always provide consistent results. SHAP also offers both local and global interpretations. It not only shows the contribution of each feature to individual predictions (local interpretation) but can also aggregate these contributions across the dataset to offer a broader view of feature importance (global interpretation).

However, SHAP's comprehensiveness comes with a computational cost. The complexity of calculating contributions for each feature across all possible combinations can be substantial, especially in models with numerous features. This makes SHAP challenging to implement in scenarios requiring real-time analysis or with very large datasets.

Below, the results are presented for the XGBoost algorithm, which were derived using the SHAP methodology. This approach emphasises a detailed and analytical understanding of the algorithm's performance.

Firstly, the type of plot below in Figure 7 is a powerful tool for interpreting complex models and is particularly useful in understanding which features have the most influence on the model's predictions.

Each bar in the plot represents a feature from the dataset. The length of the bar corresponds to the average absolute SHAP value of that feature across all the data points in the dataset. The plot essentially ranks the features by their importance.

The bar length is indicative of the magnitude of impact the feature has on the model's output. A longer bar means that the feature significantly changes the model's prediction. Considering the absolute value ensures that the overall impact of the feature is reflected, irrespective of the direction of the impact.

This bar plot serves as a global explanation method. It provides a comprehensive view of feature importance across the entire dataset or the specified subset, making it an excellent tool for gaining a holistic understanding of the model. For instance, features with very small bars might have minimal impact and could potentially be removed to simplify the model. Conversely, features with long bars are crucial to the model's decision-making process and warrant closer attention.

For instance, the predictive model seems to prioritise academic progress as a key determinant of student success, with the number of approved curricular units in the second semester ('Curricular_units_2nd_sem_approved') being the most influential factor. This emphasis on second-semester achievements suggests that the model regards continued academic performance as a critical indicator of a student's potential. On the financial front, the positive impact of the 'Tuition_fees_up_to_date' feature underscores the model's consideration of a student's financial standing as integral to their academic journey, reflecting an implicit connection between financial stability and academic continuity.

Moreover, the model takes into account the student's initial academic engagement and performance, as indicated by the approval of first-semester units and enrollment in these units, highlighting the importance of a strong start to the academic year. The role of the specific academic program and the presence of financial aid, via the 'Course' and 'Scholarship_holder' attributes, further demonstrates the model's nuanced approach to evaluating student profiles. Additionally, the model factors in grades and the frequency of evaluations during the second semester, suggesting a comprehensive analysis of ongoing academic achievements.

Conversely, the 'Debtor' status, while having a smaller positive impact, still factors into the model, pointing to a subtle acknowledgment of the complexities surrounding a student's financial obligations. The aggregation of 27 other features, albeit individually less significant, hints at the model's multifaceted nature, considering a wide array of variables that collectively contribute to the depiction of a student's academic landscape.

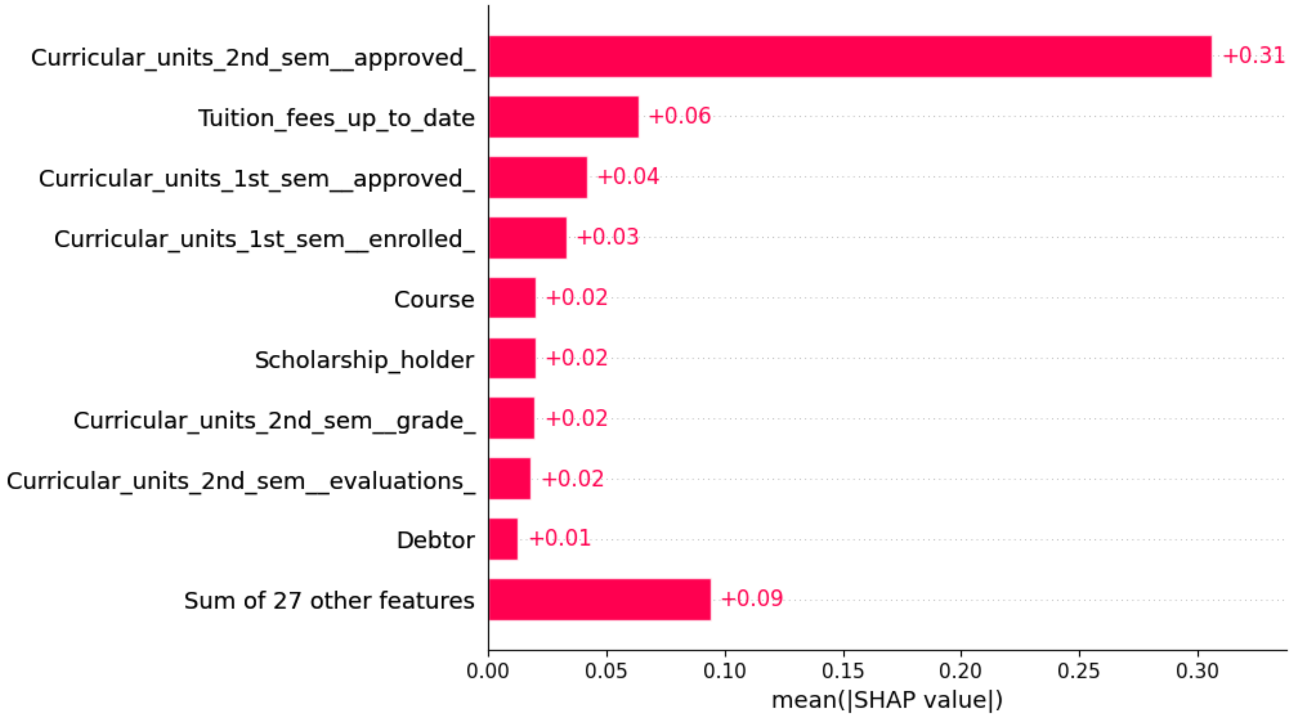


Figure 7: SHapley bar plot of the mean of SHAP values

On the other hand, the second type of plot below in Figure 8 offers a view of how different features influence the model's output, combining aspects of both global and local interpretability.

In a violin plot generated by SHAP, each feature in the dataset is represented by a row, with the features typically arranged vertically. The violin plot displays the full distribution of the SHAP values for each feature. This provides a deeper understanding of how each feature influences the model's predictions across different data points. The core element of this plot is the "violin" aspect, which is essentially a density plot that shows where the SHAP values for a particular feature are most concentrated. The thicker parts of the violin indicate a higher concentration of SHAP values, suggesting that the feature more frequently has a high impact on the model's output in that range. Conversely, thinner parts indicate that fewer data points have SHAP values in that range for the feature.

Furthermore, the colour coding within each violin plot typically indicates the value of the feature: higher values in one colour and lower values in another. This aspect helps in understanding not just the magnitude of the feature's impact but also the direction. For example, higher values of a feature might consistently lead to higher predictions from the model, which would be visible in the plot.

The SHAP violin plot is a powerful tool because it provides a comprehensive overview of how the features in a model contribute to its predictions. It goes beyond merely ranking features by importance (as in a bar plot) by illustrating the distribution and direction of their effects. This makes it highly useful for diagnosing model behaviour, understanding feature interactions, and communicating complex model dynamics in a more intuitive manner. It is particularly beneficial when it is important to understand not just which features are important, but how their values influence predictions in different ways.

At the top of the impact scale, we have features related to the approval of curricular units in both the first and second semesters (Curricular_units_1st_sem_approved and Curricular_units_2nd_sem_approved). The approval of second-semester curricular units has a notably positive impact, suggesting a strong correlation between academic progression and the model's predictions. This is closely followed by whether tuition fees are up-to-date (Tuition_fees_up_to_date), indicating the financial status of students as a significant predictor.

Enrollment in curricular units during the first semester (Curricular_units_1st_sem_enrolled) also plays a role, though its impact is more varied, hinting at a complex interaction with other factors. The specific course (Course) a student is enrolled in and whether they are a scholarship holder (Scholarship_holder) are also influential, but with a wider spread in SHAP values, reflecting a more nuanced effect on the model's output. Evaluations and grades for second-semester units (Curricular_units_2nd_sem_evaluations,Curricular_units_2nd_sem_grad) are critical too, with the plot showing that both high and low grades can significantly influence predictions, possibly affecting decisions related to the likelihood of student success.

The debt status of students ('Debtor'), generally tends to lower the model's output, which could be indicative of financial challenges affecting academic performance. Age at enrollment and the grades from previous qualifications (Age_at_enrollment, Previous_qualification_grade) introduce demographic and past performance elements into the predictive equation, each with their own spectrum of impact.

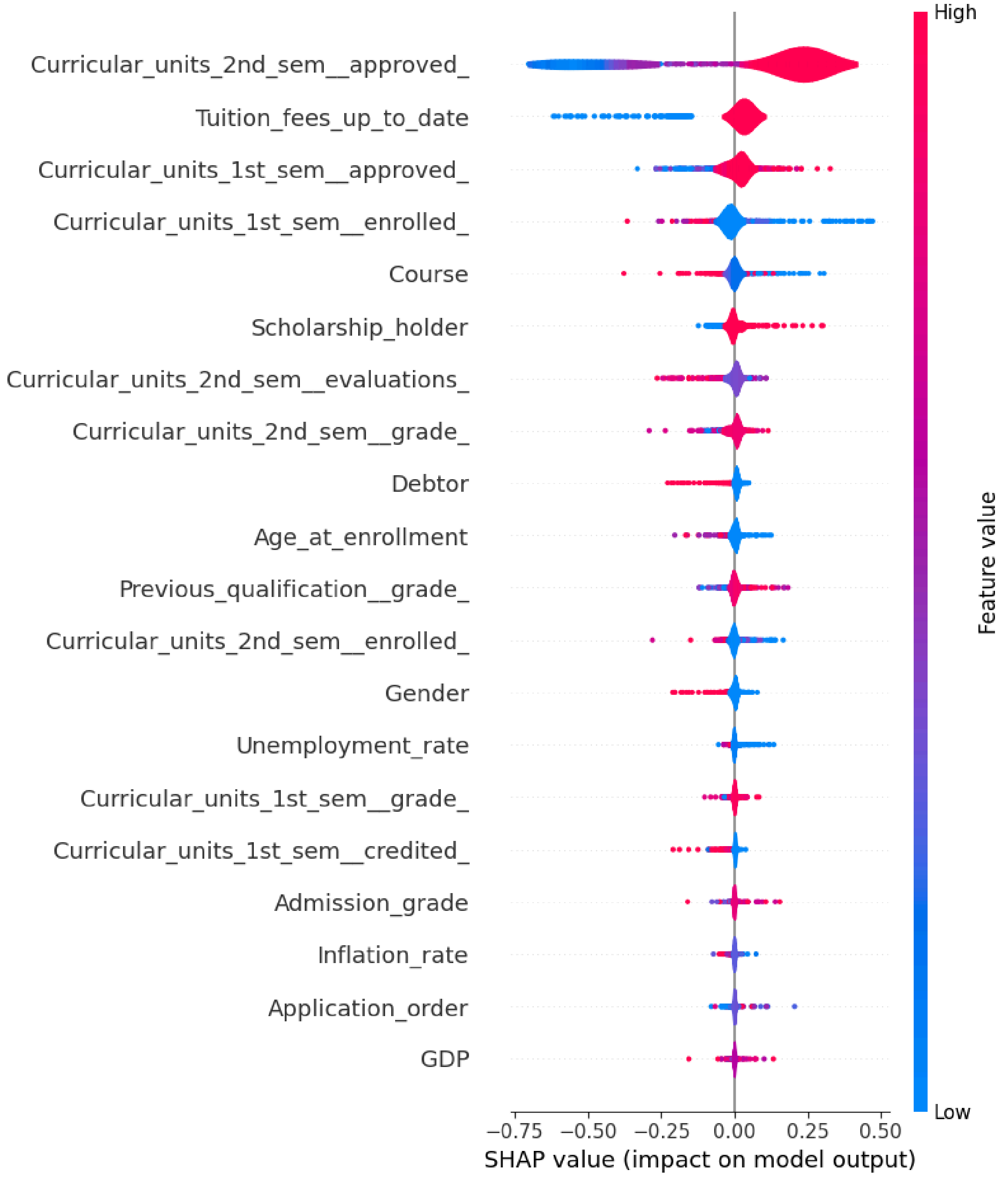


Figure 8: SHapley violin plot

Finally, in a SHapley waterfall plot, the focus is on how the input features of a single data point contribute to the model's output for that specific instance.

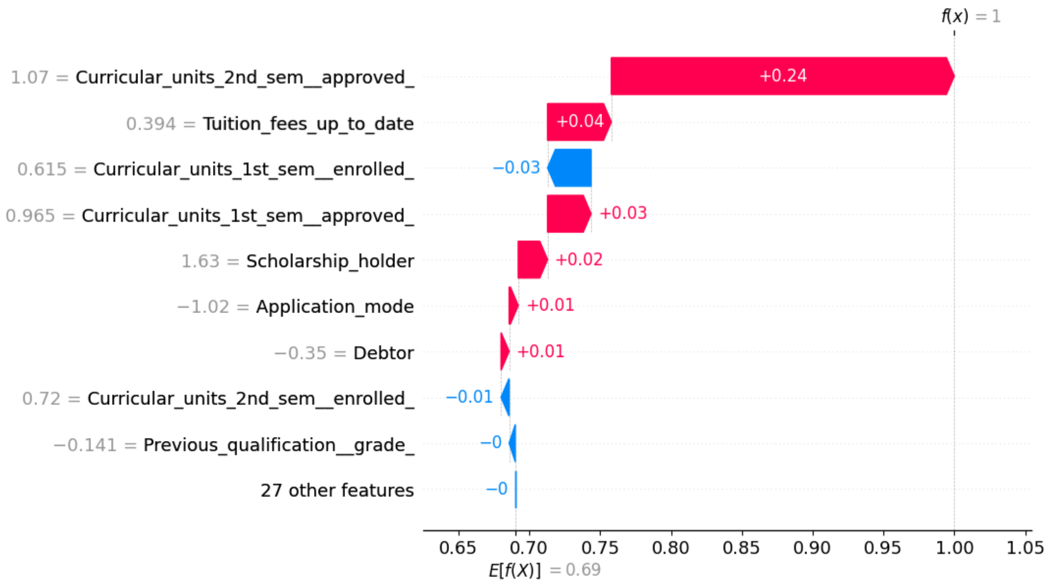
Each bar in the waterfall plot represents the contribution of a single feature to the shift from the baseline prediction to the actual model prediction for that specific data point. The length and direction of the bar indicate the magnitude and direction of the feature's impact. If a bar extends to the right, it means the feature increased the prediction value; if it extends to the left, the feature decreased the prediction value. The features are typically sorted by their impact magnitude, making it easy to see which features had the most significant positive or negative contributions to the prediction. The plot ends with the final prediction, showing how the cumulative effect of all the features transformed the baseline prediction into the model's final output for that particular instance.

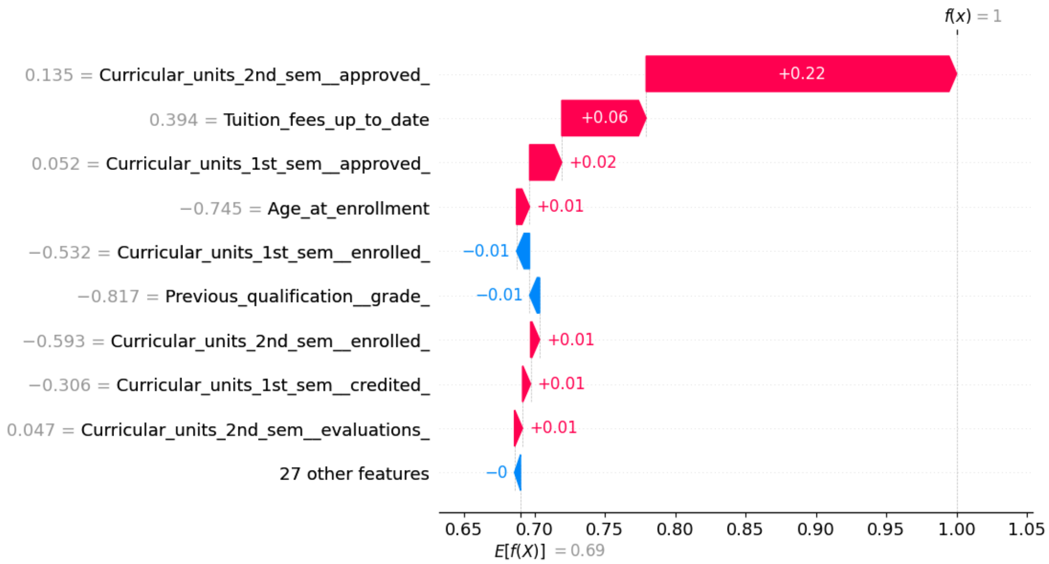
Waterfall plots are extremely useful for detailed, case-specific analysis. They provide a clear and intuitive way to understand how each feature of a specific data point contributes to the model's prediction. By analysing a waterfall plot, one can discern which features were most influential for a particular prediction and how they interacted to produce the final outcome.

Across all three plots in Figure 9, the 'Tuition_fees_up_to_date' feature consistently shows a positive impact on the model's predictions. This indicates that students who are current with their tuition payments are more likely to be predicted to graduate. The positive SHAP value for this feature suggests that financial regularity is an essential determinant of academic success according to the model, implying that students without financial delinquencies are seen as more likely to succeed.

Other common features across the plots include 'Curricular_units_1st_sem_approved', 'Curricular_units_2nd_sem_approved', and 'Curricular_units_1st_sem_enrolled', are possibly influenced by 'Tuition_fees_up_to_date'.They have positive SHAP values, although they vary in magnitude. These features relate to a student's academic engagement and achievements, reinforcing the model's prioritisation of academic performance as a significant predictor of graduation outcomes.

Given the strong influence of the 'Tuition_fees_up_to_date' feature, it is a prime candidate for fairness analysis. This is because it directly ties a student's financial status to their predicted academic success, which could introduce bias against students from less affluent backgrounds or those facing temporary financial hardships. If the model overly relies on this feature, it may unfairly disadvantage students who, for various reasons, are not able to keep their tuition fees up to date, despite their academic potential or effort.


f(x) = 1
1.07 = Curricular_units_2nd_sem__approved_
+0.24
0.394 = Tuition_fees_up_to_date
+0.04
0.615 = Curricular_units_1st_sem__enrolled_
−0.03
0.965 = Curricular_units_1st_sem__approved_
+0.03
1.63 = Scholarship_holder
+0.02
−1.02 = Application_mode
+0.01
−0.35 = Debtor
+0.01
0.72 = Curricular_units_2nd_sem__enrolled_
−0.01
−0.141 = Previous_qualification__grade_
−0
27 other features
−0
0.65
0.70
0.75
0.80
0.85
0.90
0.95
1.00
1.05
E[f(X)] = 0.69


f(x) = 1
0.135 = Curricular_units_2nd_sem__approved_
+0.22
0.394 = Tuition_fees_up_to_date
+0.06
0.052 = Curricular_units_1st_sem__approved_
+0.02
−0.745 = Age_at_enrollment
+0.01
−0.532 = Curricular_units_1st_sem__enrolled_
−0.01
−0.817 = Previous_qualification__grade_
−0.01
−0.593 = Curricular_units_2nd_sem__enrolled_
+0.01
−0.306 = Curricular_units_1st_sem__credited_
+0.01
0.047 = Curricular_units_2nd_sem__evaluations_
+0.01
27 other features
−0
0.65
0.70
0.75
0.80
0.85
0.90
0.95
1.00
1.05
E[f(X)] = 0.69

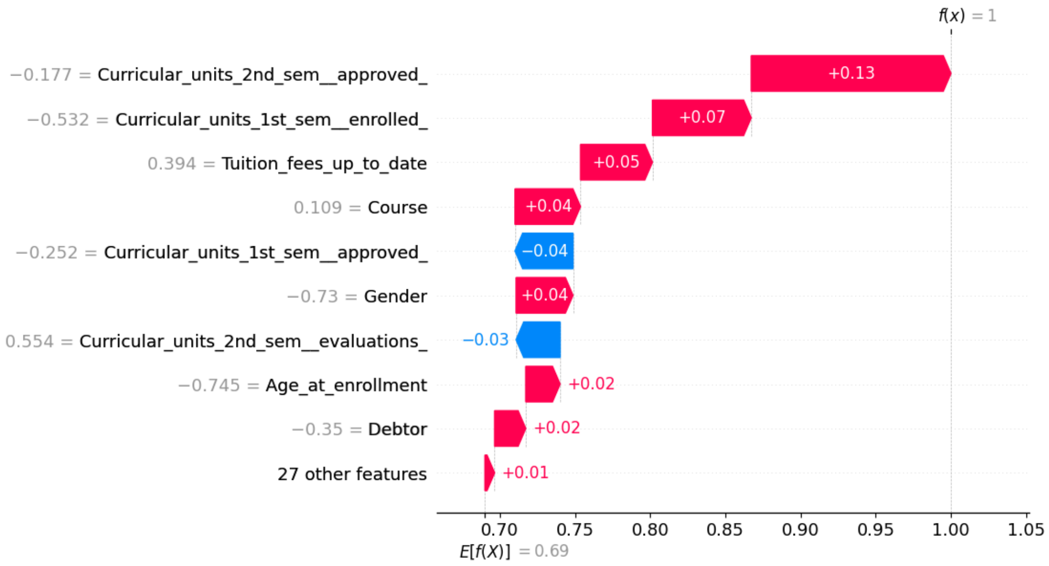


Figure 9: SHAP explanation of student with their tuition fees up to date

The provided plots above depict SHAP values for a predictive model focused on students whose tuition fees are not up to date in Figure 10 .'Tuition_fees_up_to_date' protective attribute shows a substantial negative SHAP value in each case. The same picture appears for the rest of the feature which again shows similar SHAP values. This indicates that not being current with tuition payments is a strong predictor for not graduating, according to the model. The magnitude of its impact is significant and suggests that financial delinquency is considered by the model as a key barrier to academic success.

As previously mentioned, other common features such as 'Curricular_units_2nd_sem_approved', 'Curricular_units_1st_sem_approved', and 'Curricular_units_2nd_sem_enrolled' also appear across the plots with negative SHAP values. This reinforces the idea that academic performance and engagement, as well as financial regularity, are tightly interwoven in the model's predictions.

The recurring prominence of 'Tuition_fees_up_to_date' with a negative impact verifies its importance for a fairness analysis. This feature's influence on the prediction of graduation outcomes signifies a potential area of bias, as it may disproportionately affect students from lower socioeconomic backgrounds or those experiencing temporary financial hardship.

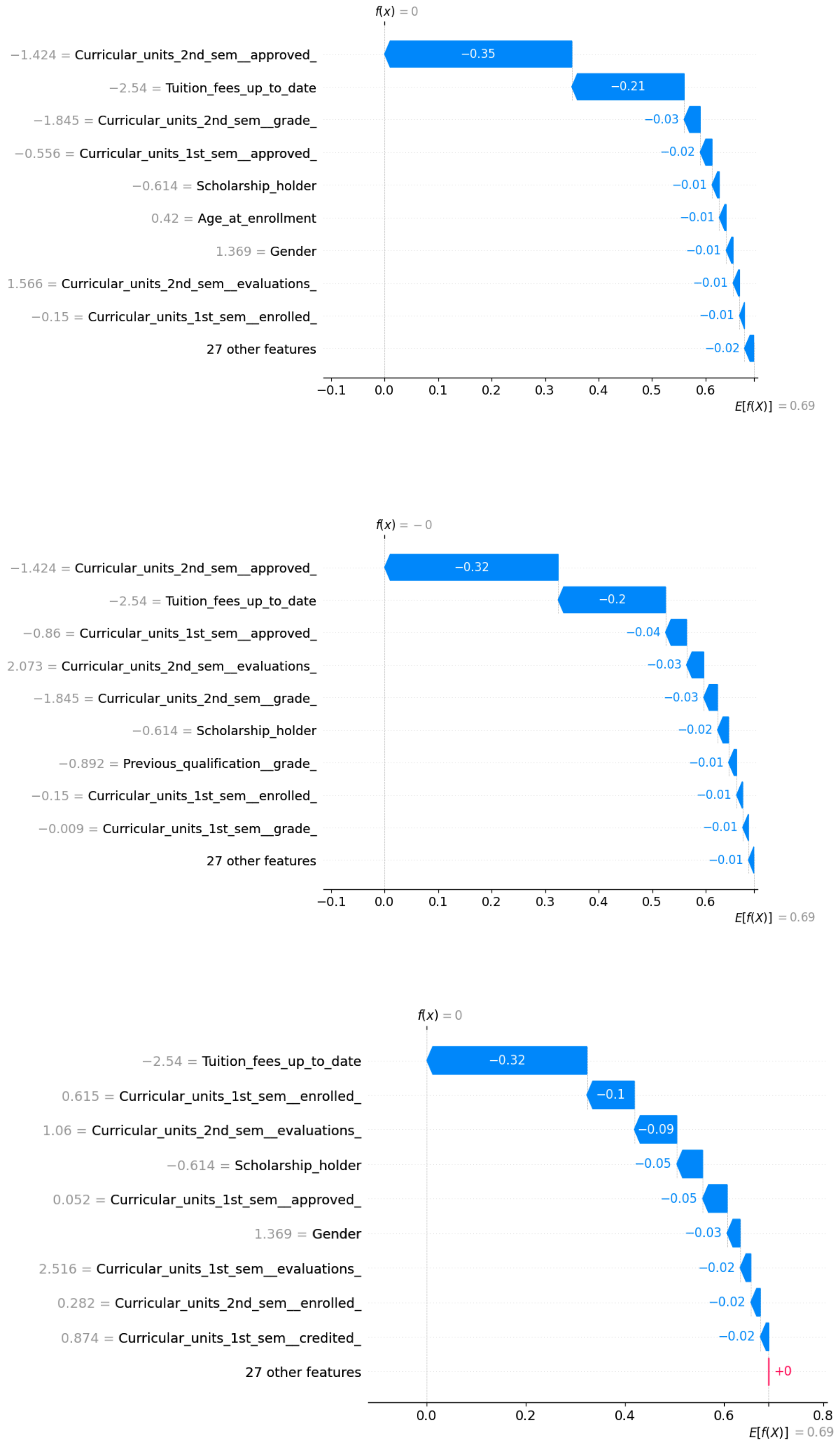


Figure 10: SHAP explanation of students with their tuition fees delayed.

6. Bias Mitigation Models

In the quickly growing fields of machine learning (ML) and artificial intelligence (AI), dealing with bias is a major challenge. As these technologies increasingly influence various aspects of society, from job recruitment to healthcare decision-making, the importance of addressing and mitigating bias in ML systems has become paramount. Biases in machine learning often arise due to skewed data, assumptions in how algorithms are made, or the socio-cultural context in which these models are deployed.

Bias mitigation models are specialised algorithms designed to identify and eliminate biases in ML systems. They aim to ensure that ML systems make decisions that are fair, equitable, and devoid of discriminatory undertones. The development and implementation of bias mitigation models involve a multifaceted approach, encompassing data preprocessing, in-processing techniques during model training, and post-processing adjustments after a model's predictions are made.

In preprocessing, the focus is on creating balanced datasets or transforming data in a way that neutralises biases. This could involve oversampling underrepresented groups or adjusting features that are disproportionately associated with certain outcomes. In-processing techniques incorporate fairness directly into the model training process, often by modifying the learning algorithms to penalise biassed predictions. Post-processing methods, on the other hand, adjust the model's output to achieve fairness objectives, typically by calibrating the results across different groups defined by sensitive attributes.

In this section, a clear and structured approach is used to explore bias mitigation models in machine learning. The discussion begins by explaining the chosen algorithm in detail, covering its theoretical basis and how it functions in reducing bias. This is followed by an examination of the algorithm's strengths and weaknesses, providing an objective perspective on its capabilities and areas for improvement. Then it is presented as a practical illustration of how the algorithm operates in a real-world context. After this, an evaluation is conducted using various metrics to assess both the performance of the model and its effectiveness in mitigating bias. The final part of this approach involves a comparison of the results obtained from the bias-mitigated model with those from the original algorithm before any bias mitigation was applied. This comparison helps to highlight the impact of the bias mitigation techniques and gives insight into the improvements achieved.

6.1 Fairness Pre-Processing

Pre-processing fairness techniques focus on preparing the data before it enters the machine learning pipeline. The first step in this approach is often identifying and understanding the potential sources of bias within the dataset. This could involve analysing historical trends, demographic imbalances, or societal biases that might be reflected in the data. Once these biases are identified, the next step is to modify the dataset to mitigate these biases. This can include techniques like balancing the dataset by either oversampling underrepresented groups or undersampling overrepresented ones, ensuring that the model is not skewed towards the majority group.

The second aspect of pre-processing fairness involves careful feature selection and transformation. This means either removing sensitive attributes that could directly lead to bias, such as race or gender, or transforming these attributes in a way that they do not disproportionately influence the model's outcome. It also involves engineering new features that could help in reducing bias. For example, instead of using direct demographic attributes, proxy variables that are less directly correlated with sensitive attributes can be created.

Finally, pre-processing may also involve the use of statistical techniques to identify and correct for biases. This could mean applying transformations to the data to reduce the correlation between sensitive attributes and the target variable, or employing statistical methods to assess and ensure that the data distribution is fair across different groups. These techniques aim to create a dataset that, when fed into a machine learning model, reduces the likelihood of perpetuating existing biases.

6.1.1 Learning Fair Representation

Learning Fair Representation is a pre-processing technique in machine learning that aims to mitigate unfair bias in the predictions made by an algorithm. It involves learning a representation of the data by minimising the amount of information regarding membership in a protected category that is present in the transformed representation while maximising all of the information which is present on the original data.

This learning algorithm aims for a middle ground between group fairness and individual fairness. Group fairness, also known as statistical parity, is meant that the proportion of members in a protected group receiving positive classification is identical to the proportion in the population as a whole. The goal of group fairness is to achieve similar predictions for different groups, regardless of their sensitive attributes. On the other hand, individual fairness means that similar individuals should be treated similarly. The goal of individual fairness is to provide similar outcomes or predictions for individuals who are similar in terms of their non-sensitive attributes, regardless of their group membership or sensitive attributes.

This method turns fairness pre-processing into an optimization problem where different terms in the optimization relate to group fairness and individual fairness. More specifically, this optimization problem preserves as much information about the individual's attributes as possible while simultaneously removing any information about membership with respect to the protected subgroup. In other words, this algorithm is designed to ensure that the proportion of members in a protected group receiving positive classification is identical to the proportion in the overall population (group fairness) and that similar individuals are treated similarly (individual fairness).

Regarding the mathematical formulation of this optimization problem, as X denotes the entire data set of individuals, Y is the binary random variable representing the classification decision for an individual and Z is a multinomial random variable where each of the K values represents one of the intermediate sets of "prototypes". Prototypes have the definition of points on the input space here. Our purpose it to minimizes the following objective: $L = A_z \cdot L_z + A_x \cdot L_x + A_y \cdot L_y$ where A_x, A_y, A_z are hyperparameters governing the trade-off between the system desiderata.

The first item in the objective is $L_z = \sum_{k=1}^{K} | M_k^+ - M_k^- |$

where $M_k^+ = P(Z = k|x^+ \epsilon X^+)\ and\ M_k^- = P(Z = k|x^- \epsilon X^-)\ \forall k \epsilon K.$ As X^+ denotes the subset of individuals that are members of the protected set while X^- denotes the subsets that are not members of the protected set. Our learning algorithm attempts to drive L_z to zero in order to ensure statistical parity which requires the probability that a random element from X^+ from a particular prototype is equal to the probability of a random element of X^- from the same prototype.

The second item in the objective is $L_x = \sum_{n=1}^{N} (x_n - \hat{x}_n)^2$ where $\hat{x}_n$ are the reconstruction of x_n from Z. The second item is a squared-error measure that constraints the mapping to Z to be a good description of X.

The final item in the objective is $L_y = \sum_{n=1}^{N} - y_n log\hat{y}_n - (1 - y_n)log(1 - \hat{y}_n).$

Here $\hat{y}_n$ is the prediction of y_n constrained by each prototype's prediction for Y, weighted by their respective probabilities. The values of the weights can be between 0 and 1. This item controls the predictions of y to be as accurate as possible.

As a result, the goal of learning fair representations is to optimize an equation with hyperparameters for the relative weightings for expressions corresponding to group fairness, individual fairness and accuracy.

Advantages of learned fair representations are numerous. First, by removing or minimising the impact of sensitive attributes, LFR techniques aim to retain the relevant information needed for the task at hand. This helps ensure that the predictions remain accurate and useful. Second, it generalize well to new, unseen data. This means that the fairness achieved during training can extend to future predictions, enhancing the system's long-term fairness. Third, it is flexible and can be adapted to different domains and contexts.

Considering the disadvantages, achieving perfect fairness often comes at the cost of decreased accuracy or performance in the task being performed. There is a trade-off between fairness and other metrics, such as predictive accuracy. Striking the right balance between fairness and accuracy can be challenging and it depends on the fairness decision of the operator. Moreover, learning fair representations can be computationally expensive, especially when using complex models or large datasets. The training process may require more time and computational resources compared to traditional models, making it less practical in certain scenarios. Finally, removing or reducing the influence of sensitive attributes in the representation may lead to some information loss. In some cases, the sensitive attributes might contain valuable information that is relevant to the decision-making process. Striking a balance between removing bias and preserving useful information is crucial.

Analysing the performance of Learning Fair Representations (LFR) and XGBoost algorithms through various evaluation and fairness metrics paints a detailed picture of their respective strengths and limitations.

For the evaluation metrics, XGBoost emerges as the superior model. It achieves a perfect Balanced Accuracy of 1, compared to LFR's 0.8476, indicating its impeccable ability in classifying each class accurately. In terms of the F1-score, XGBoost again leads with 0.9249 against LFR's 0.8972. Furthermore, the ROC-AUC score stands at 0.9518 for XGBoost, surpassing LFR's 0.8913. These scores collectively suggest that XGBoost is more adept in general predictive accuracy.

However, when it comes to fairness metrics, the narrative shifts. For the ‘Previous_qualification’, LFR records a Statistical Parity Difference of 0.5676, a Disparate Impact Ratio of 6.1083, an Equal Opportunity Difference of 0.4511, and an Average Odds Difference of 0.3550. In contrast, XGBoost shows slightly lower fairness with a Statistical Parity Difference of 0.537, a Disparate Impact Ratio of 5.8333, an Equal Opportunity Difference of 0.4573, and an Average Odds Difference of 0.3145. This trend continues in the ‘Debtor’, where LFR demonstrates lesser bias with a Statistical Parity Difference of 0.056, a Disparate Impact Ratio of 1.0896, an Equal Opportunity Difference of -0.0523, and an Average Odds Difference of -0.2151, as opposed to XGBoost's values of 0.4667, 2.9835, 0.2344, and 0.1513, respectively.

Finally, in the ‘Tuition_fees_up_to_date’, LFR again appears to be fairer with a Statistical Parity Difference of -0.507, a Disparate Impact Ratio of 0.3162, an Equal Opportunity Difference of 0.0507, and an Average Odds Difference of -0.0139, compared to XGBoost -0.6869, 0.0657, -0.4601, and -0.3334.

In summary, while XGBoost stands out in terms of overall predictive performance, LFR displays a consistent edge in fairness across different demographic groups.

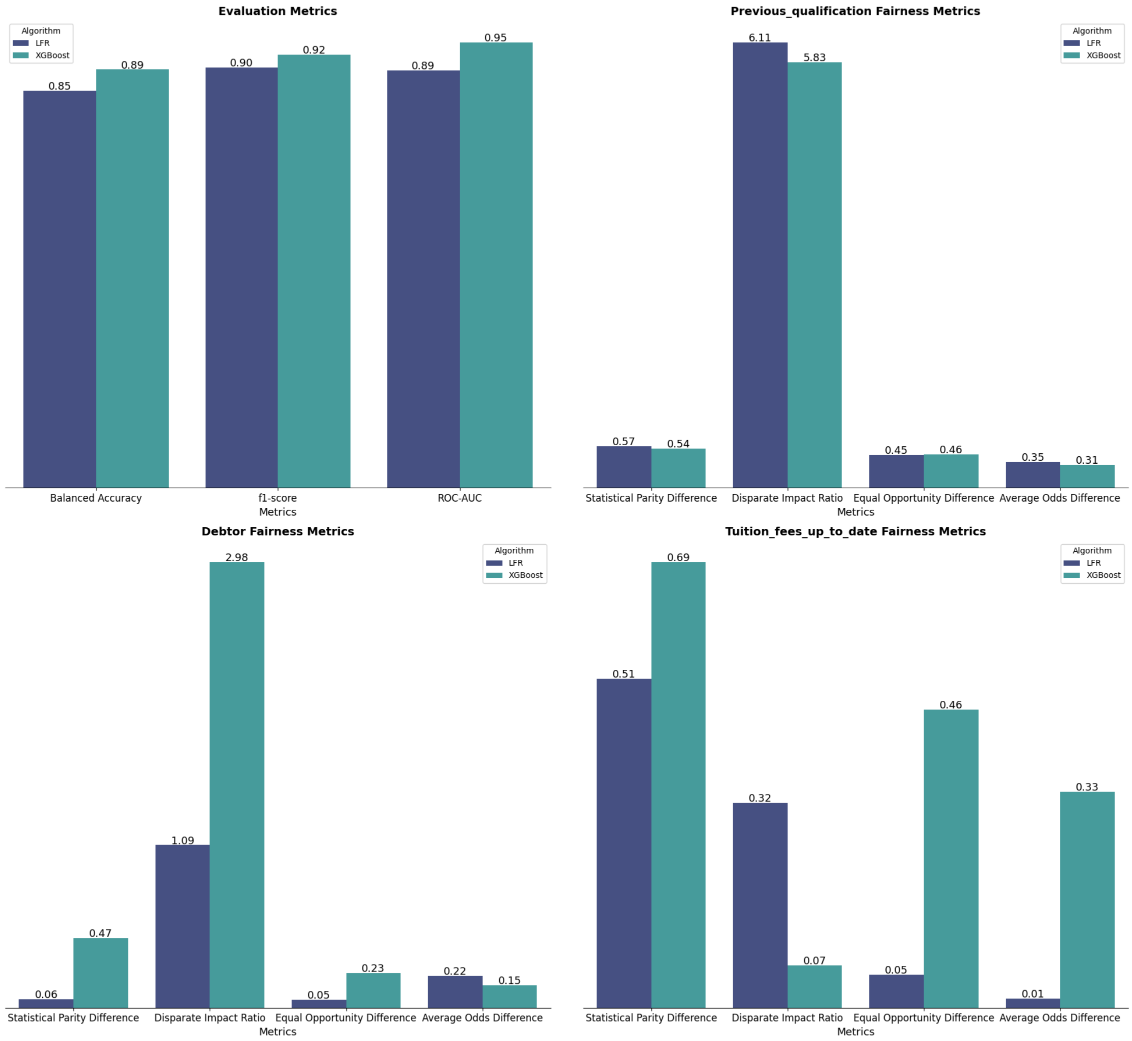


Figure 11: Evaluation and Fairness Metrics results of LFR algorithm

6.1.2 Reweighing

Reweighing refers to a technique that assigns different weights to samples in a dataset based on their attributes, particularly sensitive attributes. The goal of reweighing is to modify the sample weights in a way that reduces the influence of bias in the data during model training.

The process of reweighing involves adjusting the weights assigned to individual samples to account for the biases present in the dataset, such as sensitive attributes and their desired fairness objectives. The reweighting technique aims to give more emphasis to samples that are underrepresented, while reducing the impact of overrepresented or privileged samples.

To formalize the approach we first introduce some notation and assumptions. We assume a set of attributes $A = \{A_1, ..., A_n\}$ and their respective domains $dom(A_i), i = 1, ..., n$ have been given. A tuple X over the schema $(A_1, ..., A_n)$ is an element of $dom(A_1) \times ... \times dom(A_n)$. We denote the value of X for attribute A_i by $X(A_i)$. A labelled dataset D is a finite set of tuples over the schema $(A_1, ..., A_n, Class)$, with $dom(Class) = \{-, +\}$. We assume that a special attribute $S \in A$, called the sensitive attribute, and a special value $b \in dom(S)$, called the deprived community have been given. The semantics of the pair S, b is that it defines the discriminated community; for example, S could be “ethnicity” and b “Black”. Additionally, we transform the dataset with multiple attribute values for S into a binary one by replacing all values υ$\in dom(S) \setminus \{b\}$ with a new dedicated value w.

Considering the previous notations and assumptions, our goals is objects with $X(S) = b$ and $X(Class) =+$ will get higher weights than objects with $X(S) = b$ and $X(Class) =-$ and objects with $X(S) = w$ and $X(Class) =+$ will get lower weights than objects with $X(S) = w$ and $X(Class) =-$.

If the dataset D is unbiased then S and Class are statistically independent. As a result, the expected probability is :

$$P_{exp}(S = b \wedge Class =+) := \frac{|\{X \in D | X(S)=b\}|}{|D|} \times \frac{|\{X \in D | X(Class)=+\}|}{|D|} .$$

In reality, the observed probability is:

$$P_{obs}(S = b \wedge Class =+) := \frac{|\{X \in D | X(S)=b \wedge X(Class)=+\}|}{|D|} .$$

If the expected probability is higher than the observed probability value, it shows the bias towards class - for those objects X with X(S)=b.

Every object X will be assigned weights:

$$W(X) := \frac{P_{exp}(S=X(S) \wedge Class=X(Class))}{P_{obs}(S=X(S) \wedge Class=X(Class))}.$$

In this way, we assign a weight to every tuple according to its S and Class-values. We will call the dataset D with the added weights, D_W. On the new dataset we multiply the frequency of every object by its weight and as a result, we end-up with a discrimination-free balanced dataset.

This technique offers several advantages, primarily due to its nature as a pre-processing method. One of its key strengths is its model-agnostic approach, allowing it to be applied across various machine learning algorithms without requiring specific algorithmic adjustments. This method is particularly adept at handling class imbalances by altering the weights of instances in the training dataset, which ensures that underrepresented groups are given greater importance during model training. Such an approach is beneficial in reducing training bias, leading to fairer and more equitable predictions. Additionally, the simplicity and ease of implementation make reweighing an accessible option for many practitioners. Unlike some other techniques that might alter or remove data points, Reweighing maintains the integrity of the original data distribution, merely adjusting the significance of certain instances during the learning process.

However, the technique comes with its own set of limitations. Its effectiveness is heavily dependent on the quality of the initial data; in cases where the data is highly biassed or contains significant errors, Reweighing may not be adequate for comprehensive bias mitigation. It addresses biases primarily related to class imbalance and might not be effective against biases introduced during data collection or through feature selection. There is also a risk of overfitting to the minority class if weights are not correctly balanced, particularly in scenarios with extreme class imbalance. Moreover, while Reweighing can reduce bias, it may not eliminate it entirely, especially in cases of complex biases. Finally, there is often a trade-off between bias mitigation and predictive accuracy; reducing bias through Reweighing might lead to a decrease in overall model accuracy, particularly impacting the model's sensitivity to the majority class.

In comparing the Reweighing and XGBoost algorithms across various evaluation and fairness metrics, we observe distinct trade-offs between performance and fairness.

In terms of evaluation metrics, XGBoost demonstrates superior performance, achieving a perfect Balanced Accuracy score of 1.0 compared to Reweighing's 0.8785. This trend continues with the F1-score (XGBoost 0.9249 over Reweighing's 0.914) and ROC-AUC (XGBoost 0.9518 over Reweighing's 0.9388), where XGBoost marginally outperforms Reweighing, indicating a better balance between precision and recall, as well as a stronger capability in distinguishing between classes.

However, when assessing fairness metrics, the picture changes. In the context of 'Previous_qualification', XGBoost exhibits significantly higher disparities across all fairness metrics, including Statistical Parity Difference (XGBoost 0.537 over Reweighing's 0.4278), Disparate Impact Ratio (XGBoost 5.8333 over Reweighing's 2.925), Equal Opportunity Difference (XGBoost 0.4573 over Reweighing's 0.4481) and Average Odds Difference (XGBoost 0.3145 over Reweighing's 0.2479), compared to Reweighing. This pattern is consistent in the 'Debtor' category, where XGBoost again shows higher bias across all fairness metrics, including Statistical Parity Difference (XGBoost 0.4667 over Reweighing's 0.3438), Disparate Impact Ratio (XGBoost 2.9835 over Reweighing's 1.9949), Equal Opportunity Difference (XGBoost 0.2344 over Reweighing's 0.1448) and Average Odds Difference (XGBoost 0.1513 over Reweighing's 0.0412).

The most stark contrast is observed in the 'Tuition_fees_up_to_date' category. Here, both algorithms demonstrate biases, but XGBoost's bias is markedly more pronounced, as evidenced by its lower values in Statistical Parity Difference (XGBoost -0.6869 over Reweighing's -0.5389) and Disparate Impact Ratio (XGBoost 0.0657 over Reweighing's 0.2497) , and higher negative values in Equal Opportunity Difference (XGBoost -0.4601 over Reweighing's -0.1144) and Average Odds Difference (XGBoost -0.3334 over Reweighing's -0.0858).

In summary, while XGBoost might be the preferred choice for pure performance, Reweighing aligns better with the goal of reducing bias in predictions. This demonstrates a strategic decision-making point in model selection: opting for slightly reduced accuracy with Reweighing in exchange for more equitable and fair outcomes.

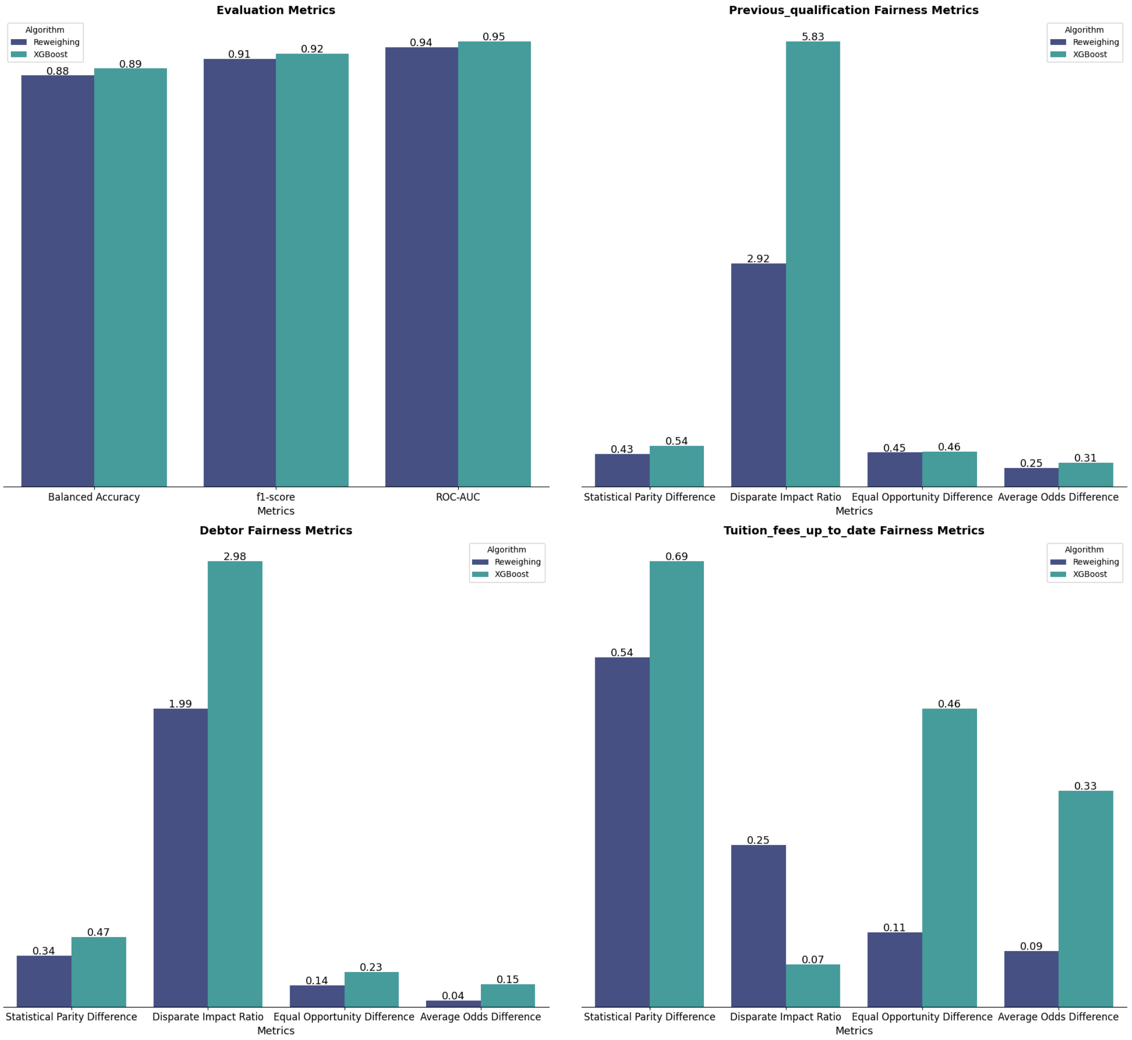


Figure 12: Evaluation and Fairness Metrics results of Reweighing algorithm

6.2 Fairness In-Processing

In-processing fairness techniques involve integrating fairness directly into the model training process. The first approach in this method is to modify the learning algorithms to be sensitive to fairness considerations. This could involve tweaking the model's objective function to not only optimize for accuracy but also for fairness metrics, ensuring that the model does not favor one group over another.

The second approach in in-processing fairness is to incorporate constraints into the model that directly address fairness. These constraints can enforce equal treatment across different groups defined by sensitive attributes. This might mean ensuring that the model has similar false positive rates across different races in a criminal justice application, or similar loan approval rates across genders in a financial application.

The third aspect of in-processing fairness is the use of ensemble methods. These methods involve training multiple models, each focusing on different aspects of the data, and then combining their outputs. This approach can ensure that various perspectives are considered, and no single group's characteristics dominate the final model's decision-making process. Ensemble methods can be particularly effective in complex datasets where a single model might struggle to balance accuracy and fairness.

6.2.1 Adversarial debiasing

The Adversarial Debiasing algorithm is a novel approach designed to mitigate unwanted biases in machine learning models. The core objective of this algorithm is to create a model that accurately predicts an outcome without being influenced by protected attributes. This is achieved through a unique method known as adversarial training, which involves the simultaneous training of two distinct models with competing objectives.

The first model, known as the Predictor, is responsible for predicting the desired output based on the input data. The primary focus during the training of the Predictor is to enhance its accuracy in making these predictions. However, this is where the second model, the Adversary, comes into play. The Adversary's role is to predict the protected attribute (like gender or race) from the outputs of the Predictor. As the Adversary trains to better predict this protected attribute, it forces the Predictor to adjust its outputs to minimise the information about the protected attribute, thereby reducing bias.

The adversarial model is motivated by a fairness intuition, which is that the outputs from the model should not include or leak information about the sensitive attribute. Ideally, in a situation without bias, this adversarial model should not be able to predict well the sensitive attribute.

The process is iterative:

a. Train the target model to predict the variable of interest.
b. Train the adversary model to use outputs from the target model to predict the protected attributes.
c. Iterate.

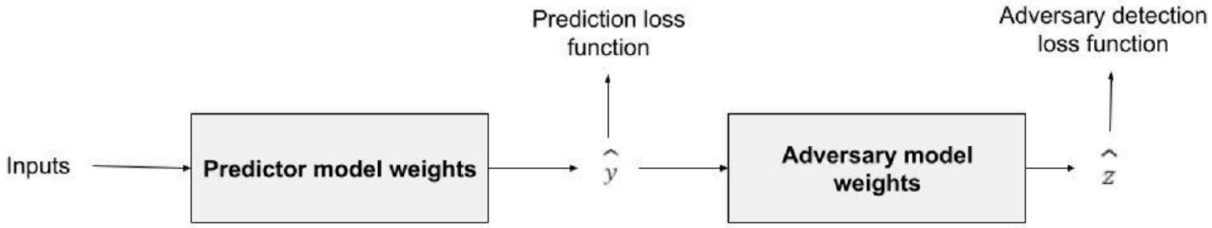


Figure 13: Adversarial Debiasing iteration process representation

In other words, a classifier network is trained to predict the target variable (Y) using the input features (X) while ignoring the sensitive attribute (Z). This step focuses on maximising the predictive accuracy of the classifier without considering fairness or biases. Then, an adversary network is trained to predict the sensitive attribute (Z). Its objective is to minimise the adversary's ability to predict the sensitive attribute (Z).

The classifier and adversary networks are trained in an iterative process. The models are updated to minimise their respective objectives. At each iteration, the gradients from the adversary network are back propagated through the classifier, encouraging the classifier to generate features that are less informative about the sensitive attribute. The classifier, in turn, updates its weights to make accurate predictions while being invariant to the sensitive attribute.

The joint optimization process continues until convergence, where the classifier becomes more robust to the influence of the sensitive attribute, and the adversary becomes less accurate in predicting it. The result is a classifier that makes predictions based on the features while being less biassed by the sensitive attribute.

One of the key strengths of Adversarial Debiasing is its ability to effectively mitigate bias. By employing an adversarial network specifically focused on predicting protected attributes, it ensures that these attributes do not influence the main predictive outcomes. This leads to more equitable and fair results. The algorithm's flexibility in enforcing various definitions of fairness, including demographic parity and equality of odds, allows it to be adapted for diverse scenarios and requirements. Furthermore, its versatility in application across different data types and predictive tasks, including both classification and regression problems, marks it as a valuable tool against bias in various domains.

However, the approach is not without its challenges. Training adversarial networks is known for its complexity, often requiring meticulous tuning of parameters and understanding of training dynamics to achieve stability and effectiveness. A significant limitation of this technique is the potential trade-off between accuracy and fairness. Striving for unbiased predictions with respect to certain attributes can sometimes lead to a reduction in the overall accuracy of the model.

Additionally, the effectiveness of Adversarial Debiasing largely depends on the representation of different groups in the training data. If certain groups are underrepresented, the algorithm may not effectively reduce biases against those groups. There's also a risk of reverse discrimination, where the model, in its attempt to debias, could become biassed against previously privileged groups.

Regarding the comparison between Adversarial Debiasing and XGBoost algorithms presents a nuanced picture when considering both evaluation metrics and fairness metrics.

In evaluation metrics, XGBoost demonstrates superior performance. It achieves a perfect score of 1 in Balanced Accuracy, compared to Adversarial Debiasing's 0.8711. This indicates that XGBoost is more effective in balancing accuracy between different classes. In the F1-Score, XGBoost again leads with a score of 0.9249 against Adversarial Debiasing's 0.8982, suggesting better precision and recall. Similarly, for the ROC-AUC metric, XGBoost's score of 0.9518 surpasses Adversarial Debiasing's 0.9354, indicating a better ability to distinguish between classes.

The scenario shifts when assessing fairness metrics. In ‘Previous_qualification’, Adversarial Debiasing demonstrates a trend towards reduced bias compared to XGBoost. Looking at the Statistical Parity Difference, Adversarial Debiasing scores 0.4981, which is slightly lower than XGBoost's 0.537. For the Disparate Impact Ratio, Adversarial Debiasing's score of 5.4833 is closer to the ideal value of 1 compared to XGBoost's higher 5.8333, indicating a reduction in disparity of impact across different groups. Regarding the Equal Opportunity Difference, Adversarial Debiasing scores

0.4008, which is notably lower than XGBoost's 0.4573. Lastly, in terms of the Average Odds Difference, Adversarial Debiasing's score of 0.2804 is lower than XGBoost's 0.3145.

In the 'Debtor' Adversarial Debiasing exhibits significantly lower bias. It scores 0.3302 in Statistical Parity Difference, much lower than XGBoost's 0.4667, implying a more equitable distribution of opportunities irrespective of Previous_qualification. For Disparate Impact Ratio, Adversarial Debiasing's score of 2.0444 is closer to 1 compared to XGBoost's 2.9835. In Equal Opportunity Difference, Adversarial Debiasing scores 0.0556, considerably lower than XGBoost's 0.2344, showing less bias in providing positive outcomes across different groups. Additionally, Adversarial Debiasing's Average Odds Difference score of 0.0053 is much lower than XGBoost's 0.1513, again indicating a more balanced rate of positive outcomes.

When considering 'Tuition_fees_up_to_date', Adversarial Debiasing maintains its advantage in reducing bias. It scores -0.531 in Statistical Parity Difference compared to XGBoost's -0.6869, both indicating some bias but with Adversarial Debiasing being less so. In Disparate Impact Ratio, Adversarial Debiasing scores 0.2143 against XGBoost's 0.0657; although both scores deviate significantly from 1, Adversarial Debiasing is relatively closer. For Equal Opportunity Difference, Adversarial Debiasing's score of -0.0668 is less biased compared to XGBoost's -0.4601, indicating fairer treatment of different debtor groups. Finally, in Average Odds Difference, Adversarial Debiasing scores -0.0646, which is significant less biased than XGBoost's -0.3334, suggesting a more balanced approach in delivering favourable outcomes.

In summary, while XGBoost shows a clear advantage in standard evaluation metrics, Adversarial Debiasing consistently shows less bias across various fairness metrics , particularly in contexts sensitive to fairness like 'Debtor' or 'Tuition_fees_up_to_date'.

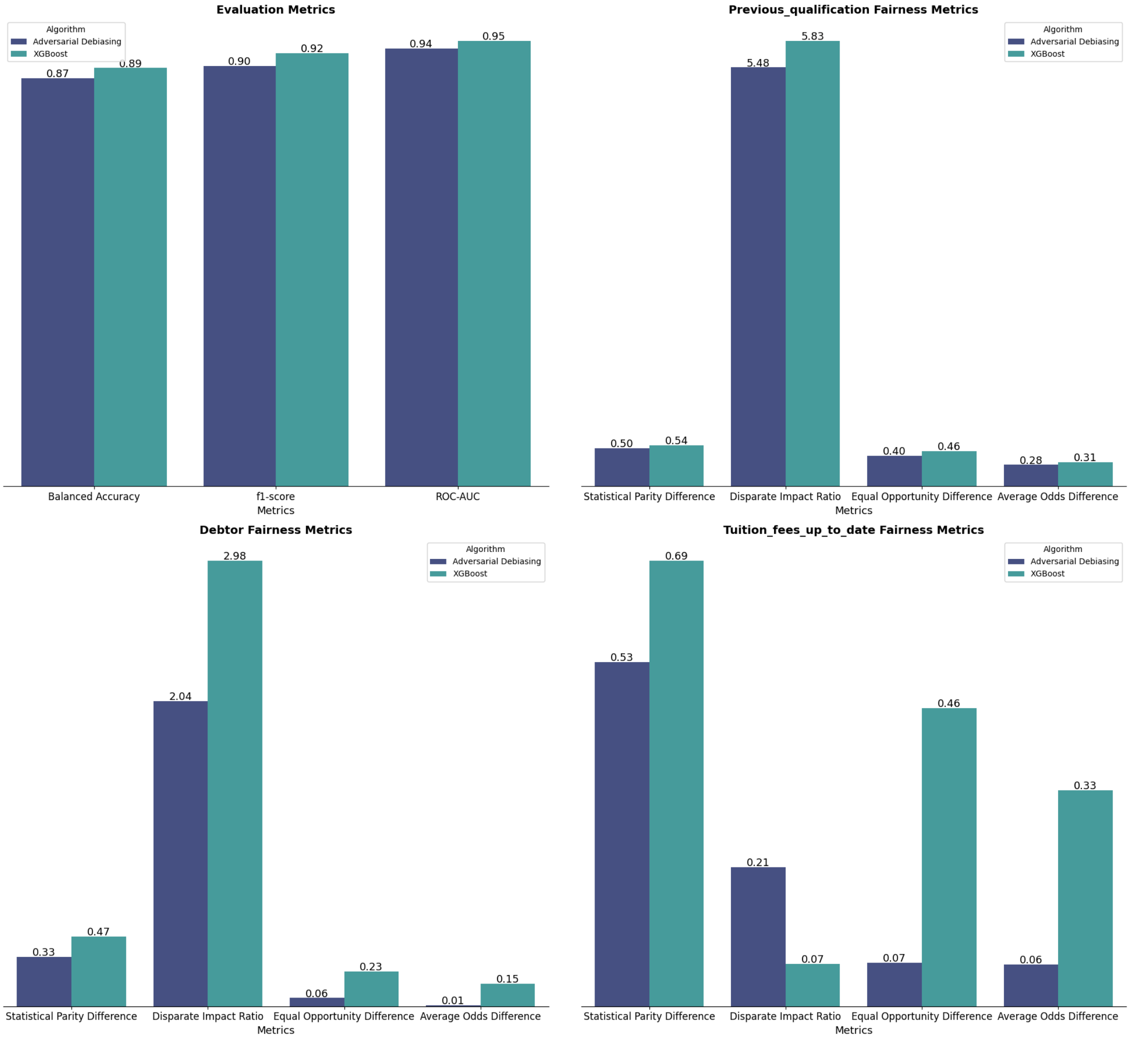


Figure 14: Evaluation and Fairness Metrics results of Adversarial Debiasing algorithm

6.2.2 Exponentiated Gradient Reduction

The study introduces a methodical way to achieve fairness in situations where we categorise things into two groups, known as binary classification. The researchers aim to solve fairness issues by suggesting a method that covers a wide range of fairness ideas, like demographic parity and equalized odds, which can be defined using straight-line formulas based on certain statistical conditions.

The preprocessing methods being used today are made for specific fairness rules and usually try to change the dataset so it works well with all types of learning models.

However, this often results in classifiers that still have biases. Meanwhile, post-processing methods, used after the model is trained, allow for a wider understanding of fairness and can show they are fair. But, these methods might not always find the most accurate fair classifier and they often need sensitive information during tests, which might not be always available.

The main part of the study is creating a new 'reductions approach'. This method looks at the basic classification method as something unknown, or a 'black box', and can handle many fairness standards. It makes sure to find the most accurate fair classifier without needing sensitive information during tests. The study shows how under these conditions, binary classification can be broken down into a series of smaller, cost-focused classification tasks. This method just needs the basic ability to use a cost-focused classification algorithm, which doesn’t have to know the specific fairness rule or sensitive information. It shows that solving these smaller tasks leads to a random classifier that is fine-tuned for the least amount of errors while still meeting the chosen fairness standards.

The method examines a binary classification environment where the training samples consist of triples (X, A, Y), with X being a feature vector, A a protected attribute, and Y a label. The feature vector X may include the protected attribute A or other features that might indirectly suggest A.The aim is to create an accurate classifier from a set of potential classifiers while meeting a certain fairness criterion. It’s important to note that the classifiers do not explicitly rely on A.

The first definition—demographic parity— can be achieved if its predictions are statistically independent of the protected attribute A.

$$P[h(X) = \widehat{y} \mid A = a] = P[h(X) = \widehat{y}] \; for\; all\; a, \widehat{y}.$$

The second definition—equalised odds—addresses the shortcomings of demographic parity by ensuring that the classifier’s predictions are conditionally independent of the protected attribute A, given the actual label Y.

$$P[h(X) = \widehat{y} \mid A = a,\; Y = y] = P[h(X) = \widehat{y} \mid Y = y] \; for\; all\; a, y\; and\; \widehat{y}.$$

Both definitions can be incorporated into a general framework of linear constraints :

$M\mu(h) \leq c$, where matrix $M \epsilon \mathfrak{R}^{|K| \times |J|}$ and vector $c \epsilon \mathfrak{R}^{|K|}$ described the linear constraints.

In typical binary classification, the goal is to identify the classifier with the least classification error, $err(h) := P[h(X) \neq Y]$. However, the objective here is to find the most accurate classifier that also meets fairness constraints. This involves solving a constrained optimization problem to minimise the error while satisfying a set of linear constraints.

$min_{h \epsilon H} err(h)$ subject to $M\mu(h) \leq c$.

The method extends the search beyond fixed classifiers to randomised classifiers, which offer better accuracy-fairness compromises. A randomised classifier makes predictions by first choosing a classifier from a distribution over the classifier set and then using it to make the prediction. Thus, it aims to solve a minimization problem that finds the optimal distribution over classifiers that minimises error while adhering to the fairness constraints.

The algorithm allows for achieving a desired accuracy-fairness tradeoff and supports a wider range of fairness definitions. This flexibility makes it suitable for various applications and scenarios where fairness needs may differ. It performs comparably or better than other approaches, especially in settings where the protected attribute is binary. The algorithm solves for the optimal points on the Pareto frontier for all classifiers in each considered class, indicating its effectiveness in balancing fairness and accuracy. Unlike some post-processing algorithms, the Exponentiated Gradient Reduction does not require access to protected attributes during testing, which is a significant advantage in scenarios where such data may not be available or its use is restricted.

However, the algorithm relies on empirical rather than true quantities, introducing an unavoidable source of statistical error. This reliance might affect the accuracy of the fairness adjustments, especially in cases where the empirical data significantly deviates from the true underlying distribution. The method involves certain constraints in its optimization algorithm, such as a bound on the magnitude of the Lagrange multipliers (λ) and a fixed number of iterations for the optimization process. These constraints can introduce errors, although they can be reduced with additional iterations, potentially at the cost of increased computational resources.

The comparison between the Exponentiated Gradient Reduction (EGR) algorithm and the XGBoost algorithm across various metrics offers insightful contrasts, particularly in balancing performance with fairness, a key goal in algorithm selection.

In terms of evaluation metrics, the EGR algorithm shows a Balanced Accuracy of 0.8858, which, while commendable, falls short of XGBoost's perfect score of 1.0. This trend continues with the F1-score, where EGR's 0.9169 is slightly lower than XGBoost's 0.9249. The ROC-AUC score is 0.8858 for EGR, again below XGBoost's superior 0.9518.

When assessing fairness metrics for 'Previous_qualification', the EGR algorithm demonstrates a Statistical Parity Difference of 0.4157 compared to XGBoost's higher 0.537, indicating less bias in the EGR algorithm. The Disparate Impact Ratio is 2.8708 for EGR, substantially lower than XGBoost's 5.8333, suggesting a more balanced outcome distribution. The Equal Opportunity Difference is 0.442 for EGR and 0.4573 for XGBoost. Similarly, the Average Odds Difference is lower for EGR (0.2343) compared to XGBoost (0.3145).

In the 'Debtor' category, EGR's Statistical Parity Difference of 0.3134 is notably lower than XGBoost's 0.4667, showing less disparity in positive outcomes. The Disparate Impact Ratio is 1.8698 for EGR versus 2.9835 for XGBoost, highlighting less bias in EGR. Interestingly, the Equal Opportunity Difference for EGR is -0.0218, compared to 0.2344 for XGBoost, suggesting EGR's more equitable treatment of positive cases. The Average Odds Difference also reflects this trend, with EGR at -0.0439 and XGBoost at 0.1513.

For 'Tuition_fees_up_to_date', EGR's Statistical Parity Difference of -0.5172 is less negative than XGBoost's -0.6869, indicating less bias against specific groups. The Disparate Impact Ratio for EGR is 0.2647, significantly less biassed than XGBoost's 0.0657. The Equal Opportunity Difference for EGR is -0.0218, less biassed than XGBoost’s -0.4601. Similarly, Average Odds Difference for EGR is 0.0168 and -0.3334 for XGBoost, indicating less bias in EGR.

In summary, while XGBoost excels in performance metrics, its fairness metrics suggest a higher bias across various attributes compared to the EGR algorithm. The EGR algorithm, despite slightly lower accuracy and predictive power, achieves more equitable outcomes, aligning better with the goal of minimising bias.

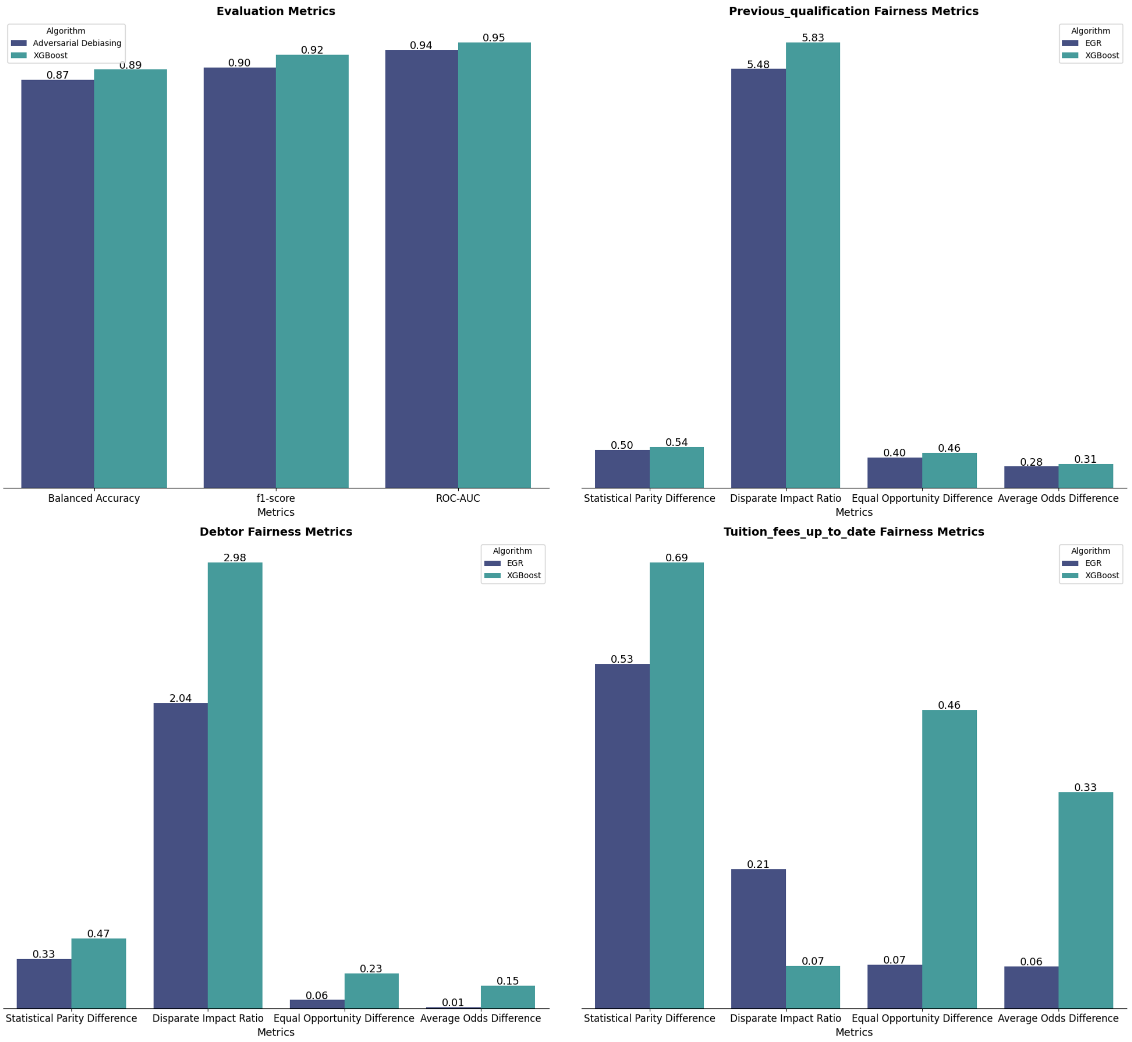


Figure 15: Evaluation and Fairness Metrics results of Exponentiated Gradient Reduction algorithm

6.3 Fairness Post-processing

Post-processing fairness techniques in machine learning come into play after a model has been trained and has made its predictions. This approach is particularly relevant in scenarios where altering the training process or the dataset itself is either not possible or insufficient to address biases. Post-processing involves a thorough examination of the model's outputs to identify and correct biases that might lead to unfair outcomes. This stage is crucial for ensuring that the final decisions made by the model are equitable across various groups, especially those defined by sensitive attributes.

One common method in post-processing is to adjust the decision thresholds for different groups. For example, if a model is found to be less accurate for a particular demographic group, the decision threshold can be altered for that group to compensate for this disparity. This method aims to equalize the performance of the model across different groups, thereby reducing unfair biases. Another approach involves recalibrating the model's outputs to ensure that the probabilities of outcomes are fair and balanced. This recalibration is often based on statistical techniques that aim to make the model's predictions more just and less biased towards any particular group.

It's important to note that while post-processing techniques are essential for mitigating biases in model outputs, they are often seen as a last resort. This is because they do not address the root cause of the bias, which might lie in the data or the model itself. However, in many practical applications, post-processing provides a crucial checkpoint to ensure fairness. It acts as a safeguard where biased decisions can have significant consequences. Thus, post-processing is a vital component in the suite of techniques used to ensure fairness in machine learning, offering a means to adjust and refine model outputs to uphold ethical standards and societal values.

6.3.1 Calibrated Equalized Odds

Calibrated Equalized Odds research explores the challenge of striking a balance between reducing differences in error rates across various population groups while keeping the probability estimates calibrated.

In order to provide clarity within the context of the study, it is necessary to begin by explaining some essential metrics that are employed in it.

Error rates across various population groups refer to the differences in how a machine learning model's predictions perform when applied to different subgroups or demographic categories within a dataset. These error rates are typically associated with binary classification tasks, where the model is tasked with categorising data points into one of two classes.

FPR (False Positive Rate) and FNR (False Negative Rate) are key error rates for the study of Calibrated Equalized Odds. False positive rate calculates the proportion of actual negative instances that the model incorrectly predicts as positive. It is also known as the Type I error rate. The formula of false positive rate is $FPR = \frac{FP}{FP+TN}$, where FP (False Positive) is the number of negative instances incorrectly classified as positive and TN (True Negative) is the number of negative instances correctly classified as negative. On the other hand, false negative rate measures the proportion of actual positive instances that the model incorrectly predicts as negative. It is also known as the Type II error rate. The formula of false negative rate is $FNR = \frac{FN}{FN+TP}$ where FN (False Negative) is the number of positive instances incorrectly classified as negative and TN (True Negative) is the number of positive instances correctly classified as positive.

Equalized Odds is a fairness concept in machine learning, particularly in binary classification, that aims to ensure that the false positive and false negative rates are roughly equal across different demographic groups or population subsets. In the context of Equalized Odds for each subgroup, the FPR and FNR should be approximately the same. The goal of Equalized Odds is to ensure that the model doesn't disproportionately affect any particular group, leading to more equitable treatment. It helps prevent scenarios where, for example, one group experiences a higher false positive rate, potentially leading to unjust outcomes, while another group might have a higher false negative rate, causing missed opportunities.

Calibrated probabilities, also known as calibrated probability estimates, refer to the predicted probabilities produced by a machine learning model that have been scaled to align with the actual probabilities in the real world. In other words, these probabilities are carefully calibrated to be accurate representations of the true likelihood of an event or outcome occurring. Furthermore, calibration ensures that the model's predictions are not only accurate but also reliable. When probabilities are not well-calibrated, users may not trust the model's predictions, leading to potentially incorrect decisions.

The primary goal of this investigation is to delve deeper into the relationship between calibration and error rates. Furthermore, the research introduces a simple post-processing algorithm, involving withholding predictive information for randomly selected inputs, to achieve fairness and maintain calibration.

The research framework focuses on a binary classification task with an input space represented by $P \subset R_k \times \{0, 1\}$. In our dataset example, (x, y) ~ P represents an individual's history and the likelihood of them to graduate from the university. Two distinct groups, G_1 and G_2, represent population subsets (e.g., different races) with varying base rates, μ_1 and μ_2, signifying the probabilities of belonging to the positive class.

The study also involves two binary classifiers, $h_1$and h_2, classifying samples from G_1 and G_2, respectively. These classifiers output the probability of a sample belonging to the positive class.

To ensure fairness and calibration, the study introduces a cost function dependent on false-positive and false-negative rates, with variations based on the group's base rate.The formula of the cost function is:
$g_t(h_t) = \alpha_t \cdot c_{f_p}(h_t) + \alpha_t \cdot c_{f_n}(h_t)$, where $c_{f_p}(h_t)$ and $c_{f_n}(h_t)$ are the false positive rate and false negative rate respectively of the classifier h_t and group G_t. Given a cost function g_t with classifiers h_1 and h_2 achieve Relaxed Equalized Odds with Calibration for groups G_1 and G_2 if both classifiers are calibrated and satisfy the constraint $g_1(h_1)$= $g_2(h_2)$.

The research assumes access to "optimal" calibrated classifiers $h_1$and h_2, which may be discriminatory but are the best available given the predictability constraints. It explores the challenge of finding a classifier, $\widehat{h_2}$, for group G_2that matches the cost of h_1 . This is achieved through an algorithm that occasionally withholds predictive information, preserving calibration while ensuring fairness.

One of the primary challenges with the Calibrated Equalized Odds algorithm is that it inherently involves trade-offs. For instance, maintaining calibration can increase disparities in false positive rates between groups. This trade-off suggests that the algorithm might be improving one aspect of fairness at the expense of another. The algorithm is found to be infeasible in situations where the best classifiers are close to being trivial, leaving little room for effective adjustment. This indicates a limitation in its applicability, particularly in settings where the classifiers are already performing at a near-optimal level without considering fairness. A significant downside is that the algorithm might involve randomly withholding predictive information to achieve fairness. This approach can be problematic, especially in critical applications, as it means that important decisions might be made based on chance rather than a comprehensive evaluation of the individual's features.

The comparison between the Calibrated Equalized Odds (CEO) algorithm and XGBoost in the context of machine learning algorithms involves a detailed assessment through both evaluation metrics and fairness metrics. These metrics provide insights into their overall predictive capabilities and their fairness across different demographic groups.

In the evaluation metrics domain, XGBoost shows a notable edge. It achieves a perfect Balanced Accuracy of 1, clearly outperforming the CEO algorithm's 0.8946. This suggests XGBoost's superior ability in accurately classifying each class. Regarding the F1-score, the CEO algorithm has a slight advantage with a score of 0.9256, slightly higher than XGBoost's 0.9249. In the ROC-AUC metric, XGBoost with a score of 0.9518 and CEO with 0.9486. These scores collectively indicate that while XGBoost is more accurate in balanced accuracy, CEO and XGBoost are nearly comparable in f1-score and ROC-AUC, demonstrating similar overall predictive performances.

Analysing the fairness metrics for the attribute 'Previous_qualification', Statistical Parity for XGBoost registers a value of 0.537, while CEO is slightly lower at 0.5361. Statistical Parity suggesting that both algorithms are almost equally fair in this regard, with a negligible advantage for CEO. In terms of Disparate Impact, both models again exhibit similar values, with XGBoost at 5.8333 and CEO marginally lower at 5.825. The Equal Opportunity Difference metric shows an identical performance for both algorithms, recorded at 0.4573. The identical scores for XGBoost and CEO suggest that both algorithms offer a similar level of fairness in correctly identifying positive outcomes for different groups. Lastly, the Average Odds Difference is another closely contested metric, with XGBoost presenting a value of 0.3145, and CEO slightly lower at 0.3133 indicating a near-equivalent performance in fairness from both algorithms.

Turning to the fairness metrics, the CEO algorithm shows a Statistical Parity Difference of 0.4488, a Disparate Impact Ratio of 2.7954, an Equal Opportunity Difference of 0.1543, and an Average Odds Difference of 0.1097 in the context of ‘Debtor’ attribute. In comparison, XGBoost records slightly higher values with a Statistical Parity Difference of 0.4667, a Disparate Impact Ratio of 2.9835, an Equal Opportunity Difference of 0.2344, and an Average Odds Difference of 0.1513. This suggests that CEO is marginally fairer in its predictions regarding ‘Debtor’.

For ‘Tuition_fees_up_to_date’, the CEO algorithm demonstrates lesser bias, evidenced by a Statistical Parity Difference of -0.6779, a Disparate Impact Ratio of 0.0753, an Equal Opportunity Difference of -0.4601, and an Average Odds Difference of -0.3264. XGBoost, however, shows very similar values: a Statistical Parity Difference of -0.6869, a Disparate Impact Ratio of 0.0657, an Equal Opportunity Difference of -0.4601, and an Average Odds Difference of -0.3334. These figures suggest that both algorithms exhibit a comparable level of fairness in terms of ‘Tuition_fees_up_to_date’.

In summary, XGBoost exhibits superior performance in terms of balanced accuracy, but CEO and XGBoost are nearly on par in f1-score and ROC-AUC metrics, indicating similar overall predictive effectiveness. In terms of fairness, especially in the context of ‘Debtor’, the CEO algorithm shows slightly better results, though the differences are not substantial. For 'Previous_qualification' and ‘Tuition_fees_up_to_date’ attribute, both models demonstrate similar levels of fairness.

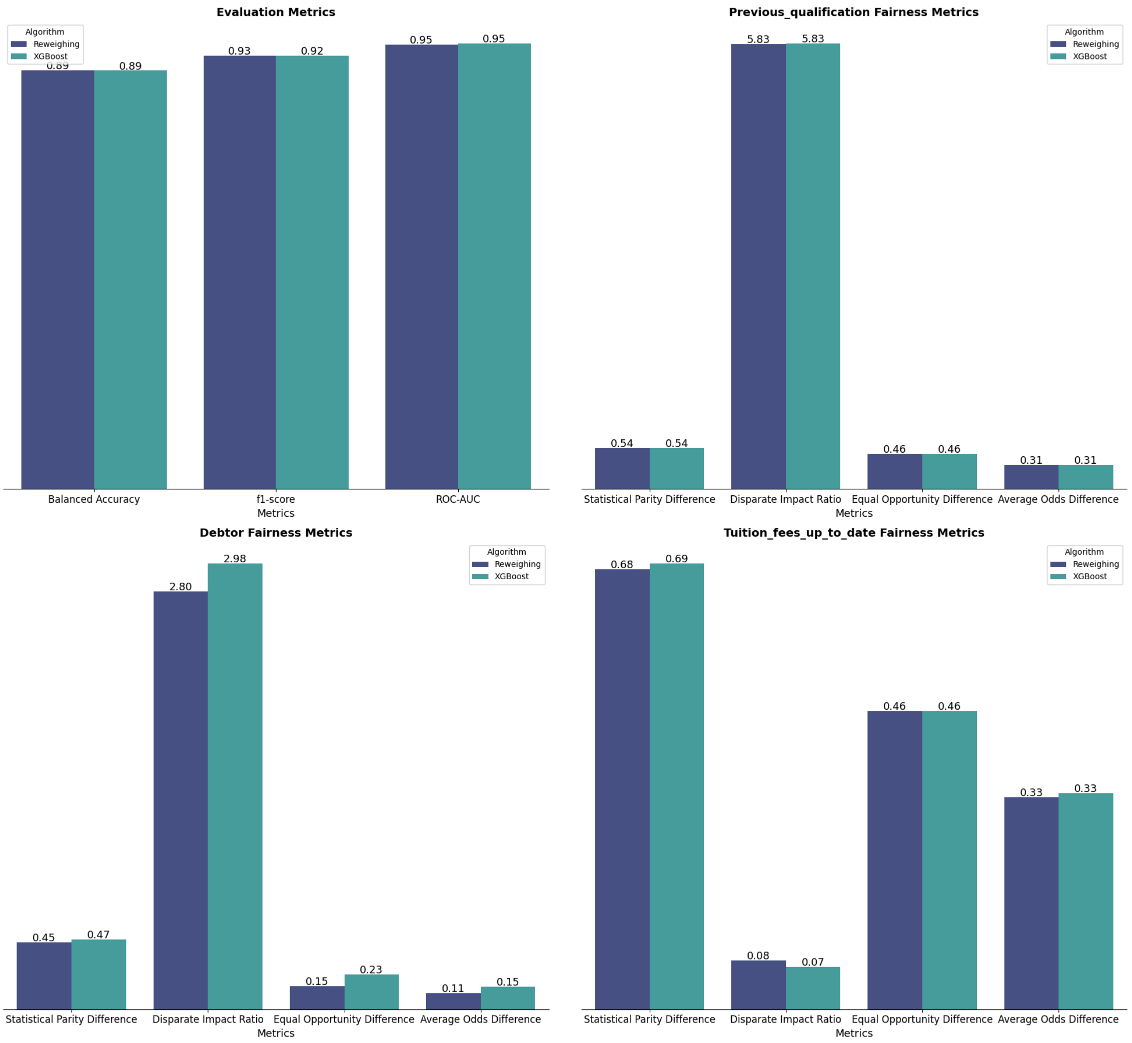


Figure 16: Evaluation and Fairness Metrics results of Calibrated Equalized Odds algorithm

6.3.2 Reject Option based Classification

To begin with, the research highlights the constraints of prior investigations and the void addressed by this study. Firstly, prior methods often necessitate preprocessing the data to remove discriminatory patterns or modifying the learning algorithm of a classifier to make it aware of discrimination. Secondly, they lack the flexibility to control discrimination effectively. An immediate consequence of the first limitation is that whenever discrimination concerning a different sensitive attribute or set of attributes requires attention, one must reprocess the historical data or classifier. Additionally, being constrained to a specific discrimination-aware classifier, such as naive Bayes or decision tree, is problematic because such a classifier may not be the most suitable choice for a given dataset.

This study introduces one user-friendly and versatile solution for discrimination-aware classification, based on the hypothesis that discriminatory decisions often cluster near the decision boundary due to inherent biases in the decision-making process.

The Reject Option based Classification (ROC) method is designed to reduce discrimination by targeting the low-confidence region of one or an ensemble of probabilistic classifiers. More specifically, ROC employs a "reject option" to label instances belonging to deprived and favoured groups in a manner that actively reduces discrimination.

These proposed solution offer several advantages over existing discrimination-aware classification methods:

1. They are not limited to a specific classifier: the solution is compatible with any probabilistic classifier.
2. The proposed methods do not require altering the learning algorithm or preprocessing the historical data. Pre-trained classifiers can be made discrimination-aware during prediction, simplifying the handling of changes in sensitive attributes.
3. These solutions grant decision makers superior control and interpretability over discrimination-aware classification.

The study focuses on a two-class problem with labels (C^+ and C^-) assigned to instances described by a fixed number of attributes. A discriminatory dataset (D) is provided, where the labels may exhibit bias concerning sensitive attributes.

It is assumed that C^+ represents the desirable label, and instances can be categorised as belonging to either a deprived group (X^d) or a favoured group (X^f), with these groups being mutually exclusive. All instances in the deprived group share specific values for certain attributes, which are referred to as sensitive attributes.

The primary objective is to develop a classifier (F) that does not make discriminatory decisions based on the sensitive attribute(s) due to legal constraints.

Performance evaluation of discrimination-aware classification methods involves reporting accuracy and discrimination metrics, with the ideal scenario being minimal accuracy loss as discrimination is reduced to zero.

In the traditional approach to classification, a learned classifier assigns an instance to a class based on the highest posterior probability, meaning the class with the greatest likelihood of being correct.

More specifically, consider a single classifier, and let $p(C^{+}|X)$ be the posterior probability of instance *X* produced by this classifier.

1. When $p(C^{+}|X)$ is close to 1 or 0 then the label for instance *X* is specified with a high degree of certainty.
2. When $p(C^{+}|X)$ is close to 0.5 then the label of instance *X* is more uncertain.

However, ROC introduces a departure from this conventional decision-making rule and introduces the concept of a *critical region*. The critical region is defined for all the instances for which $max[p(C^{+}|X),\ 1 - p(C^{+}|X)] \leq \theta,\ 0.5 < \theta < 1.$

Instances in this critical region are marked as "reject," indicating ambiguity and bias influence. To mitigate discrimination, these rejected instances are labelled as below:

1. instances from the deprived group (X^{d}) are labelled C^{+} and
2. instances from the favoured group (X^{f}) are labelled C^{-}.

while those outside the critical region are classified based on the standard decision rule.

In the case of multiple classifiers, classifier ensembles offer increased robustness. A classifier ensemble, in this context, functions as a collection of experts with diverse characteristics and biases, expected to yield more reliable results in terms of both accuracy and discrimination. The study considers the posterior probabilities produced by individual classifiers in the ensemble and combines them to make a classification decision, factoring in the accuracy of each classifier. This method leverages the strengths of multiple classifiers to provide an effective control over the accuracy-discrimination trade-off in future classifications.

However, a key challenge is its dependence on accurately identifying the decision boundary in complex datasets. This process is critical as ROC uses uncertainty around these boundaries to mitigate bias. However, achieving a balance between reducing discrimination and maintaining classification accuracy can be difficult. Overuse or inaccurate application of the reject option may lead to a decline in overall classifier performance. Additionally, implementing and integrating ROC into existing systems not originally designed with discrimination-awareness can present hurdles. The algorithm's effectiveness also varies depending on dataset characteristics and the nature of sensitive attributes involved, making it necessary to tailor the ROC approach to specific contexts.

The comparison of Reject Option based Classification and the XGBoost algorithm is essential in understanding the trade-offs between predictive accuracy and fairness in algorithmic decision-making.

In terms of evaluation metrics, which include Balanced Accuracy, f1-score, and ROC-AUC, a distinct contrast is observed between the two algorithms. The XGBoost algorithm demonstrates superior performance with a perfect score in Balanced Accuracy and significantly higher scores in F1-score and ROC-AUC. This indicates a strong ability in accurately predicting outcomes and distinguishing between classes. On the other hand, the Reject Option based Classification algorithm shows moderate performance with scores of 0.5969 in Balanced Accuracy, 0.7896 in F1-score, and 0.9448 in ROC-AUC. While these scores are commendable, especially in ROC-AUC, they are noticeably lower than those of the XGBoost algorithm.

Fairness metrics are pivotal in evaluating how equitably algorithms treat different groups. For the attribute 'Previous_qualification', the Reject Option based Classification algorithm demonstrates a significantly lower Statistical Parity Difference of 0.0315 compared to XGBoost's 0.537. This indicates a substantially more balanced treatment of different groups in terms of outcome rates by the Reject Option algorithm. In terms of Disparate Impact Ratio, the Reject Option algorithm shows a score of 1.0354, much closer to the ideal ratio of 1, and considerably lower than XGBoost's high score of 5.8333. This suggests that the Reject Option algorithm produces far less Disparate Impact compared to XGBoost. Regarding Equal Opportunity Difference, the Reject Option algorithm again shows an advantage with a score of -0.0031, indicating almost no disparity. This is in stark contrast to XGBoost's higher score of 0.4573, suggesting greater disparity in accurately identifying positive outcomes for different groups. Finally, the Average Odds Difference, is -0.0289 for the Reject Option algorithm, reflecting a more equitable treatment across groups. This is again a more favourable outcome compared to XGBoost's score of 0.3145.

For ‘Debtor’, the Reject Option based Classification algorithm shows a Statistical Parity Difference of 0.6397 and a Disparate Impact Ratio of 2.7755, suggesting some degree of bias. Its Equal Opportunity Difference and Average Odds Difference scores are lower, at 0.0769 and 0.4248, respectively, indicating relatively fairer treatment in these aspects. In comparison, the XGBoost Algorithm, while achieving lower bias in Statistical Parity Difference (0.4667) and a slightly higher Disparate Impact Ratio (2.9835), shows more significant disparities in Equal Opportunity Difference (0.2344) and Average Odds Difference (0.1513).

For 'Tuition_fees_up_to_date' fairness metrics, both algorithms display significant biases, but in different ways. The Reject Option based Classification algorithm demonstrates less disparity in comparison to XGBoost in certain metrics. It records a Statistical Parity Difference of -0.4409, an Equal Opportunity Difference of -0.3333 and a Disparate Impact Ratio of 0.5496, both lower than XGBoost's respective scores of -0.6869,-0.4601, 0.0657. Conversely, when examining the and Average Odds Difference, the Reject Option based Classification shows more bias. Its scores of -0.3663 for Average Odds Difference are higher compared to XGBoost's scores of -0.3334.

Overall, the comparison reveals a significant trade-off between accuracy and fairness. The XGBoost Algorithm excels in predictive performance but tends to show more bias in fairness metrics. In contrast, the Reject Option based Classification algorithm, while not as accurate, appears to be fairer in its predictions, particularly in the context of 'Previous_qualification.'

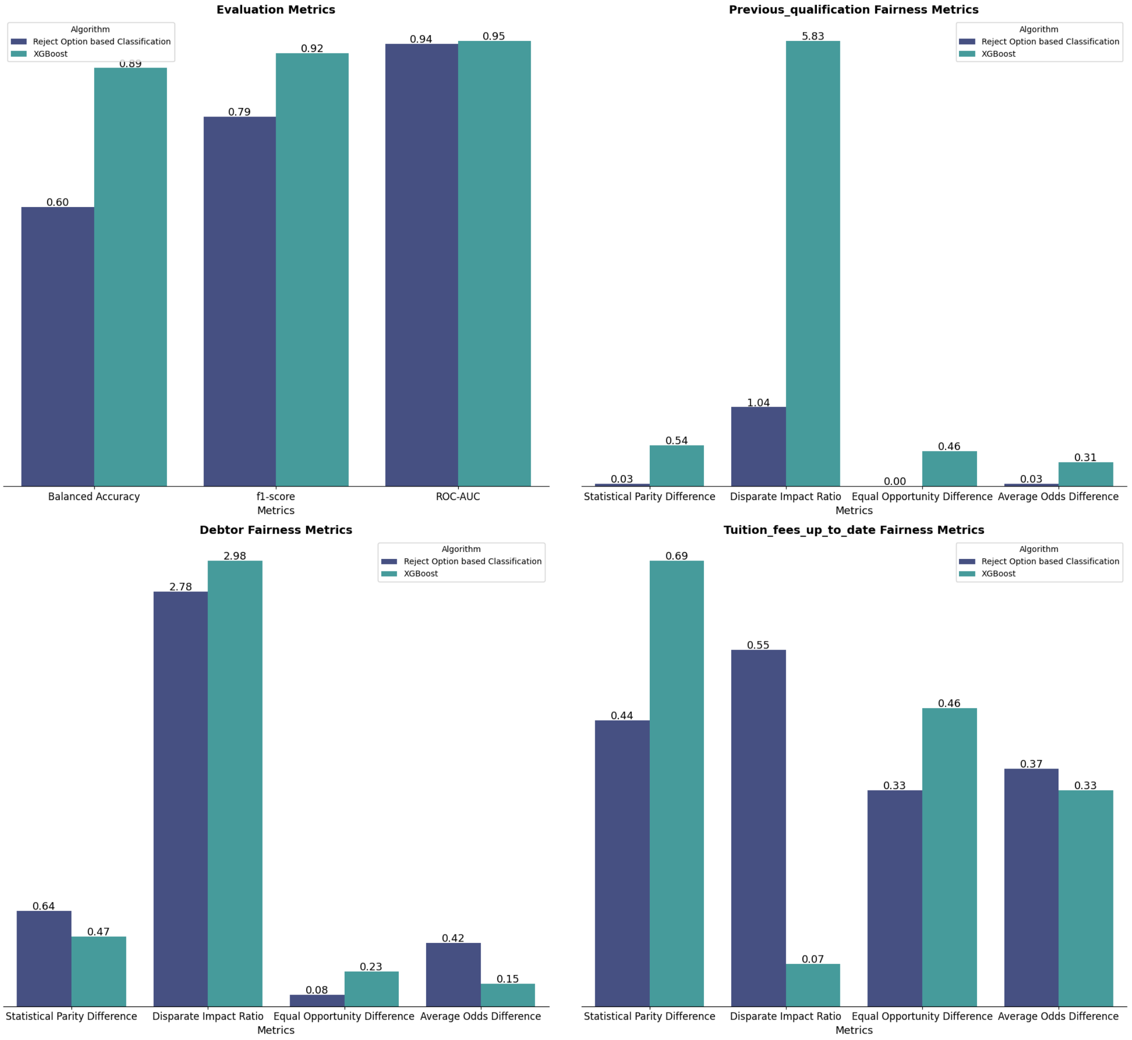


Figure 17: Evaluation and Fairness Metrics results of Reject Option based Classification algorithm

6.4 Results

The analysis of seven machine learning algorithms based on evaluation metrics and fairness metrics across various protective attributes reveals nuanced strengths and weaknesses in both performance and fairness. The summary metrics provided in Table 3, Table 4, Table 5 and Table 6.

The bar plot in Figure 18 appears to compare the algorithms across three evaluation metrics: Balanced Accuracy, F1-score, and ROC-AUC. Reject Option-based Classification is the least performing algorithm with the lowest scores across most of the metrics (Balanced Accuracy: 0.5969, F1-score: 0.7896, ROC-AUC: 0.9448). Learning Fair Representation improves significantly on Balanced Accuracy (0.8476) but still has low F1-score and ROC-AUC scores (0.8972 and 0.8913 respectively). Exponentiated Gradient Reduction is lower than the first two in ROC-AUC (0.8858) but has a higher Balanced Accuracy and ROC-AUC score(0.8858 and 0.9169 respectively). Adversarial Debiasing shows some improvement in Balanced Accuracy (0.8711) and maintains a F1-score of 0.8982, indicating it is better than the previous algorithms but still not the top performer. Reweighing, matches the Balanced Accuracy of the better-performing algorithms (0.8785), placing it higher in the rank. Calibrated Equalized Odds exhibits consistent performance across all metrics (Balanced Accuracy: 1, F1-score: 0.9256, ROC-AUC: 0.9486), suggesting a better balance than Reweighing. Finally, XGBoost stands out as the best performer with the highest Balanced Accuracy (0.85), the best F1-score (0.9312), and a strong ROC-AUC (0.9518), making it the top algorithm in this evaluation.

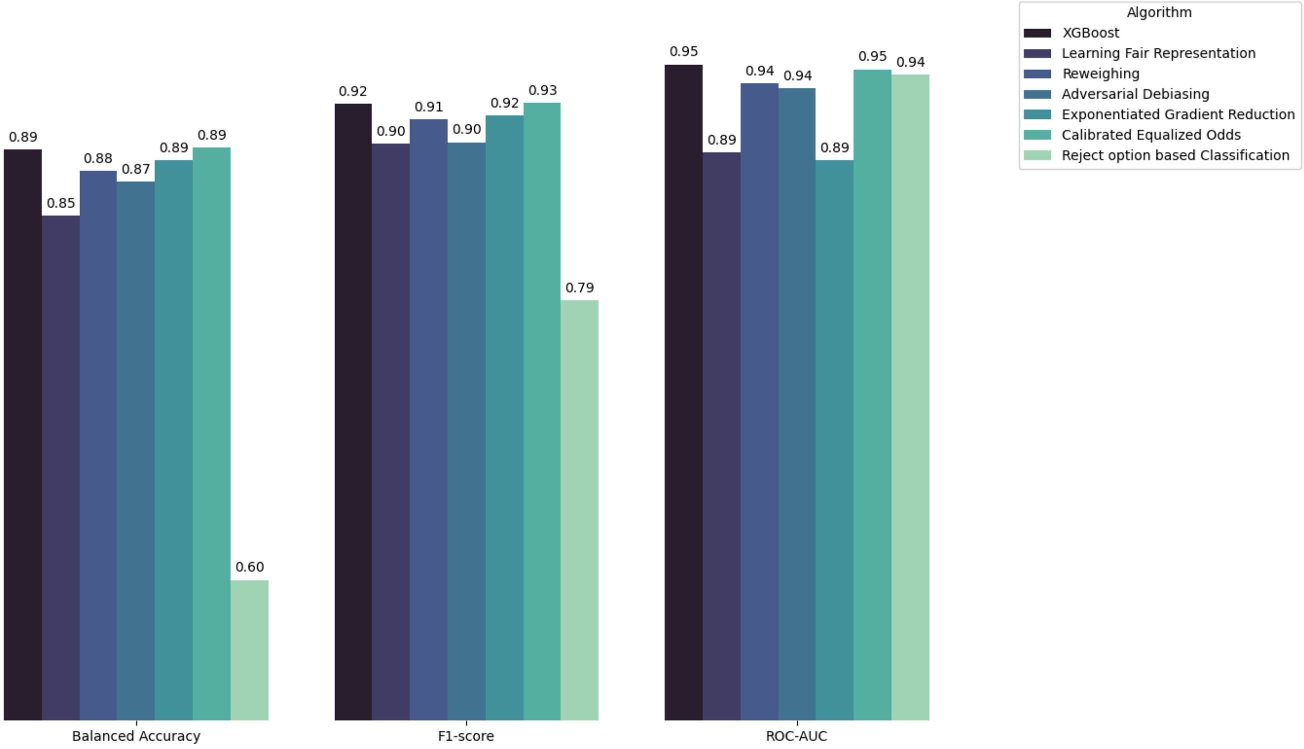


Figure 18: Evaluation metrics results of each algorithm

The bar plot in Figure 19 appears to compare the algorithms across the fairness metrics of ‘Previous_qualification’ protective attribute. Starting from the algorithm with the most potential for improvement according to the fairness metrics, Learning Fair Representation has the higher Statistical Parity Difference (0.5676), Disparate Impact Ratio (6.1089) is significantly above 1, indicating potential over-adjustment. Calibrated Equalized Odds and XGBoost both have a Disparate Impact Ratio closer to 1 (5.8250 and 5.8333 respectively), which indicates disparity. The same picture appears in their Statistical Parity Differences (0.5361 and 0.537 respectively) and Average Odds Differences (0.3133 and 0.3145 respectively) indicate that there is still disparity to be addressed. Adversarial Debiasing with a slightly better Average Odds Difference (0.2804) and Statistical Parity Difference (0.4981). Reweighing shows a much better score in Disparate Impact Ratio (2.925), yet its other metrics suggest imbalance. Reweighing shows a similar pattern to Exponentiated Gradient Reduction that has high Statistical Parity Difference and Equal Opportunity Difference (0.4157 and 0.4420 respectively), but its relatively low Disparate Impact Ratio (2.8708). Finally, Reject Option-based Classification, has the lowest disparity across all the metrics with Statistical Parity Difference (0.0315), Disparate Impact Ratio (1.0354), Equal Opportunity Difference (-0.0031) and Average Odds Difference (-0.0289), implying it is the most fair algorithm.

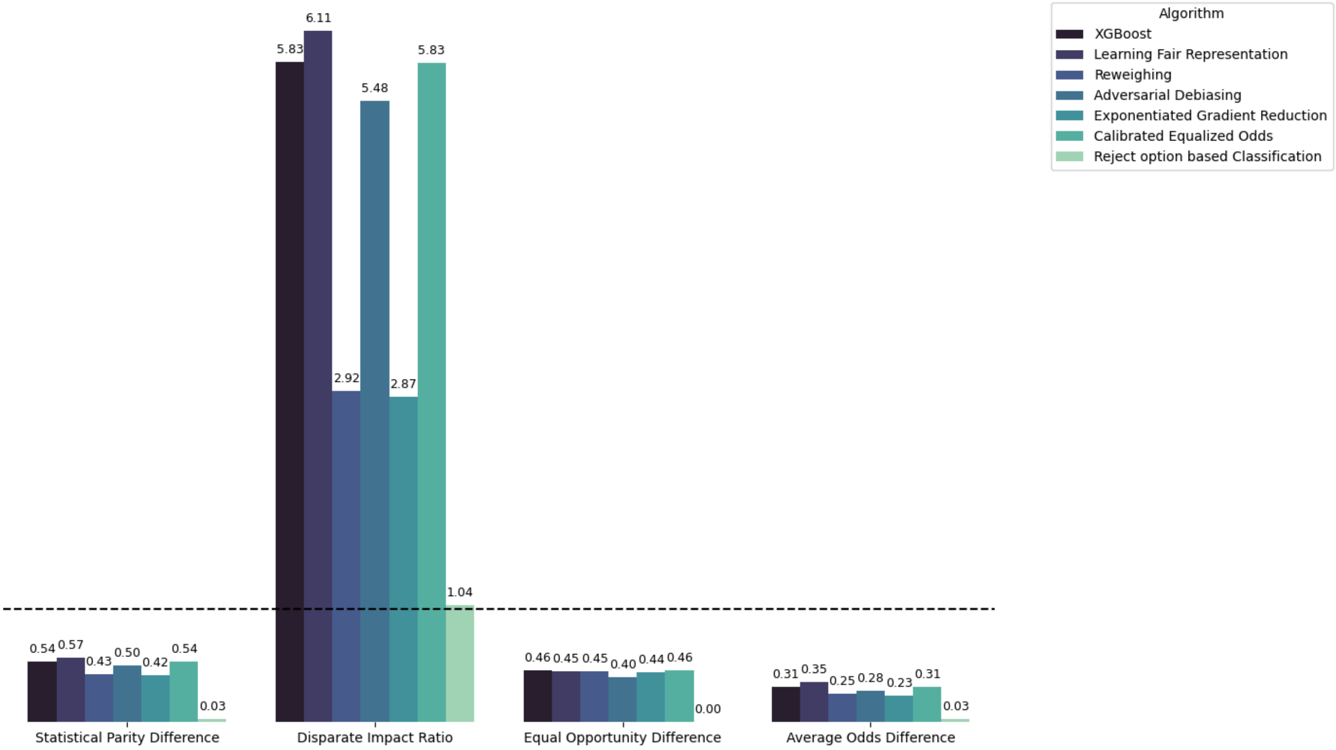


Figure 19: Fairness metrics of Previous_qualification

The bar plot in Figure 20 appears to compare the algorithms across the fairness metrics of the ‘Debtor’ protective attribute. XGBoost shows the highest Disparate Impact Ratio (2.9835). It also has a high Statistical Parity Difference (0.4667). Reject Option-based Classification has a higher Statistical Parity Difference (0.6397), but its Disparate Impact Ratio (2.7755) is lower than XGBoost, but still signifying potential over-adjustment. Calibrated Equalized Odds, with a Disparate Impact Ratio (2.7954) still distant from 1, shows it is moderately fair with Statistical Parity Difference (0.4488) and Average Odds Difference (0.1097). Reweighing follows with a better Disparate Impact Ratio (1.9949) and a lower Equal Opportunity Difference (0.1448), indicating less bias in terms of favourable outcomes across groups. Adversarial Debiasing shows the best Average Odds Difference (0.0053), very close to the ideal, and very low Equal Opportunity Difference (0.0556). Exponentiated Gradient Reduction further improves with a Disparate Impact Ratio closer to 1 (1.8698) and the smallest Equal Opportunity Difference (-0.0218), suggesting greater fairness in positive outcome rates. Lastly, Learning Fair Representation has the smallest Statistical Parity

Difference (0.0560) and the lowest Equal Opportunity Difference (-0.0523), making it the best algorithm in terms of fairness metrics provided in this plot.

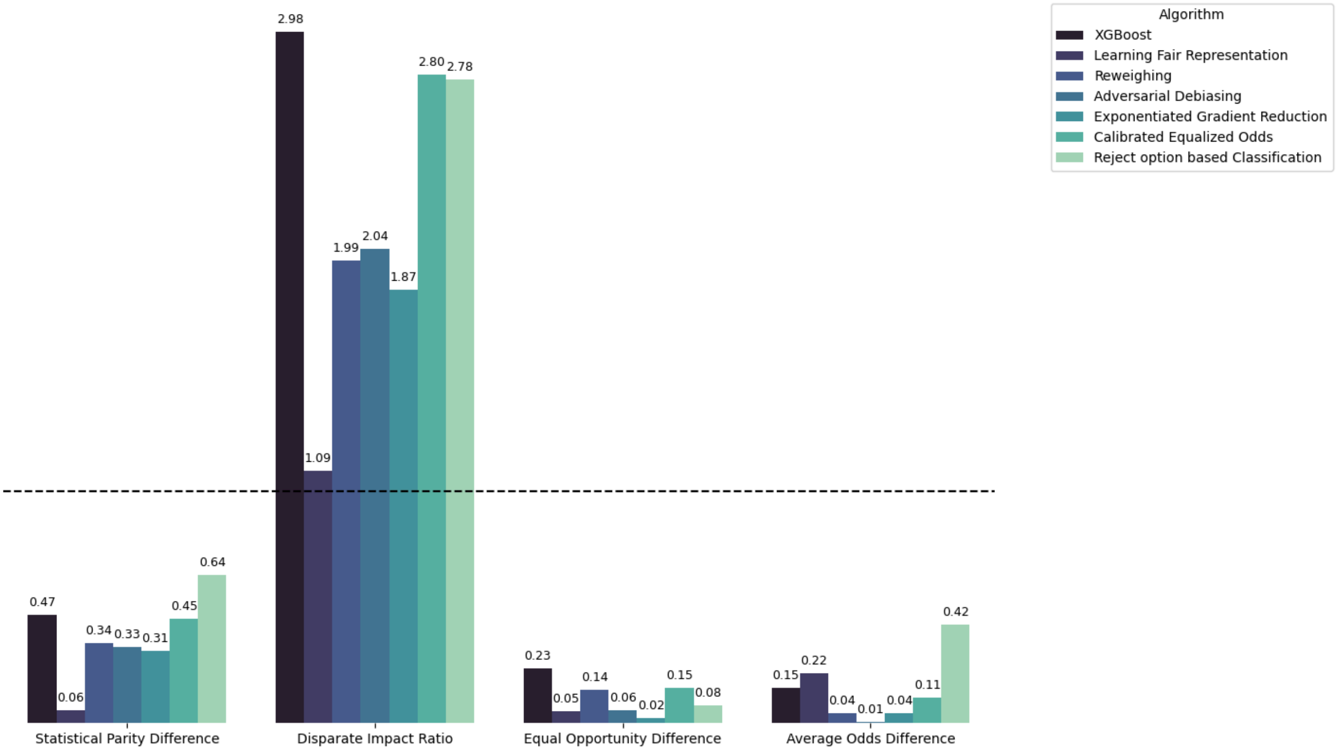


Figure 20: Fairness metrics results of Debtor

The bar plot in Figure 21 appears to compare the algorithms across the fairness metrics of the ‘Tuition_fees_up_to_date’ protective attribute. XGBoost exhibits the most significant challenges. With a Disparate Impact Ratio of only 0.0657, it indicates a substantial imbalance. Its Statistical Parity Difference is the highest at 0.6869, further underscoring the concerns. Calibrated Equalized Odds follows closely in terms of disparity, with a Disparate Impact Ratio of 0.0753 and a high Statistical Parity Difference of 0.6779, suggesting notable imbalances in its outcomes.Conversely, Learning Fair Representation, though ranked lowest in overall performance, shows a more balanced approach with a moderate Disparate Impact Ratio of 0.3162 and a Statistical Parity Difference of 0.507. Exponentiated Gradient Reduction demonstrates improvement, with a Disparate Impact Ratio closer to a balanced state at 0.2647 and a lower Equal Opportunity Difference of 0.0599, indicating a more equitable treatment of outcomes. Adversarial Debiasing also shows commendable performance with a Disparate Impact Ratio of 0.2143, coupled with a low Equal Opportunity Difference of

0.0668. It achieves better fairness in outcome distribution than Reweighing, which, despite a better Disparate Impact Ratio of 0.2497, has a slightly higher Equal Opportunity Difference of 0.1144. Reject Option-based Classification emerges as more balanced with a Disparate Impact Ratio of 0.5496 and the lowest Statistical Parity Difference among all algorithms at 0.4409. This suggests a more nuanced adjustment in balancing fairness metrics, although there's still room for improvement.

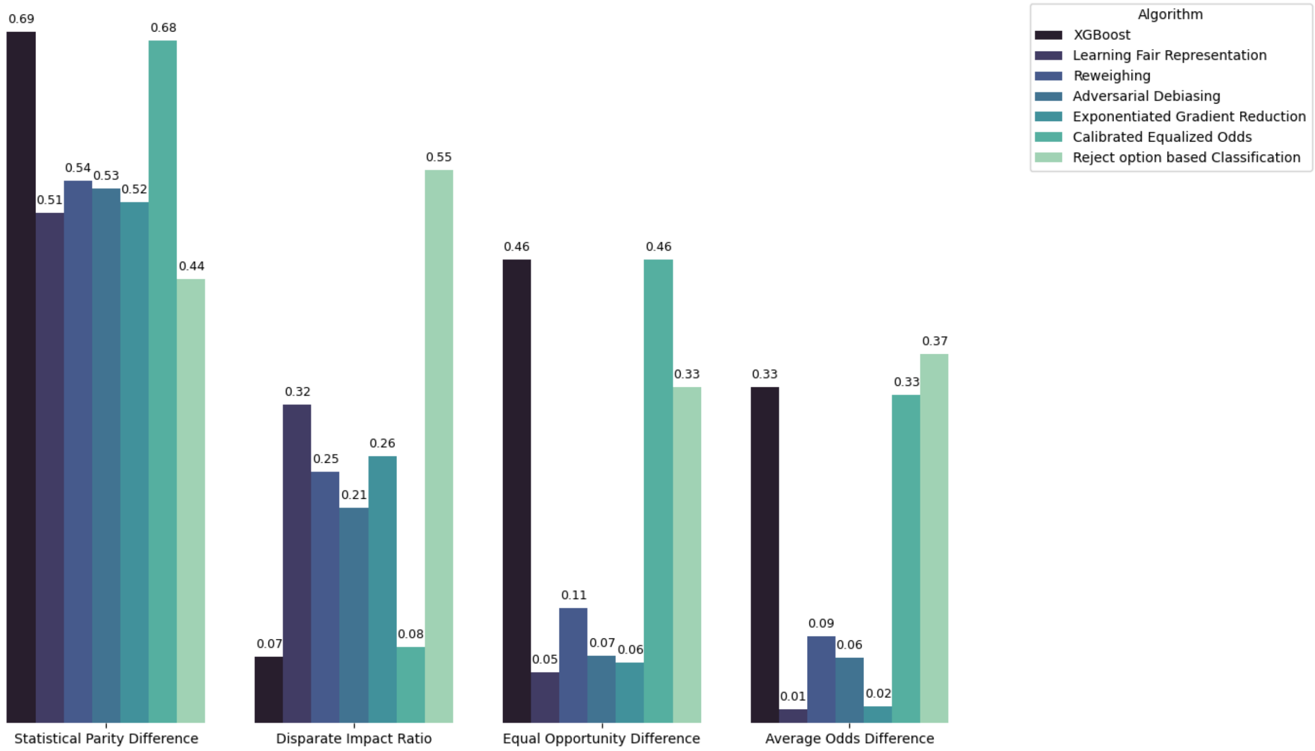


Figure 21: Fairness metrics of Tuition_fees_up_to_date

Finally, selecting the most efficient algorithm for machine learning tasks is contingent on the specific goals of the prediction model. If the primary concern is high accuracy, such as in predicting equipment failures where the costs of false negatives are substantial, XGBoost may be the algorithm of choice due to its strong performance in accuracy-related metrics. Conversely, if the aim is to balance accuracy with fairness, which is critical in applications like credit scoring to prevent discriminatory practices, then algorithms like Exponentiated Gradient Reduction, Adversarial Debiasing, or Reweighing come to the fore. These algorithms have shown the best performances in balancing both fairness and accuracy. The choice among them would be influenced by factors such as the desired fairness metric, the extent and nature of existing biases in the data and the specific trade-offs between different types of predictive errors that one is willing to accept. An optimal algorithm would thus be one that not only meets the performance criteria but also aligns with ethical standards, legal compliance, and the operational context of the application.

7. Conclusion

This thesis embarked on an in-depth exploration of balancing accuracy and fairness in machine learning algorithms, with a particular emphasis on educational data. It meticulously evaluated seven different algorithms, revealing that XGBoost excelled in accuracy but showed biases in fairness metrics. In contrast, other algorithms such as Exponentiated Gradient Reduction, Reweighing or Adversarial Debiasing, while not as accurate, demonstrated a higher degree of fairness. This difference highlights the complexity and trade-offs involved in developing fair machine learning models, making it a central theme of the research.

Therefore, the necessity of future research should aim to develop new algorithms that better balance accuracy with fairness, investigate the impact of diverse data preprocessing methods, and incorporate a broader array of fairness metrics. Such advancements are crucial in sectors where fairness is paramount, such as education and employment, underlining the significant practical applications of this research.

Reflecting on this thesis, it becomes evident that pursuing fairness in machine learning is both challenging and crucial. As algorithms increasingly influence various aspects of our lives, ensuring they operate with not just efficiency and accuracy but also fairness and justice becomes imperative.

List of Tables

Table 1: Variables types and values description

Variable Name	**Type**	**Description**
Marital Status	Integer	1 – single 2 – married 3 – widower 4 – divorced 5 – facto union 6 – legally separated
Application mode	Integer	1 - 1st phase - general contingent 2 - Ordinance No. 612/93 5 - 1st phase - special contingent (Azores Island) 7 - Holders of other higher courses 10 - Ordinance No. 854-B/99 15 - International student (bachelor) 16 - 1st phase - special contingent (Madeira Island) 17 - 2nd phase - general contingent 18 - 3rd phase - general contingent 26 - Ordinance No. 533-A/99, item b2) (Different Plan) 27 - Ordinance No. 533-A/99, item b3 (Other Institution) 39 - Over 23 years old 42 - Transfer 43 - Change of course 44 - Technological specialisation diploma holders 51 - Change of institution/course 53 - Short cycle diploma holders 57 - Change of institution/course (International)
Application order	Integer	Application order (between 0 - first choice; and 9 last choice)
Course	Integer	33 - Biofuel Production Technologies 171 - Animation and Multimedia Design 8014 - Social Service (evening attendance) 9003 - Agronomy 9070 - Communication Design 9085 - Veterinary Nursing 9119 - Informatics Engineering 9130 - Equinculture 9147 - Management 9238 - Social Service 9254 - Tourism

		9500 - Nursing 9556 - Oral Hygiene 9670 - Advertising and Marketing Management 9773 - Journalism and Communication 9853 - Basic Education 9991 - Management (evening attendance)
Daytime/evening attendance	Integer	1 – daytime 0 - evening
Previous qualification	Integer	1 - Secondary education 2 - Higher education - bachelor's degree 3 - Higher education - degree 4 - Higher education - master's 5 - Higher education - doctorate 6 - Frequency of higher education 9 - 12th year of schooling - not completed 10 - 11th year of schooling - not completed 12 - Other - 11th year of schooling 14 - 10th year of schooling 15 - 10th year of schooling - not completed 19 - Basic education 3rd cycle (9th/10th/11th year) or equiv. 38 - Basic education 2nd cycle (6th/7th/8th year) or equiv. 39 - Technological specialisation course 40 - Higher education - degree (1st cycle) 42 - Professional higher technical course 43 - Higher education - master (2nd cycle)
Previous qualification (grade)	Continuous	Grade of previous qualification (between 0 and 200)
Nationality	Integer	1 - Portuguese 2 - German 6 - Spanish 11 - Italian 13 - Dutch 14 - English 17 - Lithuanian 21 - Angolan 22 - Cape Verdean 24 - Guinean 25 - Mozambican 26 - Santomean 32 - Turkish 41 - Brazilian 62 - Romanian 100 - Moldova (Republic of) 101 - Mexican

		103 - Ukrainian 105 - Russian 108 - Cuban 109 - Colombian
Mother's qualification	Integer	1 - Secondary Education- 12th Year of Schooling or Eq. 2 - Higher Education - Bachelor's Degree 3 - Higher Education - Degree 4 - Higher Education - Master's 5 - Higher Education - Doctorate 6 - Frequency of Higher Education 9 - 12th Year of Schooling - Not Completed 10 - 11th Year of Schooling - Not Completed 11 - 7th Year (Old) 12 - Other - 11th Year of Schooling 14 - 10th Year of Schooling 18 - General commerce course 19 - Basic Education 3rd Cycle (9th/10th/11th Year) or Equiv. 22 - Technical-professional course 26 - 7th year of schooling 27 - 2nd cycle of the general high school course 29 - 9th Year of Schooling - Not Completed 30 - 8th year of schooling 34 - Unknown 35 - Can't read or write 36 - Can read without having a 4th year of schooling 37 - Basic education 1st cycle (4th/5th year) or equiv. 38 - Basic Education 2nd Cycle (6th/7th/8th Year) or Equiv. 39 - Technological specialisation course 40 - Higher education - degree (1st cycle) 41 - Specialised higher studies course 42 - Professional higher technical course 43 - Higher Education - Master (2nd cycle) 44 - Higher Education - Doctorate (3rd cycle)
Father’s qualification	Integer	1 - Secondary Education - 12th Year of Schooling or Eq. 2 - Higher Education - Bachelor's Degree 3 - Higher Education - Degree 4 - Higher Education - Master's 5 - Higher Education - Doctorate 6 - Frequency of Higher Education 9 - 12th Year of Schooling - Not Completed 10 - 11th Year of Schooling - Not Completed 11 - 7th Year (Old)

		12 - Other - 11th Year of Schooling 13 - 2nd year complementary high school course 14 - 10th Year of Schooling 18 - General commerce course 19 - Basic Education 3rd Cycle (9th/10th/11th Year) or Equiv. 20 - Complementary High School Course 22 - Technical-professional course 25 - Complementary High School Course - not concluded 26 - 7th year of schooling 27 - 2nd cycle of the general high school course 29 - 9th Year of Schooling - Not Completed 30 - 8th year of schooling 31 - General Course of Administration and Commerce 33 - Supplementary Accounting and Administration 34 - Unknown 35 - Can't read or write 36 - Can read without having a 4th year of schooling 37 - Basic education 1st cycle (4th/5th year) or equiv. 38 - Basic Education 2nd Cycle (6th/7th/8th Year) or Equiv. 39 - Technological specialisation course 40 - Higher education - degree (1st cycle) 41 - Specialised higher studies course 42 - Professional higher technical course 43 - Higher Education - Master (2nd cycle) 44 - Higher Education - Doctorate (3rd cycle)
Mother's occupation	Integer	0 - Student 1 - Representatives of the Legislative Power and Executive Bodies, Directors, Directors and Executive Managers 2 - Specialists in Intellectual and Scientific Activities 3 - Intermediate Level Technicians and Professions 4 - Administrative staff 5 - Personal Services, Security and Safety Workers and Sellers 6 - Farmers and Skilled Workers in Agriculture, Fisheries and Forestry 7 - Skilled Workers in Industry, Construction and Craftsmen 8 - Installation and Machine Operators and Assembly Workers 9 - Unskilled Workers 10 - Armed Forces Professions

		90 - Other Situation 99 - (blank) 122 - Health professionals 123 - teachers 125 - Specialists in information and communication technologies (ICT) 131 - Intermediate level science and engineering technicians and professions 132 - Technicians and professionals, of intermediate level of health 134 - Intermediate level technicians from legal, social, sports, cultural and similar services 141 - Office workers, secretaries in general and data processing operators 143 - Data, accounting, statistical, financial services and registry-related operators 144 - Other administrative support staff 151 - personal service workers 152 - sellers 153 - Personal care workers and the like 171 - Skilled construction workers and the like, except electricians 173 - Skilled workers in printing, precision instrument manufacturing, jewellers, artisans and the like 175 - Workers in food processing, woodworking, clothing and other industries and crafts 191 - cleaning workers 192 - Unskilled workers in agriculture, animal production, fisheries and forestry 193 - Unskilled workers in extractive industry, construction, manufacturing and transport 194 - Meal preparation assistants
Father's occupation	Integer	0 - Student 1 - Representatives of the Legislative Power and Executive Bodies, Directors, Directors and Executive Managers 2 - Specialists in Intellectual and Scientific Activities 3 - Intermediate Level Technicians and Professions 4 - Administrative staff 5 - Personal Services, Security and Safety Workers and Sellers 6 - Farmers and Skilled Workers in Agriculture, Fisheries and Forestry 7 - Skilled Workers in Industry, Construction and Craftsmen 8 - Installation and Machine Operators and Assembly Workers 9 - Unskilled Workers 10 - Armed Forces Professions

		90 - Other Situation 99 - (blank) 101 - Armed Forces Officers 102 - Armed Forces Sergeants 103 - Other Armed Forces personnel 112 - Directors of administrative and commercial services 114 - Hotel, catering, trade and other services directors 121 - Specialists in the physical sciences, mathematics, engineering and related techniques 122 - Health professionals 123 - teachers 124 - Specialists in finance, accounting, administrative organisation, public and commercial relations 131 - Intermediate level science and engineering technicians and professions 132 - Technicians and professionals, of intermediate level of health 134 - Intermediate level technicians from legal, social, sports, cultural and similar services 135 - Information and communication technology technicians 141 - Office workers, secretaries in general and data processing operators 143 - Data, accounting, statistical, financial services and registry-related operators 144 - Other administrative support staff 151 - personal service workers 152 - sellers 153 - Personal care workers and the like 154 - Protection and security services personnel 161 - Market-oriented farmers and skilled agricultural and animal production workers 163 - Farmers, livestock keepers, fishermen, hunters and gatherers, subsistence 171 - Skilled construction workers and the like, except electricians 172 - Skilled workers in metallurgy, metalworking and similar 174 - Skilled workers in electricity and electronics 175 - Workers in food processing, woodworking, clothing and other industries and crafts 181 - Fixed plant and machine operators 182 - assembly workers 183 - Vehicle drivers and mobile equipment operators 192 - Unskilled workers in agriculture, animal production, fisheries and forestry

		193 - Unskilled workers in extractive industry, construction, manufacturing and transport 194 - Meal preparation assistants 195 - Street vendors (except food) and street service providers
Admission grade	Continuous	Admission grade (between 0 and 200)
Displaced	Integer	1 – yes 0 – no
Educational special needs	Integer	1 – yes 0 – no
Debtor	Integer	1 – yes 0 – no
Tuition fees up to date	Integer	1 – yes 0 – no
Gender	Integer	1 – male 0 – female
Scholarship holder	Integer	1 – yes 0 – no
Age at enrollment	Integer	Age of student at enrollment
International	Integer	1 – yes 0 – no
Curricular units 1st sem (credited)	Integer	Number of curricular units credited in the 1st semester
Curricular units 1st sem (enrolled)	Integer	Number of curricular units enrolled in the 1st semester
Curricular units 1st sem (evaluations)	Integer	Number of evaluations to curricular units in the 1st semester
Curricular units 1st sem (approved)	Integer	Number of curricular units approved in the 1st semester
Curricular units 1st sem (grade)	Integer	Grade average in the 1st semester (between 0 and 20)
Curricular units 1st sem (without evaluations)	Integer	Number of curricular units without evaluations in the 1st semester
Curricular units 2nd sem (credited)	Integer	Number of curricular units credited in the 2nd semester
Curricular units 2nd sem	Integer	Number of curricular units enrolled in the 2nd

(enrolled)		semester
Curricular units 2nd sem (evaluations)	Integer	Number of evaluations to curricular units in the 2nd semester
Curricular units 2nd sem (approved)	Integer	Number of curricular units approved in the 2nd semester
Curricular units 2nd sem (grade)	Integer	Grade average in the 2nd semester (between 0 and 20)
Curricular units 2nd sem (without evaluations)	Integer	Number of curricular units without evaluations in the 1st semester
Unemployment rate	Continuous	Unemployment rate (%)
Inflation rate	Continuous	Inflation rate (%)
GDP	Continuous	GDP
Target	Categorical	Target. The problem is formulated as a three category classification task (dropout, enrolled, and graduate) at the end of the normal duration of the course

Table 2: Farness metrics of each variable of XGBoost algorithm

Attribute	**Privileged Group**	**Statistical Parity Difference**	**Disparate Impact Ratio**	**Equal Opportunity Difference**	**Average Odds Difference**
Marital_status	1	0.1456	1.2852	0.0198	0.068
Application_mode	1	**0.3112**	**1.9337**	-0.0442	0.0628
Application_order	1	0.1537	1.2591	0.0304	0.0837
Course	9500	0.0762	1.1333	0.1777	-0.0117
Daytime_evening_attendance	1	-0.0706	0.8915	-0.0203	-0.0466
Previous_qualification	1	**0.537**	**5.8333**	**0.4573**	**0.3145**
Previous_qualification_grade	130	-	-	**0.9559**	0.0624
Nationality	1	-	-	**0.9559**	0.0624
Mother_s_qualification	1	-0.0234	0.9649	0.0279	0.0272
Father_s_qualification	37	0.0732	1.1281	**0.2442**	-0.0098
Mother_s_occupation	9	0.1842	1.3967	0.1009	**0.1008**
Father_s_occupation	9	-0.0104	0.9841	-0.0452	0.0127
Admission_grade	2	-0.063	0.9063	0.0109	-0.0247
Displaced	1	-0.1727	0.76	-0.0042	-0.0971
Educational_special_needs	0	-0.1071	0.8572	-0.0445	0.0626
Debtor	0	**0.4667**	**2.9835**	**0.2344**	**0.1513**
Tuition_fees_up_to_date	1	**-0.6869**	**0.0657**	**-0.4601**	**-0.3334**
Gender	0	**0.2553**	**1.5387**	0.0699	0.0845
Scholarship_holder	0	**-0.3444**	**0.6179**	-0.0414	**-0.1851**
Age_at_enrollment	1	**0.2348**	**1.4963**	0.057	0.0713
International	0	0.0358	1.0588	0.1248	-0.0375
Unemployment_rate	10.8	-0.1042	0.859	0.0022	-0.0273
Inflation_rate	1.4	-0.0037	0.9943	-0.0484	-0.0314
GDP	1.74	-0.0085	0.987	-0.0206	-0.0126

Table 3: Evaluation metrics of each algorithm

Algorithm	Balanced Accuracy	f1-score	ROC-AUC
Learning Fair Representation	0.8476	0.8972	0.8913
Reweighing	0.8785	0.9140	0.9388
Adversarial Debiasing	0.8711	0.8982	0.9354
Exponentiated Gradient Reduction	0.8858	0.9169	0.8858
Calibrated Equalized Odds	0.8946	0.9256	0.9486
Reject Option based Classification	0.5969	0.7896	0.9448

Table 4: Fairness metrics of 'Previous_qualifications' variable

Algorithm	Statistical Parity Difference	Disparate Impact Ratio	Equal Opportunity Difference	Average Odds Difference
Learning Fair Representation	0.5676	6.1083	0.4511	0.3550
Reweighing	0.4278	2.925	0.4481	0.2479
Adversarial Debiasing	0.4981	5.4833	0.4008	0.2804
Exponentiated Gradient Reduction	0.4157	2.8708	0.4420	0.2343
Calibrated Equalized Odds	0.5361	5.8250	0.4573	0.3133
Reject Option based Classification	0.0315	1.0354	-0.0031	-0.0289

Table 5: Fairness metrics of 'Debtor' variable

Algorithm	**Statistical Parity Difference**	**Disparate Impact Ratio**	**Equal Opportunity Difference**	**Average Odds Difference**
Learning Fair Representation	0.0560	1.0896	-0.0523	-0.2151
Reweighing	0.3438	1.9949	0.1448	0.0412
Adversarial Debiasing	0.3302	2.0444	0.0556	0.0053
Exponentiated Gradient Reduction	0.3134	1.8698	-0.0218	-0.0439
Calibrated Equalized Odds	0.4488	2.7954	0.1543	0.1097
Reject Option based Classification	0.6397	2.7755	0.0769	0.4248

Table 6: Fairness metrics of 'Tuition_fees_up_to_date' variable

Algorithm	**Statistical Parity Difference**	**Disparate Impact Ratio**	**Equal Opportunity Difference**	**Average Odds Difference**
Learning Fair Representation	-0.5070	0.3162	0.0507	-0.0139
Reweighing	-0.5389	0.2497	-0.1144	-0.0858
Adversarial Debiasing	-0.5310	0.2143	-0.0668	-0.0646
Exponentiated Gradient Reduction	-0.5172	0.2647	-0.0218	0.0168
Calibrated Equalized Odds	-0.6779	0.0753	-0.4601	-0.3264
Reject Option based Classification	-0.4409	0.5496	-0.3333	-0.3663

References

[1] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning Fair Representations." International Conference on Machine Learning, 2013.

[2] F. Kamiran and T. Calders, "Data Preprocessing Techniques for Classification without Discrimination," Knowledge and Information Systems, 2012.

[3] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating Unwanted Biases with Adversarial Learning," AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society, 2018.

[4] A. Agarwal, A. Beygelzimer, M. Dudik, J. Langford, and H. Wallach, "A Reduction Approach to Fair Classification," International Conference on Machine Learning, 2018.

[5] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger, "On Fairness and Calibration," Conference on Neural Information Processing Systems, 2017.

[6] F. Kamiran, A. Karim, and X. Zhang, "Decision Theory for Discrimination-Aware Classification," IEEE International Conference on Data Mining, 2012.

[7] Lundberg, S. M., & Lee S.-I, "A Unified Approach to Interpreting Model Predictions", Journal of Machine Learning Research, 2017.

[8] Ribeiro, M. T., Singh, S., & Guestrin, C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier, 2016.

[9] Alexandra Chouldechova,Aaron Roth, "The Frontiers of Fairness in Machine Learning", 2018.

[10] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena,
Kristina Lerman, Aram Galstyan, USC-ISI, "A Survey on Bias and Fairness in Machine Learning", 2019

[11] Aileen Nielsen, “Practical Fairness: Achieving Bias and Secure Data Models”, O’REILLY, 2021.