ENSURING ETHICAL, TRANSPARENT, AND AUDITABLE USE OF EDUCATION DATA AND ALGORITHMS ON AUTOML

Kylie Griep

A Capstone Project Submitted to the
University of North Carolina Wilmington in Partial Fulfillment
of the Requirements for the Degree of
Master of Science

Department of Computer Science
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2023

Approved by

Advisory Committee

_____	_____
Karl Ricanek	Minoo Modaresnezhad

_____
Yang Song, Chair

Accepted By

_____
Dean, Graduate School

TABLE OF CONTENTS

ABSTRACT

Ensuring Ethical, Transparent, and Auditable Use of Education Data and Algorithms on AutoML. Griep, Kylie, 2023. Capstone Paper, University of North Carolina Wilmington.

Automated machine learning (AutoML) creates additional opportunities for newer users to build/test their data mining models. Even though AutoML creates the models for the user, there is still technical knowledge and tools needed to evaluate those models and due to the black-box nature, problems can arise about algorithmic biases and fairness. Biased models can lead to biased predictions and therefore an amplification of those biases in future applications. Before applying an AutoML model, researchers should follow certain frameworks to define fairness criteria, select fairness metrics, evaluate fairness across groups, and take actions to mitigate and address biases. To analyze the algorithmic bias within these models, different fairness metrics can be tested. However, fairness does not have a holistic definition — some fairness evaluation metrics are contradictory to others, and some metrics don't account for everything — which makes it difficult to pinpoint how to test for fairness. In this paper, ten fairness metrics were chosen, explored, and implemented on four AutoML tools, Vertex AI, AutoSklearn, AutoKeras, and PyCaret. We identified two open educational datasets and built both prediction and classification models on those AutoML frameworks. We report our work in evaluating different machine learning models created by AutoML and provide discussions about the challenges in evaluating fairness in those models and our effort to mitigate and resolve the problems of algorithmic bias in educational data mining. The goal of this project is to promote transparency, accountability, and the incorporation of fairness considerations throughout the educational data mining model development and deployment lifecycle.

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER 1: INTRODUCTION

Automated machine learning (AutoML) is the process of automating part of the trivial, time-consuming tasks in building machine learning pipelines [1]. The tasks that can be performed by AutoML include preprocessing the data, selecting suitable models, optimizing the model hyperparameters, and providing the predictions with the best model found [2]. In educational data mining (EDM) scenarios, when a data scientist is unavailable, or in countries where data scientists are scarce [3], AutoML can be a feasible solution for non-expert users to create reusable models for their prediction tasks effectively. Typical educational data mining tasks educators have utilized AutoML include predicting dropouts [3] and course grades [4].

Though AutoML can be a promising solution, educators and data scientists must be aware that any machine learning application can bring in biases inherent in the data itself [5], in the technology it is trained on [6], and in the way that AutoML picks the data processing and algorithms. The best models determined by AutoML are “black box models” for non-expert users. The unexamined biases can be a risk for students and institutions, especially if a biased model is saved and used in the future - it will perpetuate and amplify those biases in subsequent applications [7].

As an emerging AI technology being applied in education settings, researchers have suggested conducting in-depth and long-term studies on the ethical implications of AI in education [8]. The objective is to ensure that AI is used for positive purposes and to prevent its potentially harmful applications. One example of a fairness evaluation framework is the Fairness, Accountability, and Transparency (FAT) framework, which provides an end-to-end structured approach to assessing and addressing fairness concerns in machine learning models [9].

In this paper, we identified existing EDM problems and datasets and built AutoML models from different existing AutoML frameworks. In this process, we brought no domain knowledge to the data-processing phase and relied solely on AutoML to process the data and produce the best models. After that, we implemented several fairness evaluation metrics, so the AutoML models are evaluated on accuracy and fairness. The fairness evaluation was implemented in coding, but it can be done with office software that can manage tabular data with columns and rows as well. From our experiments, we reveal that AutoML frameworks can achieve results as accurately as experts; however, fairness could be a blind spot. To better conduct research/experiments using educational data, researchers - data mining experts and non-experts - can and should also evaluate the fairness of the models.

CHAPTER 2: RELATED WORK

While investigating the space of EDM, several core outcomes receive highlighted attention. These linchpins focus on the output such as 1) implementing EDM detection of whether a student is going to pass or fail a certain course, 2) the prediction of students' final marks, or 3) the identification of students that are likely to drop out [10]. These goals, which benefit both students and educational institutions, are to gain useful insight so that any student who may require proper counseling, encouragement, or remediation, may receive it. When looking into such topics, there is also a byproduct that guides professionals in readjusting curricular material and how it is conveyed to the students.

Considering the EDM field is vast, we focus specifically on the application of AutoML in EDM and the discussions of fairness/algorithmic biases in EDM models

AutoML in EDM

In the early phase of EDM as a young research domain, Romero and Ventura already started advocating for EDM tools that are "easy to use by educators or not expert users in data mining [11]." In addition, before the wide adoption of neural network-based models, researchers expected "parameter-free data mining algorithms" to be the solution for non-expert users [11]. Arguably, this is also what AutoML provides - even for neural networks that have a huge number of parameters [12], AutoML frameworks such as AutoKeras encapsulate them in the AutoML framework to simplify the configuration and execution.

In 2020, Tsiakmaki et al. published their work on using AutoML to reduce human efforts and inputs. It was being applied to decipher some of the complexity and dimensionality to properly address the topics at hand. Juxtaposed to classic EDM methods, a dataset containing student information that was derived from a custom

Moodle plugin was used to develop regression and classifiers to identify students that pass/fail, predict students' academic performance, and the prediction of dropouts. Capitalizing on evaluation metrics mean absolute error and accuracy, the result yielded from six traditional classifiers (Bayes classifiers, rule-based, tree-based, function-based, lazy, and meta classifiers) then compared them to Auto-WEKA, a Sequential Model-based Algorithm Configuration, provided consistent emphasis that "tools like Auto-WEKA can help people in education—both experts and novices in the field of data science [10]." In many cases, the automated machine learning model and its established hyperparameter optimization showed major t-test-supported improvements [13]. There was no overall method that performed best or significantly better than the others when comparing the approaches in each distinguished dataset, but the work of Tsiakmaki et al. highlights that tree-based classifiers are the proposed configuration in most cases.

In some cases, AutoML can be viewed as a more appropriate approach to some challenges because not only does it provide a suitable solution, but some techniques can also deliver an auto-adjusted parameter/feature set that fits optimally for the task at hand [14]. In the EU, research conducted by Garmpis et al. focused on providing insights into student performances such as dropping out or delayed graduation. Using a dataset of more than 20,000 students collected from two courses over a couple of decades, the work considered standard classification metrics like accuracy and recall measuring machine learning techniques. One of the key understandings derived from their work was that "AutoML provides us not only with the most robust model and the optimal associated variables but also with the most informative features from the initial dataset [15]." Their procedures could allow some universities to offer summer courses and reinforce teaching strategies in hopes of reducing the amount of time it takes for students to graduate. The

effects of this could also be observed through the financial burdens required for studies [16] and even in the labor market.

Fairness in EDM

In the process of reviewing the related work on fairness in educational data mining applications in the educational domain, two major agreements have been mentioned by researchers - 1) there is no holistic definition of fairness, instead, fairness can be defined in many different ways [17]; 2) some fairness evaluation metrics are contradictory to each other, thereby it is almost impossible to create a model that can achieve high fairness across many metrics at the same time [18].

In 2019, Loukina et al. examined fairness in an English exam using different definitions of fairness [17]. The potential bias from human raters could originate from test-takers' native language - even with the same level of English proficiency, the native language speaker of a certain language may receive a biased score. The authors believed a fair automated scoring system should not "introduce additional construct-irrelevant group-related variance or disadvantage any group of test-takers" [19]. The dataset was from an English language test that required test-takers to record their short verbal responses (45 seconds or one minute) to six questions. The authors selected test-takers who had Chinese, Korean, Japanese, Spanish, Arabic, and German as their native languages and sampled 26,710 responses. To evaluate the fairness of the rating system, the authors used standardized mean differences and proved that for certain groups of test-takers (for example, Japanese or German native speakers), the grading system was more likely to either over-grade or under-grade them. To evaluate the fairness, the author used 1) squared error, 2) absolute error, and 3) conditional procedure equality. The authors further demonstrated that using a pool of first language-specific machine learning models

to predict the scores of a test-taker based on his/her native language is the least fair approach (a similar approach was also discussed by Hardt et al. in 2016 [20]). By contrast, the best model they built only used linguistic features that did not have a significant variance (smaller than 3%) attributed to the native language of the test takers. As described in Hu and Rangwala's work in 2020 [21], group fairness and individual fairness are umbrella terms used to classify approaches when discussing aspects of fairness. Group fairness promotes that different groups are "treated fairly as a whole," which might not be fair to some individuals. Thus, they proposed individual fairness to make sure fairness is "achieved on an individual level [22]." In their case study, the focus was identifying at-risk students through individual fairness. Evaluating datasets collected from George Mason University and the machine learning models were evaluated using metrics of accuracy, consistency, and discrimination. The study showed that there is room for improvement when discussing fairness in educational data mining, and they suggested that "future work on fairness in educational data mining should design course-specific models [21]." It exemplified that on a university level, male and African American Students are biased against, but on a course level, the bias points in a different direction. Though focused on individual fairness, the study supported that, in some cases, individual fairness can increase group fairness.

CHAPTER 3: ALGORITHMIC BIAS EVALUATION

Fairness can mathematically be determined in multiple ways based on the metric used. Within this section, ten fairness metrics are discussed along with how to determine which should be applied based on the model trained. The fairness metrics fall under three categories: Parity-Based Metrics, Confusion Matrix-Based Metrics, and Score-Based Metrics. Based on those metrics, the 4/5th rule is applied, which results in the trained model passing or failing the fairness metric.

In Caton and Haas's definitions, fairness evaluation equations are shown as being equal to satisfactory/passing [23]. However, there is a meager chance for every group to be exactly equal. To combat this, the 4/5ths rule can be applied [24]. This is not a universal rule; however, it can be used as a baseline in determining fairness [24]. In each fairness evaluation metric, the equation is applied to each subgroup, resulting in a set of values. The minimum subgroup value is compared to 4/5ths of the maximum subgroup value. If the minimum subgroup value is equal to or higher, the 4/5ths rule passes.

In the sections below, we define there is a machine learning model that has y as the ground truth of the target value (for example, a student passes or fails a course, or the final letter a student receives), and $\hat{y}$ is the predicted label. Based on a certain protected attribute, there will be different subgroups g_1 to g_n in which n is the number of possible values on the given protected attribute. In most of the equations based on classification in this section, we assume it is a binary classification problem in which y can be 0 or 1. But we will also provide the definition for multiple-label classification with possible labels l_1 to l_m, in which m is the number of possible labels in this problem. In a prediction problem in which real values are predicted, we use y as the ground truth, and $\hat{y}$ is the predicted value.

Parity-based Metrics

Parity-based metrics will typically look at the comparison of predicted positive values and any variants of that for classification models [23]. For the parity-based metrics section, the statistical demographic parity and disparate impact were looked at.

Statistical Demographic Parity

The Statistical/Demographic Parity defines fairness as "an equal probability of being classified with the positive label" [23]. This can be determined by looking at the actual value that is being predicted in the model. Calculate the total number of actual positives in a specific subgroup and divide that by the total number of samples to get the probability of a label being classified as positive. All subgroups are then compared to other subgroups, where the minimum and maximum are found, and the model will be passed or failed based on the 4/5th rule. The disadvantage of this metric is it does not consider any differences between groups [23]. The equation is shown below:

$$\min(\Pr(\hat{y} = 1 | g_i)) \geq 4/5 * \max(\Pr(\hat{y} = 1 | g_j))$$

Disparate Impact

The Disparate Impact also looks at the classification of being positive. The positive label percentages are determined in the same manner as in Statistical/Demographic Parity; however, the comparison of the groups is different. Looking at any possible combination of two subgroups, divide the smaller positive label percentage by the bigger one. If all the ratios are higher than 80% (0.8), the model is considered fair, if not, it is unfair. The equation is shown below [23]:

$$\forall_{i,j \in [1, n]} (\Pr(\hat{y} = 1 \mid g_i) / \Pr(\hat{y} = 1 \mid g_j)) \geq 0.8$$

Confusion Matrix-based Metrics

Confusion Matrix-based Metrics are used for classification models and consider four rates: True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), and False Negative Rate (FNR) [24]. Looking at these four rates helps "to include underlying differences between groups who would otherwise not be included in the parity-based approaches" [23]. For certain Confusion Matrix-based Metrics, the matrix can be binary, looking at the overall TPR, TNR, FPR, and FNR of all values combined. These metrics are Equal Opportunity, Equalized Odds, Treatment Equality, and Equalizing Disincentives. Others can be determined based on an n-by-n matrix that will compare the predicted value and actual value based on each different predicted/actual value. These metrics are Overall Accuracy Equality and Conditional Use Accuracy Equality.

Rates:

There are four rates to be considered: True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), and False Negative Rate (FNR). Each row is looked at and put into one of the following categories. Once all samples are examined, a percentage is calculated based on how many values fall in each category divided by the total number of rows:

- The TPR checks if the predicted value is equal to the actual value and both values are positive. Ex: predicted = pass AND actual = pass.
- The TNR checks if the predicted value is equal to the actual value and if both values are negative. Ex: predicted = fail AND actual = fail.
- The FPR checks if the predicted value is NOT equal to the actual value and the predicted value is positive while the actual value is negative. Ex: predicted = pass AND actual = fail.

- The FNR checks if the predicted value is NOT equal to the actual value and the predicted value is negative while the actual value is positive. Ex: predicted = fail AND actual = pass.

These rates can then be used in the confusion matrix-based metrics below.

Equal Opportunity

Equal Opportunity looks at the TPR between groups. Each subgroup will have a TPR, and they will be compared using the 4/5ths rule to determine if the model is fair. The equation is shown below [23]:

$$\min(\Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_i)) \geq 4/5 * \max(\Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_j))$$

Equalized Odds

Equalized Odds look at the TPR and FPR between groups. Each group's TPR will be compared using the 4/5ths rule, and each group's FPR will be compared using the 4/5ths rule. If both pass, the model is fair. The equation is shown below:

$$\min(\Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_i)) \geq 4/5 * \max(\Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_j))$$

$$\& \ \min(\Pr(\hat{y} = 1 \mid y = 0 \ \& \ g_i)) \geq 4/5 * \max(\Pr(\hat{y} = 1 \mid y = 0 \ \& \ g_j))$$

Overall Accuracy Equality

Overall Accuracy Equality looks at the overall correct predictions, which are determined by adding the TPR and the TNR. This metric looks at an n-by-n matrix and is calculated by adding all the cells that have an equal predicted and actual value (the diagonal) and dividing it by the total value of the matrix. Once the values are added, the added value for each subgroup is compared using the 4/5ths rule to determine if the model is fair. The equation for binary classification is shown below [23]:

$$\min(\Pr(\hat{y} = 0 \mid y = 0 \ \& \ g_i) + \Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_i)) \geq$$

$$4/5 * \max(\Pr(\hat{y} = 0 \mid y = 0 \ \& \ g_j) + \Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_j))$$

If the confusion matrix is n-by-n in a multiple-label classification scenario, then the equation can be written as:

$$\min(\Sigma_{k\in[1,m]} \Pr(\hat{y} = l_k \mid y = l_k \,\&\, g_i)) \geq$$

$$4/5*\max(\Sigma_{k\in[1,m]} \Pr(\hat{y} = l_k \mid y = l_k \,\&\, g_i))$$

Conditional Use Accuracy Equality

Conditional Use Accuracy Equality branches off Overall Accuracy Equality by looking at the TPR and the TNR separately instead of together. This metric looks at the n-by-n matrix and is similar to Overall Accuracy Equality, however, it compares each individual cell in the diagonal instead of it as a total. Each group's individual cells will be compared using the 4/5ths rule, and each group's TNR will be compared using the 4/5ths rule. If all cells pass, the model is fair. The equation for binary classification is shown below [23]:

$$\min(\Pr(\hat{y} = 1 \mid y = 1 \,\&\, g_i)) \geq 4/5*\max(\Pr(\hat{y} = 1 \mid y = 1 \,\&\, g_j))$$

$$\&\ \min(\Pr(\hat{y} = 0 \mid y = 0 \,\&\, g_i)) \geq 4/5*\max(\Pr(\hat{y} = 0 \mid y = 0 \,\&\, g_j))$$

If the confusion matrix is n-by-n in a multiple-label classification scenario, then the equation can be written as:

$$\forall_{i,j \in [1,n],\, k\in[1,m]} \min(\Pr(\hat{y} = l_k \mid y = l_k \,\&\, g_i)) \geq$$

$$4/5*\max(\Pr(\hat{y} = l_k \mid y = l_k \,\&\, g_j)$$

Treatment Equality

Treatment Equality looks at the ratio between FPR and FNR, dividing the FPR by the FNR. Each ratio for each subgroup is compared using the 4/5ths rule to determine if it's fair. The equation is shown below:

$$\min(\Pr(\hat{y} = 1 \mid y = 0 \,\&\, g_i) / \Pr(\hat{y} = 0 \mid y = 1 \,\&\, g_i)) \geq$$

$$4/5*\max(\Pr(\hat{y} = 1 \mid y = 0 \,\&\, g_j) / \Pr(\hat{y} = 0 \mid y = 1 \,\&\, g_j))$$

Equalizing Disincentives

Equalizing Disincentives looks at the difference between TPR and FPR, subtracting FPR from TPR. Each ratio for each subgroup is compared using the 4/5ths rule to determine if it's fair. The equation is shown below:

$$\min(Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_i) - Pr(\hat{y} = 1 \mid y = 0 \ \& \ g_i)) \geq$$

$$4/5 * \max(Pr(\hat{y} = 1 \mid y = 1 \ \& \ g_j) - Pr(\hat{y} = 1 \mid y = 0 \ \& \ g_j))$$

Please also note that the TPR-FPR can be zero or a negative value in some cases. When this happens, our implementation of equalizing disincentives will return n/a.

Score-based Metrics

Score-based Metrics determine fairness for regression predictions over classification. The two metrics looked at are the Balance between Subgroups and the Difference in Squared Error.

Balance between Subgroups

The balance between Subgroups looks at "the expected predicted score" [23] for each subgroup. To calculate this, find the absolute value of the difference between the predicted value and the actual value of each sample. Using that difference, divide it by the total number of records in that subgroup to get a percentage. The 4/5ths rule is applied to those percentages to determine if it's fair. The equation is shown below:

$$\min(\sum(\text{abs}(\hat{y} - y)) \mid g_i) \geq 4/5 * \max(\sum(\text{abs}(\hat{y} - y)) \mid g_j)$$

Difference in Squared Error

Difference in Square Error measures the average squared error within each group. This measure is similar to the "overall score accuracy" from Loukina et al.[17].

$$\min(\sum(\hat{y} - y)^2 \mid g_i) \geq 4/5 * \max(\sum(\hat{y} - y)^2 \mid g_j)$$

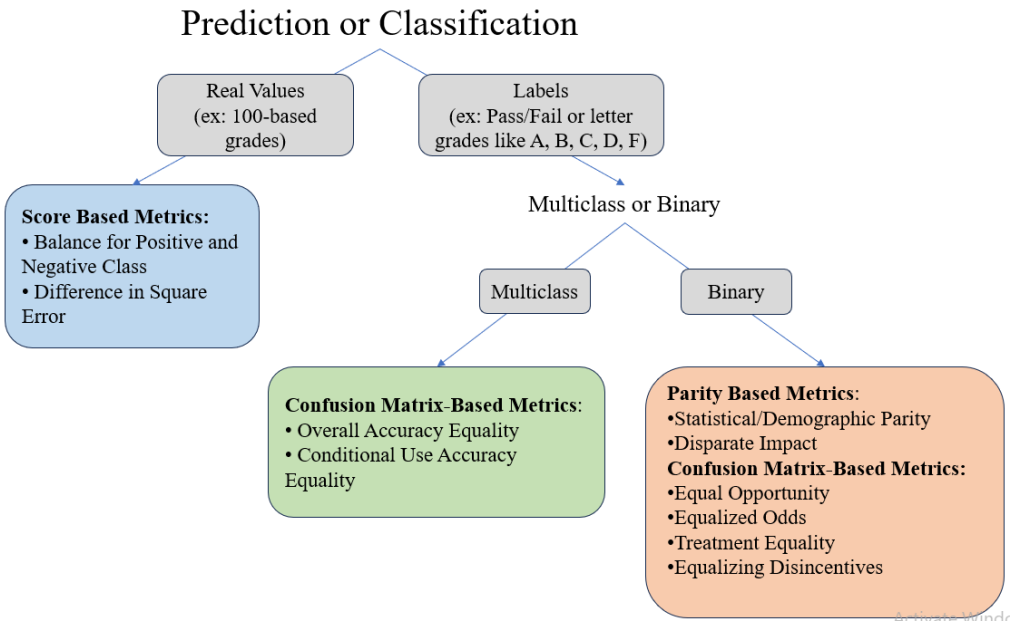

Figure 1. A taxonomy of different types of fairness evaluation metrics

To help EDM practitioners and researchers select fairness evaluation metrics, Figure 1 is a visualization of how to select the suitable fairness evaluation metrics discussed in this section.

All 10 of these fairness metrics were coded up in Python and put into a GitHub repository located here: https://github.com/kylie-g/fairness_metrics_package.git, where users can follow the instructions with their own datasets and predictions to test the fairness of their models.

CHAPTER 4: RESEARCH METHODS AND RESULTS

Research Goals

In this paper, there are two major research goals - 1) to create a package pluggable package of fairness evaluation metrics that can take the output from a given model (AutoML model from any framework or models built by experts) and evaluate the fairness, and 2) to demonstrate the fairness evaluation metrics on AutoML models build based on open educational datasets. The fairness evaluation metrics package can provide suggestions on whether a model can be accepted or rejected based on the fairness metrics of users' interests.

We also provide discussions on how different fairness metrics are related, for example, why one metric is stronger than another one, so educators and EDM researchers should understand there is no perfect fairness metric. Instead, they carefully examine the fairness evaluation metrics and identify a subset that they want to guarantee/examine on their tasks. There is also a further examination of the VertexAI AutoML models with these fairness metrics to see if retraining a new version of the model has any effect on the accuracy or fairness of the model.

It is worth mentioning that in this paper, we only discuss the algorithmic biases on protected attributes [25] and the attributes that are proxies for the protected attribute (e.g., zip code can be considered as a proxy for race [26] in a student admission system).

EDM Datasets and Tasks

The Portuguese student performance dataset is a popular dataset used in machine learning research for predicting student achievement in two distinct subjects: mathematics and the Portuguese language [27]. The mathematics dataset is slightly larger, with over 600 students and 30 variables related to their demographic information,

social and economic factors, and academic performance. In this paper, we only use the mathematics dataset because some AutoML frameworks require a minimum size of the dataset.

The Portuguese dataset can be used for different types of machine-learning tasks. For example, the dataset can be used for the course final letter classification, where the goal is to predict the grade that a student will receive in the course based on their data. In this case, the grades are grouped into five levels: A, B, C, D, or F. The five-level classification system is consistent with the original data contributor: 20-16 (A), 15-14 (B), 13-12 (C), 11-10 (D), and 9-0 (F).

Alternatively, it can be used for binary classification, where the goal is to predict whether a student will pass or fail the course based on their data. In this case, the grades are grouped into two categories: pass (A, B, or C) or fail (D or F), which were defined by the original data contributor from Portugal.

The Portugues mathematics dataset contains grades from three "periods" of the same school year each of which has a test score (in the dataset, they are "G1", "G2", and "G3"). In our model building, we used all the features except the grades from the second test ("G2") to predict the final course outcome, which was suggested by Cortez and Silva [27]. This is to simulate an at-risk student identification task after the first test in a semester.

Among the features in this dataset, we consider the "sex" column as a protected attribute, and another arguably protected attribute is "Pstatus" encoding the parent's cohabitation status (binary: 'T' - living together or 'A' - apart). Another popular dataset used for prediction tasks is the Open University Learning Analytics Dataset (OULAD) MOOC dataset created by the Open University in the United Kingdom and is publicly

available for research purposes [28]. This dataset contains data on over 30,000 students who enrolled in online courses offered by the Open University. It includes data on learners' demographic information, academic background, and performance in the course.

The OULAD MOOC dataset can be used for various prediction tasks, such as predicting student dropout rates, predicting their final grades, or identifying factors that contribute to student success in online courses.

We identify the protected attributes in the OULAD dataset as - "gender", "age_band", "imd_band" (Index of Multiple Deprivation band [29] of the place where the student lived), and "region" (the geographic region in the U.K. where the student lived). Overall, both datasets are valuable resources for EDM researchers interested in exploring various machine-learning tasks related to student achievement and success.

AutoML Framework

We tried four AutoML frameworks in our work – AutoSklearn [30], Vertex AI [31], AutoKeras [32], and PyCaret [33]. As the names suggest, AutoSklearn and AutoKeras are AutoML frameworks based on popular machine learning/deep learning modules Sklearn and Keras, with search algorithms built-in to support and accelerate the search to find the best model. Vertex AI (which used to be part of Google Cloud) is a web-based AutoML framework that supports building, deploying, and scaling machine learning models. PyCaret is an open-source library that contains multiple classifiers and regressors for rapidly choosing the best-performing algorithms. Table 1 provides a high-level comparison between those frameworks.

Table 1. A Comparison of The AutoML Frameworks

	Coding required?	Model produced	Process of redeploying models
AutoSklearn	Yes, minimal	ensemble	Can utilize pickle [34] to dump models for saving the models and for reuse of the dumped models.
Vertex AI (Google Cloud)	No	single model	Vertex AI provides GUI for this task.
AutoKeras	Yes, (somewhat) minimal	single model	Can use pickle or built-in function to save and load models.
PyCaret	Yes, minimal	single model	PyCaret has the model saving/loading functions built-in.

We built classifications and prediction models on the identified datasets with no additional attempts to apply our own understanding/domain knowledge in data preparation. For example, in the OULAD MOOC dataset, the "age_band" is an ordinal categorical column, and thereby the default one-hot encoding is not the best option. However, since we created the experiments to simulate the scenario of a non-expert building machine learning models with AutoML, we relied on the AutoML frameworks to process the raw data and for machine learning models. To make a fair comparison, when experimenting with different AutoML frameworks, we always held 25% of the samples for testing. However, Vertex AI required 1000 samples as the minimum input size, so we sampled the Portuguese dataset to meet this minimum data size requirement. Table 2 shows the performance of the best models from each AutoML framework. Using the Portuguese dataset, we trained five-level classification models that predict the students' final letters. Because the Portuguese math dataset is relatively small, we found that AutoKeras struggled with this dataset. We believe the reason is that the small tabular dataset is not suitable for neural network-based models. In addition, we also notice that AutoKeras is the AutoML framework that took the longest time to train, which is again because of the neural networks being used behind the scenes. In this task, AutoSklearn provides the best model with the highest Accuracy and F1.

We also built prediction models for students' scores with the OULAD MOOC dataset. Among the experimented AutoML frameworks, Vertex AI provides the best model with the highest r^2 score and lowest Mean Absolute Error (MAE). However, we also noticed that two AutoML frameworks provided models that produced negative r^2 scores, which means those two models are worse than predicting all the samples with the average score in the dataset. This is another example that the machine learning models made by AutoML frameworks should be examined before being applied in production.

Table 2. A Comparison of The AutoML Framework's Models

	Portuguese math classification (5 levels)		**MOOC prediction (scale 0-100)**	
	Accuracy	*F1*	*r^2*	*MAE*
AutoSklearn	**0.650**	**0.636**	-1.085	11.952
Vertex AI (Google Cloud)	0.633	0.515	**0.435**	**10.442**
AutoKeras	0.399	0.390	-14.167	14.756
PyCaret	0.552	0.552	0.181	13.216
Human expert	0.610	0.574	0.368	15.274

Fairness Results

The main goal of our paper is to develop the fairness evaluation metrics and test them on the models created by AutoML frameworks. For the classification models based on the Portuguese math dataset, we originally built the 5-level classification models, and we also converted the output to pass/fail binary output and applied the binary classification fairness evaluation metrics. Table 3 shows the results of the models from each AutoML framework that passed or failed each fairness evaluation metric.

From Table 3, we found that among the AutoML frameworks, Vertex AI created the model that passed the most fairness evaluation metrics. We also noticed that there are fairness evaluation metrics that were not passed by most of the models. Treatment Equality checks the ratio between FPR and FNR, which could range from 0 to infinity;

therefore, depending on the model and the dataset, it is one of the metrics that is harder to pass. Another challenging metric to pass is Equalizing Disincentives which checks the difference between TPR and FPR (subtracting TPR by FPR). This metric produced N/As when TPR is not larger than FPR.

Table 3. Fairness Evaluation Results of Classification Models (Binary) On the Portuguese Math Dataset

Protected Attribute	AutoML Platform	Statistics/Demographic Parity	Disparate Impact	Equal Opportunity	Equalized Odds	Treatment Equality	Equalizing Disincentives
sex	AutoSklearn	**Passed**	**Passed**	**Passed**	**Passed**	Failed	Failed
	Vertex AI	**Passed**	**Passed**	**Passed**	**Passed**	Failed	N/A
	AutoKeras	**Passed**	**Passed**	Failed	Failed	Failed	N/A
	PyCaret	**Passed**	**Passed**	Failed	Failed	Failed	Failed
Pstatus	AutoSklearn	Failed	Failed	Failed	Failed	Failed	N/A
	VertexAI	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	N/A
	AutoKeras	**Passed**	Failed	**Passed**	**Passed**	Failed	N/A
	PyCaret	**Passed**	**Passed**	**Passed**	**Passed**	Failed	Failed

Table 4 shows the fairness metrics of the five-level classification models from different AutoML frameworks. The two fairness evaluation metrics used, Overall Accuracy Equality and Conditional Use Accuracy Equality, are related because they both check the diagonal values in the n-by-n confusion matrix. Overall Accuracy Equality evaluates the sum of the diagonal value and checks the 4/5th rule; Conditional Use Accuracy Equality takes each pair of the diagonal value from two different confusion matrices and only returns a "pass" if the 4/5th rule holds between all the pairs. Therefore, from Table 4, it is not surprising for us to find that all the models can pass Overall Accuracy Equality; however, it was much harder to pass Conditional Use Accuracy Equality.

We built prediction models on the OULAD MOOC dataset and applied our scored-based fairness evaluation metrics on the protected attributes we identified earlier.

The results are listed in Table 5. Between the two fairness evaluation metrics, Balance between subgroups does not punish larger prediction errors as much as Differences in squared error, so for a model, it is usually easier to pass Balance between subgroups. By comparing the AutoML frameworks, we found that the model built by AutoSklearn can pass the most fairness evaluation metrics, followed by the model by AutoKeras.

Table 4. Fairness Evaluation Results of Classification Models (Five-Level) On Portuguese Math Dataset

Protected Attribute	**AutoML Platform**	**Overall Accuracy Equality**	**Conditional Use Accuracy Equality**
sex	AutoSklearn	**Passed**	Failed
	Vertex AI	**Passed**	Failed
	AutoKeras	**Passed**	Failed
	PyCaret	**Passed**	Failed
Pstatus	AutoSklearn	**Passed**	Failed
	VertexAI	**Passed**	Failed
	AutoKeras	**Passed**	Failed
	PyCaret	**Passed**	Failed

Table 5. Fairness Evaluation Results of Prediction Models on MOOC Dataset

	gender		**region**		**age_band**		**imd_band**	
	Balance between Subgroups	*Differences in Squared Error*	*Balance between Subgroups*	*Differences in Squared Error*	*Balance between Subgroups*	*Differences in Squared Error*	*Balance between Subgroups*	*Differences in Squared Error*
AutoSklearn	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed
VertexAI	**Passed**	Failed	**Passed**	Failed	Failed	Failed	**Passed**	Failed
AutoKeras	**Passed**	**Passed**	**Passed**	Failed	**Passed**	**Passed**	**Passed**	Failed
PyCaret	**Passed**	**Passed**	Failed	Failed	**Passed**	Failed	**Passed**	Failed

VertexAI's AutoML Model Version Comparison

VertexAI contains a feature that allows the user to retrain a new version of a previously trained model. To take advantage of this feature, a total of 10 versions were created from one trained model based on the MOOC dataset predicting final_result which produced 10 batch predictions based on each version. The versions VertexAI created did

not stem from the previous version but instead, all evolved from Version 1. Since VertexAI has a lot of black box features, it is not completely known how the process is done to create a new version of the model. Each model's versions were evaluated with F1 score, precision, and recall, which are shown in Table 6, and the predictions were evaluated on how they performed against the 8-fairness metrics using gender as the protected attribute (the two score-based metrics are being excluded since these are classification models).

Table 6. A Comparison of VertexAI's Model Versions on the MOOC Dataset

VertexAI Version	*F1 Score*	*Precision*	*Recall*
1	0.7997	87.3%	73.8%
2	0.5938	66%	54%
3	0.8701	90.9%	83.5%
4	0.6820	75.5%	62.1%
5	0.7298	80.1%	67%
6	0.6734	74.7%	61.3%
7	0.6378	70.6%	58.2%
8	0.6256	69.9%	56.6%
9	0.8312	88.7%	78.2%
10	0.8845	91.2%	85.9%

The F1 score, precision, and recall are all automatically calculated and displayed from each VertexAI model on the website, and that data is shown in Table 6. The F1 score is the average between precision and recall. From the data, we note that Versions 3 and 10 have the highest F1 scores while versions 2 and 8 are the lowest. As the versions were created, it doesn't seem like they progressively got better or worse with precision but instead fluctuated. Table 7 shows the results from the fairness metrics applied to each version.

Table 7. VertexAI Model Versions Results from Fairness Metrics

VertexAI Version (on gender)	Statistics/Demographic Parity	Disparate Impact	Equal Opportunity	Equalized Odds	Overall Accuracy Equality	Conditional Use Accuracy Equality	Treatment Equality	Equalizing Disincentives
1	**Passed**	**Passed**	**Passed**	Failed	**Passed**	Failed	Failed	Failed
2	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	**Passed**	Failed
3	**Passed**	**Passed**	**Passed**	Failed	**Passed**	**Passed**	**Passed**	**Passed**
4	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	Failed	Failed
5	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	Failed	Failed
6	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	Failed	Failed
7	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	Failed	Failed
8	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	**Passed**	Failed
9	**Passed**	**Passed**	**Passed**	Failed	**Passed**	Failed	Failed	Failed
10	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	**Passed**	Failed	**Passed**

Similar to previous models from other AutoML frameworks, none of the metrics were able to pass all across the board but Versions 3 and 10 were very close to passing 7/8. These results can be compared to the F1 score in Figure 2 below.

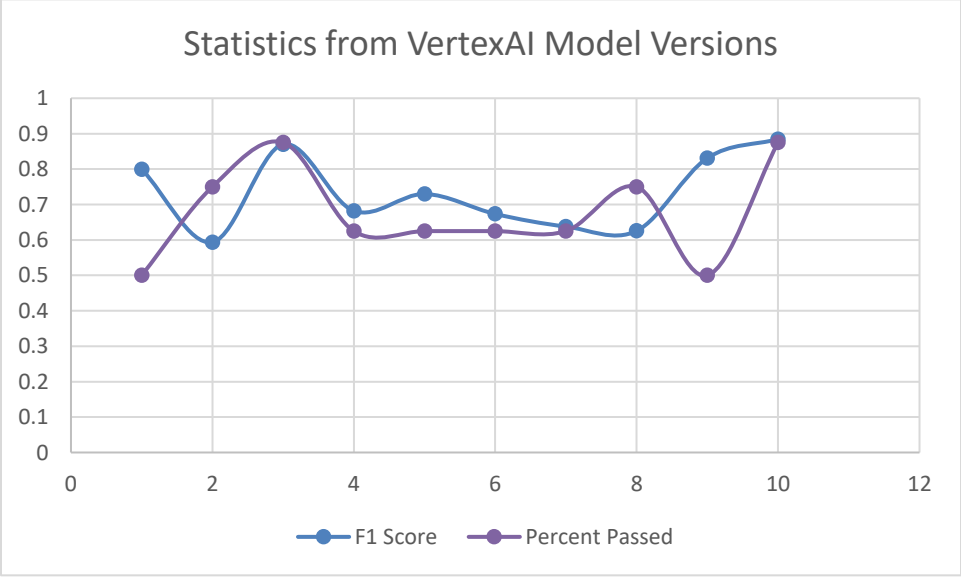

Figure 2. Statistics from VertexAI's Model Versions.

From Figure 2, it shows how the F1 score, and percent passed seemed to be a rough reflection of each other if a horizontal line was drawn at around 0.65 except for the two more precise models (3 and 10) which makes this statement untrue. Because of this, it is a bit unexpected that the two best models ended up passing the most fairness metrics, but this shows the capability of VertexAI to have a really fair model with really high accuracy.

Case Study – An Application of Fairness Evaluations

Based on the Python package created for fairness metrics mentioned above, it can be used in a real-life application involving two models developed by an undergraduate student at UNCW. These two models include a Random Forest Model and an SVM model, which analyze the OULAD dataset to build a classification model for predicting the final outcomes of students. These outcomes are categorized as Withdrawn (0), Fail (1), Pass (2), and Distinction (3). When examining binary classifications for Parity-based Metrics, such as Statistical/Demographic Parity and Disparate Impact, Withdrawn (0) and Fail (1) are considered as the *negative class*, while Pass (2) and Distinction (3) are categorized as the *positive class*. Additionally, Confusion Matrix-based Metrics were evaluated, while score-based metrics were not tested due to it being a classification model and not a regression model.

Four groups of protected attributes were analyzed, including gender, region, age_band, and imd_band, as shown in Table 8. It's important to note that the 'age_band' category '55>=' was *excluded* from the table due to only one record being predicted for it. When '55>=' is retained in the dataset, it leads to skewed results in the fairness metrics because it's either predicted correctly 1 out of 1 time or 0 out of 1 time. When the data is parsed and categorized into True Positive Rate (TPR), True Negative Rate (TNR), False

Positive Rate (FPR), and False Negative Rate (FNR), it results in a 1:0:0:0 split, causing biased comparisons between the highest and lowest values when applying the four-fifths rule. To obtain a more accurate assessment of fairness, the '55>=' variable in the 'age_band' group is omitted to eliminate this skew. This applies to any extremely small subgroup and should be adjusted accordingly.

Table 8. Random Forest and SVM Fairness Metric Results

		Statistical/demographic parity	Disparate impact	Equal opportunity	Equalized odds	Overall accuracy equality	Conditional use accuracy equality	Treatment equality	Equalizing disincentives
Random Forest	Gender	Passed	Passed	Passed	Failed	Passed	Failed	Failed	Failed
	Region	Failed	Failed	Failed	Failed	Failed	Failed	Failed	Failed
	Age_band (Removed 55<=)	Passed	Failed	Failed	Failed	Passed	Failed	Failed	Failed
	Imd_band	Failed	Failed	Failed	Failed	Failed	Failed	Failed	Failed
SVM	Gender	Passed	Passed	Passed	Failed	Passed	Failed	Failed	Failed
	Region	Failed	Failed	Failed	Failed	Failed	Failed	Failed	Failed
	Age_band (Removed 55<=)	Passed	Failed	Passed	Passed	Passed	Failed	Failed	Passed
	Imd_band	Failed	Failed	Failed	Failed	Passed	Failed	Failed	Failed

This case study demonstrates how fairness evaluation can be applied to various use cases and can provide recommendations based on the significance of protected attributes to the user and their preferences for fairness evaluation metrics. In this specific case study, gender may not be the most significant protected attribute, but there could be underlying discrimination related to gender within classrooms. Region may hold more significance because the quality of the education system varies across regions, and biases can emerge regarding students from better-performing education systems being more likely to achieve Pass or Distinction grades due to previous education. Age_band is less critical, but being a protected attribute, it still merits consideration. Imd_band is significant because it measures the poverty levels in different areas and can introduce biases such as students from more impoverished areas may not perform as well as their wealthier counterparts.

After looking at the results of the fairness metrics, due to the SVM model passing 4 more times than the Random Forest model, we can conclude that SVM is the fairer model. That in addition to the SVM having higher accuracy (SVM: 69.60% accuracy; Random Forest: 63.73% accuracy) makes it clear that SVM should be the model used between the two.

CHAPTER 5: DISCUSSION

In our work, we tried to examine the algorithmic biases in machine learning models and identify the models that are comparatively better in fairness. However, the bias may come from the data itself due to human/historical reason(s) [35]. So, is larger-scale data collection a solution to build fair machine learning models? We believe the answer is probably yes because a larger dataset is better, but the strategy of the data collection matters more than the volume itself [36].

From Tables 3 to 5, it is always good to have a machine learning model that can pass more fairness evaluation metrics, but researchers should identify the metrics that they want to examine first. Some fairness evaluation metrics are stronger and thereby harder to meet, so researchers need to make a decision on whether to pursue a model that passes those fairness metrics.

When tuning a machine learning model, it is important to start from a model that has a satisfactory overall accuracy and then turn it toward better fairness [37]. As researchers tune a machine learning model for fairness, the model is likely moving toward smoothing or over-smoothing. For example, a fair model in a prediction model is likely to predict each sample with an average target value in the dataset, which is a fair but useless model.

In our experiments, we noticed that modeling training/selection time is longer than most of us expected - training an AutoML model requires a long time to search for the appropriate models or neural network structures. For example, training a 5-class classification model with AutoSklearn takes more than 60 minutes on Google Collab (on a free Python 3 Google Compute Engine with more than 10 GB RAM and 75 GB storage space, GPU is not utilized) even the dataset itself has only 649 records.

We also found that coding and a certain level of technology literacy are still needed to use AutoML, therefore, it is not that friendly to non-technical users yet. To use AutoSklearn and PyCaret, the users still need minor coding. Here we use PyCaret, which is a less coding-intensive one as an example — the user still needs a computing environment that can execute the code (Figure 3 is a code snippet of the "minimum" coding needed for the Portuguese math score classification task). However, preparing and configuring such an environment, either on a local computer or on the cloud, is still a challenge to some smaller institutions or some non-technical educational units (for example, the department of foreign language, or the admission office in a small college) with limited staff and knowledge [38]. In addition, some online cloud computing platforms provide limited services (unless a user pays for a subscription) that makes the execution of AutoML difficult; for example, Anaconda Nucleus provides limited storage (100 MB with free access), and Google Collab has "idle timeout of 90 minutes and absolute timeout of 12 hours" [39].

```
!pip install --pre pycaret

from pycaret.classification import *
s = setup(student_df, target = 'G3',train_size=0.75)

best = compare_models()

save_model(best, 'my_best_pipeline')

loaded_model = load_model('my_best_pipeline')
print(loaded_model)
```

Figure 3. A code snippet of the "minimum" coding needed for the Portuguese math score classification task

Each fairness metric itself can be more appropriately used for certain situations over others. Figures 4 and 5 visualize TPR, TNR, FPR, and FNR within VertexAI's prediction model of the Portuguese dataset based on the group "sex". For ease of

example, the multi-class classification is converted into a binary classification of Pass (A, B, C) or Fail (D, F). The dotted line represents all the students that are predicted to pass (TPR and FPR) while all the students predicted to fail are below (TNR and FNR). The filled-in dots mean the actual value was a pass and the empty dots mean the actual value was a failure. Based on Figures 4 and 5, a better understanding can be formed of how each metric is applied. The values for this example are as follows: Male: TP=126, FP=160, TN=130, FN=116 & Total=532 and Female: TP=196, FP=164, TN=148, FN=258, & Total=766.

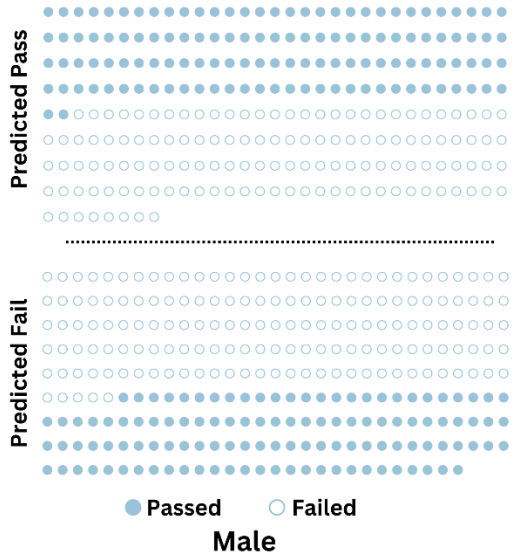

Figure 4. Vertex AI's prediction results based on the Portuguese Dataset for the Male group

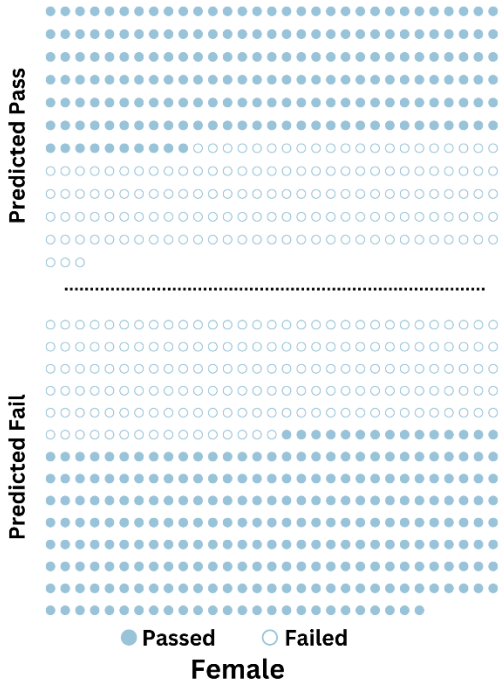

Figure 5. Vertex AI's prediction results based on the Portuguese Dataset for the Female group

Based on the visualizations, *Demographic/Statistical Parity* simply takes all the values above the dotted line (Predicted Pass) and divides them by the total number of values. For males, this would be 256/532 ≈ 0.481, and for females, this would be 344/766 ≈ 0.449. *Disparate Impact* builds off the *Demographic Parity* but divides the smaller group by the bigger group, passing if this value is higher than 0.8. Using the previous calculations, dividing female by male would result in 0.449/0.481≈ 0.933 which is greater than 0.8 and therefore would pass. These two metrics look at a very broad overview of what is predicted and not necessarily if these predictions are correct or not so that is something to consider when selecting this fairness metric to evaluate with.

Equal Opportunity will only look at the TPR from each group (the full circles above the dotted line). To calculate this, the full dots above the dotted line (predicted positive & actually positive) would be divided by the total number of full dots on the graph. For males, this would be 126/(126+116) ≈ 0.521, and for females, this would be

196/(196+258) ≈ 0.432. The problem that could arise from *Equal Opportunity* is since it doesn't account for FPR, in another scenario shown in Figure 6, the male predicted Passing results are altered to show how even if the true positive predicted values may be similar between males and females, the male group has many more false positive predictions that aren't being accounted for in this fairness metric. This is something to consider if using *Equal Opportunity*.

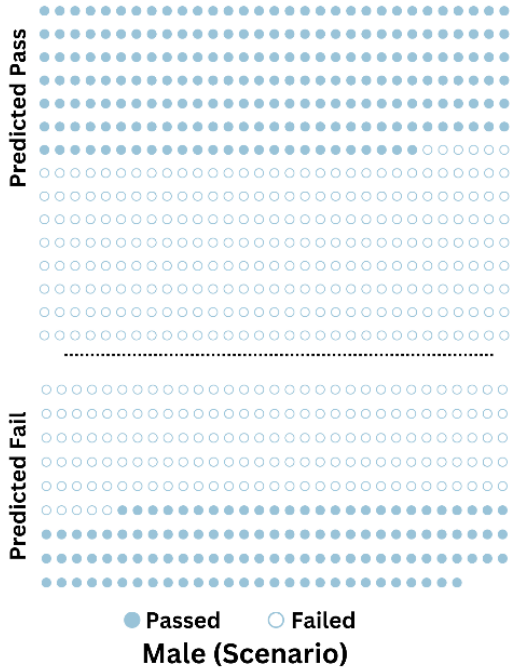

Figure 6. Vertex AI's prediction results in Scenario for the Male group

Equalized Odds, Overall Accuracy Equality, Conditional Use Accuracy, Treatment Equality, and *Equalizing Disincentives* all look at some combination of the rates. *Equalized Odds* is similar to *Equal Opportunity* in the sense that it looks at the positive predicted values that are above the dotted line but goes one step forward and in addition, will look at the empty circles (FPR). For both, it would be the same TPR calculated through *Equal Opportunity* (M: 0.521, F:0.432). The FPR would be calculated

by dividing the empty circles above the dotted line by the total number of empty circles. For males would be 130/(130+160) ≈ 0.448 and females would be 148/(148+164) ≈ 0.474. *Conditional Use Accuracy* is very similar to *Equalized Odds* except instead of looking at FPR, it will be looking at TNR alongside TPR. The main difference between the two is equalized odds looking at false positive vs conditional use accuracy looking at true negative. There are certain conditions when you would want to be more accurate with one over the other. For false positives, it may be important to look at contexts where the model prediction has high-stake consequences like if a false positive predicts a man guilty and he gets sent to jail for the next 50 years. This is where you want to minimize the differences in the False Positive Rate between each group. For True Negatives, it could be important to look at this if the context of your prediction is something where having true negatives is more important like in the medical context of predicting people who may have cancer. Those true negatives want to be prioritized because falsely predicting no cancer when there is some has a big impact. *Overall Accuracy Equality* is determined by looking at the True Positives and the True Negatives which can be calculated by adding the full circles above the dotted line and the empty circles below the dotted line to get the number predicted correctly and dividing by the total number of circles. For males would be 286/532 ≈ 0.538 and females would be 360/766 ≈ 0.470. *Treatment Equality* is FPR/FNR, where FPR is calculated above and FNR is calculated by dividing the full circles below the dotted line by the total number of full circles. For males would be 116/(116+126) ≈ 0.479 and females would be 258/(258+196) ≈ 0.568. Therefore, the male's *Treatment Equality* would be 0.448/0.479 ≈ 0.935 versus the female's *Treatment Equality* of 0.474/0.568 ≈ 0.83. Lastly, *Equalizing Disincentives* is another way to compare the values above the dotted line (predicted positive) because it is

TPR – FNR. Each fairness metric has different levels of strictness and flaws within them showing it is important to look at more than one metric to cover any holes in singular metrics. It is also important to evaluate what fairness metrics may be more beneficial regarding the model that is being created when applying them.

CHAPTER 6: CONCLUSION AND NEXT STEPS

Conclusion

Recent research has pointed out that examining fairness in EDM models should be common practice "for any person while creating an educational application that leverages [40]." For educational researchers and practitioners who include educational data mining in their research and workflow, it is relatively easy to add a fair evaluation in their EDM pipelines. However, with AutoML as an emerging EDM solution, we will see more EDM models built by non-technical users, and for those users, there are limited motivations, priorities, and practices of adding an additional step to evaluate the fairness in the AutoML models they created.

In our experiments, we built machine learning models with different AutoML frameworks and developed a collection of fairness evaluation metrics. There are both models for prediction and classification, and there were different fairness evaluation metrics we had developed that can evaluate the models of different types. Then we evaluated the fairness of the models from AutoML. We demonstrated the process of evaluating the fairness of machine learning models as one additional aspect researchers should visit to identify the models to utilize/deploy. The fairness metrics were made into a Python package for anyone to use and download to test their own prediction models whether created with AutoML or not. There was a dive into how VertexAI's feature to retrain models affects fairness, showing that VertexAI has the ability to create really fair and really accurate models based on the results. The metrics were also applied to a case study where a UNCW undergraduate student made two models and they were evaluated to see which one was fairer and therefore be used.

We hope our work reflects a proactive approach to the ethical and responsible use of AutoML in EDM, emphasizing the need to ensure transparency and address potential risks and societal implications. As more emerging AI tools will be on educators' radars, educators and researchers (including students who participate in EDM research) need to understand AI ethics, data privacy, security, and the potential implications of AI on human rights and gender equality.

The EDM-related legislation may arrive late. For example, the General Data Protection Regulation (GDPR) came into effect in 2018, but the call for data protection laws gained significant momentum in the early to mid-2000s in Europe as advancements in technology and the internet increased data collection, storage, and sharing practices [41].

Therefore, institutions should consider adjusting existing or adopting new regulatory frameworks. Since educational institutions are organizations that hold and utilize educational data, there is a need to adapt existing regulatory frameworks or create new ones specifically tailored to address the responsible development and usage of AI tools in education. This is to ensure that AI is deployed ethically and to mitigate any potential negative impacts.

Next Steps

Further updates and development upon the fairness metrics will be added when needed. As any new fairness metrics emerge, they will be added to the Python Package in further development. Other next steps may include further diving into what makes these models fair based off data about the models provided by these AutoML application since even during this experiment, VertexAI continues to expand and become less black-box.

REFERENCES

[1] J. Waring, C. Lindvall, and R. Umeton, “Automated machine learning: Review of the state-of-the-art and opportunities for healthcare,” Artif. Intell. Med., vol. 104, p. 101822, 2020.

[2] F. Hutter, L. Kotthoff, and J. Vanschoren, Automated machine learning: methods, systems, challenges. Springer Nature, 2019.

[3] G. Novillo Rangone, C. Pizarro, and G. Montejano, “Automation of an Educational Data Mining Model Applying Interpretable Machine Learning and Auto Machine Learning,” in Communication and Smart Technologies: Proceedings of ICOMTA 2021, Springer, 2022, pp. 22–30.

[4] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, “Fuzzy-based active learning for predicting student academic performance using autoML: a step-wise approach,” J. Comput. High. Educ., vol. 33, no. 3, pp. 635–667, 2021.

[5] A. Torralba and A. A. Efros, “Unbiased look at dataset bias,” in CVPR 2011, IEEE, 2011, pp. 1521–1528.

[6] D. Danks and A. J. London, “Algorithmic Bias in Autonomous Systems.,” in Ijcai, 2017, pp. 4691–4697.

[7] M. D. Rozier, K. K. Patel, and D. A. Cross, “Electronic health records as biased tools or tools against bias: a conceptual model,” Milbank Q., vol. 100, no. 1, pp. 134–150, 2022.

[8] J. Borenstein and A. Howard, “Emerging challenges in AI and the need for AI ethics education,” AI Ethics, vol. 1, pp. 61–65, 2021.

[9] K. Sokol, A. Hepburn, R. Poyiadzi, M. Clifford, R. Santos-Rodriguez, and P. Flach, “Fat forensics: A python toolbox for implementing and deploying fairness, accountability and transparency algorithms in predictive systems,” ArXiv Prepr. ArXiv220903805, 2022.

[10] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, “Implementing AutoML in educational data mining for prediction tasks,” Appl. Sci., vol. 10, no. 1, p. 90, 2019.

[11] C. Romero and S. Ventura, “Educational data mining: A survey from 1995 to 2005,” Expert Syst. Appl., vol. 33, no. 1, pp. 135–146, 2007.

[12] M. Khajah, R. V. Lindsey, and M. C. Mozer, “How deep is knowledge tracing?” ArXiv Prepr. ArXiv160402416, 2016.

[13] M. Helali, E. Mansour, I. Abdelaziz, J. Dolby, and K. Srinivas, “A scalable AutoML approach based on graph neural networks,” ArXiv Prepr. ArXiv211100083, 2021.

[14] N. Bosch, “AutoML feature engineering for student modeling yields high accuracy, but limited interpretability,” J. Educ. Data Min., vol. 13, no. 2, pp. 55–79, 2021.

[15] S. Garmpis, M. Maragoudakis, and A. Garmpis, “Assisting Educational Analytics

with AutoML Functionalities," Computers, vol. 11, no. 6, p. 97, 2022.

[16] J. Letkiewicz, H. Lim, S. Heckman, S. Bartholomae, J. J. Fox, and C. P. Montalto, "The path to graduation: Factors predicting on-time graduation rates," J. Coll. Stud. Retent. Res. Theory Pract., vol. 16, no. 3, pp. 351–371, 2014.

[17] A. Loukina, N. Madnani, and K. Zechner, "The many dimensions of algorithmic fairness in educational applications," in Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, 2019, pp. 1–10.

[18] J. Kleinberg, S. Mullainathan, and M. Raghavan, "Inherent trade-offs in the fair determination of risk scores," ArXiv Prepr. ArXiv160905807, 2016.

[19] R. Berk, H. Heidari, S. Jabbari, M. Kearns, and A. Roth, "Fairness in criminal justice risk assessments: The state of the art," Sociol. Methods Res., vol. 50, no. 1, pp. 3–44, 2021.

[20] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," Adv. Neural Inf. Process. Syst., vol. 29, 2016.

[21] Q. Hu and H. Rangwala, "Towards Fair Educational Data Mining: A Case Study on Detecting At-Risk Students.," Int. Educ. Data Min. Soc., 2020.

[22] Q. Hu and H. Rangwala, "Metric-free individual fairness with cooperative contextual bandits," in 2020 IEEE International Conference on Data Mining (ICDM), IEEE, 2020, pp. 182–191.

[23] S. Caton and C. Haas, "Fairness in Machine Learning: A Survey," ArXiv Prepr. ArXiv201004053, 2020.

[24] fairlearn, "Common fairness metrics." [Online]. Available: https://fairlearn.org/main/user_guide/assessment/common_fairness_metrics.html

[25] R. Yu, H. Lee, and R. F. Kizilcec, "Should college dropout prediction models include protected attributes?," in Proceedings of the eighth ACM conference on learning@ Scale, 2021, pp. 91–100.

[26] Y. Ritov, Y. Sun, and R. Zhao, "On conditional parity as a notion of non-discrimination in machine learning," ArXiv Prepr. ArXiv170608519, 2017.

[27] P. Cortez and A. M. G. Silva, "Using data mining to predict secondary school student performance," 2008.

[28] J. Kuzilek, M. Hlosta, and Z. Zdrahal, "Open university learning analytics dataset," Sci. Data, vol. 4, no. 1, pp. 1–8, 2017.

[29] H. Jordan, P. Roderick, and D. Martin, "The Index of Multiple Deprivation 2000 and accessibility effects on health," J. Epidemiol. Community Health, vol. 58, no. 3, pp. 250–257, 2004.

[30] M. Feurer, K. Eggensperger, S. Falkner, M. Lindauer, and F. Hutter, "Auto-sklearn 2.0: Hands-free automl via meta-learning," J. Mach. Learn. Res., vol. 23, no. 1, pp. 11936–11996, 2022.

[31] J. Katti, J. Agarwal, S. Bharata, S. Shinde, S. Mane, and V. Biradar, "University Admission Prediction Using Google Vertex AI," in 2022 First International

Conference on Artificial Intelligence Trends and Pattern Recognition (ICAITPR), IEEE, 2022, pp. 1–5.

[32] H. Jin, Q. Song, and X. Hu, “Auto-keras: An efficient neural architecture search system,” in Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019, pp. 1946–1956.

[33] U. R. Pol and T. U. Sawant, “Automl: Building An Classfication Model With Pycaret”.

[34] “11.1. pickle — Python object serialization — Python 2.7.18 documentation.” https://docs.python.org/2/library/pickle.html (accessed May 22, 2023).

[35] S. Barocas and A. D. Selbst, “Big data’s disparate impact,” Calif. Law Rev., pp. 671–732, 2016.

[36] T. Le Quy, A. Roy, V. Iosifidis, W. Zhang, and E. Ntoutsi, “A survey on datasets for fairness-aware machine learning,” Wiley Interdiscip. Rev. Data Min. Knowl. Discov., vol. 12, no. 3, p. e1452, 2022.

[37] F. Xiang, X. Zhang, J. Cui, M. Carlin, and Y. Song, “Algorithmic Bias in a Student Success Prediction Models: Two Case Studies,” in 2022 IEEE International Conference on Engineering, Technology & Education (TALE), Hongkong, China, Dec. 2020.

[38] W. R. Scull, M. A. Perkins, J. W. Carrier, and M. Barber, “Community college institutional researchers’ knowledge, experience, and perceptions of machine learning,” Community Coll. J. Res. Pract., vol. 47, no. 5, pp. 354–368, 2023.

[39] J. Bahan Pal, “ColabTricks.” https://jimut123.github.io/blogs/ML/ColabTricks.html (accessed May 22, 2023).

[40] G. Fenu, R. Galici, and M. Marras, “Experts’ View on Challenges and Needs for Fairness in Artificial Intelligence for Education,” in Artificial Intelligence in Education: 23rd International Conference, AIED 2022, Durham, UK, July 27–31, 2022, Proceedings, Part I, Springer, 2022, pp. 243–255.

[41] J. P. Albrecht, “How the GDPR will change the world,” Eur Data Prot Rev, vol. 2, p. 287, 2016.